AWS Machine Learning Engineer Nanodegree

Capstone Project Report

Angel Igareta

Abstract

This project, part of a Udacity Nanodegree course on SageMaker, focused on understanding customer behavior within the Starbucks rewards mobile app. The goal set was to predict the likelihood of a customer completing an offer after viewing it, using their demographic information and historical interactions. A LightGBM model was developed for this purpose, demonstrating a high performance with an accuracy of 68% and a high recall rate of 83%.

The model provided valuable insights into the factors influencing offer completion. For instance, moderate rewards, long-term membership, and the social channel were found to increase the likelihood of offer completion. Middle-income customers and those over 40 were more likely to complete offers, especially those with increased rewards. Easier offers were also more likely to be completed.

In summary, the model proved to be a powerful tool for capturing a high percentage of offer completions, making it a valuable asset for refining marketing strategies and optimizing promotional offers. This project showcases the potential of machine learning, specifically SageMaker, in enhancing customer engagement and driving business growth.

Definition

Project Overview

The focal point of this project is the analysis of customer behavior within the Starbucks rewards mobile app. The aim is to comprehend the reaction of customers to various promotional offers, taking into account their unique demographics and historical interactions. This understanding will enable Starbucks to fine-tune its marketing strategies, boost customer engagement, and augment revenue.

The issue at hand is the recognition of demographic groups that exhibit a positive response to specific promotional offers from Starbucks. Given the variety of offers Starbucks sends to its rewards mobile app users, the response rate can differ among customer segments. The task is to delve into customer demographics and preferences, utilizing these insights to predict the probability of a user completing a promotional offer post-viewing.

Proposed Solution

To tackle the issue, a machine learning model will be developed to predict the probability of a user completing a promotional offer after viewing it, taking into account their demographic details and past interaction history. The model will utilize customer data to determine which specific promotional offer features will be most effective for a particular customer segment.

A heuristic benchmark will be set by examining the historical response rate data. While this rule-based approach offers basic insights into the effectiveness of various offers for each customer segment, it lacks the precision and refinement of a machine learning model that considers a wider range of features and individual user interaction history.

Project Methodology

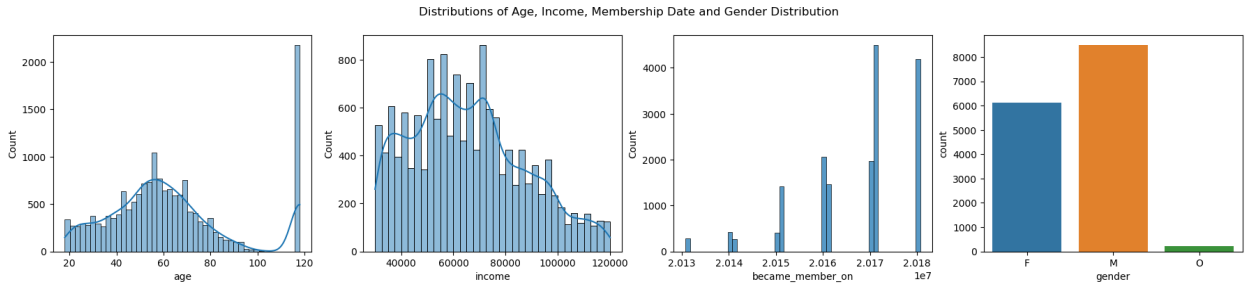
The methodology for this project involves several crucial steps. Initially, the data will be cleaned, preprocessed, and formatted. Subsequently, exploratory data analysis will be conducted to identify trends, correlations, and potential features for the machine learning model. This will be followed by feature engineering to enhance the model's predictive performance. SageMaker AutoGluon will be employed for automated model selection, comparing multiple algorithms on the validation set to identify the optimal model. This model will be trained and its hyperparameters fine-tuned using SageMaker's capabilities. The model's performance will be assessed on the test set and compared to a benchmark model. Lastly, the findings will be discussed, offering insights into customer behavior and their response to different offers.

