ONLINE RETAIL CUSTOMER SEGMENTATION CLUSTERING

SUMIT YADAV, Data Science Trainee, AlmaBetter, Bangalore

INTRODUCTION

Customer segmentation is the process of classifying customers based on their shared behaviour or other attributes. The groups should be homogeneous within them and should also be heterogeneous to each other. The main goal is to identify customers that are most profitable and loyal and the ones who churned out, to prevent further loss of customers by redefining company policies. Having a large number of customers, each with different needs it is difficult to find which customer is most important for business and target them with an appropriate strategy.

PROBLEM STATEMENT

Identify major customer segments on a transnational dataset that contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail. All customers have different-different kinds of needs. With the increase in customer base and transactions, it is not easy to understand the requirement of each customer. Segmentation can play a better role in grouping those customers into various segments.

INSPECTING DATASET

1. Importing the dataset and checking the head and tail rows to get an overall idea.
2. After that exploration of the dataset, by checking the info which gives us some intuition about null values and data types of columns present in it.
3. And then we checked the descriptive summary of the data frame which gives some quantitative idea about the dataset i.e., average values of the columns, frequency of the

values, variability, and dispersion concerns on how to spread out the values.

DATA CLEANING

Data Cleansing is the process of detecting and changing raw data by identifying incomplete, wrong, repeated, or irrelevant parts of the data. In our data, there is a feature called Customer-ID which has more than 24% missing values, Hence there is no use in having the data with no customer assignment. So we dropped it. Coming to the duplication values - We have encountered some duplicated observations in the dataset, When you have frequent duplicates in our database, you may inadvertently send multiple clustering messages to the same person CustomerID. As a Consequence, we dropped it.

FEATURE ENGINEERING

- Checking the numbers of missing values.
- Next we dropped the missing rows from the datasets because if customer records are not present other values related to that id will be meaningless in separating the cluster.
- Finally ended up with a data frame shape of 401604 records with 8 columns.
- Extracting the new feature named Year, Month, Day, Hour from the date column called InvoiceDate.
- Creating a new feature 'TotalAmount' by multiplying Quantity and UnitPrice.
- Creating a new feature 'TimeType' based on hours to define whether it’s Morning, Afternoon, or Evening.
- Checking the number of cancellations by each customer, where the InvoiceNo starting with 'C' represents cancellation.

- Renaming the count of the cancellation data and checking the top five entries o it.
- Dropping the cancellation data records from the main dataset.

EXPLORATORY DATA ANALYSIS

After feature engineering, we did some data analysis and drew some hypotheses like -

the United Kingdom has the most number of customers and most numbers purchase history as compared to other countries and Orders with mass quantity are placed by the customers from the Netherlands.

RFM - RECENCY, FREQUENCY, MONETARY

we can use the RFM-based model for finding segments where R is Recency (how recently a purchase happened), F is Frequency (how frequent transactions are made), and M is Monetary value(Value of all transactions). Recency, Frequency, and Monetary score for each customer is calculated. The latest date is assigned as a placeholder to calculate recent purchases. All the transactions are grouped using CustomerID and then aggregate lambda operations are performed. As a result of this operation, numbers will be obtained which depict the recency., frequency, and how much a specific customer spent to date. All these are stored in a new data frame RFM. Earlier the distributions of Recency, Frequency, and Monetary columns were positively skewed but after applying log transformation, the distributions appear to be symmetrical and normally distributed.

Perception

1. If the RFM of any customer is 444. His Recency is good, frequency is more and Monetary is more. So, he is the best customer.
2. If the RFM of any customer is 111. His Recency is low, frequency is low and Monetary is low. So, he is the churning customer.
3. If the RFM of any customer is 144. He purchased a long time ago but buys frequently and spends more. And so on.
4. Like this we can come up with several segments for all combinations of R, F, and M based on our use case. The higher the RFM score, the more valuable the customer is.

NORMALIZATION OF THE DATA

In machine learning, some feature values at times differ from others multiple times. The features with higher values will always dominate the learning process. Before giving our data to clustering algorithms we need to perform the data normalization task (i.e StandardScaler) which will give equal importance to each variable so that no single variable drives the model performance.

