ISRAILOV SARDOR

Capstone Proposal

domain background

The project was an experiment from Starbucks to optimize the offers sent to its customers and to determine what is the best offer on the individual level. Some people may respond negatively to the offer, so in some cases it’s best not to send any offer at all.

Once every few days, Starbucks sends 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). The project is meant to optimize the offers sent to the customers based on the certain characteristics as sex, income range, demographics etc.

Example

To give an example, a user could receive a discount offer buy 10 dollars get 2 off on Monday. The offer is valid for 10 days from receipt. If the customer accumulates at least 10 dollars in purchases during the validity period, the customer completes the offer.

Problem statement

The problem that I intend to solve consists of determining:

whether the customer will view and complete the offer or just view it.

The most crucial step would be to understand the data and its probable implications on the outcome. Data pre-processing would be the crucial step to the further results. After obtaining good results, the next step would be to identify the crucial factors for the offer to be completed and indicate the strategies for the marketing campaign: which offer to send to which person.

Datasets and inputs

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

Here is the schema and explanation of each variable in the files:

portfolio.json

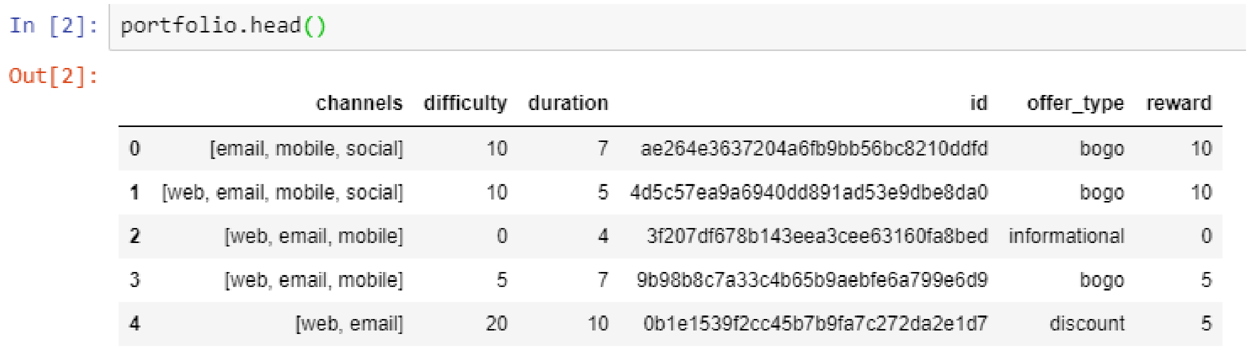
- id (string) - offer id
- offer_type (string) - type of offer ie BOGO, discount, informational
- difficulty (int) - minimum required spend to complete an offer
- reward (int) - reward given for completing an offer
- duration (int) - time for offer to be open, in days
- channels (list of strings)

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

transcript.json

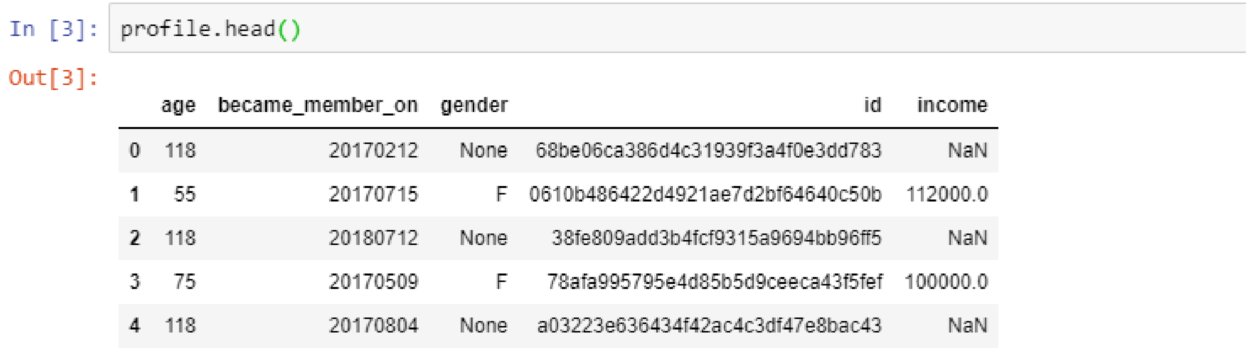
- event (str) - record description (ie transaction, offer received, offer viewed, etc.)
- person (str) - customer id
- time (int) - time in hours since start of test. The data begins at time t=0
- value - (dict of strings) - either an offer id or transaction amount depending on the record



```
In [2]: portfolio.head()
```

