Starbucks Capstone Project Report

Deepthi M | January 24, 2022

Udacity – AWS Machine Learning Engineer Nanodegree Program

I. Definition

Project Overview

Starbucks founded by Jerry Baldwin, Zev Siegl, and Gordon Bowker in 1971 is now the largest coffeehouse chain in the world. Through their constant improvisation and attention to detail, Starbucks have increased their customers immensely. Starbucks is further enhancing its customer experience with the introduction of a new Starbucks Rewards program.

Through this program, there are three types of offers that can be sent: buy-one-get-one (BOGO), discount, and informational. In a BOGO offer, a user needs to spend a certain amount to get a reward equal to that threshold amount. In a discount, a user gains a reward equal to a fraction of the amount spent. In an informational offer, there is no reward, but neither is there a requisite amount that the user is expected to spend. Offers can be delivered via multiple channels.

With this information, it would be useful to perform Customer Segmentation. Customer targeting and segmentation is a popular tactic to re-engage old customers, invite new ones and drive sales. Customer Segmentation is the process of dividing the customers into groups or clusters based on certain similarities they have so that we can market or sell to each of these groups properly.

It was in the 1950s that, customer segmentation became a formal part of modern system. The definition of customer segmentation as given by Smith in 1956 is *"**Market segmentation involves viewing a heterogeneous market as a number of smaller homogeneous markets, in response to differing preferences, attributable to the desires of customers for more precise satisfactions of their varying wants**"*.

Wedel and Kamakura also stated *"**Market Segmentation is an essential element of marketing in industrialized countries. Goods can no longer be produced and sold without considering customer needs and recognizing the heterogeneity of those needs**"* in their book *"**Market Segmentation**"*.

Therefore, customer segmentation mainly depends on three assumptions:

- The group of customers is heterogeneous
- But these heterogeneous groupings should have characteristics that can be recognized and examined.
- Exclusive promotions to fulfil the demands of customers

Problem Statement

The data set contains simulated data that mimics customer behaviour on the Starbucks rewards mobile app. Once every few days, Starbucks sends out an offer to users of its 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.

The problem statement is to combine transaction, demographic and offer data to determine which demographic groups respond best to which offer type.

Since this problem is based on customer segmentation, it is a clustering problem. Cluster analysis is using a mathematical and statistical model to detect populations of similar customers. Since this is a clustering problem (unsupervised learning), there is no target variable but instead looks at the relationships between the input variables. Like, the relationships between the offers and demographics which respond to them.

Proposed Solution

The first step would be to perform Exploratory Data Analysis (EDA) and describe the dataset. This EDA would expose various trends, patterns and relationships that are not visibly apparent. It will also help to identify obvious errors as well as detect possible outliers.

The second step would be data visualization to understand the dataset better and get a hold of the quantity of each demographic, each type of offer or event.

To identify which groups of people are most responsive to each type of offer, and how best to present each type of offer, various models including the benchmark model (K-Means) are analysed to determine which model helps us to solve the problem efficiently.

Apart from the benchmark model, the other model used to evaluate, will be the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) model. This will give us a better result as it will be able to work well with outliers and noise.

Based on the results of this model, we can determine certain useful features which will help for better marketing.

Metrics

Since this is a clustering problem (unsupervised algorithm) the metrics will help us to evaluate our model without the help of any labelled data.

The metrics I will be using are the Silhouette coefficient and the SSE Score. These metrics allow us to evaluate the models based on similarity or dissimilarity. All these together will help us finalize the number of clusters in our ultimate model.

Silhouette Coefficient: The silhouette value measures how similar a point is to its own cluster compared to other clusters. A high silhouette score is better.

SSE: Sum of the squared differences between each data point and its group's mean. If SSE is low, it is better.

II. Analysis

Dataset

profile. json
Rewards program users (17000 users x 5 fields)

- gender: (categorical) M, F, O, or null
- age: (numeric) missing value encoded as 118
- id: (string/hash)
- became_member_on: (date) format YYYYMMDD
- income: (numeric)

```
profile.head()
```

	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

First five rows of the profile file

portfolio. json

Offers sent during 30-day test period (10 offers x 6 fields)

- reward: (numeric) money awarded for the amount spent
- channels: (list) web, email, mobile, social
- difficulty: (numeric) money required to be spent to receive reward
- duration: (numeric) time for offer to be open, in days
- offer_type: (string) bogo, discount, informational
- id: (string/hash)

```
portfolio.head()
```

	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

First five rows of the portfolio file

transcript. json

Event log (306648 events x 4 fields)

- person: (string/hash)
- event: (string) offer received, offer viewed, transaction, offer completed
- value: (dictionary) different values depending on event type
    - offer id: (string/hash) not associated with any "transaction"
    - amount: (numeric) money spent in "transaction"
    - reward: (numeric) money gained from "offer completed"
- time: (numeric) hours after start of test

```
transcript.head()
```

	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'}

First five rows of the transcript file

Data Exploration

1. Profile Data

```
Shape of DataFrame: (17000, 5)

Features : ['gender', 'age', 'id', 'became_member_on', 'income']

Description of DataFrame:
                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

Number of null values:
 gender              2175
age                    0
id                     0
became_member_on       0
income              2175
dtype: int64
```

The Profile Dataset, there are 17000 rows and 5 columns. The statistical measures for the data have also been calculated using the describe () method. The Profile Dataset has quite a large number of null values in the 'gender' and 'income' columns. A pair-plot is also plotted to get a better idea of the data.