CLUSTERING

Customer segmentation has been demonstrated to benefit from clustering. Clustering is a sort of unsupervised learning that allows us to locate clusters in unlabelled datasets. Clustering techniques include Binning, Quantile based, K-means, hierarchical clustering, and DBSCAN clustering.

BINNING RFM SCORES

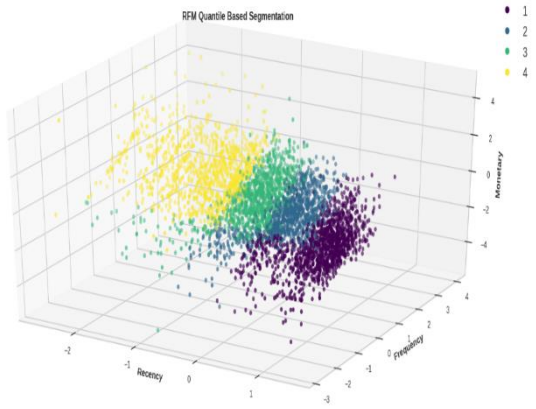
The original data values are divided into small intervals known as bins and then they are replaced by a general value calculated for that bin. The number of categories (bins) to use for each component to be created by RFM scores. The total

number of possible combined RFM scores is the product of the three values.

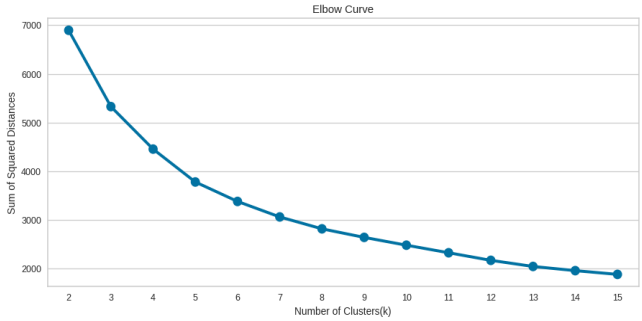
QUANTILE BASED CLUSTERING

After binning the RFM score We

Figure 1 -QUANTILE BASED SEGMENT



applied quantiles clustering to our dataset. It allows for within-cluster



skewness and internal variable scaling based on the within-cluster variation. In quantile based we used the number of categories(cut) to use for each component to be created by RFM value. we have fetched the RFM values Group for each customer by combining the factors R, F, and M. The method is fast and simple and can deal with large datasets. The drawback of this clustering creates several overlapping data points of the clusters.

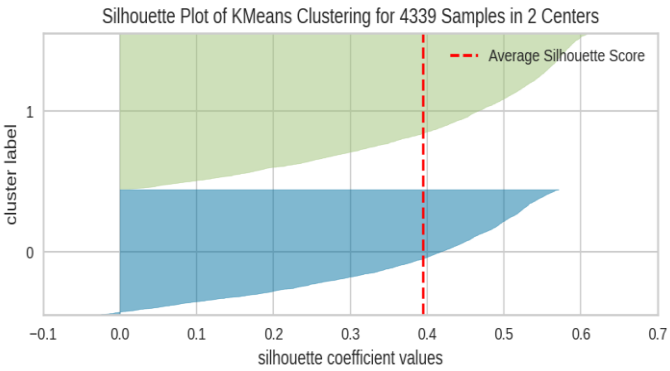
K-MEANS CLUSTERING

K-Means is an unsupervised learning algorithm used for clustering tasks that work well with complex datasets. It is an iterative algorithm that partitions the dataset into “k” pre-defined non-overlapping subgroups (clusters) where each data point belongs to only one group.

It is important to determine the optimum number of clusters i.e, “k value”. For this, we used the Elbow method. It involves running the algorithm multiple times over a loop with an increasing number of cluster choices and then plotting a score as a function of the number of clusters. When “k” increases, the centroids

are closer to cluster centroids. The improvement will decline at some point rapidly creating an elbow-like shape in the graph and that is the whole reason this method is called the elbow. We take the count of the cluster, k-value at the point where this elbow is bending. When we executed this metric, the result was not obvious and the bend is not clear as there was a sudden decline at three values – 2, 4, and 5.

