ST. XAVIER'S COLLEGE (AUTONOMOUS), KOLKATA

DEPARTMENT OF STATISTICS

30, Mother Teresa Sarani
Kolkata, 700016

CUSTOMER SEGMENTATION USING RFM MODEL: An Application of the K-Means++ Clustering Algorithm

Name : **AMRITA BHATTACHERJEE**
Roll : **403 (STSA)**
Semester : **6**
Session : **2019 - 2022**
Supervisor : **Dr. Ayan Chandra**

DECLARATION :

I affirm that I have identified all my sources and that no part of my dissertation paper uses unacknowledged materials

Signature : Amrita Bhattacherjee

Abstract

Market analysis and targeted differentiated strategies are some of the most important aspects of e-commerce business models. With the advent of advanced storage mechanisms on clouds, we now have a rapidly increasing customer database. As such, applications of data mining are becoming increasingly relevant in today’s world. Customer Segmentation is one such application where a customer’s actions, consumption behaviour and preferences are studied, and they are then categorised into relevant classes, such that each class has some similarities which can be leveraged to implement strategies specifically targeted for them. This process involves an elaborate pipeline of machine learning and data analysis techniques which include data cleaning, data pruning, exploratory data analysis, feature engineering and clustering. The idea is to obtain specific clusters or categories, derived from a customer’s behaviour, such that customers within the same category have certain similarities in their pattern of consumption while customers in different categories differ in their pattern of consumption. This project studies a sales data from the fiscal year 2019-2020. We first clean and prune the data.

We then generate the features that we require from the data at hand, and finally use these engineered features to implement the k-means++ clustering algorithm. The clusters obtained are interpreted and relevant business insights are obtained.

Contents

1. Introduction

Market segment analysis is a crucial aspect of any business venture. Apart from studies and information on budget and product/service details, understanding the market plays an important role in deciding the fate of a business. Analysing target customers has been practiced since the 19th century, but the formal study of marketing emerged much later in the early 1900s. Today, these studies are guiding forces for business owners all around the world. They not only reveal patterns and information on past transactions, but also give insights on how the business evolved over time and what is a possible trend for the future. Business owners can formulate strategies based on these revelations in order to maximise profit and make their business a successful venture.

The studies relating to market analysis largely depend on data obtained from the customers, in the form of transaction details, purchase behaviour and customer feedback. Since this is a social study, data analysis plays an extremely important part here. In early days, studies and resources were not advanced enough to be able to handle such high dimensional, high volume data. During the last few decades, however, tools and technological advancements have enabled us to gather, store, handle and analyse the high amount of data influx that we are witnessing at present.

This paper utilises one such dataset, obtained from a leading jewellery brand in India. This is a customer transaction data, containing information on all transactions in the company during the fiscal year 2019 – 2020. This information includes date, amount, and product/service detail of each transaction. We extract relevant information from this data to perform Customer Segmentation – an unsupervised clustering technique that allows us to discern patterns, if any, in customer behaviour and thus treat each customer according to their ‘value’ to the company. This ‘value’ is obtained by certain aspects of a customer’s transaction behaviour – which is explained further in Section 4.1. By ‘treating each customer accordingly’, we imply the practice of formulating targeted strategies to retain valuable customers and to convert potential customers into high-value ones.

In this project, we implement the k-means++ clustering algorithm to obtain clusters in the data. We then interpret the clusters in terms of customer value. The motivation is to provide statistical insights to market strategists, so that they can use these clusters to formulate new targeted strategies. Section 2 contains a review of past relevant research works that dealt with this field. Section 3 contains a short description of the data that we are working with. Section 4 explains the entire process in detail, consisting of the steps taken in data cleaning, the new features that we engineered from available data and the k-means++ clustering model. Section 5 finally contains analysis of the clusters obtained. We have discussed some limitations of the current setup and further scope of improvement with conclusion in Sections 6 and 7 respectively.

2. Literature Review

Customer Segmentation has been rigorously studied in Marketing for years. Obtaining underlying patterns from customer behaviour helps business owners in making several decisions in advertising, product development and many other aspects.

In 2018, Christy et al. [1] performed Customer Segmentation on the transactional data of an enterprise based of the RFM (Recency, Frequency, Monetary Value) ranking of each customer. They implemented both the k-means as well as the fuzzy c-means algorithms to obtain suitable clusters. The authors also proposed a novel method of initialising the k-means algorithm by using quantiles – an approach that showed promising results. Another recent implementation of the RFM Model is presented by Shen (2021) [2]. Shen utilised a real-world database from an online transaction platform to systematically present a guideline of performing customer segmentation given a raw data. The author not only implemented clustering techniques to obtain customer segments, but also proceeded to analyse the purchased products using association rules mining techniques. If the data has sufficient information on the purchased products, this paper could be a promising guideline for a well-rounded analysis.