2. Portfolio Data

```
Shape of DataFrame: (10, 6)

Features : ['reward', 'channels', 'difficulty', 'duration', 'offer_type', 'id']

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

Number of null values:
 reward        0
channels      0
difficulty    0
duration      0
offer_type    0
id            0
dtype: int64
```

In the Portfolio Dataset, there are 10 rows and 6 columns. The statistical measures for the data have also been calculated using the describe () method. The Portfolio Dataset doesn't have any null values or NaN values. A pair-plot is plotted to get a better idea of the data.

3. Transcript Data

```
Shape of DataFrame: (306534, 4)

Features : ['person', 'event', 'value', 'time']

Description of DataFrame:
                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

Number of null values:
 person    0
event     0
value     0
time      0
dtype: int64
```

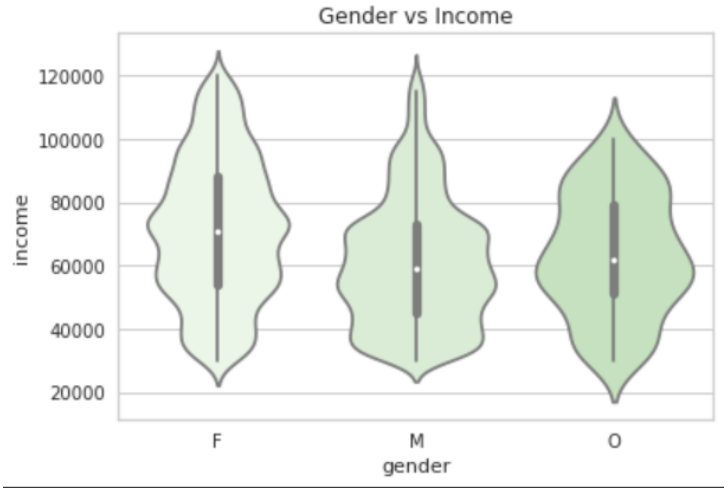
In the Transcript Dataset, there are 306534 rows and 4 columns. The statistical measures for the data have also been calculated using the describe () method. The Transcript Dataset doesn't have any null values or NaN values. A pair-plot is plotted to get a better idea of the data.

There are a few categorical variables in the three datasets like, age, gender, offer_type, event and channels.

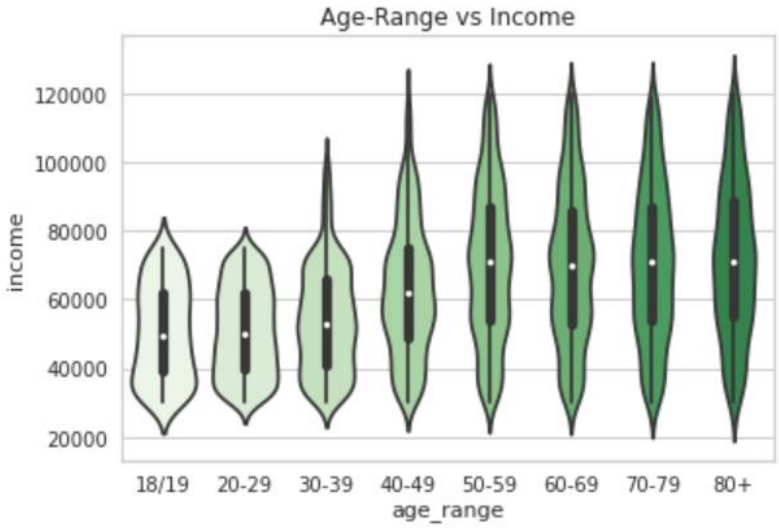
Exploratory Visualization

Various visualizations have been undertaken for a better understanding of the dataset. This also helps to extract a particular feature or summarize the entire dataset.

