2023

LEVERAGING RFM+BALANCE CUSTOMER SEGMENTATION FOR EFFECTIVE MARKETING IN BANKING

Project Report

SUPERVISOR : DR.MURUGANANTHAN VELAYUTHAM

MEET THE TEAM

MOHAMED KHAIRY MOHAMED ABDELRAOUF
LIM CHUN CHEAK
WONG ZHENG QIE
SAI NAING LYNN OO

CRYSTAL SYSTEM

Table of Contents

Abstract

The aim of this project is to implement various clustering algorithms to segment customers based on their purchasing behavior. By understanding the different groups of customers, businesses can develop targeted marketing strategies and improve customer satisfaction. In this study, we used K-means, Hierarchical Clustering (Agglomerative Clustering), and DBSCAN on a dataset containing customer transactions. We evaluated the clustering results using metrics such as Silhouette Coefficient and Calinski-Harabasz Index. Our findings provide insights into the characteristics of different customer segments and enable data-driven decision-making.

Acknowledgements

We would like to express our sincere gratitude to our supervisors and colleagues for their guidance, support, and valuable suggestions throughout this project. We also appreciate the open-source community for providing the necessary tools and resources that enabled us to carry out this research.

Section (1): Introduction

Customer segmentation is the process of dividing customers into groups based on common characteristics, such as demographics, preferences, or purchasing behavior. The primary goal of customer segmentation is to enable businesses to tailor their marketing and sales strategies to better suit the needs of different customer groups. This not only improves customer satisfaction but also increases overall profitability. Clustering algorithms are widely used in customer segmentation tasks, as they can identify hidden patterns in large datasets and group similar data points together.

In this project, we have applied three clustering algorithms - K-means, Hierarchical Clustering (Agglomerative Clustering), and DBSCAN - to a dataset containing customer transaction data. We then evaluated the performance of these algorithms using evaluation metrics such as Silhouette Coefficient and Calinski-Harabasz Index. This report presents a detailed analysis of our methodology, findings, and conclusions.

1.1 Problem Statement

Banks and financial institutions often struggle with effective customer segmentation, which leads to imprecise marketing and suboptimal customer satisfaction. The challenge lies in identifying meaningful segments based on customer transactional data that can guide targeted marketing strategies.

1.2 Aim/Objective/Deliverables

The primary aim of this project is to implement various clustering algorithms, specifically K-means, Hierarchical Clustering (Agglomerative Clustering), and DBSCAN, to segment customers based on their purchasing behavior.

The objective is to understand the different groups of customers to develop targeted marketing strategies and improve customer satisfaction.

The deliverables include a detailed analysis of the customer segments, insights on their characteristics, and recommendations for data-driven decision-making.

1.3 Scope

The scope of the project encompasses the application of clustering algorithms on a dataset containing over 1.05 million customer transactions for a bank in India. The project involves data acquisition and preprocessing, algorithm implementation, model evaluation, and data visualization and analysis. The project does not include predictive analytics techniques such as customer lifetime value prediction or churn analysis.

1.4 Similar System

The project seems to be unique in its approach to customer segmentation by combining RFM (Recency, Frequency, Monetary) model with account balance data for a bank in India. However, customer segmentation using clustering algorithms is a common practice in the field of data science and machine learning. Various studies have been conducted that apply similar techniques for customer segmentation. The following table presents the reference journals and articles for segmentation of customer for this project.

No.	Citation (Author, Title, Journal, Year)	Brief Summary	Models or Techniques used	Findings/Limitations
1.	Aliyev, M., Ahmadov, E., Gadirli, H., Mammadova, A., & Alasgarov, E. (2020). *Segmenting Bank Customers via RFM Model and Unsupervised Machine Learning*. https://arxiv.org/abs/2008.08662	The article used the real customer data of Azerbaijan Bank to retain their customers and develop marketing strategies for profitable segments and offer personalized services.	- RFM model (R: Recency, F: Frequency, M: Monetary) - K-Means Clustering - Agglomerative Clustering - DBSCAN Clustering - K-Modes	The use of DBSCAN to identify the outliers and segment the clusters again with K-Means Clustering provided the best result. Agglomerative Clustering Algorithms did not perform well due to high computing requirements and thus discarded this model.
2.	Firdaus, U., & Nugeraha Utama, D. (2021). *Development of Bank's Customer Segmentation Model based on RFM+B Approach. ICIC International 2021*, *12*(1), 17–26. https://doi.org/10.24507/icicelb.12.01.17	The authors incorporated the new feature which is the balance of customers with RFM model to improve the analysis effectively for decision makers to develop the promotion strategies for banking services.	- RFM + B model (B: Balance) - K-Means Clustering	The use of RFM + B model provided the clustering efficiency by 77.58% compared to just implementing pure RFM model. The developed model could be used for crafting marketing plans more effectively.
3.	Raiter, O. (2021). *Segmentation of Bank Consumers for Artificial Intelligence Marketing*. International Journal of Contemporary Financial Issues, *1*(1), 39–54. https://hcommons.org/deposits/item/hc:43351/	The researcher aims to segment different clusters of card holder customers into General, Targets, Savers, and Big spenders to devise marketing campaigns and policies to boost customer's satisfaction and the service quality of the bank.	- K-Means Clustering - Elbow method - Silhouette Score	The clustering analysis is useful for customer segmentation especially for financial institutions. They also concluded that Elbow method and Silhouettes score could assist in obtaining the optimal numbers of clusters with the similar results.
4.	Zakrzewska, D., & Murlewski, J. (2005, September 1). *Clustering algorithms for bank customer segmentation*. IEEE Xplore. https://doi.org/10.1109/ISDA.2005.33	The scholars conducted a comparative study of K-Means and DBSCAN clustering efficiency and scalability using high dimensionality data which also contains noises in banking data.	- K-Means Clustering - DBSCAN Clustering - Two-phased clustering process using based on K-Means and Agglomerative Hierarchical Clustering	The study deduced that three different techniques have their own advantages and disadvantages. K-Means performed well on high multidimension data. However, it relies strongly on the number of k-clusters. Two-phase clustering has a reliable efficiency when dealing with outliers and noises, but not suitable for high dimensional dataset. The DBSCAN algorithms could works well with noises and outliers, only if the appropriate parameters are selected.
5.	Rahmah, N., & Sitanggang, I. S. (2016). *Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra*. IOP Conference Series: Earth and Environmental Science, 31, 012012. https://doi.org/10.1088/1755-1315/31/1/012012	The researchers experimented to determine the optimal values for DBSCAN Clustering algorithm's parameters such as 'eps' and 'MinPts'.	- DBSCAN Clustering - K-Nearest Neighbor	The research provided the method for deciding the parameter values of DBSCAN Clustering algorithm with the help of K-Nearest Neighbor algorithms by plotting the average distance.