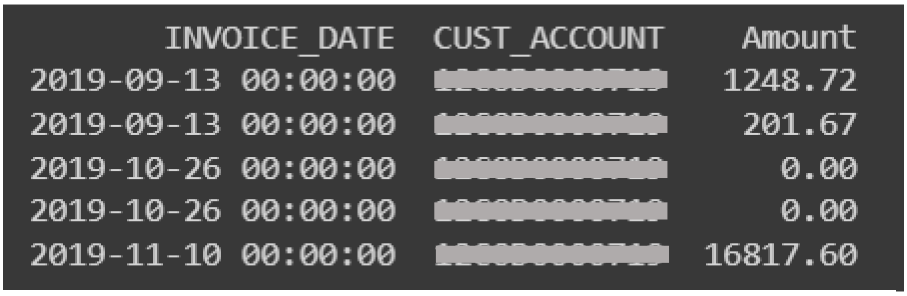
Abidar et al. [3] also provides a simple implementation of the RFM Model with k-means clustering algorithm to obtain clusters from a UK based retail company over a period of 8 months. This paper provides a comprehensive analysis of the final clusters obtained from a very fundamental perspective, which helps in discerning the effects of customer segmentation on the direct decision-making procedure by a marketing strategist. On the other hand, Namvar et al. [4] proposed a two-stage clustering procedure, first using k-means clustering based on RFM score, followed by another phase of clustering within each cluster. The authors also created customer profiles based on customer lifetime value (LTV).

3. Data Description

The data at hand is a sales transaction data. It contains the date of each transaction; the Customer ID and the amount of money transacted each time. The date is in the YYYY-MM-DD format.

The Customer ID is a unique identification number that is assigned to each customer at their first transaction with the company. Any further transactions by that same customer are billed to the Customer ID assigned originally. Therefore, in the data, we get multiple rows against each Customer ID for every customer who has purchased or availed services from the company more than once.

The amount of money is recorded in Indian Rupees. For every transaction, this amount is either positive or zero. The positive value indicates the amount of money that the customer spent that day in their purchases, whereas zero indicates that during that transaction, no money was spent by the customer. This could be attributed to free of cost services, like repairs during the warranty period. These datapoints (containing zero monetary transaction) are important because they contribute to the 'frequency' aspect of the data – that is, they contribute to information on how frequently that customer availed this company's services – which adds to the customer's value to the company.



```
       INVOICE_DATE  CUST_ACCOUNT     Amount
2019-09-13 00:00:00  [redacted]      1248.72
2019-09-13 00:00:00  [redacted]       201.67
2019-10-26 00:00:00  [redacted]         0.00
2019-10-26 00:00:00  [redacted]         0.00
2019-11-10 00:00:00  [redacted]     16817.60
```

Figure 1: First 5 entries of the dataset

Figure 1 shows the first 5 entries of the data. The column 'INVOICE_DATE' gives us the date of transaction, 'CUST_ACCOUNT' gives us the Customer ID. The Customer IDs are struck out to retain anonymity of the data. ‘Amount’ gives us the total amount of money spent at that transaction.

There are 5000 unique Customer IDs in the data, but the number of rows exceed 5000 since each customer has multiple transactions throughout the year 2019-2020. We now clean and prune the data to remove redundant or missing value datapoints, followed by a method of feature engineering, where we utilise information on last date of transaction, frequency of transactions and money transacted, to obtain a 3-dimensional feature vector against each unique Customer ID. Therefore, at the end of the feature engineering step explained in Section 4.1, we obtain 5000 datapoints each having 3 components. We plot these 3-dimensional points on the 3-dimensional Euclidean Space, which is then ready for clustering.

4. Model

4.1 Feature Engineering

We have 5000 unique Customer IDs. For each Customer ID, we want to calculate the following values :

1. **Recency (R)** – The number of days that has passed between the last transaction date of the customer and the last date of transaction available in the data.
2. **Frequency (F)** – The number of transactions made by the customer during the year 2019-2020.
3. **Monetary Value (M)** – The total amount of money spent by the customer at this company throughout the economic year 2019-2020.

These three values together form the RFM Score of a customer. A combination of low R, high F and high M indicates a very high value customer, whereas a combination of high R, low F and low M indicates that the customer is not very valuable to the company. Any combination lying in between could mean a potential high value customer who could be converted into a high value one, with targeted business strategies.

