Starbucks Capstone Challenge

Submitted by: Aman Chawla
Organization: NatWest Group
Course: Machine Learning Engineer

I. Definition

Project Overview

This project is an example of resolving classification problems using Machine Learning (ML). Here we have discussed how we can use Machine Learning technology to possibly increase the effectiveness of marketing campaign. Starbucks rolls out different kind of promotional offers in regular basis with the objective of increasing revenue. However in world of customization each customer has different needs and different preferences and react differently to these offers. A positive response from a customer is likely to motivate customer to do shopping whereas undesirable offers can be annoying experience for customer which is a negative experience for customer and undesirable experience for Starbucks. Therefore I have attempted to use ML to predict whether customer is likely to react positively to any particular offer or not.

Problem Statement

As Starbucks is concerned with enhances customer experience, Starbucks need to effectively manage marketing campaign it offers to its customer through its app.

When Starbucks advertise promotional offers to its customer, there are different ways in which customer can react.

- Ad can motivate the customer to make a purchase. This can be a positive experience for customer as customer is getting personalized experience.
- If offer is of no interest to customer, he/she is likely to ignore the ad. This is neutral experience for customer as ad has no impact on customer.
- If customer is advertised ads too many times where he/she is not interested, it can be frustrating experience and likely to demotivate customer to make a purchase. This is negative experience for customer and impact Starbucks in negative way.

Starbucks would want to send offer to its customer where a customer is possibly going to react in a positive way and would want to avoid sending any offer to customers who are likely to react in neutral or negative way. Starbucks would like to manage online offer program effectively for revenue increase and customer experience.

Metrics

In this project, I was provided with three datasets: customer dataset with customer demographics, portfolio customer with offer features and transcript dataset with customer transactions with Starbucks as well customer interactions with promotional offers. We started with analyzing each datasets and then understanding it. Major challenge was in reading transcript dataset with customer interactions with offer received. Lifecycle of customer is like this; customer receives the offer at some point of time, customer views the offer and if customer is interested offer is viewed and if offer is attractive enough customer completes the offer. Challenge was to link these three stages and then defining success criteria. I chose success criteria if customer views the offer and then completes it within offer period and average transaction amount in offer period is greater than that of overall period. This however posed another challenge by making data highly unbalanced which had some poor results with low model performance. Out of ~50k instances of customer receiving offers, there were only ~750 instances (1.5%) where offer had any success with the customer.

There was another critical dimension to this project which was choosing the right metric for model performance evaluation. Since data was highly unbalances with only 1.5% of final dataset having classification of successful offer. To maximize revenue, we want our model to predict those customers who are likely to make offer successful (High number of True Positive) whereas for sake of better customer experience, we also want to avoid sending offers to those customers who are less likely to respond to offer in desired way (high number of True Negative). In ML projects, high TP can be achieved by increasing sensitivity of the model and high TN can be achieved by increasing specificity of the model. Since both of these metrics works in opposite way which means one can be increased at the expense of other, we had to choose one as more important than other one. I selected sensitivity as most important metric for model evaluation as we want to send those offers to customers where customer is likely to respond to offer in desired way. We can afford some decrease in the specificity which would mean that our model is likely to recommend offers to our customer even though they are less likely to respond in desired way.

II. Analysis

Data Exploration

Here I am using dataset provided by Udacity.

There are 3 data files required for this project.

- portfolio.json - containing offer ids and metadata about each offer (duration, type, etc.)
    - There are 10 offers.
    - 2 are of type informational, 4 are of type BOGO and 4 are of type discount
- profile.json - demographic data for each customer
    - There are total of 17000 customer covered
    - 50% customers are male, 36% customers are female. Data seems to be missing for 13% customer
- transcript.json - records for transactions, offers received, offers viewed, and offers completed
    - There are 4 type of records in this table – ‘offer received’, ‘transaction’, ‘offer viewed’, ‘offer completed’.
    - There is a time variable which represent time at which record was generated.
    - Each offer is send to approx. 6000 customers which shows uniform distribution.
    - Distribution of persons for offer type shows that BOGO and Discount offer have ~14000 unique persons and information has ~10000 unique persons which is similar to distribution of offer type in portfolio dataset.