1.5 Programming Languages Used and Why

We chose Python as the programming language for this project due to its versatility, simplicity, and extensive library support. Python has become the preferred language for data science and machine learning tasks because of its easy-to-read syntax, powerful libraries such as pandas, NumPy, scikit-learn, and seaborn, and its strong community support. These libraries and the flexibility of Python allow for efficient data manipulation, analysis, and visualization, making it an ideal choice for this project.

1.6 Techniques Used in the Project

In this project, we used three clustering algorithms to segment customers based on their purchasing behavior:

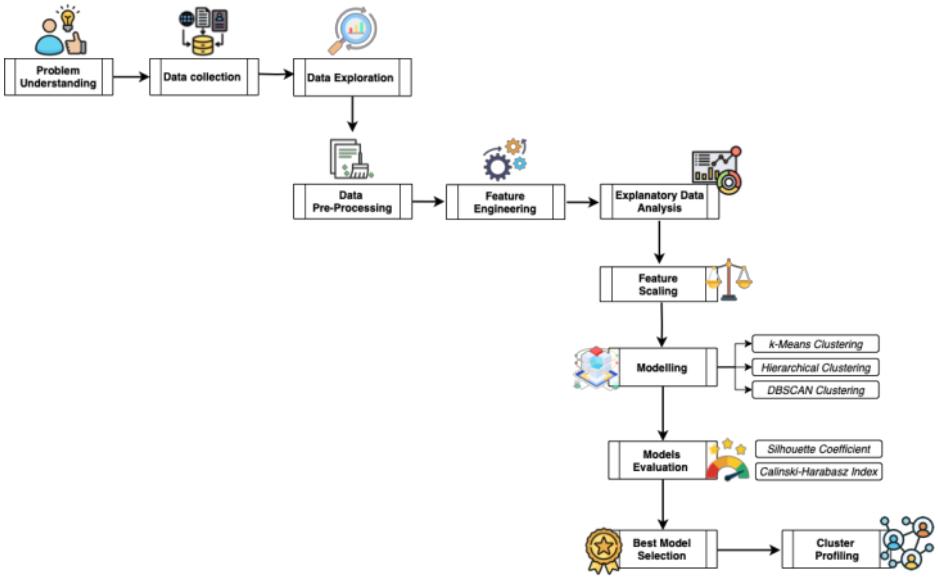
1. *K-means:* A centroid-based algorithm that partitions the data into K clusters based on the mean distance between data points and their assigned centroids. The algorithm iteratively adjusts the centroids until convergence is achieved.

2. *Hierarchical Clustering (Agglomerative Clustering):* A linkage-based algorithm that builds a tree of clusters by successively merging the closest pairs of clusters. This algorithm results in a dendrogram representation, which can be cut at different levels to obtain the desired number of clusters.

3. *DBSCAN (Density-Based Spatial Clustering of Applications with Noise):* A density-based algorithm that groups data points based on their density. It identifies clusters as dense regions separated by areas of lower point density, allowing for the identification of arbitrary-shaped clusters and noise points.

4. *Evaluation metrics:* The clustering results were evaluated using metrics such as Silhouette Coefficient and Calinski-Harabasz Index.

Methodology Flowchart



1.7 Project Elements

The project can be divided into several key components:

1. Problem understanding: The project aims to segment the customer base to develop targeted marketing strategies for different clusters of customers for offering personalized products and boost the customer satisfaction and retention.

2. Data acquisition and preprocessing: We acquired a dataset containing customer transactions and performed necessary preprocessing steps, such as removing irrelevant columns, handling missing values, and scaling the data.

3. Clustering algorithm implementation: We implemented K-means, Hierarchical Clustering, and DBSCAN algorithms on the preprocessed dataset.

4. Model evaluation: We calculated evaluation metrics, such as Silhouette Coefficient and Calinski-Harabasz Index, to assess the quality of the clustering results.

5. Visualization and analysis: We created visualizations to explore the centroids, cluster distributions, and other patterns in the data.

Section (2): Implementation

2.1 Data Preparation

The dataset consists of 1.05 million transactions by over 800,000 customers for a bank in India. The data consists of information such as customer location, customer gender, account balance at the time of transaction, customer age, and transaction details. The dataset is downloaded from Kaggle via the link : https://www.kaggle.com/datasets/shivamb/bank-customer-segmentation

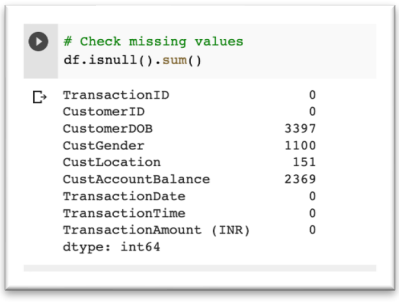
Original Metadata (before preprocessing)

Variable	Type	Description	Examples
TransactionID	ID	Unique transaction ID	T1, T2, T3…
CustomerID	ID	Unique customer ID	C5841053, C2142763…
CustomerDOB	Date	Date of birth of customer	10/1/94, 4/4/57…
CustGender	Nominal	Customer Gender	M,F
CustLocation	Nominal	Customer Location	MUMBAI, NEW DELHI…
CustAccountBalance	Numerical	Account balance at the time of transaction	17819.05, 2270.69…
TransactionDate	Date	Transaction date	2/8/16, 2/12/16…
TransactionTime	Numerical	Transaction time	143207, 141858…
TransactionAmount	Numerical	Amount of transaction	25, 27999, 459…

