Udacity Machine Learning Engineer Nanodegree Programme

Oluwaseyi Emmanuel Ogunnowo

Capstone Project (Starbucks Corporation Challenge)

1. DEFINITION

1.1. Project Overview

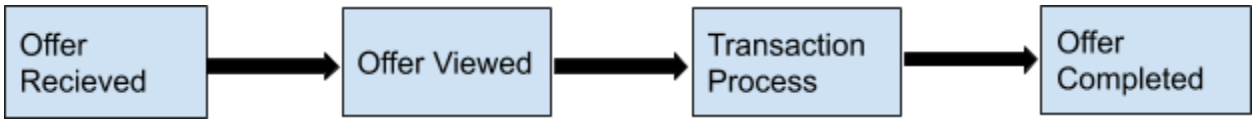
This project aims to identify consumers who will respond positively to a sales promotion offer made by starbucks. Starbucks dispenses messages to customers containing a type of offer. Offers made by Starbucks in these messages include the following: buy-one-get-one (BOGO), discount offers and informational offers. Each customer receives one of these offers.

BOGO offers require that a customer spends a particular amount or purchases the required amount of items to qualify. Discount offers give customers the chance to purchase certain items at lesser value. The last category which is informational, is not necessarily an offer but merely gives information about certain products. Customers and consumers alike receive these offers through a variety of channels. These channels include: web, email, mobile and social media.

Generally, companies make offers or deploy ads to their customers for a number of reasons, this action is referred to as sales promotion. Some of the reasons for sales promotion include: increase sales, gain market share from competition, gain new distribution opportunities and so on (Blattgerg & Briesch, 2010). Sale promos demand a significant amount of resources and are only successful when the objectives are achieved.

1.2. Problem Statement

Starbucks like any other organisation sends out promos to reward loyal customers, maximize profit and also gain new customers. The flow of a pertinent offer is as follows:



In line with the above, some consumers do not complete the offer. In certain cases, offers are only received and not viewed while others are viewed and no transaction is conducted. There are also cases where transactions not impacted by offers are conducted. The preceding scenarios which depict incomplete offers, points to the inability to match customers with offers they are prone to completing. As such, this project sets determine if a particular customer will respond to an offer or not.

1.3. Metrics

The term evaluation metrics denotes a tool that measures the performance of a model (Hossin & Sulaiman, 2015). For this project, a confusion matrix will be generated detailing the True Positives (TP), False Positives (FP), False Negatives (FN), True Negatives (TN). The purpose of this is to depict the number of correct and wrong predictions made by the model. After this, the model's accuracy score will be generated. Accuracy measures the ratio of correct predictions

over the total number of instances evaluated. Precision and Accuracy are of utmost concern when classification problems are faced in Machine Learning (Visa, Ramsay, Ralescu, & Knaap, 2011). While this project portrays the accuracy scores of all models used, this is not sufficient. As such it is important to depict the number of correct or true predictions against false ones. This is the reason for the Confusion Matrix as used in this project.

2. ANALYSIS

2.1. Data Exploration

This project makes use of three distinct datasets, namely: portfolio, profile and transcript.

Portfolio: This dataset contains details on offers made by Starbucks to its consumers. Size: 10 rows and 6 columns. Features in this dataset include:

a. difficulty (int) - minimum required spend to complete an offer
b. reward (int) - reward given for completing an offer
c. id (string) - offer id
d. offer_type (string) - type of offer ie BOGO, discount, informational
e. duration (int) - time for offer to be open, in days
f. channels (list of strings)

	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 portfolio dataset among other things contains the three different offers made by Starbucks. These include BOGO (buy-one-get-one), discount offers, in which a user or customer gains a fraction of the original amount spent and informational, which merely gives information on other products and so on. Each offer has a duration or validity period before it expires. Offers are contained in the *offer_type* dataset. There are also requisite amounts a user must spend before qualifying for a particular offer. These amounts are contained in the *difficulty* column. Also

included in the dataset is the *channels* column which stipulates the medium by which consumers receive offers. Each offer had an *Id*, a categorical variable that identifies the offer.

