Machine Learning Engineer Nanodegree

Sharmin Ahmed

October 26, 2020

Starbucks Capstone Project Report

Project Overview

As of early 2020, Starbucks company operates over 30,000 locations worldwide in more than 70 countries[1], being the largest coffee house chain. This only became possible because of Starbucks' customer-centric culture. In 2011, with the launching of the Starbucks app, the company has granted its customers the ability to order, pay, and collect their beverages without the torture of queuing. Once every few days, Starbucks would send out an offer to users of the mobile app. An offer can be merely an advertisement for a drink or an actual offer such as a discount or BOGO (buy one get one free). Some users might not receive any offers during certain weeks.

By effectively introducing these a referral program, and a reward system, Starbucks is attracting millions of customers. For company profit and building a better user experience, Starbucks has to customize the offers for the users. Because, if everyone gets the same offer they might not use them. They offers can be like giving more

[1] https://en.wikipedia.org/wiki/Starbucks

discounts to heavy users, free offers, limited discounts, etc.

As I, myself, am a coffee lover this problem piqued my interest. Addressing this problem will give me insights into marketing schemes and the pattern of users' behavior for an offer.

Problem Statement

The problem is to determine what offers to send to which customers. Determination can be based on their history, such as previous purchases, gender, age groups, income range, etc. The project's goal is to use machine learning models to predict if a user will complete an offer within the given timeframe. So that based on that prediction it can be analyzed what kind of offers to send customers.

Evaluation Metrics

Since this is a classification problem[2], the performance of the models will be evaluated using following evaluation matrices:

- **Precision:** It is the number of true positives divided by the number of true positives, plus the number of false positives.
- **Recall:** It is the number of true positives divided by the number of true positives plus the number of false negatives.

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

[2] https://www.kdnuggets.com/2020/04/performance-evaluation-metrics-classification.html

- **F-1 score:** F-1 score is a way of combining the precision, and it is defined as the harmonic mean of the model's precision and recall.

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Analysis

I. Data Exploration and Cleaning

We would explore dataset one by one and from the insights clean the data accordingly

1.a Insights from 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
2	[web, email, mobile]	0	4	3f207df678b143eea3cee63160fa8bed	informational	0
3	[web, email, mobile]	5	7	9b98b8c7a33c4b65b9aebfe6a799e6d9	bogo	5
4	[web, email]	20	10	0b1e1539f2cc45b7b9fa7c272da2e1d7	discount	5

- There are three types of offers : 'bogo', 'informational' and 'discount'. These data would be better for modeling, if applied one hot encoding method.
- ‘channels’ column needs to be processed. Information is be extracted and also applied with one hot encoding method.
- ‘duration' column should be converted into hours as 'time' in transcript.json is also in hours.
- id needs to be renamed to offer_id to avoid confusion.

1.b After cleaning portfolio:

	difficulty	duration_hour	offer_id	reward	web	email	mobile	social	bogo_offer	informational_offer	discount_offer
0	10	168	ae264e3637204a6fb9bb56bc8210ddfd	10	0	1	1	1	1	0	0
1	10	120	4d5c57ea9a6940dd891ad53e9dbe8da0	10	1	1	1	1	1	0	0
2	0	96	3f207df678b143eea3cee63160fa8bed	0	1	1	1	0	0	1	0
3	5	168	9b98b8c7a33c4b65b9aebfe6a799e6d9	5	1	1	1	0	1	0	0
4	20	240	0b1e1539f2cc45b7b9fa7c272da2e1d7	5	1	1	0	0	0	0	1

2.a Insights from 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

- The dataset has 2175 missing values on: 'gender', 'income' column. Corresponding age value of those records are set to default 118. These missing values are cleaned.
- There are three 'gender' categories- M (8484), F (6129) and O (212). O could be associated with missing value records.
- Income is grouped into 'average', 'above_average', 'high' buckets and ‘age’ is grouped into 'young_adult', 'middle_aged', 'old', 'elderly' buckets for model purpose.
- membership_days and member_category: 'regular', 'long_term' are calculated from ‘became_member_on column’.
- Id is renamed to customer_id.