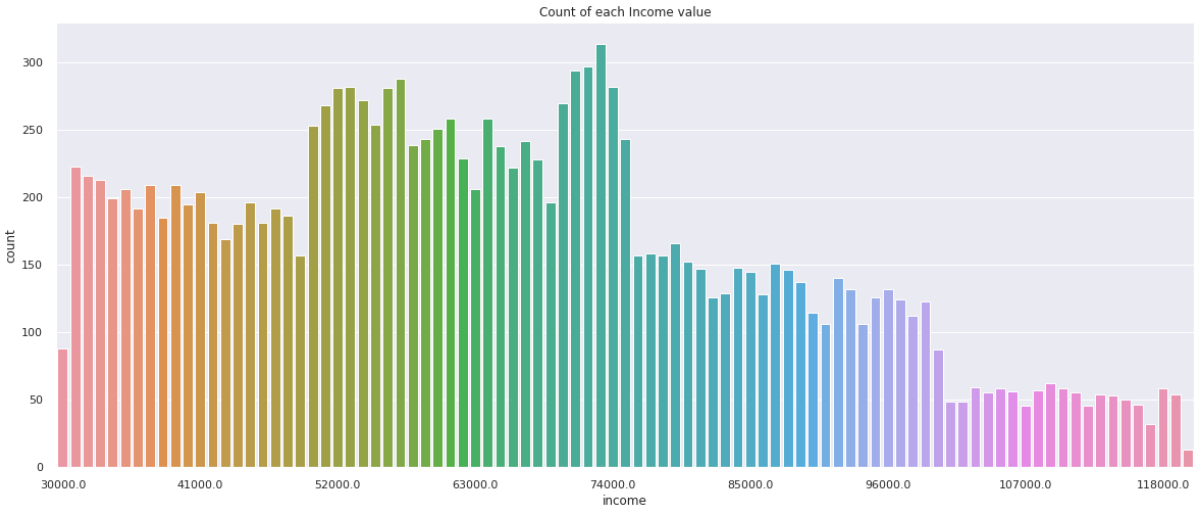
1. Gender vs Income
   This plot relates the 'Gender' with the 'Income'. This will help us understand the relationship with 'Gender' and 'Income', if any. This will also help determine if certain genders are more inclined towards a particular income and therefore will respond more to offers.



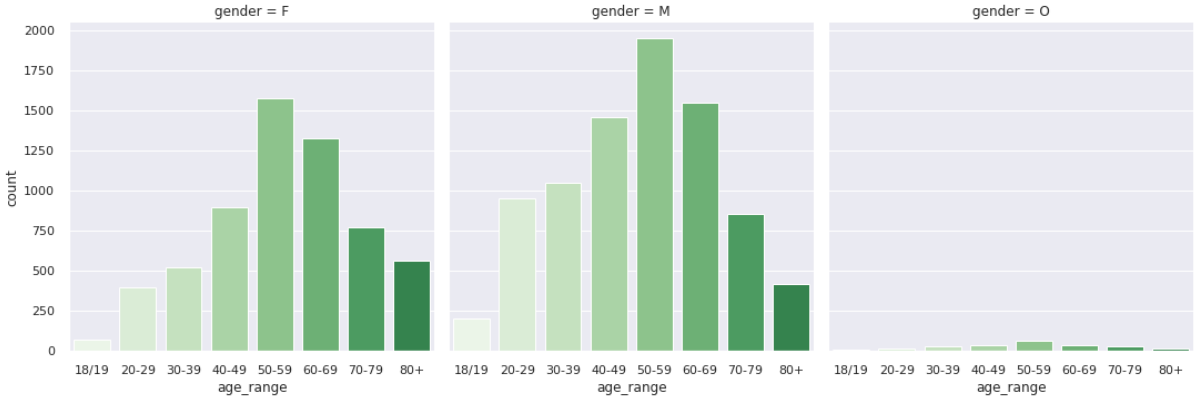
2. Age-Range vs Income
   This plot relates the 'Age-Range' with the 'Income'. This will help us understand the relationship with 'Age-Range' and 'Income', if any. This will also help determine, like the previous one, if certain ages are more inclined towards a particular income and therefore will respond more to offers.



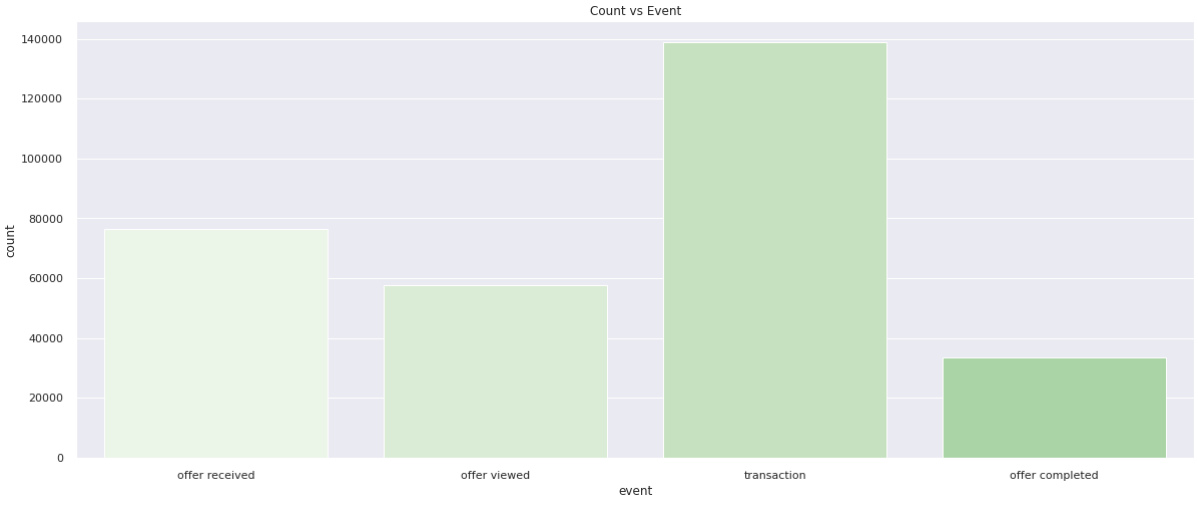
3. Income Count
   The count of people for a particular income will also help with the determination of how many people will be inclined towards responding to an offer.



