Project Overview

The Starbucks Capstone Challenge is an classic example of a sales problem where we want to make an offer to the customers that they can't refuse. We can solve the problem by looking at the past behavior of the customer and predict what items or offers the customer will be most interested in. By surfacing the right offers to the customer we can generate more sales. We have with us the information of past transactions of customers and related offers which were made to the customers.

Research and Citation

- https://www.sciencedirect.com/science/article/abs/pii/S0957417412006148
- https://martech.org/machine-learning-for-next-best-offers/
- https://towardsdatascience.com/implementing-a-profitable-promotional-strategy-for-starbucks-with-machine-learning-part-1-2f25ec9ae00c

Dataset

We had three datasets:

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

1. 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)
2. 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
3. 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

Problem Statement

We needed to device a solution which will tell us which customer should get which type of offer (discount, bogo, informational) or should the customer be given any offer at all. The goals of the system are as follow

```
~Predict, based on customer past transcript, what offer will be most suitable for him
```

To achieve this goal we performed following steps.

Explore and Process the Data

We explored the three dataset and tried to find the relationships between each of them. At the end of this steps we had achieved following outcomes

- Processed all the customer data
- Linked all the transaction data with the customer
- Link all the offer data with the customers

Clean the Data

Since the dataset had missing feature values we had to find a solution for them before we could train the model on the data. Following feature had missing values

- age
- gender
- income

Train a Model

- Finally we used the cleaned data to train a model and predict possible offer type for a particular customer

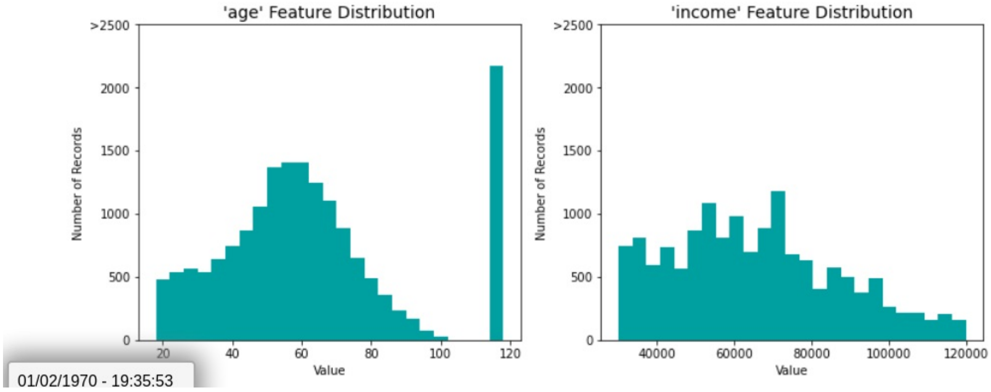
Data Exploration

The starbuck dataset had following statistical features

- Total Customers were 170000
- Total Transaction were 138953

  The customer dataset also had missing values for feature gender, age and income. Since our feature set for customer was already pretty limited and the test data space was not very large it was not an option to drop these missing values column.

  The general description of **age** and **income** feature is as following



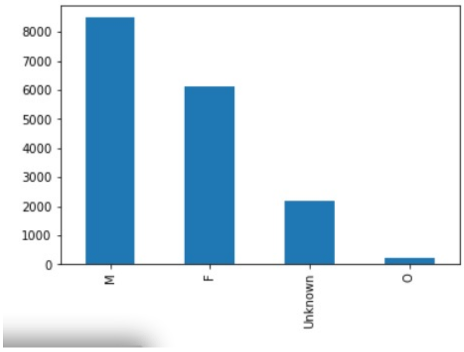
Age

We can assume the age **118** is a default age for all the customer whose age is unknown thus these rows can be considered as missing

Income

We can not see the missing row in the graph above but similarly to **age** there are **2175** missing income value that need to be filled before proceeding with that data

Gender



There are **2175** missing gender values.

Dealing with Missing Data

Since ~ 12% of data was missing we used machine learning to predict the missing data based on the data we have available

But before doing any kind of prediction we needed to add more feature to our customer dataset, so that we capture most of the variance available

