การจัดกลุ่มและการวิเคราะห์ตะกร้าตลาดของลูกค้าที่เป็นธุรกิจ (B2B) กรณีศึกษาบริษัทขายอุปกรณ์สำนักงาน

B2B CUSTOMER SEGMENTATION AND MARKET BASKET ANALYSIS: A CASE STUDY OF AN OFFICE SUPPLIER COMPANY

สุทิวัส ถิ่นบูรณะกุล

SUTIWAT TINBURANAKUL

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์

ศูนย์วิเคราะห์ข้อมูลดิจิทัลอัจฉริยะพระจอมเกล้าลาดกระบัง

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2566

KMITL-2023-SC-M-017-071

B2B CUSTOMER SEGMENTATION AND MARKET BASKET ANALYSIS: A CASE STUDY OF AN OFFICE SUPPLIER COMPANY

SUTIWAT TINBURANAKUL

AN INDEPENDENT STUDY SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE
IN DATA SCIENCE AND ANALYTICS
KMITL DIGITAL ANALYTICS AND INTELLIGENCE CENTER SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2023

KMITL-2023-SC-M-017-071

หัวข้อการค้นความอิสระ การจัดกลุ่มและการวิเคราะห์ตะกร้าตลาดของลูกค้าที่เป็นธุรกิจ (B2B) กรณีศึกษาบริษัทขายอุปกรณ์สำนักงาน

ชื่อนักศึกษา นายสุทิวัส ถิ่นบูรณะกุล

รหัสประจำตัว 64605116

ปริญญา วิทยาศาสตรมหาบัณฑิต (วิทยาการข้อมูลและกาวิเคราะห์)
ศูนย์วิเคราะห์ข้อมูลดิจิทัลอัจฉริยะพระจอมเกล้าลาดกระบัง

พ.ศ. 2566

อาจารย์ที่ปรึกษาการค้นคว้าอิสระ ผู้ช่วยศาสตราจารย์ ดร.พรพิมล ชัยวุฒิศักดิ์

บทคัดย่อ

ในสถานการณ์ธุรกิจที่แข่งขันอย่างรุนแรงในปัจจุบัน ธุรกิจระหว่างผู้ค้ากับหน่วยธุรกิจกำลังมองหาวิธีนวัตกรรมในการเข้าใจและบริการลูกค้าให้ดียิ่งขึ้นอยู่เสมอ งานวิจัยนี้มีวัตถุระสงค์เพื่อประเมินการแบ่งกลุ่มลูกค้าในบริษัทจำหน่ายอุปกรณ์สำนักงานรูปแบบธุรกิจระหว่างผู้ค้ากับหน่วยธุรกิจ โดยใช้การจัดกลุ่มข้อมูลด้วยวิธีเคมีน สำหรับแบ่งกลุ่มลูกค้าตามพฤติกรรมในการซื้อสินค้าได้แก่ ข้อมูลของลูกค้าที่มาซื้อสินค้าครั้งล่าสุด ความถี่ของลูกค้าในการซื้อสินค้า มูลค่าการซื้อของลูกค้าและความหลากหลายของหมวดหมู่สินค้าที่ลูกค้าซื้อในแต่ละครั้ง จากนั้นประยุกต์ใช้อัลกอริทึมเอไพรโอไรและอัลกอริทึมเอฟพีโกรธในการวิเคราะห์ตะกร้าสินค้าเพื่อค้นหาชุดสินค้าที่ถูกซื้อด้วยกันบ่อยและกฎสมาชิกสินค้าในการซื้อของในแต่ละกลุ่มลูกค้า โดยเปรียบเทียบผลลัพธ์และประสิทธิภาพในการทำงานระหว่างอัลกอริทึมเอไพรโอไรและอัลกอริทึมเอฟพีโกรธ นอกจากนี้กฎสมาชิกสินค้าที่ถูกสร้างขึ้นโดยอัลกอริทึมเอไพรโอไรและอัลกอริทึมเอฟพีโกรธยังถูกวิเคราะห์ผ่านเกณฑ์เพิ่มเติมโดยวิธีกฎความสัมพันธ์ เพื่อตรวจสอบความน่าเชื่อถือของกฎสมาชิกสินค้าในแต่ละกลุ่มลูกค้า การวิเคราะห์พฤติกรรมของลูกค้าผ่านการแบ่งกลุ่มลูกค้าและการวิเคราะห์ตะกร้าสินค้าไม่เพียงแต่เสริมคุณภาพของการแบ่งกลุ่มลูกค้า แต่ยังเป็นข้อมูลทางธุรกิจที่มีประโยชน์ต่อโอกาสการขายผ่านกลยุทธ์การเสนอขายสินค้าอีกด้วย รวมทั้งการแนะนำให้ลูกค้าซื้อผลิตภัณฑ์ที่เกี่ยวเนื่องกับสินค้าหลักเพิ่มเติม กลยุทธ์การมัดรวมสินค้า และประสิทธิภาพของธุรกิจโดยรวม

Independent Study Title	B2b Customer Segmentation and Market Basket Analysis: A Case Study of an Office Supplier Company
Students	Sutiwat Tinburanakul
Student ID	64605116
Degree	Master of Science (Data Science and Analytics)
	KMITL-Digital Analytics and Intelligence Center
Year	2023
Independent Study Advisor	Asst.Prof.Dr.Pornpimol Chaiwuttisak

Abstract

In today's highly competitive business landscape, B2B companies are constantly seeking innovative ways to better understand and serve their customers. This study explores customer segmentation in a B2B office supply company using K-means clustering to segment customers into distinct groups based on recency, frequency, and monetary Value (RFM) analysis and number of product section variety, we employ the Apriori and FP-Growth algorithms. These Market Basket algorithms are used to identify frequent itemsets and association rules among purchased products. By comparing the results and runtime performance of Apriori and FP-Growth, we aim to determine which algorithm is more efficient for uncovering meaningful business insights within the context of B2B office supplies. Furthermore, association rules generated by these algorithms serve as additional criteria to justify the segmentation results obtained from K-means clustering. This integration of customer behavior analysis through Market Basket algorithms not only enhances the quality of customer segmentation but also provides valuable insights into cross-selling opportunities, product bundling strategies, and overall business performance.

Keywords: Business-to-Business (B2B), Customer Segmentation, K-means Clustering, Market Basket Analysis, Apriori Algorithm, FP-Growth Algorithm, Association Rules

Acknowledgments

I would like to extend my heartfelt gratitude to all those who have played a pivotal role in the successful completion of this study, including the invaluable contribution of Lyreco Thailand in providing the essential research data.

First and foremost, I wish to express my deep appreciation to my thesis advisor, Assoc.Prof.Dr.Pornpimol Chaiwuttisak, for their unwavering guidance, expertise and invaluable feedback that have significantly shaped this study.

I am immensely grateful to the faculty and staffs at King Mongkut's Institute of Technology Ladkrabang for fostering an environment conducive to academic growth and for providing the necessary resources for this endeavor.

My sincere thanks go out to my family for their unwavering support and encouragement throughout this challenging academic journey. I also want to acknowledge and thank my friends and colleagues for their continued encouragement, discussions, and moral support, which have enriched this research.

Lastly, I extend my special appreciation to Lyreco Thailand for their generous contribution of research data. Without their cooperation and willingness to share their valuable data, this thesis would not have been possible.

This thesis represents a collective effort, and I am profoundly thankful to all of you for being an integral part of this significant milestone in my academic and professional life.

Sutiwat Tinburanakul

TABLE OF CONTENTS

TABLE OF CONTENTS

(Continued)

Chapter **Page**

TABLE OF CONTENTS

(Continued)

Chapter **Page**

TABLE OF CONTENTS

(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

List Of Figures

(Continued)

Figure **Page**

List Of Figures

(Continued)

Figure **Page**

List Of Figures

(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Research Background

In the dynamic realm of business-to-business (B2B) commerce, organizations face a constant pursuit of growth and competitiveness. The modern landscape is characterized by rapid technological advancements, evolving consumer preferences, and intensifying market competition. In this context, the imperative to optimize sales strategies is more pronounced than ever, urging companies to adopt innovative approaches that can effectively propel their B2B sales performance. This study delves into the strategic amalgamation of customer segmentation and marketing basket analysis as potent tools for boosting B2B sales, with a comprehensive case study conducted within the context of an office supply company.

Businesses engaged in B2B transactions operate in a distinct environment, catering to the needs of other businesses rather than individual consumers. The intricacies of B2B sales involve longer sales cycles, larger transaction volumes, and complex decision-making processes. Therefore, an enhanced understanding of the underlying factors influencing B2B purchasing behaviors is critical to optimize sales approaches.

Traditionally, B2B companies have employed generic strategies that cast a wide net, aiming to capture a broad range of potential clients. However, this one-size-fits-all approach is increasingly proving to be inadequate in the face of diverse and discerning B2B customers. As a result, contemporary B2B enterprises are turning to advanced analytical methods to discern nuanced customer preferences, segment the market, and tailor their strategies to different client clusters.

Data mining has gained popularity especially in the last two decades due to the advancements in computing power and data storage technology, this provided us the

ability to mine large quantity of data. Extracting knowledge and hidden information from data using wide variety of techniques that found its useful applications in various contexts. As an example, data mining is widely use in marketing to identify and analyze customer segments and their behavior so the company can provide better service to each customer segment and adjust promotional campaigns according to their needs and motivations.

Companies nowadays have access to vast amounts of historical sales and customer data but lack in extracting useful insight from that data. Big data is seen as a valuable resource and although the concept of data mining is still new and developing, companies in a variety of industries are relying on it for making strategic decisions. Facts that otherwise may go unnoticed can be now revealed by the techniques that sift through stored data. When applying mining tools and techniques we attempt to find useful relationships, patterns and anomalies that can help management team make better business decisions.

Data mining tools allow us to analyze and ascertain valuable information for business strategies and allow us to get to know our customers better. Managerial insights and experiences alone are no longer the only factor trusted when it comes to decision-making. Data driven decisions can lead to higher company performance.

The main point of interest for companies is to understand dependencies among purchases. Consumers buy various combinations of products on a single shopping trip. It is very important for companies to get to know what their customers are buying. Some products have higher affinity to be sold together and hence the companies can benefit from this affinity if special offers and promotions are offered for these products. Data mining techniques are highly valued for the useful information they provide so that the companies can serve customers better and generate higher profits.

Research have been done in marketing to show that there are demand interdependencies among certain related products. Companies tend to exploit this tendency by adjusting price promotions in a profit-maximizing way. They can also exploit these product associations by incorporating them into promotional strategies. Analyzing purchases in multiple categories allows companies to benefit from promotion and other

marketing activities. Incorporation of product interdependencies into a pricing strategy is an effective way of boosting profits.

For a classic example, Mulhern and Leone (1991) study the impact of price promotions on cake mix and cake frosting. Their main objective is to evaluate the overall profitability of implicit price bundling. Reducing the price of cake mix increase purchases of both cakes mix and frosting and the overall profit improves. The study shows how promotions have positive impact on the sales of a complementary product.

In the recent years, analyzing customer shopping baskets has become quite appealing to trading companies and retailers. Advanced technology and specific algorithms made it possible for companies to gather information and to find insight on their customers on what they frequently buy. The introduction of electronic point of sale increased the use and application of transactional data in market basket analysis. In retail business analyzing such information is highly useful for understanding customer spending behavior. Mining purchasing patterns allows companies to adjust promotions based on the findings so they can cross-sell products. Identifying buying rules is crucial for every successful business. Transactional data is used for mining useful information on co-purchases and adjusting promotion and advertising accordingly.

An example retail use case that combines RFM-analysis and k-means is provided by Hsu and Huang (2020). In their research they want to identify VIP customers. VIP customers are buyers of critical products which are not purchased by the average customer. In their approach, they apply the RFM-analysis on over 600,000 transactions from around 3800 customers. The segmentation is based on the 20%-quantile of RFM-values.

The significance of this study is underscored by the transformative potential that customer segmentation and marketing basket analysis hold for B2B sales optimization. The synergy between these two approaches has the capacity to revolutionize how B2B enterprises engage with their clients, create value, and drive revenue growth. The ultimate objective is to provide insights and practical strategies that equip B2B companies, such as

the office supplier in the case study, with the tools needed to excel in an increasingly competitive marketplace.

The structure of this study is designed to holistically address the objectives and provide a comprehensive understanding of the topic. Following this introductory chapter, the subsequent sections will delve into the theoretical underpinnings of customer segmentation and marketing basket analysis, elucidating their theoretical frameworks and practical applications within the B2B landscape.

The case study of the office supply company will then be meticulously presented, delving into the company's challenges, strategies, and outcomes. The study will culminate in a synthesis of findings, drawing connections between theory and practice. This synthesis will pave the way for the formulation of recommendations and implications for B2B companies aspiring to enhance their sales strategies through customer segmentation and marketing basket analysis.

1.2 Purpose of The Study

1) Identify Customer Segments: Determine distinct customer segments within the B2B clientele of the office supply company. This involves categorizing customers based on their purchasing behavior during the study period using K-Means clustering.
2) Assess Algorithm Performance: Evaluate and compare the performance of different market basket algorithms such as Apriori, FP-Growth and association rule, in terms of their ability to discover meaningful patterns and associations within transaction data and efficiency in terms of execution time, scalability, and accuracy in uncovering valuable insights from transaction datasets.
3) Enhance Data-Driven Decision-Making: Apply the segmentation and marketing basket analysis techniques to a real-world scenario within the office supply company. Demonstrate the effectiveness of these methods in a practical business context and value of data-driven decision-making within the organization. Showcase how segmentation and basket analysis can provide actionable insights that lead to improved business outcomes.

1.3 Scope of the Study

The data that is used in this independence study is based on validated sales transactions from a business-to-business (B2B) office supplier and trading company from Thailand during 1 year period from 1st July 2022 to 30th June 2023. Focusing on only commercially registered companies that averagely spend less than 40,000 Baht monthly, these companies are so called "Small-Medium business" (SMB) or "Field Sales company" which are under the responsible of field sales representatives.

The companies in this study are only located in the following 17 provinces of Thailand:

- Bangkok
- Chachoengsao
- Chonburi
- Nakhon Pathom
- Nakhon Ratchasima
- Nonthaburi
- Pathum Thani
- Phetchaburi
- Phra Nakhon Si Ayutthaya
- Prachinburi
- Prachuap Khiri Khan
- Ratchaburi
- Rayong
- Samut Prakan
- Samut Sakhon
- Samut Songkram
- Saraburi

1.4 Expected Benefits of The Study

1) Segmentation Insights: The study might reveal distinct customer segments within the B2B customer base, such as small businesses, medium-sized enterprises, and large corporations. Each segment could have unique preferences, buying patterns, and needs.
2) Basket Analysis Patterns: By conducting marketing basket analysis, we could uncover interesting product associations. For example, we might find that when customers purchase office printers, they often also buy printer cartridges and paper. This information can guide cross-selling and upselling strategies.
3) Sales Improvement Strategies: The study might suggest specific actions to boost sales based on the insights gained. These strategies could include targeted promotions, product bundling, tailored product recommendations, or improving customer engagement through personalized approaches. For instance, small businesses might be interested in cost-effective office bundles, while larger corporations might prefer bulk discounts.
4) Performance Measurement: The expected outcome could involve defining metrics to measure the success of the implemented strategies. This might include tracking changes in sales revenue, customer retention, average order value, and customer satisfaction within each segment.

1.5 Terminology

1) Business to Business (B2B) refers to the dealings of the business activities between one business enterprises with other. The products and services involved in these types of marketing activities are specially meant for the further production related to goods and services. Under B2B, process related to buying and selling takes longer time to execute as the decision- making involved under this takes place at more than one level. (Josan, 2018)

2) Customer segmentation is an unsupervised-learning process and utilizes different clustering approaches which have the goal to separate customer data based on similarity. Hereby, similarity is measured by an objective function such as Euclidean distance. It should be noted that customer behavior is a continuous process, with customer needs, wants and satisfaction changing over time. Accordingly, the processes and underlying procedures implemented in companies must be flexible to accommodate this high level of dynamism (Griva et al., 2021).
3) K-means clustering is a popular unsupervised machine learning algorithm used for partitioning a dataset into distinct groups or clusters based on the similarity of data points. The main goal of K-means clustering is to group data points that are similar to each other into the same cluster while keeping different clusters as dissimilar as possible. This algorithm is commonly used in various fields, including data analysis, image processing, and customer segmentation. (Gomes and Meisen, 2023)
4) Market Basket Analysis (MBA) also known as association rule learning or affinity analysis, is a data mining technique that can be used in various fields, such as marketing, bioinformatics, education field, nuclear science etc. The main aim of MBA in marketing is to provide the information to the retailer to understand the purchase behavior of the buyer, which can help the retailer in correct decision making. (Kaur and Kang, 2016)
5) The Apriori algorithm is a classic association rule mining algorithm used in data mining and market basket analysis. It is designed to discover interesting relationships and patterns in large datasets, particularly in transactional databases such as retail sales or e-commerce records. The primary goal of the Apriori algorithm is to find frequent itemsets and generate association rules based on these itemsets. (Martinez and Escobar, 2021)
6) The FP-Growth (Frequent Pattern Growth) algorithm is an advanced data mining and pattern recognition algorithm used for mining frequent itemsets and generating association rules in large datasets. It was introduced as an improvement over the Apriori algorithm and is designed to be more efficient, especially when dealing with large and dense datasets. (Martinez and Escobar, 2021)

7) Association rule mining is a data mining technique used to discover interesting relationships, patterns, or associations within large datasets. The primary goal of association rule algorithms is to identify rules that describe how items in a dataset tend to co-occur or be associated with one another. These rules are typically expressed in the form of "if X, then Y," where X and Y represent sets of items, and the rule indicates that there is an association between the presence of items in X and the presence of items in Y in transactions. (Kaur and Kang, 2016)

Chapter 2

THEORY AND LITERATURE REVIEW

This section discusses the theories and related research applied in the case study of customer segmentation and market basket analysis of B2B office supplier company.

2.1 Business-to-Business (B2B)

Business-to-business (B2B), also called B-to-B, is a form of transaction between businesses, such as one involving a manufacturer and wholesaler, or a wholesaler and a retailer. Business-to-business refers to business that is conducted between companies, rather than between a company and individual consumer.

Business-to-business transactions are common in a typical supply chain, as companies purchase components and products such as other raw materials for use in the manufacturing processes. Finished products can then be sold to individuals via business-to-consumer transactions.

Any form of supply chain is a business-to-business model because it entails businesses transacting with one another until the peculiar needs of each business are met. Manufacturing companies, wholesalers and retail businesses often engage in business-to-business transactions before their finished products get to the end user. Aside from the exchange of goods and services between businesses, information is also exchanged via business-to-business model. When employers and employees from different companies interact with one another, exchange information and pass knowledge, this is also a form of business-to-business practice. In many countries, business to business transactions occupy a major position in the country's commerce.

2.2 Customer Segmentation

Customers can have several types of characteristics and can be of different importance to a company. For companies to know which customers are of significance, a segmentation of customers' needs to be done (McDonald and Dunbar, 2012). The theory of segmentation is the process where identifying characteristics of different customers and dividing them into groups. What companies often do when segmenting their customers is to divide them based on how much revenue they contribute to the company based on their purchase volumes.

Identifying and classifying customers leads to a better understanding of who the customers are and what type of demand the customers require. Some customer groups can have a high degree of innovation where changes within the customer group over time often occur. For these types of customers, we need to be aware of the requirement changes to meet the customer demand in the best way which also fulfils the customer needs (Chen et al., 2004).

According to Chen et al. (2004) from companies' perspectives, customers are of different significance and to be able to stay in the market, companies need to distribute their attention unevenly, meaning that they need to move attention from the non-profit consumers to the ones with higher profit. For a company to continue gaining profit, they need to pay a lot of attention to the customers that consume their products or services frequently or in greater volumes to create groups that are most profitable.

When a company has the right knowledge about the customer requirements it will give them the ability to divide the customers more easily into segmentation groups. Furthermore, the company can more easily find out what satisfies their customers and even surprises them. This kind of information can be used for further improvements to their services or products. These days, customer service is as important for the customers as the actual product or service, and it is important that the companies have this part set. Finally, segmenting customers can simplify the choices of how much and what the

company should put emphasis on when it comes to the degree of services that the distinct groups should get (Buttle, 2009).

Machine learning and artificial intelligence have clear advantages over traditional statistical methods when: (a) there are a multitude of variables available for analysis, (b) the associations between the variables are uncertain (and likely to be highly complex), (c) the values of each variable are evolving constantly (such as in the case of a GPS), and (d) when understanding correlations between variables are more important that causation. The great strength of machine learning models is in making predictions, especially where an atheoretical prediction will work well. This is the reason that machine learning models are evaluated on criteria such as scalability, real-time implementation, and cross-validated predictive accuracy rather than on internal and external validity and theoretical foundations which are more suited to the traditional models. Artificial intelligence and machine learning in marketing science are currently gaining more importance to leverage predictive segmentation (Verma et al., 2021).

2.3 RFM Analysis

The Recency-Frequency-Monetary (RFM) analysis has emerged as a prominent method for customer segmentation and marketing optimization. Rooted in the principles of customer behavior analysis, RFM analysis provides a structured framework for businesses to categorize customers based on their recency of purchases, frequency of transactions, and monetary contributions.

2.3.1 Recency

Recency signifies the time elapsed since a customer's last purchase. It is computed by subtracting the customer's last transaction date from a reference date. Recency scores are determined by segmenting the time intervals and assigning appropriate weights to represent recency levels.

2.3.2 Frequency

Frequency pertains to the number of transactions conducted by a customer within a designated timeframe. Calculation of frequency involves tallying the occurrences of purchases. Similar to recency, frequency scores are assigned by dividing the range of transactions and allotting scores accordingly.

2.3.3 Monetary

Monetary denotes the amount of money spent by a customer over a specified period. Monetary scores are derived by categorizing spending levels and assigning scores reflective of customers' monetary contributions.

RFM analysis facilitates customer segmentation by dividing the customer base into distinct groups with varying RFM score combinations. These segments provide valuable insights into customer behavior and preferences. High RFM score segments often comprise loyal and valuable customers who can be targeted for premium services, loyalty programs, and personalized marketing campaigns. On the other hand, low RFM score segments may require tailored re-engagement strategies to renew interest and increase their level of involvement.

According to Gomes and Meisen (2023) study, The RFM analysis is by far the most popular feature selection method for feature selection methods for customer representation with 44 of 55 publications that use this feature selection methods. In some works, e.g., Stormi et al. (2020) the RFM-analysis is extended by additional features.

An example retail use case that combines RFM-analysis and k-means is provided by Hsu and Huang (2020). In their research they want to identify VIP customers. VIP customers are buyers of critical products which are not purchased by the average customer. In their approach, they apply the RFM-analysis on over 600,000 transactions from around 3800 customers. The segmentation is based on the 20%-quantile of RFM-values.

2.4 Clustering Analysis Method

Most research, using Data Driven segmentation to group individuals in the market, use a statistical method based on one of the family of cluster analysis (Omran et al., 2007). Cluster analysis is a method for the analysis and organizing an enormous bulk of multivariate or scientific data to decrease information overload and to discover relationship and classes in unorganized data sets. It can assist in discovery the structure or causality in complex bodies of data.

Dolnicar (2002) indicates that cluster analysis is "a toolbox of highly interdisciplinary techniques of multivariate data analysis" by dividing number of individuals into subgroups based on "a pre-specified criterion (e.g., minimal variance within each resulting cluster) which is assumed to reflect the similarity of individuals within the subgroups and the dissimilarity between them".

Jain (2017) proposes clustering analysis is a prominent technique to segment market based on benefit sought. Several studies by Brunner and Siegrist (2011) and Liu et al. (2011) have used benefit segmentation with factor and cluster analysis method to segment market.

Table 2.1 Summary of clustering classifications and the common algorithms used to achieve partitioning, hierarchical and advanced algorithm.

Clustering Classification	Description	Common Algorithms
Partitioning	Partitions the data into a user specified number of groups. Each point belongs to one group. Does not work well for irregularly shaped clusters.	K-means, K-medoids
Hierarchical	Decomposes data into a hierarchy of groups, each larger group contains a set of subgroups.	Agglomerative Clustering, Divisive Clustering, Ward's method, Nearest Neighbor
Advanced	To solve the question of high dimensionality and large data set.	DBSCAN, Fuzzy, OPTICS, SNN

Table 2.1 showed that there are three types of clustering analysis: Partitioning Clustering, Hierarchical Clustering, Advanced Clustering (Kassambara, 2018). Specifically, partitioning clustering is used to classify observations within a data set into multiple groups based on their similarity. Hierarchical clustering is an alternative approach to partitioning clustering for grouping objects based on their similarity in contrast to partitioning clustering, hierarchical clustering does not require to pre-specify the number of clusters to be produced.

Partitioning Clustering includes three methods: K-Mean, K-Medoids, Clara; Hierarchical Clustering has two methods such as Agglomerative Clustering, Divisive Clustering; and Advanced Clustering includes four methods as Hierarchical K-mean, Model Based Clustering, Fuzzy, SOM, DBSCAN (Kassambara, 2018).

Hierarchical clustering can be subdivided into two types: Agglomerative clustering and Divisive clustering. With the data types of this study, numeric data, the Agglomerative clustering is used for analyzing data. Agglomerative clustering in which, each observation is initially considered as a cluster of its leaf. Then, the most similar clusters are successively merged until there is just one single big cluster (root). The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It's also known as AGNES (Agglomerative Nesting). The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects named dendrogram (Kassambara, 2018).