Data Exploration and Preprocessing

The phase of data exploration and preprocessing entailed the analysis and cleaning of three datasets provided for the challenge. These datasets encompassed information about rewards program users, offers, and user interaction events.

Profile Dataset

The user profile dataset comprised demographic details for 17,000 rewards program users, including age, date of membership initiation, and income. The mean age was 62.53 years, and the mean income was $65,404. The dates of membership initiation spanned from 2013 to 2018.

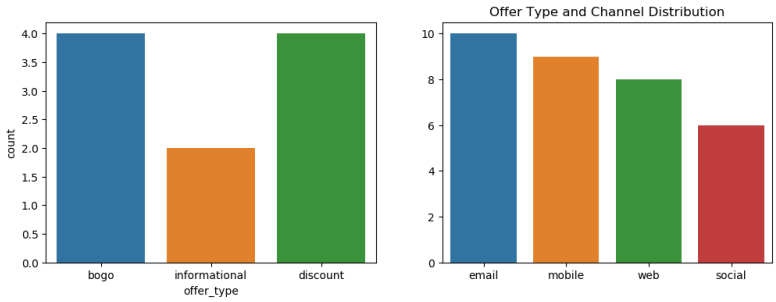


Upon examining the user profile dataset, we observed a substantial number of customers with the age listed as '118', which we inferred as a default value. These values were substituted with NULL. We also discovered that when one of the primary predictors (age, income, gender) was absent, all of them were absent. These missing values, constituting 11% of the population, were eliminated.

For feature engineering, the 'became_member_on' column was converted into 'membership_days', denoting the number of days since the customer initiated the membership. This conversion was done to facilitate model comprehension. The 'gender' column was one-hot encoded to accommodate models that do not permit categorical features. The 'age' feature was categorized into six buckets ['18-24', '25-34', '35-44', '45-54', '55-64', '65+'] to prevent model overfitting.

Portfolio Dataset

The offer portfolio dataset included details of 10 offers dispatched during a 30-day trial period, encompassing rewards and offer types (BOGO, discount, and informational).



To prepare the features for prediction, the 'channel' column was converted from an array format into several boolean columns, each representing a distinct channel. This conversion enabled us to analyze the predictive power of each channel. The 'offer type' columns were also transformed into boolean dummy columns.

Transcript Dataset

The user interaction transcript dataset included an event log of 306,534 user interactions, encompassing timestamps, transaction amounts, and rewards. The average transaction amount was $12.78, with a standard deviation of $30.25. The rewards ranged between 2 and 10, with an average value of 4.9 and a standard deviation of 2.89.

From this dataset, we designed two metrics for experimentation:

1. **Likelihood of viewing an offer:** This metric aids in understanding if customers are likely to view the offers dispatched to them.
2. **Likelihood of completing an offer, given it was viewed:** This metric, by considering only those instances where the customer has viewed the offer, aids in analyzing the effectiveness of offers, conditional on them being viewed.

During this process, we discovered that some customers received the same offer multiple times, with a high frequency of 17%. We deduplicated the dataset to be on a customer-offer level by taking the minimum of time and maximum amount and offering reward. Finally, we merged the customer and offer features with the target definitions into a single dataset, resulting in one row per customer and offer. The mean likelihood of viewing an offer was 90%, and the likelihood of completing it after viewing it was 37%, indicating an imbalanced dataset.

Ultimately, we chose to concentrate on the second metric: **"Likelihood of completing an offer, given it was viewed"**. This decision was driven by the metric's ability to more accurately assess offer effectiveness. By focusing on the performance of offers post-viewing, we can directly measure their impact, thereby aiding in the refinement of Starbucks marketing strategies.

Implementation

This section delves into two pivotal steps in our machine learning project: model selection and hyperparameter tuning.

Model selection is a crucial step where we identify the most suitable model that can accurately predict unseen data. In our project, we strive to find a model that can effectively predict customer responses to promotional offers based on their demographics and past interactions.

After the model selection, we proceed to hyperparameter tuning. Hyperparameters are set before the learning process commences and can significantly impact the model's performance. Through hyperparameter tuning, we experiment with different values to find the optimal configuration that enhances the model's accuracy, aiming to maximize the likelihood of a user completing a promotional offer post-viewing.

