Starbucks Capstone Challenge

CAPSTONE PROJECT

Udacity Machine Learning Engineer Nanodegree Program

Essi Lehtola, April 2020

The project background

In this Udacity Machine Learning Nanodegree Capstone project I am going to analyse and process data that mimics customer behavior on the Starbucks rewards mobile app.

Starbucks is the biggest international coffeehouse and roastery chain in the world and has coffee shops in over 60 countries. In the app the company is sending out different types of offers (for example informational offers, discounts and 'buy one get one' offers) and my goal is to use the data provided to determine how the customers react to different offers.

Problem Statement

In this project I was able to decide how to use the data myself. I decided to find an answer to a real business question: is a customers going to buy something in the influence of a certain offer based on past customer behaviour?

I will build a machine learning model that can answer to that problem. When customer and offer details are given to the model it will predict if the customer will make a purchase on Starbucks or not.

Datasets and Inputs

There are three simplified data tables provided by Starbucks:

1) Portfolio.json

Contains 6 columns and 10 rows of offer related data.

There are 10 different offers in total: 4 of them are bogos (buy one get one), 4 are discounts and 2 informational offers.

The data columns are:

- 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)

	id	offer_type	duration	difficulty	reward	channels
0	ae264e3637204a6fb9bb56bc8210ddfd	bogo	7	10	10	[email, mobile, social]
1	4d5c57ea9a6940dd891ad53e9dbe8da0	bogo	5	10	10	[web, email, mobile, social]
2	3f207df678b143eea3cee63160fa8bed	informational	4	0	0	[web, email, mobile]
3	9b98b8c7a33c4b65b9aebfe6a799e6d9	bogo	7	5	5	[web, email, mobile]
4	0b1e1539f2cc45b7b9fa7c272da2e1d7	discount	10	20	5	[web, email]
5	2298d6c36e964ae4a3e7e9706d1fb8c2	discount	7	7	3	[web, email, mobile, social]
6	fafdcd668e3743c1bb461111dcafc2a4	discount	10	10	2	[web, email, mobile, social]
7	5a8bc65990b245e5a138643cd4eb9837	informational	3	0	0	[email, mobile, social]
8	f19421c1d4aa40978ebb69ca19b0e20d	bogo	5	5	5	[web, email, mobile, social]
9	2906b810c7d4411798c6938adc9daaa5	discount	7	10	2	[web, email, mobile]

2) Profile.json

Contains all customer related details in 17 000 rows and 5 columns.

- 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

id	gender	age	income	became_member_on
79edb810789c447e8d212a324b44cc16	F	70	39000.0	20160310
a9a20fa8b5504360beb4e7c8712f8306	None	118	NaN	20160116
400d0536e8794cbb855b0d882d67cbda	F	21	72000.0	20170917
cb23b66c56f64b109d673d5e56574529	M	60	113000.0	20180505
c02b10e8752c4d8e9b73f918558531f7	None	118	NaN	20151211
6d5f3a774f3d4714ab0c092238f3a1d7	F	45	54000.0	20180604
2cb4f97358b841b9a9773a7aa05a9d77	M	61	72000.0	20180713
01d26f638c274aa0b965d24cefe3183f	M	49	73000.0	20170126
9dc1421481194dcd9400aec7c9ae6366	F	83	50000.0	20160307
e4052622e5ba45a8b96b59aba68cf068	F	62	82000.0	20170722

3) Transcript.json

Contains events that happened during the testing period in 306 534 rows and 4 columns.
This is the most important table but also hardest to preprocess. Events column contains relevant details such as if the action was a transaction or if an offer was received or viewed. Value column on the other hand provides data like offer ids and transaction amounts depending on event. In this table there are of course person id's and timestamps as well.

- 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

time	event	person	value
0	offer received	6e014185620b49bd98749f728747572f	{'offer id': 'f19421c1d4aa40978ebb69ca19b0e20d'}
24	transaction	6e014185620b49bd98749f728747572f	{'amount': 3.8200000000000003}
66	offer viewed	6e014185620b49bd98749f728747572f	{'offer id': 'f19421c1d4aa40978ebb69ca19b0e20d'}
66	transaction	6e014185620b49bd98749f728747572f	{'amount': 5.79}
66	offer completed	6e014185620b49bd98749f728747572f	{'offer_id': 'f19421c1d4aa40978ebb69ca19b0e20d...
108	transaction	6e014185620b49bd98749f728747572f	{'amount': 5.36}
168	offer received	6e014185620b49bd98749f728747572f	{'offer id': '2906b810c7d4411798c6938adc9daaa5'}
264	offer viewed	6e014185620b49bd98749f728747572f	{'offer id': '2906b810c7d4411798c6938adc9daaa5'}
312	transaction	6e014185620b49bd98749f728747572f	{'amount': 115.0}
312	offer completed	6e014185620b49bd98749f728747572f	{'offer_id': '2906b810c7d4411798c6938adc9daaa5...

Solution statement

When the question is will a customer make a purchase during the influence period of an offer the solution can be found in this way:

- I familiarize myself with the data available and gather and combine as much customer and offer details as possible

- I transfer that information to a machine readable form so that it can be used as an input for a machine learning model and also separate the data to train and test sets

- I will test few models to find a good one and the model will find patterns on data. I might need to tune the model a bit we to find best possible prediction rate. The model can then be used to predict if an offer has an influence on someone or not.

Benchmark model

Since this is a classification task I will test how the gathered and preprocessed data works with a simple Decision Tree Classifier.

After training it with a training set we can test the performance with a test set. That will tell us the percentage of the correct labels so that it is easy to compare other models to it. We can also use a simple confusion matrix to see if the classification was done correctly.

Evaluation metrics

I will evaluate if the models work with percentages of correctly predicted labels.

Along with that I will also use confusion matrix that shows how many of the labels were predicted correctly and how many went wrong and how (false negatives or false positives).

Project Design

The workflow will be following:

1. Gathering data from different sources

   First, I will have upload the data and analyze it a bit so that all relevant details can be found. I also need to combine the data from different sources so that I have good amount of relevant information for a model.

   We have customer ages, genders, incomes and dates when they became members but also actions they have made during the test period. We also have information like offer durations and difficulties.

   The hardest part is to collect information for example on how many offers customer has viewed and completed and how much money they have spent to Starbucks. The offers are different and the offer related customer actions are collected as records containing timestamps and event details so collecting data together might be tricky. To be able to see if an offer really had an effect on a person it requires quite a lot of preprocessing

2. Preprocessing data for machine readable form

   Second, the data collected needs to be transferred to a machine readable form.

   Empty values are not really usable in a model, but I am pretty sure there is enough data to remove all rows with null values.

We also need to change all text to numbers. To do that we can use encoding (counting all unique values and replacing each of them with a number) or one-hot-encode them (move all options to columns and give a value 1 if the original value matches the column name).

Encoding

gender	gender
M	1
F	2
F	2

One-hot-encoding

event
offer received
offer viewed
offer completed

offer received	offer viewed	offer completed
1	1	1

To make the data easier for a model to use I think it will be beneficial to scale/normalize numbers as well.

The last step here is to separate the data to training and testing sets.

3. Finding a good machine learning model

As a last step of the whole process, I need to find a good model to use to predict the result as accurately as possible.

I will use a Decision Tree Classifier as a benchmark. After that we can compare the results to results of other models and even try to improve the prediction rate using hyperparameter tuning. It is easy to see if a model performs better than others by looking at the percentage of the correctly labeled rows.

In the end we should have a model that predicts the result of a combination of a customer and an offer. For example, what happens if a 60 years old lady gets a discount - is she going to use it or not.