2.5 K-Mean Clustering

K-means clustering is a widely used data analysis technique in the field of machine learning and data mining. It is an unsupervised learning algorithm that partitions a dataset into distinct groups or clusters based on the similarity of data points. The "K" in K-means represents the number of clusters that the algorithm aims to identify.

The primary goal of K-means clustering is to assign each data point to one of K clusters in a way that minimizes the within-cluster variance or distance between data points within the same cluster. The algorithm iteratively refines the cluster assignments until convergence, ensuring that data points are grouped together with others that are most similar to them.

The key steps involved in K-means clustering are as follows:

1) Initialization: Choose K initial cluster centroids (representative points). This can be done randomly or using other strategies.
2) Assignment: Assign each data point to the nearest centroid, typically based on Euclidean distance or other distance metrics.
3) Update: Recalculate the centroids of each cluster by taking the mean of all data points assigned to that cluster.
4) Repeat: Repeat the assignment and update steps until convergence or a predefined stopping criterion is met. Convergence occurs when the cluster assignments no longer change significantly.
5) Result: The final cluster assignments and centroids represent the output of the K-means algorithm.

K-Means clustering holds significant importance as a versatile tool for uncovering patterns and structure in data. Its simplicity, efficiency, and interpretability make it a valuable asset in various research domains, contributing to informed decision-making and knowledge discovery. While the algorithm is not without limitations, its adaptability and wide range of applications underscore its enduring relevance in the field of machine learning and data analysis.

According to Gomes and Meisen (2023) study, The K-means clustering is by far the most popular classification method with 41 out of 105 studies, due to its computationally efficient and scales well to large datasets, making it suitable for real-time or big data applications and also its interpretability.

2.5.1 Elbow Method

The Elbow Method is a popular and intuitive technique used to determine the optimal number of clusters (k) in K-means clustering, a widely employed unsupervised machine learning algorithm. K-means clustering seeks to partition a dataset into k distinct clusters based on the similarity of data points. The Elbow Method aids in the selection of an appropriate value for k by assessing the trade-off between the number of clusters and the resulting within-cluster variability.

In K-means clustering, the algorithm assigns data points to clusters to minimize the within-cluster variability, which is often measured as the sum of squared distances between data points and their cluster centroids.

Selecting the appropriate number of clusters (k) is crucial. Too few clusters may oversimplify the data, while too many clusters may overfit the data, resulting in poor generalization.

The Elbow Method examines the relationship between the number of clusters (k) and the within-cluster variability. As k increases, the within-cluster variability typically decreases because the data points are closer to their respective cluster centroids. The Elbow Point represents the value of k at which this decrease in within-cluster variability starts to slow down, resembling the shape of an "elbow" in a plot.

The Elbow method has several steps:

1) Initially, researchers perform K-means clustering on the dataset with a range of k values, typically starting from a small number (e.g., 2) and incrementally increasing it.
2) For each k value, the within-cluster variability (often measured as the sum of squared distances) is computed as a metric of how well the data points are grouped within clusters.
3) The computed within-cluster variabilities for different k values are then plotted on a graph. As k increases, the within-cluster variability generally decreases. The Elbow

Point corresponds to the point on the plot where the rate of decrease in within-cluster variability begins to slow down, forming an "elbow" shape.

4) The optimal number of clusters is typically chosen at or just before the Elbow Point. The rationale is to strike a balance between minimizing within-cluster variability (closeness of data points within clusters) and avoiding excessive complexity (too many clusters).

In conclusion, the Elbow Method serves as a valuable tool for selecting the optimal number of clusters in K-means clustering by balancing within-cluster variability and model complexity. Its principles, application, and significance span various research and practical domains, aiding researchers and practitioners in data-driven decision-making and pattern recognition tasks.

2.6 Market Basket Analysis

Market Basket Analysis as well known by the term association rule learning or affinity analysis which is basically a data mining procedure that is being used widely in the field of education, nuclear science, bioinformatics and marketing. In this MBA method, the purchase behavior of the buyer is analyzed, and this information is handed over to the retailer so that it can help retailers in better decision-making (Kaur and Kang, 2016). In a world of competitive markets, to sustain a good position in the market is always a challenge for organizations as it always depends on the organizations capabilities of decision making and understanding the behavior of customers. Hence examining customer-buying pattern is crucial for an organization.

2.6.1 Association Rules

Association rules are a fundamental concept in data mining that focus on discovering interesting relationships or patterns within datasets. These patterns typically involve the co-occurrence of items in transactions. Association rules are used to uncover dependencies between different items in a dataset, such as items frequently purchased

together in a retail transaction or items frequently viewed together on an e-commerce website.

An association rule has two main components:

- Antecedent: This is the set of items on the left-hand side of the rule.
- Consequent: This is the item (or set of items) on the right-hand side of the rule.

For example, an association rule might look like: {Milk, Bread} => {Eggs}. This rule suggests that if a customer buys both Milk and Bread, they are likely to buy Eggs as well.

Databases on customer transactions have been made available to facilitate the development of methods that find product groups or items automatically in the database (Samboteng et al., 2022). An example is the transaction data for the supermarket. The transaction data lists all items that a customer purchases in a single buying transaction. The seller wants to know whether the customer always purchases the product together. Sellers may use this information to establish a supermarket layout to allow for optimal arrangements of the items for promotional purposes, the segmentation of buyers, and the associations to provide information by means of a 'if - then' relationship calculated on the basis of probabilistic evidence. The idea of an association rule is to examine all possible connections between items and to select only those which will most likely be indicators of dependence. The term precedent is usually used to indicate the "IF," which is the result of which is "THEN." Antecedent is a group of items which have no common relationship in this analysis. The association rules are developed in two stages, the following (Ruswati et al., 2018):

1) Analyzing patterns of high frequency. This phase seeks a combination of items which meet the minimum support value requirements in the database. The rule "A => B," (1) is supported by the likelihood that a transaction will concurrently involve attributes or sets of A and B attributes. The mathematical equation is: support (A => B) = P (A alternatively B) (2) Info: A => B = items that are shown

together P (A oscillating B) — probability of A and B transactions divided by the total transaction number.

2) Association Rules Establishment. Once all high frequency patterns are detected, see the association rule, which meets the minimum confidence requirements by calculating whether the A then B rule is confident. The mathematical confidence formula is: Trust (A => B) = P (B | A) (3) Info: A => B = items simultaneously displayed P (B | A) = probability of the number of X and Y transactions divided by the number of X-containing transactions.

Support is the measure of how frequently a particular itemset appears in the dataset. It is calculated as the ratio of the number of transactions containing the itemset to the total number of transactions. Mathematically:

$$Support\ (A) = \frac{the\ amount\ the\ transaction\ contains\ A}{total\ transactions} \times 100\% \quad (2.1)$$

By the following formula, the supporting value for the two items is achieved.

$$Support\ (A \cap B) = \frac{the\ amount\ the\ transaction\ contains\ A\ and\ B}{total\ transactions} \times 100\% \quad (2.2)$$

Equation 2.1 and 2.2, the support formula in association rule mining is a critical metric that quantifies the frequency of occurrence of a specific itemset or combination of items in a dataset. It is typically represented as Support(A), where A represents the itemset of interest. it calculates support by dividing the number of transactions in which the itemset A appears by the total number of transactions in the dataset. The result is often multiplied by 100% to express it as a percentage. In essence, support measures how often a particular itemset occurs within the dataset, reflecting its prevalence. This metric serves as the foundation for identifying frequent itemsets and generating meaningful association rules. By setting a minimum support threshold, analysts can filter out infrequent itemsets, ensuring that the rules generated are based on patterns that occur with sufficient frequency to be of practical significance.

A frequent itemset is an itemset whose support is above a predefined threshold. It is a set of items that appear together in transactions frequently enough to be considered interesting. A higher support value indicates that the itemset or rule is more common or frequent in the dataset. Conversely, a lower support value suggests that the rule occurs less frequently. Support is often used in combination with other metrics like confidence and lift to filter and select interesting rules. We can set a minimum support threshold to focus on rules that meet a certain level of significance. Rules with support values above this threshold are considered more meaningful. Be cautious when interpreting support alone. Extremely high support values may indicate trivial or common patterns that might not be very insightful. It's often valuable to consider other metrics like confidence, lift, or conviction to gain a more comprehensive understanding of the rule's significance (Brin et al., 1997).

Confidence is a measure of the likelihood that an association rule is true. For an association rule A => B, confidence is calculated as the ratio of the support of the combined itemset {A, B} to the support of the antecedent itemset {A}. The main idea is to search for frequent item sets first (a set of items that meet minimum support). Delete itemset based on the predetermined minimum support level from the second transaction database. Next, create itemset association rules, which meet the minimum data base confidence value.

$$Confidence\ (A,B)\ = \frac{the\ amount\ the\ transaction\ contains\ A\ and\ B}{the\ amount\ the\ transaction\ contains\ A} \times\ 100\% \quad (2.3)$$

Equation 2.3, the confidence formula in association rule mining is a vital metric that assesses the strength of an association between items in a rule. It helps quantify how often a specific consequence (item B) occurs when a certain condition (item A) is present in a transaction. Confidence is calculated as Confidence (A, B), where A represents the antecedent (condition), and B represents the consequent (outcome) of the association rule. The formula computes confidence by dividing the support of the combined itemset A and B by the support of the antecedent itemset A alone. In essence, it measures the conditional probability of item B occurring given that item A is present. A higher confidence value indicates a stronger association between the items.

Confidence values range from 0 to 1. A confidence of 1 indicates a perfect association, meaning that whenever the antecedent is present, the consequent is also present in the same transaction. A confidence of 0 indicates no association between the antecedent and the consequent. Higher confidence values indicate a stronger association between the antecedent and consequent. Similar to support, we can set a minimum confidence threshold to filter and select interesting rules. Rules with confidence values above this threshold are considered more reliable and meaningful.

Confidence should be considered in relative terms. Compare the confidence values of different rules to identify which rules exhibit stronger associations. Rules with higher confidence values are typically more actionable and dependable. Martinez and Escobar (2021) suggest that it is important to strike a balance between support and confidence when interpreting rules. High-confidence rules may be less frequent, but they are strong associations. Conversely, high-support rules may have lower confidence and could be less informative.

Lift is a measure of the strength of an association rule, indicating how much more likely the antecedent and consequent are to occur together compared to if they were independent. It's calculated as the ratio of the confidence of the rule to the support of the consequent.

$$Lift\ (A \rightarrow B) = \frac{Support\ (A \cap B)}{Support\ (A) \times Support\ (B)} \tag{2.4}$$

Equation 2.4, the lift formula in association rule mining is a valuable metric that measures the significance of an association between items in a rule, considering the baseline likelihood of finding those items together. Lift is represented as Lift(A⟶B), where A stands for the antecedent (condition) and B represents the consequent (outcome) of the association rule. The formula computes lift by dividing the confidence of the rule (i.e., the probability of item B occurring given item A) by the support of item B. In essence, it quantifies how much more likely item B is to be purchased when item A is present, compared to its likelihood in the absence of item A.

If the rule had a lift of 1, it would imply that the probability of occurrence of the antecedent and that of the consequent are independent of each other. When two events are independent of each other, no rule can be drawn involving those two events. If the lift is > 1, that lets us know the degree to which those two occurrences are dependent on one another and makes those rules potentially useful for predicting the consequent in future data sets. If the lift is < 1, that lets us know the items are substitute to each other. This means that presence of one item has negative effect on presence of other item and vice versa. The value of lift is that it considers both the support of the rule and the overall data set.

Leverage is a measure used in association rule mining to assess the statistical significance of a rule. It quantifies the difference between the observed frequency of the antecedent and consequent occurring together in transactions and what would be expected if they were independent. The formula for calculating leverage for an association rule {A} => {B} is as follows.

$$Leverage\ (A \rightarrow B) = Support\ (A \cup B) - Support\ (A) \times Support\ (B) \quad (2.5)$$

Equation 2.5, the leverage formula in association rule mining is a metric that assesses the extent to which the occurrence of two items together (as indicated by a rule) deviates from what would be expected if they were statistically independent. In this formula, A represents the antecedent (condition), B is the consequent of the association rule, and Support (A ∪ B) measures how often both A and B appear together in transactions. The subtraction of the product of Support (A) and Support (B) from Support (A ∪ B) quantifies the degree of the association.

The leverage value can range from -1 to 1. It has the following interpretations.

- If Leverage = 1: It indicates a perfect positive association. The presence of the antecedent guarantees the presence of the consequent, and there are no unexpected occurrences.

- If Leverage > 0: It suggests a positive association. The presence of the antecedent and consequent together is more frequent than what would be expected by chance.

- If Leverage = 0: It implies that the antecedent and consequent are independent of each other. Their co-occurrence is as expected if they were chosen at random.

- If Leverage < 0: It indicates a negative association. The presence of the antecedent and consequent together is less frequent than expected by chance. This suggests that the presence of one item may reduce the likelihood of the other.

Conviction is a metric used in association rule mining to measure the dependency or the lack of dependency between the antecedent and consequent of a rule. It evaluates how much the confidence in a rule's consequent being true is affected by the absence of the antecedent. Conviction is particularly useful for identifying rules where the consequent is highly dependent on the antecedent. The formula for calculating conviction for an association rule {A} => {B} is as follows.

$$Conviction(A \rightarrow B) = \frac{1 - Support\ (B)}{1 - Confidence(A \rightarrow B)} \quad (2.6)$$

Equation 2.6, the conviction formula in association rule mining is a metric used to assess the strength of implication or causation between items in a rule. In this formula, A represents the antecedent (condition) and B is the consequent (outcome) of the association rule. Conviction quantifies how much more likely item B is to occur in the absence of item A compared to the likelihood suggested by the rule's confidence.

Conviction values can range from 0 to positive infinity.

- If Conviction = 1: It implies that the antecedent and consequent are independent. The absence of the antecedent has no impact on the likelihood of the consequent occurring.

- If Conviction > 1: It indicates positive association and dependence between the antecedent and consequent. A conviction greater than 1 suggests that the rule's consequent is highly dependent on the antecedent, and the absence of the antecedent significantly reduces the likelihood of the consequent occurring.

- If Conviction < 1: It implies negative association or independence. A conviction less than 1 suggests that the antecedent and consequent are less dependent on each other than expected by chance. The rule's confidence is less than what would be expected based on the independence assumption.

Similar to other metrics, we can set a threshold for conviction to filter and select interesting rules. Rules with high conviction values are considered to have a strong dependency between the antecedent and consequent. Conviction should be considered alongside other metrics like support, confidence, lift, and leverage to get a more comprehensive understanding of the rule's significance and practical utility.

2.6.2 Apriori Algorithm

The Apriori algorithm is a specific algorithm used to mine association rules from a dataset. It was introduced by Agrawal and Srikant in 1994. The Apriori algorithm works by iteratively discovering frequent itemsets (sets of items that occur together frequently) and then generating association rules based on those itemsets.

The key idea behind the Apriori algorithm is that if an itemset is frequent, then all of its subsets must also be frequent. This property is known as the "apriori" property, which means that any subset of a frequent itemset is also frequent. The Apriori principle states that if an itemset is frequent, then all of its subsets must also be frequent. This principle guides the algorithm's process of generating candidate itemsets and pruning infrequent ones.

The Apriori algorithm has several steps:

1) Find all frequent individual items (itemsets of size 1) in the dataset.

2) Generate candidate itemsets of size 2 by combining frequent items from step 1
3) Prune candidate itemsets that contain subsets that are not frequent.
4) Repeat steps 2 and 3 to generate larger candidate itemsets until no more can be generated.
5) Generate association rules from the frequent itemsets.

The Apriori algorithm's significance lies in its ability to extract valuable insights from large transactional datasets. It enables businesses and researchers to make data-driven decisions, optimize processes, and enhance user experiences. Moreover, it forms the foundation for more advanced association rule mining techniques and data mining methodologies.

2.6.3 FP-Growth Algorithm

FP-Growth (Frequent Pattern Growth) is a data mining algorithm introduced by Jiawei Han, Jian Pei, and Yiwen Yin in their paper titled "Mining Frequent Patterns without Candidate Generation" in 2000. The algorithm was designed to address some of the limitations of traditional frequent pattern mining algorithms like Apriori. FP-Growth revolutionized the way frequent itemsets are discovered in large datasets by introducing the concept of the FP-Tree.

The FP-Growth algorithm has several steps:

1) FP-Growth starts by scanning the dataset to count the support of each individual item. Support represents the frequency with which an item appears in the transactions. The support counts are used to determine the frequent items that will be used to build the FP-Tree.
2) After obtaining the support counts, the algorithm constructs the FP-Tree, a specialized data structure that captures the relationships and patterns among items in the dataset. The tree begins with a root node. For each transaction in the dataset, FP-Growth inserts the items into the tree, following existing branches if they already exist or creating new branches if necessary. The items are inserted based on their support counts, with more frequent items closer to the root. This

process creates a hierarchical structure where each node represents an item and its occurrence in transactions.

3) Next, FP-Growth uses the FP-Tree to build the Conditional Pattern Base and the Header Table. For each item in the Header Table (sorted by support counts), the algorithm constructs the Conditional Pattern Base. This base comprises paths through the FP-Tree that include the item, capturing the sequences of transactions in which the item appears. The Header Table contains pointers to the first node of each item in the FP-Tree, facilitating efficient navigation.
4) FP-Growth begins mining frequent itemsets by starting with the least frequent item in the Header Table. It recursively explores the Conditional Pattern Base associated with that item, building new sub-FP-Trees and discovering frequent patterns. For each item, the algorithm extracts frequent 1-itemsets and then combines them with the Conditional Pattern Base to find frequent 2-itemsets. This process continues, generating frequent k-itemsets from (k-1)-itemsets until no more frequent itemsets can be found.
5) If the FP-Tree contains a single path, meaning that all items appear in a linear sequence, FP-Growth employs a special optimization. This optimization allows the algorithm to generate frequent itemsets directly from the single path without needing recursive exploration. This is a significant speed improvement in scenarios where single paths are common.
6) Once frequent itemsets are mined, association rules can be generated similarly to other algorithms. Confidence and lift can be calculated for the rules, helping to identify meaningful relationships between items.

The FP-Growth algorithm's significance lies in its efficiency and scalability for mining frequent itemsets and generating association rules from large and complex datasets. It has been a major advancement in data mining, enabling the extraction of valuable insights from diverse transactional data sources. FP-Growth's efficient approach has made it a preferred choice in various research domains and practical applications.

According to Swaminathan and Shavanas (2013) as well as Patil (2022), a comparative analysis of the performance of the Apriori and FP-Growth algorithms on extensive datasets with lower minimum support thresholds reveals an interesting trend. Despite both algorithms yielding identical itemset results, FP-Growth stands out due to its notable speed advantage over Apriori. This enhanced efficiency makes FP-Growth the preferred choice in scenarios characterized by large-scale datasets and lower minimum support requirements.

2.7 Pearson Correlation

Pearson correlation, often referred to as Pearson's correlation coefficient or Pearson's r, is a widely used statistical measure that quantifies the strength and direction of a linear relationship between two continuous variables. Named after its developer, Karl Pearson, this coefficient plays a fundamental role in statistics, data analysis, and various scientific disciplines.

The Pearson correlation coefficient, denoted as "r," is calculated using the following formula:

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}} \quad (2.7)$$

From equation 2.7, where:

- X_i and Y_i represent individual data points from two variables, X and Y.
- $\bar{X}$ and $\bar{Y}$ are the means (average values) of the X and Y variables, respectively.

The Pearson correlation coefficient can take values between -1 and 1, and its interpretation is as follows:

- r=1: Indicates a perfect positive linear relationship between the two variables. As one variable increases, the other also increases proportionally.

- r=−1: Indicates a perfect negative linear relationship between the two variables. As one variable increases, the other decreases proportionally.
- r=0: Suggests no linear relationship between the variables. They are not correlated in a linear fashion.

Pearson correlation is a fundamental statistical measure that quantifies linear relationships between continuous variables. Its wide-ranging applications in research and various fields underscore its importance as a tool for understanding and analyzing data relationships. Researchers and practitioners continue to rely on Pearson correlation to gain insights into the complex interplay between variables.

2.8 Analysis of Variance (ANOVA)

Analysis of Variance (ANOVA) is a powerful statistical technique used to examine the differences among group means in a dataset. Developed by Sir Ronald A. Fisher in the early 20th century, ANOVA has become a cornerstone of experimental design and hypothesis testing, enabling researchers to draw conclusions about the impact of various factors on a response variable.

ANOVA is based on the principle of partitioning the total variability in a dataset into different components to assess the contributions of various factors. The key components include:

1) Total Variability (Total Sum of Squares, SST): This represents the overall variation in the data.
2) Between-Group Variability (Between Sum of Squares, SSB): This measures the variation between the means of different groups or treatment levels. It assesses whether there are statistically significant differences among the groups.
3) Within-Group Variability (Within Sum of Squares, SSW): This quantifies the variation within each group. It represents the random variation or error within the groups.

The ANOVA statistic F is calculated as the ratio of the between-group variability (SSB) to the within-group variability (SSW), divided by the degrees of freedom associated with each component.

$$F = \frac{SSB/df_B}{SSW/df_W} \quad (2.8)$$

From equation 2.8, where:

- SSB: Between Sum of Squares
- SSW: Within Sum of Squares
- df_B: Degrees of freedom for SSB
- df_W: Degrees of freedom for SSW

The F-statistic from ANOVA follows an F-distribution, and its interpretation depends on the p-value associated with it:

- If the p-value is small (typically less than a chosen significance level, e.g., 0.05), it suggests that at least one group mean is significantly different from the others. In this case, post-hoc tests or pairwise comparisons are conducted to identify which specific groups differ from each other.
- If the p-value is not small, it implies that there is insufficient evidence to conclude that the group means are significantly different. Thus, there are no statistically significant differences among the groups.

ANOVA assumes that the data follow a normal distribution and that the variances within groups are approximately equal. Violations of these assumptions can affect the validity of ANOVA results. Additionally, ANOVA assesses group means but does not provide information about individual group differences.

Analysis of Variance (ANOVA) is a powerful statistical tool for comparing group means and assessing the impact of various factors on a response variable. Its principles, formula, interpretation, applications, and variants make it an essential tool in experimental

design, hypothesis testing, and data analysis across numerous scientific disciplines. Researchers and practitioners continue to rely on ANOVA to gain insights and make informed decisions in their respective fields.

2.9 Post-hoc Analysis

Post-hoc analysis, a critical component of statistical research and hypothesis testing, plays a pivotal role in uncovering insights that go beyond the initial findings of an experiment or data analysis. Post-hoc, a Latin term meaning "after the event," refers to the examination of data or the application of statistical tests after the primary analysis has been completed. This literature review delves into the concept of post-hoc analysis, its objectives, methodologies, and implications in various research domains.

Post-hoc analysis helps researchers uncover patterns or relationships in data that may not have been evident during the initial analysis. It allows for a deeper exploration of data beyond the primary research question.

In experiments with multiple groups or conditions, post-hoc analysis is employed to compare specific pairs of groups and determine which pairs exhibit significant differences. This is particularly common after conducting an Analysis of Variance (ANOVA) or a similar omnibus test.

2.9.1 Tukey's Honestly Significant Difference (Tukey's HSD)

Tukey's Honestly Significant Difference (HSD) test, named after its developer John Tukey, is a widely used statistical method for conducting pairwise comparisons in the context of Analysis of Variance (ANOVA) or other omnibus tests. Tukey's HSD is designed to identify which specific pairs of group means are significantly different from each other when there are multiple groups to compare. This literature review provides an overview of Tukey's HSD, its principles, formula, and applications in various research domains.

Tukey's HSD addresses a common question that arises after conducting an ANOVA or a similar test, which is whether there are statistically significant differences between the means of specific groups. Its key principles include:

- Multiple Comparisons: In experiments with three or more groups, conducting multiple pairwise comparisons is essential to pinpoint which pairs of groups have significant differences in their means. Tukey's HSD enables these comparisons while controlling the overall Type I error rate.
- Simultaneous Testing: Tukey's HSD simultaneously evaluates all possible pairwise comparisons, avoiding the need to perform numerous individual t-tests or comparisons, which would increase the risk of Type I errors.

The formula for Tukey's HSD statistic is as follows:

$$HSD = q \times \sqrt{\frac{MSW}{n}} \tag{2.9}$$

From equation 2.9, where:

- HSD is the Tukey HSD value.
- q is the critical value from the Studentized range distribution table (also known as the q-table), which depends on the chosen significance level (usually 0.05) and the degrees of freedom.
- MSW is the Mean Square Within (the within-group variance or error term) obtained from the ANOVA.
- n is the number of observations per group.

The formula results in a critical value, and the differences between group means are considered statistically significant if they exceed this critical value. If the absolute difference in means between two groups exceeds the HSD value, then those two groups are considered significantly different from each other.

Tukey's Honestly Significant Difference (HSD) test is a powerful statistical method for conducting pairwise comparisons after omnibus tests like ANOVA. Its principles, formula, and applications make it a valuable tool in various research domains, enabling researchers to identify specific group differences while controlling for multiple comparisons. However, researchers should exercise caution and ensure that the underlying assumptions are met for valid results.

CHAPTER 3

RESEARCH METHODOLOGY

The research methodology chapter aims to provide a detailed description of the research design, data collection and analysis methods, as well as the ethical considerations considered during the research process. In this chapter, we will explain the rationale behind the chosen research strategy, the techniques used to collect and analyze the data, and the measures taken to ensure the validity and reliability of the results. By providing a clear and comprehensive overview of the methodology used in this study, this chapter aims to enable other researchers to replicate the research and assess the credibility of the findings.