We calculate these values using the following methods :

Recency (R)

As defined above, Recency is the number of days that has passed between the date of transaction of the customer and the last date of transaction available in the entire data. We obtain all dates of transaction for the i^{th} customer, $i = 1\ to\ 5000$. From this list of dates, we extract the last date of transaction and store it in an array. We repeat this for 5000 customers, thus obtaining a list of dates, such that the list has 5000 components – one for each customer. We then find the most recent date in the entire dataset, say D_0. Finally, we calculate the difference (in days) between D_0 and D_i, where D_i is the last date of transaction for the i^{th} customer ($i = 1\ to\ 5000$).

Frequency (F)

Frequency is the number of transactions made by the customer during the year 2019-2020. For this, we obtain the list of all dates of transaction for the i^{th} customer ($i = 1\ to\ 5000$) and then calculate the number of unique dates in this list. Running through all 5000 customers, we obtain an array of frequency values, such that the length of the array is 5000.

Monetary Value (M)

We run a loop through the unique Customer ID's and extract the values under the column 'Amount' for the i^{th} customer ($i = 1\ to\ 5000$). We then calculate the sum of these values to obtain the total amount transacted by the i^{th} customer. Finally, this value is stored in the feature vector 'Monetary Value', which is also of length 5000.

The three feature vectors are now ready – Recency, Frequency and Monetary Value. We now create 3-dimensional data points for the i^{th} customer by taking the i^{th} observation in each of the feature vectors. Finally, our dataset contains 5000 rows, each row containing 3 components.

4.2 Feature Scaling

To obtain best results, we scale this dataset. To do this, from each column, we subtract the mean of observations for that column and divide the resultant values by the standard deviation of those observations :

$$x'_{ij} = \frac{x_{ij} - \overline{x_\iota}}{\sigma_i} \qquad \ldots(1)$$

where,

x_{ij} : j^{th} observation of the i^{th} column, $j = 1\ to\ 5000, i = 1\ to\ 3$

$\overline{x_\iota}$: mean of observations of the i^{th} column = $\frac{1}{5000}\sum_{j=1}^{5000} x_{ij}$ for $i = 1\ to\ 3$

σ_i : standard deviation of observations of the i^{th} column = $\sqrt{\frac{1}{5000}\sum_{j=1}^{5000}\left(x_{ij} - \overline{x_\iota}\right)^2}$ for $i = 1\ to\ 3$

We perform this method independently for each of the three columns. The final dataset is now ready for clustering.

4.3 The K-Means++ Clustering Algorithm

The k-means clustering algorithm is one of the simplest and earliest clustering techniques in the field of machine learning. It is most effective because of how fast it runs, and because of its strong performance in finding clusters that are more or less convex-shaped. Over the 1950s and 1960s, several k-means-type algorithms were introduced by researches from fields as varied as Mathematics, Botany and Electronics. However, the term ‘k-means’ was first coined by James MacQueen in 1967 in his article ‘Some Methods for Classification and Analysis of Multivariate Observations’ [5] at the University of California. The version of k-means that is traditionally used is described in this section.

Simply put, the k-means algorithm tries to find k number of representative centers or prototypes from a given set of multivariate data, such that each prototype is used to identify a cluster and any given datapoint is assigned to one of the k prototypes obtained from the algorithm. We will refer to these prototypes as ‘cluster centers’ for future reference. For now, we will assume that the value of k is known to us. An efficient and simple method of choosing this value of k is later discussed in Section 4.4. The algorithm follows the steps mentioned below to reach its solution –

Step 1 : Randomly select k points from the dataset and treat them as initial cluster centers.

Step 2 : Iterate through the entire dataset and assign each datapoint to its nearest cluster center. The nearest center is decided by its Euclidean distance.

Step 3 : Recompute the cluster centers by calculating the mean of all observations assigned to that particular cluster.

Step 4 : Repeat Steps 2 and 3 iteratively until cluster assignments stabilise, that is, they do not change anymore.

When new datapoints are given, they are assigned to the nearest cluster center. Figure 2 shows a visual representation of how the k-means algorithm works. For this illustration, we have used a synthetic toy data.