2.2 Data Pre-processing

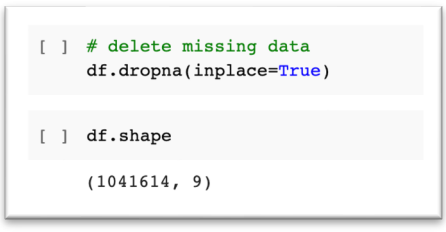
2.2.1 Remove NA values

All NA values are less than 1% of the dataset, hence they are removed. The resulting dataset has 1041614 transactions after removing missing values.



```
# Check missing values
df.isnull().sum()
```

```
TransactionID              0
CustomerID                 0
CustomerDOB             3397
CustGender              1100
CustLocation             151
CustAccountBalance      2369
TransactionDate            0
TransactionTime            0
TransactionAmount (INR)    0
dtype: int64
```



```
# delete missing data
df.dropna(inplace=True)
```

```
df.shape
```

```
(1041614, 9)
```

2.2.2 Remove customers that are underaged or with invalid birth year

Customers more than 10 years old, which is the legal age in India to have a bank account will be considered in the data, any customers with DOB>2006 will be removed. There are also exceptionally high number of customers with DOB as 1800-01-01, this is probably the default value. A customer being

born in 1800 would mean that he/she will be 216 years old in 2016 (the transaction period), which doesn't make sense, hence these data will also be deleted. The resulting dataset becomes 910040 rows.

```
df['CustomerDOB'].value_counts()

1800-01-01    56292
1989-01-01      809
1990-01-01      784
1991-06-08      698
1991-01-01      665
              ...
2051-02-12        1
2052-03-20        1
2047-09-26        1
2041-04-10        1
2044-10-24        1
Name: CustomerDOB, Length: 17233, dtype: int64
```

```
[ ] # Remove data for DOB = 1800-01-01 and DOB >2006
    df = df.loc[~(df['CustomerDOB'] == '1800-01-01')]
    df = df.loc[~(df['CustomerDOB'].dt.year >= 2006)]

[ ] df.shape

    (901140, 10)
```

2.3 Feature engineering

In this section, four new features such as customer age, recency, frequency and monetary are created from the existing features as below.

2.3.1 Customer Age

Original dataset has customer's date of birth information; hence it can be used to compute a customer's age. Customer's age is calculated using the difference in years between transaction date and CustomerDOB.

```
#Calculate customer age:
df['CustomerAge'] =df['TransactionDate'].dt.year - df['CustomerDOB'].dt.year
```

2.3.2 Recency

Recency is how recent the customer last made a purchase, measured in days. It is calculated using max transaction date (latest purchase date) minus the min transaction date (first purchase date). The resulting dataset is then grouped by CustomerID, transforming the whole dataset from transaction level to customer level.

```
RFM_df['Recency'] = (RFM_df['TransactionDate2'] - RFM_df['TransactionDate1']).dt.days
RFM_df.head()
```

```
RFM_df.Recency.value_counts()
```

```
0      669064
1        2677
31       2033
30       1994
61       1726
        ...
287        15
58         12
265        10
266         7
206         5
Name: Recency, Length: 252, dtype: int64
```

It is found that there are recency value of zero is the highest and the zero mean very recent. Therefore, they are converted into 1 using below code.

```
# Convert zero recency to 1
RFM_df['Recency'] = RFM_df['Recency'].apply(lambda x: 1 if x==0 else x)
RFM_df.Recency.value_counts()
```

```
1      671741
31       2033
30       1994
61       1726
5        1472
        ...
287        15
58         12
265        10
266         7
206         5
Name: Recency, Length: 251, dtype: int64
```

2.3.3 Frequency

Frequency is how frequent the customers purchase. It is calculated by the number of transactions made by the customers in the period.

```
# Construct raw data for clustering from transactional data
RFM_df = df1.groupby("CustomerID").agg({
                                        "TransactionID" : "count",
                                        "CustGender" : "first",
                                        "CustLocation":"first",
                                        "CustAccountBalance"  : "last",
                                        "TransactionAmount (INR)" : "sum",
                                        "CustAge" : "median",
                                        "TransactionDate2":"max",
                                        "TransactionDate1":"min",
                                        "TransactionDate":"median"
                                    })
# Reset index of new dataframe
RFM_df = RFM_df.reset_index()
RFM_df.head()

# Rename Columns to Frequency and Monetary
RFM_df.rename(columns={"TransactionID":"Frequency"},inplace=True)
RFM_df.rename(columns={"TransactionAmount (INR)":"Monetary"},inplace=True)
```

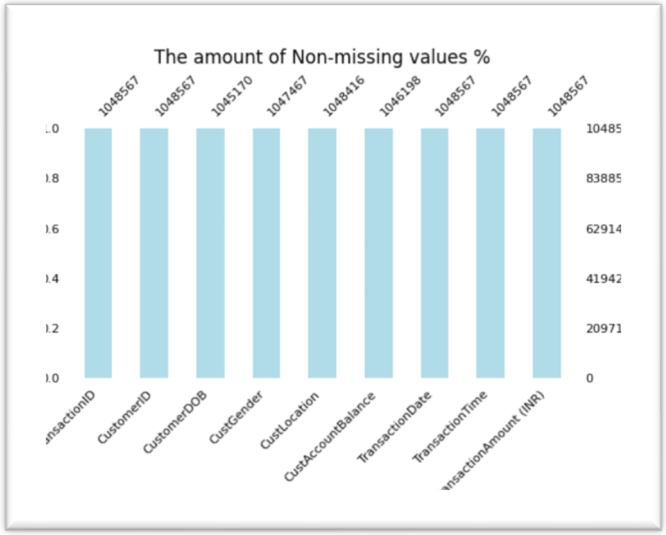
2.3.4 Monetary

Monetary is how much the customer spent in their purchases. It is calculated by the sum of all transaction amount of a customer. The transaction amount of customer is changed into monetary feature after the data has been aggregated as seen in above codes.