2.b After cleansing profile:

	age	became_member_on	gender	customer_id	income	membership_days	age_group	income_range	member_category
1	55.0	2017-07-15	F	0610b486422d4921ae7d2bf64640c50b	112000.0	1198	old	high	regular
3	75.0	2017-05-09	F	78afa995795e4d85b5d9ceeca43f5fef	100000.0	1265	elderly	high	regular
5	68.0	2018-04-26	M	e2127556f4f64592b11af22de27a7932	70000.0	913	elderly	above_average	regular
8	65.0	2018-02-09	M	389bc3fa690240e798340f5a15918d5c	53000.0	989	elderly	above_average	regular
12	58.0	2017-11-11	M	2eeac8d8feae4a8cad5a6af0499a211d	51000.0	1079	old	above_average	regular

3.a Insights from transcript:

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

- ‘person’ is renamed to ‘customer_id’.
- The dataset has no missing values.
- The 'value' column is a dictionary and needs to be processed. offer_id and amount is extracted to newly created corresponding columns.
- All events in this dataset are: 'transaction', 'offer received', 'offer viewed' and 'offer completed'. Offers are processed with one hot encoding and transaction event are extracted to transaction_df
- other events excluding transaction are extracted to offer_df

.

3.b after cleansing transcript: transaction_df and offer_df are created:

transaction_df:

	customer_id	time	amount
12654	02c083884c7d45b39cc68e1314fec56c	0	0.83
12657	9fa9ae8f57894cc9a3b8a9bbe0fc1b2f	0	34.56
12659	54890f68699049c2a04d415abc25e717	0	13.23
12670	b2f1cd155b864803ad8334cdf13c4bd2	0	19.51
12671	fe97aa22dd3e48c8b143116a8403dd52	0	18.97

offer_df:

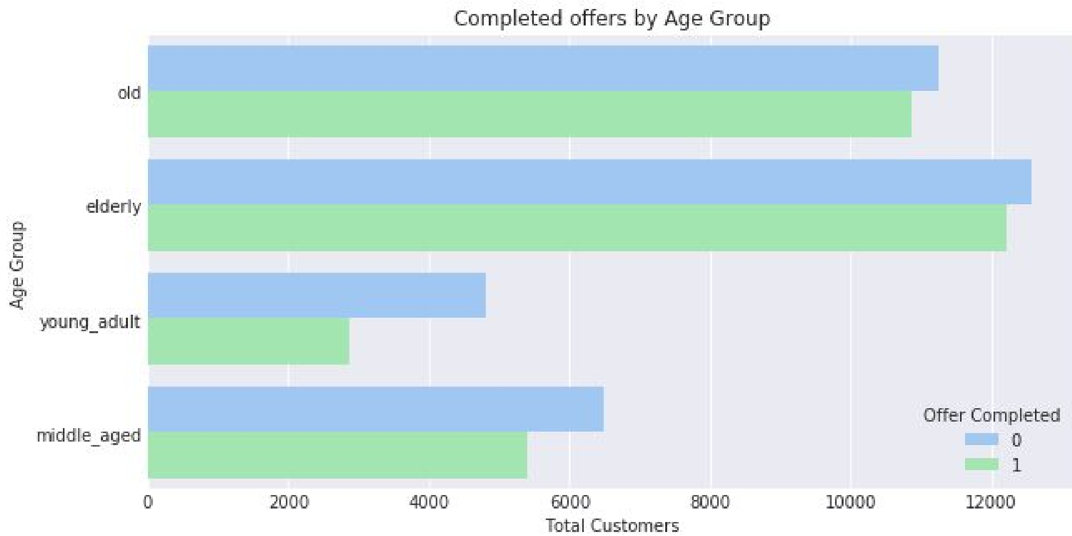
	customer_id	offer_id	time	offer completed	offer received	offer viewed
306497	a6f84f4e976f44508c358cc9aba6d2b3	2298d6c36e964ae4a3e7e9706d1fb8c2	714	1	0	0
306506	b895c57e8cd047a8872ce02aa54759d6	fafdcd668e3743c1bb461111dcafc2a4	714	1	0	0
306507	8dda575c2a1d44b9ac8e8b07b93d1f8e	0b1e1539f2cc45b7b9fa7c272da2e1d7	714	0	0	1
306509	8431c16f8e1d440880db371a68f82dd0	fafdcd668e3743c1bb461111dcafc2a4	714	1	0	0
306527	24f56b5e1849462093931b164eb803b5	fafdcd668e3743c1bb461111dcafc2a4	714	1	0	0