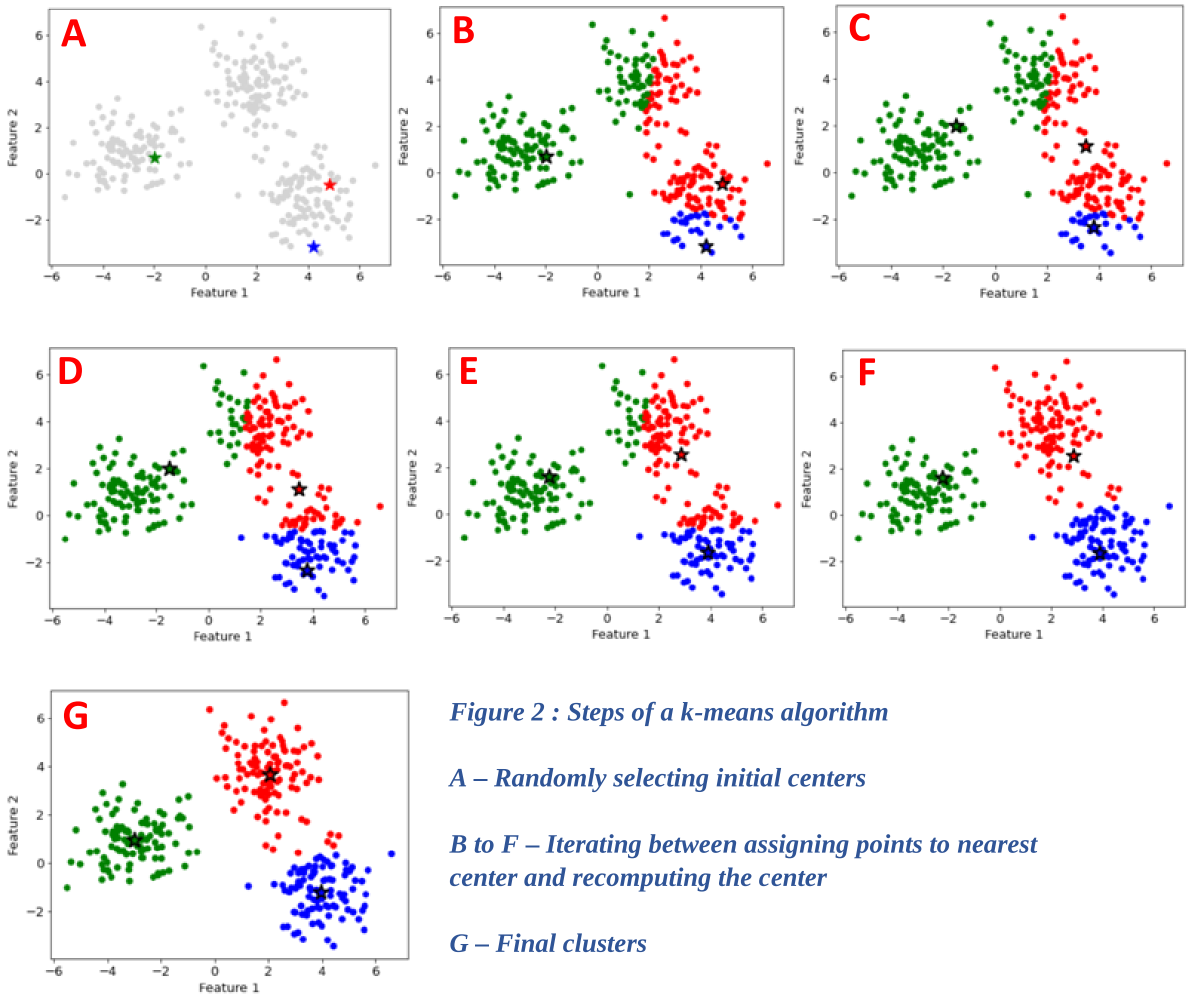


Figure 2 : Steps of a k-means algorithm

A – Randomly selecting initial centers

B to F – Iterating between assigning points to nearest center and recomputing the center

G – Final clusters

Now, let us understand this algorithm mathematically.

Suppose we have a dataset with n observations $\{x_1, x_2, x_3, \dots, x_n\}$ where each x_i is a p-dimensional point on the Euclidean Space. We want to partition this dataset into clusters such that the sum of squared distances of each point from its nearest center is minimum. Let us assume that we need k clusters, and that k is known to us. Now, to understand the algorithm, let us introduce some notations –

$\{\boldsymbol{\mu_1}, \boldsymbol{\mu_2}, \boldsymbol{\mu_3}, \dots, \boldsymbol{\mu_k}\}$: k cluster centers, each having p components. These centers are initially randomly selected from the dataset