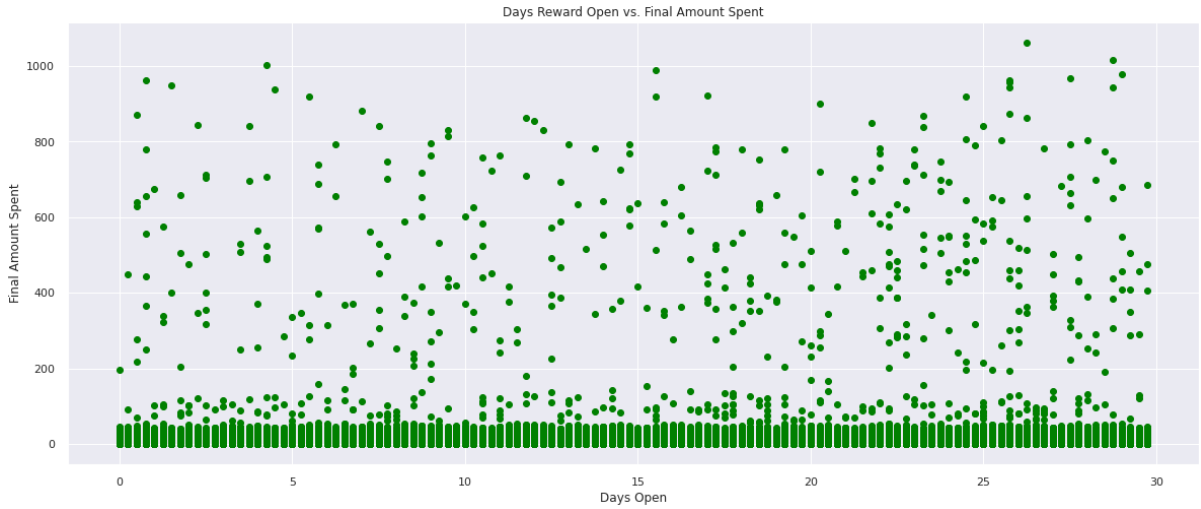
4. Age-Range vs It's Count vs Gender
   This is just to understand the distribution of the data through the files.



5. Event vs Count
   This plot helps us to visualize the distribution in the number and types of the offers in the dataset.



6. Days Reward Open vs. Final Amount Spent
   This plot helps visualize the relationship between the days a reward was open to the final amount that was spent.



Algorithms and Techniques

The algorithms and techniques used in this project were:

- Sklearn pre-processing
- PCA
- K-Means
- HDBSCAN

1. Sklearn pre-processing is often used to pre-process, prepare, transform the data to fit into the model. It changes the raw features into a form that is more usable by the estimators.
   1.1. The dataset is also scaled and standardized. Scaling is normalizing the features of the data. Standardization is a scaling technique where all the feature values are centred around the mean and therefore have a value equal to 0 and a standard deviation of 1.
   1.2. These techniques help us to bind the dataset together by making then uniform and will prevent only certain features from taking over the entire dataset.

2. PCA stands for Principal Component Analysis and is yet another unsupervised algorithm. This is most commonly used for reducing the dimensionality of the dataset and the complexity which makes the machine learning algorithms like K-Means run faster and easier. Using PCA ensures that there is also no loss of data even though the dimension is being reduced.

3. K-Means is also an unsupervised algorithm which is widely used for clustering problems. This clusters the data by separating it into clusters or groups of equal variances and aiming at less SSE within each cluster. This algorithm however, requires the number of clusters to be mentioned in prior. The three steps in this algorithm are:
   3.1. Choose k sample from the dataset. Then loop through the next two steps
   3.2. Assign each sample or data point to its nearest cluster
   3.3. Creating new centroids by taking the average of all data points in the preceding centroid.