4. merged_data:

Once the cleaning is done, the datasets: portfolio, profile, offer_df and transaction_df are merged into a big data frame in order to analyze the relation between customer behaviour on an offer. **‘money_spent’** and **‘offer_completed’** attributes are also calculated in the process.

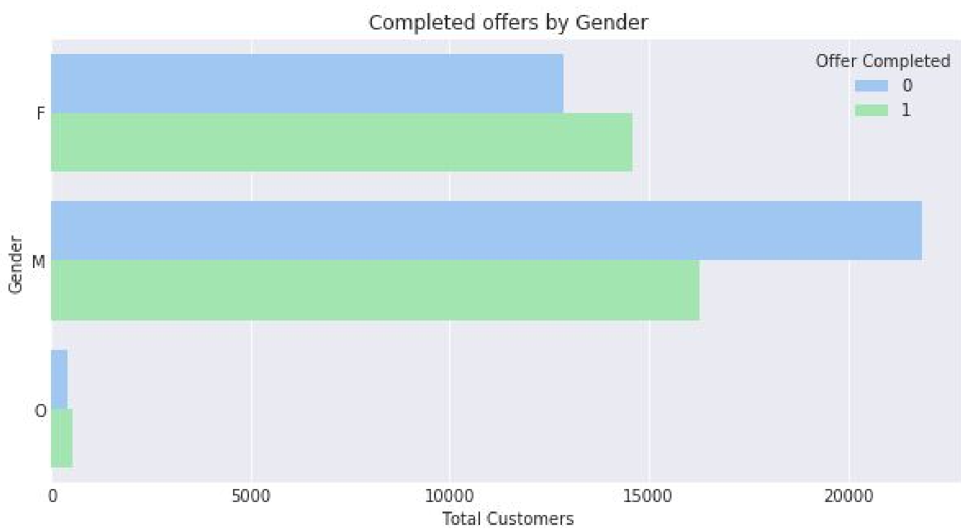
II. Exploratory Visualization

1. **Offer completion by age_group:**



