**Online Retail Customer Segmentation**

**Lovejeet Singh**

**Data science trainee,**

**AlmaBetter, Bangalore**

**Abstract:**

Online Retail Customer Segmentation effected by many factors such as Customer frequency. which tells How frequent a particular customer doing a purchase, what amount they paid. How much money they spend during the whole year, which product the buy in bulk or frequent. Etc.

Our experiment and Analysis can help us to understand what could be the feature that can related to each other or comes under a cluster. That can help us to separate the customer by their loyalty level, which helps to manage the customer engagements by giving the offers according to their loyalty level. We are performing data wrangling, feature engineering, data analysis, Anomaly detection, RFM model building, and Clustering with the help of unsupervised machine learning algorithms.

1. **Problem Statement**

In this project, our task is to identify major customer segments on a transnational data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail. The company mainly sells unique all-occasion gifts. Many customers of the company are wholesalers.

The main objective is to build a clustering model, which could help them in clustering/separating the customer by their loyalty level. This would in turn help them to create customer segment and provide a detailed information about a particular customer.

**The company provides us dataset. The dataset holds distinctive features.**

* Online Retail.xlsx – holds information about different features along with each customerID.

**Following are the features insides in our dataset:**

* **InvoiceNo:** Invoice number. Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation.
* **StockCode:** Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product.
* **Description:** Product (item) name. Nominal.
* **Quantity:** The quantities of each product (item) per transaction. Numeric.
* **InvoiceDate:** Invoice Date and time. Numeric, the day and time when each transaction was generated.
* **UnitPrice:** Unit price. Numeric, Product price per unit in sterling.
* **CustomerID:** Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer.
* **Country:** Country name. Nominal, the name of the country where each customer resides

**Following are the library we will use in our analysis and model building: -**

* Pandas: - Pandas is for solving data wrangling and exploratory.
* NumPy: - NumPy is for numerical problem solving
* Matplotlib: - Matplotlib is for Data Visualization
* Seaborn: - Seaborn is for Data Visualization
* Scikit Learn: - Scikit learn for Machine learning

1. **Steps involved:**

* **Exploratory Data Analysis**

After loading the datasets, we performed this method by comparing our different variable. This process helped us figuring out various aspects, distribution and relationships among each variable. It gave us a better idea of which feature behaves in which manner compared to another variable. The EDA is a very important step while building a machine learning model.

* **Null values Treatment**

In Our dataset there are two features CustomerID and Description has null/zero values, in which CustomerID has 135080 observations has value as zero while in Description only has 1454 observation which contains zero value. For both features CustomerID and Description we decided to simple delete those observation which has value zero. Because we could not fill the customerID and Description with other or random value.

* **Anomaly Detection**

Isolation Forest is an algorithm to detect outliers that returns the anomaly score of each sample using the Isolation Forest algorithm which is based on the fact that anomalies are data points that are few and different. Isolation Forest is a tree-based model. In these trees, partitions are created by first randomly selecting a feature and then selecting a random split value between the minimum and maximum value of the selected feature. we perform univariate and multivariate anomaly detection and We found 38 anomalies and we decided to delete those anomalies.

* **Feature Engineering**

We can make a new feature by our Datetime Feature which is InvoiceDate to extract Day, Month, Year, Hour and Minute. We also create a new feature called TotalAmount by multiplying Unitprice and Quantity. After extract Day, Month, Year, Hour and Minute we can make another new feature called Day\_time\_type, which tells us the three label Morning, Afternoon and Evening according to the hourly time frame.

* **RFM model Building**

RFM is a method used to analyze customer value. RFM stands for Recency, Frequency, and Monetary. Where RECENCY stands for, how recently did the customer visit our website or how recently did a customer purchase? And FREQUENCY stands for, how often do they visit or how often do they purchase? And MONETARY stands for, how much revenue we get from their visit or how much do they spend when they purchase? The.

* **RFM Analysis**

RFM Analysis is a marketing framework that is used to understand, and analyses customer behaviour based on the above three factors recency, Frequency, and Monetary. RFM Analysis will help the businesses to segment their customer base into different homogenous groups so that they can engage with each group with different targeted marketing strategies.