Derived Features

- With the help of transcript data we added following 33 new columns to the customer Data Set

  total_offer_received
  Total number of offer received of any type by the customer
  total_offer_viewed
  Total number of offer viewed by the customer
  total_offer_completed
  Total number of offer completed of any type completed by the customer
  total_email_offers
  Total number of offers sent through email

total_web_offers
Total number of offers sent through web
total_social_offers
Total number of offers sent through social platform
total_mobile_offers
Total number of offers set through mobile
total_reward_earned
Total number of reward earned by completing the offers
total_spent_on_offer
Total amount of money spent at starbucks
average_offer_difficulty
Average difficulty of received offers
average_offer_duratio
Average expiry period of received offers
total_offer_completed_bogo
Total number of offer received of bogo type by the customer
total_offer_completed_discount
Total number of offer received of discount type by the customer
total_offer_completed_informational
Total number of offer received of informational type by the customer
average_offer_difficulty_bogo
Average difficulty of received bogo offers
average_offer_difficulty_discount
Average difficulty of received discount offers
average_offer_difficulty_informational
Average difficulty of informational offers
average_offer_duration_bogo
Average expiry period of received bogo offers
average_offer_duration_discount
Average expiry period of received discount offers
average_offer_duration_informational
Average expiry period of received informational offers
total_email_offers_bogo
Total number of bogo offers sent through email
total_email_offers_discount
Total number of discount offers sent through email
total_email_offers_informational
Total number of informational offers sent through
total_mobile_offers_bogo
Total number of bogo sent through mobile
total_mobile_offers_discount
Total number of discount sent through mobile
total_mobile_offers_informational
Total number of informational sent through mobile
total_offer_received_bogo
Total number of bogo offers recieved by customer
total_offer_received_discount
Total number of discount offers recieved by customer
total_offer_received_informational
Total number of informational offers recieved by customer
total_reward_earned_bogo
Total reward earned on bogo offers
total_reward_earned_discount
Total reward earned on discount offers
total_reward_earned_informational
Total reward earned on informational offers
total_social_offers_bogo
Total number of bogo offers sent through social channels
total_social_offers_discount
Total number of discount offers sent through social channels
total_social_offers_informational
Total amount spent on informational offers sent through social channels
total_spent_on_offer_bogo
Total amount spent on bogo offers
total_spent_on_offer_discount
Total amount spent on discount offers
total_spent_on_offer_informational
Total number of informational offers
total_offer_viewed_bogo
Total number of bogo offers viewed by customer
total_offer_viewed_discount
Total number of discount offers viewed by customer
total_offer_viewed_informational
Total number of informational offers viewed by customer
total_web_offers_bogo
Total number of bogo offers sent through web
total_web_offers_discount

Total number of discount offers sent through web

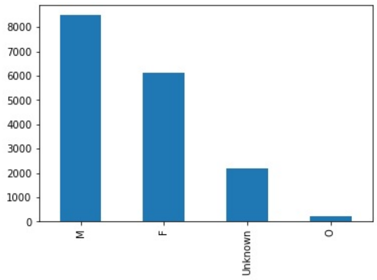
total_web_offers_informational

Total number of informational offers sent through web

Missing Gender Values

With the extended list of features, we then decided to use machine learning to predict the genders of customer with missing gender data. The reason behind using a machine learning approach is to make sure that the data portrays the same trend as the rest of the data.

We decided to benchmark multiple machine learning algorithm to see what will work best to predict these missing values. Since this was a multiclass classification problem we decided to use following algorithms to benchmark.



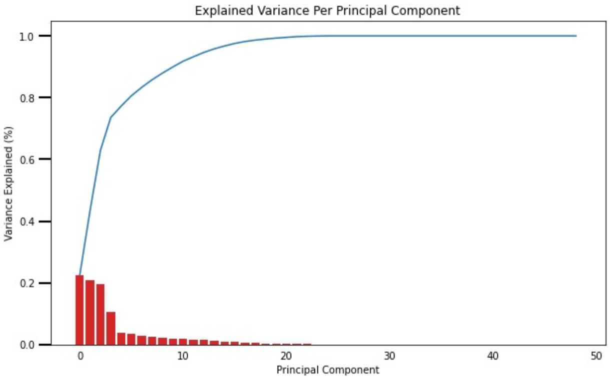
The distribution of gender is shown below The unknown bar shows the count of missing gender data

Feature Scaling

Since the data was not originally scaled and it is important scale the data. We decided to do simple standardization

PCA

Then we did Principle Component Analysis to find out the number of principle component that cover most of the data variance.



The explained variance graph showed that in order to cover the most of the explaned variance we needed atleast 25 principle component

With this informational we retrained the PCA.