The elbow method also picks up the range of the k values and draws the silhouette graph. It calculates the silhouette coefficient of every point. It calculates the average distance of points within its cluster and the average distance of the points to its next closest cluster. The plot of the silhouette is between -1 to 1. Observing the plot and checking which of the k values is closest to 1, i.e n_cluster= 2, which also has fewer outliers which means a less negative value.



EXTRACT :

From the Elbow curve, 5 appears to be at the elbow and hence can be considered as the number of clusters. n_clusters=4 or 6 can also be considered based on the objective of segmentation.

1. If we go by the maximum Silhouette Score as the criteria for selecting an optimal number of clusters, then n_clusters=2 can be chosen.
2. If we look at both of the graphs at the same time to decide the optimal number of clusters, we can take the intersection of the set of good

n_clusters obtained from both the graphs. So 4 appears to be a good choice, having a decent Silhouette score as well as near the elbow of the elbow curve.

The problem with the k-means clustering is that it is sensitive to outliers means Cluster Centroids can be dragged by

outliers, or outliers might get their cluster instead of being ignored. And Choosing the optimal k values manually is a tough job, This issue arises when the elbow curve tends to show the smoothening effect which signifies hard to find the optimal value of k concerning the precipitous degradation of the sum of squared distances.

HIERARCHICAL CLUSTERING

Hierarchical clustering is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters, where each cluster is distinct from the other cluster, and the objects within each cluster are broadly similar to each other.

Two techniques are used in this algorithm- Agglomerative and Divisive. We have used the agglomerative approach which starts by making n clusters and aggregates the data points until K(2,3) clusters are obtained.

Interpretation:

1. We can set a threshold distance and draw a horizontal line (Generally, we try to set the threshold in such a way that it cuts the tallest vertical line). We can set this threshold at 70 or 50 and draw a horizontal line in the dendrogram graph.
2. The number of clusters will be the number of vertical lines which are being intersected by the line drawn using the

threshold. The larger threshold (y=70) results in 2 clusters while the smaller (y=50) results in 3 clusters.

DBSCAN CLUSTERING

Hierarchical clustering work for finding the convex clusters. In other words, they are suitable only for compact and well-separated clusters. Moreover, they are also severely affected by the presence of noise and outliers in the data.

- Density-based spatial clustering of applications with noise (DBSCAN) is an alternative to K-Means and hierarchical clustering. It does not require us to specify the number of clusters, as the clusters are formed by a process of linking neighbours together.
- It avoids outliers and identifies nested clusters within the data. The data is muddled and does not have a major visible nested cluster, yet it has identified 3 clusters.

CONCLUSION

1. We started with a simple binning and quantile-based simple segmentation model first then moved to more complex models.
2. Then we moved to k-means clustering and visualized the results with different numbers of clusters. As we know there is no assurance that k-means will lead to the global best solution. We moved forward and tried Hierarchical Clustering and DBSC clustering as well.
3. We created several useful clusters of customers based on different metrics and methods to categorize customers based on their

behavioural attributes to define their evaluability, loyalty, profitability, etc for the business.

4. Segments depend on how the business plans to use the results, and the level of granularity they want to see in the clusters.
5. Keeping these points in view we clustered the major segments based on our understanding as per different criteria as shown in the summary data frame.

After applying different clustering methods we divided customers into many segments

Binning (4 Clusters)

1. **Lost customer**
2. **Average customers**
3. **Good customer**
4. **Best customer**

Quantile Cut (4 Clusters)

1. **Lost customer**
2. **Average customer**
3. **Good customer**
4. **Best customer**

K-Means (2 Clusters)

1. **Best customer**
2. **Lost customer**

K-Means (4 Clusters)

1. **Best customer**
2. **Recently visited average customers**
3. **Good customers**
4. **Average customers**

K-Means (5 Clusters)

1. **Lost customer**
2. **Recently visited average customers**
3. **Good customers**
4. **Average customers**

5. **Best customers**

Agglomerative (2 Clusters)

1. **Average customers**
2. **Best customers**

Agglomerative (3 Clusters)

1. **Best customers**
2. **Average customers**
3. **Lost customers**

DBSCAN (4 Clusters)

1. **Recently visited average customers**
2. **Lost customers**
3. **Good customers**
4. **Losing good customers**

Reference :

1. Towards data science
2. Geeks for geeks
3. Machine learning mastery
4. Analytic Vidhya
5. Wikipedia