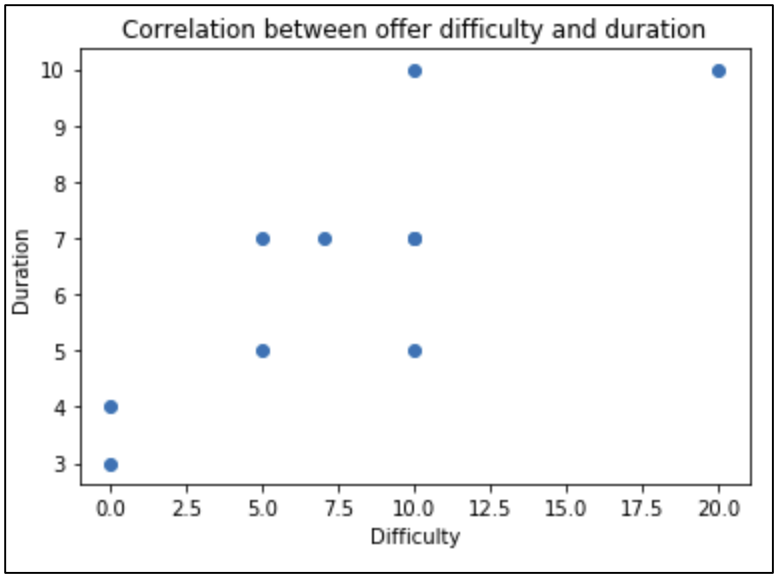
Exploratory Visualization

In this project, we are given three datasets and we would explore each data set one by one and then combine these datasets to create meaningful data with features which can be then used for modelling purpose.

Dataset 1: Portfolio

	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
3	[web, email, mobile]	5	7	9b98b8c7a33c4b65b9aebfe6a799e6d9	bogo	5
8	[web, email, mobile, social]	5	5	f19421c1d4aa40978ebb69ca19b0e20d	bogo	5
4	[web, email]	20	10	0b1e1539f2cc45b7b9fa7c272da2e1d7	discount	5
5	[web, email, mobile, social]	7	7	2298d6c36e964ae4a3e7e9706d1fb8c2	discount	3
6	[web, email, mobile, social]	10	10	fafdcd668e3743c1bb461111dcafc2a4	discount	2
9	[web, email, mobile]	10	7	2906b810c7d4411798c6938adc9daaa5	discount	2
2	[web, email, mobile]	0	4	3f207df678b143eea3cee63160fa8bed	informational	0
7	[email, mobile, social]	0	3	5a8bc65990b245e5a138643cd4eb9837	informational	0

- This dataset contain information of offers that Starbucks sends to its customer.
- There are 4 channel through which these offers are send to customer. Email is common to every offer.
- Other attributes of the offers are:
    - Difficulty – This is indicator of how easy or convenient offer can be availed by customer.
    - Duration – Duration for which offer is valid
    - Id - unique key of offer
    - Offer_type – Type of offer. Here we have 3 types of offer: buy one get one (BOGO), discount and informational
    - Reward – Benefit to customer on completing the offer
- Scatter plot between difficulty and reward shows some form of positive correlation. Since correlation is not perfect, we can try using both attributes simultaneously during model development.



Dataset 2: Profile

	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

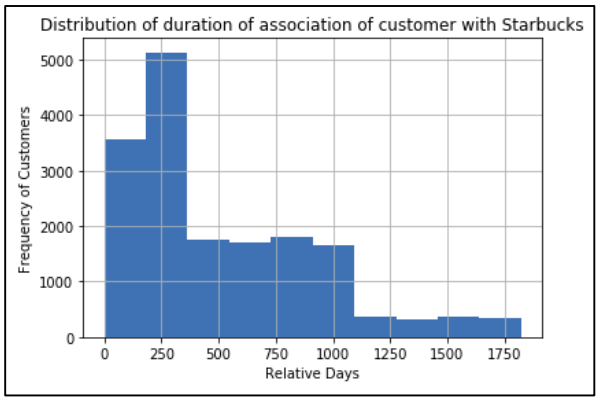
- Here we have demographic information of customer
  - Age
  - Become_member_on – data at which customer started relationship with Starbucks.
  - Gender
  - Income

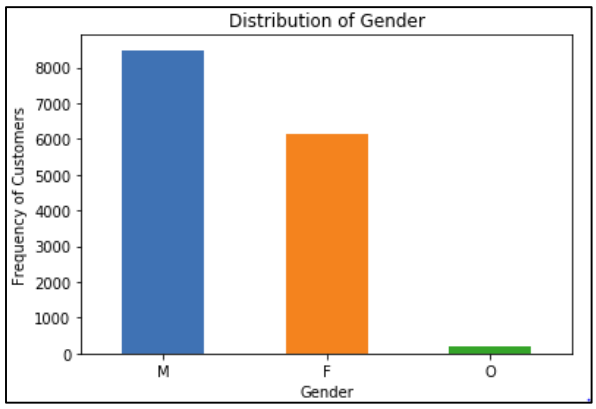
- There are total of 17000 customers whose data is available

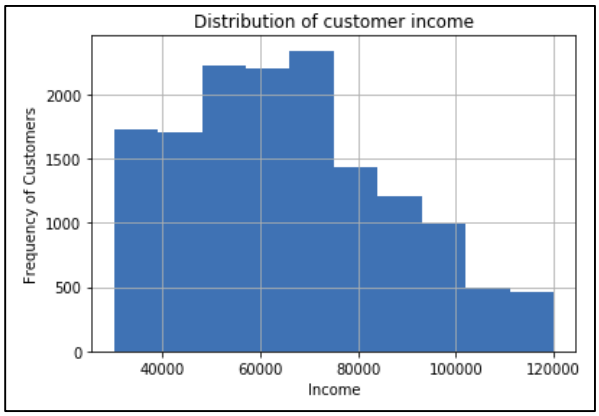
- Age is 118 where the age data is missing. In python code I checked that other demographic information is also missing for cases where age = 118. Data is missing for 2175 customers

