Udacity Data Science Nano Degree – Capstone Project

Starbucks Data Set

Project Overview

This data set contains simulated data that mimics customer behavior on the Starbucks rewards mobile app. 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). Some users might not receive any offer during certain weeks.

Not all users receive the same offer, and that is the challenge to solve with this data set.

Every offer has a validity period before the offer expires. As an example, a BOGO offer might be valid for only 5 days. You'll see in the data set that informational offers have a validity period even though these ads are merely providing information about a product; for example, if an informational offer has 7 days of validity, you can assume the customer is feeling the influence of the offer for 7 days after receiving the advertisement.

Goal of the Project and Problem Statement

We will be exploring the Starbuck's Dataset which simulates how people make purchasing decisions and how those decisions are influenced by promotional offers.

In the Project, Based on the information given above, it tries to answer following questions:

- What is the proportion of client who have completed the offers based on Gender?
- What is the proportion of client who have completed the offers based on their Age?
- What is the proportion of client who have completed the offers based on their Income Level?
- What are the most important features that help drive the offers in customers?

Data Sets

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

Implementation and Strategy

- Wrangle and combine the data from offer portfolio, customer profile, and transaction.
- Visualization to provide answers to the questions
- Important feature driver for the offers, deducted by different models that will be optimized.

Metrics

To quantify to performance of the used models, I will use accuracy and F-score metrics for. F1 score provides a better sense of model performance compared to purely accuracy as takes both false positives and false negatives in the calculation. With an uneven class distribution, F1 may usually be more useful than accuracy. Accuracy measures how well a model correctly predicts whether an offer is successful. However, if the percentage of successful or unsuccessful offers is very low, accuracy is not a good measure of model performance.

Business Context

The solution here aims to analyze how people make purchasing decisions and how those decisions are influenced by promotional offers. Every individual in the dataset has some hidden attributes that influence their buying patterns and are related to their discernible characteristics. Individuals produce different events, including accepting offers, opening offers, and making buys. However, there are a few things to watch out for in this data set. Customers do not opt into the offers that they receive; in other words, a user can receive an offer, never actually view the offer, and still complete the offer. For example, a user might receive the "buy 10 dollars get 2 dollars off offer", but the user never opens the offer during the 10-day validity period. The customer spends 15 dollars during those ten days. There will be an offer completion record in the data set; however, the customer was not influenced by the offer because the customer never viewed the offer.

Data Exploration

Portfolio

The Portfolio dataset doesnt hold that much information. Only ten entries, and the data is already quit tidy. The quantitative data is missing the metric it is meassured with.

	reward	difficulty	duration	offer_type	offer_id	email	mobile	social	web
0	10	10	168	bogo	ae264e3637204a6fb9bb56bc8210ddfd	1	1	1	0
1	10	10	120	bogo	4d5c57ea9a6940dd891ad53e9dbe8da0	1	1	1	1
2	0	0	96	informational	3f207df678b143eea3cee63160fa8bed	1	1	0	1
3	5	5	168	bogo	9b98b8c7a33c4b65b9aebfe6a799e6d9	1	1	0	1
4	5	20	240	discount	0b1e1539f2cc45b7b9fa7c272da2e1d7	1	0	0	1
5	3	7	168	discount	2298d6c36e964ae4a3e7e9706d1fb8c2	1	1	1	1
6	2	10	240	discount	fafdcd668e3743c1bb461111dcafc2a4	1	1	1	1
7	0	0	72	informational	5a8bc65990b245e5a138643cd4eb9837	1	1	1	0
8	5	5	120	bogo	f19421c1d4aa40978ebb69ca19b0e20d	1	1	1	1
9	2	10	168	discount	2906b810c7d4411798c6938adc9daaa5	1	1	0	1

Offer Type is not yet one hot encoded, because is more useful this way to visualization

Profile

Thhe Profil data just needed minor changes. The data is not one hot encoded to get better visualization.