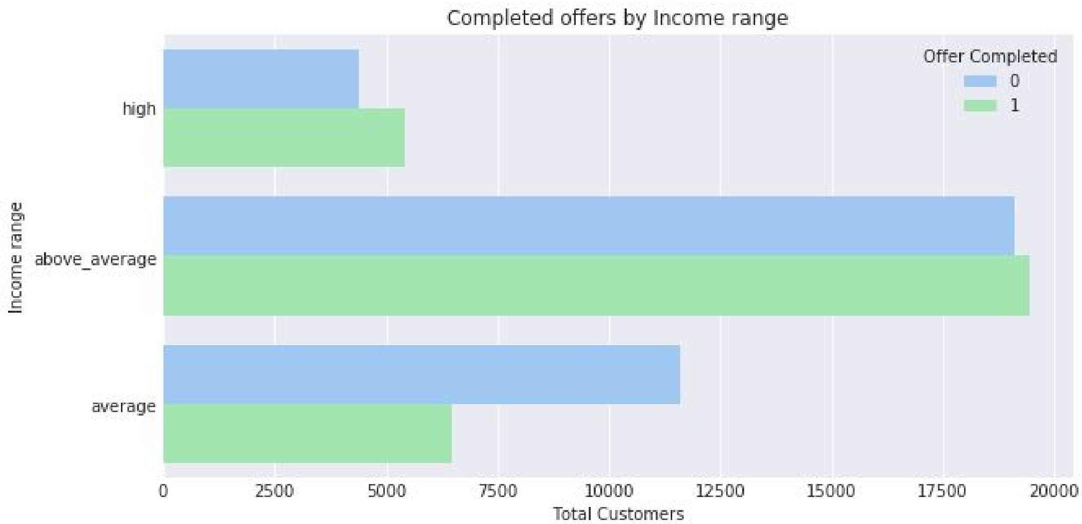
age_group are categorized into - **young_adult (17-30) , middle_aged (31-45), old (46-60) and elderly (60-105)**. From the visualization we can see 45- 105 aged people completed maximum offers.

2. **Offer completion by Gender:**



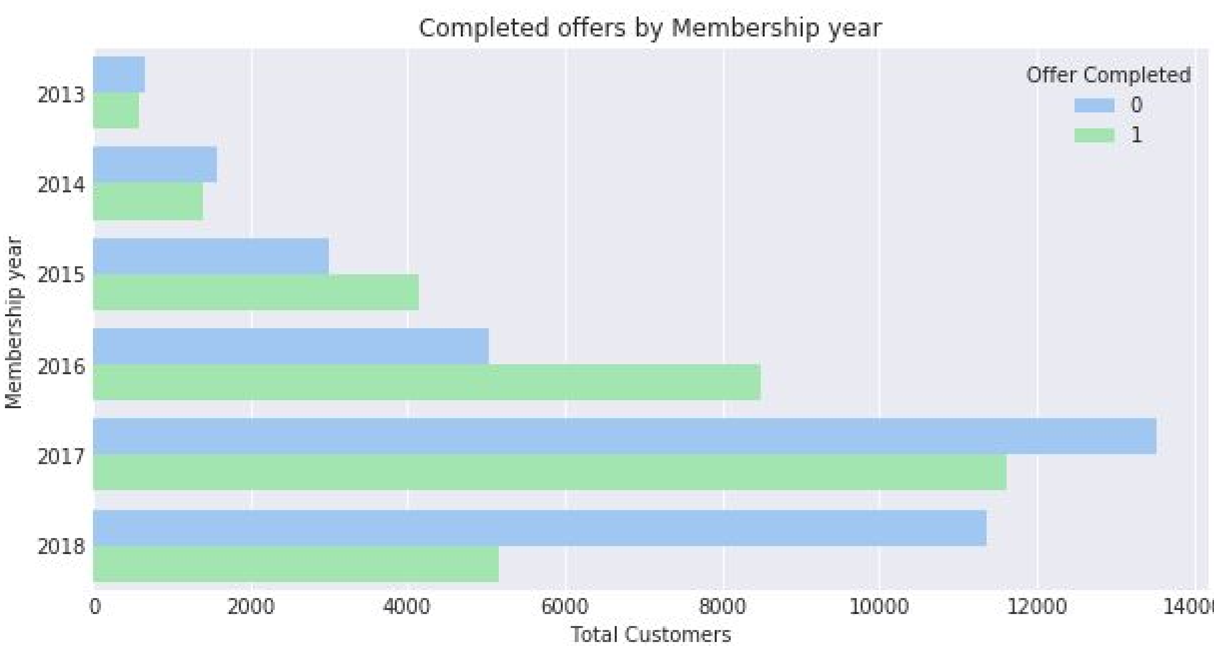
Male numbers are more in the dataset. Hence, Male completed more offers also. Although not completed offer numbers were also higher in Males.