Model Selection

For automated model selection, we employed SageMaker AutoGluon. This is an automated machine learning (AutoML) and model selection tool that simplifies the process of selecting the best-performing model for a given dataset and problem. It accomplishes this by running a variety of machine learning algorithms on the data, each with different hyperparameters and architectures, and then comparing their performance on a validation set. The algorithms tested by AutoGluon include, but are not limited to, Neural Networks, Decision Trees, Random Forests, Gradient Boosting Machines, and Support Vector Machines.

Before feeding the data into the model, we performed several preprocessing steps. We initially eliminated any NaN values from our target column, 'offer_completed_after_view', and converted the column to a boolean type. We also dropped several features that were not suitable for training, such as 'became_member_on', 'age_group', 'person', 'offer_id', and 'offer_viewed'. We then divided our processed dataset into training, validation, and test sets with a ratio of 70%, 15%, and 15% respectively. This was done to ensure that our model was trained on a majority of the data, while still having a substantial amount of unseen data for validation and testing. We saved these datasets in S3 and uploaded them using the sess.upload_data() function. The datasets were saved without their column headers as SageMaker does not expect them when training.

In our AutoGluon predictor, we utilized the 'offer_completed_after_view' label and defined our problem as a binary classification problem. We set 'average_precision' as our evaluation metric due to its appropriateness for unbalanced binary classification problems and limited the training time to 30 minutes. Upon model fitting, AutoGluon evaluated 14 different models. The

'WeightedEnsemble_L2', 'NeuralNetFastAI_BAG_L1', and 'LightGBMXT_BAG_L1' emerged as the top performers, achieving validation scores of 0.64, 0.63, and 0.63 respectively. The corresponding prediction times were 4.22 seconds, 2.10 seconds, and 1.07 seconds.

For simplicity and interpretability, we decided to proceed with the 'LightGBMXT_BAG_L1' model. LightGBM is a gradient boosting framework that uses tree-based learning algorithms. It is known for its efficiency and effectiveness, with a relatively high average precision score and a lower prediction time compared to the other top models. Moreover, LightGBM is readily available in SageMaker, which makes it a convenient choice for our project. Its interpretability also allows us to gain insights into the features that are most influential in predicting customer behavior, which is valuable for our objective of optimizing Starbucks' promotional strategies.

Model Training and Hyperparameter Tuning

For the model training and hyperparameter tuning, we utilized Amazon SageMaker's built-in capabilities. As discussed in the previous section, we selected the LightGBM classification model, a gradient boosting framework that uses tree-based learning algorithms, for our task.

We first retrieved the Docker image, training script, and pre-trained model tarball for the LightGBM classification model. The pre-trained model was then fine-tuned to our specific task.

The hyperparameters for the LightGBM model were retrieved and the metric for evaluation was set to 'average_precision'. This metric is suitable for our task as it considers both precision and recall to compute the score. The following hyperparameters and range of values were explored:

- **Learning Rate:** This hyperparameter controls the step size shrinkage used in updates, making the model more robust by preventing overfitting. We explored a range from 0.01 to 0.2.
- **Number of Leaves:** This is the maximum number of leaves (end nodes) in one tree. We tried values from 2 to 50.
- **Feature Fraction:** This is the fraction of features used when fitting the model. We explored a range from 0.5 to 1, meaning that at each split in the tree, the model would consider 50% to 100% of the available features.
- **Bagging Fraction:** This is the fraction of data to be used for bagging. We explored a range from 0.5 to 1, meaning that the model would use 50% to 100% of the data for bagging at each iteration.
- **Bagging Frequency:** This is the frequency for performing bagging. We tried values from 1 to 10.
- **Maximum Depth:** This is the maximum depth of the tree. We tried values from 1 to 10.

- **Minimum Data in Leaf:** This is the minimum number of data points required in a leaf node. We tried values from 1 to 30.
- **Extra Trees:** This is a boolean parameter that, when set to True, will make the model check only one randomly-chosen threshold for each feature when evaluating node splits. We tried both True and False. We included it because it showed the best performance in the model selection step.