	gender	age	customer_id	became_member_on	income
0	None	NaN	68be06ca386d4c31939f3a4f0e3dd783	2017-02-12	121411.602503
1	F	55.0	0610b486422d4921ae7d2bf64640c50b	2017-07-15	112000.000000
2	None	NaN	38fe809add3b4fcf9315a9694bb96ff5	2018-07-12	121411.602503
3	F	75.0	78afa995795e4d85b5d9ceeca43f5fef	2017-05-09	100000.000000
4	None	NaN	a03223e636434f42ac4c3df47e8bac43	2017-08-04	121411.602503

Became member on needs to be changed to a date and can later be used to calculate the number of days a customer ist already a member. There was a lot of missing data considering the age, as the customer just entered the maximum age. All this data and the missing income data was replaced by a random normal distribution of the respective columns.

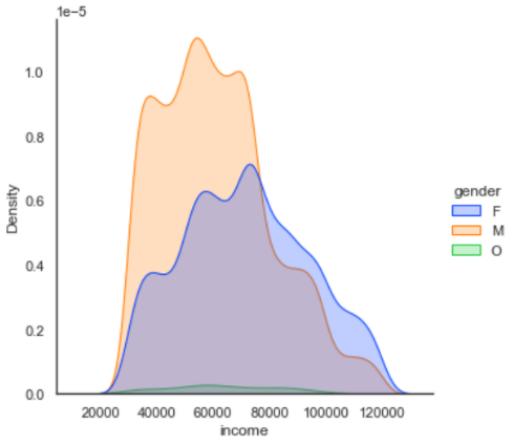
Transcript

The transcript data did not need that much cleaning just a separation from a nested column.

	customer_id	event	time	amount	reward	offer_id
0	78afa995795e4d85b5d9ceeca43f5fef	offer received	0	NaN	NaN	9b98b8c7a33c4b65b9aebfe6a799e6d9
1	a03223e636434f42ac4c3df47e8bac43	offer received	0	NaN	NaN	0b1e1539f2cc45b7b9fa7c272da2e1d7
2	e2127556f4f64592b11af22de27a7932	offer received	0	NaN	NaN	2906b810c7d4411798c6938adc9daaa5
3	8ec6ce2a7e7949b1bf142def7d0e0586	offer received	0	NaN	NaN	fafdcd668e3743c1bb461111dcafc2a4
4	68617ca6246f4fbc85e91a2a49552598	offer received	0	NaN	NaN	4d5c57ea9a6940dd891ad53e9dbe8da0

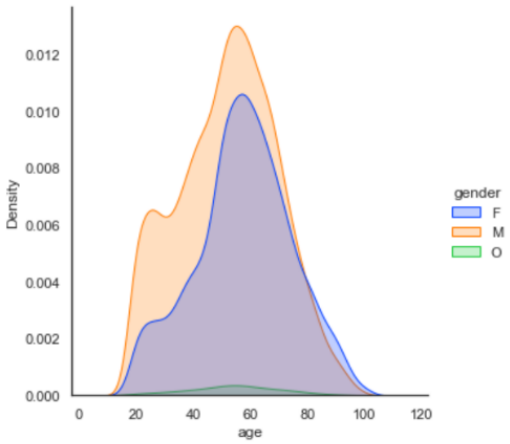
Data Vizualsation

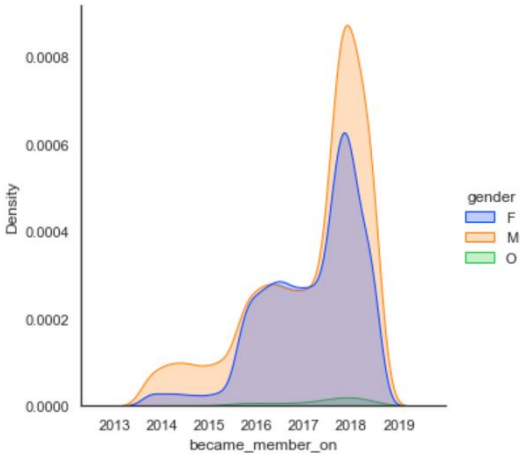
In the data visualization, we explore in more detail how the different quantitative and numerical values are distributed over the dataset and if there are some abnormalities that need to be considered. First, we look at the income and age distribution depending on gender in the dataset, aswell as the year the customer joined:



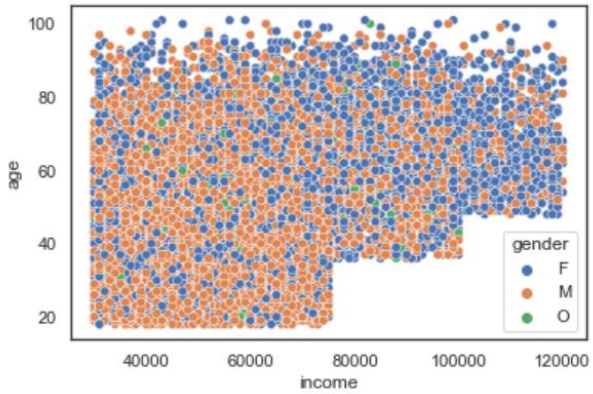
We can see that the data for income and age follows roughly a normal distribution, with the mean around 70.000 Dollar income and an age of 60.

In the became a member year we see that most members joined in the last year before the promotion, or in the year of the promotion.

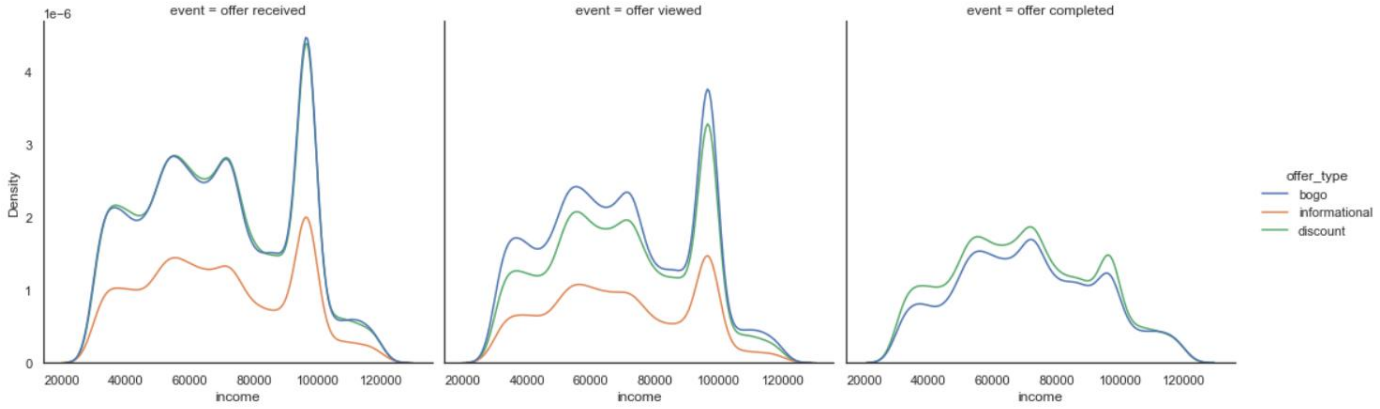




In the following, some advanced plotting with two quantitative and numerical data. In the following a scatter plot of income an age by gender. We can see that the data is sumulated data as we have two steps in the income correlated to age. The gender distributions seems, that more female customer have a better income with higher age.



In the following, the data is separated by the different offer types made and if the customer completed, viewed or receveid an offer.



We can see that the distribution for offer made and offer viewed follows a similar distribution. With a peak for the income around 100.000 Dollar. Informational offer being the least, and not in the offer completed as there is no benefit in the information offer.

Data Preprocessing

For training a model, we need to create a label, which we consider a success or non success offer.