$\boldsymbol{I_{ij}}$: an indicator variable $\in \{0, 1\}$ such that $I_{ij} = 1$ if the i^{th} datapoint is assigned to the j^{th} cluster and $I_{ij} = 0$ otherwise, , $i = 1\ to\ n$ and $j = 1\ to\ k$

We now define our objective function as follows –

$$\boldsymbol{J = \sum_{i=1}^{n} \sum_{j=1}^{k} I_{ij} \left\| x_i - \mu_j \right\|^2} \qquad \ldots(2)$$

which is sum of squares of the distances of each datapoint from its assigned cluster center. Our aim is to find such $\{\mu_j\}_{j=1\ to\ k}$ and $\{I_{ij}\}_{\substack{i=1\ to\ n \\ j=1\ to\ k}}$ such that the value of J is minimised. We do this by first randomly selecting $\{\mu_j\}_{j=1\ to\ k}$. Then we iteratively repeat two steps –

- Minimise J with respect to $\{I_{ij}\}_{\substack{i=1\ to\ n \\ j=1\ to\ k}}$, keeping $\{\mu_j\}_{j=1\ to\ k}$ fixed. This is done by assigning 0s and 1s in the following way –

$$I_{ij} = \begin{cases} 1, \ if\ j = arg\ min_{m=1\ to\ k}\ (\|x_i - \mu_m\|^2) \\ 0, \qquad\qquad\qquad\qquad otherwise \end{cases} \qquad \ldots(3)$$

- Minimise J with respect to $\{\mu_j\}_{j=1\ to\ k}$ keeping $\{I_{ij}\}_{\substack{i=1\ to\ n \\ j=1\ to\ k}}$ fixed. Since J is quadratic in μ_j, we can minimise J by simply by equating $\frac{\partial J}{\partial \mu_j}$ to zero and solving for μ_j. By doing this, we obtain –

$$\mu_j = \frac{\sum_{i=1}^{n} I_{ij} x_i}{\sum_{i=1}^{n} I_{ij}} \qquad \ldots(4)$$

Note that at each iteration, μ_j is actually the mean of all datapoints assigned to the j^{th} cluster. This is where this algorithm gets its name ‘k-means’ from. The above two steps run iteratively until the set of values $\{I_{ij}\}_{\substack{i=1\ to\ n \\ j=1\ to\ k}}$ become constant, or until a maximum number of iterations is reached. MacQueen [5] has derived in detail the properties of this algorithm, showing that J is bound to be minimised at every iteration, thus confirming convergence of the algorithm. However, this convergence could get stuck at a local minimum instead of the global minimum.

The initial choice of centers heavily influences the results. This is another limitation of the k-means algorithm.

Further, note that if the k initial centers are chosen very close to each other, not only will there be a higher possibility of being stuck at a bad optimum value, but it will also take longer (a greater number of iterations) to reach the desired result. To avoid this limitation, we implement a modified version of k-means algorithm, called the k-means++ algorithm. This algorithm contains a simple modification called 'effective seed initialisation'. Here, the first center is chosen at random. Then, the following steps are followed to select the remaining k-1 centers –

Step 1 : Calculate distances of every point from this randomly chosen cluster

Step 2 : Assign a probability of choosing each point as the next center, where this probability is proportional to the square of the distance of the point from the nearest center.

Step 3 : Select the next center by sampling it based on the probabilities calculated in Step 2.

This means that points farther away from the nearest center are more likely to be chosen than points close to the previous centers. This lets the initial choice of centers be more spaced out, thus making it more probable that the optimum clusters will be obtained through a lesser number of iterations.

It should be noted that, in the data that we are working with, there is no ground truth. In other words, there are no 'correct assignment' of clusters. We only want to discern any underlying clustering pattern in the data. Therefore, cluster labels do not have any pre-defined significance when considered individually. They are only used to differentiate elements of one cluster from another. They do not give us any information on the type of datapoints in the cluster, but just tell us that elements within the cluster are similar in some way.

4.4 *The Elbow Method*

In general, when we use k-means clustering algorithm on labelled data (supervised machine learning), we know the number of categories that are required to be obtained. However, since, here we are dealing with unlabelled data, we do not know the optimum number of clusters.

In such a scenario, several factors could govern the choice of k. The primary factor is the problem statement and the end objective of the problem. For this particular project, it would have been convenient if the marketing strategist provided us with a number of categories that the company can afford to handle – since the motivation is to devise targeted strategies for each category to maximise customer influx. However, when such external information is not available, a very popular and simple method of obtaining a more or less optimum number of clusters is the 'Elbow Method'.