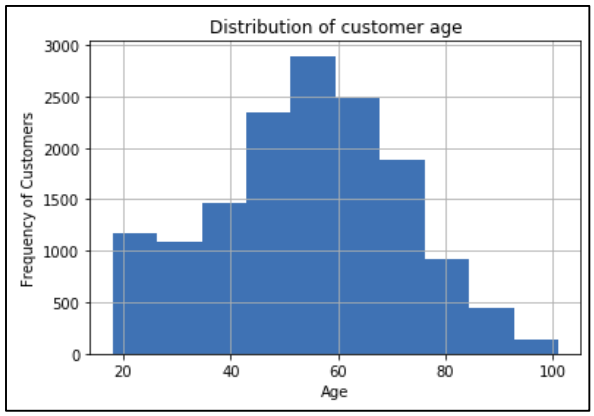
- There is one feature that we can pull from Become_member_on which for how long customer has been associated with Starbucks.

- Distribution plot of all 4 demographics (age, gender, income and duration of association with Starbucks) shows no outliers.









Dataset 3: Transcript

- Transcript contain data for 4 type of events
    - When Customers does a transaction

	event	person	time	value
12654	transaction	02c083884c7d45b39cc68e1314fec56c	0	{'amount': 0.8300000000000001}

    - When Customer receives an offer

	event	person	time	value
0	offer received	78afa995795e4d85b5d9ceeca43f5fef	0	{'offer id': '9b98b8c7a33c4b65b9aebfe6a799e6d9'}

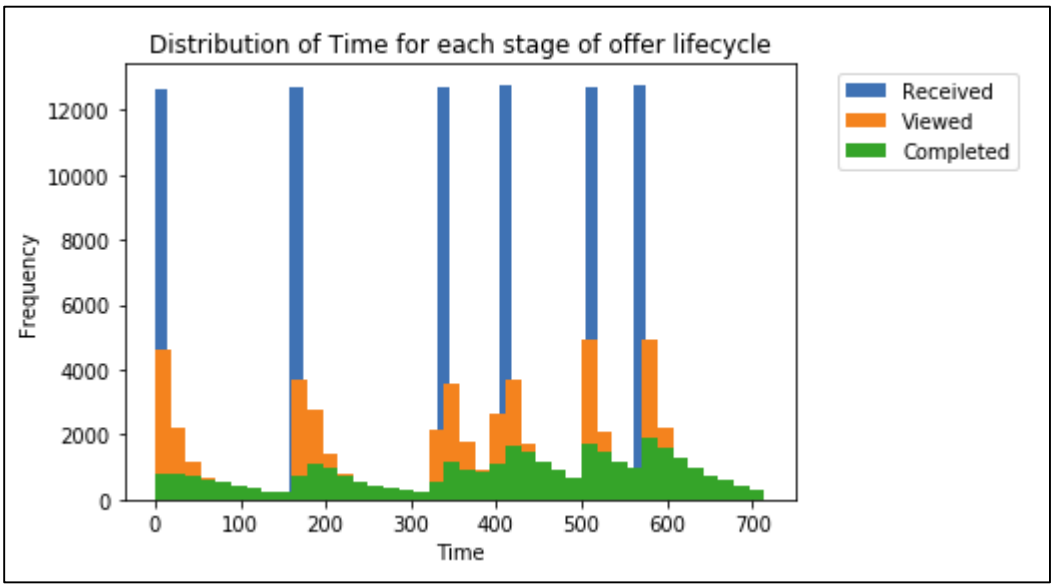
- When Customer views an offer

	event	person	time	value
12650	offer viewed	389bc3fa690240e798340f5a15918d5c	0	{'offer id': 'f19421c1d4aa40978ebb69ca19b0e20d'}

- When Customer completes the offer

	event	person	time	value
12658	offer completed	9fa9ae8f57894cc9a3b8a9bbe0fc1b2f	0	{'offer_id': '2906b810c7d4411798c6938adc9daaa5', 'reward': 2}

- Distribution Plot of 'Time' column for each stage of offer lifecycle with customer (receive, view and complete shows some interesting points



- Plot shows that Time for event 'offer received' has highest and straight bars which means that offer were rolled out 6 times during campaign period

- Subsequent stages – View and Complete has shorter peaks which means that only fraction of offer received are viewed and fraction of offer viewed are completed by customer.

- Important: For Data Cleaning and data preparation for modelling, it is important to link these 3 event data into single dataset to have complete understanding.