- A Success would be an offer views and offer completed
- A Nonsuccess would be an offer viewed and no offer competed

However, there are also customers that receive an offer and buy something nevertheless of the offer. Different data cleaning processes have already been made, e.g. filling in income and wrong age data with the normal distribution, for more detail see above. In the following the data is being prepared to fit better into a model and to create and label that can mark an transaction as successful or not.

The major part of the preprocessing is to define a label. For this purpose, a function has been implemented that goes through the whole dataset and checks with different conditions if a transaction is based on a former viewed offer. A label is assigned if not

	customer_id	event	time	amount	offer_id	offer_succ	num_off	duration	offer_type
0	0009655768c64bdeb2e877511 8f	offer received	168	NaN	5a8bc65990b245e5a138643cd4eb9837	0	0	72.0	informational
1	0009655768c64bdeb2e8775116 8f	offer viewed		NaN	5a8bc65990b245e5a138643cd4eb9837	0	0	72.0	informational
2	0009655768c64bdeb2e877511632db8f	transaction	228	22.16	NaN	0	0	NaN	NaN
3	0009655768c64bdeb2e877511632db	offer received	336	NaN	3f207df678b143eea3cee63160fa8bed	0	0	96.0	informational
4	0009655768c64bdeb2e8775	offer viewed		NaN	3f207df678b143eea3cee63160fa8bed	0	0	96.0	informational
5	0009655768c64bdeb2e877511632db8	offer received	408	NaN	f19421c1d4aa40978ebb69ca19b0e20d	0	0	120.0	bogo
6	0009655768c64bdeb2e8775116 2c 8f	transaction		8.57	NaN	0	0	NaN	NaN
7	0009655768c64bdeb2e877511632c 8f	offer completed	414	NaN	f19421c1d4aa40978ebb69ca19b0e20d	0	0	120.0	bogo
8	0009655768c64bdeb2e877511632db8	offer viewed		NaN	f19421c1d4aa40978ebb69ca19b0e20d	0	0	120.0	bogo
9	0009655768c64bdeb2e8775116321	offer received	504	NaN	fafdcd668e3743c1bb461111dcafc2a4	0	0	240.0	discount
10	0009655768c64bdeb2e8775116 2db8f	transaction		14.11	NaN	0	0	NaN	NaN
11	0009655768c64bdeb2e8775116 2db8f	offer completed	528	NaN	fafdcd668e3743c1bb461111dcafc2a4	0	0	240.0	discount
12	0009655768c64bdeb2e8775116320	offer viewed		NaN	fafdcd668e3743c1bb461111dcafc2a4	0	0	240.0	discount
13	0009655768c64bdeb2e877511632db8f	transaction		13.56	NaN	0	0	NaN	NaN

In this dataset before processing, we can see that the event doesn't follow a pattern which makes the definition of a successful offer difficult, as the offer received for an informational campaign can result in an purchase, but in between there were multiple other offers made, but not viewed. And Bogo or discount was only successful if the if the transaction is completed.

In further steps, all important data in linked back together and some new features are being calculated. Such as number of offers made to a customer, membership days. All data is changed into integer and numerical to be calculable. Categorical data is changed with one hot encoding.

customer_id	offer_id	mdays	num_off	amount	age	income	reward	difficulty	email	mobile	social	web	discount	informational	M	O	offer_succ
377511632db8f	0	619	1	22	33	72000	0	0	1	1	1	0	0	1	1	0	1
377511632db8f	1	619	3	8	33	72000	0	0	1	1	0	1	0	1	1	0	1
377511632db8f	2	619	0	14	33	72000	0	0	0	0	0	0	0	0	1	0	0
377511632db8f	2	619	0	13	33	72000	0	0	0	0	0	0	0	0	1	0	0
377511632db8f	2	619	0	10	33	72000	0	0	0	0	0	0	0	0	1	0	0