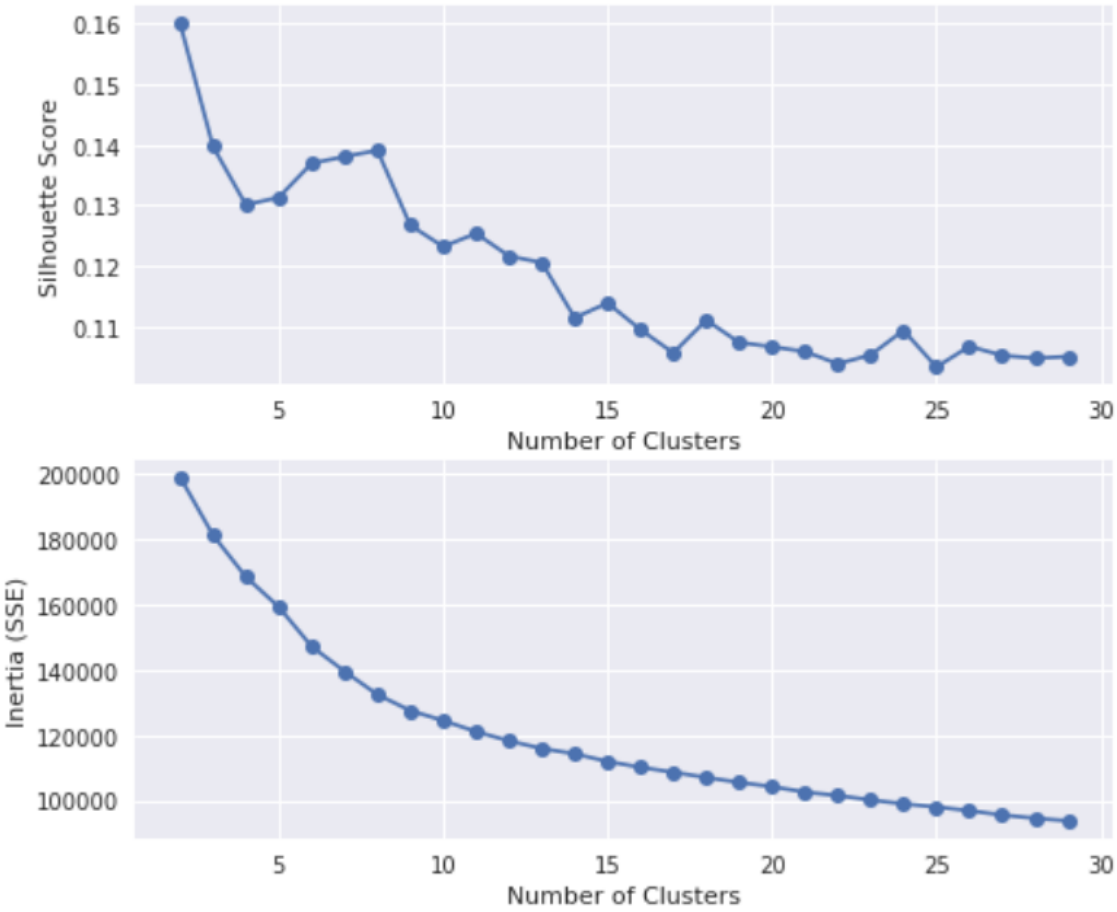
4. HDBSCAN stands Horizontal Density Based Spatial Clustering Application of Noise. This algorithm is an extension of DBSCAN by changing it to a hierarchical clustering algorithm. There are five steps involved in HDBSCAN algorithm (from the HDBSCAN docs):
   4.1. Transforming the space according to density
   4.2. Construct a minimum spanning tree from a distance weighted graph
   4.3. Construct a cluster hierarchy of connected components
   4.4. Condense the cluster hierarchy based on minimum cluster size
   4.5. Extract the stable clusters from the condensed tree

The dataset once pre-processed will be fit into the K-Means model, followed by its evaluation. The data is then fit through the PCA and then the HDBSCAN to get a clear idea of the clusters.

Benchmark

The benchmark model used for this project is the K-Means. The K-Means is an unsupervised algorithm that helps us with clustering problems. Our data is first fit with the K-Means after passing it through the StandardScaler. This is because K-Means doesn't perform automatic scaling.

The K-Means is measured using the Silhouette score, and the SSE score. These quantitative scores not only help us to determine the number of clusters but also to improve our model later.



K-Means was chosen as the benchmark as although it works fairly well, it may not be able to detect outliers or noise. It also has another disadvantage of having to predefine the number of clusters. These problems can be fixed using our HDBSCAN model.