- I have linked ‘Offer Viewed’ and ‘Offer Completed’ transactions with ‘Offer Completed’ on logic if time is between ‘time at which offer is send’ and ‘time at which next offer is sent’.

Algorithms and Techniques

It seems we have very limited dataset to create a lot of features.

From profile dataset, we can only get 4 demographic features and from portfolio dataset as we can get 5 features.

From transaction transcript, we can get transaction related information but we again are clueless about time as it is arbitrary number and we are not able to get more insights from time variable as well.

From data I would like to create binary classification for combination of customer and offer and then predict whether customer is likely to respond to offer in positive way or not.

Since we would be dealing only with binary classification problem if customer should be offered with offer or now, we would like to use simple algorithms which can be used in binary classification problem.

I have used models that are generally used for binary classification problems such as RandomForest and Logistics Regression.

Logistics Regression was chosen as benchmark for comparing result with other models.

Benchmark Model

For benchmark, we can use a simple linear/logistics model.

III. Methodology

Data Pre-Processing

Data Pre-processing has been done at 6 stages

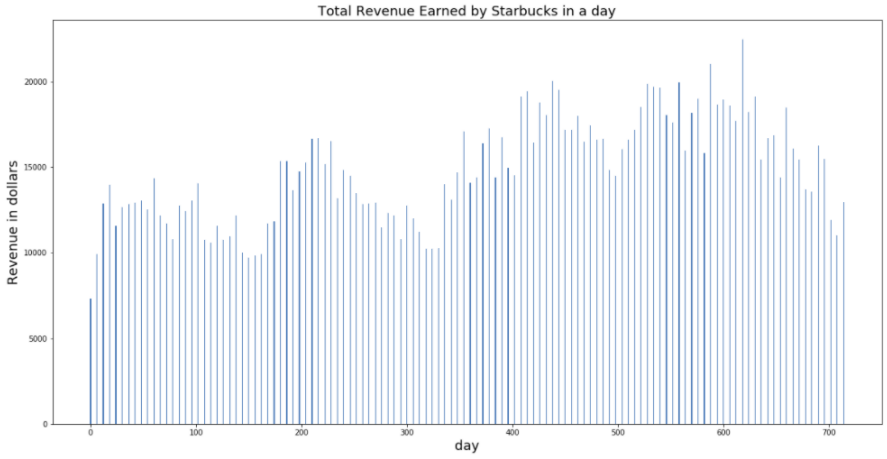
1. Processing Transactions data
2. Processing ‘Offer Received’ data
3. Processing ‘Offer Viewed’ data and linking it with ‘Offer Received’ data
4. Processing ‘Offer Completed data’ and linking it with already linked ‘Offer Received’ and ‘Offer Viewed’ to have single view of user interaction with customer
5. Linking single view with transactions data and defining success criteria for offer completed by customer. At this stage we have classifier label whether offer is successful or not.
6. Introducing features from Profile (customer features) and Portfolio (offer features) for complete dataset with features (X) and classifier label (y)

Stage 1: Processing Transactions data

- I created dataframe by name - transaction_transcript

- Since 'value' column contain the transaction amount, I created a new variable to store the amount.

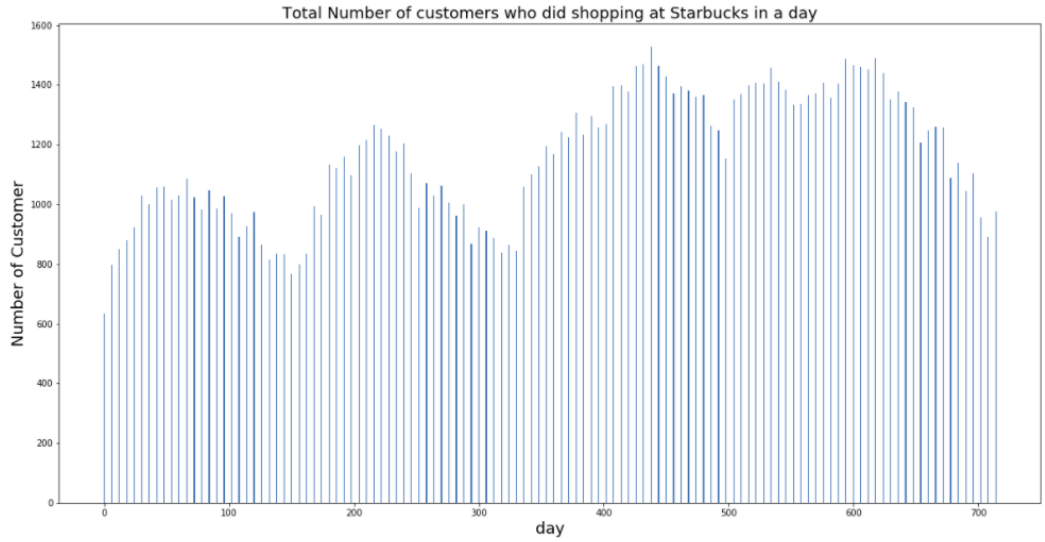
- I checked the min and max spent by all customers in a day and checked distribution of how much revenue Starbucks earned in a day.
  Max Amount spend in a day = $22,461.34
  Min Amount spend in a day = $7,305.57