The objective of this study is to investigate the effectiveness of segmentation and marketing basket analysis in boosting B2B sales for an office supplier company. Specifically, this study aims to identify the key customer segments for the company's products, analyze their purchasing behavior and preferences, and develop targeted marketing strategies based on the identified patterns. The research will be conducted through a case study of an office supplier company, which will be selected based on the availability of relevant data and the willingness to participate in the study. The findings of this study are expected to contribute to the existing literature on B2B sales and marketing strategies, as well as provide practical insights for companies seeking to improve their sales performance through segmentation and marketing basket analysis.

3.1 Research Questions

The research aims to investigate the following research questions:

- What are the key customer segments for the office supplier company's products?
- What are the purchasing behavior and preferences of the identified customer segments?
- Can a comparative analysis of market basket algorithms, including Apriori, FP-Growth, and association rule, reveal insights into their performance in terms of pattern discovery, efficiency in execution time, scalability, and accuracy for uncovering valuable associations within transaction datasets?
- How can targeted marketing strategies be developed based on the identified patterns to boost B2B sales for the office supplier company?

3.2 Research Approach

A quantitative approach will be used in this study, Quantitative research is a valuable method for identifying patterns and trends in large datasets related to B2B sales in the office supply industry. In the context of a case study on boosting B2B sales using segmentation and marketing basket analysis for an office supplier company, quantitative research can provide valuable insights into customer purchasing patterns, preferences, and behaviors.

One approach to conducting quantitative research in this context is through analysis of transactional data. By analyzing customer purchase history, the researcher can identify patterns and trends in customer behavior, such as the types of products customers tend to purchase together, and the frequency with which customers make purchases. This information can be used to develop targeted marketing strategies, such as bundling products together in a "basket" to encourage additional purchases.

Overall, quantitative research is a valuable tool for identifying patterns and trends in large datasets related to B2B sales in the office supply industry. By using survey research

and analysis of transactional data, the researcher can gain insights into customer behavior, preferences and develop targeted marketing strategies to boost sales and improve customer satisfaction.

3.3 Research Framework

The research framework for this study on "Boosting B2B Sales using Segmentation and Marketing Basket Analysis: Case Study of Office Supplier Company" aims to provide a step-by-step guide for investigating the effectiveness of segmentation and marketing basket analysis in boosting B2B sales. The framework involves creating business goals and objectives, extracting and cleaning relevant data, feature engineering, customer segmentation through clustering model using RFM analysis as criteria, and running market basket analysis on each cluster. By following this framework, this study aims to provide practical insights for office supplier companies seeking to improve their sales performance through targeted marketing strategies and customer segmentation.

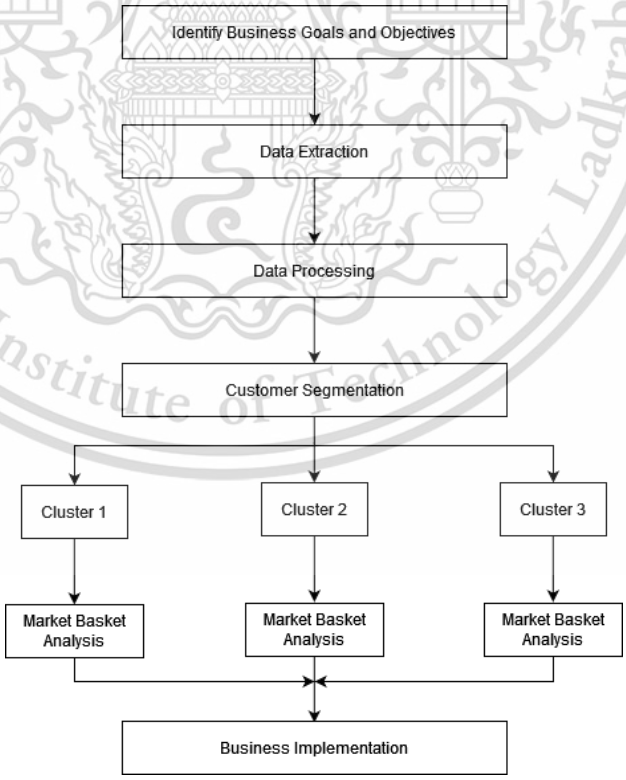


Figure 3.1 Research Framework

Figure 3.1 illustrates a research framework encompassing six sequential steps, commencing with the clear identification of business goals and objectives. Subsequently, relevant data is sourced and extracted using database management software, followed by a critical data cleansing and processing phase to ensure data quality and reliability. The data is then segmented utilizing the K-Means clustering algorithm, grouping similar data points into clusters. Each cluster undergoes a thorough market basket analysis, employing three distinct algorithms to identify itemsets yielding valuable insights. Ultimately, these insights are leveraged to inform and enhance business processes, facilitating data-driven decision-making and potential optimizations within the organization's operations.

3.3.1 Identify Business Goals and Objectives

The first step in this research framework is to create the business goals and objectives for the office supplier company. This will involve understanding the company's sales performance, identifying areas for improvement, and setting specific objectives for boosting B2B sales through segmentation and marketing basket analysis.

3.3.2 Data Extraction

The second step is to extract the relevant data from the company's internal records, such as sales data and customer information. The data extraction process will involve selecting the appropriate data sources and extracting the necessary data fields.

The main data source of this project comes from the company database which stores on Oracle server. The data can be retrieved using SQL query via a program called “Dbeaver”.

DBeaver is a popular open-source database management tool. It is designed to work with various database management systems (DBMS) and provides a unified interface for interacting with them. DBeaver supports a wide range of databases, including MySQL, PostgreSQL, Oracle, Microsoft SQL Server, SQLite, and many more with DBeaver, we can

perform a variety of tasks related to database management and development. Some of its key features include:

- Database Connection: DBeaver allows you to establish connections to multiple databases simultaneously. It provides a connection manager where you can configure connection settings, including host, port, username, and password.
- SQL Editor: The SQL editor in DBeaver enables you to write and execute SQL queries against your connected databases. It offers syntax highlighting, code completion, and formatting for various database-specific SQL dialects.
- Data Viewing and Editing: DBeaver provides a user-friendly interface for browsing and modifying the data in your databases. You can view table structures, browse and edit rows, sort and filter data, and perform basic data manipulation tasks.

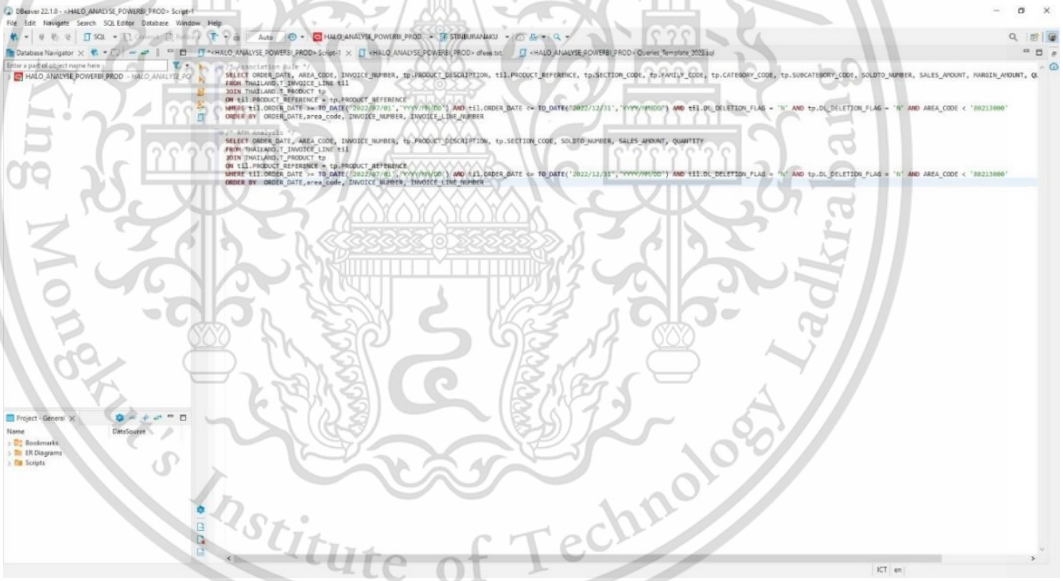
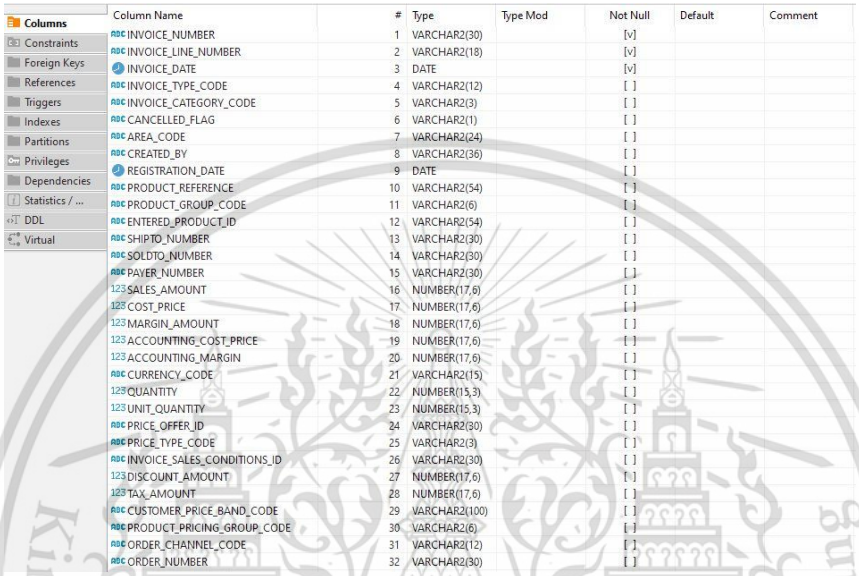


Figure 3.2 Dbeaver program interface.

Figure 3.2 shows Dbeaver program interface along with SQL queries for the dataset. Company data are stored in schemas based on country, for this project we will be focusing on Thailand data which store under “Thailand” schema. There are 75 tables in Thailand schema storing data in almost every aspect of the company.

T_INVOICE_LINE is a table containing validated company sales data, the table is updated monthly, usually 2-3 working days after the first day of the month after all sales from last month are verified by accounting department. The table consists of 68 columns and 14,672,254 rows of data (as of 1st August 2023).



Columns
Constraints
Foreign Keys
References
Triggers
Indexes
Partitions
Privileges
Dependencies
Statistics / ...
DDL
Virtual

Column Name	#	Type	Type Mod	Not Null	Default	Comment
INVOICE_NUMBER	1	VARCHAR2(30)		[v]		
INVOICE_LINE_NUMBER	2	VARCHAR2(18)		[v]		
INVOICE_DATE	3	DATE		[v]		
INVOICE_TYPE_CODE	4	VARCHAR2(12)		[ ]		
INVOICE_CATEGORY_CODE	5	VARCHAR2(3)		[ ]		
CANCELLED_FLAG	6	VARCHAR2(1)		[ ]		
AREA_CODE	7	VARCHAR2(24)		[ ]		
CREATED_BY	8	VARCHAR2(36)		[ ]		
REGISTRATION_DATE	9	DATE		[ ]		
PRODUCT_REFERENCE	10	VARCHAR2(54)		[ ]		
PRODUCT_GROUP_CODE	11	VARCHAR2(6)		[ ]		
ENTERED_PRODUCT_ID	12	VARCHAR2(54)		[ ]		
SHIPTO_NUMBER	13	VARCHAR2(30)		[ ]		
SOLDTO_NUMBER	14	VARCHAR2(30)		[ ]		
PAYER_NUMBER	15	VARCHAR2(30)		[ ]		
SALES_AMOUNT	16	NUMBER(17,6)		[ ]		
COST_PRICE	17	NUMBER(17,6)		[ ]		
MARGIN_AMOUNT	18	NUMBER(17,6)		[ ]		
ACCOUNTING_COST_PRICE	19	NUMBER(17,6)		[ ]		
ACCOUNTING_MARGIN	20	NUMBER(17,6)		[ ]		
CURRENCY_CODE	21	VARCHAR2(15)		[ ]		
QUANTITY	22	NUMBER(15,3)		[ ]		
UNIT_QUANTITY	23	NUMBER(15,3)		[ ]		
PRICE_OFFER_ID	24	VARCHAR2(30)		[ ]		
PRICE_TYPE_CODE	25	VARCHAR2(3)		[ ]		
INVOICE_SALES_CONDITIONS_ID	26	VARCHAR2(30)		[ ]		
DISCOUNT_AMOUNT	27	NUMBER(17,6)		[ ]		
TAX_AMOUNT	28	NUMBER(17,6)		[ ]		
CUSTOMER_PRICE_BAND_CODE	29	VARCHAR2(100)		[ ]		
PRODUCT_PRICING_GROUP_CODE	30	VARCHAR2(6)		[ ]		
ORDER_CHANNEL_CODE	31	VARCHAR2(12)		[ ]		
ORDER_NUMBER	32	VARCHAR2(30)		[ ]		

Figure 3.3 Column names and types of T_INVOICE_LINE table

Figure 3.3 shows number of columns, column names and column types of T_INVOICE_LINE table. There are 68 columns in this table consists of date, number and varchar2 types.

The columns of T_INVOICE_LINE that are used in this study are:

- ORDER_DATE: The date where sales transaction occurs.
- INVOICE_NUMBER: A unique identifier assigned to an invoice used for tracking purposes and helps in organizing and referencing invoices in accounting and record-keeping systems.
- AREA_CODE: A unique identifier assigned to salespersons.
- SOLDTO_NUMBER: A unique identifier assigned to customers.

- PRODUCT_REFERENCE: A unique identifier assigned to a specific product or item. It is used to distinguish and track individual products within a company's inventory or catalog, also known as a product code or SKU (Stock Keeping Unit).
- SALES_AMOUNT: Refers to the total value or monetary value of goods.
- MARGIN_AMOUNT: Refers to the difference between the selling price of a product and its associated cost of production or acquisition. It represents the profit or margin earned on each unit sold.
- QUANTITY: Refers to the numerical amount or count of a particular item or product.
- COUNTRY_CODE: An alphabetical code used to identify countries and regions. It is a standardized two-letter, in this case TH for Thailand.

T_PRODUCT is a table containing validated company products table, the table is updated daily. The table consists of 59 columns and 51,209 rows of data (as of 1st August 2023).

Columns | Constraints | Foreign Keys | References | Triggers | Indexes | Partitions | Privileges | Dependencies | Statistics / ... | DDL | Virtual

Column Name	#	Type	Type Mod	Not Null	Default	Comment
PRODUCT_REFERENCE	1	VARCHAR2(54)		[v]		
PRODUCT_DESCRIPTION	2	VARCHAR2(120)		[ ]		
PRODUCT_GROUP_DESCRIPTION	3	VARCHAR2(120)		[ ]		
CATALOG_PAGE	4	VARCHAR2(9)		[ ]		
PRODUCT_TYPE_CODE	5	VARCHAR2(6)		[ ]		
SECTION_CODE	6	VARCHAR2(9)		[ ]		
FAMILY_CODE	7	VARCHAR2(18)		[ ]		
CATEGORY_CODE	8	VARCHAR2(27)		[ ]		
SUBCATEGORY_CODE	9	VARCHAR2(36)		[ ]		
LOCAL_PRODUCT_MANAGER_CODE	10	VARCHAR2(9)		[ ]		
MATERIAL_TYPE_CODE	11	VARCHAR2(12)		[ ]		
BONUS_CALCULATION_FLAG	12	VARCHAR2(1)		[ ]		
SUPPLIER_ID	13	VARCHAR2(39)		[ ]		
SUPPLIER_PRODUCT_REFERENCE	14	VARCHAR2(105)		[ ]		
GROUP_SUPPLIER_ID	15	VARCHAR2(30)		[ ]		
GROUP_PRODUCT_MANAGER_CODE	16	VARCHAR2(90)		[ ]		
LYRECO_BRAND	17	VARCHAR2(90)		[ ]		
MANUFACTURER	18	VARCHAR2(90)		[ ]		
STATUS_CODE	19	VARCHAR2(9)		[ ]		
PRICING_GROUP_CODE	20	VARCHAR2(9)		[ ]		
PURCHASING_PRICE	21	NUMBER(17,6)		[ ]		
PURCHASING_PRICE_END_DATE	22	DATE		[ ]		
PURCHASING_PRICE_CURRENCY_CO...	23	VARCHAR2(9)		[ ]		
UNIT_PURCHASING_PRICE	24	NUMBER(17,6)		[ ]		
COUNTRY_CODE	25	VARCHAR2(5)		[v]		

Figure 3.4 Column names and types of T_PRODUCT table

Figure 3.4 shows number of columns, column names and column types of T_PRODUCT table. There are 58 columns in this table consists of date, number and varchar2 types.

The columns of T_PRODUCT that are used in this study are:

- PRODUCT_REFERENCE: A unique identifier assigned to a specific product or item. It is used to distinguish and track individual products within a company's inventory or catalog, also known as a product code or SKU (Stock Keeping Unit).
- PRODUCT_DESCRIPTION: Refers to a written explanation that provides name, information about a product's features, specifications, uses, and benefits.
- SUBCATEGORY_CODE: In the product hierarchy or categorization, a subcategory is positioned at a higher level than product reference.
- CATEGORY_CODE: In the product hierarchy or categorization, a category is positioned at a higher level than subcategory code.
- FAMILY_CODE: In the product hierarchy or categorization, a family is positioned at a higher level than category code.
- SECTION_CODE: In the product hierarchy or categorization, a section is positioned at the highest level and at a higher than family code.

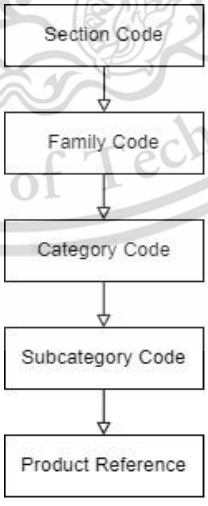


Figure 3.5 Product hierarchy

From figure 3.5 shows product hierarchy with the biggest hierarchy be section code and product reference the smallest. There are 21 section codes, 150 family codes, 558 category codes, 1,079 subcategory code and 51,209 product references.

From the scope of this study the criteria will be set as follow.

The transaction period is 6 months between July 1st 2022 to December 30th 2022, the ORDER_DATE will be set to only retrieve sales data in this period.

This study will be focusing on "Field Sales" customers. The AREA_CODE is set to only retrieve from salesperson code of anything below 80213000. The first two digit from the left refer to organization ID, the third and fourth digit refer to country ID, the fifth and sixth digit refer to sales office and the last two digit represent salesperson ID. In the fifth and sixth digit position any numbers below 30 refer to field sales offices and anything above 30 refer to corporate sales offices. There are 58 sale representatives under field sales office and 26 sale representatives under field sales corporate sales office.

In this case, we include this process in data extraction process where we accounted for transaction that are flagged as DL_DELETION_FLAG = "N" and BUSINESS_SCOPE_FLAG = "Y", both statements filter out sale transactions that are cancelled before a customer make a payment.

Transaction code is also accounted, we only consider these codes where TRANSAC_ITEM_CATEGORY_CODE in (G2N, G2NN, G2W, L2N, L2NN, L2W, TAN, TAS, Z002-TAN, ZECH, ZRB2, ZREG, ZREN, ZRET), this is to filter out free of charge products, gift products, repair products, refurbished products, surplus products, and promotional bundle products.

Sales office is limited to focus only on field sales offices only where AREA_CODE < 80213000. Sales period is also limited where the focusing period is between 1st July 2022 to 30th June 2023.

```
/* Association Rule */
SELECT ORDER_DATE, AREA_CODE, INVOICE_NUMBER, tp.PRODUCT_DESCRIPTION, til.PRODUCT_REFERENCE, tp.SECTION_CODE, tp.FAMILY_CODE, tp.CATEGORY_CODE,
tp.SUBCATEGORY_CODE, SOLDTO_NUMBER, SALES_AMOUNT, MARGIN_AMOUNT, QUANTITY, til.COUNTRY_CODE
FROM THAILAND.T_INVOICE_LINE til
JOIN THAILAND.T_PRODUCT tp
ON til.PRODUCT_REFERENCE = tp.PRODUCT_REFERENCE
WHERE til.ORDER_DATE >= TO_DATE('2022/07/01','YYYY/MM/DD') AND til.ORDER_DATE <= TO_DATE('2023/06/30','YYYY/MM/DD')
AND til.DL_DELETION_FLAG = 'N' AND tp.DL_DELETION_FLAG = 'N' AND AREA_CODE < '80213000'
AND til.TRANSAC_ITEM_CATEGORY_CODE IN ('G2N', 'G2NN', 'G2W', 'L2N', 'L2NN', 'L2W', 'TAN', 'TAS', 'Z002', 'ZECH', 'ZRB2', 'ZREG', 'ZREN', 'ZRET')
ORDER BY  ORDER_DATE, AREA_CODE, INVOICE_NUMBER, INVOICE_LINE_NUMBER
```

Figure 3.6 SQL query for RFM analysis and market basket analysis

Figure 3.6 shows SQL query command to create a historical sales transaction regarding criteria that were mentioned previously. Transaction period was set to 12 months from 1st July 2022 to 30th June 2023. Sales area was set to retrieve only transaction from field sales territories. Flag and transaction code was used to filter out unrelated transactions that could impact the result of the study.

Table 3.1 Data table of historical sales transactions

COUNTRY_CODE	ORDER_DATE	AREA_CODE	INVOICE_NUMBER	PRODUCT_DESCRIPTION	SECTION_CODE	FAMILY_CODE	CATEGORY_CODE	SUBCATEGORY_CODE	SOLDTO_NUMBER	SALES_AMOUNT	QUANTITY
TH	00-01-00	80210212	2212399672	SCOTT I-FOLD PAPER TOWE	2	2001	2001001	2001001002	130007016	1512	24
TH	00-01-00	80210212	2212399672	WASTE BAG 28X36" BLK 1K(	2	2005	2005002	2005002000	130007016	270	5
TH	00-01-00	80210212	2212399672	WASTE BAG INDUSTRIAL TH	2	2005	2005002	2005002000	130007016	275	5
TH	00-01-00	80210212	2212399672	PK24 SCOTT ESSENTIAL STD	2	2001	2001002	2001002003	130007016	5040	24
TH	00-01-00	80210212	2212399672	KIMSOFT JUMBO ROLL TISS	2	2001	2001002	2001002004	130007016	1020	12
TH	00-01-00	80210212	2212399829	SCOTT SLIM ROLL HAND TO	2	2001	2001001	2001001003	130124457	2600	13
TH	00-01-00	80210212	2212399829	POLY-BRITE TOILET CLEANE	2	2002	2002003	2002003000	130124457	280	2
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1359.81	1
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1359.81	1
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1323	1
TH	00-01-00	80210212	2212400614	PK12 ORCA 102 SLIDE BIND	14	14002	14002002	14002002000	130059446	62	1
TH	00-01-00	80210212	2212400614	PK12 ORCA 302 SLIDE BIND	14	14002	14002002	14002002000	130059446	83	1
TH	00-01-00	80210212	2212400614	PK100 ORCA CLEAR COVER	14	14001	14001004	14001004001	130059446	271	1
TH	00-01-00	80210212	2212400614	PENTEL ZEH-03 HI-POLYME	10	10008	10008005	10008005000	130059446	156	12
TH	00-01-00	80210212	2212400614	PK4+2 UHU GLUE STICK MEI	11	11001	11001001	11001001000	130059446	678	2
TH	00-01-00	80210212	2212400614	PK6 SCOTCH 500 CLEAR TAF	11	11002	11002001	11002001002	130059446	422	2
TH	00-01-00	80210212	2212400614	PK6 LYRECO STICKY NOTE 7	9	9004	9004003	9004003001	130059446	274	2
TH	00-01-00	80210212	2212400614	BX12 HORSE H-9100 WOOD	10	10001	10001001	10001001001	130059446	90	2
TH	00-01-00	80210212	2212400614	DOUBLE A WIREBOUND NO	9	9002	9002003	9002003003	130059446	492	12
TH	00-01-00	80210212	2212400614	JUMBO HL110 HAND TRUCK	13	13006	13006002	13006002004	130059446	2991	1
TH	00-01-00	80210212	2212400901	RM500 LYRECO PREMIUM P	8	8001	8001001	8001001001	130046215	740	1
TH	00-01-00	80210213	2212399613	PK2250 HORSE #A2 LABEL S	7	7003	7003001	7003001000	130034672	744	24

Table 3.1 shows an imported dataset SQL query into a data table. The table consists of 12 columns and 914,454 rows.

Table 3.2 Data type and description of data table

Column Name	Type	Description
COUNTRY_CODE	Nominal	A standardized code or abbreviation that uniquely identifies the country or region associated with each sales transaction or customer in the sales data table. It serves as a reference to indicate where the sale or customer is located geographically.