There are no abnormalities in the dataset.

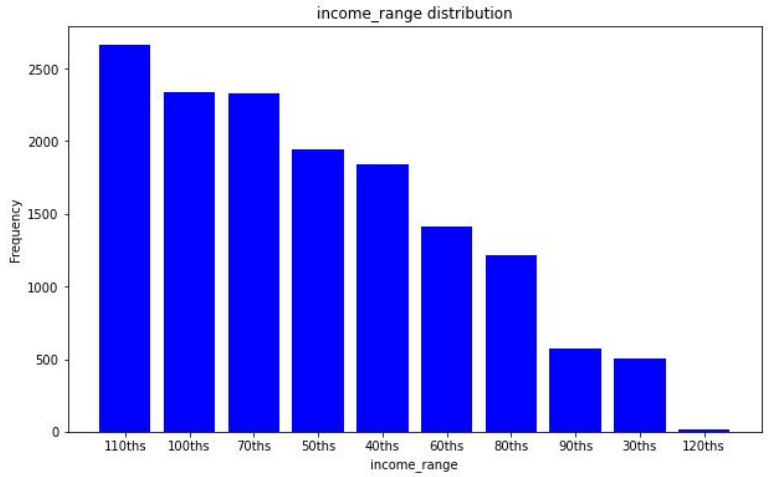
Profile: This dataset contains details of customers or consumers of Starbucks' products. Size: 17000 rows and 5 columns.

a. Features in this dataset include:
b. id (str) - customer id
c. age (int) - age of the customer
d. became_member_on (int) - date when customer created an app account
e. gender (str) - gender of the customer (note some entries contain 'O' for other rather than M or F)
f. income (float) - customer's income

	gender	age	id	became_member_on	income
0	NaN	118	68be06ca386d4c31939f3a4f0e3dd783	20170212	NaN
1	F	55	0610b486422d4921ae7d2bf64640c50b	20170715	112000.0
2	NaN	118	38fe809add3b4fcf9315a9694bb96ff5	20180712	NaN
3	F	75	78afa995795e4d85b5d9ceeca43f5fef	20170509	100000.0
4	NaN	118	a03223e636434f42ac4c3df47e8bac43	20170804	NaN

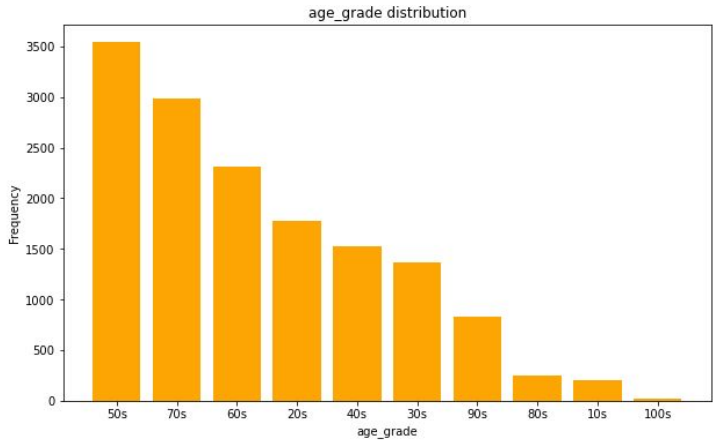
The profile dataset contains information on customers including gender, age, the date in which memberships commenced, income of consumers and their respective IDs. The dataset contains null values which can affect the accuracy of the analysis if they are not treated. It is noted that the NaN values in the gender and income column have 118 as ages. These rows were dropped in from the dataset. This dataset is notable for its missing values.