2.4 Explanatory Data Analysis (EDA)

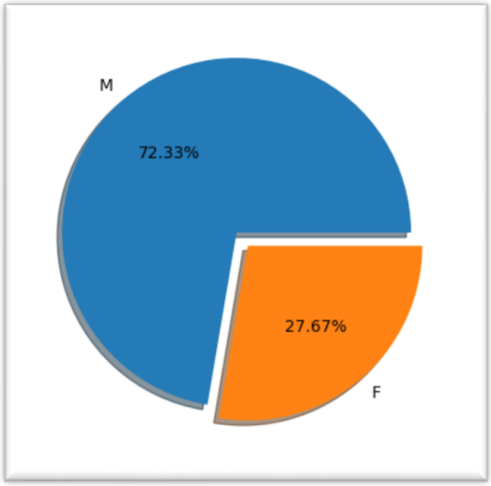
In this section, we will discuss the results of the various graphs and visualizations that were generated during the project. Understanding the insights derived from these visualizations will help us make informed decisions and provide targeted solutions for each customer segment.

2.4.1 Non-missing Values



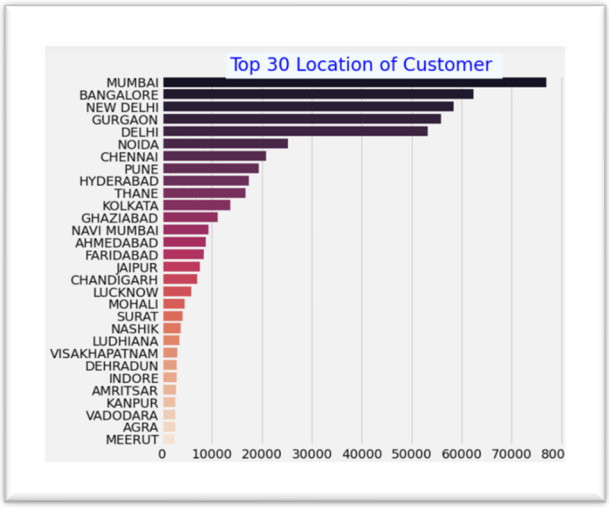
The dataset was found to be complete with no missing values. This indicates that the data is reliable and can be used to draw meaningful insights.

2.4.2 Gender Distribution



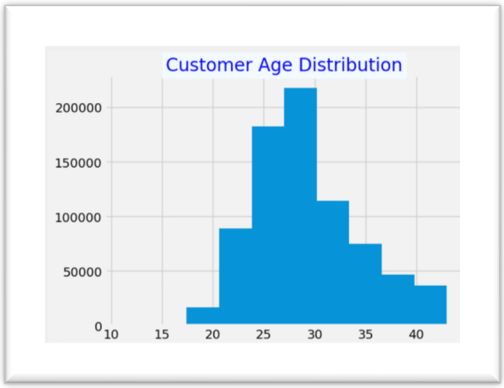
The gender distribution among customers shows that 72.33% are male, while 27.67% are female. This information can be used to tailor marketing strategies and offers to suit the preferences of the majority of the customer base while ensuring that the needs of the minority are not overlooked.

2.4.3 Top and Lowest Locations of Customers



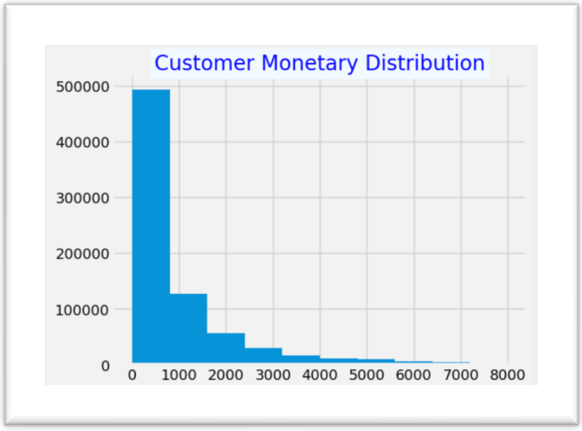
The top three locations with the highest number of customers are Mumbai, Bangalore, and New Delhi. In contrast, the locations with the lowest number of customers are Vadodara, Agra, and Meerut. This information can help businesses identify areas with potential for growth and focus marketing efforts on regions where the customer base is strong.

2.4.4 Customer Age Distribution



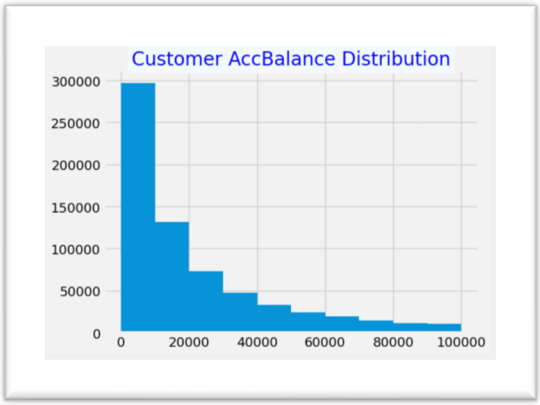
The age distribution of customers reveals that there are no customers below the age of 18 or above the age of 50. The majority of customers fall within the 25-29 age group. This information can be used to develop products and services that cater to the specific needs and preferences of this age group, thereby increasing customer satisfaction and loyalty.

2.4.5 Customer Monetary Distribution



The monetary distribution shows that most customers have an income above 400,000 and close to 500,000, while a smaller segment of customers have an income below 100,000. This data can be used to segment customers based on their purchasing power and develop targeted offers and promotions that cater to their financial capabilities.

2.4.6 Customer Account Balance Distribution



The majority of customers have an account balance between 250,000 and 300,000, while a smaller proportion of customers have an account balance below 50,000. This information can be used to assess the overall financial health of the customer base and identify opportunities to offer tailored financial products and services to enhance customer engagement and loyalty.

2.4.7 Bivariate Analysis – CustAccBalance, Frequency, Monetary, CustAge

1. Older customers generally spend more amount (monetary) and has higher account balance.
2. Customers with lower account balance spend more amount (monetary) as compared to customer with high account balance.
3. Customers with lower account balance also spend more frequently as compared to customers with high account balance.
4. Customers who spend frequently tend to be older in age.

