Maryrose Roque
Capstone Project for Udacity Nanodegree in Machine Learning Engineering
10 March 2023

Optimising Starbucks App Promotions Through User Segmentation

What can make Starbucks Promotions More Successful?

INTRODUCTION

Starbucks Corporation, the most popular coffee chain in the world, has an estimated revenue of over $26.5 billion (2019) and around 33,833 store locations in 70 countries (2021). Since the opening of the first Starbucks in 1971, its growth over the last 50 years clearly shows its success in fulfilling its mission to *"inspire and nurture the human spirit - one person, one cup, and one neighbourhood at a time."* (Starbucks Marketing Strategy. https://www.studysmarter.co.uk/explanations/business-studies/business-case-studies/starbucks-marketing-strategy/)

The success of Starbucks stems from its efficient use of its marketing mix, delivering the right product with the right price at the right place with the reinforcement of the right promotions delivered across various channels.

Starbucks is well-known for offering the right product due to its very uniform quality of food, beverage, and service. Enter any Starbucks establishment and you will always get what you expect from this trademark brand. This is probably one of the reasons behind its ability to foster long-term customer relationships and patronage.

In addition to the appeal of consistency, Starbucks is able to add a dose of uniqueness in the design of its store across different locations, like the Starbucks branch in Japan that resembles a traditional tea house or the one in Bangkok that resembles a Thai farmhouse. Aside from these cultural adaptations, Starbucks has a pricing system that draws in their target demographic in each country.

Starbucks has a very strong promotion strategy that do not only include traditional channels like billboards, newspapers, and magazines. It believes in using the word of mouth and the power of Instagram by encouraging its customers to post pictures of their products with the promise of rewards. It also encourages customer loyalty and purchases through the launch of Starbucks Rewards. Coupled with a mobile app that provides recommendations based on purchase history and a convenient "order and pay" system that allows customers to pay in advance before picking up their beverage, it is no wonder that Starbucks has captured a following of this magnitude.

Due to its intriguing success, the Starbucks Reward system has been chosen as the topic for the Udacity Nanodegree in Machine Learning Engineering Capstone Project. The data provided for this project simulates the offers sent out to the users of its mobile app as well as their response to these offers.

In the simulation, Starbucks sends out an offer to its users. This can be an informational advertisement about a new drink or product or an actual offer that allows the user to redeem a discount or a BOGO (a buy one get one free) promotion after purchasing a minimum amount within a set duration.

The objective of the study is to track patterns in customer response based on their demographic in order to optimise the distribution of Starbucks promotions by using completed orders as the metric of success.

STATEMENT OF THE PROBLEM

In the simulation of the Starbucks rewards mobile app used in this project, the user is sent an offer once every few days. This offer can merely be an informational advertisement about an ongoing Starbucks promotion or an actual offer that could either be a discount or a BOGO (buy one get one free).

Each offer comes with a minimum spend that entitles the user to the discount as well as a validity period before the offer expires. Not all users receive the same offer and some users may not receive an offer during certain weeks. It is also possible for a user to receive the same offer more than once.

It is possible for the user to complete an offer without viewing it. In this case, it can be assumed that the user would make a transaction even without an offer and it would be best to distribute the offers to the users who will be encouraged by an offer to make a purchase.

By combining transactional data and user demographic, the aim of this machine learning project is to : (1) predict whether the user will be converted by the offer or promotion, (2) identify the important factors that lead to conversion of the user, and (3) segment the users in order to effectively distribute the offers.

INITIAL EXPLORATION OF THE DATASETS PROVIDED

The data provided includes: (1) a portfolio that consists of the 10 offers that were sent to the customers, (2) the customer profile which includes details like age, gender, income, and date of membership, and (3) a transcript of the transactions and the times when the offers were received, viewed and completed.

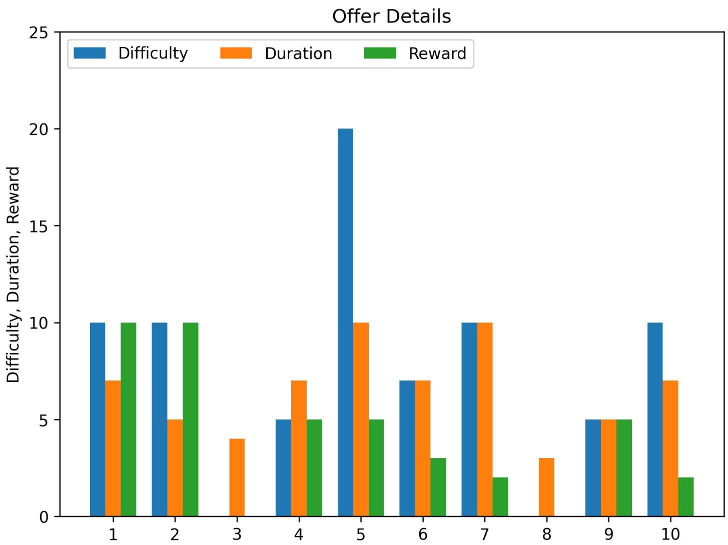
porfolio.json

The dataset contains offer ids and metadata about each offer:

- **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)** - identifies the channel/channels where the offer was sent (email, social, mobile, web)

	reward	channels	difficulty	duration	offer_type	id
0	10	[email, mobile, social]	10	7	bogo	ae264e3637204a6fb9bb56bc8210ddfd
1	10	[web, email, mobile, social]	10	5	bogo	4d5c57ea9a6940dd891ad53e9dbe8da0
2	0	[web, email, mobile]	0	4	informational	3f207df678b143eea3cee63160fa8bed
3	5	[web, email, mobile]	5	7	bogo	9b98b8c7a33c4b65b9aebfe6a799e6d9
4	5	[web, email]	20	10	discount	0b1e1539f2cc45b7b9fa7c272da2e1d7
5	3	[web, email, mobile, social]	7	7	discount	2298d6c36e964ae4a3e7e9706d1fb8c2
6	2	[web, email, mobile, social]	10	10	discount	fafdcd668e3743c1bb461111dcafc2a4
7	0	[email, mobile, social]	0	3	informational	5a8bc65990b245e5a138643cd4eb9837
8	5	[web, email, mobile, social]	5	5	bogo	f19421c1d4aa40978ebb69ca19b0e20d
9	2	[web, email, mobile]	10	7	discount	2906b810c7d4411798c6938adc9daaa5

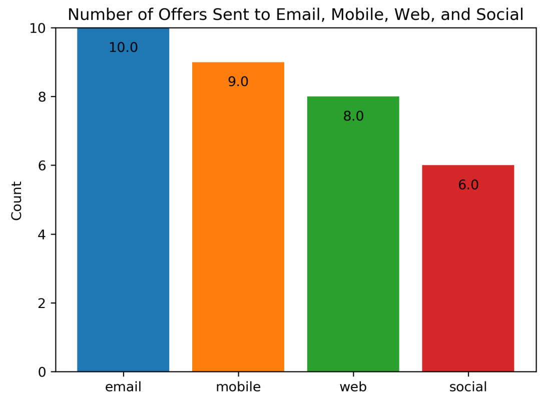
The bar graph below shows the levels of difficulty, duration, and reward for each offer. As the informational promotions merely contain advertisements that do not result to app rewards, they do not have values for difficulty or reward.



A comparison of the duration between BOGO and discount offers is that BOGO offers tend to have a lower difficulty level and duration compared to discount offers. Informational offers seem to have the lowest duration of validity.

One notable feature demonstrated by this graph is that the reward difficulty, reward, and duration levels are not usually proportional to each other. Offer 5, for example, requires a minimum spend of 20 but only entitles the user to a discount of 4. The user, however, has more time to avail of this offer compared to Offer wherein the user redeems a reward of the same value as the minimum spend.

The bar graph that all offers were sent via email, 9 were sent via the mobile app, 8 were sent through the web and 6 through social media.



These details bring about the questions:

- Which qualities influence the user to complete an offer?
- Which of these offers lead to a high transaction amount?
- Which channels would lead to more offer completions?

profile.json

The dataset contains details on the customer's demographic.

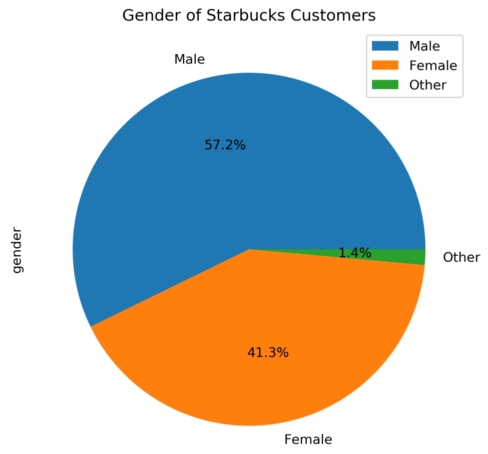
- **age (int)** - age of the customer
- **became_member_on (int)** - date when customer created an app account
- **gender (str)** - gender of the customer (note some entries contain 'O' for other rather than M or F)
- **id (str)** - customer id
- **income (float)** - customer's income

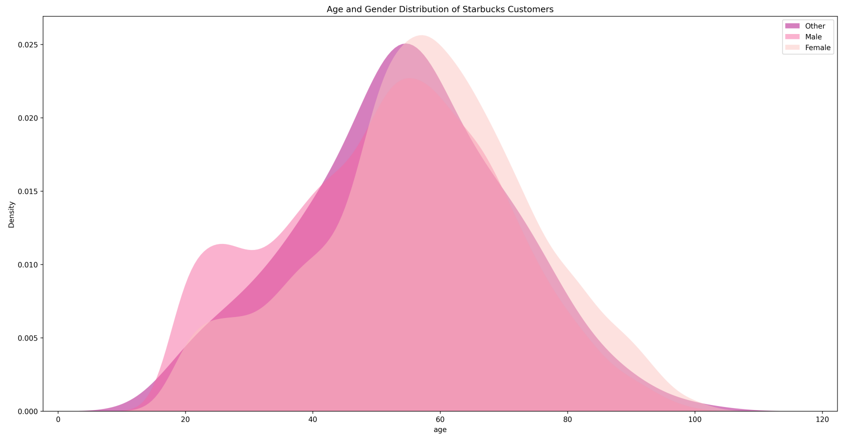
	gender	age	id	became_member_on	income
0	None	118	68be06ca386d4c31939f3a4f0e3dd783	20170212	NaN
1	F	55	0610b486422d4921ae7d2bf64640c50b	20170715	112000.0
2	None	118	38fe809add3b4fcf9315a9694bb96ff5	20180712	NaN
3	F	75	78afa995795e4d85b5d9ceeca43f5fef	20170509	100000.0
4	None	118	a03223e636434f42ac4c3df47e8bac43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

One glance at the user profile shows that it has some NaN values. Another value worth looking into would be the maximum age of the users. It is unlikely to have a user of age 118 as the oldest known living person is a 115-year-old woman in Spain. Conveniently, the users who are 118 years of age have NaN values in their income and gender columns.