Training and Benchmarking

1. Metrics Used

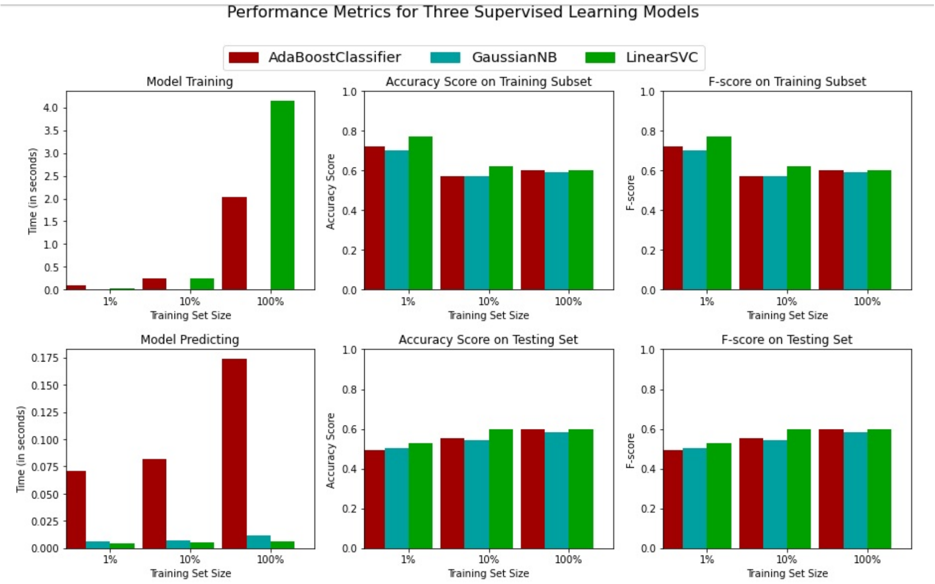
   We used the accuracy score, f{β} score where β is 0.5 and training time to compare the benchmarking models

2. Benchmarking

   For Benchmarking the model we used following the models

   - AdaBoost Classifier
   - Guassion Naive Bayes
   - Linear SVC

   All three models were trained on 1%, 10% and 100% of dataset to see their benchmark values. The result of benchmarking are shown



below

3. Training

   Since AdaBoost Classifier was fractionally better than other two and had very small training duration we decided to use to model the data and fine tune it further

   1. Tuning Parameters

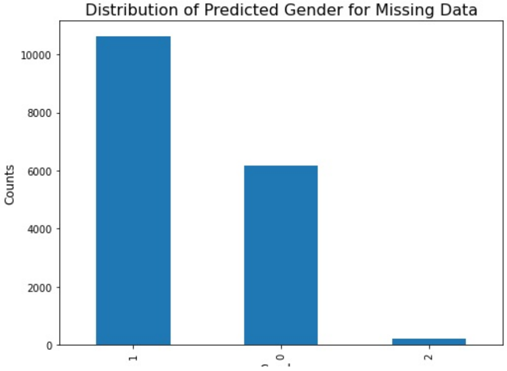
      n_estimators
      we used a range of 80 - 102 estimators on GridSearch to achieve the fine tuned model

   2. Tuned Model Score

      - We achieve 60% accuracy and 0.604 f{β} score.
      - Since the data we used to predict gender had very little co-relation with gender value thus this score is good enough and will atleast give better value than doing a random guess

Predict Data

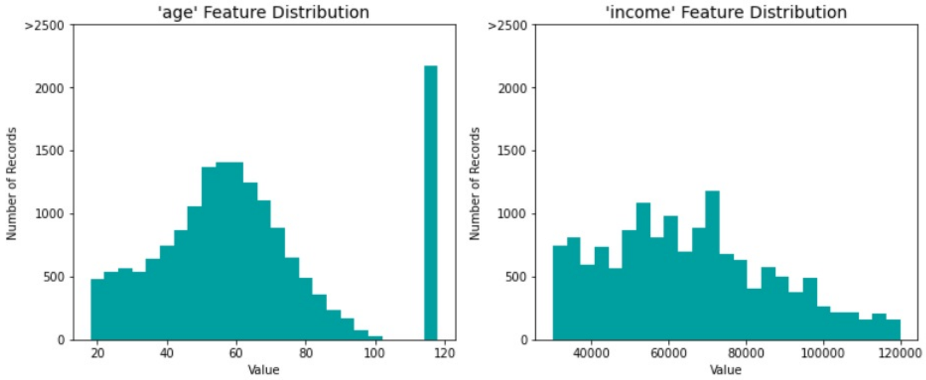
Finally we used the fine tuned model to predict the genders for missing ones. The distribution after prediction was following



Missing Income and Age Data

Similar to gender data we decided to predict the value of income and data column but unlike gender these values were Continuous and need regression algorithms to model them

