Udacity Starbucks Capstone Project Proposal

Machine Learning Engineer Nanodegree

Victor Barros

Domain Background

Starbucks is an American multinational company, with the largest coffee chain in the world. In addition to cappuccino and espresso, Starbucks offers other types of drinks, such as a variety of teas, as well as foods such as sandwiches. Starbucks stores can be inside other stores, such as bookstores and shopping malls.

The company has released a data set that simulates the behavior of customers using the Starbucks rewards mobile app. The Starbucks app is able to share and advertise offers with frequent customers. Consumers can also use it as a payment method in stores. Often the company sends different types of offers in a given period (based on consumer behavior).

Promotions can be discount ads on a particular drink, or they can also be discount promotions or buy one and get one free, it is also important to know that all promotions have an expiration date. The company provided a series of information on the behavior, consumption and demographics of its customers, in such a way that the solution allows building customized promotions for each customer that encourage minimum consumption within relevant consumer preferences.

References:

- Wikipedia.org - https://pt.wikipedia.org/wiki/Starbucks
- Binary Classifiers Applied to Marketing - https://medium.com/datasparq-technology/binary-classifiers-applied-to-marketing-f0fca6d968a6

Problem Statement

The company is looking for a solution that can allow customers to increase their consumption through personalized advertisements and promotions. Thus, considering that the company has the ability to capture and process various relevant data on individual consumption, location, frequency, among other information, it can accurately target the ads and promotions to each customer, at the right time.

This type of problem can be solved with *Binary Classifier*, where we consider as input data: total consumption, channel, location, income and gender. We can expect an output of labels to buy or not to buy.

Datasets and Inputs

The company has made available a set of transactional data that reflect consumer behavior within the application. The data is contained in three files:

- *portfolio.json* - containing offer ids and meta data about each offer (duration, type, etc.)
- *profile.json* - demographic data for each customer
- *transcript.json* - records for transactions, offers received, offers viewed, and offers completed

The format and dictionary of the files were made available below:

portfolio.json

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

	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

	difficulty	duration	reward
count	10.000000	10.000000	10.000000
mean	7.700000	6.500000	4.200000
std	5.831905	2.321398	3.583915
min	0.000000	3.000000	0.000000
25%	5.000000	5.000000	2.000000
50%	8.500000	7.000000	4.000000
75%	10.000000	7.000000	5.000000
max	20.000000	10.000000	10.000000

Table 1 portfolio.json data description

profile.json

- age (int) - age of the customer
- became_member_on (int) - date when customer created an app account
- gender (str) - gender of the customer (note some entries contain 'O' for other rather than M or F)
- id (str) - customer id
- income (float) - customer's income

	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

	age	became_member_on	income
count	17000.000000	1.700000e+04	14825.000000
mean	62.531412	2.016703e+07	65404.991568
std	26.738580	1.167750e+04	21598.299410
min	18.000000	2.013073e+07	30000.000000
25%	45.000000	2.016053e+07	49000.000000
50%	58.000000	2.017080e+07	64000.000000
75%	73.000000	2.017123e+07	80000.000000
max	118.000000	2.018073e+07	120000.000000

Table 2 profile.json data description

transcript.json

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

	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'}

	time
count	306534.000000
mean	366.382940
std	200.326314
min	0.000000
25%	186.000000
50%	408.000000
75%	528.000000
max	714.000000

Table 3 transcript.json data description

When we analyze the available data sets, we are able to find some relevant information that can be used in the application. All data sets can be easily linked after data cleaning and normalization.

It will be necessary to organize the data related to the consumer profile, identifying characteristics such as volume of orders and total spent.

The data set seems to be balanced, some fields have outliers that will need to be treated and it will be important to understand which features will still be more meaningful for classification.

Solution Statement

Considering that the spread solution delivers improvements in the quality of sending promotional offers to consumers, and that they have a greater chance of conversion, it will be necessary to analyze a series of characteristics in our data set, in such a way that it is possible to find possible groups based on information demographic vs. consumption history.

For this type of problem, the solution may possibly include models that perform better for classification, such as *RandomForestClassifier*, which can be trained to deliver a better recommendation based on demographic or consumption characteristics.

The expected solution should cover the following aspects:

- Determine the best channel and promotion, based on characteristics and consumer consumption history
- It should work for both old customers and new consumers

Benchmark Model

To compare the results of the model that will be built and trained, I intend to build a benchmark that contains a cluster analysis (*KNeighborsClassifier*) or also evaluate whether a linear regression (*LogisticRegression*) can also present good results in the face of the problem.

Evaluation Metrics

To assess the quality of the model, I believe that a composition of evaluation metrics is necessary, such as: accuracy, precision, recall or F1 score.

The evaluation formulas can be found below:

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \qquad \text{Precision} = \frac{\text{TP}}{\text{TP + FP}} \qquad \text{Accuracy} = \frac{\text{Correct Guesses}}{\text{Total Population}} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}}$$

$$F_1 = \frac{2}{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} = \frac{\mathrm{tp}}{\mathrm{tp} + \frac{1}{2}(\mathrm{fp} + \mathrm{fn})}.$$

Project Design

As a proposal to develop a solution that can deliver value to the consumer and at the same time deliver performance improvement for the company, the following application structure is proposed:

- Setup development environment
- Cleaning and transforming the data set (scale, adjust fields, normalize date, remove outliers, rename fields, one-hot encoded)
- Make an analysis of the data set, mainly understanding the possible intersections based on the expected model (Confusion Matrix for example)
- Explore different solutions based on models compatible with the type of problem (*RandomForestClassifier, XGBoost, LightGBM or other*), seeking to find the one that delivers the best cost benefit for the company
- Evaluate performance metrics, seeking to deliver the most optimized result possible
- Data sanity check
- Final project report with summary of findings and conclusion on the best approach