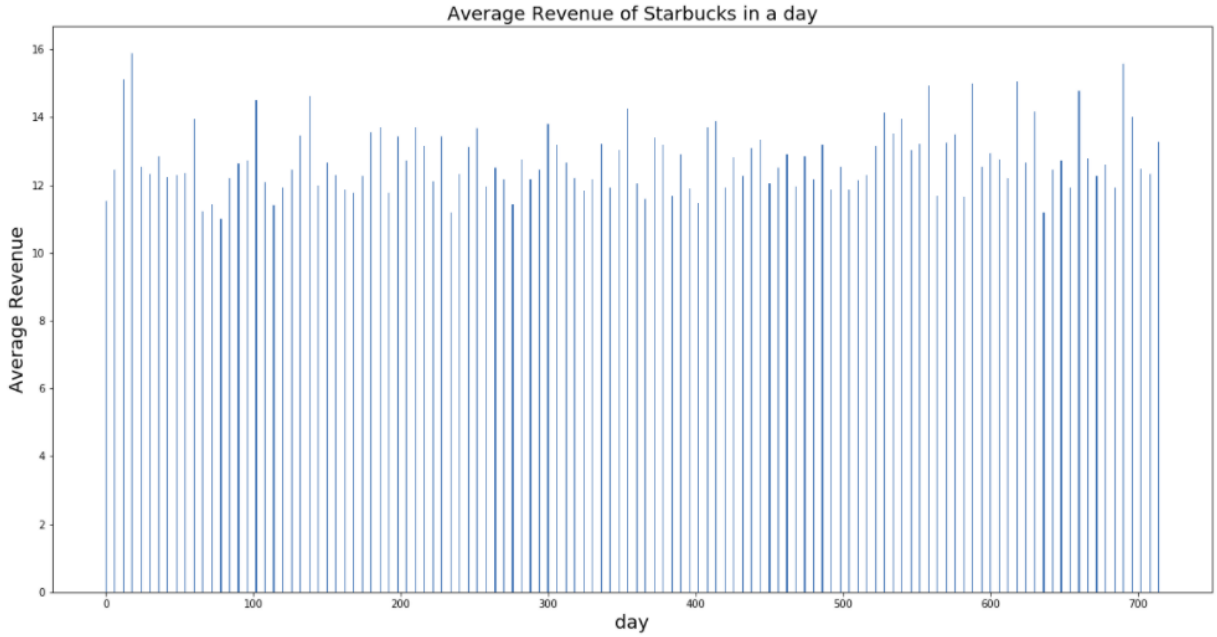
- One thing to notice was that customer spending was higher in later stage of campaign period. To further investigate reason for this trend, I checked the min

and max spent by all customers in a day and checked distribution of how much revenue Starbucks earned in a day

Max People who spend at Starbucks in a day = 1528
Min People who spend at Starbucks in a day = 633



- Since this also shows same trend, I plotted trend of average revenue per day which came out to be flatter trend. This means that higher spending was offset by more people doing shopping with Starbucks.



- I also checked if any customer did more than one transaction in a particular day and if there was need to consolidate data for each unit of time and person. I found out dataset had unique rows for combination of ‘person id’ and ‘time’ which means that there was no need of consolidation.

Stage 2: Processing 'Offer Received' data

- I created dataframe by name - offer_received_transcript.
- I merged portfolio data with dataframe - offer_received_transcript
- I checked that there were total of 16994 customers which is close to 17000 customers.
- I checked the unique value of 'time' column. Offer were sent to customer at fixed time only i.e. 0, 168, 336, 408, 504 and 576.
- I checked that at one point of time, one customer was offered only one product which means that we can fairly assume that customer would be under influence of offer till the time he/she is offered new offer.
- I checked how many times customer was offered with an offer

No of times customer was offered offer	No of Customers
1	73
2	610
3	2,325
4	4,988
5	5,931
6	3,067
Total	16,994

- I checked how many types of offer customer received

No of type of offer	No of Customers
1	1,019
2	8,460
3	7,515
Total	16,994

- Since customer could be offered one more offers more than once, I checked if at one point of time customer was offered more than one offers or not. It turned out that customer was offered only one product at one point of time. Turned out, at one point of time, only one offer was sent to customer which is good sign for joining data further.