Further exploration of the profile dataset also revealed that:

1.) The app has more male users than female users.



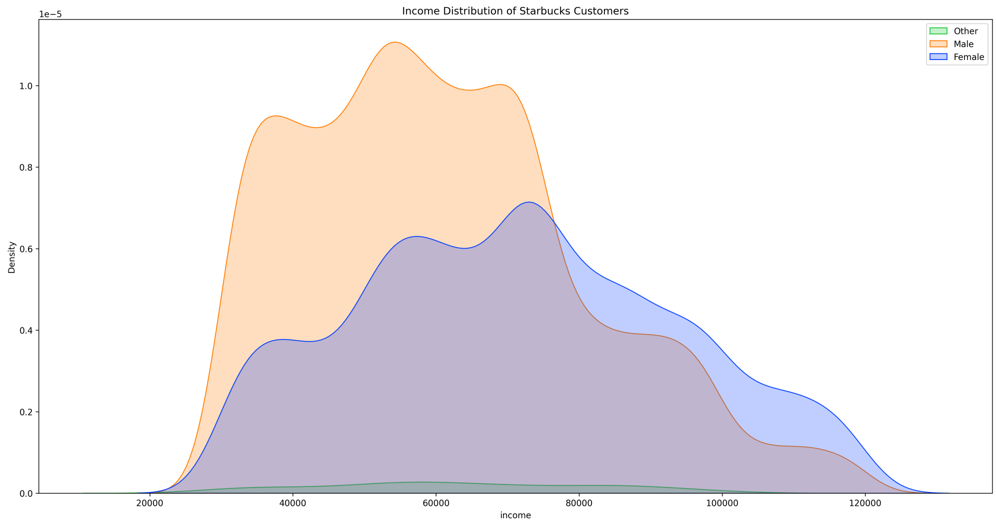


2.) There are more males aged 18 to 40 than there are females.

3.) The majority of the users are between 45 to 73 years of age.

4.) Female users have a higher income than the male users.

5.) Majority of the users earn around 49,000 to 80,000.



These details bring about the questions:

- Which of these users will make a transaction without a promotion?
- Which of these users will make a transaction because of a promotion?

transcript.json

This dataset contains customer transactions and defines when the customer receives, views, and completes the offer.

- **event (str)** - record description (transaction, offer received, offer viewed, offer completed)
- **person (str)** - customer id
- **time (int)** - time in hours since start of test. The data begins at time t=0
- **value - (dict of strings)** - consists of an offer id for offer received and viewed, an offer id and reward for offer completed, and the amount spent for transactions.

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

Transaction events contain a value for amount spent. Both offer received and offer viewed contain the offer id, whereas offer completed contains both the offer id and the reward received.

DATA PREPROCESSING AND FEATURE ENGINEERING

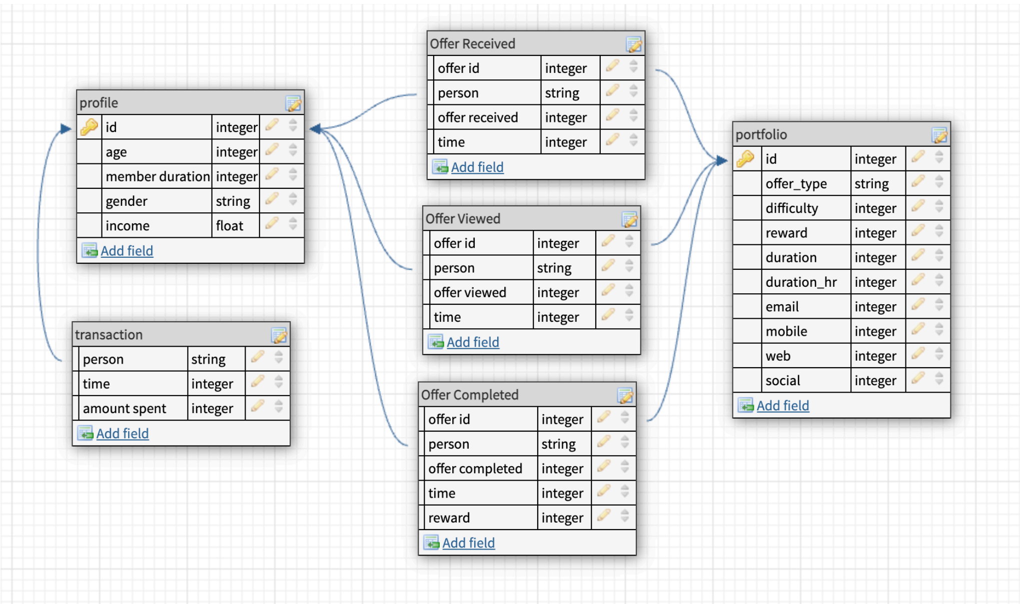
The first step was to clean the portfolio dataset by removing the rows with NaN values and ages that are above 118. The duration of membership was extracted from the date time object to feature engineer a column that contains the number of days the user has been a member using the difference between the maximum 'became member on' date and the actual 'became member on' date.

	id	age	gender	income	became_member_on	membership_duration
1	0610b486422d4921ae7d2bf64640c50b	55	F	112000.0	2017-07-15	376
3	78afa995795e4d85b5d9ceeca43f5fef	75	F	100000.0	2017-05-09	443
5	e2127556f4f64592b11af22de27a7932	68	M	70000.0	2018-04-26	91
8	389bc3fa690240e798340f5a15918d5c	65	M	53000.0	2018-02-09	167
12	2eeac8d8feae4a8cad5a6af0499a211d	58	M	51000.0	2017-11-11	257

As for the portfolio dataframe, a new column called offer which assigns numbers 1 - 10 for each type of offer. This will be used as offer_id in the final table for readability. MultiLabelBinarizer was then used to one hot encode the list of channels. For the purpose of scaling and comparing the duration of the offer to the times it was received, viewed, and completed, a duration_hr column was feature engineered.

	id	offer	offer_type	difficulty	duration	reward	channels	web	email	social	mobile	duration_hr
0	ae264e3637204a6fb9bb56bc8210ddfd	1	bogo	10	7	10	[email, mobile, social]	0	1	1	1	168
1	4d5c57ea9a6940dd891ad53e9dbe8da0	2	bogo	10	5	10	[web, email, mobile, social]	1	1	1	1	120
2	3f207df678b143eea3cee63160fa8bed	3	informational	0	4	0	[web, email, mobile]	1	1	0	1	96
3	9b98b8c7a33c4b65b9aebfe6a799e6d9	4	bogo	5	7	5	[web, email, mobile]	1	1	0	1	168
4	0b1e1539f2cc45b7b9fa7c272da2e1d7	5	discount	20	10	5	[web, email]	1	1	0	0	240
5	2298d6c36e964ae4a3e7e9706d1fb8c2	6	discount	7	7	3	[web, email, mobile, social]	1	1	1	1	168
6	fafdcd668e3743c1bb461111dcafc2a4	7	discount	10	10	2	[web, email, mobile, social]	1	1	1	1	240
7	5a8bc65990b245e5a138643cd4eb9837	8	informational	0	3	0	[email, mobile, social]	0	1	1	1	72
8	f19421c1d4aa40978ebb69ca19b0e20d	9	bogo	5	5	5	[web, email, mobile, social]	1	1	1	1	120
9	2906b810c7d4411798c6938adc9daaa5	10	discount	10	7	2	[web, email, mobile]	1	1	0	1	168

Before extracting any information from the values dictionary, the transcript table was joined to the profile table using the id and person values. The table was then separated by event: transaction, offer received, offer viewed, and offer completed.



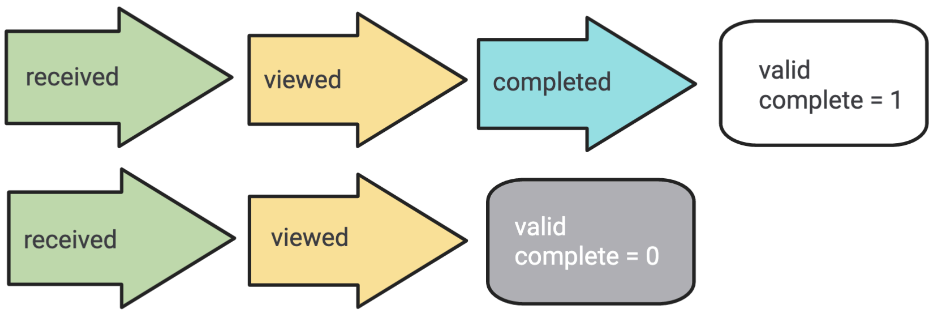
The diagram shows how the tables were merged.

After extracting the offer id and other values from the dictionary, the offer id and person columns were used to merge the offer received, offer viewed, and offer completed tables. These were then merged with the portfolio table using the offer id. After ensuring that all the id columns were the same, the duplicate columns were dropped and the values for offer viewed, offer received and offer completed were then replaced with 0 for NaN values and 1 for values that match its respective column.

Unfortunately, because a user can receive the same offer more than once, the joins generated additional rows. To solve this issue, the columns offer received, offer viewed, offer completed, time received, time viewed, time completed, reward (from the portfolio) and reward received (from the transcript) were compared to look for inconsistencies:

- The time viewed should not be less than the time received as it is impossible to view the offer before it was received.
- Time viewed should be zero if the offer was not viewed.
- A reward should not be received if the offer was not completed.
- The offer was completed before the end of the offer period but the reward received is 0.
- The offer was completed after the offer period was over and yet the user received a reward.
- The reward indicated with the offer should be the same as the reward received.

After all the inconsistent rows were dropped, a valid_complete column was feature engineered to indicate that the offer was received, viewed, and completed. This will later on be used to identify and label the buyers who avail of BOGO and discount promotions.