III. Methodology

Data Pre-processing

Profile Data Pre-processing

- Fixing the null values in the gender and income columns wherever the age is 118.
- Replacing the column's null values with mean values
- Renaming the column names for better understanding
- Converting datatype of 'became_member_on' column to datetime

- Including a new column named 'days_as_member' which indicates the number of days a member has been a member till 26th July 2018 (the latest date present in the dataset)
- Creating a new 'age_range' column based on 'age'

Portfolio Data Pre-processing

- Renaming the column names for better understanding
- One hot encoding 'channels' and 'offer_type' columns. The encoding is done to ensure the data is able to fit well into our clustering model
- Dropping the above original columns after encoding

Transcript Data Pre-processing

- Renaming the column names for better understanding
- Removing the customers that are not present in the profile file
- One hot encoding 'event' column
- Converting 'time' column to 'days' along with appropriate values
- Separating the key: value pairs in the 'value' column into two different columns 'transcript_amount' and 'transcript_offer_id'

Implementation

The important part of the implementation process is combining all the three datasets together, with their columns and features in mind. The combining is done by collecting all the data for each customer id. Additional features are also introduced by calculating the count and average of BOGO and discount offers, as well as the total number of offers that have been viewed, completed, received, and spent to name a few.

These additional features help us add more functionality to the dataset and understand the number of clusters better.

Some features are also dropped like the 'customer_id' since it is unique and the 'became_member_on' as its datatype is datetime.

The dataset is then passed through an Imputer to double check if there are any null values after combining and thus replacing these with the median value of the column.

Then it is scaled and standardized with the help of the StandarScaler. This returns a scaled ndarray. This is finally ready to go through the PCA.

The result of the PCA is now ready to go for K-Means. The SSE and Silhouette Score for each graph is determined and the best cluster is chosen. The K-Means model is prepared and visualized.

Advanced PCA is applied on the original data and is prepared for HDBSCAN. A training job is done and an endpoint is deployed. Using this endpoint, we can use to "predict" our clusters in HDBSCAN.

The labels are obtained and added to our data. The data can now be analysed.

Refinement

The initial solution is the K-Means. This was the benchmark model. For this model, the data was just scaled using only the StandardScaler. The model produced intermediate results.

The parameters had to adjusted in order to have better results. Therefore, the data went through a detailed PCA algorithm. I fetched only the top variance contributing components and this was later fit into the HDBSCAN.

This was the final solution. The HDBSCAN with the transformed data performed far better than the K-Means as the number of clusters were clearer and more put together and not all over the place. Initially the amount of noise was almost over 60%, but after the refinement this reduced to only 0.6%. This helped in easier understanding of the customers and the reward system.

I also decided to keep the data classified as noise as it adds to the credibility.