Out[2]:

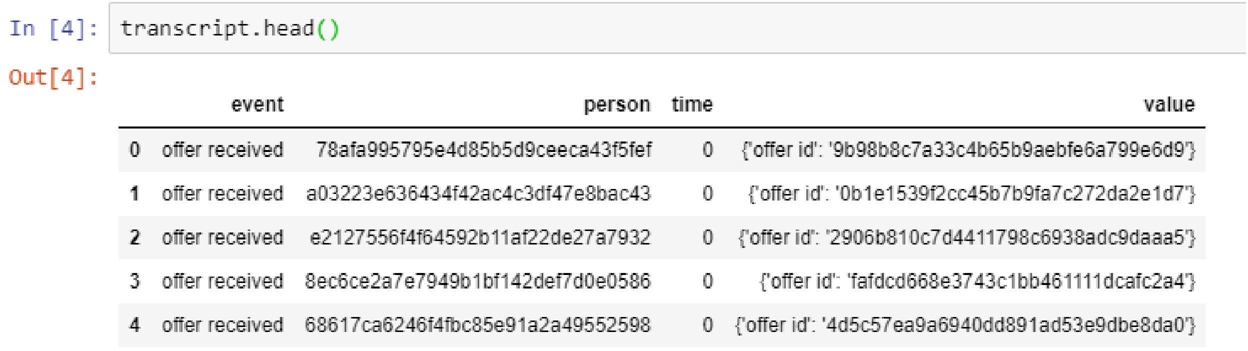
	channels	difficulty	duration	id	offer_type	reward
0	[email, mobile, social]	10	7	ae264e3637204a6fb9bb56bc8210ddfd	bogo	10
1	[web, email, mobile, social]	10	5	4d5c57ea9a6940dd891ad53e9dbe8da0	bogo	10
2	[web, email, mobile]	0	4	3f207df678b143eea3cee63160fa8bed	informational	0
3	[web, email, mobile]	5	7	9b98b8c7a33c4b65b9aebfe6a799e6d9	bogo	5
4	[web, email]	20	10	0b1e1539f2cc45b7b9fa7c272da2e1d7	discount	5



```
In [3]: profile.head()
```

Out[3]:

	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



```
In [4]: transcript.head()
```

Out[4]:

	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'}

solution statement

The first thing to do would be to understand and visualize the data. Then, we would eliminate the outliers, the (NaN) rows in the table that would not contribute to our training or test. Afterwards, I would combine 3 data tables in a big one with preprocessing steps. For example, instead of having male, female we would have 0 or 1. Use data scaling as the way to normalize the input data. Afterwards, we would try several models of Sklearn and Pytorch CNN and choose the best one.

evaluation metrics

The metric that I intend to use for binary classification is Accuracy.$\frac{TP+TN}{TP+TN+FP+FN}$

project design

I have identified the following steps to project resolution:

- Visualising&Data analysis
- Pre-processing data
- Scaling the numerical features
- Testing supervised models
- Evaluating the results on the accuracy. If the results to be improves use GridSearchSV or Bayesian hyper-parameter tuning.
- Conclusion and evaluation of the crucial factors for the decision i.e. which columns contribute most for the decision of whether the customer will view&complete an offer or just view it.