# Machine Learning Engineer Nanodegree

## Capstone Proposal

Chen Tong
Jun 26rd, 2019

## Proposal

### Domain Background

Starbucks is an American coffeehouse chain. Once every few days, Starbucks sends out an offer to users via different ways such as 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 offer during certain weeks. An important characteristic regarding this capstone is that not all users receive the same offer. As part of marketing strategy, we always want to figure out if a customer will spend more by giving a sound offer. Providing right offer to right customer could help build loyalty of the brand and product and as a result increasing sales margins in the long run.
The goal is to create a predictor to answer a question, that is, if a customer will response and complete a given offer? There are several articles about how to establish such predictor, for example, Who Might Respond to Starbucks' offer?. This predictor could be made by leveraging different classification models.

### Problem Statement

I chose to build a model that predicts whether or not a customer will complete an offer, meaning that the customer will receive a offer, view the offer and finish the offer before expire day. Thus, to solve this problem, I will establish a binary classifier.

One potential solution has 3 steps:

1. Preprocess combined transaction, demographic and offer data. This dataset describes an offer's attributes, the user's demographic data and whether the offer was successful. Also, split datasets into train, valuation and test datasets.
2. Training XGBoost Model as baseline.
3. Training PyTorch deep learning model and improve the results.

### Datasets and Inputs

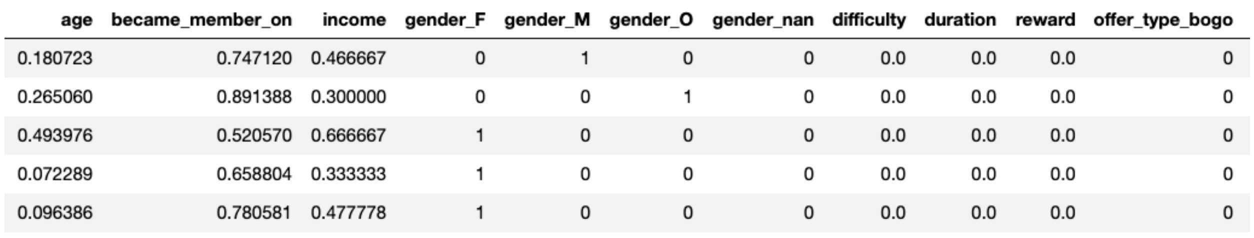
The datasets are sourced from Starbucks and data points are simulated to mimic customer behavior on the Starbucks rewards mobile app. Three files are included:

- portfolio.json - containing offer ids and meta data about each offer, such as the duration and the amount a customer need spend to complete it for an offer.
- profile.json - demographic data for each customer including their age, gender and income.
- transcript.json - records for transactions, offers received, offers viewed, and offers completed.

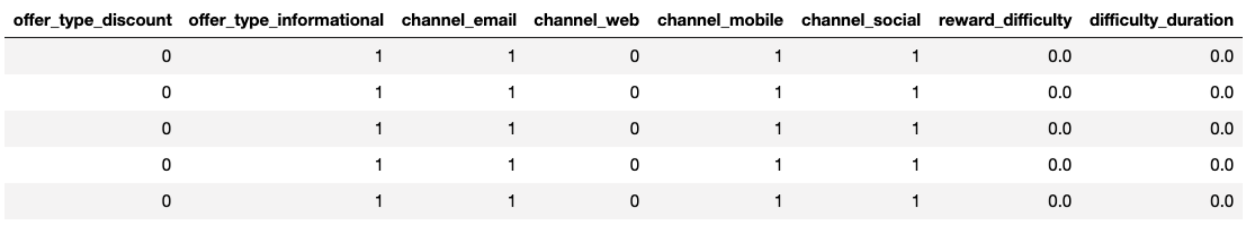
These files are cleaned, processed, transformed and joined into one table which contains demographic data of a customer, attributes of the offer and whether the customer complete the offer. The final dataset is 76277 * 20 in shape and columns or features are age, became_member_on, income, gender_F, gender_M, gender_O, gender_nan, difficulty, duration, reward, offer_type_bogo, offer_type_discount, offer_type_informational, channel_email, channel_web, channel_mobile, channel_social, reward%difficulty, difficulty%duration.

Labels values are balanced because the number of occurrences of positive label is 34809 while the number is 41468.