In this method, we calculate the value of the objective function, J (equation 2) for a certain set of values of k. This range needs to be set after seeing the data at hand. We then run the k-means++ algorithm for all values of k within this range and note down the final value of J that is obtained after convergence each time. We plot these final values of J against the corresponding value of k to obtain the Elbow Plot. From this plot, that value of k is chosen where the curve has a sharp bend such that the magnitude of its slope suddenly falls. This value of k is chosen to be our optimum number of clusters. Figure 3 shows an example of an Elbow Plot on some toy data. In this figure, $k = 3$ seems like a prudent choice.

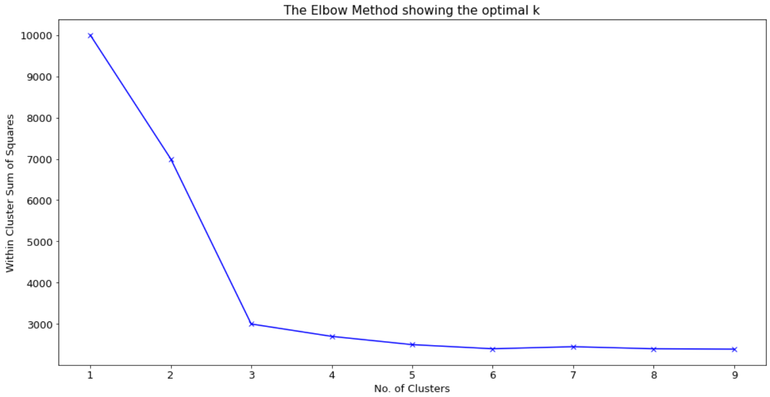


Figure 3 : Illustration of an elbow plot on toy data

5. Cluster output and analysis

We now implement the above algorithm to our cleaned and standardised dataset. To do this, we first implement the elbow method to find an optimum number of clusters. Figure 4 shows the elbow plot obtained for our data –

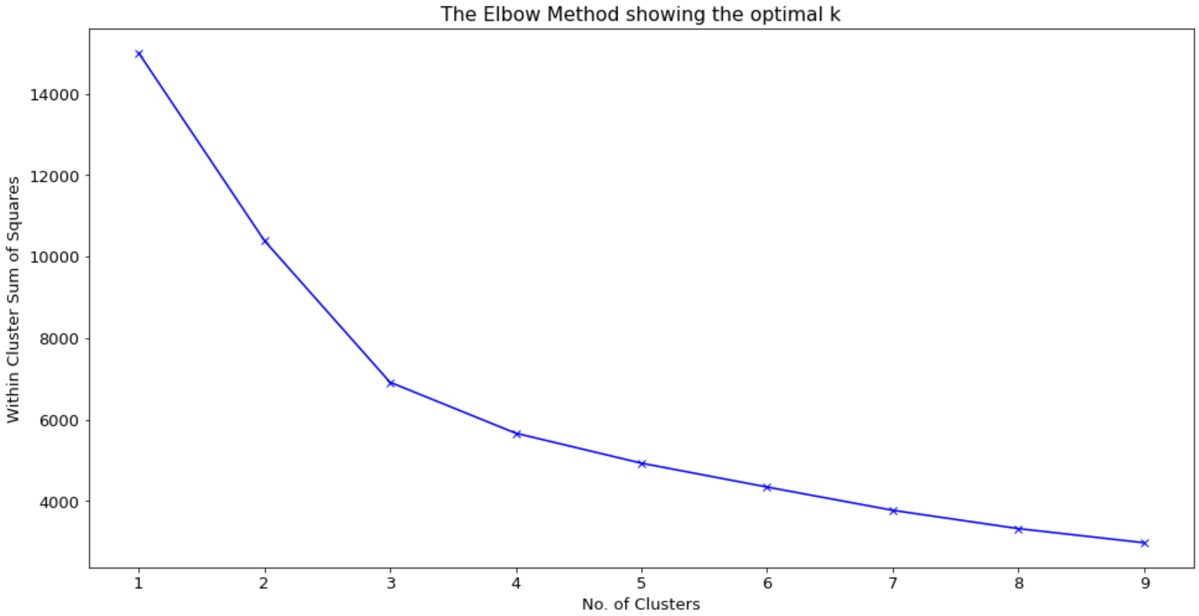


Figure 4 : Elbow Plot of the Customer Transaction Data

From the above figure, it can be seen that a prudent choice of k could be 3, 4 or 5. We take the mean and thus select $k = 4$ for our clustering algorithm.