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13 entries, 0 to 12
Data columns (total 18 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   customer_id    13 non-null     object
 1   offer_id       13 non-null     int64
 2   offer_succ     13 non-null     int32
 3   num_off        13 non-null     int32
 4   amount         13 non-null     int32
 5   age            13 non-null     int32
 6   income         13 non-null     int32
 7   reward         13 non-null     int32
 8   difficulty     13 non-null     int32
 9   email          13 non-null     int32
 10  mobile         13 non-null     int32
 11  social         13 non-null     int32
 12  web            13 non-null     int32
 13  discount       13 non-null     int32
 14  informational  13 non-null     int32
 15  M              13 non-null     int32
 16  O              13 non-null     int32
 17  mdays          13 non-null     int32
dtypes: int32(16), int64(1), object(1)
memory usage: 1.1+ KB
```

Data Implementation

The data has been cleand and prepared for the implementation. The goal was to analyse the drivers of an effective offer, with the label variable being. First step therefore is to divide the dataset into features dataset and label dataset.

Afterwards the data is put into the model pipeline where all the features with high values a scaled to a min max range between 1 and 0. All binomial values are kept.

	mdays	num_off	amount	age	income	reward	difficulty	email	mobile	social	web	discount	informational	M	O
23734	0.215934	0.000000	0.001970	0.000000	0.496767	0.0	0.00	0	0	0	0	0	0	0	0
11823	0.291209	0.166667	0.006897	0.413793	0.477778	0.0	0.00	0	0	0	0	0	0	1	0
28263	0.458242	0.833333	0.028571	0.425287	0.466667	0.5	0.25	1	1	1	1	0	0	0	0
16217	0.475824	0.500000	0.014778	0.701149	0.911111	0.0	0.00	0	0	0	0	0	0	0	0
25855	0.028571	0.000000	0.003941	0.264368	0.144444	0.0	0.00	0	0	0	0	0	0	0	0

The Distribution of the label is a bit uneven, but not to imbalanced that it can not used for training anymore. Since the label is so imbalanced we use the f1 score to get a better grip on model performance to just accuracy. The f1 score takes both false positives and false negatives into account. F1 score gives more weightage to true positive, which is what we are looking for, as we want to see successful offers.

```
Results of the split
------
Training set has 31818 samples.
Testing set has 13637 samples.

Labels distribution
------
y_train labels distribution
0    25856
1     5962
Name: offer_succ, dtype: int64
y_test labels distribution
0    11078
1     2559
Name: offer_succ, dtype: int64
```

Define Function run model, to run three models in a row to compare the results to each other. For the best results, we compare three different models DecisionTreeClassifier, GaussianNB and RandomForestClassifier. Since we intend to analyze the feature importance to determine the drivers of an effective offer, a decision tree would provide good interpretability for us to analyze. Gausian NB is taken for a reference.

```
##########################################################
DecisionTreeClassifier trained on 31818 samples.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - -
MSE_train: 0.0000
MSE_test: 0.0000
Training accuracy:1.0000
Test accuracy:1.0000
              precision    recall  f1-score   support

           0     1.0000    1.0000    1.0000     11078
           1     1.0000    1.0000    1.0000      2559