ORDER_DATE	Interval	A timestamps or date-time values that indicate when a sales order was initiated or recorded in the sales data table.
AREA_CODE	Nominal	A unique identifier assigned to salespersons.
INVOICE_NUMBER	Nominal	A unique identifier assigned to an invoice used for tracking purposes and helps in organizing and referencing invoices in accounting and record-keeping systems.
PRODUCT_DESCRIPTION	Nominal	Refers to a written explanation that provides name, information about a product's features, specifications, uses, and benefits.
SECTION_CODE	Nominal	In the product hierarchy or categorization, a section code is positioned at the highest level and at a higher than family code.
FAMILY_CODE	Nominal	In the product hierarchy or categorization, a family is positioned at a higher level than category code
CATEGORY_CODE	Nominal	In the product hierarchy or categorization, a category is positioned at a higher level than subcategory code.
SUBCATEGORY_CODE	Nominal	In the product hierarchy or categorization, a subcategory is positioned at a higher level than product reference.
SOLDTO_NUMBER	Nominal	A unique identifier assigned to customers.
SALES_AMOUNT	Ratio	Refers to the total value or monetary value of goods
QUANTITY	Ratio	Refers to the numerical amount or count of a particular item or product.

Table 3.2 shows data type (Nominal, Ordinal, Interval or Ratio) and description of each column.

3.3.3 Data Processing

The third step is to process the extracted data to ensure that it is accurate and consistent. This will involve identifying and correcting any missing, incomplete, or erroneous data. The method of identifying and deleting corrupt or incorrect information from historical data is data cleaning. The cleansing process stands out as a crucial

component of data science projects undertaken both in academic settings and in practical applications. While datasets employed for instructional and experimental purposes are often pre-processed and devoid of errors, this is not typically the case when dealing with real-world datasets. Numerous factors can lead to inaccuracies in such datasets, necessitating careful identification and remediation of any errors to ensure their optimal utilization. Invalid records can mean deterioration by introducing noise or false information to the future model. Cleaning data is often used in the process of eliminating data that is not important or required. Part of the work is to know which information is relevant or can provide value to the algorithm and handle it for each particular case. Data is duplicated in another common cause. Due Databases are from large organizations and often the data is replicated from various sources.

The next step is to engineer new features from the cleaned data that can be used for customer segmentation and market basket analysis. This may involve creating new variables that capture customer behavior and preferences, such as the frequency and recency of purchases, the average order value, and the product categories purchased. The approach involved the development, transformation, and deletion of features, and the consistency of the model generated with those features was checked over all the cases. Features used to train a machine learning model influence its performance. The better the characteristics are, the better the output would be.

In this study the main features that will be used to decide customer segments are RFM analysis and number of product segments that customers bought.

Recency - How recently a customer made a purchase. Customers who have made more recent purchases are often more engaged and valuable. The recency value is calculated based on when was the last time a customer bought something compare to the last day of study period (30th June 2023). The smaller number of recency reflects the more recent purchase, later the recency value will be converted so that the more recent purchase will have higher recency value.

	CustomerName	LastPurchaseDate	Recency
0	110000011	2022-11-30	30
1	110000042	2022-08-08	144
2	110000045	2022-12-22	8
3	110000050	2022-09-19	102
4	110000059	2022-12-07	23
5	110000092	2022-11-29	31
6	110000102	2022-07-15	168
7	110000116	2022-10-11	80
8	110000137	2022-12-09	21
9	110000153	2022-08-09	143

Figure 3.7 Recency value

Figure 3.7 shows the first 10 rows of a recency table with 3 columns: CustomerName indicates customer ID, LastPurchaseDate shows the latest purchase of the customer and Recency shows the value different between the LastPurchaseDate and the last day of the study period, a higher number indicates a longer period.

Frequency - How frequently a customer makes purchases. Customers who make frequent purchases may be more loyal or engaged with the brand. The frequency value is calculated based on how many times a customer purchase during the study period. The more numbers of purchase, the higher the frequency value.

	CustomerName	Frequency
0	110000011	14
1	110000042	1
2	110000045	49
3	110000050	1
4	110000059	22
5	110000092	6
6	110000102	1
7	110000116	1
8	110000137	12
9	110000153	3

Figure 3.8 Frequency value

Figure 3.8 shows the first 10 rows of a frequency table with 2 columns: CustomerName indicates customer ID and Frequency which indicates the number of times

a customer purchases during the study period counting from distinct number of INVOICE_NUMBER

Monetary - How much money a customer spends. Customers who spend more are typically more valuable to a business. The monetary value is calculated based on how much a customer spend during the study period. The higher a customer spent, the higher the monetary value.

	CustomerName	Monetary
0	110000011	32901.00
1	110000042	3950.00
2	110000045	38122.60
3	110000050	684.00
4	110000059	22984.00
5	110000092	6514.52
6	110000102	4140.00
7	110000116	8890.00
8	110000137	48745.00
9	110000153	2399.00

Figure 3.9 Monetary value

Figure 3.9 shows the first 10 rows of a monetary table with 2 columns: CustomerName indicates customer ID and Monetary which indicates the spending amount that a customer spent during the study period from summing SALES_AMOUNT.

Section Variety – The number of product segments that customer purchase. The higher the number indicates that a customer buy more variety of products from the company during the study period. Company prefers customers who purchase multiple product segments as this can be an indicator that the company could be the main office supplier of the customers.

	CustomerName	Section Variety
0	110000011	14
1	110000042	1
2	110000045	14
3	110000050	2
4	110000059	12
5	110000092	10
6	110000102	7
7	110000116	2
8	110000137	13
9	110000153	4

Figure 3.10 Section Variety value.

Figure 3.10 shows the first 10 rows of a product segment table with 2 columns: CustomerName indicates customer ID and number of segments which a customer purchased during the study period counting from SECTION_CODE.

After the RFM calculation and section variety, all 4 tables are merged on 'CustomerName'.

	CustomerName	Recency	Frequency	Monetary	Section Variety
0	110000011	15	31	59587.64	14
1	110000042	165	2	7900.00	1
2	110000045	1	117	100502.74	14
3	110000050	284	1	684.00	2
4	110000059	28	39	46695.26	12

Figure 3.11 Merging RFM and Section Variety.

Figure 3.11 shows the first 10 rows with 5 columns: CustomerName, Recency, Frequency, Monetary and Segment Variety by merging data tables from figure 3.8 to figure 3.11 by CustomerName.

Next step is to normalize the value into a comparable scale through variable normalization. Variable normalization addresses the issue of disparate scales among features in a dataset. When variables span different ranges, some features may dominate the analysis due to their larger values, leading to biased results and diminished model performance. Normalization seeks to transform variables into a comparable scale, ensuring that each feature contributes proportionally to the analysis.

The min-max normalization is used in this study. This method scales data to a specified range, often [0, 1], by subtracting the minimum value from each data point and dividing by the range. The scale is from 0 to 100.

	CustomerName	Recency	Frequency	Monetary	Section Variety
0	110000011	72.27	92.53	84.94	93.62
1	110000042	18.83	20.20	37.29	7.67
2	110000045	97.23	99.70	92.56	93.62
3	110000050	7.37	7.55	3.65	20.04
4	110000059	56.52	95.36	80.54	84.78

Figure 3.12 Min-Max Normalization of RFM and Section Variety

Figure 3.12 shows the result after changing the name of normalized columns. This finalized table will be used to analyze customer segmentation in the next step.

3.3.4 Customer Segmentation

The fourth step is to segment the customers using a clustering model based on RFM analysis and product segment as the criteria. This will involve identifying the key segments that are most valuable to the company and developing targeted marketing strategies for each segment.

The table from figure 3.12 Is then used as variables to segment customers, sklearn library is imported to Jupyter Notebook. K-mean clustering is the algorithm used for the

segmentation. Before we can run K-mean clustering algorithm, we must identify an appropriate number of clusters and one way to identify an optimal number of clusters for the study is to use elbow method.

The elbow method is a graphical technique used to determine the optimal number of clusters in a K-means clustering algorithm. K-means is an unsupervised machine learning algorithm that aims to partition a dataset into a certain number of clusters, where each data point belongs to the cluster with the nearest mean (centroid). The elbow method is analyzed using sklearn.cluster.KMeans library.

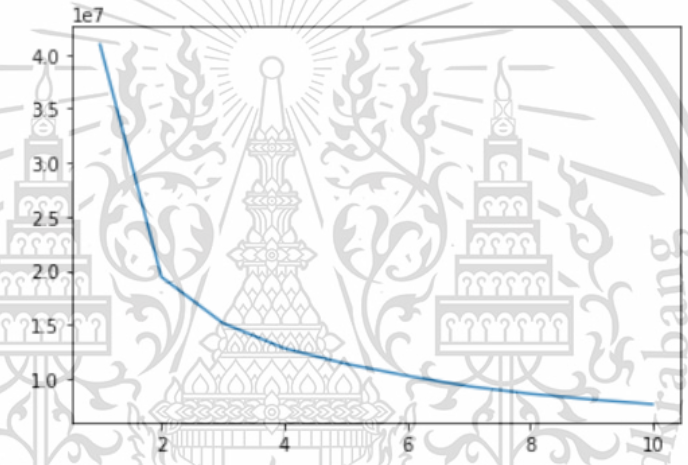


Figure 3.13 Elbow method visualization

Figure 3.13 shows the result of running elbow method algorithm on the dataset. The X-axis represents number of clusters and the Y-axis represents the value of within-cluster sum of squares (WCSS), the WCSS represents the sum of squared distances between data points and their assigned cluster centroids. Once the loop has iterated through all the K values and collected their corresponding WCSS scores. It creates a line plot with K values on the x-axis and the WCSS scores on the y-axis.

The idea is to choose the number of clusters at the point where adding an extra cluster doesn't lead to a significant gain in model performance. This can be seen on the plot as the "elbow" where the variance reduction starts to slow down. Kneed library is

then used to identify the elbow point. The knee (or elbow) point is calculated simply by instantiating the KneeLocator class with x, y and the appropriate curve and direction.

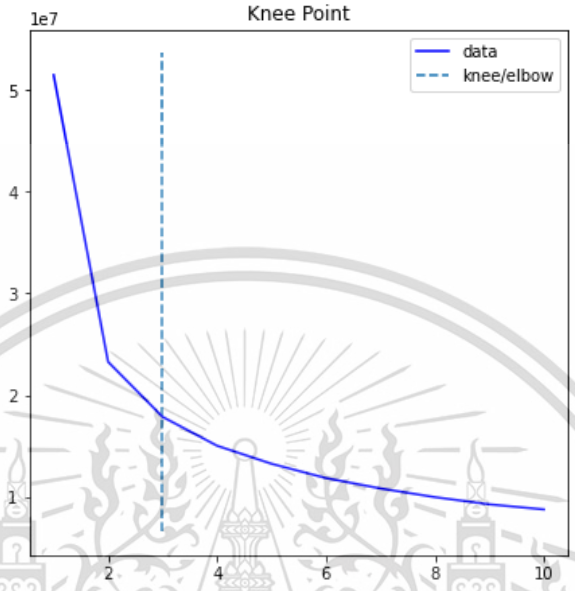


Figure 3.14 Elbow method with KneeLocator result

Figure 3.14 shows the result of a location of an optimal elbow point which KneeLocator indicates that 3 is the optimal number of clusters for this dataset.

After we've performed a K-Means clustering algorithm on the dataset using four variables: recency, frequency, monetary, and segment bought. K-Means assigned each data point to one of three clusters based on similarity in these variables. Once we have these clusters, we need to assess whether the clustering is meaningful and whether the differences in the means of the variables across these clusters are statistically significant. Analysis of Variance (ANOVA) can be used in this case to validify the differences in the means and statistically significant.

Analysis of Variance (ANOVA) is a statistical method used to compare the means of two or more groups. In this case, we are comparing the means of the four variables across the three clusters. The null hypothesis (H0) is that there are no differences in means

between the clusters for each variable. The alternative hypothesis (H1) is that there are significant differences.

When we run the ANOVA test for each variable, we'll get an F-statistic and a p-value. If the p-value is less than the chosen significance level (e.g., 0.05), we can reject the null hypothesis. This means that at least one cluster's mean for that variable is significantly different from the others. If the p-value is higher than the significance level, we fail to reject the null hypothesis, indicating no significant differences.

If we find significant differences in ANOVA for any variable, we might want to perform post hoc tests like Tukey's HSD to determine which specific pairs of clusters have significantly different means. This helps to understand the nature of the differences and relationships between the clusters.

The primary purpose of Tukey's HSD test is to perform pairwise comparisons between group means to identify which groups are significantly different from each other. In other words, it helps you determine which specific clusters are driving the observed significant differences in means.

```
ANOVA for Recency:
F-statistic: 7810.6240
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
=============================================================================
       group1               group2         meandiff p-adj  lower   upper  reject
-----------------------------------------------------------------------------
High Value Customers    Low Value Customers  30.0597  -0.0 29.1119 31.0076   True
High Value Customers Medium Value Customers  51.0548  -0.0 50.0909 52.0187   True
 Low Value Customers Medium Value Customers   20.995  -0.0 20.0371  21.953   True
-----------------------------------------------------------------------------
=========================================
ANOVA for Frequency:
F-statistic: 26020.4937
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
=============================================================================
       group1               group2         meandiff p-adj  lower   upper  reject
-----------------------------------------------------------------------------
High Value Customers    Low Value Customers  31.1468  -0.0 30.5159 31.7776   True
High Value Customers Medium Value Customers  62.4365  -0.0  61.795 63.0781   True
 Low Value Customers Medium Value Customers  31.2898  -0.0 30.6522 31.9273   True
-----------------------------------------------------------------------------
=========================================
ANOVA for Monetary:
F-statistic: 20111.3915
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
=============================================================================
       group1               group2         meandiff p-adj  lower   upper  reject
-----------------------------------------------------------------------------
High Value Customers    Low Value Customers  30.0116  -0.0 29.3147 30.7084   True
High Value Customers Medium Value Customers  60.6367  -0.0  59.928 61.3453   True
 Low Value Customers Medium Value Customers  30.6251  -0.0 29.9208 31.3294   True
-----------------------------------------------------------------------------
=========================================
ANOVA for Section Bought:
F-statistic: 12351.2024
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
=============================================================================
       group1               group2         meandiff p-adj  lower   upper  reject
-----------------------------------------------------------------------------
High Value Customers    Low Value Customers   23.773  -0.0  22.954 24.5919   True
High Value Customers Medium Value Customers   55.693  -0.0 54.8602 56.5258   True
 Low Value Customers Medium Value Customers    31.92  -0.0 31.0923 32.7477   True
-----------------------------------------------------------------------------
=========================================
```

Figure 3.15 ANOVA and Tukey HSD results

Figure 3.15 shows a result of ANOVA and Tukey HSD test. In summary, the ANOVA and Tukey HSD tells us that there are significant differences in the behavior of the customers among the three clusters created. This validates the effectiveness of the clustering approach in capturing meaningful variations in customer behavior related to variables. The result of statistical test on variables and clusters will be further discuss later in chapter 4.

3.3.5 Market Basket Analysis

The fifth step is to run market basket analysis on each customer segment to identify the products that are frequently purchased together. This will involve analyzing the transaction data and identifying the association rules that exist between the different

products. After identifying customer clusters, we can now assign cluster type to customers and their transactions during the study period.

The study will be focusing on product subcategory level, product subcategory level is one hierarchy level above product reference. A product reference is a unique identifier assigned to a specific product or item. It is used to distinguish and track individual products within a company's inventory or catalog, also known as a product code or SKU (Stock Keeping Unit). There are currently 1,079 product subcategories in the company.

After identifying customer clusters, we can now assign cluster type to customers and their transactions during the study period.

	COUNTRY_CODE	INVOICE_NUMBER	QUANTITY	SUBCATEGORY	cluster
0	TH	2212414718	1	OTHER FIRST AID & MEDICAL EQUIPMENT(FIRST AID ...	High Value Customers
1	TH	2212416505	1	CLEAR/TRANSPARENT(ADHESIVES)	High Value Customers
2	TH	2212416505	3	CORRECTION ROLLERS & TAPES(CORRECTION PRODUCTS)	High Value Customers
3	TH	2212416505	1	ORIGINAL(LASER CARTRIDGES)	High Value Customers
4	TH	2212416505	1	DISPOSABLE CUPS, SAUCERS & GLASSES(CATERING SU...	High Value Customers

Figure 3.16 Tagging High Value Customers to invoice number

Figure 3.16 shows a part of data table which consists of column COUNTRY_CODE, INVOICE_NUMBER, QUANTITY, SUBCATEGOTY and cluster. In figure 3.21 shows a table of High Value Customers cluster.

Then the data table from figure 3.22 is unstacked to pivot INVOICE_NUMBER and SUBCATEGORY.

SUBCATEGORY	#9 (4"X9") (ENVELOPES & POSTROOM)	1-HOLE PUNCHES(HOLE PUNCHES)	10"X13"(ENVELOPES & POSTROOM)	16GB(FLASH MEMORY)	2-HOLE PUNCHES(HOLE PUNCHES)	2-RING BINDERS(BINDERS)	2.5"(FLASH MEMORY)	32GB(FLASH MEMORY)
INVOICE_NUMBER								
2212399541	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399542	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399543	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399544	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399555	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 529 columns

Figure 3.17 Unstacking INVOICE_NUMBER and SUBCATEGORY

Figure 3.17 shows a part of data table which has product subcategories as columns and invoice numbers as rows, the cell value indicates the presence of each product subcategory in the invoice. If the cell value is 0, it indicates that product subcategory is absent in that invoice. Cell value of 1 or more than 1 indicates that product subcategory is present in that invoice. The data table is then encoded so that if the cell value is more than 1, it is set to 1. The result table only contains values of 0 or 1.

Each of the resulting clusters represents a unique subset of the data, and at this step we will apply basket analysis to them. Basket analysis, also known as market basket analysis or association rule mining, is a data mining technique that explores the relationships and associations between items within a dataset. In this context, each cluster serves as a 'basket,' and we aim to identify frequent itemsets and association rules within these baskets.

The basket analysis algorithms, Apriori, FP-growth, and Association Rule mining, are subsequently applied to each cluster independently. These algorithms excel in extracting valuable patterns from transactional data, and their utilization on individual clusters enables us to tailor the analysis to the unique characteristics of each subset.

The Apriori algorithm, a classic and widely-used approach, identifies frequent itemsets by iteratively generating candidate itemsets and pruning infrequent ones. It's particularly useful for uncovering associations between items that occur together frequently within a cluster.

The FP-growth algorithm, on the other hand, employs a different strategy by constructing a compact data structure called a FP-tree. This allows for faster and more efficient mining of frequent itemsets, making it suitable for larger datasets and clusters with varying sizes.

Lastly, Association Rule mining explores the discovered frequent itemsets to identify interesting and actionable rules, such as "if item A is purchased, then item B is likely to be purchased as well.".

In our analysis, we utilize both the Apriori and FP-Growth algorithms to extract purchasing patterns within each cluster. Additionally, we assess the efficiency of these algorithms by measuring their runtime on our specific dataset and criteria. Subsequently, we further validate the results obtained from Apriori and FP-Growth by subjecting them to the scrutiny of association rule evaluation, incorporating essential metrics such as confidence and lift values. In summary, our approach involves the following key steps:

1) Applying the Apriori and FP-Growth algorithms to identify purchasing patterns within each cluster.
2) Evaluating the efficiency of both algorithms by measuring their runtime on the dataset.
3) Employing association rules, including metrics like confidence and lift, to validate and substantiate the findings from Apriori and FP-Growth.

These steps collectively contribute to a comprehensive understanding of the purchasing behaviors in each cluster and ensure the robustness and reliability of our analytical results. Further elaboration and discussion of these processes will be provided in Chapter 4.

3.3.6 Business Implementation

The final step in this research framework is to implement the findings and recommendations of the study into the business operations of the office supplier company. This will involve developing a comprehensive action plan based on the insights gained from the customer segmentation and market basket analysis and monitoring the results to ensure the effectiveness of the implemented strategies.

By following this research framework, the study will provide a comprehensive understanding of the effectiveness of segmentation and marketing basket analysis in boosting B2B sales for an office supplier company. The results of this study will be useful for companies seeking to improve their sales performance through targeted marketing strategies and customer segmentation. The framework also allows for the replication of the study in other companies and industries, and the findings can contribute to the existing literature on B2B sales and marketing strategies.

3.4 Research Instrument

The research tools consist of hardware and software, including cloud-based applications, with the following details:

3.4.1 Hardware

- Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz
- RAM 32.0 GB
- NVIDIA GeForce RTX 2070
- 1 TB SSD Storage

3.4.2 Software and Operating System

- Windows 10 Home Build 19045.3448
- DBeaver Database Management Solution
- Jupyter Notebook

3.4.3 Cloud-Based Application

- Google Colab
- Google Drive

Chapter 4

RESULTS AND DISCUSSIONS

This chapter presents the culmination of an in-depth exploration into the art and science of enhancing B2B sales through the strategic utilization of segmentation and marketing basket analysis. In the previous chapters, we laid the groundwork by exploring the theoretical framework and methodological approaches related to segmentation and marketing basket analysis in the context of B2B sales. This chapter serves as a platform for the synthesis, reflection, and discussion of the empirical findings presented in the results chapter. As we navigate through the nuances and implications of our case study with the Office Supplier Company, we aim to provide a comprehensive understanding of how segmentation and marketing basket analysis can be strategically leveraged to bolster B2B sales outcomes. Additionally, we delve into the broader context, critically evaluating our findings against existing literature, industry trends, and best practices. Through this discussion, we seek to uncover actionable insights, potential limitations, and avenues for future research, ultimately contributing to the evolving landscape of B2B sales strategies in an era marked by data-driven decision-making.

4.1 Exploratory Data Analysis

The Exploratory Data Analysis (EDA) phase of this study served as the initial gateway into deciphering the intricacies of the company's sales patterns, customer behaviors, and transactional dynamics. In this chapter, we delve into the pivotal role that EDA played in our research, illuminating the foundational insights it provided and its influence on subsequent analytical steps.

EDA serves as a vital precursor to the more advanced statistical analyses that followed, helping us gain a comprehensive understanding of the dataset's characteristics,

uncover potential outliers, patterns, and correlations, and establish the groundwork for informed decision-making. By delving into the nuances of EDA, we aim to provide a detailed account of the key findings and observations that emerged during this phase of our research, shedding light on the initial clues that directed our subsequent analytical endeavors. Through a careful examination of the data exploration process, we not only enhance our appreciation of the research's empirical underpinnings but also set the stage for a deeper comprehension of the impacts of segmentation and marketing basket analysis on B2B sales within the Office Supplier Company's context.

We start off by introducing the data table of historical sales transactions. The data table is a result of SQL query from company's database system through DBeaver database management system.

Table 4.1 Data table of historical sales transaction

COUNTRY_CODE	ORDER_DATE	AREA_CODE	INVOICE_NUMBER	PRODUCT_DESCRIPTION	SECTION_CODE	FAMILY_CODE	CATEGORY_CODE	SUBCATEGORY_CODE	SOLDTO_NUMBER	SALES_AMOUNT	QUANTITY
TH	00-01-00	80210212	2212399672	SCOTT I-FOLD PAPER TOWE	2	2001	2001001	2001001002	130007016	1512	24
TH	00-01-00	80210212	2212399672	WASTE BAG 28X36" BLK 1K(	2	2005	2005002	2005002000	130007016	270	5
TH	00-01-00	80210212	2212399672	WASTE BAG INDUSTRIAL TH	2	2005	2005002	2005002000	130007016	275	5
TH	00-01-00	80210212	2212399672	PK24 SCOTT ESSENTIAL STD	2	2001	2001002	2001002003	130007016	5040	24
TH	00-01-00	80210212	2212399672	KIMSOFT JUMBO ROLL TISS	2	2001	2001002	2001002004	130007016	1020	12
TH	00-01-00	80210212	2212399829	SCOTT SLIM ROLL HAND TO	2	2001	2001001	2001001003	130124457	2600	13
TH	00-01-00	80210212	2212399829	POLY-BRITE TOILET CLEANE	2	2002	2002003	2002003000	130124457	280	2
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1359.81	1
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1359.81	1
TH	00-01-00	80210212	2212400614	SAFETY JOGGER X2020P S3	3	3007	3007001	3007001000	130059446	1323	1
TH	00-01-00	80210212	2212400614	PK12 ORCA 102 SLIDE BIND	14	14002	14002002	14002002000	130059446	62	1
TH	00-01-00	80210212	2212400614	PK12 ORCA 302 SLIDE BIND	14	14002	14002002	14002002000	130059446	83	1
TH	00-01-00	80210212	2212400614	PK100 ORCA CLEAR COVER	14	14001	14001004	14001004001	130059446	271	1
TH	00-01-00	80210212	2212400614	PENTEL ZEH-03 HI-POLYME	10	10008	10008005	10008005000	130059446	156	12
TH	00-01-00	80210212	2212400614	PK4+2 UHU GLUE STICK MEI	11	11001	11001001	11001001000	130059446	678	2
TH	00-01-00	80210212	2212400614	PK6 SCOTCH 500 CLEAR TAI	11	11002	11002001	11002001002	130059446	422	2
TH	00-01-00	80210212	2212400614	PK6 LYRECO STICKY NOTE 7	9	9004	9004003	9004003001	130059446	274	2
TH	00-01-00	80210212	2212400614	BX12 HORSE H-9100 WOOD	10	10001	10001001	10001001001	130059446	90	2
TH	00-01-00	80210212	2212400614	DOUBLE A WIREBOUND NO	9	9002	9002003	9002003003	130059446	492	12
TH	00-01-00	80210212	2212400614	JUMBO HL110 HAND TRUCK	13	13006	13006002	13006002004	130059446	2991	1
TH	00-01-00	80210212	2212400901	RM500 LYRECO PREMIUM P	8	8001	8001001	8001001001	130046215	740	1
TH	00-01-00	80210213	2212399613	PK2250 HORSE #A2 LABEL S	7	7003	7003001	7003001000	130034672	744	24

Table 4.1 shows an imported dataset SQL query into a data table. The table consists of 12 columns and 914,454 rows. Transaction period was set to 12 months from 1st July 2022 to 30th June 2023. Sales area was set to retrieve only transaction from field sales territories. Flag and transaction code was used to filter out unrelated transactions that could impact the result of the study.