A SageMaker Estimator instance was created with the retrieved Docker image, training script, pre-trained model, and the defined hyperparameters. We then set up a hyperparameter tuner with the SageMaker Estimator, the objective metric name set to 'average_precision', the defined hyperparameter ranges, and the strategy set to 'Bayesian'. The Bayesian optimization strategy was chosen as it uses past evaluation results to choose the next hyperparameters to evaluate, thus making the tuning process more efficient.

The tuner was set to maximize the objective metric and was allowed to run a maximum of 20 jobs, with 3 jobs running in parallel. After the tuning job was completed, we retrieved from the logs and analyzed the hyperparameters of the first 10 combinations, to investigate if there were less complex configurations that could potentially offer similar performance.

Rank	Average Precision	Validation Time (s)	Learning Rate	Number of Leaves	Feature Fraction	Bagging Fraction	Bagging Frequency	Max Depth	Min Data in Leaf	Extra Trees
1	0.67	31.48	0.11	50	1	1	7	7	2	1
2	0.66	81.71	0.12	50	0.5	0.51	7	9	27	1
3	0.66	31.4	0.04	48	1	0.91	7	6	7	1
4	0.66	65.69	0.03	37	1	0.98	7	6	26	1
5	0.65	31.57	0.02	24	1	0.85	7	9	26	1
6	0.65	31.53	0.09	47	0.96	0.84	6	4	6	1
7	0.64	31.5	0.09	18	0.89	0.53	7	7	30	1
8	0.64	31.41	0.01	22	0.81	0.5	2	5	4	1
9	0.63	31.61	0.05	11	0.97	0.5	7	7	15	1
10	0.61	31.51	0.05	50	0.93	0.96	6	1	28	1

The selected model was configured with the following hyperparameters: a bagging fraction and feature fraction of about 1.0, bagging frequency of 7, a learning rate of roughly 0.109, maximum depth of 7, minimum data in a leaf of 2, and number of leaves of 50. The model also employed the 'gbdt' boosting type, had early stopping rounds set to 30, and used the extra trees method. This configuration yielded an average precision of approximately 66.7%, the highest among all tested combinations, while maintaining similar complexity to the other combinations.

Evaluation

In this final section, we evaluate our predictive model's performance, particularly its capability to predict if a customer will complete an offer post-viewing. We compare our model's performance against a baseline model to gauge its relative efficiency. We also scrutinize the best-performing model's characteristics, potential biases, and improvement areas. Additionally, we study the features that significantly contribute to predicting a user's offer completion. This thorough evaluation helps us understand our model's pros and cons and offers insights for future improvements.

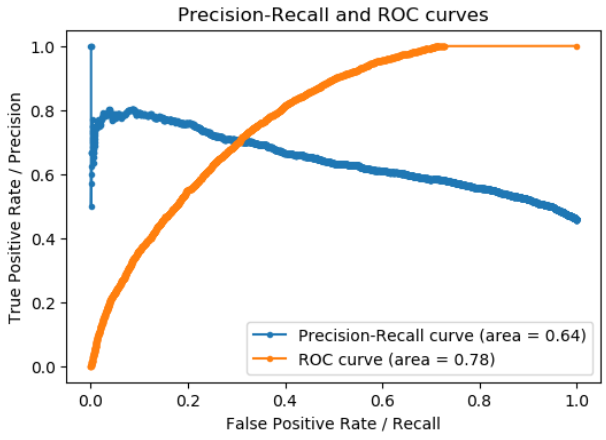
Model Performance

This section will showcase the model's performance, assessed independently of the threshold, by setting a fixed threshold and comparing it to a baseline.