3. Offer completion by Income range:



Income range are categorized into **average (29999-50000), above_average (50001-90000), and high (90001-120001)** category. People with above_average income completes more offer.

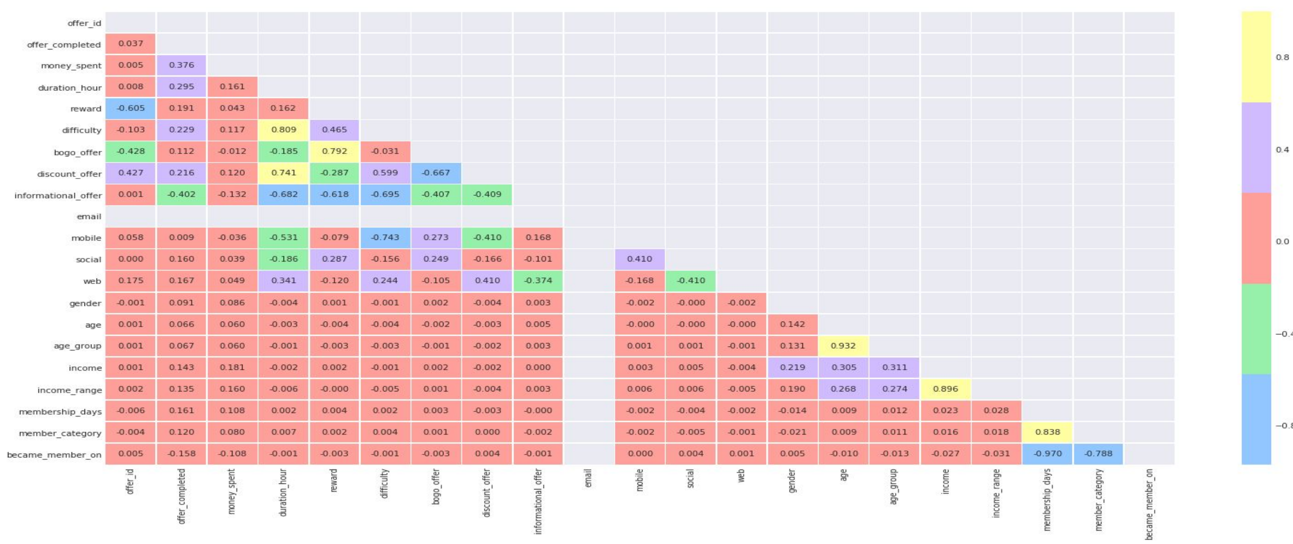
4. Offer completion by Membership year:



Most members are from 2016-2018. 2016-2017 members have higher offer completion rates. New members are more active.

III. Data-preprocessing

- **Categorical encoding:** For model purpose, the categorical valued columns are encoded with numerical values. The prospective columns are : gender, age_group, income_range, member_category, offer_id.
- **Normalizing features:** The numerical valued columns 'money_spent', 'duration_hour', 'reward', difficulty', 'income', 'membership_days' are normalized for better fitting.
- **Correlation matrix:** A heatmap of the correlation matrix of merged_data is plotted to select best features.



- **Feature selection:** After visualization, 'money_spent', 'duration_hour', 'reward', 'difficulty', 'bogo_offer', 'discount_offer', 'informational_offer', 'social', 'web', 'member_category', 'became_member_on', 'gender', 'age_group', 'income_range' attributes are selected as independent variables.
- **Target Selection:** For dependent variable 'offer_completed' attribute is selected.
- **Missing_values:** Any missing values are amputated.
- **train-test split:** whole dataset are splitted into train and test sets on a 70:30 ratio.

