Starbucks Capstone Project Proposal

Udacity Machine Learning Engineer Nanodegree

Lu Liu

June 28, 2020

1 Introduction

The Starbucks Capstone project is part of the Udacity Machine Learning Engineer Nanodegree. Udacity partnered with Starbucks to provide a real-world business problem and simulated data mimicking their customer behaviour.

2 Domain Background

Starbucks Corporation is an American coffeehouses company and it is one of the world's largest coffeehouse chain. Once every few days, Starbucks sends out an offer to users of the Starbucks rewards mobile app. The offers contain three types: an advertisement for a drink, an actual offer such as a discount and BOGO (buy one get one free). Some users might not receive any offer during certain weeks. Starbucks wants to utilise technology to provide their customers a better personalised experience and boost their marketing performance. To be specific, they want to target the right customers who are most likely to make a purchase and also know about customers who might not want an offer.

The motivations of this project is to gain more machine learning hands-on practice and have a better understanding about how the technology can make an impact on the real business.

3 Problem Statement

Based on the description above, the problem that needs to be solved in this project is to predict whether a customer will complete an offer or not based on the given demographics and historical data. This problem can be treated as a supervised learning problem. Specifically, it can be treated as a binary classification problem where '1' represents that the customer will complete the offer and '0' represents that the customer will not complete the offer.

4 Datasets and Inputs

The following overview of the datasets are provided by Udacity:

- The program used to create the data simulates how people make purchasing decisions and how those decisions are influenced by promotional offers.

- Each person in the simulation has some hidden traits that influence their purchasing patterns and are associated with their observable traits. People produce various events, including receiving offers, opening offers, and making purchases.
- As a simplification, there are no explicit products to track. Only the amounts of each transaction or offer are recorded.
- There are three types of offers that can be sent: buy-one-get-one (BOGO), discount, and informational. In a BOGO offer, a user needs to spend a certain amount to get a reward equal to that threshold amount. In a discount, a user gains a reward equal to a fraction of the amount spent. In an informational offer, there is no reward, but neither is there a requisite amount that the user is expected to spend. Offers can be delivered via multiple channels.
- The basic task is to use the data to identify which groups of people are most responsive to each type of offer, and how best to present each type of offer.

The provided datasets contain three datasets:
profile.json: Rewards program users (17000 users x 5 fields)

- gender: (categorical) M, F, O, or null
- age: (numeric) missing value encoded as 118
- id: (string/hash)
- became_member_on: (date) format YYYYMMDD
- income: (numeric)

portfolio.json: Offers sent during 30-day test period (10 offers x 6 fields)

- reward: (numeric) money awarded for the amount spent
- channels: (list) web, email, mobile, social
- difficulty: (numeric) money required to be spent to receive reward
- duration: (numeric) time for offer to be open, in days
- offer_type: (string) bogo, discount, informational
- id: (string/hash)

transcript.json: Event log (306648 events x 4 fields)

- person: (string/hash)
- event: (string) offer received, offer viewed, transaction, offer completed
- value: (dictionary) different values depending on event type
- offer id: (string/hash) not associated with any ”transaction”

- amount: (numeric) money spent in "transaction"
- reward: (numeric) money gained from "offer completed"
- time: (numeric) hours after start of test

5 Solution Statement

In this project three models will be used to solve this supervised problem: Logistic Regression, XGBoost and Neural Networks. The reason that these three algorithms are chosen is because Logistic Regression is one of the most commonly used ones to solve classification problem, XGBoost is considered as one of the top performers on machine learning tasks. Additionally, Neural Networks normally perform well when the training set is big enough, for this project, we can absolutely give it a try.

6 Benchmark Model

Logistic Regression will be chosen as the benchmark model since it is relatively straightforward and efficient to implement.

7 Evaluation Metrics

Confusion matrix along with accuracy, precision, recall and F2 score will be considered in the evaluation of the model performance.

The table below is the illustration of Confusion matrix:

		Prediction		
		0	1	Total
Actual	0	$d - TrueNegative$	$c - FalsePositive$	$c + d$
	1	$b - FalseNegative$	$a - TruePositive$	$a + b$
	Total	$b + d$	$a + c$	N

a - True Positive: Send an offer the customer will be likely to use the offer.

b - False Negative: Send an offer the customer will not be likely to use the offer.

c - False Positive: Do not send an offer the customer will be likely to use the offer.

d - True Negative: Do not send an offer the customer will not be likely to use the offer.

In this project, we do not want to optimize accuracy only, because in this business case, we want to avoid the situate of sending offers to customers who are not likely to use them as much as possible (b - False Negative). Recall is often used when the cost of False Negative is high, and we also want to take Precision into consideration. Therefore, we chose F2 score as the final evaluation metrics. The formula for F2 score can be described as below:

$$F_2 = \frac{(1+2^2) * Precision * Recall}{(2^2 * Precision) + Recall} \tag{1}$$

8 Project Design

The workflow for this project can be described as:

- Data pre-processing: Includes data cleaning, merging and dealing with the invalid values such as missing values, NaN etc.
- Data exploration: Understand the pattern and distribution of the data and the correlations among the features.
- Data transformation: Includes data transformation before feeding into the model (one hot encoding), normalization if needed.
- Model training: Logistic Regression model as the bench mark model, XGBoost and Neural Network model.
- Model evaluation and comparison between a selected model and the benchmark model.
- Document the experiments in the final report.
- Upload to Git.