Data Distribution



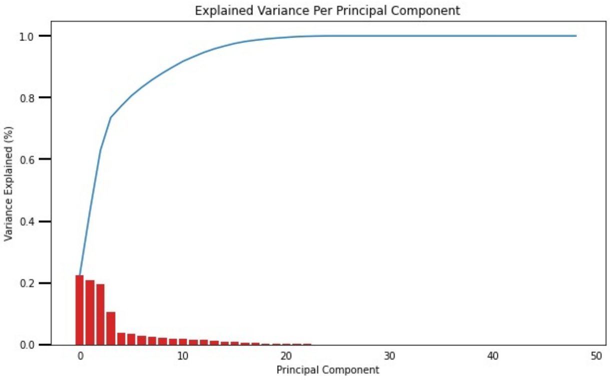
From the distribution of data we can see that large number of customer have age 118 which is probably default age set in case of missing data. So we first of all removed this value and replaced it with Nan, secondly, it looks like both the dataset are skewed to left side so the prediction should also be similar

Feature Scaling

Since the data was not originally scaled and it is important scale the data. We decided to do simple standardization

PCA

Then we did Principle Component Analysis to find out the number of principle component that cover most of the data variance.



The explained variance graph showed that in order to cover the most of the explaned variance we needed atleast 25 principle component

With this informational we retrained the PCA.

Training and Benchmarking

1. Metrics Used

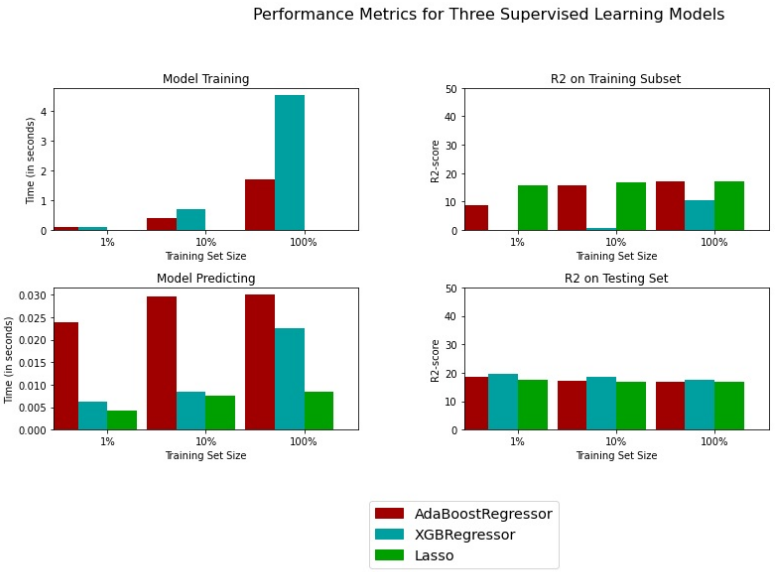
   Since these were regression problem we use following metrics

   - RMSE

2. Benchmarking

   The algorithm used in benchmarking were

   - AdaBoost Regressor
   - XGB Regressor
   - Lasso Bench Marking result is as followed



From the benchmark it turned out tha AdaBoost Regressor is the best option marginally.

3. Training

   1. Age

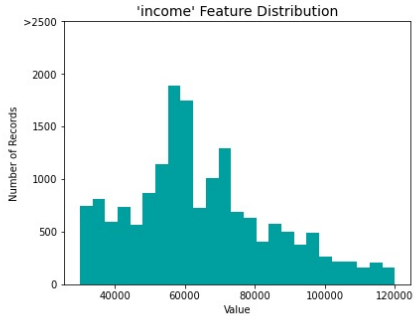
      - We achieved RMSE of ~16
      - That means the values predicted by our model had an error of around 16 years, but again since the data we used to predict is just the transactions data, it is highly unlike we could have achieved a better value with other models

   2. Income

      - We achieved RMSE of ~19468
      - That means the values predicted by our model had an error of around 19000 dollars, but again since the data we used to predict is just the transactions data, it is highly unlike we could have achieved a better value with other models

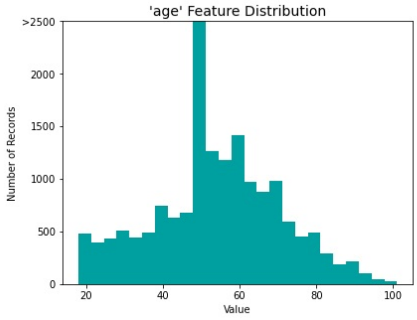
Predict Data

1. Income



Income value distribution after prediction

2. Age



Age value distribution after prediction

Modeling the Customer Data to Predict offers

Initially we had thought we will use classification model to do the prediction of offer type but in order to do that we need to manually label the customer data and based on some heuristics. So instead we changed the approach as we started the modeling customer data