IIII. Algorithm and Techniques

Benchmark Model: In real-world applications, logistic regression is the most commonly used for addressing classification problems[3]. That's why, I will build a logistic regression model and use it as a benchmark. **Logistic regression** predicts categorical outcomes (binomial / multinomial values of y) based on the concept of probability. Logistic Regression uses a cost function, defined as the 'Sigmoid function' :

$$f(x) = \frac{1}{1 + e^{-(x)}}$$

Models for comparison: We will explore three more models for this project:

Random Forest Classifier, Support Vector Classification and Gradient Boosting Classifier.

1. **Random Forest Classifier:** It is an ensemble tree-based learning algorithm. The Random Forest Classifier is a set of decision trees from a randomly selected subset of training set. It aggregates the votes from different decision trees to decide the final class of the test object. It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.
2. **Support Vector Classification:** The SVCs aim to find the best hyperplane (also called decision boundary) that best separates (splits) a dataset into two classes (binary classification problem). Depending on the number of the input features, the decision boundary can be a line (if we had only 2 features) or a hyperplane if we have more than 2 features in our dataset.
3. **Gradient Boosting Classifier:** Gradient boosting models are effective at classifying complex dataset. It uses the AdaBoosting method combined with weighted minimization, after which the classifiers and weighted inputs are recalculated. The objective of Gradient Boosting classifiers is to minimize the loss, or the difference between the actual class value of the training example and the predicted class value.

---

[3] **Yang, Y. and Loog, M., 2018. A benchmark and comparison of active learning for logistic regression. *Patte#rn Recognition*, *83*, pp.401-415.**

Model Tuning: For the parameter tuning, GridSearchCV is used which implements a "fit" and a "score" method. Models were tuned and best fitted models were tested with test_set.

IV. Result

Models are tuned using GridSearchCV and then precision, recall and f1-score are measured.

Logistic Regression: Tuned penalty and C parameters were l1 and 100 respectively. The best fitted model was tested with test_set and precision, recall and f1-score were all measured as average 0.88.

Logistic Regression

	precision	recall	f1-score	support
0	0.86	0.92	0.89	10541
1	0.90	0.83	0.87	9410
avg / total	0.88	0.88	0.88	19951

Random Forest Classifier: Tuned max_depth, min_samples_leaf and n_estimators parameters were 5, 2 and 100 respectively. The best fitted model was tested with test_set and precision, recall and f1-score were all measured as average 0.90. The result was higher than benchmark threshold 0.88.

Random Forest Classifier

	precision	recall	f1-score	support
0	0.94	0.87	0.90	10541
1	0.87	0.94	0.90	9410
avg / total	0.90	0.90	0.90	19951

Support Vector Classification (SVC): Tuned C and gamma parameters were 1000 and 1 respectively. The best fitted model was tested with test_set and precision, recall and f1-score were all measured as average 0.90. The result was higher than benchmark threshold 0.88.

Support Vector Classification

	precision	recall	f1-score	support
0	0.89	0.92	0.90	10541
1	0.91	0.87	0.89	9410
avg / total	0.90	0.90	0.90	19951

Gradient Boosting Classifier: Tuned parameters were learning_rate: 0.1, min_samples_spit: 2, min_samples_leaf: 3, max_depth: 5, subsample: 0.95, n_estimators: 100. The best fitted model was tested with test_set and precision, recall and f1-score were all measured as average 0.92. The result was higher than benchmark threshold 0.88.

Gradient Boosting Classifier:

	precision	recall	f1-score	support
0	0.94	0.90	0.92	10541
1	0.89	0.93	0.91	9410
avg / total	0.92	0.92	0.92	19951

V. Model Comparison

The benchmark model’s precision, recall, f1-score were 0.88. RandomForest and SVM performed a little better achieving 0.90. The best performance was given by **Gradient Boosting Classifier** : 0.92.