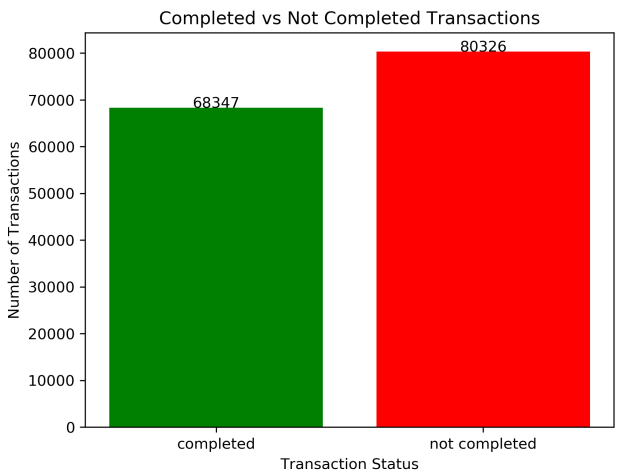
The transaction table was then merged and inconsistent rows were removed using the offer type, time, and reward as reference. A label column was then created to segment the users:

USER TYPE	
BUYS WITHOUT PROMO	the transaction was made before the offer was received
BUYS WITHOUT PROMO	the offer was completed without viewing
INFORMATIONAL	the offer was viewed and the amount spent is greater than 0

USER TYPE	
BOGO BUYER	the offer was viewed and completed for BOGO offer type
DISCOUNT BUYER	the offer was viewed and completed for discount offer type

A new column called completed was created to reflect the purchases completed by informational, BOGO, and discount buyers. The feature engineered columns that were used for the purpose of verification and labelling were then dropped as they are unnecessary for training.

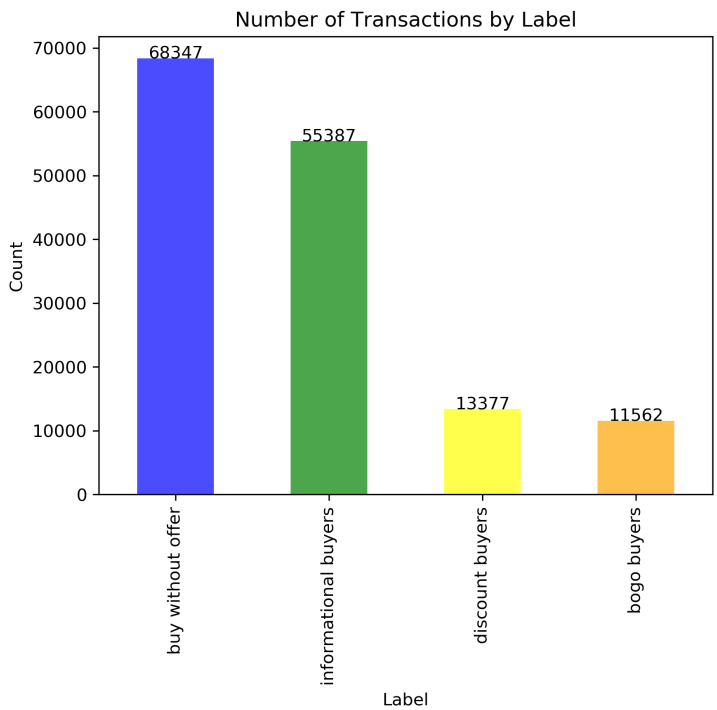
ANALYSING THE JOINED TABLES



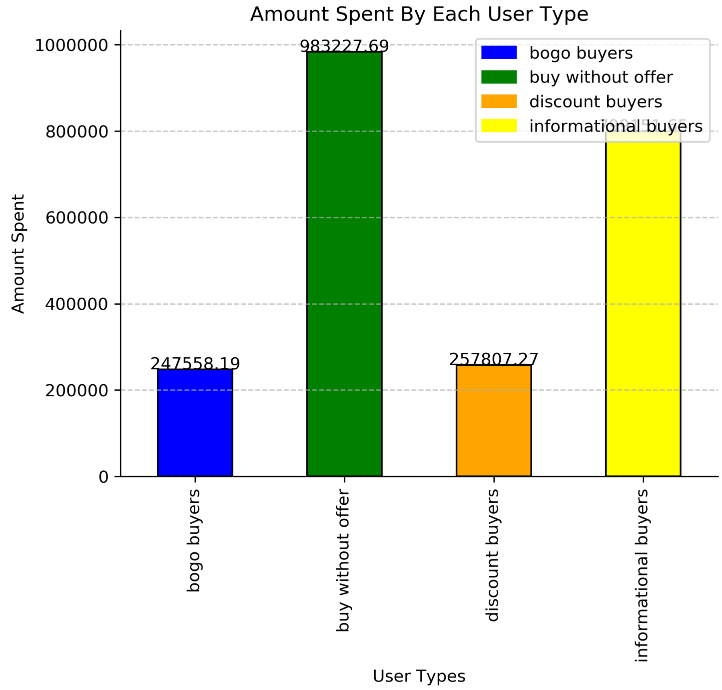
The bar chart on the left shows the number of completed and uncompleted offers. Take note, however, that the number of completed offers here to refer to the offers that were viewed and completed while "not completed" refers to the number of transactions outside the offer.

Although the study of these values is relevant as it can be used to predict whether a customer will make a purchase due to an offer or not as well as the underlying offer features that converted the customer, it is, unfortunately insufficient in providing insights to the demographic where the offers should be distributed.

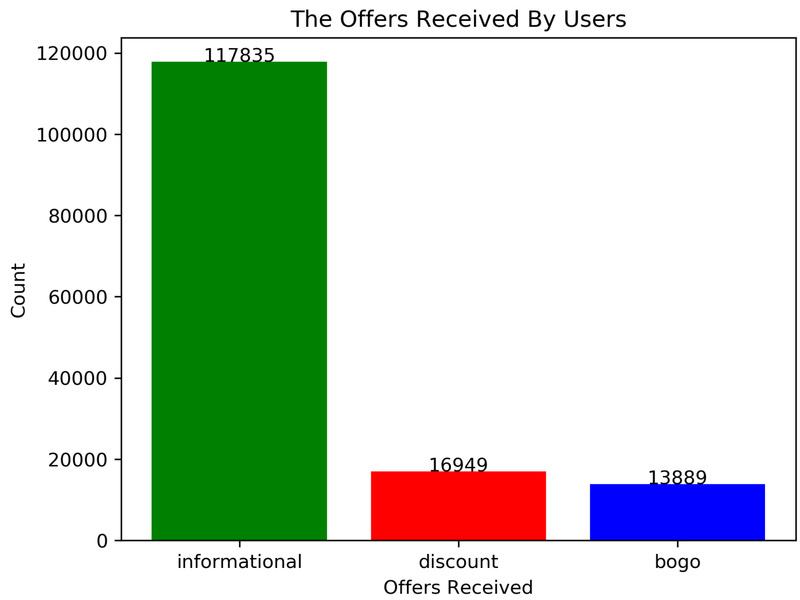
A look at the bar charts for the labelled columns show that there are more transactions from users who would make a purchase without an offer and that these users spend more in total than those who purchase with offers. This is quite logical as the lack of a discount means a higher amount would be spent on a product and offers that are sporadically sent would mean that regular customers would buy products at their original cost. It is, however, important to note sending offers to customers who would make a purchase with



or without it means that promotions are not being distributed efficiently. These promotions are best used as a motivator for promotion enthusiasts to buy more products.



It should also be noted that next to the "buyers without offers," the informational offers seem to generate the most number of transactions and the highest amount spent. Although this is probably due to the fact that more informational offers are sent than all the other promotions, it is still worth noting which demographic is influenced by each type of promotion in the goal of optimising Starbucks' promotion distribution to generate more transactions.



Seeing that both scenarios are relevant to the optimisation of Starbucks' promotion distribution, it poses two types of problems that will be solved by using machine learning algorithms, deviating from the original procedure that was indicated in the project proposal. The first will be a binary classification problem which will predict if the offer compels the user to complete it and the second one will be a multi-class classification problem which will predict the user type.

IMPLEMENTATION

BINARY CLASSIFICATION - MODEL SELECTION

To choose the best model for these problems, the performance of Decision Tree, Random Forest, and LightGBM will be compared using baseline models that will allow a comparison of important features and chosen metrics before moving on to a more rigorous cross validation evaluation.

The performance of the models will be compared using KFold Cross Validation. The metrics that will be used to compare them would be the f1 score, which will give a general evaluation of each model, and recall, which will measures the model's ability to predict true positives thereby decreasing false negatives. As the model is meant to search for customers who would go for the offer, it would be beneficial to have lower false negatives and miss out on the customers who are willing to take the offer.

The features that will be included in training the binary classification model are: 'age', 'income', 'membership_duration', ‘amount_spent', 'difficulty', 'duration', 'reward', 'web', 'email', 'social', ‘mobile' and the hot encoded values of gender and offer type: 'offer_type = _bogo', 'offer_type = _discount’, 'offer_type = _informational', 'gender = _F', 'gender = _M’, 'gender = _O’.

To predict if a user will complete any of the offers and ensure that they have viewed the offer beforehand, the values in the feature engineered completed column which contains 1 if the offer was viewed and the offer was completed or there was an amount registered before the duration of the offer has ended, will be label encoded and used as the target variable.

MULTI-CLASS CLASSIFICATION - MODEL SELECTION

The multi-class classification problem, on the other hand, has imbalanced target labels, where the “buyers without offer” and “informational buyers” are clearly dominant as shown in the previous bar charts. In this case, the StratifiedKFold validation will be used to ensure that each label will be proportionally distributed along each fold. To measure the performance of each model, the f1 score will be used to measure the model's performance as this metric tends to give robust results when faced with an imbalanced dataset.

For the purpose of segmenting and predicting the user type of each user, the second model uses the label encoded label column as the target value. The features that will be included in the training are: 'age', 'income', 'membership_duration', 'time', 'difficulty', 'duration', 'reward', 'web', 'email', 'social', ‘mobile' and the hot encoded values of gender and offer type: 'gender = _F', 'gender = _M’, 'gender = _O’.

REFINEMENT

The datasets used for both problems will be divided into training, validation and test sets. For the initial training and cross validation, the training set will be used to find the best model.

After selecting the model with the best performance using the train and valid sets, the model will undergo hyperparameter tuning by running a randomised search cross validation on a subset of selected hyper parameters with values that are closest to the default values to:

1.) Save computing overhead. The project was run on a ml.t3.medium throughout the hyperparameter tuning process.

2.) Avoid overfitting.

3.) Explore the parameter space more quickly, giving a better idea of how the hyper parameters can be adjusted for future iterations.

4.) Attain a better search space. Often, the best hyperparameters lie in a lower range because they offer simpler models with better generalization capabilities.

The chosen range of values for hyperparameter tuning will be gradually increased based on the results. At the same time, the best score for each iteration will be recorded.

Once the best score has stopped increasing and the best model is identified, the test data will be used to evaluate the model's performance using the metrics indicated and a confusion matrix to identify the model's error count.

RESULTS

BINARY CLASSIFICATION MODEL: WILL THE USER COMPLETE THE OFFER?