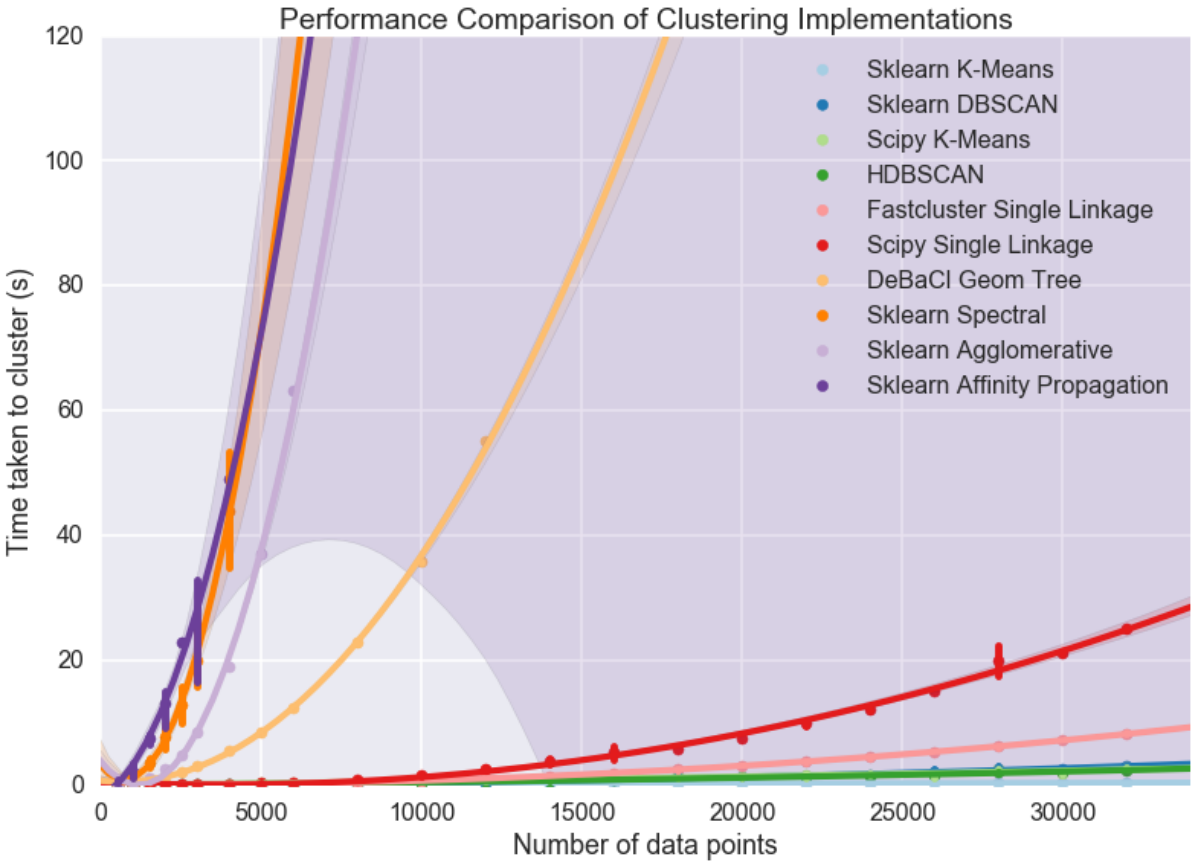
IV. Results

Model Evaluation and Validation

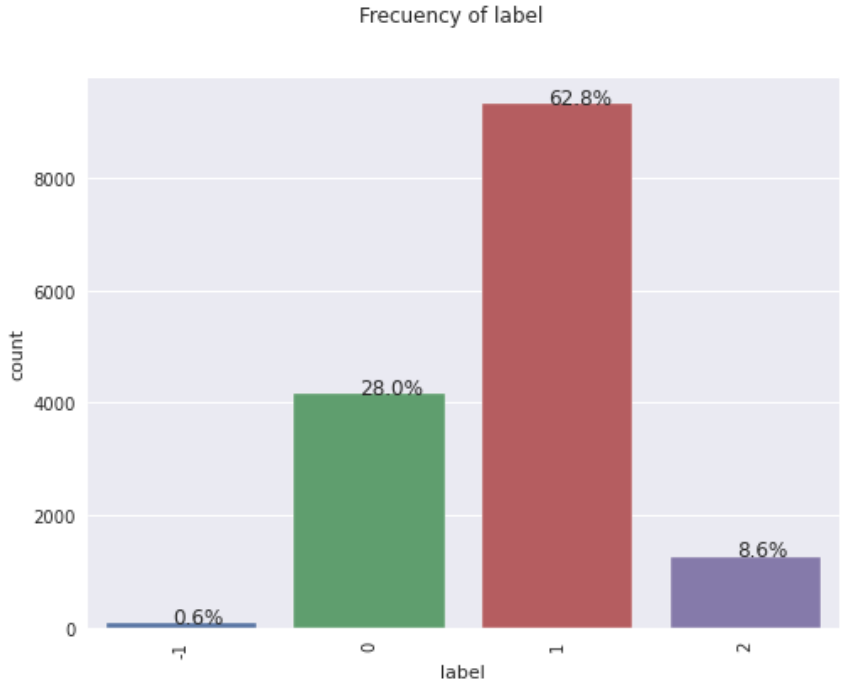
HDBSCAN

The HDBSCAN model was chosen as it has all the advantages of DBSCAN but also overcomes a disadvantage of it not able to handle varying density clusters. HDBSCAN is very efficient and does all the work for us, so therefore it is very easy to use.

The comparison of HDBSCAN with other clustering algorithms gives us a fair idea of how efficient it is.



Source: HDBSCAN Docs

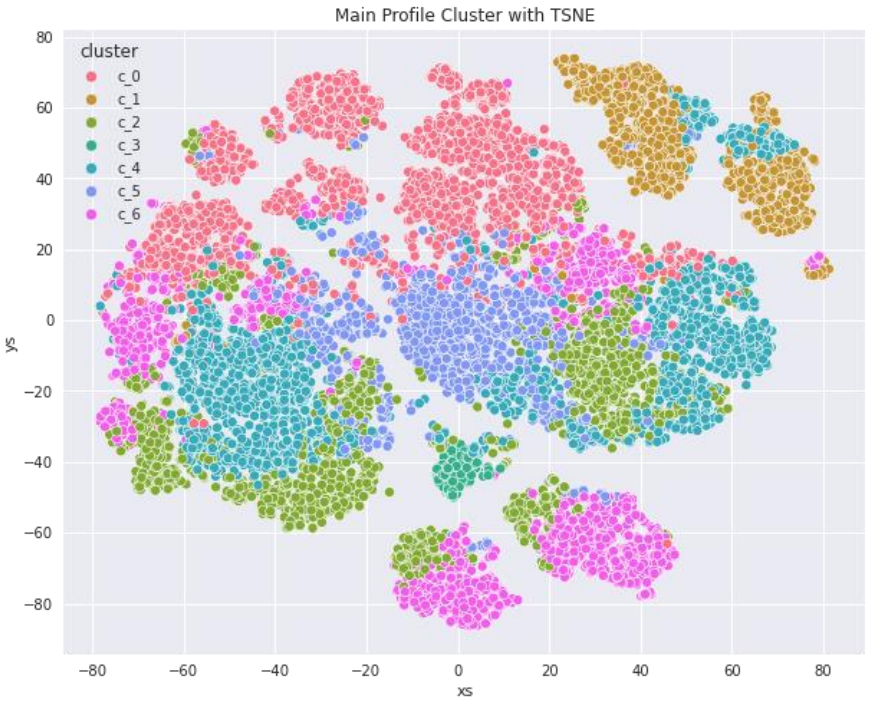


Looking at the distribution of the data and the amount of noise, the HDBSCAN fits pretty good for our data.