Below are visualizations and explanations on the peculiarities of the dataset:



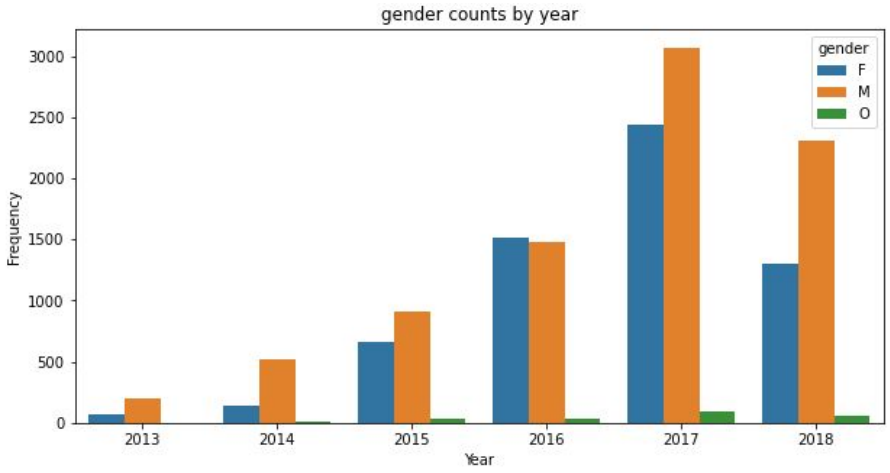
Income Distribution

The chart above visualizes the income range distribution in the dataset. Accordingly, it is seen that the highest salary range is the 110ths (i.e 110,000 upwards) with a frequency of about 2,500. From the graph, we can also identify that majority of Starbucks subscribers are high income earners, as majority of the members earn between 50,000 and 110,000.



Age Distribution

It is seen from the chart above that most Starbucks subscribers are within the 50s age bracket. Majority of Starbucks customers are within the ages of 20 and 50



Gender Distribution by Year

This graph depicts the gender distribution by year. Except for 2016, all other years had more males subscribing to Starbucks.

Transcript: This dataset contains details on the offers made to consumers of Starbucks' products. Size: 306534 rows and 4 columns. The Features in this dataset include:

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

	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

The third dataset displays information on the offers received, completed and transactions made. The dataset contains the ID of Starbucks customers (with IDs of those with Age 118 excluded), the event that has taken place (if an offer was received, viewed or transactions were made). The dataset also contains the ID of the type of offer they have received. The next step at analysing this data was to divide the dataset into two distinct sets namely: offer_id (containing events: offer_recieved, offer_viewed, completed) and transaction_df (containing transaction events).