* **Standardization of features**

Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it. To scale our data, we use StandardScaler which is available in preprocessing class present in scikit learn library. This step is more important when we use distance-based algorithm like K-Means and Hierarchical clustering.

* **Fitting different models**

For modelling we tried various Clustering algorithms like:

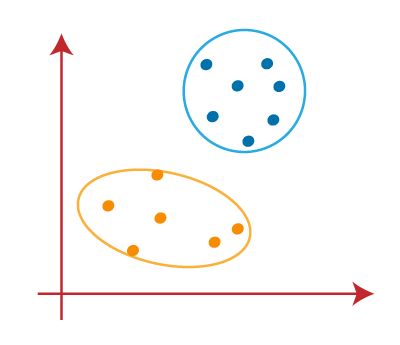
* 1. K-Means Clustering
  2. Hierarchical Clustering
  3. DBSCAN
* **Finding optimal number of clusters**

For finding optimal number of cluster following are the methods we used.

* 1. Silhouette Score
  2. Elbow Method
  3. Dendrogram

1. **Algorithms:**
2. **K-Means Clustering**

The K-means clustering algorithm computes centroids and repeats until the optimal centroid is found. It is presumptively known how many clusters there are. It is also known as the flat clustering algorithm. The number of clusters found from data by the method is denoted by the letter ‘K’ in K-means.



The following stages will help us understand how the K-Means clustering technique works-

* **Step 1:** First, we need to provide the number of clusters, K, that need to be generated by this algorithm.
* **Step 2:** Next, choose K data points at random and assign each to a cluster. Briefly, categorize the data based on the number of data points.
* **Step 3:** The cluster centroids will now be computed.
* **Step 4*:*** Iterate the steps below until we find the ideal centroid, which is the assigning of data points to clusters that do not vary.

1. **Hierarchical Clustering**

Hierarchical clustering*,*also known as hierarchical clusteranalysis*,* is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters*,*where each cluster is distinct from each other cluster, and the objects within each cluster are broadly like each other.

The main output of Hierarchical Clustering is a [dendrogram](https://www.displayr.com/what-is-dendrogram/)*,*which shows the hierarchical relationship between the clusters

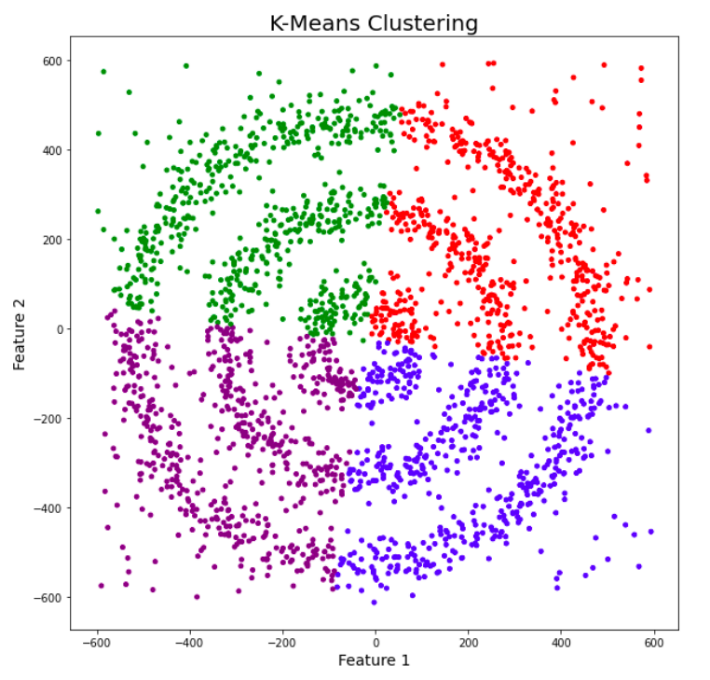
Chart, scatter chart

Description automatically generated

the distance between two clusters has been computed based on the length of the straight line drawn from one cluster to another. This is commonly referred to as the Euclidean distance. Many other distance metrics have been developed.