With the value of k now known, we feed our data to the k-means++ algorithm with parameter $k = 4$. For this problem, we utilise the standard implementation of k-means++ algorithm from Scikit-learn library in Python. For application-oriented readers, we inform that the parameter specifications are : ***n_clusters* = *4*** and ***random_state* = *0***. The clusters thus obtained are shown in Figure 5 –

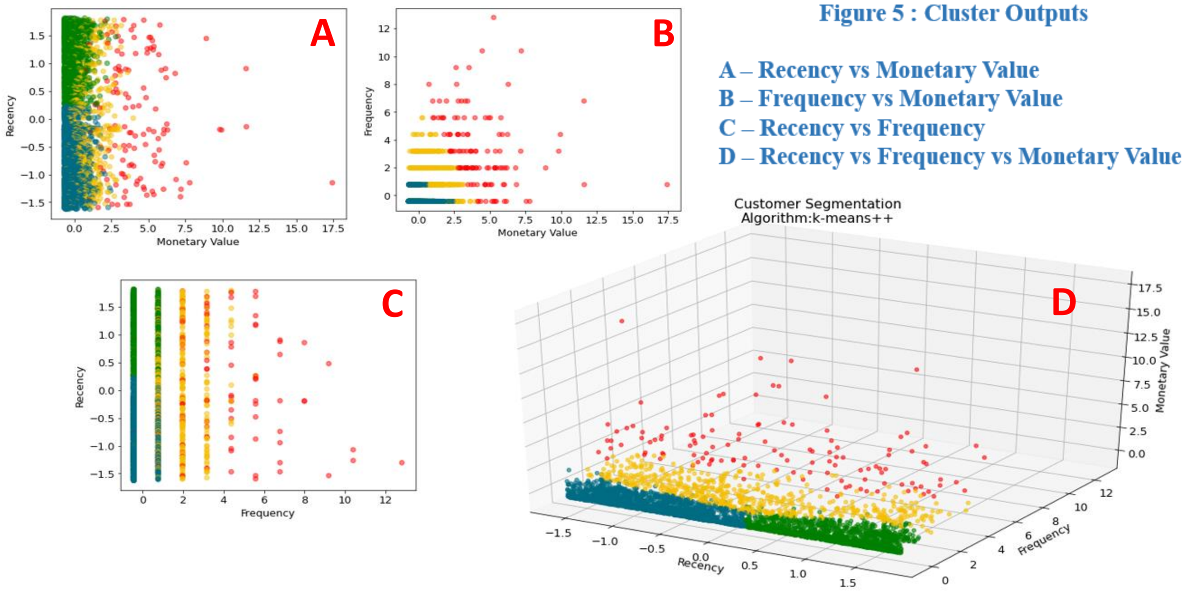


Figure 5 : Cluster Outputs

A – Recency vs Monetary Value
B – Frequency vs Monetary Value
C – Recency vs Frequency
D – Recency vs Frequency vs Monetary Value

The 4 clusters are distinguished by 4 different colours in the plots. For further reference, we denote the Blue, Green, Yellow and Red clusters by 1, 2, 3 and 4 respectively.

1. Clusters 1 and 2 correspond to **low Monetary Value** and **low Frequency** customers. There is also a logical explanation to both of these features existing in low value together – that is, if a customer does not purchase often, then their monetary transaction is bound to be low. Only exceptions are one-time customers who purchased heavy orders. These exceptions fall in Cluster 4, as indicated by the lower-right points in Figure 5B.

2. **Recency** does not seem to act as a major factor in distinguishing Clusters 1 and 2 from 3 and 4. However, Clusters 1 and 2 are themselves different in terms of Recency alone. This is further verified by the figures 5A, 5B and 5C, where figure 5A clearly indicates a rough margin between Clusters 1 and 2, and this differentiable margin vanishes in Figure 5B, when Recency is not plotted. In Figure 5B, Clusters 1 and 2 simply overlap each other.

3. In terms of customer value, Cluster 1 falls above Cluster 2. Cluster 2 contains customers who are not frequent, not recent and not heavy spenders. Cluster 1 contains customers who are not frequent, and their few transactions have not been very heavy in terms of monetary

value. However, **they are recent**, indicating that regular targeted ads and other marketing strategies might consolidate their loyalty.

4. Cluster 3 contains low to medium frequency customers with comparatively higher amounts of transactions than 1 and 2, but lower than 4. This cluster could be treated as **potential** high-value customers, i.e., customers who could become highly loyal with the correct strategies. This cluster contains customers whose monetary transactions and frequency are higher than Clusters 1 and 2, with Recency varying between low to high.