By analyzing the results of these graphs, businesses can gain valuable insights into their customer base and make data-driven decisions to improve their products, services, and marketing strategies. Ultimately, this will lead to increased customer satisfaction, loyalty, and revenue growth.

Key Findings from EDA

- Older customers generally spend more amount (monetary) and has higher account balance.
- Customers with lower account balance spend more amount (monetary) as compared to customer with high account balance.
- Customers with lower account balance also spend more frequently as compared to customers with high account balance.

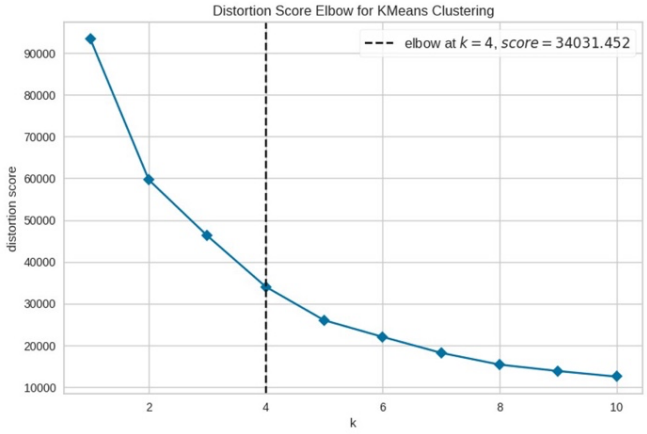
2.5 Building Clustering Models

Before constructing the models, the optimal number of clusters suitable of the dataset is determined using Elbow Plot as in below.

```
# Idenfying no. of clusters using Elbow method
import random
from yellowbrick.cluster import KElbowVisualizer
from sklearn.cluster import KMeans

k_means_viz = KMeans(init = 'k-means++', random_state = 42)
visualizer = KElbowVisualizer(k_means_viz, k=(1,11), timings=False)

# Generate Elbow plot
visualizer.fit(df_sub_scaled)
visualizer.show()
```



As it can see from the plot, the appropriate number of clusters is four and thus, we implemented with 4 clusters in building the models.

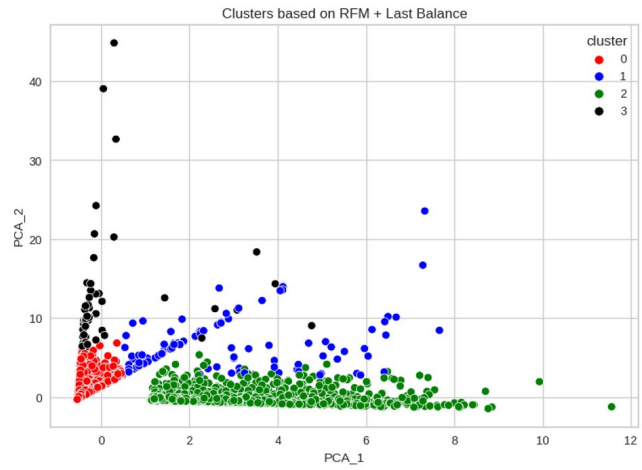
2.5.1 K-Means Clustering

```
# Create the k-means model with 4 clusters
k_means = KMeans(n_clusters = 4, init = 'k-means++', random_state = 42)

# Fit the model to the scaled data
k_means.fit(df_sub_scaled)

# Get the cluster labels for each observation
labels_km = k_means.predict(df_sub_scaled)

# Add a column for the cluster labels
cluster_labels = pd.Series(labels_km, index=df_sub_scaled.index)
```

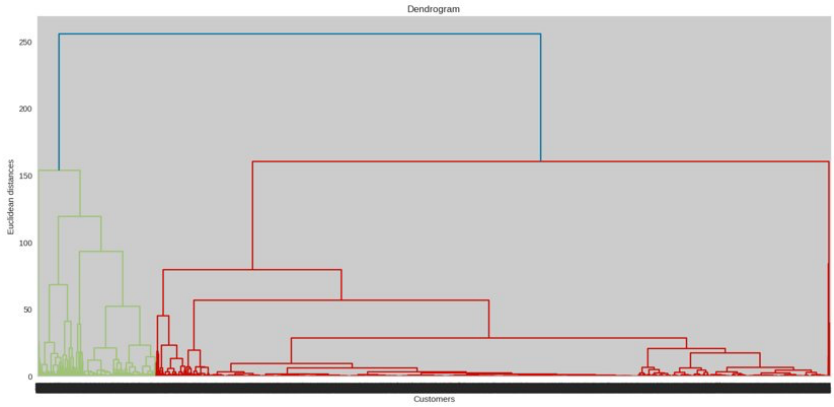


The k-Means clustering is built with four clusters using the codes above and the plot clearly shows the clusters which has been formed. The plot was generated using two dimensional PCA since it is not possible to plot in four dimensions. The clusters seems appropriate.

2.5.2 Hierarchical Clustering

```
# Plot Dendrogram
import scipy.cluster.hierarchy as sch
plt.figure(1, figsize = (16 ,8))
dendrogram = sch.dendrogram(sch.linkage(df_sub_scaled, method  = "ward"))

plt.title('Dendrogram')
plt.xlabel('Customers')
plt.ylabel('Euclidean distances')
plt.show()
```