Offer_df: Here the unique values in the events column have been one hot encoded

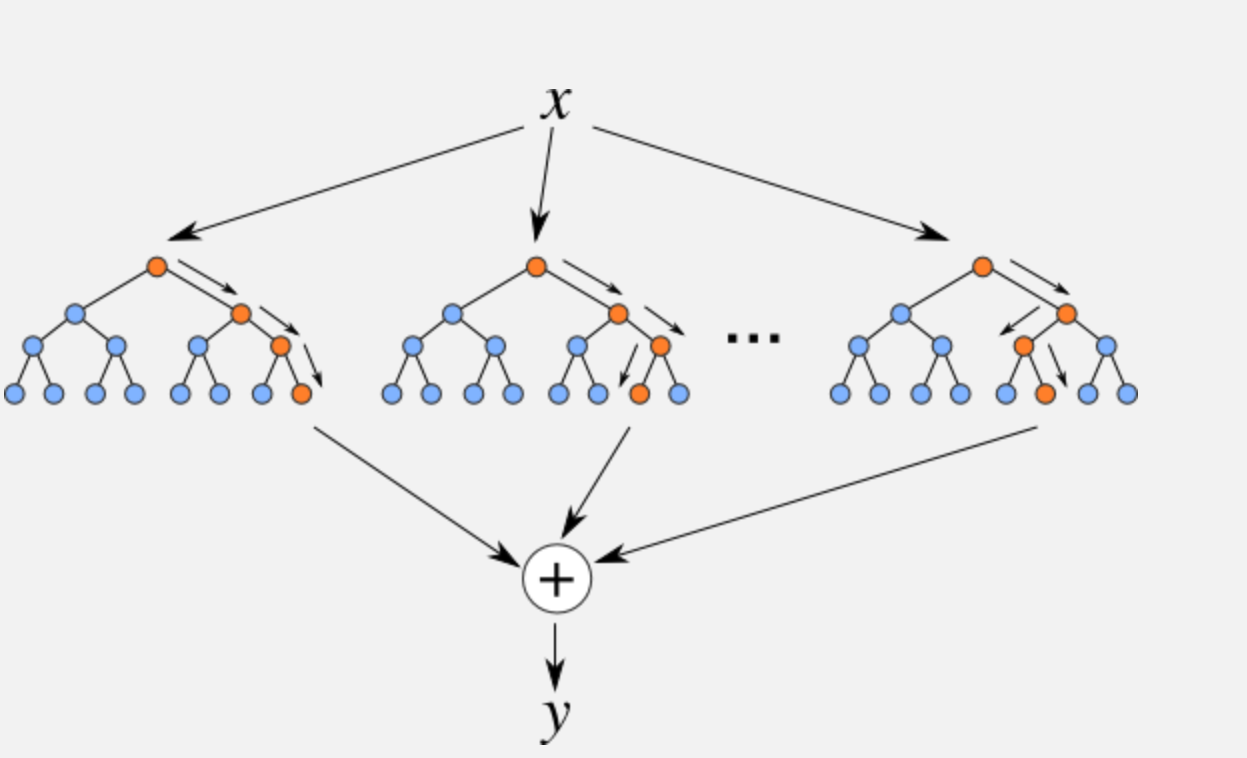
	customer_id	offer_id	time	offer received	offer viewed	offer completed
0	78afa995795e4d85b5d9ceeca43f5fef	9b98b8c7a33c4b65b9aebfe6a799e6d9	0	1	0	0
1	e2127556f4f64592b11af22de27a7932	2906b810c7d4411798c6938adc9daaa5	0	1	0	0
2	389bc3fa690240e798340f5a15918d5c	f19421c1d4aa40978ebb69ca19b0e20d	0	1	0	0
3	2eeac8d8feae4a8cad5a6af0499a211d	3f207df678b143eea3cee63160fa8bed	0	1	0	0
4	aa4862eba776480b8bb9c68455b8c2e1	0b1e1539f2cc45b7b9fa7c272da2e1d7	0	1	0	0
...	...	...	...	...	...	...
148800	84fb57a7fe8045a8bf6236738ee73a0f	5a8bc65990b245e5a138643cd4eb9837	714	0	1	0
148801	abc4359eb34e4e2ca2349da2ddf771b6	3f207df678b143eea3cee63160fa8bed	714	0	1	0
148802	8dda575c2a1d44b9ac8e8b07b93d1f8e	0b1e1539f2cc45b7b9fa7c272da2e1d7	714	0	1	0
148803	8431c16f8e1d440880db371a68f82dd0	fafdcd668e3743c1bb461111dcafc2a4	714	0	0	1
148804	24f56b5e1849462093931b164eb803b5	fafdcd668e3743c1bb461111dcafc2a4	714	0	0	1

Transaction_df:

	customer_id	event	time	amount
0	02c083884c7d45b39cc68e1314fec56c	transaction	0	0.83
1	9fa9ae8f57894cc9a3b8a9bbe0fc1b2f	transaction	0	34.56
2	54890f68699049c2a04d415abc25e717	transaction	0	13.23
3	b2f1cd155b864803ad8334cdf13c4bd2	transaction	0	19.51
4	fe97aa22dd3e48c8b143116a8403dd52	transaction	0	18.97
...	...	...	...	...
123952	24f56b5e1849462093931b164eb803b5	transaction	714	22.64
123953	b3a1272bc9904337b331bf348c3e8c17	transaction	714	1.59
123954	68213b08d99a4ae1b0dcb72aebd9aa35	transaction	714	9.53
123955	a00058cf10334a308c68e7631c529907	transaction	714	3.61
123956	76ddbd6576844afe811f1a3c0fbb5bec	transaction	714	3.53

2.2. Algorithms and Techniques

This project utilizes a Random Forest Classifier. The model was chosen after the benchmarking process which involved training each model with the combined dataset and choosing the one with the highest accuracy score is chosen. The Random Forest Classifier consists of a collection of classifiers or decision trees where each tree casts a vote for the label of input x (Tomasi, 2017). In other words, a Random Forest Classifier consists of different decision trees with each having the same number of nodes. It merges the decision of multiple trees to find an answer (Schott, 2019). Here is a graphical representation of the decision trees in a Random Forest Classifier



Source: Schott (2019).