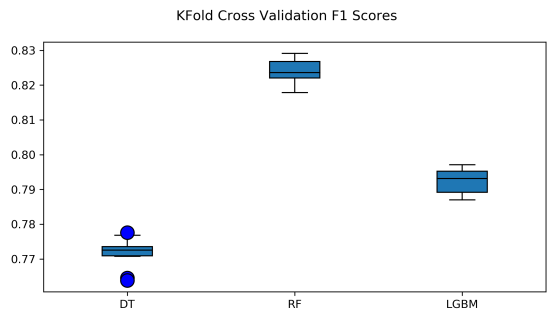
Model Selection

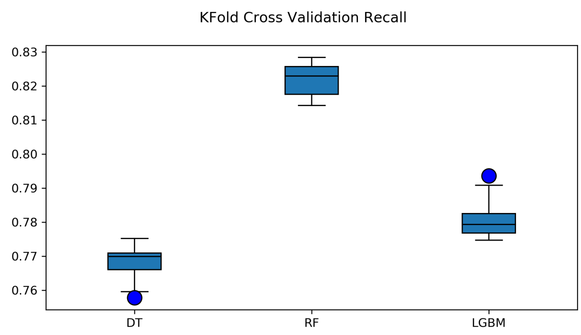
The baseline runs for each model generated the following metric scores where LightGBM outperformed the other 3 models in both metrics.

	F1 Score	Recall
Decision Tree	0.75	0.76
Random Forest	0.70	0.65

	F1 Score	Recall
LightGBM	0.72	0.69

A comparison of the KFold Cross Validation scores across 10 folds, however, show that the Random Forest gave the best results in both F1 Score and Recall with mean of 0.82 for both metrics. The box plots below also show that the results of the Random Forest are quite consistent.





When comparing multiple models, using the results of a cross-validation for model selection is preferred because it allows for a more accurate estimate of model performance. Train-test split can lead to overfitting or underfitting of the model on the validation set and cross-validation helps to reduce the variance in the estimate of model performance by averaging over multiple splits of the data. In this case, the Random Forest Classifier will be selected for tuning.

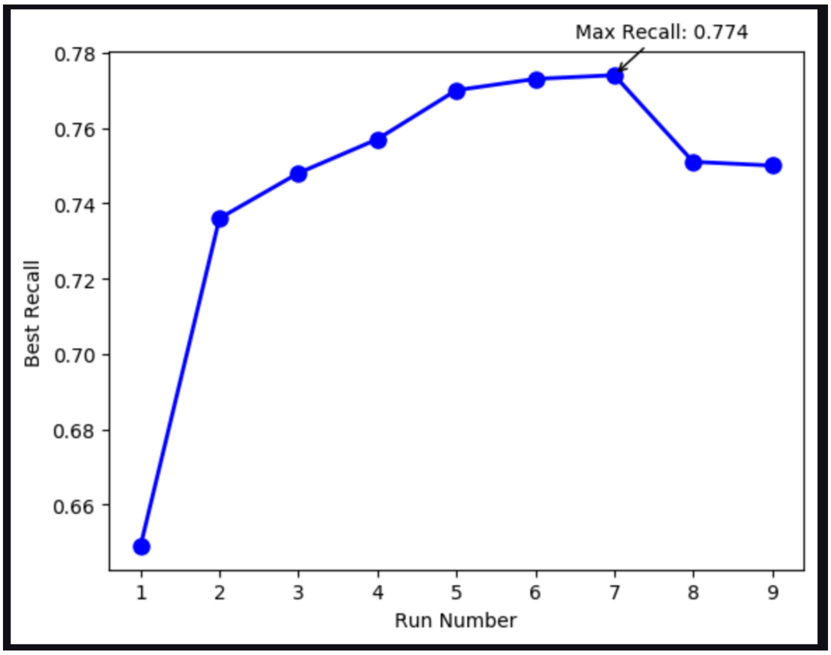
Baseline Model and Hyperparameter Tuning

As indicated above, the baseline run of the Random Forest Model generated the following F1 and recall scores respectively, 0.70 and 0.65. As there is a need to decrease the number of false negatives while improving model performance, recall will be used as the metric for hyper parameter tuning.

The hyperparameters chosen for tuning of this model are max depth, max features, minimum samples leaf, minimum samples split, n_estimators:

Hyperparameters	Why are they relevant?
Max Depth	This controls the maximum depth of each decision tree in a random forest, affecting how well the model can learn from the training data. If the "max depth" is too high, the model might try too hard to fit the training data and end up not being able to generalise to new, unseen data (which is bad). If the "max depth" is too low, the model might not be able to capture all the important patterns in the training data and again, not be able to generalise to new data
Max Features	This determines the maximum number of features to consider when splitting each node in a decision tree, affecting the diversity of the model.
Minimum Samples Leaf	This determines the minimum number of samples required to be at a leaf node in a decision tree, affecting the generalisation of the model.
Minimum Samples Split	This determines the minimum number of samples required to split an internal node in a decision tree, affecting the robustness of the model. A higher minimum samples split can lead to more robust decision trees that are less sensitive to noise in the data, while a lower minimum samples split can lead to decision trees that are sensitive to noise.
N Estimators	This determines the number of decision trees to include in a random forest. A higher n_estimators can lead to better performance because the model is more likely to capture the true underlying patterns in the data, while a lower n_estimators can lead to less accurate predictions.

The best model was chosen by monitoring the model's performance through several runs. The graph below shows how the model's performance slowly dropped due to overfitting as the model's hyperparameters were increased.

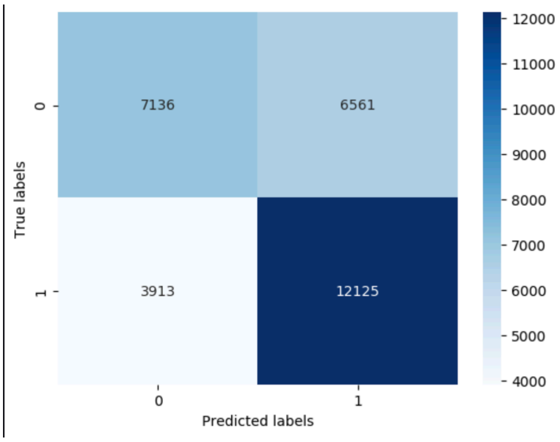


The model's best recall score was 0.774 with the following best parameters:

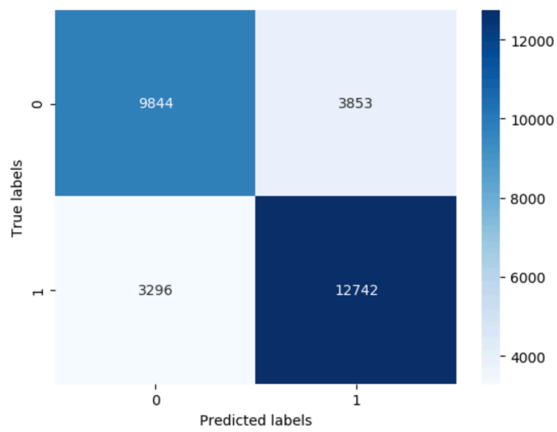
Hyperparameters	
n_estimators	2400
min_samples_split	12
min_samples_leaf	5
max features	sqrt
max depth	20

Testing the best model with the test set generated an f1 score of 0.78 and a recall of 0.80, which is comparatively better than the baseline model's score of 0.70 and 0.65, respectively.

A comparison of the baseline confusion matrix (left) and the best model's confusion matrix (right) also shows a degree of improvement in the number of false negatives in the predictions.



Baseline Model



Best Model

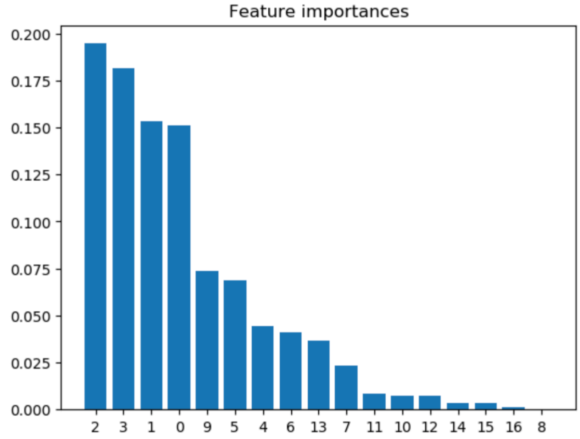
Feature Importance

Feature Importance refers to the scores calculated to represent the importance of each feature. A higher score would mean that the corresponding feature will have a high effect on the model.

According to this list, age, income, and membership duration play a big role in determining whether a user will complete an offer or not. These are followed by amount spent, difficulty, duration, reward, web, email, social, and mobile.

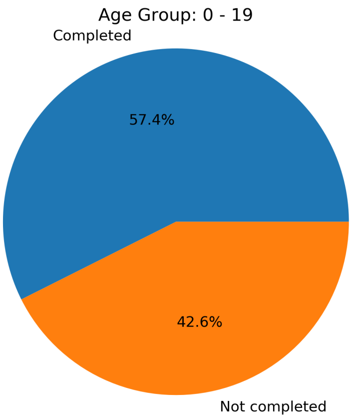
As each offer tend to have a difficulty level that corresponds to the amount that need to be spent, it is no wonder that the amount spent has made it to the list of important features. It does make sense that users who complete more offers would have spent more on products in order to be able to do this.

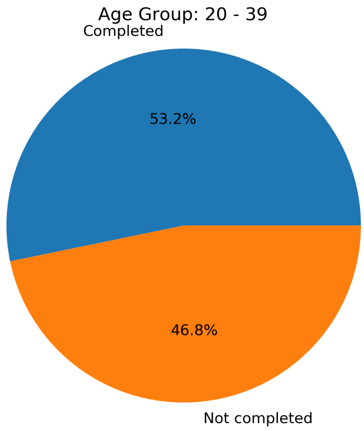
```
Feature ranking:
1. feature 2 age (0.194701)
2. feature 3 income (0.181556)
3. feature 1 membership_duration (0.153233)
4. feature 0 amount_spent (0.151066)
5. feature 9 difficulty (0.073865)
6. feature 5 duration (0.068864)
7. feature 4 reward (0.044291)
8. feature 6 web (0.041274)
9. feature 13 email (0.036563)
10. feature 7 social (0.023419)
11. feature 11 mobile (0.008489)
12. feature 10 offer_type = _bogo (0.007570)
13. feature 12 offer_type = _discount (0.007176)
14. feature 14 offer_type = _informational (0.003394)
15. feature 15 gender = _F (0.003214)
16. feature 16 gender = _M (0.001326)
17. feature 8 gender = _O (0.000000)
```

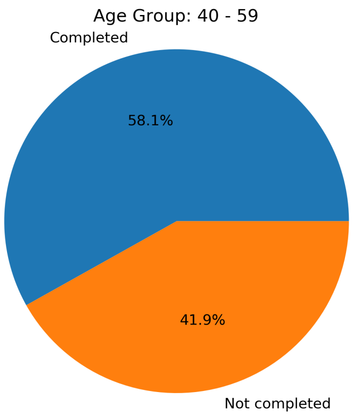


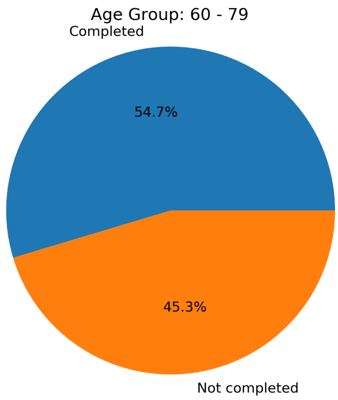
Predicted Offer Completion By Age

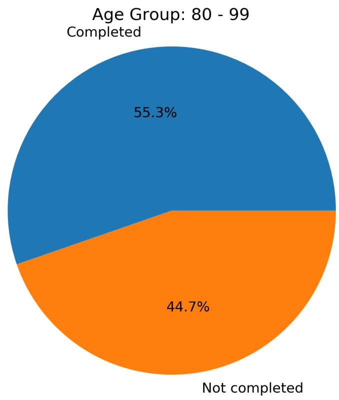
The model predicts that users aged 40 - 59 have a higher percentage of completed offers while users in the 20 - 39 bracket have the lowest. This can indicate that more offers should be distributed to users who are aged 40 - 59 as they are more likely to be encouraged to buy products to complete an offer.