Below are the first 5 rows of the dataset. The first column is label. The rest are features.



| age | became_member_on | income | gender_F | gender_M | gender_O | gender_nan | difficulty | duration | reward | offer_type_bogo |
|---|---|---|---|---|---|---|---|---|---|---|
| 0.180723 | 0.747120 | 0.466667 | 0 | 1 | 0 | 0 | 0.0 | 0.0 | 0.0 | 0 |
| 0.265060 | 0.891388 | 0.300000 | 0 | 0 | 1 | 0 | 0.0 | 0.0 | 0.0 | 0 |
| 0.493976 | 0.520570 | 0.666667 | 1 | 0 | 0 | 0 | 0.0 | 0.0 | 0.0 | 0 |
| 0.072289 | 0.658804 | 0.333333 | 1 | 0 | 0 | 0 | 0.0 | 0.0 | 0.0 | 0 |
| 0.096386 | 0.780581 | 0.477778 | 1 | 0 | 0 | 0 | 0.0 | 0.0 | 0.0 | 0 |



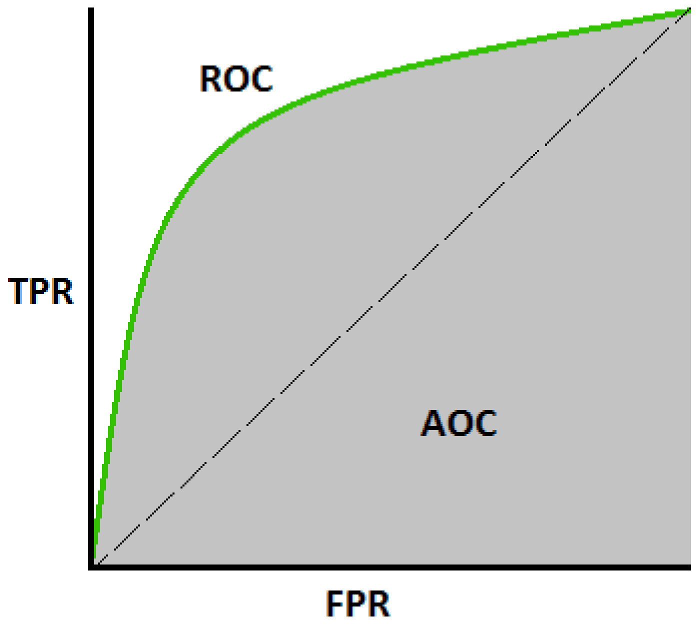
| offer_type_discount | offer_type_informational | channel_email | channel_web | channel_mobile | channel_social | reward_difficulty | difficulty_duration |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 | 0.0 | 0.0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0.0 | 0.0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0.0 | 0.0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0.0 | 0.0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0.0 | 0.0 |

## Solution Statement

Recall the problem, "if a customer will response and complete a given offer?". In dataset, we have features profiling customers and other features about offer descriptions. But we didn't include any information from transaction because the transaction may hint the answer. The label is 1 (complete) or 0 (incomplete). Thus, this is a binary classification problem.

Some common methods are decision tree, random forests, support vector machines(svm) and neural networks(nn). In this project, we will use boosting random forests (XGBoost) model as benchmark model and use neural networks(pytorch) as solution model.

ROC-AUC is used as metric for model evaluations. It computes area under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores. Higher is better. According to this article, we choose ROC-AUC because the datasets are balanced and we only care about the final class predictions. See roc curve image:



The model is replicable because once the hyperparameter and datasets are set, the model articles should be similar.

## Benchmark Model

We use XGBoost model as benchmark. Hyperparameters are

- max_depth=5,
- eta=0.2,
- gamma=4,
- min_child_weight=6,
- subsample=0.8,
- objective='binary:logistic',

The ROC-AUC score is 0.71594. This result will be used as the threshold to determine the improvement of solution model of which the ROC-AUC score is higher.

## Evaluation Metrics

ROC-AUC is used as evaluation metric. It computes area under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores. This will quantify the performance of both the benchmark model and the solution model. The reason is that the datasets are balanced and also, we only care about the final class predictions and we don’t want to tune threshold.

## Project Design

The project will follow this workflow for approaching a solution:

1. Data Processing

    1. Data Cleaning
        - Profile: figure out how to deal with null values for age, gender and income. Simple way may discard these values. But if they make up a big portion of data, I may use another category to represent them.
        - Portfolio: split channels into its own columns.
        - Transcript: make an offer completion column or dataset based on customer's behaviors(events).
    2. Data Transformation: Join these tables into a data set. The first column is whether the offer is completed, and the rest are features from profile and portfolio.
2. Feature Engineering
    1. Determine features by exploratory analyze features to choose a small number of uncorrelated features. If features are too many, I may consider reducing dimensionality by performing a PCA.
    2. Prepare final datasets: split dataset into train, valuation and test datasets.
3. Train XGBoost Model on SageMaker
4. Train Deep Learning Model on SageMaker
5. Hyperparameter Tuning Deep Learning Model on SageMaker
6. Conclusion

## Reference

1. Starbucks Wiki
2. Udacity Starbucks Project Overview
3. Accuracy
4. F1 score
5. Starbuck’s Capstone Challenge
6. Starbucks Capstone Challenge Dataset Customer Offer Success Prediction
7. Who Might Respond to Starbucks’ offer?