When using a Random Forest Classifier, as used in this project (with entropy as a criterion), the algorithm is working behind the scenes with this formula:

$$Entropy = \sum_{i=1}^{C} -p_i * \log_2(p_i)$$

Source: Schott (2019).

2.3. Benchmarking

Benchmarking is the process of comparing models in their ability to learn patterns from a dataset. Flowing from this definition, the models/ classifiers stated above will be trained with the data and their results compared with each other. The classifier or model with the highest accuracy will be chosen. These models were imported and initialized as such:

lr = LogisticRegression(random_state=42), rfc = RandomForestClassifier(random_state=42), gbc = GradientBoostingClassifier(random_state=42)

Each of these models were trained with data from our combined_df. The models were compared in terms of their training performance by their accuracy scores. The following are the scores of the models:

a. Logistic Regression: 0.863
b. RandomForestClassifier: 0.999
c. GradientBoostingClassifier: 0.911

The testing accuracy scores of the model were also generated and compared with each other. The following are the scores of the models:

a. LogisticRegression: 0.843
b. RandomForestClassifier: 0.925
c. GradientBoostingClassifier: 0.909

From the preceding, the RandomForestClassifier was chosen for this project as it has the highest accuracy score. This is because of the non-linear nature of the predictions that are needed to be made.

3. METHODOLOGY

3.1. Data Preprocessing

From the datasets pictorized above, there is a need for preprocessing, to prepare the features for the models. In the Portfolio dataset, there is a need to separate the values in the *Channels* column and also one hot encode them. The values in the *Offer_type* column also need to be one hot encoded. This is achieved with the **pd.get_dummies()** function.

Following the cleaning processes done on the Profile dataset as identified above, *Age grades* and *Income range* columns were created from the *Age* and *Income* columns of the dataset. The Became_member_on column which denotes the dates in which memberships began was formatted to date format with the **datetime module.** The dates taking the format of (yyyy-mm-dd) were separated into Became_member_year (Year), Became_member_month (Month), and Became_member_day (Day) columns. This was to create ample opportunity to

visualize years when memberships commenced. These columns were also One Hot Encoded with the method identified above.

3.2. Implementation and Refinement

I commenced this project by *loading and exploring the datasets* provided, viewing the first five (5) rows to know the inherent columns and data types. This was followed by *data cleaning*, which involved taking care of null values (NaN) and duplicate ones. The third step was the *data visualization*, which pictorized the data in hand and also generated insights. The next step was the data preprocessing stage which involved feature engineering, creating new features from existing ones.

The distinct datasets were merged to create a combined dataset which had in total 66501 rows and 53 columns. Numerical values in the dataset were normalized with the MinMaxScaler() method of sklearn.preprocessing. The data was also split with the train_test_split() method of sklearn.model_selection. Parameters of this spit include the following: (test_size = 0.3, random_state = 42). The test_size parameter specifies the value of the testing set, while random_state is used to avoid getting different values for the train and test sets of the data every time the data is split. The data was split into X_train, X_test, y_train and y_test sets

1. x_train: The training part of the first sequence (x)
2. x_test: The test part of the first sequence (x)
3. y_train: The training part of the second sequence (y)
4. y_test: The test part of the second sequence (y) (Stojiljković, 2020)

The models, lr, rfc and gbc were fed the train sets of the data with their accuracy scores generated. This was followed by the choice of the rfc model due to its accuracy score. The RandomForestClassifier was fed the test set. The predictions showed the need for tuning the algorithm. This was achieved with the following parameters: (n_estimators=60, criterion='entropy', random_state=42). N_estimators determine the number of trees the model must build.