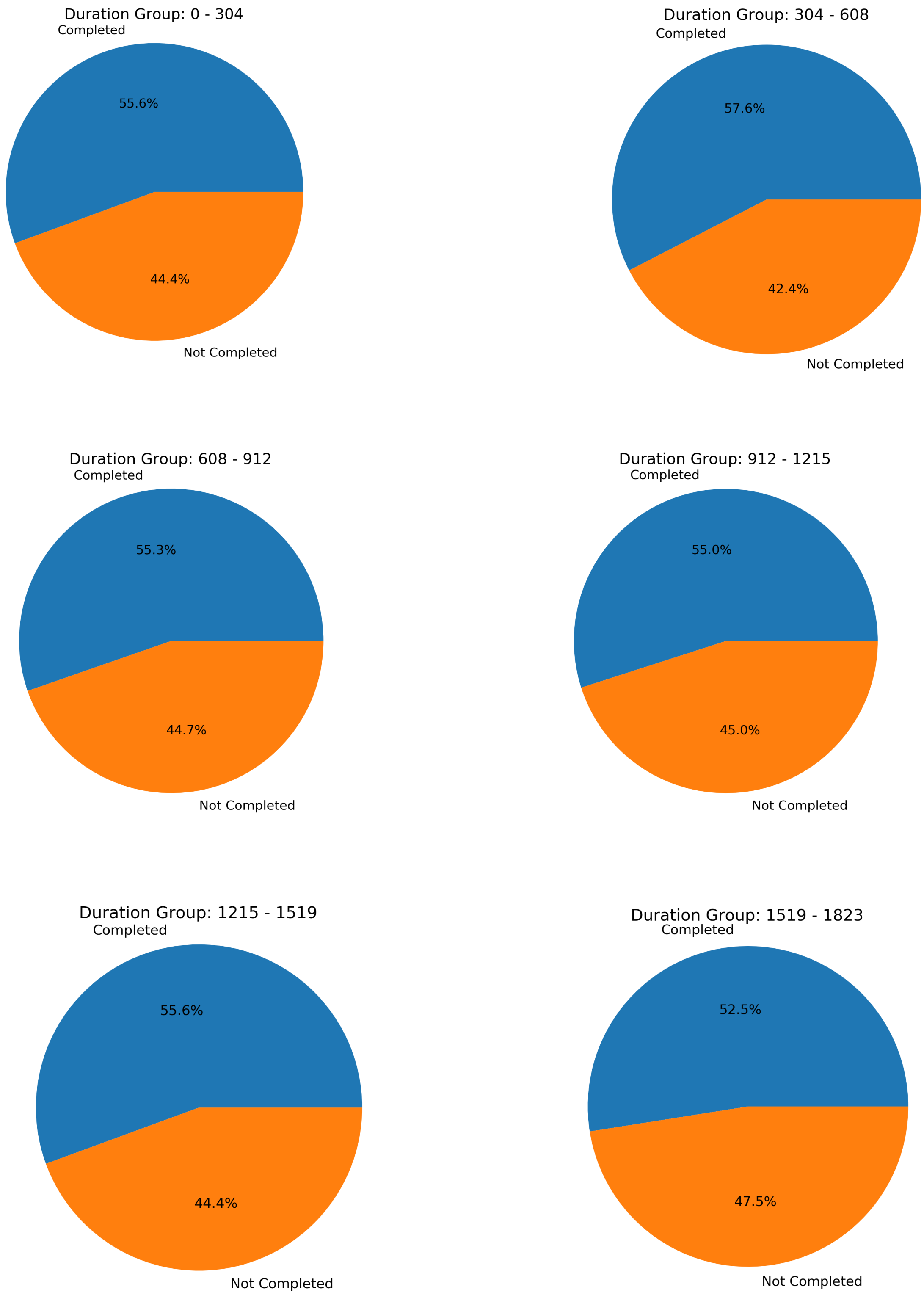






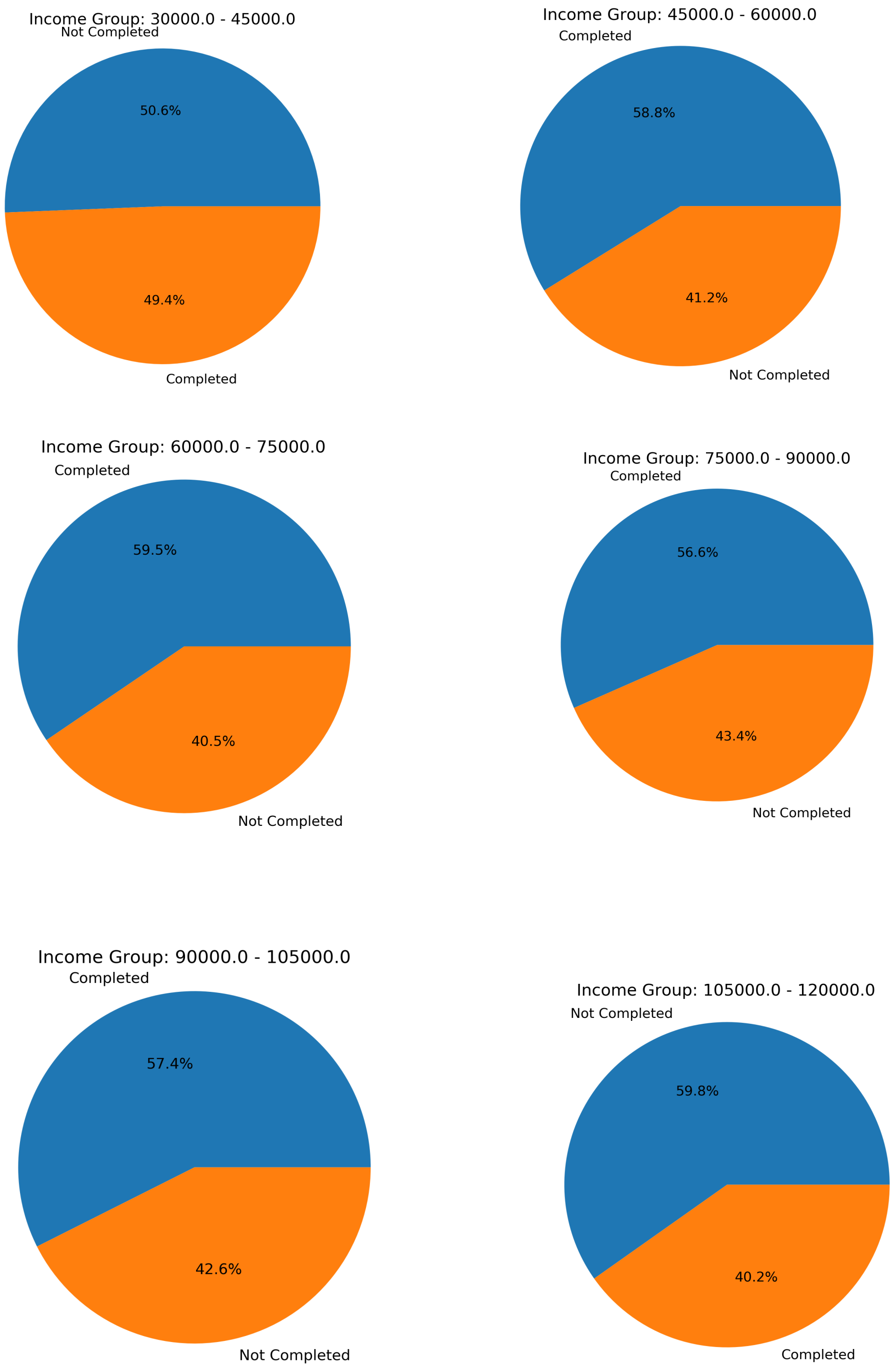
Predicted Offer Completion By Membership Duration

The model predicted that users with a membership duration between 304 to 608 days have a higher percentage of completed offers while the oldest range with a duration of 1519 to 1823 days have the lowest percentage of completed offers. This can suggest that offers should be sent to new members who are more likely to use them.



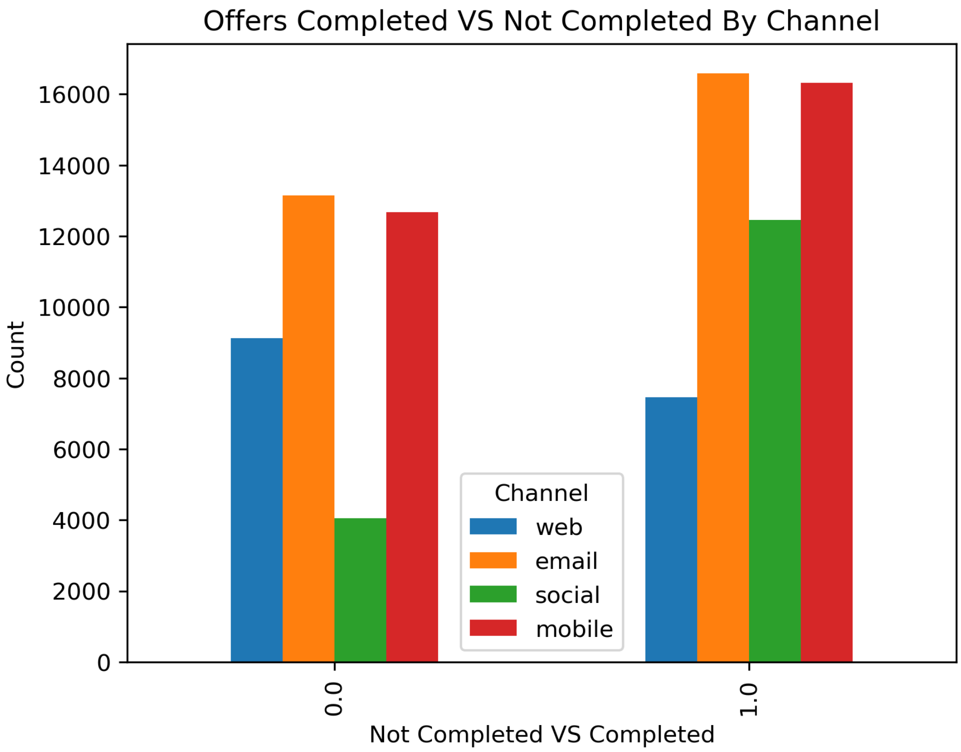
Predicted Offer Completion By Income

The model predicted that the users from the lowest income group (30,000 to 45,000) have the lowest percentage of completed offers while the users from higher income groups tend to have a higher percentage of completed offers. Although this suggests that users of higher income should be targeted by promotions, it can also be interpreted as users with more spending capability have more opportunities to complete offers. In this case, Starbucks may want to look into offers that are more feasible for people with a lower incomes, leading to more purchases from this demographic.