Table 4.2 Data type and unique value

Column Name	Type	Unique Value
COUNTRY_CODE	Nominal	1
ORDER_DATE	Interval	365
AREA_CODE	Nominal	113
INVOICE_NUMBER	Nominal	175,186
PRODUCT_DESCRIPTION	Nominal	7,553
SECTION_CODE	Nominal	18
FAMILY_CODE	Nominal	104
CATEGORY_CODE	Nominal	339
SUBCATEGORY_CODE	Nominal	573
SOLDTO_NUMBER	Nominal	15,399
SALES_AMOUNT	Ratio	21,048
QUANTITY	Ratio	551

Table 4.2 shows data type (Nominal, Ordinal, Interval or Ratio) and unique value of each column of the historical sales transaction data table (table 4.1). 1 unique COUNTRY_CODE means the dataset comes from only 1 country. 365 unique ORDER_DATE represents 12 months study period. 113 AREA_CODE indicates there are 113 sales representatives with 175,186 unique invoices from 15,399 customers. There are 7,553 unique products sold during the study period all of which come from 573 subcategories, 339 categories, 104 product families and 18 sections.

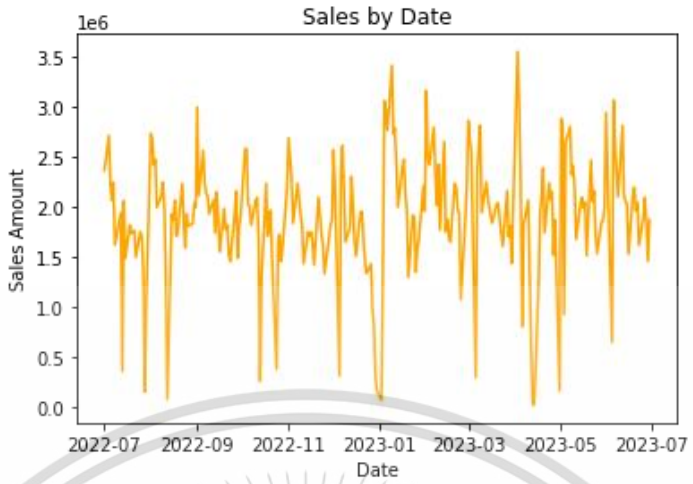


Figure 4.1 Sales amount by date

From Figure 4.1, a line chart is created using Matplotlib, where the x-axis represents dates, the y-axis represents sales amounts in x1,000,000 Baht unit excluding sales from Saturday and Sunday. Notice some huge drops in sales, these occur during Thailand public holiday.

Top 10 Sales By Date

	ORDER_DATE	SALES_AMOUNT	RANK
196	2023-04-03	3547217.14	1
136	2023-01-09	3408624.84	2
153	2023-02-01	3157687.21	3
242	2023-06-06	3063028.61	4
133	2023-01-04	3056070.72	5
197	2023-04-04	3043057.23	6
44	2022-09-01	2991468.06	7
134	2023-01-05	2959532.00	8
239	2023-06-01	2940793.36	9
217	2023-05-02	2880772.40	10

Figure 4.2 Top 10 sales amount by date

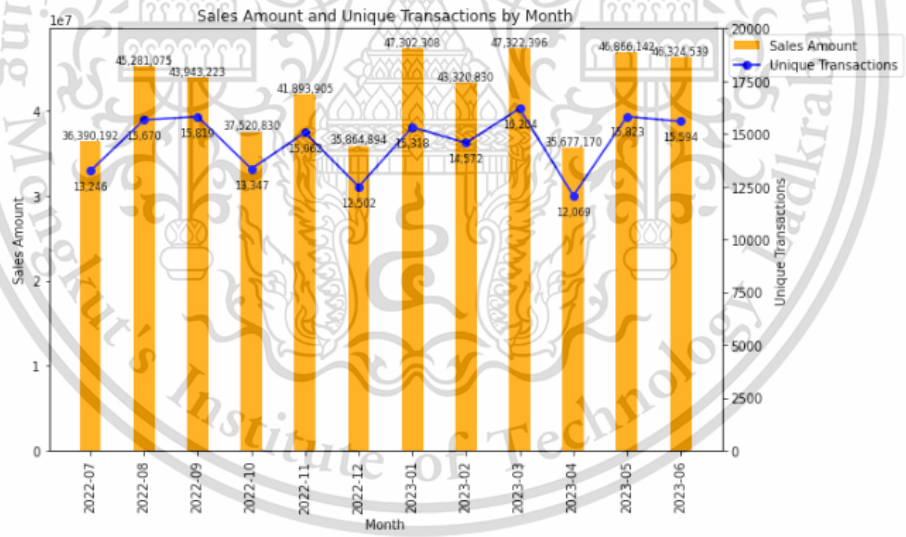
Figure 4.2 shows 10 highest sales amount by date. 3rd April 2023, had the highest sales amount of 3,547,217.14 Baht. It's worth noting that the highest sales tend to occur consistently during the initial week of each month.

Least 10 Sales By Date

	ORDER_DATE	SALES_AMOUNT	RANK
205	2023-04-14	19415.00	1
204	2023-04-13	25861.00	2
131	2023-01-02	69812.88	3
30	2022-08-12	80001.08	4
19	2022-07-28	151363.86	5
130	2022-12-30	162618.00	6
216	2023-05-01	163322.20	7
74	2022-10-13	257941.01	8
176	2023-03-06	294324.83	9
111	2022-12-05	308344.32	10

Figure 4.3 Least 10 sales amount by date

Figure 4.3 shows 10 least sales amount by date. 14th April 2023, had the least sales amount of 19,415.00 Baht. It's worth noting that the least sales occur during Thailand national holiday.




```
Pearson Correlation Coefficient: 0.9499858598113966
P-Value: 2.265501973071313e-06
```

Figure 4.4 Sales amount and unique transactions by month

In Figure 4.4, the sales amounts and unique transactions for each month are displayed, with March 2023 recording the highest sales amount of 47,322,396 Baht and highest 16,204 unique transactions, while April 2023 had the lowest sales amount at

35,677,170 Baht and lowest 12,069 unique transactions. The lower sales figure for April is not surprising, as this month experienced a notable decline in sales, which can be attributed to the occurrence of five Thailand holidays, Chakri Memorial Day on 6th-7th April and Thailand's National Holiday 13rd, 14th and 17th April. The high positive correlation coefficient suggests that as the sales amount increases, the number of unique invoices also tends to increase. This finding implies that periods with higher sales tend to have more unique invoices. The strong positive correlation suggests that the relationship is consistent and robust.

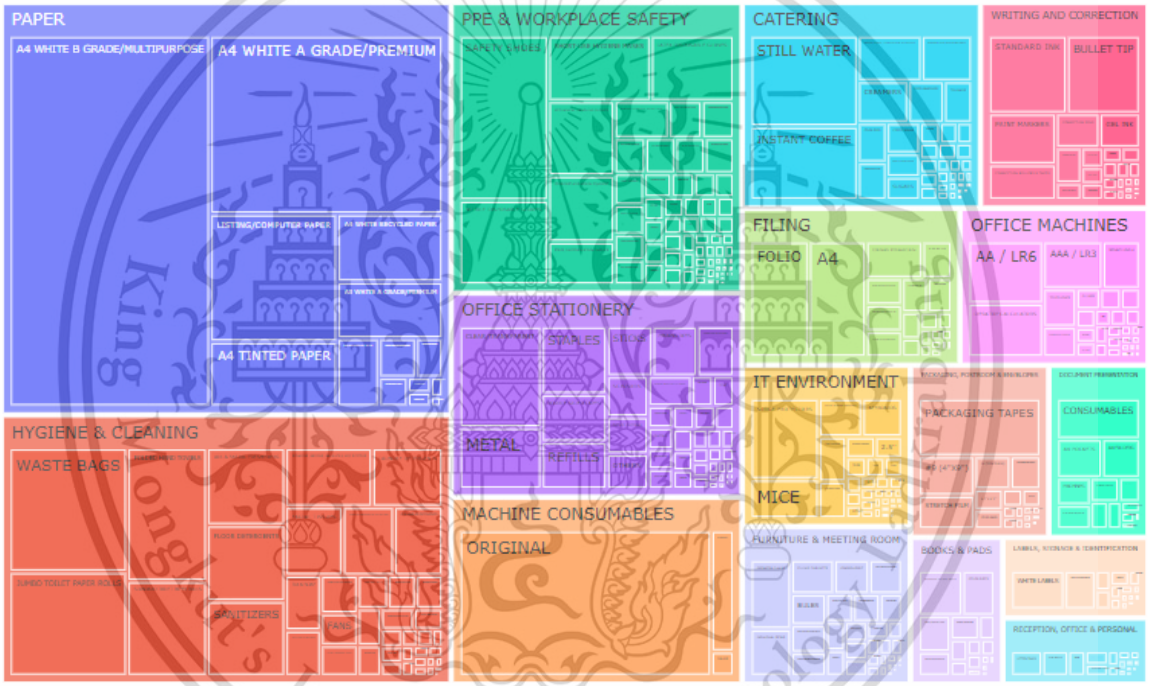


Figure 4.5 Treemap visualization representing product sections and its subcategories

In Figure 4.5, a treemap visualization illustrates the product sections and their respective subcategories. The size of each section box corresponds to the revenue generated, with larger sections denoting higher revenue. Within each product section, subcategories are depicted as smaller boxes, and their size indicates the revenue generated by each subcategory. Notably, there are a total of 17 boxes representing the 17 distinct product sections sold throughout the study period. This graphical

representation provides a clear and intuitive overview of the revenue distribution across product sections and their subcategories.

4.2 Feature Engineering

In the pursuit of enhancing business-to-business (B2B) sales, the strategic utilization of data-driven approaches has become indispensable. One such approach, known as Recency, Frequency, and Monetary (RFM) analysis, has emerged as a powerful tool for understanding customer behavior and tailoring marketing strategies to specific customer segments. In the context of our case study, which focuses on an office supplier company, the ability to precisely target customers and optimize marketing efforts holds the potential to significantly boost sales and improve overall business performance.

RFM analysis relies on three key metrics: Recency, which quantifies how recently a customer made a purchase; Frequency, which measures the number of purchases within a given period; and Monetary, which gauges the total monetary value of those purchases. However, extracting meaningful insights from these raw metrics often requires a careful process of feature engineering. Feature engineering involves the transformation and creation of new features derived from the original data to improve the performance of analytical models. It is a crucial step in RFM analysis, as it enables the conversion of raw RFM values into actionable information that can inform marketing decisions.

One more indicator is introduced along with RFM analysis, the segment bought is an indicator to measure how many product segments that customers bought during the study period. The higher the number indicates that a customer buy more variety of products from the company. Company prefers customers who purchase multiple product segments as this can be an indicator that the company could be the main office supplier of the customers.

In this section, we delve into the critical role of feature engineering in RFM analysis within the context of our case study. We explore the methodologies employed to

engineer informative features, the rationale behind each transformation, and the resultant benefits in terms of segmentation and marketing basket analysis. Through an in-depth examination of the feature engineering process, we shed light on how businesses in the office supply industry can leverage RFM analysis to refine their understanding of customer behavior and, ultimately, elevate their B2B sales strategies.

4.2.1 Recency

How recently a customer made a purchase. Customers who have made more recent purchases are often more engaged and valuable. The recency value is calculated based on when was the last time a customer bought something compare to the last day of study period (31st December 2022). The smaller number of recency reflects the more recent purchase, later the recency value will be converted so that the more recent purchase will have higher recency value.

```
recency_df = df1.groupby(by='SOLDTO_NUMBER', as_index=False)['ORDER_DATE'].max()
recency_df.columns = ['CustomerName', 'LastPurchaseDate']
recent_date = recency_df['LastPurchaseDate'].max()
recency_df['Recency'] = recency_df['LastPurchaseDate'].apply(lambda x: (recent_date - x).days)
recency_df.head()
```

	CustomerName	LastPurchaseDate	Recency
0	110000011	2023-06-15	15
1	110000042	2023-01-16	165
2	110000045	2023-06-29	1
3	110000050	2022-09-19	284
4	110000059	2023-06-02	28

Figure 4.6 Recency code and table

Figure 4.6 The code begins by creating a new DataFrame called recency_df by grouping the original DataFrame df1 by a column named 'SOLDTO_NUMBER' while also aggregating the maximum 'ORDER_DATE' for each group. This effectively identifies the most recent purchase date for each unique 'SOLDTO_NUMBER'. The columns in the new DataFrame are then renamed to 'CustomerName' and 'LastPurchaseDate' for clarity. Next, the code finds the maximum value within the 'LastPurchaseDate' column in recency_df, which represents the most recent purchase date among all customers. This maximum

date is stored in the variable recent_date. Finally, the code calculates the 'Recency' for each customer by applying a lambda function to the 'LastPurchaseDate' column. The lambda function subtracts each customer's 'LastPurchaseDate' from the recent_date and uses the '.days' attribute to convert the time difference into the number of days. This computes how many days have passed since each customer's last purchase, effectively measuring the recency of their interactions with the business. The results are added as a new column 'Recency' in the recency_df.

```
count   15399.00
mean       81.90
std        97.80
min         0.00
25%        11.00
50%        35.00
75%       123.00
max       364.00
Name: Recency, dtype: float64
Median of 'Recency': 35.0
Mode of 'Recency': 3
```

Figure 4.7 Descriptive statistics of recency

Figure 4.7, the recency dataset consists of 15,399 data points. On average, these events or interactions occurred averagely 81.90 days, with a wide-ranging span from 0 to 364 days. The dataset's distribution is somewhat right-skewed, as indicated by a median of 35 days. Notably, the most frequent recency value is 3 days, suggesting a significant occurrence of very recent events or interactions.

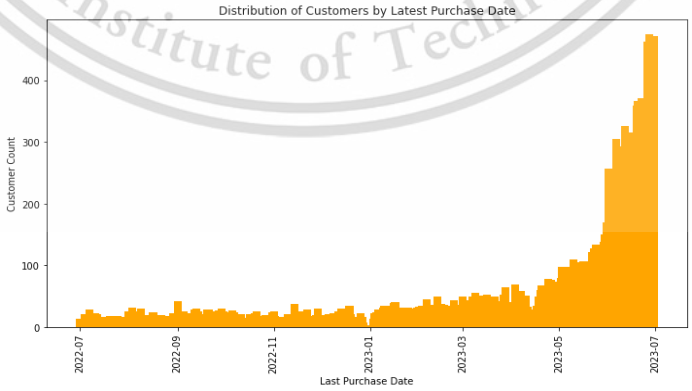


Figure 4.8 Distribution of customers by latest purchase date

Figure 4.8, a distribution plot using Matplotlib, showing the customer count on the y-axis and the 'Last Purchase Date' on the x-axis, helping visualize customer distribution based on their latest purchase dates over time. The graph depicting the distribution of customers by their latest purchase dates reveals a heavily left-skewed pattern. In this distribution, a majority of customers have made their most recent purchases relatively recently, clustered towards the left side of the graph. This suggests that a significant portion of the customer base has engaged with the business recently, indicating a strong and active customer presence.

4.2.2 Frequency

How frequently a customer makes purchases. Customers who make frequent purchases may be more loyal or engaged with the brand. The frequency value is calculated based on how many times a customer purchase during the study period. The more numbers of purchase, the higher the frequency value.

```
frequency_df = df.drop_duplicates().groupby(by=['SOLDTO_NUMBER'], as_index=False)['INVOICE_NUMBER'].nunique()
frequency_df.columns = ['CustomerName', 'Frequency']
frequency_df.head()
```

	CustomerName	Frequency
0	110000011	31
1	110000042	2
2	110000045	117
3	110000050	1
4	110000059	39

Figure 4.9 Frequency code and table

Figure 4.9, the code create a new DataFrame called frequency_df. It begins by using the drop_duplicates() method on the original DataFrame df. This method is applied to remove any duplicate rows in the DataFrame, ensuring that each unique combination of data is represented only once. Afterward, the code groups the deduplicated DataFrame by the 'SOLDTO_NUMBER' column, effectively grouping transactions by customer. Within each customer group, it calculates the number of unique invoices by using the ‘nunique()’ method on the 'INVOICE_NUMBER' column. This count represents the frequency of purchases or transactions made by each customer.

```
count   15399.00
mean       11.38
std        16.14
min         1.00
25%         2.00
50%         6.00
75%        14.00
max       315.00
Name: Frequency, dtype: float64
Median of 'Frequency': 6.0
Mode of 'Frequency': 1
```

Figure 4.10 Descriptive statistics of frequency

Figure 4.10, the dataset comprises 15,399 data points, reflecting the frequency of interactions, and the average (mean) interaction frequency stands at approximately 11.38, indicating an average of 11.38 transactions per customer. The data exhibits a wide range, with the lowest interaction frequency being 1 and the highest reaching 315. The median interaction frequency, which is the middle value when all data points are arranged in ascending order is 6, implying that half of the customers have bought approximately six times or less. The mode, representing the most frequently occurring value, is 1, indicating that a notable proportion of customers have bought only once.

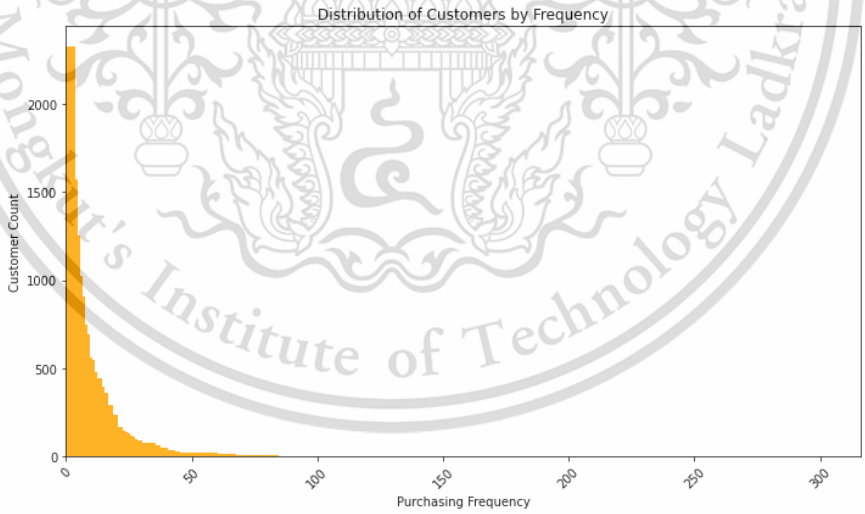


Figure 4.11 Distribution of customers by purchasing frequency

Figure 4.11, The graph portraying the distribution of customers by their interaction frequency reveals a distinctly right-skewed pattern. Within this distribution, the majority of customers are situated on the left side, indicating that a significant proportion of customers

have relatively low transaction frequencies. This suggests that most customers engage with the business infrequently, as evidenced by the concentration of data points towards the lower end of the frequency spectrum. However, there are comparatively fewer customers on the right side of the graph, representing a smaller subset of customers who exhibit significantly higher transaction frequencies. This right-skewed distribution implies that while the majority of customers buy from the company sparingly, there is a notable segment of highly engaged customers who interact with the business frequently.

4.2.3 Monetary

How much money a customer spends. Customers who spend more are typically more valuable to a business. The monetary value is calculated based on how much a customer spend during the study period. The higher a customer spent, the higher the monetary value.

```
monetary_df = df.groupby(by='SOLDTO_NUMBER', as_index=False)['SALES_AMOUNT'].sum()
monetary_df.columns = ['CustomerName', 'Monetary']
monetary_df.head()
```

	CustomerName	Monetary
0	110000011	59587.64
1	110000042	7900.00
2	110000045	100502.74
3	110000050	684.00
4	110000059	46695.26

Figure 4.12 Monetary code and table

Figure 4.12, the code begins by creating a new DataFrame called monetary_df. It does so by grouping the original DataFrame df by a column 'SOLDTO_NUMBER'. This grouping operation aggregates data for each unique 'SOLDTO_NUMBER'. Within each customer group, the code calculates the sum of 'SALES_AMOUNT' using the '.sum()' method. This step totals the spending amounts for each customer, representing their overall monetary value to the company.

```
count     15399.00
mean      32970.16
std       61313.82
min      -41469.84
25%        4429.50
50%       13452.00
75%       36009.00
max     1689109.44
Name: Monetary, dtype: float64
Median of 'Monetary': 13452.0
Mode of 'Monetary': 0.0
```

Figure 4.13 Descriptive statistics of monetary

Figure 4.13, The dataset comprises 15,399 data points, reflecting the total monetary value of transactions for each customer. The average (mean) monetary value stands at approximately 32,970.16 Baht, indicating an average transaction value of this amount across all customers. The dataset exhibits considerable variability, with the lowest monetary value recorded as -41,469.84 Baht, which suggest instances of refunds or credit notes. The median monetary value is 13,452 Baht, signifying that half of the customers have transaction values equal to or less than this amount. The mode is 0.0, suggesting that a notable portion of customers has transactions with zero monetary value.

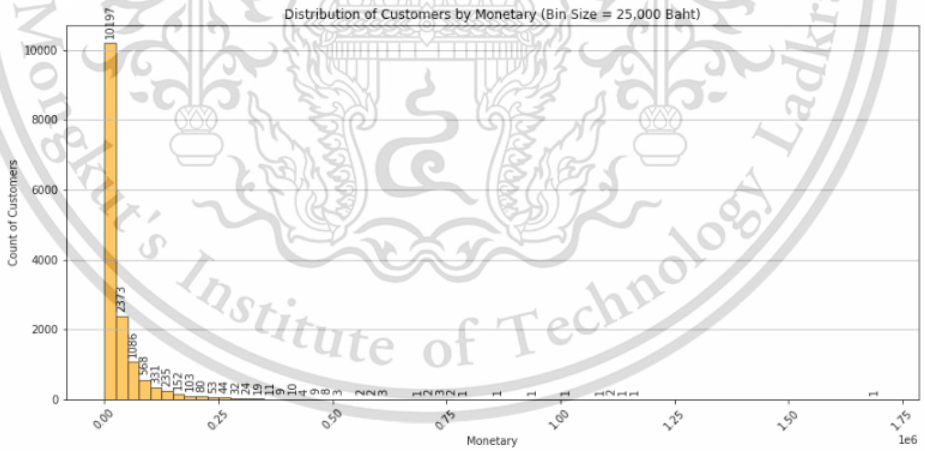


Figure 4.14 Distribution of customers by monetary

Figure 4.14, the graph depicting the distribution of customers by their monetary spending patterns illustrates a heavily right-skewed distribution. Within this distribution, most customers are situated on the left side, indicating that a significant proportion of customers engage in transactions with relatively lower monetary values. This suggests that

most customers have transaction values clustered towards the lower end of the monetary spectrum. Conversely, there are fewer customers on the right side of the graph, representing a smaller subset of customers who engage in significantly higher-value transactions. This right-skewed distribution implies that while a substantial portion of customers conducts transactions with relatively modest values, there is a notable segment of high-value customers who contribute significantly to the overall monetary value.

4.2.4 Section Variety

The number of product segments that customer purchase. The higher the number indicates that a customer buy more variety of products from the company during the study period. Company prefers customers who purchase multiple product segments as this can be an indicator that the company could be the main office supplier of the customers.

```
product_df = df.groupby(by='SOLDTO_NUMBER', as_index=False)['SECTION_CODE'].nunique()
product_df.columns = ['CustomerName', 'Section Bought']
product_df.head()
```

	CustomerName	Section Bought
0	110000011	14
1	110000042	1
2	110000045	14
3	110000050	2
4	110000059	12

Figure 4.15 Section variety code and table

Figure 4.15, the code create a new DataFrame called product_df. It does so by grouping the original DataFrame df by column 'SOLDTO_NUMBER'. This grouping operation aggregates data for each unique 'SOLDTO_NUMBER'. Within each customer group, the code calculates the number of unique 'SECTION_CODE' values using the '.nunique()' method. This step effectively counts how many different product sections each customer has bought from. In other words, it measures the diversity of product categories or sections purchased by each customer.

```
count   15399.00
mean        6.70
std         4.43
min         1.00
25%         3.00
50%         6.00
75%        10.00
max        17.00
Name: Section Bought, dtype: float64
Median of 'Section Bought': 6.0
Mode of 'Section Bought': 1
```

Figure 4.16 Descriptive statistics of section variety

Figure 4.16, The dataset encompasses 15,399 data points, reflecting the number of distinct product sections each customer has purchased from. On average, customers have engaged with approximately 6.70 unique product sections, suggesting a moderate level of product diversity among customers. The dataset exhibits a standard deviation of 4.43, indicating some variability in the number of sections customers have bought from. The minimum value of 1 indicates that there are customers who have purchased from just one product section, while the maximum value of 17 indicates a few customers who have engaged with a wide array of product sections. The median is 6, signifying that half of the customers have purchased from six or fewer product sections. The mode is 1, suggesting that a notable portion of customers has purchased from just one product section.