Stage 3: Processing 'Offer Viewed' data and linking it with 'Offer Received' data

- I created dataframe by name – 'offer_viewed_transcript'.
- I checked that customer viewed only one offer at any point of time.
- I checked that no customer had more records in 'offer_viewed_transcript' as compared to 'offer_received_transcript'.
- I checked that 37% customers (6366) viewed all the offers they received. For remaining 62% customers, they missed to see the offer they received at least once. For such scenario where customer did not view the latest offer they received, it would be assumed that offer had no impact on buying pattern.
- I merged all rows of this dataset with 'offer_received_transcript' which can be used to see relation between how many offers were viewed out of all received offers.
- Output DF name - offer_received_viewed_transcript

Stage 4: Processing 'Offer Completed data' and linking it with already linked 'Offer Received' and 'Offer Viewed' to have single view of user interaction with customer

- I created the dataframe with name - offer_completed_transcript
- During analysis I found that this dataframe had duplicate records so I removed duplicates
- I also found that a customer could register 'offer completed' event multiple times. So in such scenario, I have only taken first instance.
- Output DF name - offer_received_viewed_completed_transcript

Stage 5: Linking single view with transactions data and defining success criteria for offer completed by customer. At this stage we have classifier label whether offer is successful or not.

- I created a flag which represent if a campaign has been completed or not sing criteria: offer is viewed and completed within offer validity period.
- I linked single view with transactions data using logic that time of transaction should be between two subsequent offer received timings.
- I defined a flag if transaction is between offer period or now
- Output DF Name - transaction_transcript_merged_complete

Stage 6: Introducing features from Profile (customer features) and Portfolio (offer features) for complete dataset with features (X) and classifier label (y)

- I calculated average transaction amount per customer.
- I calculated average transaction amount where transaction is within offer period and campaign has been completed.
- I created a binary classifier - offer_success as 1 where average transaction amount for offer period is greater than average transaction amount for entire period
- I merged Profile data and removed those records where customer demographic data is missing
- I merged Portfolio data and created dummy variable for categorical variables.
- Final dataset created for modelling purpose. Name - transaction_offer_person_relation_3

Implementation

After data processing, I had various features given in table

Feature Source	Feature
Profile (Customer)	Age, Income, Association Days and Gender
Portfolio (Offers)	Duration, Difficulty, Reward, Channel and Offer type
Transaction	Average Transaction Amount, Number of times customer did transaction during campaign period

I also had 2 classifier (label or output that would be predicted by model). I added figures of distribution as well.

- offer_success – scenario where customer views and completes the offer and transaction amount during offer period is higher than average transaction amount.

  Distribution

Value of offer_success	No of records	%age
0	48,873	98.48%
1	756	1.52%

- campaign_completed - where customer views and completes the offer.

  Distribution

Value of campaign_completed	No of records	%age
0	48,000	96.72%
1	1,629	3.28%

Using these, I tried to do model development.

Refinement

- I started with classifier - Campaign completed and Logistic regression for base lining.
- However I got unsatisfactory results.
- Then I tried using Random tree forest as well which also gave unsatisfactory results as our data has been highly unbalanced.
- Then I used prediction method – predict_proba which calculate probability and found out that probability is quite continuous in nature which makes model result highly sensitive
- I then introduced two new features from transactions dataset and then rerun all the models.
- I also repeated same process with other classifier - Offer Successful seeking better results.

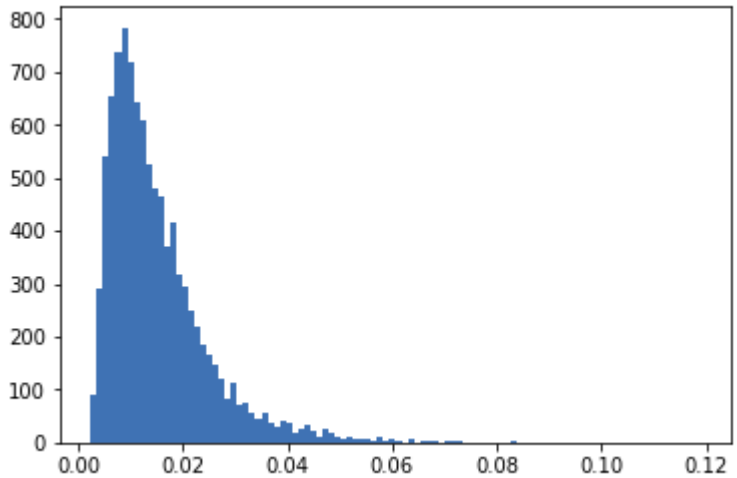
Here is summary of model that I ran.