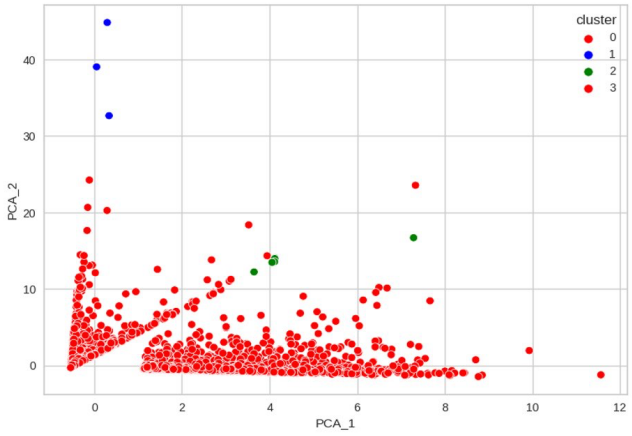


Firstly, the dendrogram is generated before building the hierarchical clustering model so as to see the hierarchical nature of the data. It also helps determine the number of clusters to be formed.

```
# Build Agglomerative clustering
from sklearn.cluster import AgglomerativeClustering
HClustering = AgglomerativeClustering(n_clusters = 4,
                                      affinity = 'euclidean', linkage ='average')

HClustering = HClustering.fit_predict(df_sub_scaled)

df_sub_hc = df_sub.copy()
df_sub_hc['Label'] = HClustering
df_sub_hc.head()
```



As per dendrogram and Elbow Plot, it is decided to create four clusters using Agglomerative Clustering. The plot represents the nature of clusters which has been generated by hierarchical clustering algorithm. The result is not satisfying compared to K-Means clustering algorithm.

2.5.3 DBSCAN Clustering

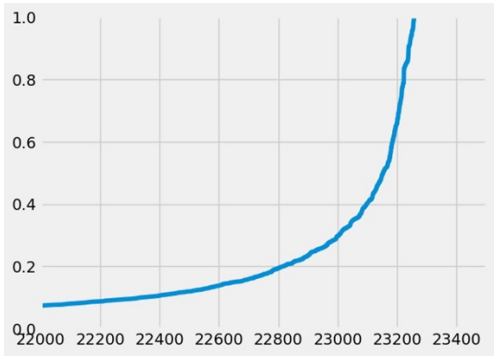
Due to the fact that the DBSCAN clustering is very sensitive to hyperparameters, the optimal hyperparameters are identified using the approach recommended by (Rahmah & Sitanggang, 2016). Hyperparameters in DBSCAN includes min_samples and eps. The min_samples is twice the number of dimensions/features. Therefore, it will be 8 (2 x 4 features). Then using the min_samples size, the k-Nearest Neighbor is implemented to obtain the average distance using Elbow plot which is suitable for optimal values of eps.

```
# Build KNN model to determine the average distance for eps value
from sklearn.neighbors import NearestNeighbors
neighbors = NearestNeighbors(n_neighbors=8)
neighbors_fit = neighbors.fit(df_sub_scaled)
distances, indices = neighbors_fit.kneighbors(df_sub_scaled)
```

```
# Plot average distance to get the optimal value for eps
distances = np.sort(distances, axis=0)
average_distance = np.mean(distances)
plt.plot(distances)

# Zoom in the plot to get optimized eps value
plt.xlim(22000, 23500)
plt.ylim(0, 1)

# Display the plot
plt.show()
```



The optimal values should be selected using the crook of the elbow or the maximum curvature of the curve. The plot shows that the potential eps values ranges approximately from 0.28 to 0.38. Hence, the values are attempted to get the most appropriate value and 0.32 yielded the best.

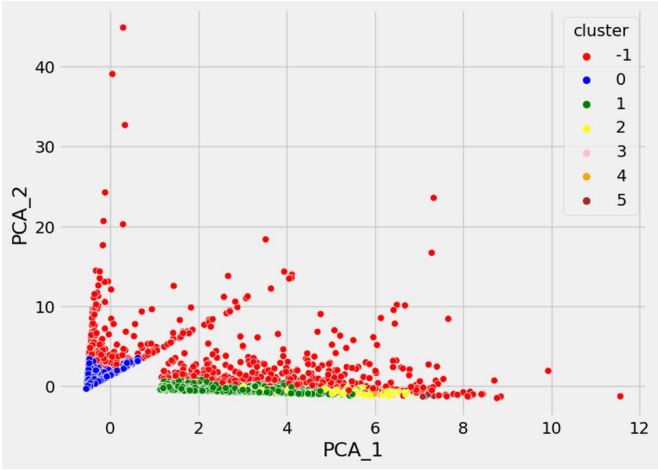
```
# Build DBSCAN Clustering model using optimzed parameters
from sklearn.cluster import DBSCAN

# Create DBSCAN object with specified hyperparameters
# tested with 0.28, 0.30, 0.32, 0.34, 0.36, 0.38 -> 0.32 yields the best
dbscan = DBSCAN(eps=0.32, min_samples=8)

# Fit DBSCAN to the scaled data
dbscan.fit(df_sub_scaled)

# Get the predicted cluster labels for each data point
labels_dbs = dbscan.labels_

df_sub_dbscan = df_sub.copy()
df_sub_dbscan['Label'] = labels_dbs
df_sub_dbscan.head()
```



Subsequently, the DBSCAN is implemented using eps values of 0.32 and min_samples of 8. The plot displays the clusters formed by the DBSCAN. It generated (7) different clusters. However, the proportion and the nature of the clusters is not properly segmented as compared to K-Means clustering.

2.6 Model Evaluation

The evaluation metrics used are Silhouette coefficient which ranges from -1 to 1. The higher values of coefficient represents the better clustering for the model. In additon, the Calinski-Harabasz index is also applied in this project. A higher value of the index indicates that the clusters are well separated and compact. The below code is used to generate the metrics for each model. Here, just an example code for K-Means is provided.

```
# Evaluation metrics for k-Means Clustering
from sklearn.metrics import silhouette_score, calinski_harabasz_score

# Calculate the Silhouette coefficient
silhouette_coef = silhouette_score(df_sub_scaled, labels_km)

# Calculate the Calinski-Harabasz index
calinski_harabasz_index = calinski_harabasz_score(df_sub_scaled, labels_km)

print("Evaluation metrics for k-Means Clustering")
print("==========================================")
print(f"Silhouette coefficient: {silhouette_coef}")
print(f"Calinski-Harabasz index: {calinski_harabasz_index}")
print("==========================================")
```

```
Evaluation metrics for Hierarchical Clustering
==============================================
Silhouette coefficient: 0.933674066576957
Calinski-Harabasz index: 1184.2474110387245
==============================================


Evaluation metrics for k-Means Clustering
==========================================
Silhouette coefficient: 0.7850288848637171
Calinski-Harabasz index: 13532.497950398647
==========================================


Evaluation metrics for DBSCAN Clustering
==========================================
Silhouette Coefficient: 0.7520810080027036
Calinski-Harabasz Index: 3281.314796395641
==========================================
```