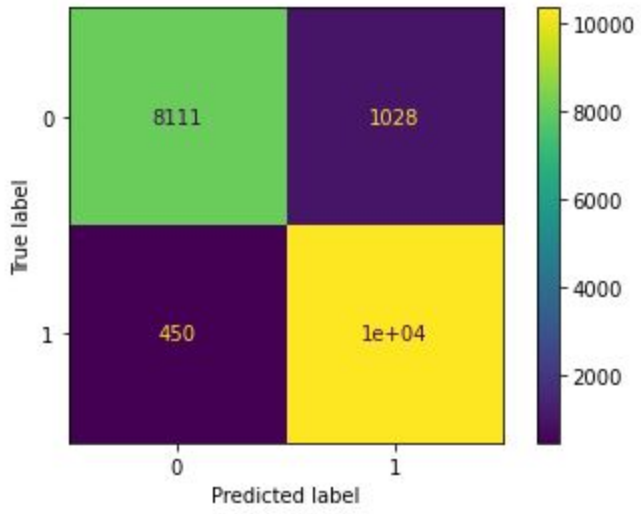
4. RESULTS

4.1. Model Evaluation, Validation and Justification

Following from the benchmarking procedures outlined in the previous sections of this report, the Random Forest Classifier model was chosen as the classifier for this project. The model was evaluated by feeding it with unseen data, (i.e. the test set: X_test). As such, the model was used to predict outputs (in this case, the response of a customer to an offer) based on this test set. The

predictions made were compared with the original outputs (y_test), and a confusion matrix was generated. Below is the diagram of the confusion matrix generated after predictions:

```
True positive =  8111
False positive =  1028
False negative =  450
True negative =  10362
```



With the preceding, I affirm that the model performed well and generated the needed predictions based on the data given to it.

Each of these models were trained with data from our combined_df. The models were compared in terms of their performance by their accuracy scores. The Random Forest Classifier had a test accuracy score of 0.925. This is in comparison with the Logistic Regression and Gradient Boosting Classifier models which had 0.843 and 0.909 accuracy scores respectively.

5. CONCLUSION

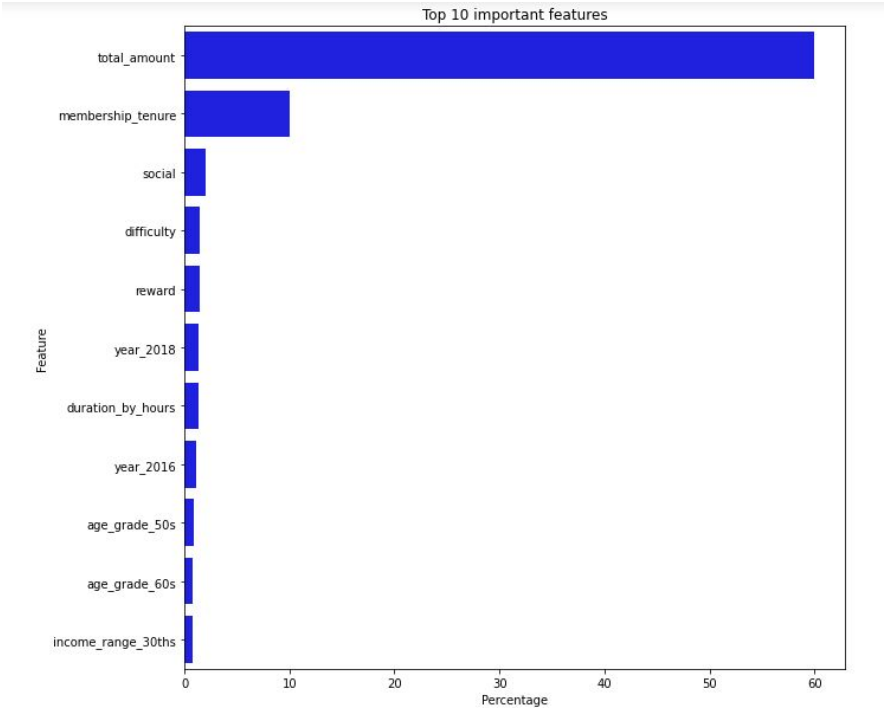
5.1. Free-Form Visualization

In the nearest future, Starbucks will disseminate more offers to customers. However doing this without insight may be problematic and even incur significant losses in resources and revenue. To avert this, I identified the offers that were most successful and those that were least successful. Below is a table and bar chart depicting these