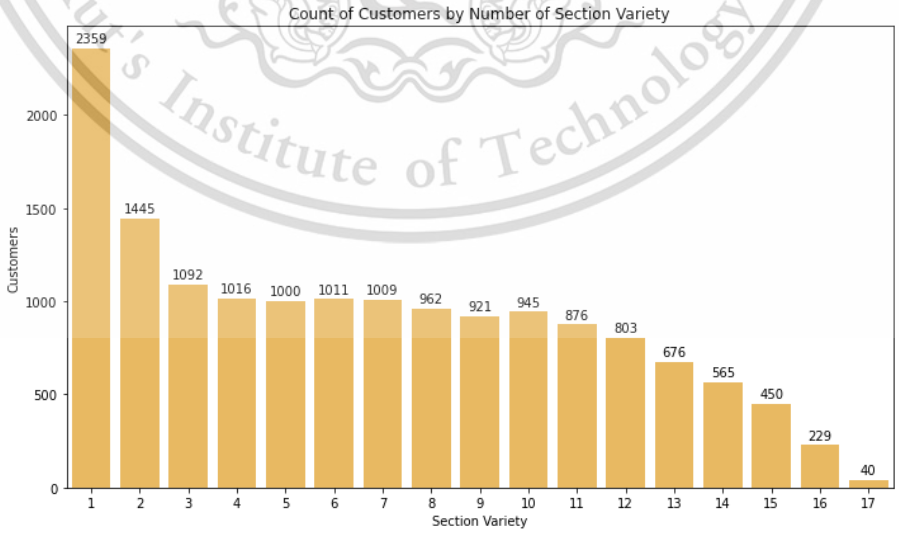


Figure 4.17 Distribution of customers by section variety

Figure 4.17, The bar chart illustrates the distribution of customers based on the number of product sections they have purchased from. This data, collected from 15,399 customers, showcases a diverse pattern of customer purchasing behaviors. The chart reveals that the most common scenario is customers buying from just one product section, with a count of 2,359 customers falling into this category. As the number of sections bought increases, the count gradually decreases, illustrating that fewer customers diversify their purchases across multiple sections. Interestingly, there is a notable drop in the number of customers who have purchased from 17 different sections, which indicates a small, highly diversified segment of customers.

4.2.5 Normalization

RFM analysis relies on three fundamental metrics: Recency, Frequency, and Monetary. These metrics capture essential aspects of customer engagement, such as how recently a customer made a purchase, how frequently they do so, and the monetary value of their transactions. However, the utility of RFM analysis extends beyond the mere calculation of these metrics; it hinges on the process of data normalization. Data normalization is a transformative step that renders the RFM metrics comparable, ensuring that each contributes meaningfully to segmentation and marketing basket analysis.

After the RFM calculation and segment bought, all 4 tables are merged together on ‘CustomerName’.

```
rfms_df = recency_df.merge(frequency_df, on='CustomerName') \
          .merge(monetary_df, on='CustomerName').drop(columns='LastPurchaseDate') \
          .merge(product_df, on='CustomerName')
rfms_df.head()
```

	CustomerName	Recency	Frequency	Monetary	Section Variety
0	110000011	15	31	59587.64	14
1	110000042	165	2	7900.00	1
2	110000045	1	117	100502.74	14
3	110000050	284	1	684.00	2
4	110000059	28	39	46695.26	12

Figure 4.18 Merging DataFrame

Figure 4.18, the code merge the recency_df, frequency_df, monetary_df and product_df DataFrames using the '.merge()' method with the 'CustomerName' column as the common key. This step combines the information of customers into a single DataFrame. The code also uses the '.drop()' method to remove the 'LastPurchaseDate' column from the intermediate DataFrame. This column was redundant or not needed in the final analysis. The resulting DataFrame, rfms_df, now contains customer names, recency, frequency, monetary, and product diversity data, providing a comprehensive overview of customer behavior.

After merging all the DataFrames, the variables then normalized by min-max scaling. The purpose of this normalization method is to transform data so that it falls within a consistent and standardized range, making it easier to compare and analyze different variables or datasets.

```
rfms_df['R_rank'] = rfms_df['Recency'].rank(ascending=False)
rfms_df['F_rank'] = rfms_df['Frequency'].rank(ascending=True)
rfms_df['M_rank'] = rfms_df['Monetary'].rank(ascending=True)
rfms_df['S_rank'] = rfms_df['Section Variety'].rank(ascending=True)


rfms_df['R_rank_norm'] = (rfms_df['R_rank']/rfms_df['R_rank'].max())*100
rfms_df['F_rank_norm'] = (rfms_df['F_rank']/rfms_df['F_rank'].max())*100
rfms_df['M_rank_norm'] = (rfms_df['M_rank']/rfms_df['M_rank'].max())*100
rfms_df['S_rank_norm'] = (rfms_df['S_rank']/rfms_df['S_rank'].max())*100

rfms_df.drop(columns=['R_rank', 'F_rank', 'M_rank', 'S_rank'], inplace=True)
rfms_df.drop(columns=['Recency', 'Frequency', 'Monetary', 'Section Variety'], inplace=True)

rfms_df = rfms_df.rename(columns={'R_rank_norm': 'Recency', 'F_rank_norm': 'Frequency',\
                                  'M_rank_norm': 'Monetary', 'S_rank_norm': 'Section Variety'})

rfms_df.head()
```

	CustomerName	Recency	Frequency	Monetary	Section Variety
0	110000011	72.27	92.53	84.94	93.62
1	110000042	18.83	20.20	37.29	7.67
2	110000045	97.23	99.70	92.56	93.62
3	110000050	7.37	7.55	3.65	20.04
4	110000059	56.52	95.36	80.54	84.78

Figure 4.19 Normalizing the variables

Figure 4.19, a series of operations are performed on the rfms_df DataFrame to derive normalized rankings based on the 'Recency' column. The process begins by

calculating the rank of each 'Recency' value in descending order using the '.rank()' method with ascending=False. This results in a new column named 'R_rank,' where each entry represents the rank of the corresponding 'Recency' value. Lower 'Recency' values receive higher ranks, reflecting their recency in customer transactions. To make these rank values more interpretable and universally applicable, the code proceeds to create another new column called 'R_rank_norm.' In this column, the previously calculated 'R_rank' values are normalized by dividing each rank value by the maximum rank present in the 'R_rank' column. This normalization operation scales the rank values to a percentage scale ranging from 0 to 100. This transformation facilitates a more intuitive understanding of the relative recency of customer transactions, making it easier to interpret and use for subsequent analyses. Finally, the original 'R_rank' and the original 'Recency' column is dropped using the '.drop()' method and 'R_rank_norm.' is renamed to 'Recency' to simplify the next process. The rest of the variables are done the same way except the order is rank normally.

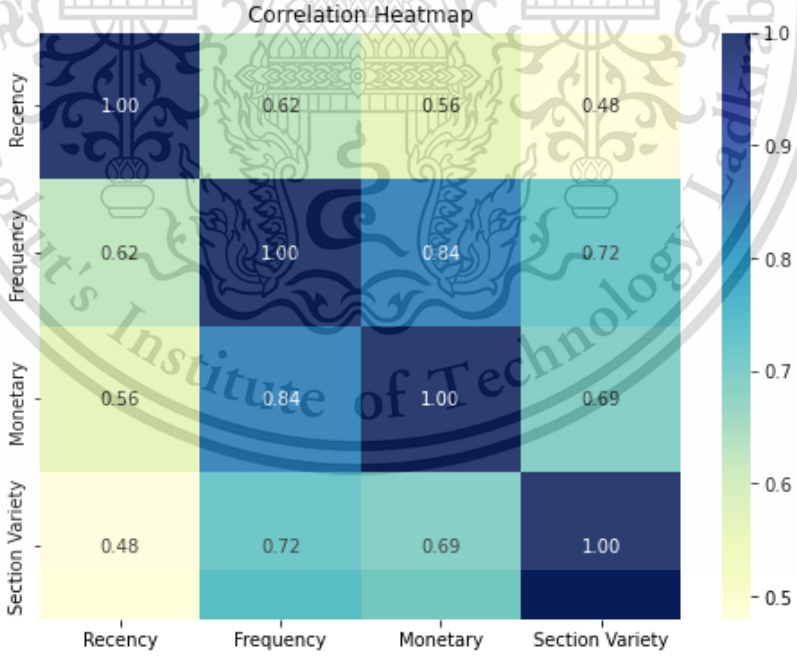


Figure 4.20 Variables correlation heatmap

Figure 4.20, the resulting heatmap provides a visual representation of the correlations between the RFM metrics in the dataset. Strong and positive correlations are displayed in blue color gradient while weaker correlations are displayed in yellow color. The annotated values within each cell indicate the strength and direction of the correlation between the corresponding variables. The correlation analysis indicates that Recency, Frequency, Monetary, and Section Variety metrics are interrelated, providing valuable insights for customer segmentation and marketing strategies. Understanding these relationships can help in identifying customer segments, targeting recent and frequent customers for revenue growth, and tailoring product offerings to suit different customer preferences and behaviors.

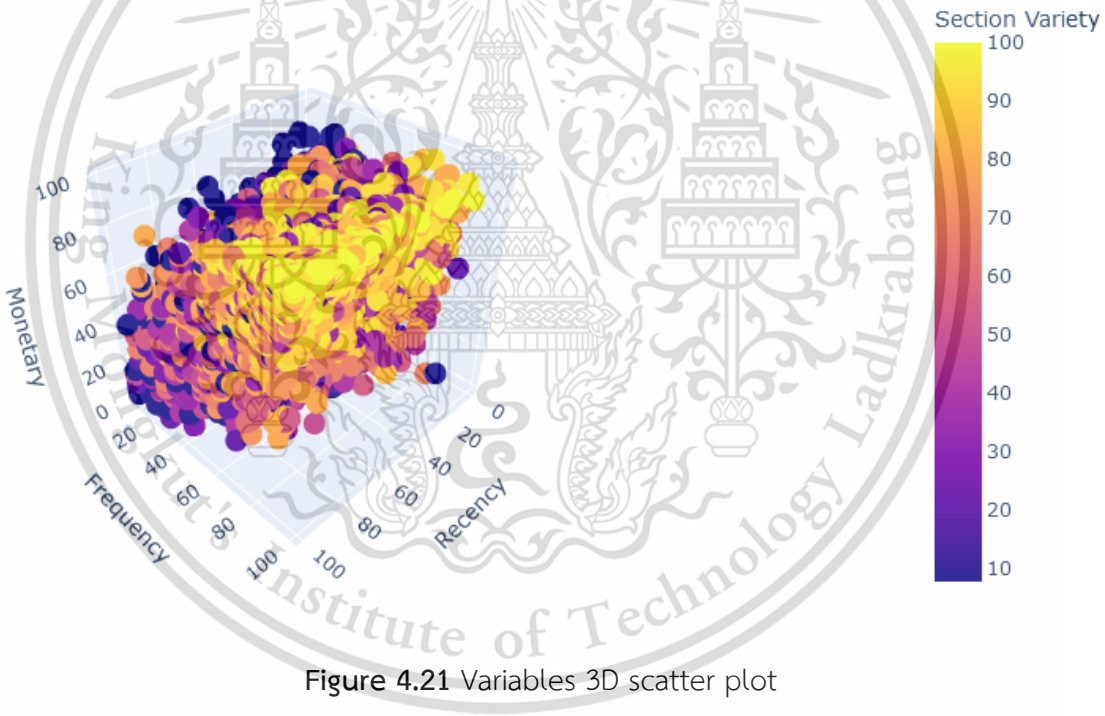


Figure 4.21 Variables 3D scatter plot

Figure 4.21, the 3D scatter plot allows for the multidimensional representation of the dataset and the simultaneous representation of three numeric variables—'Recency,' 'Frequency,' and 'Monetary'—while adding a color dimension to represent number of 'Section Variety.'

4.3 Customer Segmentation

Understanding customers and tailoring marketing strategies to their unique needs and behaviors has become paramount for organizations seeking to thrive and excel. This holds especially true for Business-to-Business (B2B) enterprises, where the complexities of catering to a diverse clientele of corporate clients necessitate a granular and nuanced approach to customer engagement. In the context of B2B sales, customer segmentation emerges as a critical strategy that enables companies to categorize their clients based on common attributes, preferences, and purchase behaviors. Through segmentation, organizations can unlock insights into their customer base, identify high-value segments, and craft customized marketing initiatives that resonate with the unique demands of each group.

K-means clustering stands out as a versatile and widely embraced method. According to Gomes and Meisen (2023) - A review on customer segmentation methods for personalized customer targeting in e-commerce use cases, k-means is the most frequently used customer segmentation method in their surveyed literature (41 of 105) during 2000 and 2022. The goal of the k-means algorithm is to partition a set of data points into k segments which minimize the distance between the data. The popularity of k-means can be explained by its simplicity and applicability to large scale datasets.

Before using k-means clustering, the appropriate number of clusters needs to be identified, this is done by running multiple iterations of different number of clusters then use elbow method to identify the appropriate number of clusters for the study dataset.

The elbow method is a heuristic used to determine the optimal number of clusters (K) for a K-means clustering algorithm. It helps in finding the "elbow point" in a plot of the within-cluster sum of squares (WCSS) as a function of the number of clusters. WCSS measures the compactness or dispersion of data points within each cluster. The idea behind the elbow method is to identify the point at which increasing the number of clusters does not significantly reduce the WCSS, resembling an "elbow" in the plot.

```
X=rfms_df.drop("CustomerName",axis=1)

k_values = range(1,11)
scores = []
for k in k_values:
  kmeans = KMeans(n_clusters=k,random_state=42)
  kmeans.fit(X)
  scores.append(kmeans.inertia_)

plt.plot(k_values,scores,'o-')
plt.xticks(k_values)
plt.show()
```

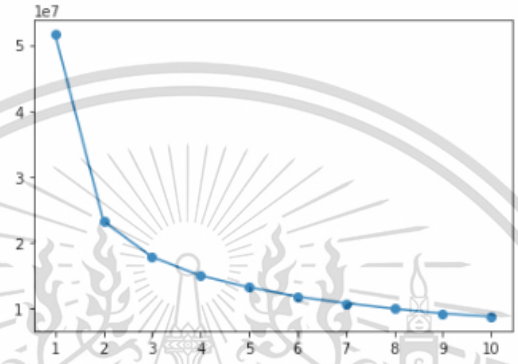


Figure 4.22 K-Means iterations

Figure 4.22, the implementation of the elbow method for determining the optimal number of clusters (K) in a K-means clustering algorithm. This method helps in selecting the most suitable K value by analyzing the within-cluster sum of squares (WCSS) as a function of different K values. The code begins by initializing the range of K values to be evaluated, the WCSS represents the sum of squared distances between data points and their assigned cluster centroids. Once the loop has iterated through all the K values and collected their corresponding WCSS scores. It creates a line plot with K values on the x-axis and the WCSS scores on the y-axis.

```
Kneedle = KneeLocator(k_values,scores, curve='convex',direction='decreasing')
print("Optimal Number of Clusters:",Kneedle.elbow)
Kneedle.plot_knee()
```

```
Optimal Number of Clusters: 3
```

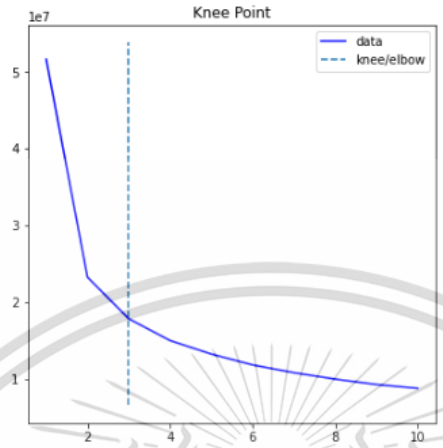


Figure 4.23 Knee Point locator for elbow method

Figure 4.23, the code snippet further refines the process of identifying the optimal number of clusters (K) using the elbow method by employing the 'KneeLocator' class. This class helps to automatically locate the "elbow point" in the curve of within-cluster sum of squares (WCSS) values, simplifying the selection of K. The code starts by initializing the 'KneeLocator' object, which is a utility for detecting the knee or elbow point in a curve. It is part of the 'kneed' library. Once the 'KneeLocator' object is initialized, it automatically identifies the knee or elbow point using the specified curve analysis and direction, the elbow point is where the WCSS reduction rate starts to slow down significantly. The elbow point corresponds to the optimal K value that strikes a balance between model complexity and cluster quality.

In this case, it suggests that 3 clusters is the most suitable choice. The convex decreasing curve indicates that the decrease in scores becomes less significant after 3 clusters, suggesting that adding more clusters doesn't provide much improvement in clustering quality. Therefore, 3 clusters can be considered the optimal number.

In addition to utilizing the elbow method through the 'KneeLocator' function, the author of this research also sought input from domain experts, including the company's

director of sales and director of marketing, to determine the optimal number of clusters for their study. This collaborative approach incorporated valuable insights from experts intimately familiar with the research topic and its real-world implications. Remarkably, the consensus reached by these experts aligns with the elbow method's findings, further validating the choice of 3 clusters as the most suitable grouping for the data. This convergence between data-driven analysis and expert opinion strengthens the credibility and robustness of the selected clustering solution, demonstrating a holistic approach to decision-making in this research endeavor.

Once the optimal number of clusters has been determined using the elbow method, the next step in the K-means clustering process involves the actual clustering of the data into the specified number of clusters. This is achieved by running the K-means algorithm with the pre-determined number of clusters as a parameter. The algorithm iteratively assigns each data point to the nearest cluster center and then recalculates the cluster centers based on the newly assigned data points. This process continues until convergence, which occurs when the cluster assignments no longer change significantly or a predefined number of iterations is reached. After convergence, each data point will belong to one of the identified clusters, and the centroids of these clusters represent the center points of the clusters. This step effectively segments the data into meaningful and homogeneous groups.

```
kmeans = KMeans(n_clusters=3, init="k-means++",\
        n_init=10, tol=1e-04, random_state=42)

kmeans.fit(X)
```

```
KMeans(n_clusters=3, random_state=42)
```

Figure 4.24 K-Means clustering for 3 clusters

Figure 4.24, this code snippet is a Python script using the scikit-learn library to perform K-Means clustering on a dataset represented by the variable ‘X’. K-Means is an unsupervised machine learning algorithm used for clustering data points into a specified

number of clusters, which in this case is set to 3 (n_clusters=3). The ‘init’ parameter is set to "k-means++," which is a smart initialization method for selecting the initial cluster centroids to help speed up convergence. The ‘n_init’ parameter determines the number of times K-Means will be run with different initializations, and ‘tol’ sets the tolerance for convergence. The ‘random_state’ parameter ensures reproducibility of the results by seeding the random number generator. After configuring these parameters, the K-Means algorithm is fit to the dataset ‘X’ using ‘kmeans.fit(X)’, and it will assign each data point to one of the three clusters based on their similarity, with the goal of minimizing the within-cluster variance.

```
clusters=pd.DataFrame(X,columns=rfms_df.drop("CustomerName",axis=1).columns)
clusters['label']=kmeans.labels_
clusters["label"] = clusters["label"].replace \
({0: "Medium Value Customers", 1: "Low Value Customers", 2: "High Value Customers"})
clusters.head()
```

	Recency	Frequency	Monetary	Section Variety	label
0	72.27	92.53	84.94	93.62	High Value Customers
1	18.83	20.20	37.29	7.67	Low Value Customers
2	97.23	99.70	92.56	93.62	High Value Customers
3	7.37	7.55	3.65	20.04	Low Value Customers
4	56.52	95.36	80.54	84.78	High Value Customers

Figure 4.25 DataFrame result of K-Means algorithm

Figure 4.25, post-processing the results of the K-Means clustering performed earlier. It creates a Pandas DataFrame called clusters to store the data points from the original dataset X. Each column in this DataFrame corresponds to one of the features in the dataset, and it is labeled accordingly. Next, the K-Means cluster labels obtained from the previous step are added as a new column called 'label' in the clusters DataFrame. To make these labels more interpretable, a dictionary-based replacement is used to map the numeric cluster labels (0, 1, 2) to descriptive labels such as "Medium Value Customers", "Low Value Customers" and "High Value Customers" This step provides a more intuitive understanding of which cluster each data point belongs to.

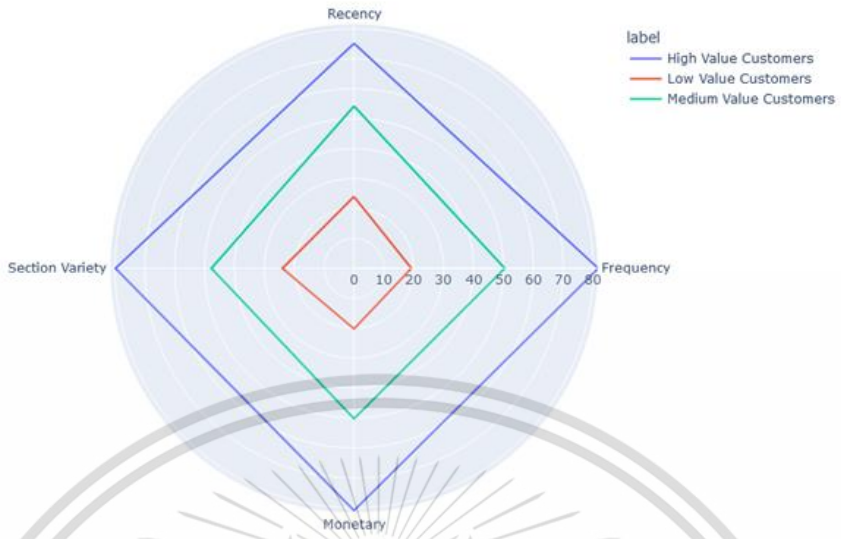


Figure 4.26 Polar plot for variables

Figure 4.26, a polar plot using Plotly Express to visualize the average feature values within different clusters where the radial axis ('r') represents the mean feature values, the angular axis ('theta') represents the features, and different cluster labels are color-coded. This plot offers an intuitive way to compare how "Medium Value Customers," "Low Value Customers," and "High Value Customers" differ in terms of their average feature values, providing insights into cluster characteristics.

Values of variables are as follow:

Table 4.3 Means of Variables of each customer cluster

	High Value Cluster	Medium Value Cluster	Low Value Cluster
Recency	75.05	54.47	23.96
Frequency	81.74	50.51	19.34
Monetary	80.90	50.31	20.33
Section Variety	79.85	41.72	24.17

Table 4.3 shows a table of means of recency, frequency, monetary and section variety variables for each of customer cluster. Notice a sizable difference of every

variable means from each cluster where high value customer cluster has the highest means in every variable, low value customer cluster has the lowest means and medium value customer cluster has means value between the high and low group.

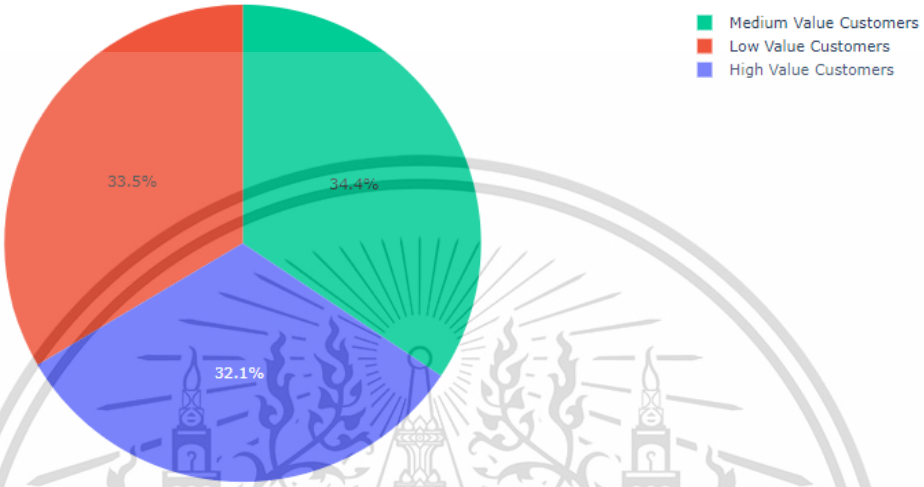


Figure 4.27 Pie chart distribution of customer clusters

Figure 4.27, a pie chart visualization using Plotly Express to illustrate the distribution of data points across different customer value categories. In the resulting pie chart, each slice represents a cluster label, and its size is proportional to the number of data points in that cluster category. The color scheme applied is blue, green, and red, which corresponds to 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers' respectively. This visualization offers a clear and concise view of the distribution of customers among these value categories, aiding in understanding the composition of the dataset in terms of customer value.

Number of customers are as follow:

- High Value Customer: 32.1% or 4,946 customers
- Medium Value Customer: 34.4% or 5,292 customers
- Low Value Customer: 33.5% or 5,161 customers

```
rfms_df_with_3cluster = rfms_df.assign(cluster=clusters['label'])
rfms_df_with_3cluster.head()
```

	CustomerName	Recency	Frequency	Monetary	Section Variety	cluster
0	110000011	72.27	92.53	84.94	93.62	High Value Customers
1	110000042	18.83	20.20	37.29	7.67	Low Value Customers
2	110000045	97.23	99.70	92.56	93.62	High Value Customers
3	110000050	7.37	7.55	3.65	20.04	Low Value Customers
4	110000059	56.52	95.36	80.54	84.78	High Value Customers

Figure 4.28 Label customer cluster back into DataFrame

Figure 4.28, the provided code snippet augments an existing DataFrame called ‘rfms_df’ with an additional column labeled 'cluster' obtained from the clusters DataFrame. This new 'cluster' column stores the cluster labels assigned to each data point during the K-Means clustering process, indicating whether a customer belongs to the "High Value Customers", "Medium Value Customers" or "Low Value Customers" category. The ‘.assign()’ method adds this column to the original ‘rfms_df’.