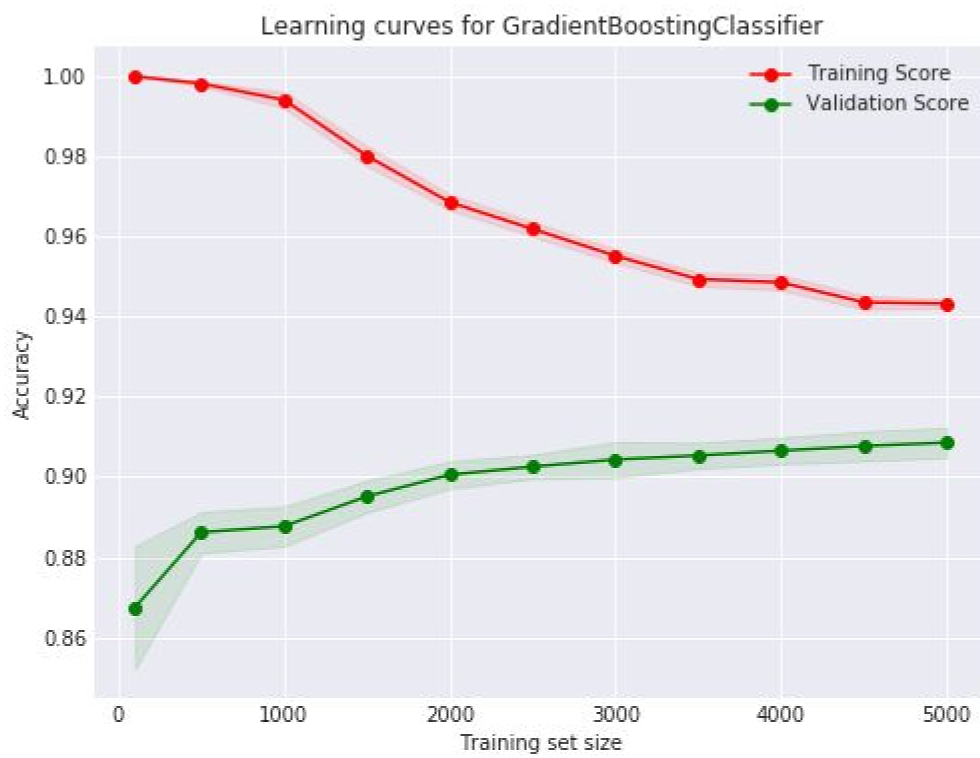
Performance of the models

Model	Precision	Recall	f1-score
Logistic Regression (Benchmark)	0.88	0.88	0.88
Random Forest Classifier	0.90	0.90	0.90
SVM Support Vector Classification	0.90	0.90	0.90
Gradient Boosting Classifier	0.92	0.92	0.92

VI. Robustness of Gradient Boosting

To analyze how the machine learning models perform as overall, one tool that can help us is the learning curve. The learning curve provides an overall look of how the model performs and how it will generalize to data that it has not seen before.

The following plot is ideally the desired learning curve of a machine learning model. The validation score of Gradient Boosting Classifier converges to a similar value of that of the training score. This proves the robustness of the model.



VII. Conclusion

Reflection

Summary of the work in this project:

1. The three datasets were explored
2. From the insights of the dataset, they were cleaned
3. Three datasets were combined and money_spent in transaction, and offer completed and not completed were separated.
4. Pre-process input data: categorical encoding, normalizing, selecting features for modeling.
5. Developing machine learning models and tuning with GridSearchCV.
6. Comparison of models and choosing the best one for the problem.

Most time consuming and difficult part of this project was handling the corner cases for constructing the merged data set. I also took a large amount of time to accurately categorize the offers that were only viewed but not completed. During model parameter tuning, I faced some problems as computation power was not sufficient to tune all the parameters. In the end, the experience to handle a real world problem was worthwhile.

Improvement

For further improvement :

- Considering keeping the users' profession in data would prove to be a better feature.
- Money spent by users is a better feature than income.
- The models can be more fine tuned with sufficient computation power.
- Other models like Neural Network can be applied to extract complex features.