    accuracy                         1.0000     13637
   macro avg     1.0000    1.0000    1.0000     13637
weighted avg     1.0000    1.0000    1.0000     13637
```

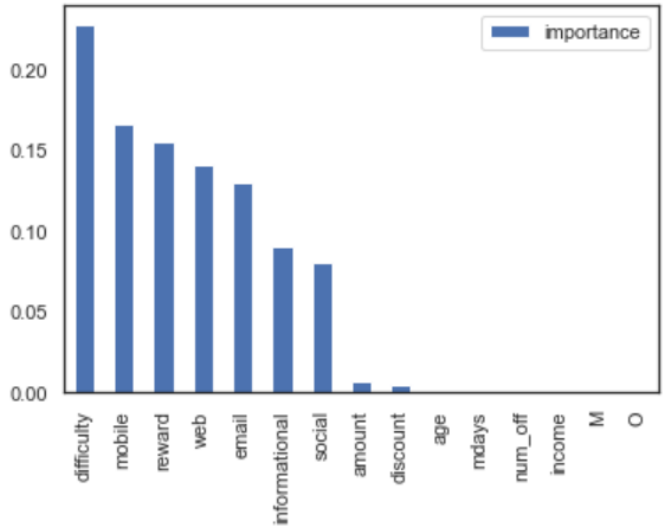
The Resulst is not usable, but I cannot find the reason why I get a perfect score. It also doesn't offer an error message. I already tried to see if I have a linear dependence in my dataset to the label, but also couldn't find it. The reviewer might have more insight to want went wrong.

Refinment: Although i seem to be getting a perfect score. Which is probably due to an error somewhere. After checking with Stackoverflow and other resources I could not find where the error is terminating. I also don't get an error, so I continue.

As it is an task in the Udacity Rubrics i will refine the model, although considering the score there is nothing to refine.

Results

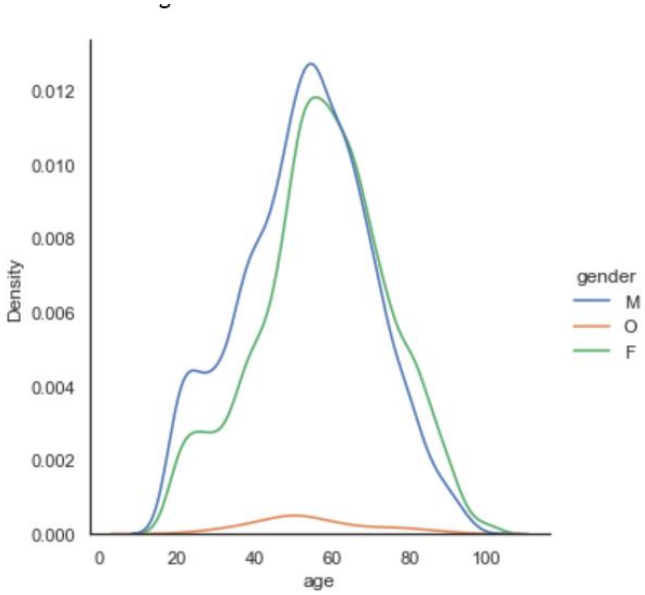
Question 1: What are the most important features to predict an offer



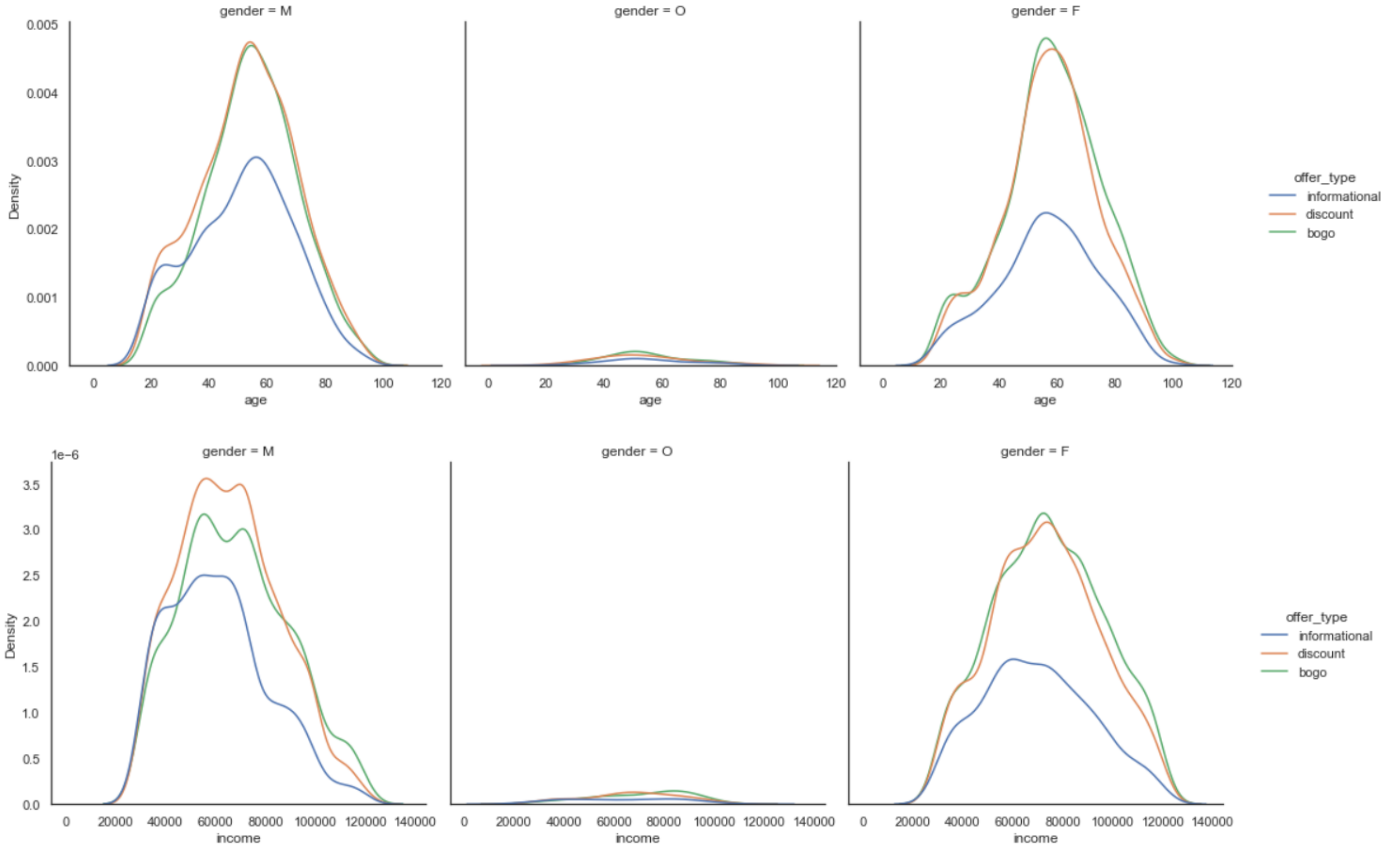
It seems that the difficulty and the mobile app are the best predictor to a successful offer. The difficulty was the minimum required spend to complete an offer. This is surprising but that what the prediction offers. Probably the less to spend to complete an offer the easier it is to be successful.

Question 2
From the below charts we can see that the offers that are completed is distributed across Gender, Age and Income.