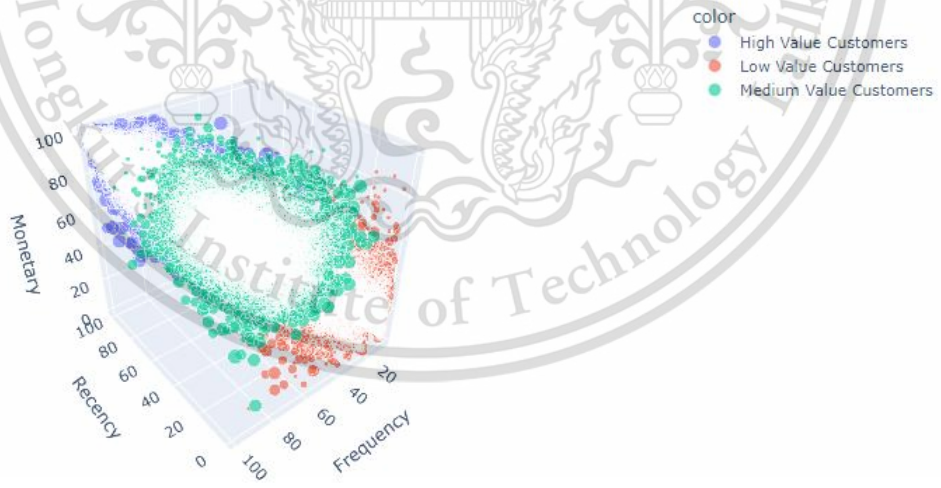


Figure 4.29 3D scatter plot of different clusters

Figure 4.29, a 3D scatter plot using Plotly Express to visualize the relationships between four features: 'Recency', 'Frequency', 'Monetary,' and 'Section Variety' while also

incorporating cluster information from the K-Means clustering. Each data point in the scatter plot represents a customer, with their position in the 3D space determined by their 'Recency', 'Frequency' and 'Monetary' values. The 'Section Variety' feature is represented by the size of each data point, providing an additional dimension of information. Furthermore, the color of each data point is determined by the cluster label assigned during K-Means clustering ('High Value Customers', 'Medium Value Customers' or 'Low Value Customers'). This 3D scatter plot provides a multi-dimensional view of customer characteristics and how they relate to the cluster assignments. It allows for the exploration of potential patterns or groupings in the data, which can be valuable for understanding customer behavior and tailoring marketing or business strategies to different customer segments.

4.4 Analysis of Variance (ANOVA) and Post Hoc tests

Segmentation is a valuable tool for businesses seeking to boost sales and optimize resource allocation. However, while segmentation through methods like K-means clustering provides an initial framework for categorizing customers, it is crucial to delve deeper into the characteristics of these segments. This chapter focuses on a pivotal aspect of such an analysis: the application of Analysis of Variance (ANOVA) and Post Hoc tests to validate and interpret the significance of the identified customer segments. In the context of our case study with an office supplier company, this exploration sheds light on the effectiveness of the segmentation strategy and identifies specific variables that significantly impact purchasing behaviors among B2B clients.

ANOVA (Analysis of Variance) and post hoc tests are statistical techniques used after K-means clustering to evaluate the significance of differences between clusters and to perform pairwise comparisons between clusters, respectively.

ANOVA helps assess whether the clusters generated by K-means are statistically significant. If ANOVA reveals significant differences in the means of the variables among the clusters, it suggests that the clustering has effectively grouped data points based on

those variables. In essence, it validates that the clusters are not randomly assigned but are distinct and meaningful groups within the data.

ANOVA can identify which variables that are most influential in distinguishing between clusters. Variables with low p-values in the ANOVA test indicate that they contribute significantly to the differences observed between clusters. This information can guide feature selection and refinement of the clustering model.

When ANOVA detects significant differences in means, post hoc tests like Tukey's Honestly Significant Difference (HSD) can be applied. These tests provide pairwise comparisons between clusters, revealing which specific clusters differ significantly from each other in terms of the selected variables. This information is valuable for understanding the specific characteristics that differentiate one cluster from another.

ANOVA and post hoc tests provide deeper insights into the characteristics of each cluster. They help answer questions like which clusters are responsive to specific variables or which clusters represent high-value or low-value customer segments. This understanding is essential for tailoring marketing strategies, product offerings, and other business decisions to specific customer segments.

By assessing the statistical significance of clusters and variables, ANOVA and post hoc tests can help evaluate the quality of the clustering model. Poorly defined or unstable clusters may result in non-significant ANOVA results, indicating the need for refining the clustering approach.

```python
for variable in variables:
    print(f"ANOVA for {variable}:")

    # Get data for each cluster
    cluster_data = [df[df['cluster'] == cluster][variable] for cluster in df['cluster'].unique()]

    # Perform ANOVA
    f_statistic, p_value = f_oneway(*cluster_data)
    print(f"F-statistic: {f_statistic:.4f}")
    print(f"P-value: {p_value:.4f}")

    if p_value < 0.05:
        print("Reject null hypothesis: There is a significant difference in means.")

        # Perform Tukey's HSD post hoc test
        posthoc = pairwise_tukeyhsd(df[variable], df['cluster'], alpha=0.05)
        print(posthoc)
    else:
        print("Fail to reject null hypothesis: No significant difference in means.")

    print("=" * 40)
```

Figure 4.30 ANOVA and Tukey's HSD post hoc test

Figure 4.30, the Python code snippet conducts an Analysis of Variance (ANOVA) to investigate the statistical differences in a variable, referred to as 'Recency', 'Frequency', 'Monetary' and 'Section Variety' across three distinct customer clusters: 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers'. This analysis aims to determine whether there are significant variations in the mean values of the variables among these customer segments. The code iterates through each variable and prints out the results of the ANOVA, including the F-statistic and associated p-value. If the p-value is less than 0.05, indicating statistical significance, the code proceeds to perform Tukey's Honestly Significant Difference (HSD) post hoc test to identify which specific clusters exhibit significantly different means. Conversely, if the p-value is greater than or equal to 0.05, the code concludes that there are no significant differences in means across the clusters.

4.4.1 Recency

This analysis aims to determine whether there are significant variations in the mean values of the 'Recency' variable among three customer clusters.

```
ANOVA for Recency:
F-statistic: 7923.7225
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
               Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================================================
       group1                group2         meandiff p-adj  lower    upper   reject
---------------------------------------------------------------------------------
High Value Customers    Low Value Customers -51.1937  -0.0 -52.1535 -50.2339   True
High Value Customers Medium Value Customers -20.9988  -0.0 -21.9528 -20.0449   True
 Low Value Customers Medium Value Customers  30.1949  -0.0  29.2513  31.1385   True
---------------------------------------------------------------------------------
```

Figure 4.31 ANOVA and Tukey's HSD post hoc test for Recency

Figure 4.31, the output reveals the results of conducting an Analysis of Variance (ANOVA) on the 'Recency' variable, segmented across three distinct customer clusters: 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers'.

The ANOVA test has yielded a highly significant result with F-statistic of 7923.7225 and an extremely low P-value of 0.0000. This implies that there is strong evidence to reject the null hypothesis. In other words, there is a statistically significant difference in the mean 'Recency' values among the three customer clusters.

The Tukey's Honestly Significant Difference (HSD) post hoc test has been performed to determine which specific pairs of customer clusters exhibit significant differences in their 'Recency' means. This test corrects for multiple comparisons.

The results of the Tukey's HSD test indicate that all pairwise comparisons among the customer clusters have extremely low adjusted p-values (close to 0.0). This signifies that there are statistically significant differences in the mean 'Recency' values between all pairs of customer clusters.

In conclusion, based on the ANOVA and Tukey's HSD test results, it is evident that the 'Recency' variable has a significant impact on customer segmentation. The three customer clusters, categorized as 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers' exhibit distinct and statistically significant differences in their 'Recency' means.

4.4.2 Frequency

This analysis aims to determine whether there are significant variations in the mean values of the 'Frequency' variable among three customer clusters.

```
ANOVA for Frequency:
F-statistic: 26320.1243
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
               Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================================================
       group1               group2         meandiff p-adj  lower    upper   reject
---------------------------------------------------------------------------------
High Value Customers    Low Value Customers -62.5068  -0.0 -63.1455 -61.8682   True
High Value Customers Medium Value Customers -31.2573  -0.0 -31.8921 -30.6226   True
 Low Value Customers Medium Value Customers  31.2495  -0.0  30.6216  31.8774   True
---------------------------------------------------------------------------------
```

Figure 4.32 ANOVA and Tukey's HSD post hoc test for Frequency

Figure 4.32, the output reveals the results of conducting an Analysis of Variance (ANOVA) on the 'Frequency' variable, segmented across three distinct customer clusters: 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers'.

The ANOVA test has yielded a highly significant result with F-statistic of 26320.1243 and an extremely low P-value of 0.0000. This implies that there is strong evidence to reject the null hypothesis. In other words, there is a statistically significant difference in the mean 'Frequency' values among the three customer clusters.

The Tukey's Honestly Significant Difference (HSD) post hoc test has been performed to determine which specific pairs of customer clusters exhibit significant differences in their 'Frequency' means. This test corrects for multiple comparisons.

The results of the Tukey's HSD test indicate that all pairwise comparisons among the customer clusters have extremely low adjusted p-values (close to 0.0). This signifies that there are statistically significant differences in the mean 'Frequency' values between all pairs of customer clusters.

In conclusion, based on the ANOVA and Tukey's HSD test results, it is evident that the 'Frequency' variable has a significant impact on customer segmentation. The three

customer clusters, categorized as 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers' exhibit distinct and statistically significant differences in their 'Frequency' means.

4.4.3 Monetary

This analysis aims to determine whether there are significant variations in the mean values of the 'Monetary' variable among three customer clusters.

```
ANOVA for Monetary:
F-statistic: 20178.6296
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
             Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================================================
       group1              group2        meandiff p-adj  lower    upper   reject
---------------------------------------------------------------------------------
High Value Customers    Low Value Customers -60.6444  -0.0  -61.352 -59.9368   True
High Value Customers Medium Value Customers -30.6572  -0.0 -31.3605 -29.9539   True
 Low Value Customers Medium Value Customers  29.9871  -0.0  29.2915  30.6828   True
---------------------------------------------------------------------------------
```

Figure 4.33 ANOVA and Tukey's HSD post hoc test for Monetary

Figure 4.33, the output reveals the results of conducting an Analysis of Variance (ANOVA) on the 'Monetary' variable, segmented across three distinct customer clusters: 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers'.

The ANOVA test has yielded a highly significant result with F-statistic of 20178.6296 and an extremely low P-value of 0.0000. This implies that there is strong evidence to reject the null hypothesis. In other words, there is a statistically significant difference in the mean 'Monetary' values among the three customer clusters.

The Tukey's Honestly Significant Difference (HSD) post hoc test has been performed to determine which specific pairs of customer clusters exhibit significant differences in their 'Monetary' means. This test corrects for multiple comparisons.

The results of the Tukey's HSD test indicate that all pairwise comparisons among the customer clusters have extremely low adjusted p-values (close to 0.0). This signifies

that there are statistically significant differences in the mean 'Monetary' values between all pairs of customer clusters.

In conclusion, based on the ANOVA and Tukey's HSD test results, it is evident that the 'Monetary' variable has a significant impact on customer segmentation. The three customer clusters, categorized as 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers' exhibit distinct and statistically significant differences in their 'Monetary' means.

4.4.4 Section Variety

This analysis aims to determine whether there are significant variations in the mean values of the 'Section Variety' variable among three customer clusters.

```
ANOVA for Section Bought:
F-statistic: 12570.4659
P-value: 0.0000
Reject null hypothesis: There is a significant difference in means.
          Multiple Comparison of Means - Tukey HSD, FWER=0.05
=============================================================================
      group1                 group2          meandiff p-adj  lower    upper   reject
-----------------------------------------------------------------------------
High Value Customers    Low Value Customers -55.8514  -0.0 -56.6793 -55.0235   True
High Value Customers Medium Value Customers -32.0818  -0.0 -32.9047 -31.2589   True
 Low Value Customers Medium Value Customers  23.7696  -0.0  22.9556  24.5836   True
-----------------------------------------------------------------------------
```

Figure 4.34 ANOVA and Tukey's HSD post hoc test for Section Variety

Figure 4.34, the output reveals the results of conducting an Analysis of Variance (ANOVA) on the 'Section Variety' variable, segmented across three distinct customer clusters: 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers'.

The ANOVA test has yielded a highly significant result with F-statistic of 12570.4659 and an extremely low P-value of 0.0000. This implies that there is strong evidence to reject the null hypothesis. In other words, there is a statistically significant difference in the mean 'Section Variety' values among the three customer clusters.

The Tukey's Honestly Significant Difference (HSD) post hoc test has been performed to determine which specific pairs of customer clusters exhibit significant differences in their ‘Section Variety’ means. This test corrects for multiple comparisons.

The results of the Tukey's HSD test indicate that all pairwise comparisons among the customer clusters have extremely low adjusted p-values (close to 0.0). This signifies that there are statistically significant differences in the mean 'Monetary' values between all pairs of customer clusters.

In conclusion, based on the ANOVA and Tukey's HSD test results, it is evident that the 'Section Variety' variable has a significant impact on customer segmentation. The three customer clusters, categorized as 'High Value Customers', 'Medium Value Customers' and 'Low Value Customers' exhibit distinct and statistically significant differences in their 'Section Variety' means.

4.5 Market Basket Analysis

Building upon the foundations of our customer segmentation findings, we now transition to a data-driven exploration of Market Basket Analysis (MBA). MBA is a crucial analytical approach in understanding how distinct customer segments interact with product assortments. It uncovers associations and patterns in product co-purchases, facilitating the formulation of precision-targeted marketing strategies.

We have meticulously segmented our customer base into three distinct clusters: ‘High Value Customers’, ‘Medium Value Customers’ and ‘Low Value Customers’. Each cluster represents a unique segment of B2B clientele, characterized by specific preferences, behaviors, and purchasing patterns.

We have chosen to employ three prominent association rule mining algorithms: Apriori, FP-Growth, and Association Rules. These algorithms are renowned for their ability to unearth hidden patterns within transactional data and identify product associations that hold significance for specific customer segments.

Our strategy involves applying each of these algorithms independently to the transactional data of every customer cluster. By doing so, we aim to extract tailored insights for each segment and uncover the most relevant product associations within ‘High Value Customers’, ‘Medium Value Customers’ and ‘Low Value Customers’. This approach not only allows us to pinpoint cross-selling opportunities but also aligns our marketing and sales strategies precisely with the distinct preferences of each segment.

Following the application of these algorithms to our customer clusters, our research will delve into a comprehensive comparative analysis. We will evaluate the performance of Apriori, FP-Growth, and Association Rules within the context of each cluster. This rigorous examination will provide us with invaluable insights into the algorithm's efficiency, scalability, and ability to uncover actionable patterns within different customer groups.

4.5.1 High Value Customers Cluster

The High Value Customers cluster is characterized by four key variables: Recency, Frequency, Monetary and Section Variety, each exhibiting a mean value of 75.05, 81.74, 80.90, and 79.85, respectively.

```
df_group_high = df_merged[df_merged["cluster"] == "High Value Customers"]
df_group_high.head()
```

	COUNTRY_CODE	INVOICE_NUMBER	QUANTITY	SUBCATEGORY	cluster
0	TH	2212414718	1	OTHER FIRST AID & MEDICAL EQUIPMENT(FIRST AID ...	High Value Customers
1	TH	2212416505	1	CLEAR/TRANSPARENT(ADHESIVES)	High Value Customers
2	TH	2212416505	3	CORRECTION ROLLERS & TAPES(CORRECTION PRODUCTS)	High Value Customers
3	TH	2212416505	1	ORIGINAL(LASER CARTRIDGES)	High Value Customers
4	TH	2212416505	1	DISPOSABLE CUPS, SAUCERS & GLASSES(CATERING SU...	High Value Customers

Figure 4.35 High Value Customers DataFrame

Figure 4.35, this DataFrame is a subset of the original data, consisting solely of rows where the "cluster" column matches the label ‘High Value Customers’ In essence, it filters and retains only those customers who belong to the ‘High Value Customers’ cluster, effectively isolating this particular segment from the broader dataset.

```
basket_group_high = (df_group_high[df_group_high['COUNTRY_CODE']=='TH'].groupby(['INVOICE_NUMBER','SUBCATEGORY'])\
                     ['QUANTITY'].sum().unstack().reset_index().fillna(0).set_index('INVOICE_NUMBER'))
basket_group_high.head()
```

SUBCATEGORY	#9 (4"X9") (ENVELOPES & POSTROOM)	1-HOLE PUNCHES(HOLE PUNCHES)	10"X13"(ENVELOPES & POSTROOM)	16GB(FLASH MEMORY)	2-HOLE PUNCHES(HOLE PUNCHES)	2-RING BINDERS(BINDERS)	2.5"(FLASH MEMORY)	32GB(FLASH MEMORY)
INVOICE_NUMBER								
2212399541	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399542	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399543	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399544	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399555	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 529 columns

```
row_count_high = len(basket_group_high)
print("Number of unique invoices:", row_count_high)
```

```
Number of unique invoices: 125361
```

Figure 4.36 High Value Customers Cluster Invoice Grouping

Figure 4.36, grouping the data by two key columns: "INVOICE_NUMBER" and "SUBCATEGORY." The "INVOICE_NUMBER" column represents individual purchase transactions, while "SUBCATEGORY" denotes the product subcategory associated with each item in the transaction. Within each transaction (invoice), we calculate the sum of quantities for each unique subcategory of products. The result is a table where rows represent individual transactions (invoices), columns correspond to product subcategories, and the values indicate the total quantity of products purchased from each subcategory within each transaction. To enhance the structure of this table, any missing values (i.e., cases where no products were purchased from a particular subcategory in a given transaction) are filled with zeros using the 'fillna(0)' method. The resulting dataset is then set to be indexed by "INVOICE_NUMBER.". Notably, the figure 529, situated in the bottom left corner, represents the count of unique subcategories that the 'High Value Customers' cluster purchased and the figure 125,349 represents the count of unique invoices during the specified period.

```
def encode_units(x):
    if x <= 0:
        return 0
    if x >= 1:
        return 1

basket_sets_high = basket_group_high.applymap(encode_units)
```

Figure 4.37 Encoding Function

Figure 4.37, this function is used to encode values into binary units. It checks if the input x is less than or equal to zero, and if so, it returns 0; otherwise, if x is greater than or equal to 1, it returns 1. Essentially, it converts values into binary representations: values less than or equal to zero become 0, and values greater than or equal to 1 become 1.

4.5.1.1 High Value Customers cluster – Apriori algorithm

The primary goal is to find frequent itemsets within a 'High Value Customers' dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The 'apriori' function is used to perform this task. It is typically provided by a library 'mlxtend' in Python. The runtime of the Apriori algorithm is also measured as part of an effort to assess and compare its performance against the FP-growth algorithm.

```
start_time = time.time()
frequent_itemsets_high_ap = apriori(basket_sets_high, min_support = 0.01, use_colnames=True)
end_time = time.time()
elapsed_time_high_ap = end_time - start_time

print("Elapsed time: ", elapsed_time_high_ap, " seconds")

frequent_itemsets_high_ap['unique_invoices'] = frequent_itemsets_high_ap['support'] * row_count_high
frequent_itemsets_high_ap = frequent_itemsets_high_ap[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_high_ap['size'] = frequent_itemsets_high_ap['itemsets'].apply(lambda x: len(x))

frequent_itemsets_high_ap[(frequent_itemsets_high_ap['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  8.658976078033447  seconds
```

	itemsets	support	unique_invoices	size
258	(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.037085	4649.0	2
182	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.034109	4276.0	2
281	(WASTE BAGS(WASTE MANAGEMENT), STANDARD TOILET PAPER ROLLS(WASHROOM SUPPLIES & PERSONAL CARE))	0.032634	4091.0	2
255	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.032291	4048.0	2
169	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), BULLET TIP(MARKERS))	0.032004	4012.0	2
139	(AA / LR6(BATTERIES / TORCHES / CHARGERS), AAA / LR3(BATTERIES / TORCHES / CHARGERS))	0.031206	3912.0	2
228	(FLOOR DETERGENTS(DETERGENTS), WASTE BAGS(WASTE MANAGEMENT))	0.030336	3803.0	2
130	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.029467	3694.0	2
174	(STANDARD INK(BALL PENS), BULLET TIP(MARKERS))	0.028573	3582.0	2
188	(STAPLES(STAPLING), CLEAR/TRANSPARENT(ADHESIVES))	0.026396	3309.0	2

Figure 4.38 Result of Apriori algorithm to High Value Customer cluster

Figure 4.38, ‘frequent_itemsets_high_ap’ is assigned the result of applying the Apriori algorithm to the dataset ‘basket_sets_high’. The ‘min_support’ parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset using the ‘len()’ function and assigns this value to the 'size' column for that particular row. After this the ‘size’ is set to 2, selecting only those itemsets that meet the criterion of having a size of 2. This provide insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the Apriori algorithm, the itemset ‘(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))’ has the highest support of approximately 3.71%, indicating its presence in a significant portion of transactions, corresponding to 4,649 unique invoices. The elapsed time, calculated by subtracting the start time from the end time, is approximately 8.66 seconds. This duration quantifies the computational efficiency of the Apriori algorithm for the ‘High Value Customers’ dataset and parameter settings.

4.5.1.2 High Value Customers cluster – FP-Growth algorithm

The primary goal is to find frequent itemsets within a 'High Value Customers' dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The 'fpgrowth' function is used to perform this task. It is typically provided by a library 'mlxtend' in Python. The runtime of the FP-Growth algorithm is also measured as part of an effort to assess and compare its performance against the Apriori algorithm.

```
start_time = time.time()
frequent_itemsets_high_fp=fpgrowth(basket_sets_high,min_support=0.01,use_colnames=True)
end_time = time.time()
elapsed_time_high_fp = end_time - start_time

print("Elapsed time: ", elapsed_time_high_fp, " seconds")

frequent_itemsets_high_fp['unique_invoices'] = frequent_itemsets_high_fp['support'] * row_count_high
frequent_itemsets_high_fp = frequent_itemsets_high_fp[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_high_fp['size'] = frequent_itemsets_high_fp['itemsets'].apply(lambda x: len(x))

frequent_itemsets_high_fp[(frequent_itemsets_high_fp['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  2.8661961555480957  seconds
```

	itemsets	support	unique_invoices	size
158	(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.037085	4649.0	2
198	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.034109	4276.0	2
106	(WASTE BAGS(WASTE MANAGEMENT), STANDARD TOILET PAPER ROLLS(WASHROOM SUPPLIES & PERSONAL CARE))	0.032634	4091.0	2
150	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.032291	4048.0	2
103	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), BULLET TIP(MARKERS))	0.032004	4012.0	2
136	(AA / LR6(BATTERIES / TORCHES / CHARGERS), AAA / LR3(BATTERIES / TORCHES / CHARGERS))	0.031206	3912.0	2
212	(FLOOR DETERGENTS(DETERGENTS), WASTE BAGS(WASTE MANAGEMENT))	0.030336	3803.0	2
100	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.029467	3694.0	2
154	(STANDARD INK(BALL PENS), BULLET TIP(MARKERS))	0.028573	3582.0	2
161	(STAPLES(STAPLING), CLEAR/TRANSPARENT(ADHESIVES))	0.026396	3309.0	2

Figure 4.39 Result of FP-Growth algorithm to High Value Customer cluster

Figure 4.39, 'frequent_itemsets_high_fp' is assigned the result of applying the FP-Growth algorithm to the dataset 'basket_sets_high'. The 'min_support' parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset using the 'len()' function and assigns this value to the 'size' column for that particular row. After this the 'size' is set to 2, selecting only those itemsets that meet the criterion of

having a size of 2. This provides insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the FP-Growth algorithm, the itemset '(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' has the highest support of approximately 3.71%, indicating its presence in a significant portion of transactions, corresponding to 4,649 unique invoices. The elapsed time, calculated by subtracting the start time from the end time, is approximately 2.87 seconds. This duration quantifies the computational efficiency of the FP-Growth algorithm for the 'High Value Customers' dataset and parameter settings.

4.5.1.3 High Value Customers cluster – Association Rule

Running an Association Rule after applying both the Apriori and FP-Growth algorithms is a crucial step in the context of frequent itemset mining and market basket analysis. Association rule mining, specifically using metrics like confidence and lift, helps us uncover meaningful patterns and relationships within the dataset. Frequent itemset mining algorithms, such as Apriori and FP-Growth, identify itemsets that co-occur frequently, but they do not provide information about the strength or significance of these associations. Association rules, on the other hand, quantify the relationships between items in terms of how often they appear together and how likely one item is to occur when another is present in a transaction.

A study of Brin et al. (1997), is pivotal in quantifying the degree of association between items by comparing observed co-occurrence to random chance. Numerous subsequent studies and applications, including market basket analysis and recommendation systems, have consistently demonstrated that high-confidence rules with lift values greater than 1 are indicative of meaningful and actionable associations.