Justification

In the initial benchmark model (K-Means) the number of clusters were found to be 7. This was a reasonably large number of clusters for the dataset given.

The number of clusters could have also been 2 as it had a high Silhouette Score but its SSE score was also high which is not desired so the next best result was 7.



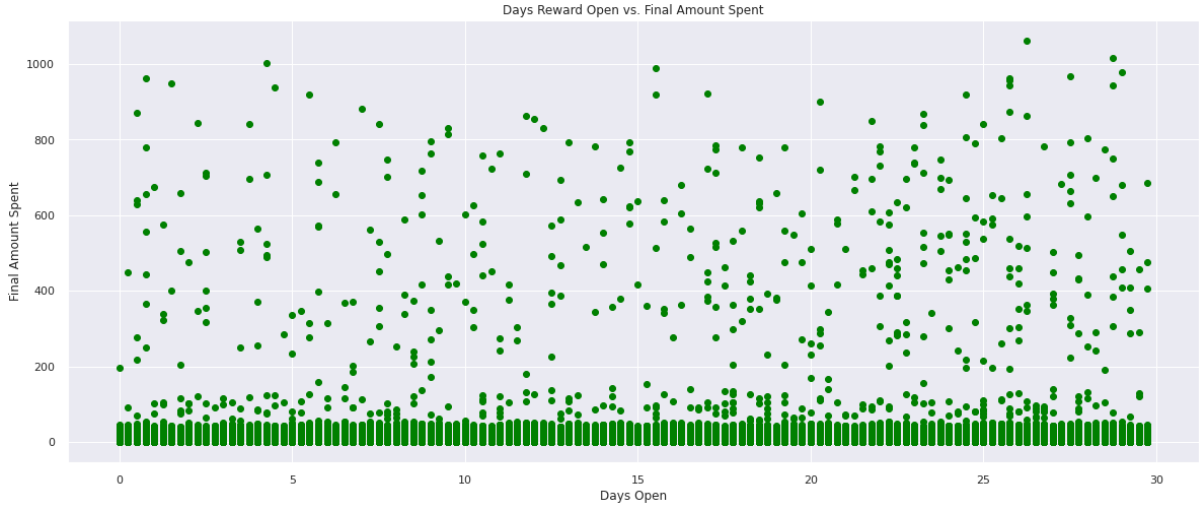
Although this model performed good, I felt the fitting of the data was incorrect and the number of clusters being this high will not prove to be useful to us.

Using the HDBSCAN model, the number of clusters came to a 4, including noise. Since the noise was insignificant, the final number of clusters were 3. This compact number of clusters will help us to group our customers better and provide quantitative and helpful results for marketing.

The amount of noise was found to be 0.6%, followed by cluster_0 at 28%, cluster_1 at 62.8%, cluster_2 at 8.6%.

V. Conclusion

Free-Form Visualization



I feel this visualization provided the most important feature of this project. This shows us the relationship between the 'Number of Days an Offer is Open' and the 'Final Amount Spent'.

This provides us with an important fact, that people or customers are more likely to respond if the offer is open for a longer time. We can see the amount spent increases after 15 days. Therefore, if the offer is valid for at least half a month, customers tend to spend more.

This not only increases the money flow but also would increase the number of customers as it acts as a potential marketing strategy.

Reflection

Overall, this was one big learning experience for me and I had lots of fun. Working on this project was a whole rollercoaster from start to end. I learnt a lot about machine learning in general and the steps in it. It introduced me to new algorithms and techniques, while teaching me about the best practices while creating and training machine learning models. I have always been interested in AWS Sagemaker and this was a great learning path for me.

All of the projects were very interesting and really made me use my brain. The capstone project was in particular challenging as we had to do everything from scratch, but all in all it was a fun process.

The final model and solution fit my expectations and the various steps mentioned in this report would surely help to solve every machine learning problem.

Improvement

Some improvements I would make would be to explore the dataset even more and research about each feature explicitly. The number of data was quite less for building a proper model, but it all worked out in the end.

I initially considered using DBSCAN for my main model, but I not only found it easier to use HDBSCAN but was also able to understand it's working better.

If I had more time, I would have definitely researched more about DBSCAN and tried to implement it here.

There always exists a better solution than the one at hand, I will continue to explore and improve the current solution.