Threshold Agnostic

This method allows us to evaluate the model across various thresholds, giving a holistic view of its predictive capabilities. We use two main metrics:

- **Precision-Recall Area Under the Curve (PR AUC):** This metric measures the area under the precision-recall curve, useful for evaluating prediction success in highly imbalanced classes. In our case, a high PR AUC score indicates that the model accurately identifies a large proportion of customers who completed the offer (high recall), and there is a high likelihood that a predicted customer completed the offer (high precision).
- **Receiver Operating Characteristic Area Under the Curve (ROC AUC):** This metric measures the area under the ROC curve, a tool balancing the trade-off between the true positive rate (TPR) and false positive rate (FPR). A high ROC AUC score suggests that the model effectively distinguishes between customers who did or didn't complete the offer.



Our model achieved a PR AUC score of approximately 0.644 and an ROC AUC score of approximately 0.780, indicating a reasonable performance in identifying customers who will complete an offer after viewing it, with a stronger ability to differentiate between customers who completed the offer and those who did not.

Fixed Threshold

The selection of a threshold can greatly influence metrics in imbalanced classification problems, and while there may not be a single "optimal" threshold, it's useful to select one to compute metrics like precision, recall, and F1-score, and to generate a confusion matrix. These provide further insights into the model's performance.

A common strategy is to choose the threshold that maximizes the F1-score, balancing precision and recall. However, the ideal threshold depends on the specific business context and the relative importance of precision and recall. In our scenario, the threshold that maximized the F1-score was approximately 0.35. Using this threshold, we obtained the following results:

Metric	Offer Not Completed	Offer Completed
Precision	0.85	0.55
Recall	0.58	0.83
F1-score	0.69	0.66

The results show our model has higher precision for non-completing customers but higher recall for completing ones. This indicates the model is adept at identifying non-completing customers but may misclassify some as completing. However, it accurately identifies a large portion of completing customers. The model's overall accuracy is 0.68, correctly classifying 68% of instances. This balance can be adjusted depending on whether precision or recall is prioritized.

Comparison with baseline

Baseline models serve as a reference point to gauge a model's performance. We're comparing our LightGBM model to a baseline model that predicts the majority class, using McNemar's test to determine if the performance difference is statistically significant. The McNemar's test confusion matrix reveals:

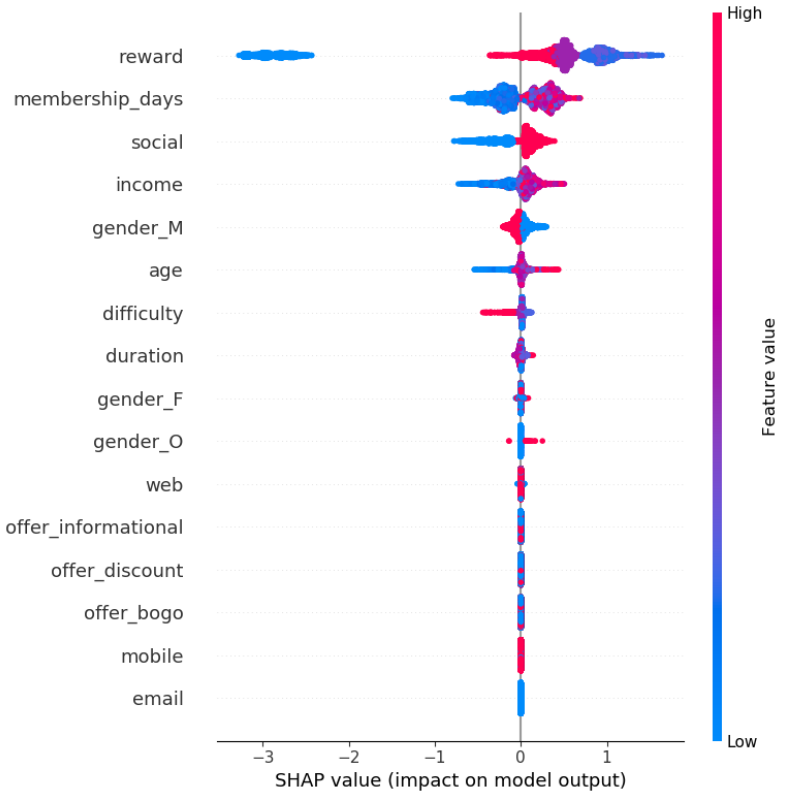
- 456 customers didn't complete the offer and were accurately predicted by both models.
- 2213 customers completed the offer, correctly predicted by the LightGBM model but not the baseline.