	offer_id	count	success_percentage	offer_type
9	fafdcd668e3743c1bb461111dcafc2a4	6652	75.20	discount
1	2298d6c36e964ae4a3e7e9706d1fb8c2	6655	72.28	discount
8	f19421c1d4aa40978ebb69ca19b0e20d	6576	60.74	bogo
5	5a8bc65990b245e5a138643cd4eb9837	6643	54.72	informational
7	ae264e3637204a6fb9bb56bc8210ddfd	6683	53.25	bogo
4	4d5c57ea9a6940dd891ad53e9dbe8da0	6593	49.98	bogo
2	2906b810c7d4411798c6938adc9daaa5	6631	47.29	discount
6	9b98b8c7a33c4b65b9aebfe6a799e6d9	6685	47.28	bogo
0	0b1e1539f2cc45b7b9fa7c272da2e1d7	6726	41.85	discount
3	3f207df678b143eea3cee63160fa8bed	6657	36.76	informational

Each offer has an index figure. In terms of response, the most successful offer is the discount offer with ID (fafdcd668e3743c1bb461111dcafc2a4) with a 75% success rate. The least successful offer was the informational offer with ID (3f207df678b143eea3cee63160fa8bed) and rate of 36.76%. From the preceding we can tell that customers tend to respond to discount offers more. This is closely followed by bogo offers and then informationals.

However, there is also a need to understand the factors that influence a customer's decision regarding an offer. The following picturizes the most influential factors, in accordance with the features in the combined dataset.



The graph above gives detail on the top ten most influence features which determine if a customer will complete an offer or not. However, for the purpose of brevity, the top four features will be discussed.

Total Amount: The total amount of money spent by a customer will determine to a large extent, if they will respond to an offer or no

membership tenure: The membership tenure of customers also determines if they will respond to an offer or not

social: Social is one of the channels by which customers receive offers. from the visualization above, customers who received offers through social channels are more likely to respond to offers

difficulty: difficulty denotes the minimum amount required to be spent before an offer can be completed. The chart shows that difficulty influences a customers decision to respond to an offer or not

5.2. Reflection

This project helped me hone my analytical skills as a budding machine learning engineer. Through this project I learnt a fundamental truth in Machine Learning, it is not only about the coding but thinking and fashioning out a solution to the problem at hand. This is my first Udacity Nanodegree programme and it was a pleasure to work on this project.

5.3. Improvements

The project makes predictions concerning the response of Starbucks customers to offers made by the company. While the Random Forest Classifier used in the project achieves this aim, the predictions are merely short-term as it does not consider the long-term effects of economic factors which to a large extent influence customer choice. As such, certain offers may only be relevant in the future. Furthermore, other classifiers could be used to generate more robust predictions.

References

Blattberg, R. C. and Briesch R., A.(2010). Sales Promotions. In Ö. Özer & R. Phillips, (Eds.). (2012). The Oxford handbook of pricing management. Oxford University Press.

Hossin, M., & Sulaiman, M. N. (2015). A review on evaluation metrics for data classification evaluations. International Journal of Data Mining & Knowledge Management Process, 5(2),

Olson, R. S., La Cava, W., Orzechowski, P., Urbanowicz, R. J., & Moore, J. H. (2017). PMLB: a large benchmark suite for machine learning evaluation and comparison. BioData mining, 10(1), 1-13.

Schott, M. (2019). Random Forest Algorithm for Machine Learning. Retrieved from https://medium.com/capital-one-tech/random-forest-algorithm-for-machine-learning-c4b2c8cc9feb

Stojiljković, M. (2020). Split Your Dataset With scikit-learn's train_test_split(). Retrieved from https://realpython.com/train-test-split-python-data/

Tomasi, C. (2017). Random forest classifiers.

Visa, S., Ramsay, B., Ralescu, A. L., & Van Der Knaap, E. (2011). Confusion Matrix-based Feature Selection. *MAICS*, *710*, 120-127.