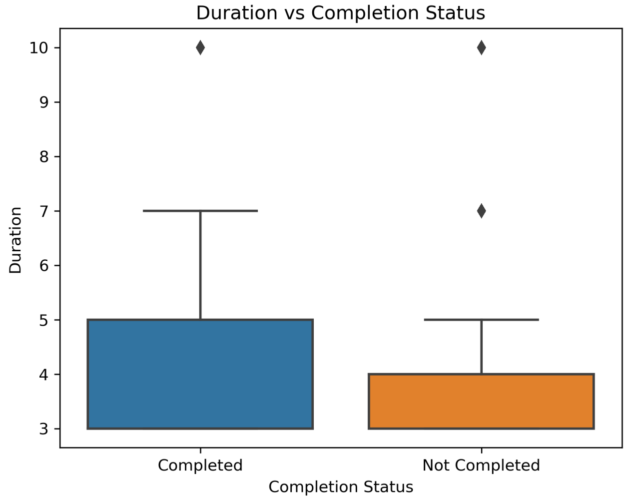
Predicted Offer Completions By Channel

The model predicted that most of the completed offers come from users who were informed by mobile and email. These seem to be the best channels for informing members about promotions.



Predicted Offer Completion Based On Duration, Difficulty and Reward

The predicted completed offers indicate that that there are lower difficulty, duration and rewards tend to result to more completed offers. This can mean that users complete offers based on their convenience rather than the value of the reward offered. It is also noticeable that the offers with high rewards, high difficulty and low duration tend to be uncompleted. it seems that Starbucks will need to focus on feasibility rather than enticing users to spend more at a short period of time with high rewards.



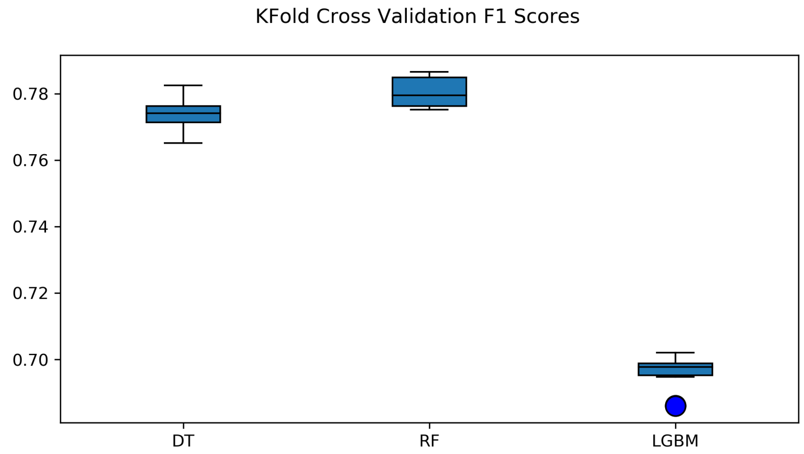
MULTI-CLASSIFICATION MODEL: WHICH GROUP DOES THE USER BELONG TO?

Model Selection

The baseline runs for each model generated the following metric scores where Decision Tree Classifier outperformed the other 3 models in both metrics.

	F1 Weighted
Decision Tree	0.77
Random Forest	0.65
LightGBM	0.70

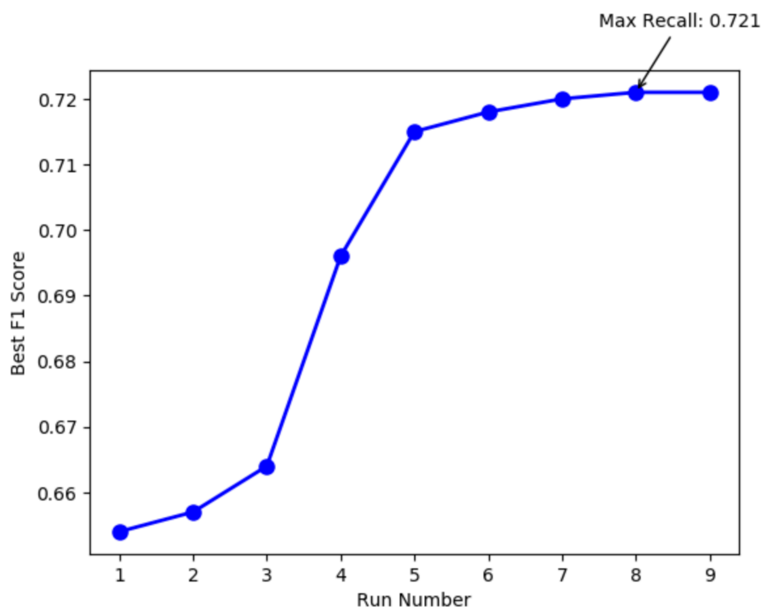
A comparison of the stratified cross validation scores across 10 folds show that the Random Forest generated the best result with a mean f1 score of 0.780. Although the decision tree comes very close, it is clear from the box plots below that the Random Forest generated the highest mean and the most consistent results across all folds.



Baseline Model and Hyperparameter Tuning

As indicated above, the baseline run of the Random Forest Model generated a weighted f1 score of 0.65. This will be considered the baseline for hyperparameter tuning. As the same model is used for this dataset, the same hyperparameters will be used in the process.

The best model was chosen by monitoring the model's performance through several runs. The graph below shows how the model's performance has started plateauing as the hyper parameters are increased.



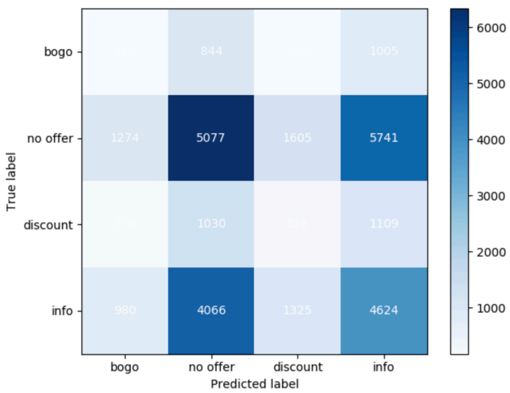
The model's best recall score was 0.774 with the following best parameters:

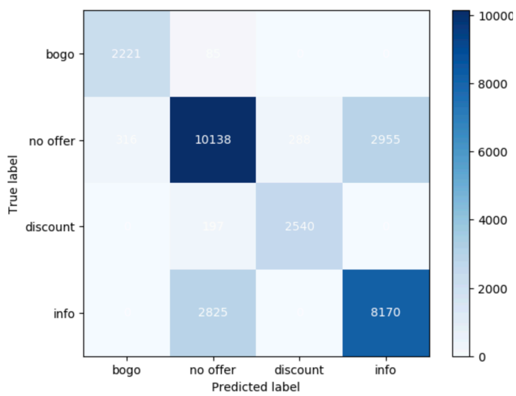
Hyperparameters	
n_estimators	2400
min_samples_split	8
min_samples_leaf	5
max features	sqrt
max depth	23

Results From The Best Model

Testing the best model with the test set generated an f1 score of 0.77 which is comparatively better than the baseline model's score of 0.65.

A comparison of the baseline confusion matrix (left) and the best model's confusion matrix (right) also shows a degree of improvement in the number of errors in the predictions.

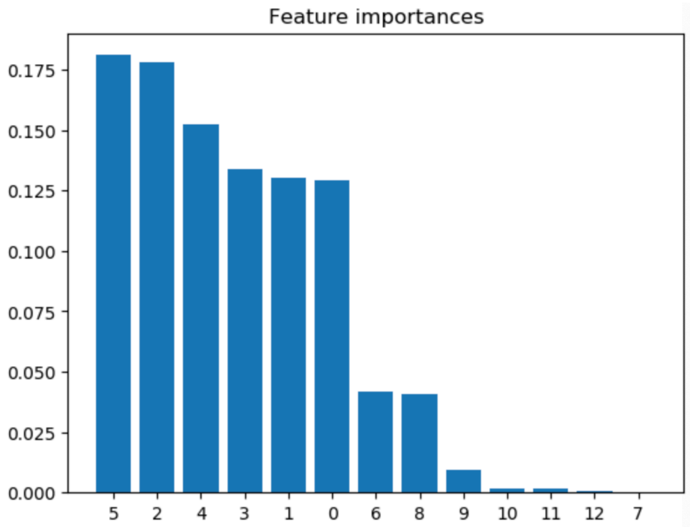




Feature Importances

The feature importances generated by the multi classification models are very much similar to those of the binary classification models. Age, income, membership, duration, and reward play a big role in predicting the buyer group.