Clustering instead of Classification

The idea behind this new approach was that the users have difference in their transaction behavior and they can be grouped or cluster based on those behavior and finally once we have found those clusters we can easily apply heuristics on those clusters to see which offer will fit which cluster

Dataset

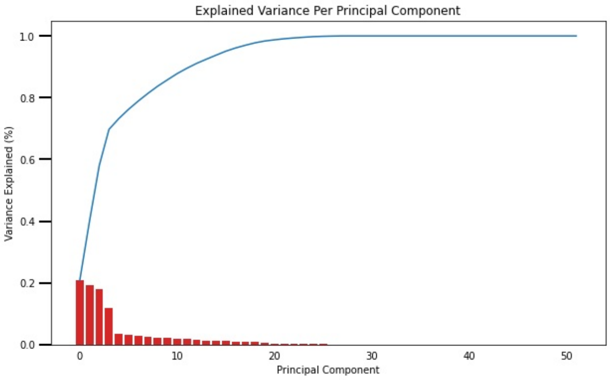
We had 170000 customers with 53 feature each. The data is not a lot but was be enough perform the clustering operation

Feature scaling

We standardized the features to make sure all of them will have the same importance while clustering

PCA

Then we performed the PCA with all the data that we had created, the graph looked similar to before.



1. Component Breakdown

   This time we also did a breakdown of top three component to see which features covered the most variance

   1. Principle Component 1

      - total_social_offers_discount
      - total_web_offers_discount
      - total_offer_received_discount
      - total_mobile_offers_discount
      - total_email_offers_discount

   2. Principle Component 2

      - total_email_offers
      - total_mobile_offers
      - total_offer_received
      - total_social_offers
      - total_web_offers

   3. Principle Component 3

      - total_social_offers_informational
      - total_web_offers_informational
      - total_offer_received_informational
      - total_mobile_offers_informational
      - total_email_offers_informational

   4. Principle Component Discussion

      - Looks like the most variance is captured by discount offers
      - Second thing that we can see from the above analysis is that the medium of delivering offer also captures a lot of variance
      - Informational offers also capture some variance

Modeling

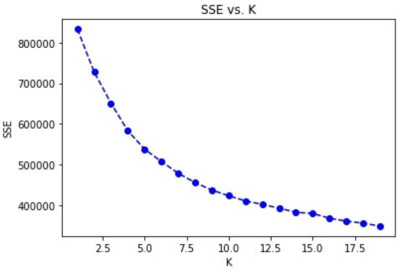
1. Choosing the clustering algorithm

   We decided to use K-nearest neighbors algorithm for clustering. Due to following reasons

   - It works on all kind of data
   - It can do multi cluster grouping

2. Finding the best n - number of clusters

To find the optimum number of clusters and we ran the algorithm multiple time on the data and used the elbow method to determine the optimum n



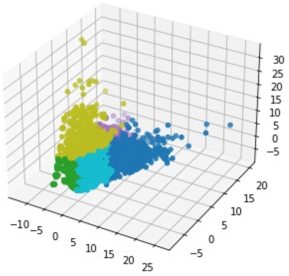
1. 6 is the optimum number of clusters

   Looking at the graph of Sum squared error vs K, we can see that 6 is the number of clusters which lies the elbow. This also falls inline with the fact that there are around the same numbers of offer categories

   - Discount
   - Bogo
   - Informational
   - No Offers
   - Sleeping Dogs

3. Predicting Cluster of customer

   Using the 6 as K we predicted clusters for each customer and assigned them. We also visualized the clusters to see how the show up on multi dimensional space

4. Grouping data on clusters

   After prediction we grouped the data over clusters so that we can apply some heuristics on each cluster and find out which offer would be best for which customer

   1. Heuristics

      Sleeping Dogs
      Group of customer whose spending is less during offers and more outside offers
      Discount
      Group who has completed more than 50% Discount offer and has completed more discount offer than any other type
      Bogo
      Group who has completed more than 50% Bogo offer and has completed more bogo offer than any other type
      Informational
      Group who has spent more than 30% during Informational offer and has viewed more isformational offer than any other type

   2. Cluster and offers

Cluster	Offer Type
0	Sleeping Dog
1	Discount
2	Discount
3	Discount
4	Bogo
5	Sleeping Dog

Conclusion

Now that we have a model that can predict the cluster of customer based on it past transaction, we can easily then decide which offer to send to them on quick heuristics. The model can also be retrained every 15 days on the latest customer data to identify new customer groups and also see any change in behavior.