5. Cluster 4 only contains customers with **very high monetary transactions**. The most valuable customers within this cluster are the ones lying on the bottom-right of figures 5A and 5C, and top-right of figure 5B. The other points within this cluster represent less frequent, less recent customers with heavy transaction amount. A prospective way of treating these points is by capitalising on the fact that they spent a heavy amount on their transaction (however less recent or less frequent), thereby indicating that they must have some loyalty, for example, they might trust the brand value. Therefore, targeted differentiated strategies can turn them into frequent customers.

Therefore, from the clusters obtained, we can see how the customers are automatically segmented according to their value to the company. With these segments, a marketing strategist can accurately identify which customers should be targeted and how.

6. Drawbacks of the Model

The problem at hand, being an unsupervised setup, lacks ground truth. The primary limitation in this scenario is the fact that we cannot compare cluster performance based on its ‘accuracy’. In other words, with unsupervised clustering techniques, there are no right answers. It is merely an optimization problem, where we are aiming towards a good convergence point. That being said, there have been several alternative methodologies to gain an insight on algorithm performance of an unsupervised clustering problem. These include Silhouette Scores and other Gap-Statistic methods that are beyond the scope of this paper. The authors look forward to implementing these methods in future as the project progresses.

Another limitation of the k-means++ clustering algorithm in particular is its inefficiency in clustering non-convex shapes. As a result, even though Customer Segmentation problems often use k-means++ algorithm because of its simplicity and speed, other clustering options such as hierarchical clustering and DBSCAN could perform better in discerning other patterns in the data.

Next comes the problem of initial bias. As discussed very briefly under Section 4.3, this algorithm largely depends on the initial choice of centers. As a result, a bad choice of initial centers could potentially trap the algorithm at a bad local optimum, thus converging to a cluster output that is not ideal at all. Although there have been several methods devised to minimise this risk, there is no concrete solution to eradicate this problem.

Despite the above limitations, the k-means++ algorithm with RFM Model has a widespread application in the field of marketing, and continues to be one of the most popular approaches to solve market segment analysis problems.

7. Conclusion

This paper presents an effective utilisation of Machine Learning algorithms in a business setup. The use of statistical learning is so versatile that it finds its application in various disciplines. We have explained how a simple transaction data can reveal so much information about a customer's behavioural pattern, when the right tools are implemented. With more high-dimensional data, containing information on the purchased products, demography and other aspects, one can study product preferences within each segment. This will help in making decisions that relate to promotional offers and targeted advertisements. Therefore, these studies can reveal information that helps maximising company revenue.

An interesting avenue of further research could be the study of how the targeted strategies helped in increasing customer influx and retaining high-value customers. This could be done by collecting transaction data over multiple economic years and analysing how the customer segments changed over the years, as new strategies were employed. If highly recent, highly frequent and heavy-spending customers increase in number, then we can infer that the targeted strategies have been successful in their purpose.

8. References

[1] Christy, A.J., et al. RFM ranking – An effective approach to customer segmentation. Journal of King Saud University – Computer and Information Sciences (2018), https://doi.org/10.1016/j.jksuci.2018.09.004

[2] Boyu Shen. 2021. E-commerce Customer Segmentation via Unsupervised Machine Learning. In The 2nd International Conference on Computing and Data Science (CONFCDS 2021), January 28–30, 2021, Stanford, CA, USA. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3448734.3450775

[3] Lahcen Abidar, Dounia Zaidouni, and Abdeslam Ennouaary. 2020. Customer Segmentation With Machine Learning: New Strategy For Targeted Actions. In 13th International Conference on Intelligent Systems: Theories and Applications (SITA'20), September 23–24, 2020, Rabat, Morocco. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3419604.3419794

[4] Namvar, Morteza & Gholamian, Mohammad & Khakabimamaghani, Sahand. (2010). A Two Phase Clustering Method for Intelligent Customer Segmentation. Intelligent Systems, Modelling and Simulation, International Conference on. 215-219. 10.1109/ISMS.2010.48.

[5] MacQueen, James. "Some methods for classification and analysis of multivariate observations." *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. No. 14. 1967.

Acknowledgement

I acknowledge my sincerest thanks and gratitude to my supervisor, Dr. Ayan Chandra for his constant guidance and help while completing this project. I would also like to thank the authorities of St. Xavier’s College (Autonomous), Kolkata for giving me this opportunity to explore new and exciting fields of research and implement them into this paper.

I acknowledge all the researchers who have paved the way for future students like me to continue and explore the field of Statistics and Statistical Learning. I have cited every reference that have guided and helped me during this project.