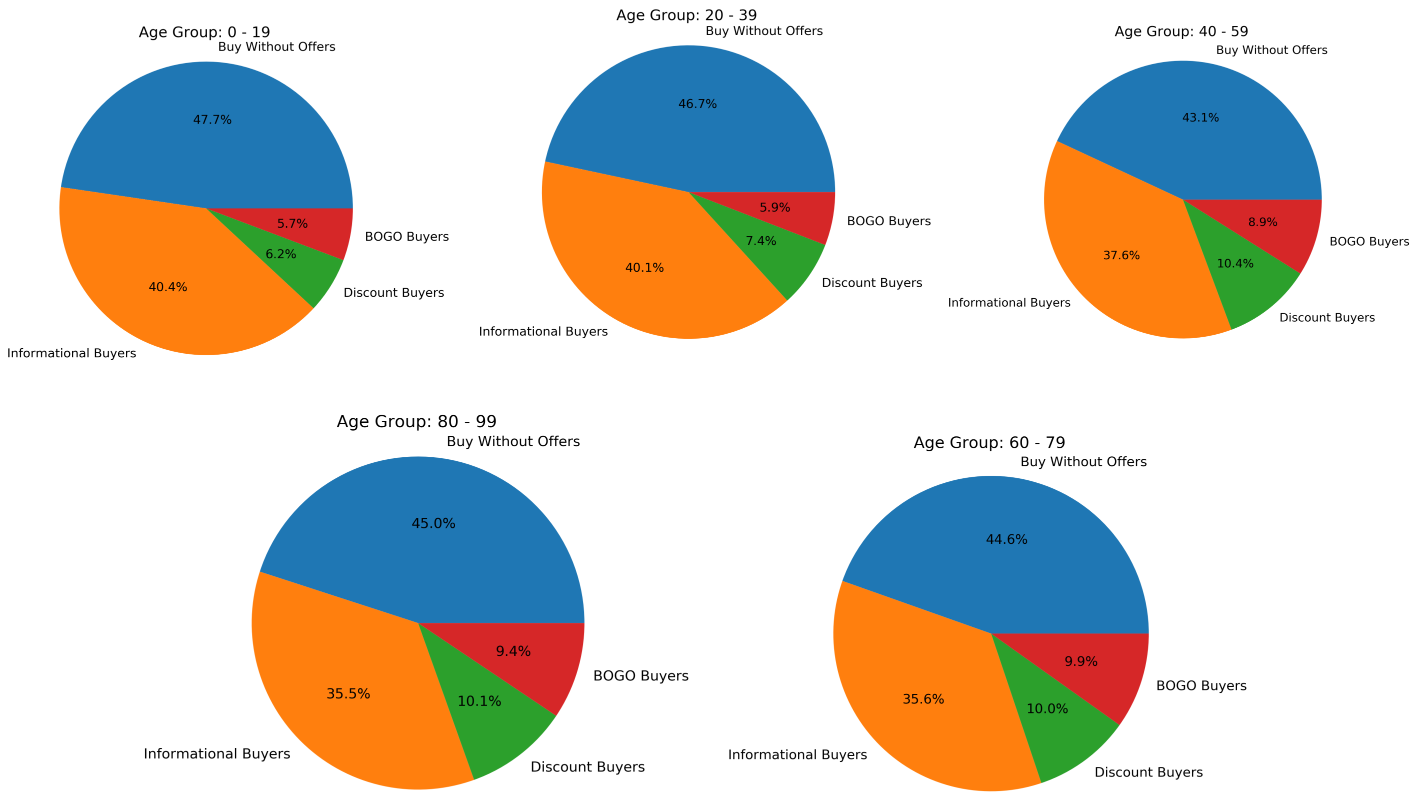
```
Feature ranking:
1. feature 5 age (0.180973)
2. feature 2 income (0.177840)
3. feature 4 membership_duration (0.152427)
4. feature 3 difficulty (0.133604)
5. feature 1 duration (0.130295)
6. feature 0 reward (0.129032)
7. feature 6 web (0.041689)
8. feature 8 email (0.040847)
9. feature 9 social (0.009654)
10. feature 10 mobile (0.001462)
11. feature 11 gender = _F (0.001429)
12. feature 12 gender = _M (0.000749)
13. feature 7 gender = _O (0.000000)
```



Predicted User Types Based on Age

The predicted user types by age show that there is a bigger percentage of buyers without promotions among the younger age group. It is interesting to see that the same age group would also have a bigger number of informational buyers at 40.4% followed by the age group within 20-29 at 40.1%. The highest percentage of BOGO buyers can be found at the ages of 40 - 99 with percentages that range from 8.9 - 9.9% which would also be the same demographic where the discount buyers are much higher.

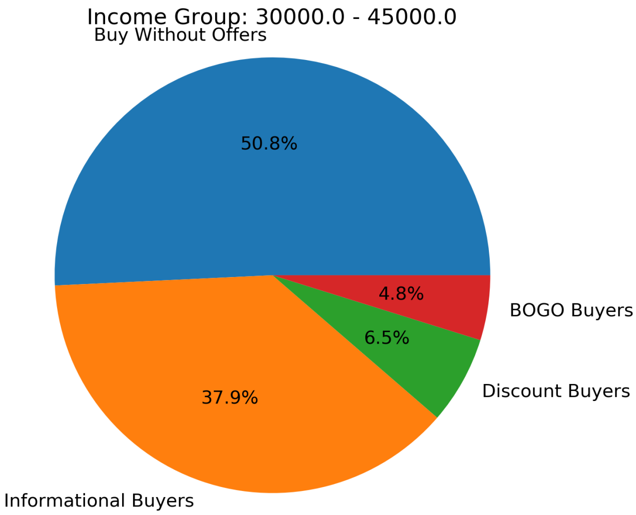
This could suggest that the younger group ranging from below 19 to 39 years old should receive more informational promotions while the older group between 40 to 99 should receive more BOGO and discount offers.

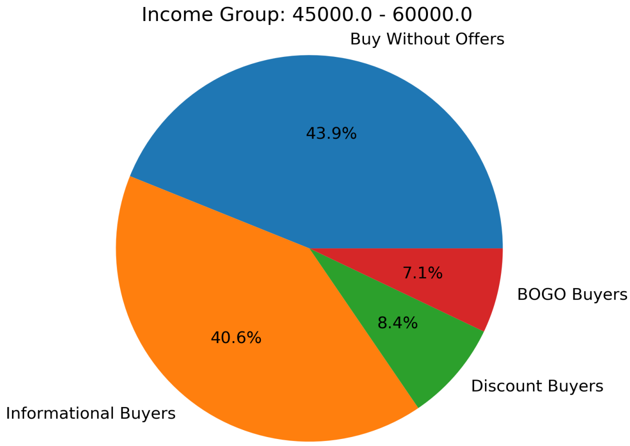


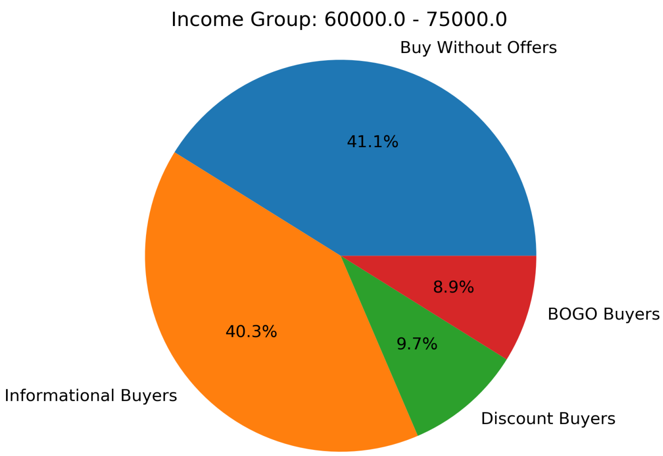
Predicted User Types By Income

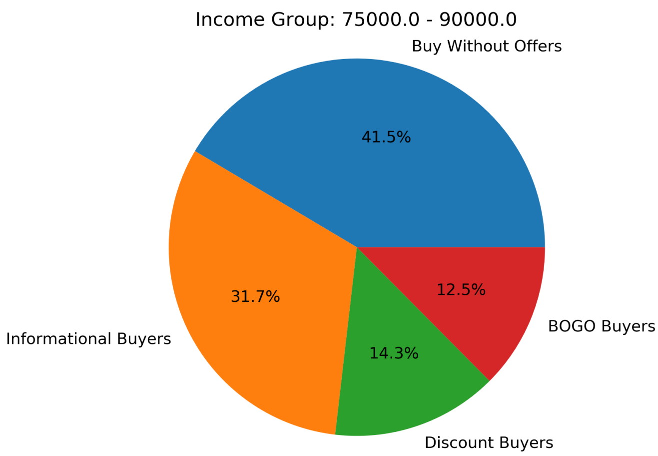
The model predicts that there is a higher percentage of users who buy without offers at the highest and lowest income ranges at 50.8% and 55.0% respectively. Informational buyers, on the other hand, seem to have a higher population among users with incomes ranging from 45,000 to 75,000. More BOGO and discount buyers can be found within the income range of 75,000 - 105,000.

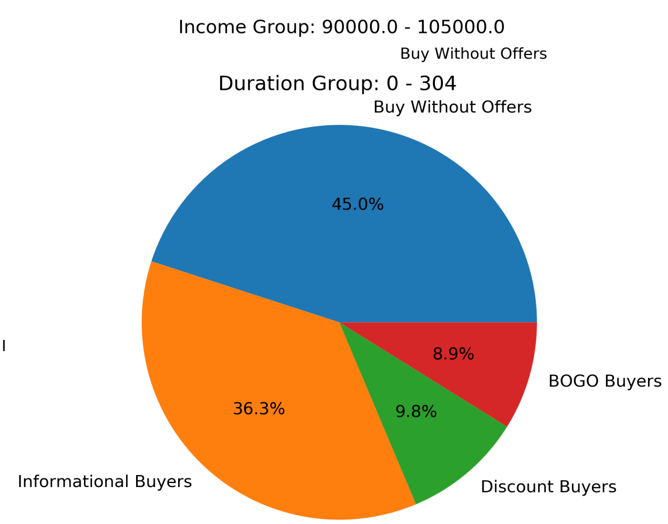
This suggests that people with an income of 45,000 - 75,000 should receive more informational promotions whereas BOGO and discount offers should be sent to users with an income range of 75,000 - 105,000.