Classifier	Algorithm	Result	Confusion Matrix	Output
offer_success	RandomForestClassifier	Sensitivity: 0.0 Specificity: 1.0	True Positives: 0 True Negatives: 9764 False Positives: 0 False Negatives: 162	Not acceptable
offer_success	LogisticRegression Using predict	Sensitivity: 0.0 Specificity: 1.0	True Positives: 0 True Negatives: 9764 False Positives: 0	Not acceptable

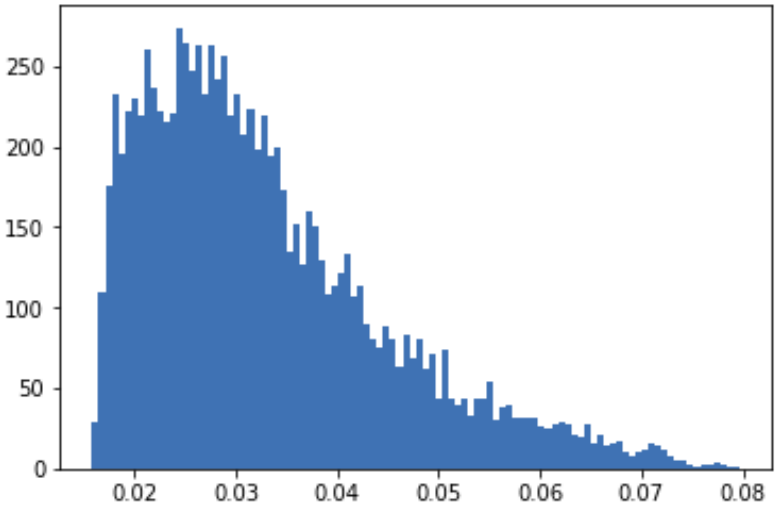
			False Negatives: 162	
offer_success	LogisticRegression By using predict_proba and adjusting threshold manually	Sensitivity: 0.84 Specificity: 0.35	True Positives: 136 True Negatives: 3463 False Positives: 6301 False Negatives: 26	Result are acceptable but very sensitive to threshold
campaign_completed	RandomForestClassifier	Sensitivity: 0.0 Specificity: 1.0	True Positives: 1 True Negatives: 9595 False Positives: 1 False Negatives: 329	Not acceptable
campaign_completed	LogisticRegression Using predict	Sensitivity: 0.0 Specificity: 1.0	True Positives: 0 True Negatives: 9596 False Positives: 0 False Negatives: 330	Not acceptable
campaign_completed	LogisticRegression By using predict_proba and adjusting threshold manually	Sensitivity: 0.84 Specificity: 0.35	True Positives: 136 True Negatives: 3463 False Positives: 6301 False Negatives: 26	Result are acceptable but very sensitive to threshold

Here is the probability distribution from our base model for both classifiers.

Probability Distribution of offer_success from Logistic regression



Probability Distribution of offer_success from Logistic regression



Such probability distribution makes ML model highly sensitive.

IV. Results

Model Evaluation and Validation

To understand how model performance was evaluated, let's understand the confusion matrix.

Actual Outcome	Model Prediction	Confusion Matrix	What it means for Starbucks (Interpretation)
Offer Successful	Offer Successful	True Positive	**Desirable Event** Offer is send to customer who is likely to respond in a positive way.
Offer Successful	Offer not Successful	False Negative	**Undesirable Event** Offer is not sent to customer who is likely to respond in positive way. It can lead to revenue loss as we are missing sending offer to customer who are likely to respond in positive way.

Offer not Successful	Offer Successful	False Positive	**Undesirable Event** Offer is send to customer who is not likely to respond in positive way. It can lead to poor customer experience as customer is receiving offers which are not required by customer.
Offer not Successful	Offer not Successful	True Negative	**Desirable Event** Offer is not send to customer who is not likely to respond in a positive way. It would lead to cost saving as Starbucks can reduce offer volume to be send to customer.

Confusion Matrix

		Model Prediction	
		Offer not Successful	Offer Successful
Actual Outcome	Offer not Successful	True Negative	False Positive
	Offer Successful	False Negative	True Positive

Since our highest priority is to maximize True Positive, we can choose Sensitivity or Recall as most important evaluation matrix.

Sensitivity = $\frac{\mathrm{TP}}{\mathrm{Tp+FN}}$

It means from all offers that are going to be successful, how many our model is able to correctly predict. High sensitivity is critical to success of marketing campaign.

Second most important criteria can be to have high True Negative as we want to avoid sending offers to customer who are not likely to respond in positive manner. For this we can choose Specificity as second most important criteria.

Specificity = $\frac{TN}{TN+FP}$

Specificity means out of all customers who are not likely to respond in positive way, how many model predict accurately.

Justification

So far we have got unsatisfactory results on using ML to do any prediction.

I used robust method of model evaluation which is using confusion matrix however we failed to gat impressive results.

This is because our data is highly unbalanced. Proportion of offer being successful or leading to increment in revenue is only 1.5% and offer completed within offer validity period is only 3%

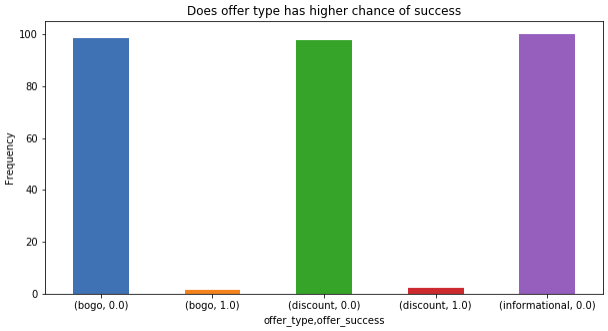
We could look at gathering mode insights on user such as what kind of products customer like at Starbucks and offer them product based offers rather than generic offers like BOGO or discount. Since Starbucks mostly run on brick and mortar model which means most of sales happen in Starbucks store, there are chances that online offer does not really motivate customer to visit Starbucks store.