- 1821 customers completed the offer, correctly predicted by the baseline but not the LightGBM model.
- 2551 customers completed the offer and were accurately predicted by both models. .

With a p-value of 0.000 from McNemar's test, the performance difference between the models is statistically significant. The LightGBM model correctly predicted more customers who completed the offer than the baseline model, indicating its superior performance. However, the business context and the costs of false positives and negatives should be considered when selecting a model.

Model Interpretability

In this section, we will use the SHAP (SHapley Additive exPlanations) summary plot to understand the model's features and their influence on offer completion. SHAP is a tool that visualizes the impact of features in a machine learning model. Each dot on the plot represents a feature's effect on the model's prediction for an instance. The y-axis corresponds to features, while the x-axis represents the SHAP value, indicating how much each feature alters the prediction. The dot's color signifies the feature's actual value, with warm colors indicating high values and cool colors indicating low values.



Key insights from the top important features in the SHAP feature importance summary plot are:

1. **Reward:** The relationship between reward and offer completion probability is non-linear. Initially, as the reward increases, so does the likelihood of offer completion. However, after a certain point, further increases in reward seem to decrease the probability of completion. This could stem from various factors, such as customers viewing extremely high rewards as dubious or unrealistic, or it could be an illustration of the diminishing returns phenomenon. Further investigation on this will be carried out in the next step.
2. **Membership Days:** The model indicates a positive correlation between the number of membership days and the probability of completing the order, suggesting that long-term members are more likely to complete offers, possibly due to increased loyalty.
3. **Social Channel:** The social channel emerges as the most effective medium for sharing offers, significantly outperforming web, mobile, and email. This could be attributed to the viral nature of social media, where users can easily share and discuss offers.
4. **Income:** Higher income levels are associated with a higher probability of offer completion. This could be because higher-income individuals have more disposable income to spend, making them more likely to complete offers.
5. **Gender:** Male users seem to have a higher probability of not completing the offer compared to female and other gender users.
6. **Age:** There is a positive correlation between age and the likelihood of offer completion.
7. **Difficulty:** The model shows a negative correlation between the difficulty of the offer and the probability of its completion. This suggests that customers are more likely to complete offers that are easier to understand and fulfill.
8. **Duration, Offer Type:** The model does not show clear patterns or significant importance for the duration of the offer or its type.

These insights can aid in designing more effective marketing strategies, tailoring offers to specific customer segments, and optimizing the channels used for sharing offers. Further investigation will be conducted on complex features and their dependencies.

Detailed Analysis of Specific Features

In this section, we delve deeper into the influence of specific features on offer completion, particularly focusing on the interaction between 'reward' and 'membership days', 'income', 'gender', and 'age'.

Reward and Membership Days

The interaction between 'reward' and 'membership days' significantly impacts offer completion:

- **No Reward (0):** Offers without rewards are less likely to be completed by long-term members.
- **Low Rewards (2 or 3):** Moderate rewards enhance offer completion, especially among long-term members.
- **Medium Reward (5):** A reward of 5 has a consistent impact on offer completion across all membership durations.
- **High Reward (10):** High rewards generally promote offer completion, although their effectiveness slightly diminishes for long-term members.

These findings suggest that reward strategies should be customized for different customer segments, considering both the reward amount and membership duration. Generally, a low reward yields higher chances of offer completion.

Income and Reward Amount

The interaction between 'income' and 'reward' significantly influences offer completion:

- **Low Income (30k-50k):** Customers in this income bracket are less likely to complete offers, with the likelihood decreasing as the reward amount increases.
- **Middle Income (50k-100k):** Customers with middle-range incomes are more likely to complete offers, and this likelihood increases with the reward amount.
- **High Income (>100k):** High-income customers, though fewer, show a decreased likelihood of offer completion, especially for rewards less than 8.

These findings suggest that income level and reward amount significantly influence offer completion, and these factors should be considered when designing marketing strategies.