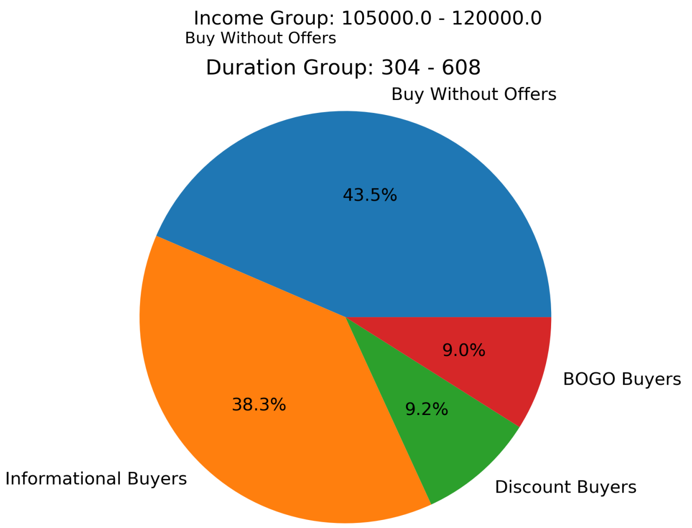






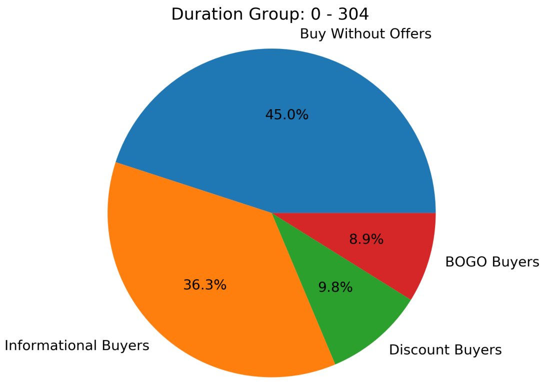


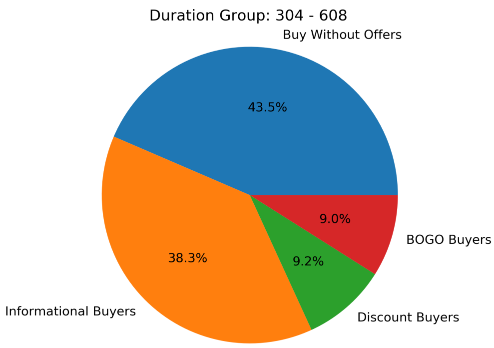


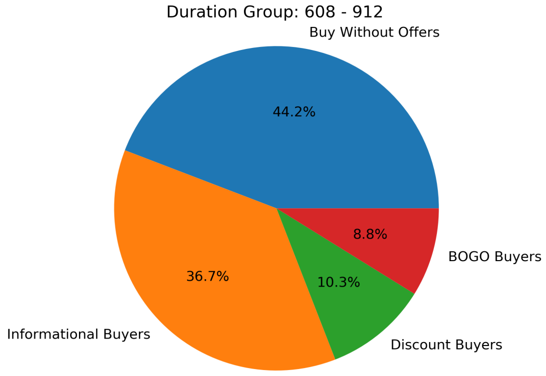


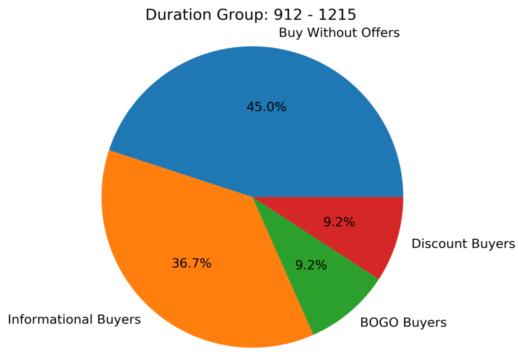
User Types By Membership Duration

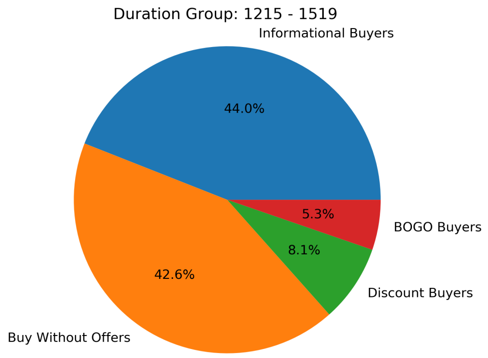
The model predicts that the oldest members (with membership durations of 1519 - 1823) tend to buy without offers at 48% whereas the other members have a higher percentage of users who will buy products with an offer. This result seems to suggest that it is best to send offers to newer members encourage them to buy more products.

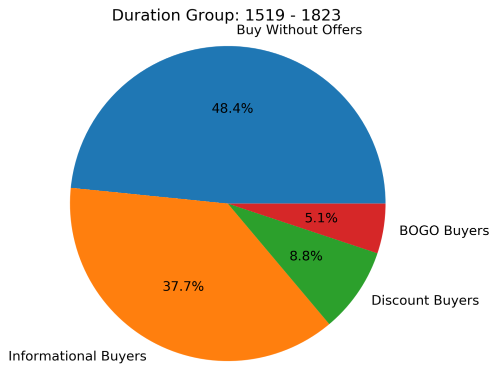








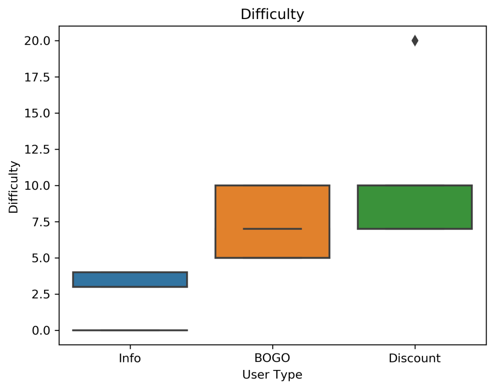


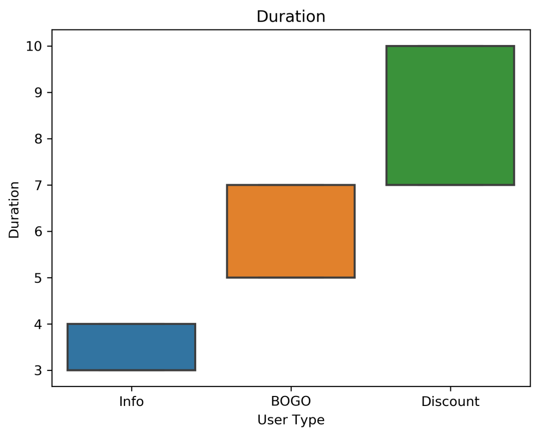


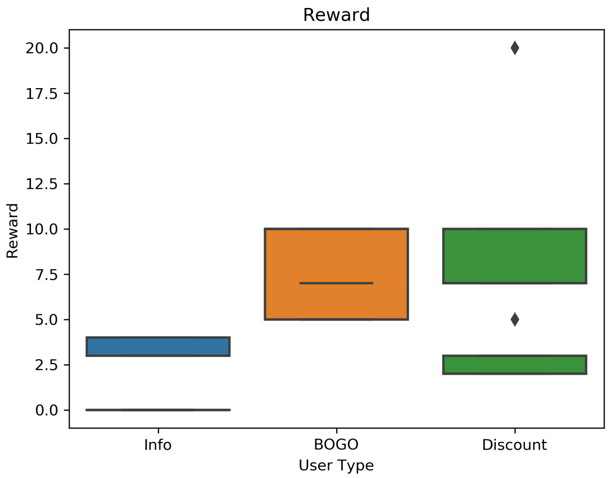
Duration, Difficulty, Reward and User Types

Discount Users seem to prefer a longer duration compared to info and BOGO users, however this may not be dictated by preference but rather by the duration given for each offer as discounted offers tend to have the longest duration among the 3.

What is noticeable for the comparison of difficulty and reward is that BOGO users tend to go for BOGO offers with a bigger range of difficulty and reward, perhaps due to the proportional nature of reward and difficulty in BOGO offers.







CONCLUSION

The results generated by the model has provided some strong insights:

- Users aged 40 - 59 have a higher percentage of completed offers while users in the 20 - 39 bracket have the lowest, indicating that more offers should be distributed to users who are aged 40 - 59 as they are more likely to be encouraged to buy products to complete an offer.
- Offers should be sent to new members who are more likely to use them and might encourage loyalty and regular buying habits in the future.
- Users from the lowest income group (30,000 - 45,000) tend to have more uncompleted offers. Although this might indicate that offers should be distributed to users with higher income, it is more plausible that the lowest income group would have a harder time completing offers and Starbucks may need to look into devising offers that are more feasible for people with lower incomes, leading to more purchases from this demographic.
- Members have a higher chance of completing offers when they are sent through mobile and email.
- Users tend to complete offers with lower difficulty and reward, indicating that convenience and feasibility is preferred over the value of the reward. This indicates that perhaps offers with high difficulty and lower rewards may not necessarily lead to more purchases.
- The younger age groups tend to make a purchase after receiving informational offers.
- Users ages 40 - 99 tend to complete discount and BOGO offers.
- Users with an income of 45,000 to 75,000 tend to complete more informational promotions while users with an income range of 75,000 - 105,000 tend to complete more BOGO and discount offers.

Although the models have provided some insights to customer motivation to complete offers and make purchases, the confusion matrices indicate that both models require further improvements in order to solve the problem. The binary classification model, for instance, still has a high a number of false negatives, indicating that it has missed 3,296 completed offers, which can potentially change the distribution of percentages for each demographic.

As for the multi-class classification model, it has misclassified 2955 informational buyers that should have been buyers without offer and 2825 buyers without offer that should have been informational buyers. This shows that the model requires further improvement to segment users. As the classes are not as imbalanced as that of the multi class classification problem, it is possible that decreasing the features or using a different metric to tune the model can improve the model's performance. As a good recall is of importance for this problem, perhaps using the F-beta score with a value of 2 or 3 can be useful here.

It is quite noticeable that insufficient data and diversity of data have led to lower prediction scores and insights, especially in the case of of the multi-class classification model, where the users who make purchases due to informational offers and buyers who buy without offers are much higher than the users who complete BOGO and discount offers. In this case, the model may be improved by: (1) The addition of more data, (2) Having only 3 user types where users who tend to complete BOGO and discount offers are combined, and (3) Synthesising new samples using SMOTE to remedy that imbalance.

As the insights from the visuals are highly influenced by the data given, this study could also benefit from more data, particularly with regards to the variety of offer types and transactions. From the very beginning, it was quite noticeable that the informational offers dominated the other offers: BOGO and discount. Also, with only 10 types of offers to base this study on, it is quite hard to say whether the qualities of the offer were the reasons for its success.

REFERENCE:

* Fontanella, C. (2022, August 18). A beginner's Guide to Customer Behavior Analysis. HubSpot Blog. https://blog.hubspot.com/service/customer-behavior-analysis#:~:text=Customer%20behavior%20refers%20to%20an,enticing%20products%20and%20service%20offers. Retrieved February 26, 2023.

* Starbucks Marketing Strategy. https://www.studysmarter.co.uk/explanations/business-studies/business-case-studies/starbucks-marketing-strategy/ (n.d.). StudySmarter UK. Retrieved 27 February 2023.

* The 10 Largest Coffee Companies In The World – Zippia. https://www.zippia.com/advice/largest-coffee-companies/ (2022, September 14). Retrived 27 February 2023.

* What is Consumer Research? on JSTOR. https://www.jstor.org/stable/2489249 (n.d.). Retrieved 28 February 2023

* What Is Consumer Behavior in Marketing? - Factors, Model & Definition. https://study.com/academy/lesson/what-is-consumer-behavior-in-marketing-factors-model-definition.html (n.d.). study.com. Retrieved February 28, 2023.

* LightGBM Classifier. https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMClassifier.html

* Decision Tree Classifier. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

* Random Forest Classifier. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html