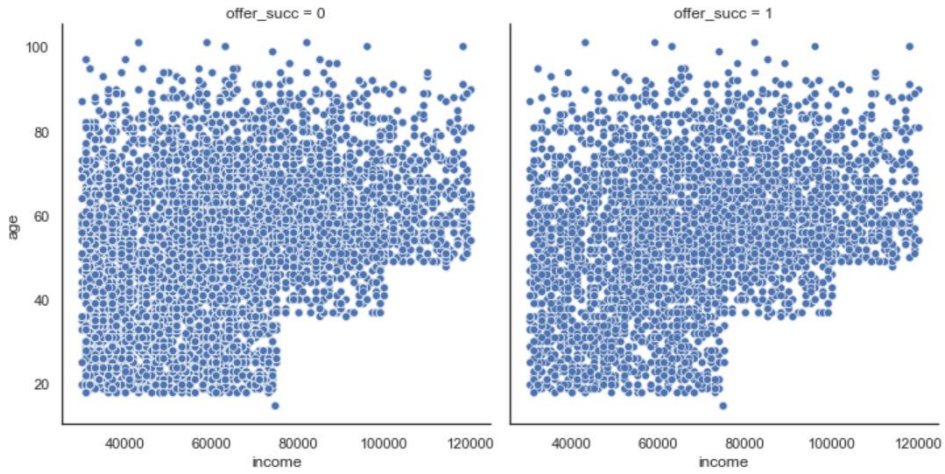
Based on Gender, the distribution look very similar and the density for the males that got a successful offer it a bit higher than females. Based on the Age group, the age range of 49-59 is most active while completing the offers followed by 59-69 and 39-49 age range.



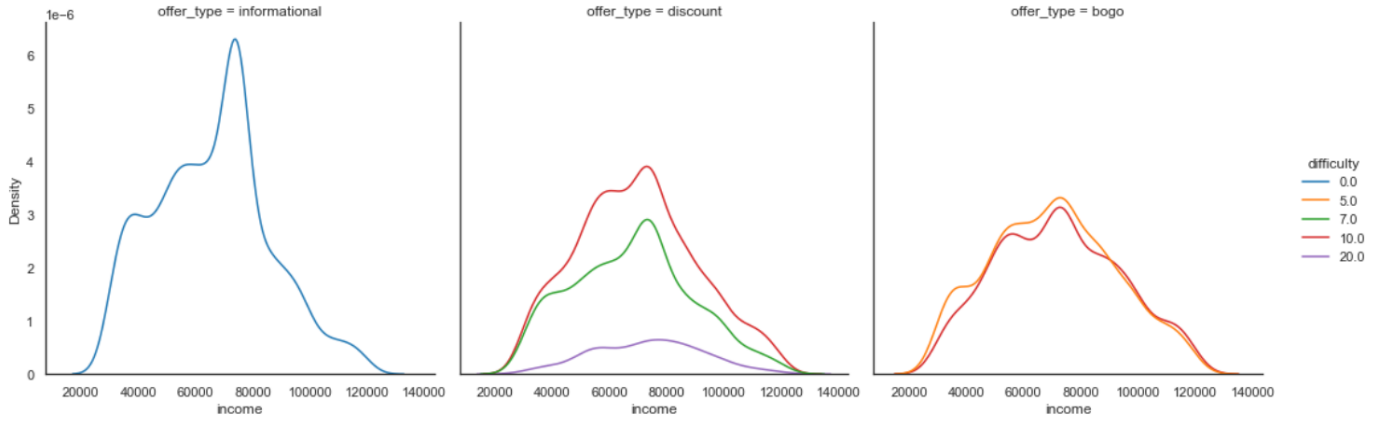
Separating by gender, we can see that with male and female the discount and bogo offer are the most successful. The information offer is not as successful. Looking also at the income we can see, females and males behave the same. People with income ranging from 50-80k are the most active ones to complete offers.



Looking further if there is a significant difference between succesful and non succesful offers regarding age and income. We can see from the figure below that, there is virtual no difference ragarding age and income distribution.



Looking lastly at the difficulty of an offer and dividing it by the offer type, we can see that with the discount offer the minimum amount of 10 Dollar is quit successful, which means when there is a discount people tent to buy more expensive drinks. The informational doesn't offer a difficulty because there is no money to spend on it.



Reflection

To solve the capstone project, the data was loaded, explored, visualized, preprocessed and put into three different models which were then refined with grid search. Started with three different data files, all were combined to get a label for a successful transaction based on an offer. The question of the most important feature in predicting the success of an offer where made with three different supervised learning models. Other questions were answered with visualizing the processed data.

Results showed that, the difficulty was the most important feature. In the second question we looked at what were the features of a successful offer and we found that the distribution is normal, where middle income and middle age where the most succcefull offers- Furthermore we found that an offer with a discount of 10 Dollar worked very well.

Challenging in this capstone problem was on one hand getting a understanding of the problem within the transcript data. Meaning that defining what an actual succesfull offer is, wasn't easy. Even harder was to find a solution within pandas capabilities and find a vectorized solution. I couldn't find a way to vetorzie the problem, there for I created a function that with a lot of if statement which where not ideal and slow in the calculation. Creating different models with scikit learn was medium difficult due to a lot of online available data and the lessons with its attached notebooks.

Improvement

- Finding a vetorized filtering of the transcript data set, to get a faster label column
- Optimizing the machine learning models and finding the error in the data that causes a perfect score
- Finding different question to ask and solve within the dataset