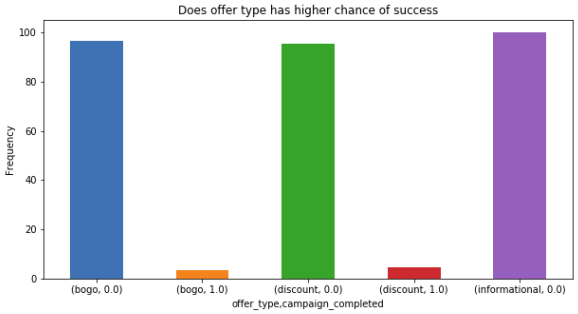
V. Conclusion

Free-Form Visualization

I tried to plot few of offers to see quality of dataset and of any feature has any significant impact on our chosen classifiers.

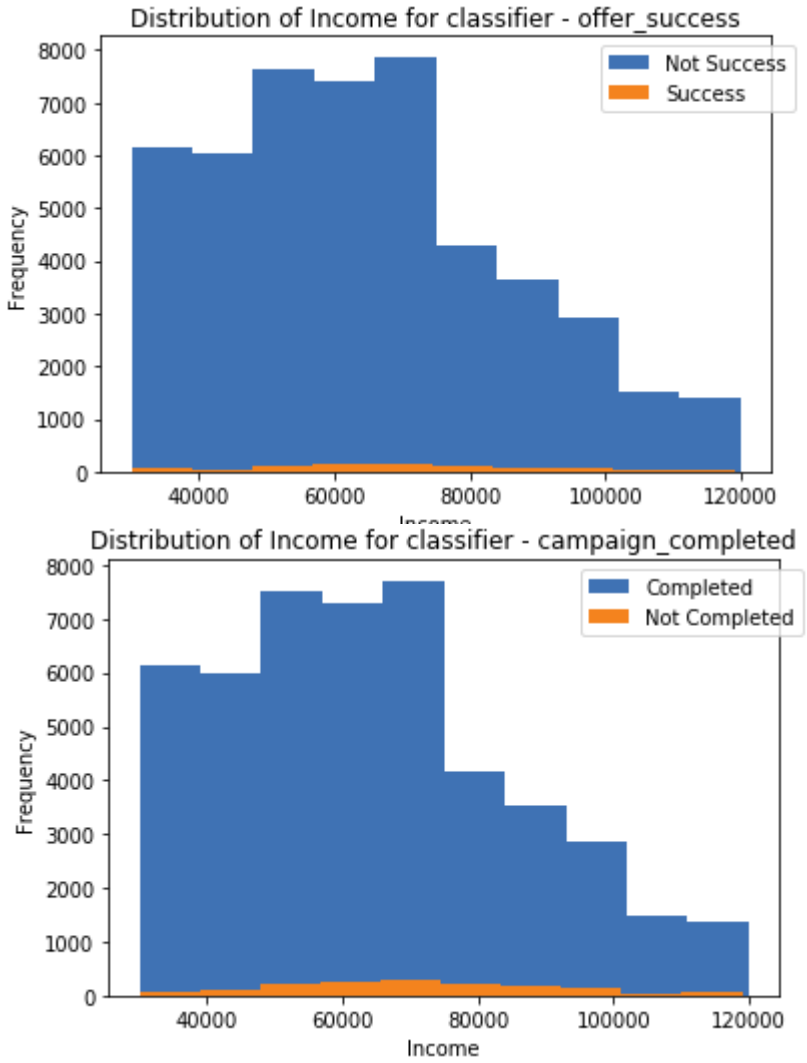
Q1. What type of offers have higher success chances?





Answer – As seen from graph, answer is No. All offer have very low success chances.

Q2. Does income has any impact on success chances?



Answer – Marginal impact.

Reflection

Challenge was getting wide variety of features which could improve model performance.

Another challenge was to create single view capturing the user journey with offer.

Improvement

We could look at gathering mode insights on user such as what kind of products customer like at Starbucks and offer them product based offers rather than generic offers like BOGO or discount. Since Starbucks mostly run on brick and mortar model which means most of sales happen in Starbucks store, there are chances that online offer does not really motivate customer to visit Starbucks store.

Another recommendation would be that Starbucks can take survey of customers who visit store to see if customer is really influenced by any offers or not. This would help Starbucks in 2 ways. One, if survey results say that customer does not really get annoyed with online offers, Starbucks can continue sending offers through online mode which would not impact customer experience. Second, if customer is not influenced by any offers, Starbucks can reduce amount of offer it sends online and save money as well resources on planning.