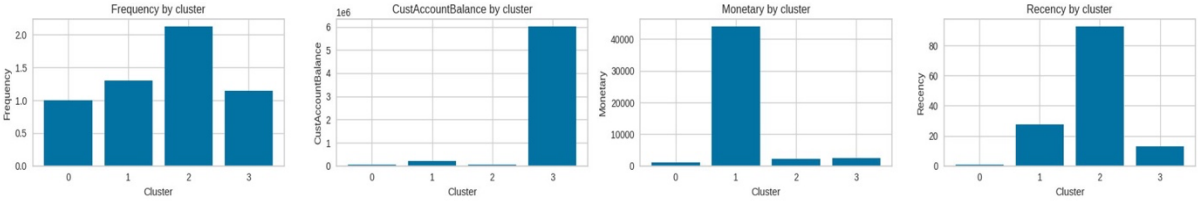
Looking at the generated metrics, the highest score of Silhouette is obtained by hierarchical clustering model. However, it has the lowest index value. As observed from the plot in previous section in hierarchical clustering, the clusters are badly segmented. Thus, it will not be selected. Then, the K-Means and DBSCAN model are compared and found that the K-Means achieved the higher score in coefficient and the largest in Calinski-Harabasz index score. Therefore, K-Means clustering model will be used to segment the customer based for this project.

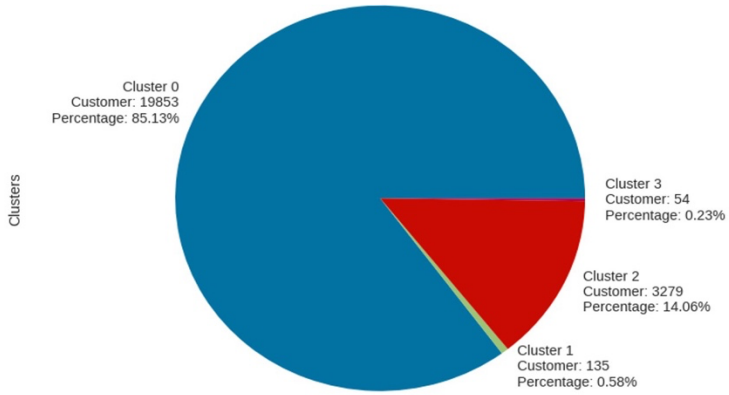
Section (3): Discussion and Recommendation

3.1 Cluster Interpretation and Marketing Strategies

In this section, the clusters obtained from the K-Means clustering model is interpreted using the RFM+Balance and provided the recommended marketing strategies for each clusters for decision-making purposes.

	0	1	2	3
Frequency	1.00	1.30	2.13	1.15
CustAccountBalance	66288.66	216095.86	64702.47	6020118.34
Monetary	1027.25	44028.13	2249.66	2435.41
Recency	1.00	27.80	92.76	13.13





The majority of the customers fall into cluster 0, accounted for 85 % of the whole customer base. The second largest clusters is cluster 2 which comprises 14 %. The cluster 1 is ranked at third with 0.6 %, and the smallest cluster is cluster 3 which just has 0.2 %. The interpretation and recommended marketing strategies for each cluster is delivered as below.

Cluster 0:

Interpretation: This cluster has the lowest values for all RFM variables, indicating customers who have made only one purchase, have a low account balance, and spent very little. These customers are unlikely to make frequent purchases/transactions in the near future, so it might be beneficial to focus on retention strategies that encourage them to remain as customers.

Marketing strategies: New but inactive users. Better mobile and web access such as User Interface (UI) and security aspect. Convenient access and seamless customer services. Partnerships with various merchants to encourage this cluster to utilize the bank service. Newsletter or emails subscription to educate them about differentiated products and services than other banks.

Cluster 1:

Interpretation: This cluster has slightly higher frequency than Cluster 0 and has the highest monetary value among all clusters. Due to this, the average account balance for Cluster 1 is just the 2nd highest (behind Cluster 3). This indicates that they spend or transact the most compared to other clusters.

Marketing strategies: Active and high-spending or transactions users. Enable privileged banking services to the. Better return on investments (ROI) or interest rate (IR) products for them such as unit trust, fixed deposit to encourage savings. Products education such as banassurance for comprehensive financial planning. Offer mobile app solution to track spending.

Cluster 2:

Interpretation: They have the highest frequency and recency among all clusters. However, their account balance is the lowest among all clusters and have the 2nd lowest monetary value spent. They only spend slightly higher than Cluster 0 due to higher frequency and recency. This suggests that they are more likely to be active users but merely average spenders.

Marketing strategies: Active but average income and spendings users. Offer lower loan interest to reward for their loyalty. Promote products such as unit trust, fixed deposit or saving plans as part of money management program. Reward them with higher interest rate after saving certain amounts.

Cluster 3:

Interpretation: RFM variables for Cluster 3 are relatively moderate compared to other clusters. This indicates that they are more likely to be idle users and transact lesser with this bank. Due to this, they have the highest amount of account balance. This suggests that there are more potential businesses and investments that could be tapped for the bank from this cluster.

Marketing strategies: Idle but high-income users. Enable privileged banking services to them. Offer great credit cards promotion with substantial cashbacks or services such as balance transfer to encourage their spendings. Suggest several higher ROI investments to retain their loyalty. Priority customer services and support.

3.2 Further Discussion: Leveraging Insights for Practical Applications

The insights derived from the customer segmentation analysis can be used in various practical forms to drive business growth, enhance customer satisfaction, and optimize marketing efforts. In this section, we will discuss how businesses can utilize these insights effectively and the benefits that can be gained from doing so.