Gender and Reward Amount

Interesting patterns are revealed in the interaction of 'gender':

- **Male:** Male customers tend to have slightly negative SHAP values, indicating a lower likelihood of offer completion, particularly for higher reward amounts.
- **Non-Male (Female or Other):** Non-male customers show slightly positive SHAP values, suggesting a higher likelihood of offer completion, especially for higher rewards.

While these insights could be used to tailor marketing strategies, it's crucial to avoid reinforcing stereotypes or creating discriminatory practices. Any actions based on these insights should promote fairness and equality.

Age and Reward Amount

The interaction of 'age' significantly impacts:

- **Younger Age (18-40):** Younger customers are less likely to complete offers, particularly those with higher rewards. This could be due to limited disposable income, access to a wider range of deals, or selective shopping habits.
- **Older Age (>40):** Customers older than 40 are more likely to complete offers, especially those with higher rewards. This could be attributed to greater financial stability, higher brand loyalty, or a higher value placed on the convenience and savings provided by the offers.

These insights suggest that age and reward amount significantly influence offer completion. However, age-based marketing strategies should be implemented ethically, respecting all age groups and avoiding any form of age discrimination.

Conclusions

The LightGBM model developed, which was designed to predict if a customer will complete an offer, exhibits substantial effectiveness. By setting a threshold at roughly 0.35, optimized for the F1-score, the model successfully identifies 83% of customers who finalize an offer, as evidenced by a high recall rate of 0.83. However, the model's precision is 0.55, indicating that the model's prediction of a customer completing an offer is accurate 55% of the time. The model's overall accuracy stands at 0.68.

The Precision-Recall and ROC curves further corroborate the model's performance, assessing its capacity to differentiate between customers who will and will not complete an offer. The model has achieved a PR AUC score of about 0.644 and an ROC AUC score of around 0.780. These scores suggest a high proficiency in identifying customers who will finalize an offer after viewing it, and a superior ability to distinguish between customers who completed the offer and those who did not.

The model also provides several key insights about the features that have a higher predictive power for identifying customers that will complete an offer:

- **Reward:** Moderate rewards increase offer completion likelihood, particularly among long-term members. No reward or high rewards decrease this likelihood.
- **Membership Days:** Long-term members are more inclined to complete offers.
- **Social Channel:** The social channel is the most effective for sharing offers, outperforming web, mobile, and email.

- **Income:** Middle-income customers are more likely to complete offers, especially with increased rewards. Low and high-income customers show a decreased likelihood of offer completion.
- **Difficulty:** Easier offers are more likely to be completed.
- **Duration, Offer Type:** These factors do not significantly influence offer completion.
- **Gender:** Non-male customers are more likely to complete offers, especially those with higher rewards. Male customers are less likely.
- **Age:** Customers older than 40 are more likely to complete offers, especially those with higher rewards. Younger customers are less likely.

These insights should guide marketing strategies. However, it's crucial to ensure fairness and non-discrimination based on gender or age. When tailoring offers based on these insights, ethical implications and business objectives should be taken into account.

In summary, this model is particularly efficient when the aim is to capture as many offer completions as possible, even at the risk of some false positives. The results indicate that the model could be a beneficial tool for improving marketing strategies and optimizing promotional offers.

Next Steps

The future course of action should involve training a new LightGBM model, concentrating exclusively on the features that showed high predictive power. These features are reward, membership_days, income, social, difficulty, and duration, while intentionally omitting gender and age-related features. This strategy could potentially improve the model's interpretability, decrease noise, and enhance performance. If the performance does not improve, it would be necessary to assess whether the inclusion of gender and age aligns with the business goals.

Another approach worth considering is the creation of a two-step model. The initial model would predict whether a customer will view an order. Assuming the customer views it, a subsequent model would then predict if they will complete the order. The target population for this second model should be those who viewed the offer. Examining the features that influence a customer to both view and complete an order in this two-step process could potentially increase the accuracy and interpretability of our predictive approach. By segmenting the problem into two stages, we might be able to identify distinct patterns that are specific to each stage, potentially enhancing the overall accuracy of our predictions.