1. **DBSCAN**

DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise. It groups ‘densely grouped’ data points into a single cluster. It can identify clusters in large spatial datasets by looking at the local density of the data points. The most exciting feature of DBSCAN clustering is that it is robust to outliers. It also does not require the number of clusters to be told beforehand, unlike K-Means, where we have to specify the number of centroids.



DBSCAN requires only two parameters: epsilon and minPoints. Epsilon is the radius of the circle to be created around each data point to check the density and minPoints is the minimum number of data points required inside that circle for that data point to be classified as a Core point.

1. **Finding Optimal number of clusters:**

Model can be evaluated by various metrics such as:

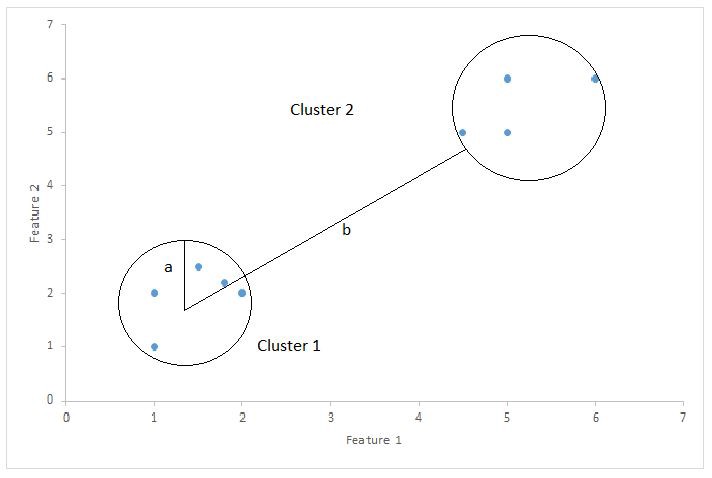
* 1. **Silhouette Score**

Silhouette Coefficient or silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.

1: Means clusters are well apart from each other and clearly distinguished.

0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.

-1: Means clusters are assigned in the wrong way.



Silhouette Score = (b-a) / max(a,b)

where

a= average intra-cluster distance i.e., the average distance between each point within a cluster.

b= average inter-cluster distance i.e., the average distance between all clusters.

* 1. **Elbow method**

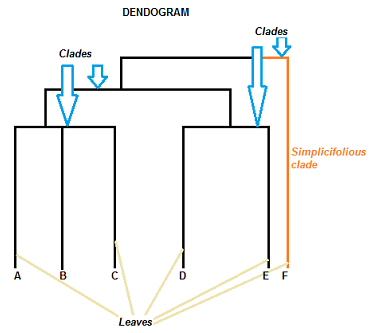
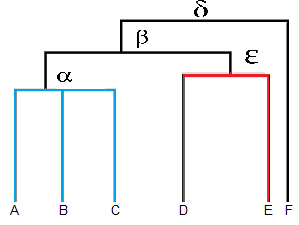
Chart, line chart

Description automatically generated

The elbows method is used to determine the optimal number of clusters in k-means clustering. The elbow method plots the value of the cost function produced by different values of *k*. As you know, if *k* increases, average distortion will decrease, each cluster will have fewer constituent instances, and the instances will be closer to their respective centroids. However, the improvements in average distortion will decline as *k* increases. The value of *k* at which improvement in distortion declines the most is called the elbow, at which we should stop dividing the data into further clusters.

* 1. **Dendrogram**

A dendrogram is a type of [tree diagram](https://www.statisticshowto.com/how-to-use-a-probability-tree-for-probability-questions/) showing hierarchical clustering — relationships between similar sets of data. They are frequently used in biology to show clustering between genes or samples, but they can represent any type of grouped data.

1. **Conclusion:**

That is, it! We reached the end of our exercise. Starting with loading the data so far, we have done EDA, null values treatment, Feature Engineering, anomaly detection, RFM model building, RFM model analysis and then Implementation. In all these models our optimal number of clusters are 2. And the silhouette score is 0.47 on (frequency and monetary feature of RFM model). So, the performance of our cluster based unsupervised machine learning model is good So, we can deploy this model for solve business problem.

Images Source: Google , Towardsdatascience, Greeksforgeeks