3.2.1 Personalized Marketing Strategies

By understanding the demographics, financial status, and geographic distribution of the customer base, businesses can develop personalized marketing strategies that appeal to specific customer segments. This includes tailoring advertising messages, promotional offers, and product recommendations based on factors such as age, gender, income, and location. Personalized marketing has been shown to increase customer engagement, improve conversion rates, and boost customer loyalty.

3.2.2 Product and Service Development

The insights into customer age and income distribution can be used to create products and services that cater to the specific needs and preferences of the target audience. For example, businesses can develop financial products tailored to customers in the 25-29 age group, with features that cater to their unique financial needs and goals. Similarly, businesses can create customized offers and promotions that suit the financial capabilities of customers with different income levels. This

targeted approach to product development can help businesses differentiate themselves from competitors and foster customer loyalty.

3.2.3 Geographically targeted Expansion

Identifying the top and lowest locations of customers can help businesses strategize their expansion plans. They can focus on strengthening their presence in the top-performing regions while identifying opportunities for growth in underrepresented areas. This geographically targeted approach to expansion can lead to a more efficient allocation of resources and greater market penetration.

3.2.4 Customer Retention and Loyalty Programs

The customer segmentation analysis provides insights into the financial health of the customer base, which can be used to develop customer retention and loyalty programs. By offering tailored financial products, incentives, and rewards to customers based on their account balances and income levels, businesses can encourage repeat business and foster long-term customer relationships. This not only enhances customer satisfaction but also increases the lifetime value of each customer.

3.2.5 Data-driven Decision-making

The insights generated through customer segmentation analysis can also help businesses make more informed decisions in various aspects of their operations. By understanding the composition of their customer base, businesses can allocate resources more efficiently, prioritize product development initiatives, and identify potential risks and opportunities. This data-driven approach to decision-making can lead to better outcomes and increased profitability.

In summary, the insights derived from the customer segmentation analysis offer businesses a wealth of information that can be used to drive growth, enhance customer satisfaction, and optimize marketing efforts. By leveraging these insights in practical forms, businesses can create personalized marketing strategies, develop products and services that cater to the specific needs of their customer base, and make data-driven decisions that lead to better outcomes. Ultimately, these efforts will result in increased customer loyalty, higher customer lifetime value, and long-term business success.

Section (4): Conclusion

In conclusion, this project demonstrated the utility of clustering algorithms in identifying distinct customer segments based on their purchasing behavior. Through the implementation of K-means, Hierarchical Clustering, and DBSCAN algorithms, we gained valuable insights into the characteristics of different customer groups, enabling data-driven decision-making for targeted marketing strategies and improved customer satisfaction.

The evaluation of the clustering results using metrics such as Silhouette Coefficient and Calinski-Harabasz Index allowed us to compare the performance of the clustering algorithms and choose the most appropriate method for our dataset. While each clustering algorithm provided unique insights into the customer segmentation, a combination of these methods could potentially yield even more accurate and detailed customer profiles.

Future research could explore additional clustering algorithms or incorporate other customer attributes, such as demographics or preferences, to further refine the segmentation and enhance our understanding of the different customer groups. Additionally, the implementation of predictive analytics techniques, such as customer lifetime value prediction or churn analysis, could complement the findings of this project and support more effective decision-making for businesses.

References

Aliyev, M., Ahmadov, E., Gadirli, H., Mammadova, A., & Alasgarov, E. (2020). *Segmenting Bank Customers via RFM Model and Unsupervised Machine Learning*. https://arxiv.org/abs/2008.08662

Bennett, R., & Kottasz, R. (2012). Customer segmentation and customer profiling: A comparison of the perceptions of academics and practitioners. *Journal of Marketing Management*, 28(5-6), 558-580. https://www.tandfonline.com/doi/full/10.1080/0267257X.2012.658840

Firdaus, U., & Nugeraha Utama, D. (2021). *Development of Bank's Customer Segmentation Model based on RFM+B Approach*. *ICIC International 2021*, *12*(1), 17–26. https://doi.org/10.24507/icicelb.12.01.17

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35(2), 137-144. https://www.sciencedirect.com/science/article/pii/S0268401214001066

Kumar, V., & Reinartz, W. (2016). *Creating enduring customer value*. Journal of Marketing, 80(6), 36-68. https://journals.sagepub.com/doi/10.1509/jm.15.0414

Li, S., Sun, B., & Wilcox, R. T. (2005). Cross-selling sequentially ordered products: An application to consumer banking services. *Journal of Marketing Research*, 42(2), 233-239. https://journals.sagepub.com/doi/10.1509/jmkr.42.2.233.62203

Lim, K. S., & Siau, K. (2017). Analyzing the effectiveness of customer segmentation using demographic, geographic, and psychographic variables. *Journal of Computer Information Systems*, 57(4), 313-321. https://www.tandfonline.com/doi/full/10.1080/08874417.2016.1183448

Rahmah, N., & Sitanggang, I. S. (2016). *Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra*. IOP Conference Series: Earth and Environmental Science, 31, 012012. https://doi.org/10.1088/1755-1315/31/1/012012

Raiter, O. (2021). *Segmentation of Bank Consumers for Artificial Intelligence Marketing*. International Journal of Contemporary Financial Issues, *1*(1), 39–54. https://hcommons.org/deposits/item/hc:43351/

Smith, A. N., Fischer, E., & Yongjian, C. (2012). How does brand-related user-generated content differ across YouTube, Facebook, and Twitter? *Journal of Interactive Marketing*, 26(2), 102-113. https://www.sciencedirect.com/science/article/pii/S1094996812000095

Tam, J. L., & Kim, W. G. (2019). The effects of customer segmentation on customer satisfaction and service recovery. *International Journal of Hospitality Management*, 77, 141-150. https://www.sciencedirect.com/science/article/pii/S0278431918301997

Wedel, M., & Kannan, P. K. (2016). Marketing analytics for data-rich environments. *Journal of Marketing*, 80(6), 97-121. https://journals.sagepub.com/doi/10.1509/jm.15.0413

Zakrzewska, D., & Murlewski, J. (2005, September 1). *Clustering algorithms for bank customer segmentation*. IEEE Xplore. https://doi.org/10.1109/ISDA.2005.33