After consulting with domain experts and trial and error adjusting the appropriate confidence and lift value, considering the insights gained from the realm of association

rule mining, it has been determined that setting a threshold of lift>=3 and confidence>=0.3 is an effective and appropriate criteria value for the specific dataset under examination. The foundational work of Agrawal et al. (1993) on the Apriori algorithm, and the subsequent introduction of confidence and lift metrics by Brin et al. (1997), have underscored the significance of these measures in assessing rule reliability and the degree of item association.

```
as_high = association_rules(frequent_itemsets_high_ap,metric='support',min_threshold=0.01)
as_high[(as_high['lift']>=3)&(as_high['confidence']>=0.3)].sort_values(by=['support'], ascending=False).head(10)
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
316	(STAPLES(STAPLING))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.085952	0.125805	0.037085	0.431462	3.429616	0.026272	1.537619
363	(STANDARD TOILET PAPER ROLLS(WASHROOM SUPPLIES & PERSONAL CARE))	(WASTE BAGS(WASTE MANAGEMENT))	0.067716	0.124488	0.032634	0.481918	3.871184	0.024204	1.689908
79	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	0.048237	0.060545	0.031206	0.646932	10.685124	0.028285	2.660835
78	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	0.060545	0.048237	0.031206	0.515415	10.685124	0.028285	1.964079
256	(FLOOR DETERGENTS(DETERGENTS))	(WASTE BAGS(WASTE MANAGEMENT))	0.063680	0.124488	0.030336	0.476387	3.826758	0.022409	1.672059
148	(STANDARD INK(BALL PENS))	(BULLET TIP(MARKERS))	0.094280	0.100948	0.028573	0.303071	3.002238	0.019056	1.290019
176	(STAPLES(STAPLING))	(CLEAR/TRANSPARENT(ADHESIVES))	0.085952	0.097167	0.026396	0.307100	3.160523	0.018044	1.302976
120	(AIR & FABRIC FRESHENERS(AIR CARE))	(WASTE BAGS(WASTE MANAGEMENT))	0.055432	0.124488	0.025423	0.458627	3.684093	0.018522	1.617206
226	(DISHWASHER DETERGENTS(DETERGENTS))	(WASTE BAGS(WASTE MANAGEMENT))	0.048979	0.124488	0.024417	0.498534	4.004661	0.018320	1.745905
279	(INSTANT COFFEE(HOT DRINKS & SUPPLIES))	(SUGARS(HOT DRINKS & SUPPLIES))	0.057131	0.036263	0.023700	0.414828	11.439350	0.021628	1.646930

Figure 4.40 Result of Association Rule to High Value Customer cluster

Figure 4.40, the provided code is conducting association rule mining to extract association rules based on a minimum support threshold of 0.01. Subsequently, it filters these rules using criteria of a minimum lift value of 3 and a minimum confidence value of 0.3.

The most prevalent itemset in our analysis is '(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS)),' which has a support of around 3.71%. This support value signifies how frequently this itemset appears in our dataset. Moreover, this association is supported by a lift value of 3.43, indicating that the co-occurrence of these items is significantly higher than what would be expected by chance. The confidence value of 0.43 suggests that there's a 43% likelihood of customers purchasing 'METAL(FIXING

ACCESSORIES, CLIPS & FASTENERS)' when they've bought 'STAPLES(STAPLING).' These figures comfortably meet our stringent criteria for strong associations, which require a minimum lift value of 3 and a minimum confidence value of 0.3.

4.5.1.4 High Value Customers cluster – Summary

Upon conducting association rule mining to validate and justify the outcomes of both the Apriori and FP-Growth algorithms, we find a consistency in the results. Notably, '(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' emerges as the most frequent itemset, exhibiting a robust support value of approximately 3.71% or 4,649 times these items appear together out of 125,361 transactions. This observation aligns seamlessly with the stringent criteria we had set, demanding a minimum lift value of 3 and a minimum confidence value of 0.3, both of which are comfortably met. The convergence of findings from multiple methodologies reinforces the significance of this association, affirming it as a reliable and substantial pattern within the dataset.

```
print("Apriori Elapsed time: ", elapsed_time_high_ap, " seconds")
print("FP-Growth Elapsed time: ", elapsed_time_high_fp, " seconds")
```

```
Apriori Elapsed time:  8.658976078033447  seconds
FP-Growth Elapsed time:  2.8661961555480957  seconds
```

Figure 4.41 Comparing algorithms runtime of High Value Customer cluster

Figure 4.41, the provided elapsed times indicate the runtime of Apriori and FP-Growth algorithms, applied to the same dataset. Apriori took approximately 8.66 seconds to complete its execution, whereas FP-Growth achieved the same task significantly faster, with a runtime of approximately 2.87 seconds. This comparison suggests that FP-Growth outperformed Apriori in terms of computational efficiency for the given dataset and parameter settings, making it a potentially more favorable choice for association rule mining tasks when runtime is a critical factor. The elapsed time results are in accordance with the findings of Swaminathan and Shavanas (2013) as well as Patil (2022), both of

whom observed that the FP-Growth algorithm exhibits superior performance compared to the Apriori algorithm when applied to datasets of similar size and criteria.

4.5.2 Medium Value Customers Cluster

The Medium Value Customers cluster is characterized by four key variables: Recency, Frequency, Monetary and Section Variety, each exhibiting a mean value of 54.47, 50.51, 50.31, and 41.72, respectively.

```
df_group_med = df_merged[df_merged["cluster"] == "Medium Value Customers"]
df_group_med.head()
```

	INVOICE_NUMBER	QUANTITY	COUNTRY_CODE	SUBCATEGORY	cluster
689	2212478855	6	TH	AAA / HR3(BATTERIES / TORCHES / CHARGERS)	Medium Value Customers
690	2212478855	8	TH	AA / HR6(BATTERIES / TORCHES / CHARGERS)	Medium Value Customers
691	2212561171	6	TH	AA / HR6(BATTERIES / TORCHES / CHARGERS)	Medium Value Customers
692	2212561171	12	TH	AAA / HR3(BATTERIES / TORCHES / CHARGERS)	Medium Value Customers
693	2212600161	3	TH	STANDARD COLOUR NOTES(REPOSITIONABLE NOTES)	Medium Value Customers

Figure 4.42 Medium Value Customers DataFrame

Figure 4.42, this DataFrame is a subset of the original data, consisting solely of rows where the "cluster" column matches the label 'Medium Value Customers' In essence, it filters and retains only those customers who belong to the 'Medium Value Customers' cluster, effectively isolating this particular segment from the broader dataset.

```
basket_group_med = (df_group_med[df_group_med['COUNTRY_CODE']=='TH'].groupby(['INVOICE_NUMBER','SUBCATEGORY'])\
                    ['QUANTITY'].sum().unstack().reset_index().fillna(0).set_index('INVOICE_NUMBER'))
basket_group_med.head()
```

SUBCATEGORY	#9 (4"X9") (ENVELOPES & POSTROOM)	1-HOLE PUNCHES(HOLE PUNCHES)	10"X13"(ENVELOPES & POSTROOM)	16GB(FLASH MEMORY)	2-HOLE PUNCHES(HOLE PUNCHES)	2-RING BINDERS(BINDERS)	2.5"(FLASH MEMORY)	32GB(FLASH MEMORY)
INVOICE_NUMBER								
2212399603	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399607	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399609	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399611	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399617	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 499 columns

```
row_count = len(basket_group_med)
print("Number of unique invoices:", row_count)
```

```
Number of unique invoices: 38531
```

Figure 4.43 Medium Value Customers Cluster Invoice Grouping

Figure 4.43, grouping the data by two key columns: "INVOICE_NUMBER" and "SUBCATEGORY." The "INVOICE_NUMBER" column represents individual purchase transactions, while "SUBCATEGORY" denotes the product subcategory associated with each item in the transaction. Within each transaction (invoice), we calculate the sum of quantities for each unique subcategory of products. The result is a table where rows represent individual transactions (invoices), columns correspond to product subcategories, and the values indicate the total quantity of products purchased from each subcategory within each transaction. To enhance the structure of this table, any missing values (i.e., cases where no products were purchased from a particular subcategory in a given transaction) are filled with zeros using the 'fillna(0)' method. The resulting dataset is then set to be indexed by "INVOICE_NUMBER.". Notably, the figure 499, situated in the bottom left corner, represents the count of unique subcategories that the 'Medium Value Customers' cluster purchased and the figure 38,531 represents the count of unique invoices during the specified period.

4.5.2.1 Medium Value Customers cluster – Apriori algorithm

The primary goal is to find frequent itemsets within a 'Medium Value Customers' dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The 'apriori' function is used to perform this task. It is typically provided by a library 'mlxtend' in Python. The runtime of the Apriori algorithm is also measured as part of an effort to assess and compare its performance against the FP-growth algorithm.

```
start_time = time.time()
frequent_itemsets_med_ap = apriori(basket_sets_med, min_support = 0.01, use_colnames=True)
end_time = time.time()
elapsed_time_med_ap = end_time - start_time

print("Elapsed time: ", elapsed_time_med_ap, " seconds")

frequent_itemsets_med_ap['unique_invoices'] = frequent_itemsets_med_ap['support'] * row_count_med
frequent_itemsets_med_ap = frequent_itemsets_med_ap[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_med_ap['size'] = frequent_itemsets_med_ap['itemsets'].apply(lambda x: len(x))

frequent_itemsets_med_ap[(frequent_itemsets_med_ap['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  1.1980671882629395  seconds
```

	itemsets	support	unique_invoices	size
97	(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.028678	1105.0	2
115	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.026861	1035.0	2
144	(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.025746	992.0	2
142	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.023228	895.0	2
108	(BULLET TIP(MARKERS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.023072	889.0	2
94	(CLEAR/TRANSPARENT(ADHESIVES), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.022735	876.0	2
89	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE A GRADE/PREMIUM(WHITE OFFICE PAPER))	0.021359	823.0	2
119	(CLEAR/TRANSPARENT(ADHESIVES), STAPLES(STAPLING))	0.020295	782.0	2
110	(BULLET TIP(MARKERS), STANDARD INK(BALL PENS))	0.020140	776.0	2
93	(BULLET TIP(MARKERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.019880	766.0	2

Figure 4.44 Result of Apriori algorithm to Medium Value Customer cluster

Figure 4.44, 'frequent_itemsets_med_ap' is assigned the result of applying the Apriori algorithm to the dataset 'basket_sets_med'. The 'min_support' parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset using the 'len()' function and assigns this value to the 'size' column for that particular row. After this the 'size' is set to 2, selecting only those itemsets that meet the criterion of having a size of 2. This provide insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the Apriori algorithm, the itemset '(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' has the highest support of approximately 2.87%, indicating its presence in a significant portion of transactions, corresponding to 1,105 unique invoices. The elapsed time, calculated by subtracting the start time from the end

time, is approximately 1.20 seconds. This duration quantifies the computational efficiency of the Apriori algorithm for the 'Medium Value Customers' dataset and parameter settings.

4.5.2.2 Medium Value Customers cluster – FP-Growth algorithm

The primary goal is to find frequent itemsets within a 'Medium Value Customers' dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The 'fpgrowth' function is used to perform this task. It is typically provided by a library 'mlxtend' in Python. The runtime of the FP-Growth algorithm is also measured as part of an effort to assess and compare its performance against the Apriori algorithm.

```
start_time = time.time()
frequent_itemsets_med_fp=fpgrowth(basket_sets_med,min_support=0.01,use_colnames=True)
end_time = time.time()
elapsed_time_med_fp = end_time - start_time

print("Elapsed time: ", elapsed_time_med_fp, " seconds")

frequent_itemsets_med_fp['unique_invoices'] = frequent_itemsets_med_fp['support'] * row_count_med
frequent_itemsets_med_fp = frequent_itemsets_med_fp[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_med_fp['size'] = frequent_itemsets_med_fp['itemsets'].apply(lambda x: len(x))

frequent_itemsets_med_fp[(frequent_itemsets_med_fp['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  0.6370904445648193  seconds
```

	itemsets	support	unique_invoices	size
94	(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.028678	1105.0	2
119	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.026861	1035.0	2
95	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), STAPLES(STAPLING))	0.025746	992.0	2
83	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.023228	895.0	2
90	(BULLET TIP(MARKERS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.023072	889.0	2
121	(CLEAR/TRANSPARENT(ADHESIVES), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.022735	876.0	2
114	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE A GRADE/PREMIUM(WHITE OFFICE PAPER))	0.021359	823.0	2
99	(CLEAR/TRANSPARENT(ADHESIVES), STAPLES(STAPLING))	0.020295	782.0	2
82	(BULLET TIP(MARKERS), STANDARD INK(BALL PENS))	0.020140	776.0	2
93	(BULLET TIP(MARKERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.019880	766.0	2

Figure 4.45 Result of FP-Growth algorithm to Medium Value Customer cluster

Figure 4.45, 'frequent_itemsets_med_fp' is assigned the result of applying the FP-Growth algorithm to the dataset 'basket_sets_med'. The 'min_support' parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset

using the ‘len()’ function and assigns this value to the 'size' column for that particular row. After this the ‘size’ is set to 2, selecting only those itemsets that meet the criterion of having a size of 2. This provide insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the FP-Growth algorithm, the itemset ‘(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))’ has the highest support of approximately 2.87%, indicating its presence in a significant portion of transactions, corresponding to 1,105 unique invoices. The elapsed time, calculated by subtracting the start time from the end time, is approximately 0.64 seconds. This duration quantifies the computational efficiency of the FP-Growth algorithm for the ‘Medium Value Customers’ dataset and parameter settings.

4.5.2.3 Medium Value Customers cluster – Association Rule

Running an Association Rule algorithm after applying both the Apriori and FP-Growth algorithms is a crucial step in the context of frequent itemset mining and market basket analysis. Association rule mining, specifically using metrics like confidence and lift, helps us uncover meaningful patterns and relationships within the dataset. Frequent itemset mining algorithms, such as Apriori and FP-Growth, identify itemsets that co-occur frequently, but they do not provide information about the strength or significance of these associations. Association rules, on the other hand, quantify the relationships between items in terms of how often they appear together and how likely one item is to occur when another is present in a transaction.

```
as_med = association_rules(frequent_itemsets_med_ap,metric='support',min_threshold=0.01)
as_med[(as_med['lift']>=3)&(as_med['confidence']>=0.3)].sort_values(by=['support'], ascending=False).head(7)
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
66	(CLEAR/TRANSPARENT(ADHESIVES))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.085334	0.103397	0.026861	0.314781	3.044384	0.018038	1.308491
124	(STAPLES(STAPLING))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.065090	0.103397	0.025746	0.395534	3.825384	0.019015	1.483298
75	(STAPLES(STAPLING))	(CLEAR/TRANSPARENT(ADHESIVES))	0.065090	0.085334	0.020295	0.311802	3.653909	0.014741	1.329075
99	(FLOOR DETERGENTS(DETERGENTS))	(WASTE BAGS(WASTE MANAGEMENT))	0.042719	0.083906	0.019698	0.461118	5.495618	0.016114	1.699989
135	(STANDARD TOILET PAPER ROLLS(WASHROOM SUPPLIES & PERSONAL CARE))	(WASTE BAGS(WASTE MANAGEMENT))	0.043757	0.083906	0.019387	0.443060	5.280410	0.015715	1.644871
43	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	0.033090	0.026446	0.016454	0.497255	18.802481	0.015579	1.936476
42	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	0.026446	0.033090	0.016454	0.622179	18.802481	0.015579	2.559172

Figure 4.46 Result of Association Rule to Medium Value Customer cluster

Figure 4.46, the provided code is conducting association rule mining to extract association rules based on a minimum support threshold of 0.01. Subsequently, it filters these rules using criteria of a minimum lift value of 3 and a minimum confidence value of 0.3.

The most prevalent itemset in our analysis is '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' which has a support of around 2.69%. This support value signifies how frequently this itemset appears in our dataset. Moreover, this association is supported by a lift value of 3.04, indicating that the co-occurrence of these items is significantly higher than what would be expected by chance. The confidence value of 0.31 suggests that there's a 31% likelihood of customers purchasing 'METAL(FIXING ACCESSORIES, CLIPS & FASTENERS)' when they've bought 'CLEAR/TRANSPARENT(ADHESIVES)' These figures comfortably meet our stringent criteria for strong associations, which require a minimum lift value of 3 and a minimum confidence value of 0.3.

4.5.2.4 Medium Value Customers cluster – Summary

Upon conducting association rule mining to validate and justify the outcomes of both the Apriori and FP-Growth algorithms, an inconsistency in the results has emerged. Notably, from the Apriori and FP-Growth algorithms, the itemset '(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS &

FASTENERS))' emerges as the most frequent itemset, exhibiting a support value of approximately 2.87%, signifying that these items appear together 1,105 times out of 38,531 transactions. However, it's important to note that this observation does not align with the stringent criteria we had set, which demanded a minimum lift value of 3 and a minimum confidence value of 0.3 from association rule mining.

Instead, association rule mining indicates that the itemset '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' with the second-highest support value of approximately 2.69% is a more appropriate rule. This itemset successfully meets both the minimum lift and confidence value criteria, demonstrating that it has strong statistical significance and practical relevance within the dataset. This inconsistency in results underscores the importance of scrutinizing the outcomes of different algorithms and criteria to arrive at the most meaningful and actionable associations in the data.

```
print("Apriori Elapsed time: ", elapsed_time_med_ap, " seconds")
print("FP-Growth Elapsed time: ", elapsed_time_med_fp, " seconds")
```

```
Apriori Elapsed time:  1.1980671882629395  seconds
FP-Growth Elapsed time:  0.6370904445648193  seconds
```

Figure 4.47 Comparing algorithms runtime of Medium Value Customer cluster

Figure 4.47, the provided elapsed times indicate the runtime of Apriori and FP-Growth algorithms, applied to the same dataset. Apriori took approximately 1.20 seconds to complete its execution, whereas FP-Growth achieved the same task significantly faster, with a runtime of approximately 0.64 seconds. This comparison suggests that FP-Growth outperformed Apriori in terms of computational efficiency for the given dataset and parameter settings, making it a potentially more favorable choice for association rule mining tasks when runtime is a critical factor. The elapsed time results are in accordance with the findings of Swaminathan and Shavanas (2013) as well as Patil (2022), both of whom observed that the FP-Growth algorithm exhibits superior performance compared to the Apriori algorithm when applied to datasets of similar size and criteria.

4.5.3 Low Value Customers Cluster

The Low Value Customers Cluster is characterized by four key variables: Recency, Frequency, Monetary and Section Variety, each exhibiting a mean value of 23.96, 19.34, 20.33, and 24.17, respectively.

```
df_group_low = df_merged[df_merged["cluster"] == "Low Value Customers"]
df_group_low.head()
```

	INVOICE_NUMBER	QUANTITY	COUNTRY_CODE	SUBCATEGORY	cluster
127	2212427333	5	TH	LABELS FOR ELECTRONIC MACHINES&TAPE G(ELECTRONIC & MECHANICAL LABELLING)	Low Value Customers
128	2212546021	5	TH	LABELS FOR ELECTRONIC MACHINES&TAPE G(ELECTRONIC & MECHANICAL LABELLING)	Low Value Customers
495	2212460765	1	TH	A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER)	Low Value Customers
496	2212460765	24	TH	STAPLES(STAPLING)	Low Value Customers
662	2212410489	3	TH	AA / LR6(BATTERIES / TORCHES / CHARGERS)	Low Value Customers

Figure 4.48 Low Value Customers DataFrame

Figure 4.48, this DataFrame is a subset of the original data, consisting solely of rows where the "cluster" column matches the label 'Low Value Customers' In essence, it filters and retains only those customers who belong to the 'Low Value Customers' cluster, effectively isolating this particular segment from the broader dataset.

```
basket_group_low = (df_group_low[df_group_low['COUNTRY_CODE']=='TH'].groupby(['INVOICE_NUMBER','SUBCATEGORY'])\
                    ['QUANTITY'].sum().unstack().reset_index().fillna(0).set_index('INVOICE_NUMBER'))
basket_group_low.head()
```

SUBCATEGORY INVOICE_NUMBER	#9 (4"X9") (ENVELOPES & POSTROOM)	1-HOLE PUNCHES(HOLE PUNCHES)	10"X13"(ENVELOPES & POSTROOM)	16GB(FLASH MEMORY)	2-HOLE PUNCHES(HOLE PUNCHES)	2-RING BINDERS(BINDERS)	2.5"(FLASH MEMORY)	32GB(FLASH MEMORY)
2212399596	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399597	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399633	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399636	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2212399656	0.0	0.0	0.0	0.0	0.0	7.0	0.0	0.0

5 rows × 460 columns

```
row_count_low = len(basket_group_low)
print("Number of unique invoices:", row_count_low)
```

```
Number of unique invoices: 11243
```

Figure 4.49 Low Value Customers Cluster Invoice Grouping

Figure 4.49, grouping the data by two key columns: "INVOICE_NUMBER" and "SUBCATEGORY." The "INVOICE_NUMBER" column represents individual purchase transactions, while "SUBCATEGORY" denotes the product subcategory associated with each item in the transaction. Within each transaction (invoice), we calculate the sum of quantities for each unique subcategory of products. The result is a table where rows represent individual transactions (invoices), columns correspond to product subcategories, and the values indicate the total quantity of products purchased from each subcategory within each transaction. To enhance the structure of this table, any missing values (i.e., cases where no products were purchased from a particular subcategory in a given transaction) are filled with zeros using the ‘fillna(0)’ method. The resulting dataset is then set to be indexed by "INVOICE_NUMBER.". Notably, the figure 460, situated in the bottom left corner, represents the count of unique subcategories that the 'Low Value Customers' cluster purchased and the figure 11,243 represents the count of unique invoices during the specified period.

4.5.3.1 Low Value Customers cluster – Apriori algorithm

The primary goal is to find frequent itemsets within a ‘Low Value Customers’ dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The ‘apriori’ function is used to perform this task. It is typically provided by a library ‘mlxtend’ in Python. The runtime of the Apriori algorithm is also measured as part of an effort to assess and compare its performance against the FP-growth algorithm.

```
start_time = time.time()
frequent_itemsets_low_ap = apriori(basket_sets_low, min_support = 0.01, use_colnames=True)
end_time = time.time()
elapsed_time_low_ap = end_time - start_time

print("Elapsed time: ", elapsed_time_low_ap, " seconds")

frequent_itemsets_low_ap['unique_invoices'] = frequent_itemsets_low_ap['support'] * row_count_low
frequent_itemsets_low_ap = frequent_itemsets_low_ap[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_low_ap['size'] = frequent_itemsets_low_ap['itemsets'].apply(lambda x: len(x))

frequent_itemsets_low_ap[(frequent_itemsets_low_ap['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  0.22039508819580078  seconds
```

	itemsets	support	unique_invoices	size
80	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.020279	228.0	2
71	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.020012	225.0	2
88	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), STAPLES(STAPLING))	0.019301	217.0	2
87	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.016366	184.0	2
68	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE A GRADE/PREMIUM(WHITE OFFICE PAPER))	0.015832	178.0	2
70	(CLEAR/TRANSPARENT(ADHESIVES), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.015654	176.0	2
83	(CLEAR/TRANSPARENT(ADHESIVES), STAPLES(STAPLING))	0.015565	175.0	2
82	(CLEAR/TRANSPARENT(ADHESIVES), STANDARD INK(BALL PENS))	0.014587	164.0	2
77	(BULLET TIP(MARKERS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.014142	159.0	2
73	(STANDARD INK(BALL PENS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.013964	157.0	2

Figure 4.50 Result of Apriori algorithm to Low Value Customer cluster

Figure 4.50, 'frequent_itemsets_low_ap' is assigned the result of applying the Apriori algorithm to the dataset 'basket_sets_low'. The 'min_support' parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset using the 'len()' function and assigns this value to the 'size' column for that particular row. After this the 'size' is set to 2, selecting only those itemsets that meet the criterion of having a size of 2. This provide insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the Apriori algorithm, the itemset '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' has the highest support of approximately 2.03%, indicating its presence in a significant portion of transactions, corresponding to 228 unique invoices. The elapsed time, calculated by subtracting the start time from the end time, is approximately 0.22 seconds.

This duration quantifies the computational efficiency of the Apriori algorithm for the ‘Low Value Customers’ dataset and parameter settings.

4.5.3.2 Low Value Customers cluster – FP-Growth algorithm

The primary goal is to find frequent itemsets within a ‘Low Value Customers’ dataset. Frequent items represent combinations of items that frequently co-occur together in transactions or baskets of items. This analysis can be useful in various applications such as market basket analysis or recommendation systems. The ‘fpgrowth’ function is used to perform this task. It is typically provided by a library ‘mlxtend’ in Python. The runtime of the FP-Growth algorithm is also measured as part of an effort to assess and compare its performance against the Apriori algorithm.

```
start_time = time.time()
frequent_itemsets_low_fp = fpgrowth(basket_sets_low, min_support = 0.01, use_colnames=True)
end_time = time.time()
elapsed_time_low_fp = end_time - start_time

print("Elapsed time: ", elapsed_time_low_fp, " seconds")

frequent_itemsets_low_fp['unique_invoices'] = frequent_itemsets_low_fp['support'] * row_count_low
frequent_itemsets_low_fp = frequent_itemsets_low_fp[['itemsets', 'support', 'unique_invoices']]
frequent_itemsets_low_fp['size'] = frequent_itemsets_low_fp['itemsets'].apply(lambda x: len(x))

frequent_itemsets_low_fp[(frequent_itemsets_low_fp['size'] == 2)].sort_values(by=['support'], ascending=False).head(10)
```

```
Elapsed time:  0.1329944133758545  seconds
```

	itemsets	support	unique_invoices	size
73	(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.020279	228.0	2
68	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.020012	225.0	2
80	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), STAPLES(STAPLING))	0.019301	217.0	2
86	(STANDARD INK(BALL PENS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.016366	184.0	2
67	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS), A4 WHITE A GRADE/PREMIUM(WHITE OFFICE PAPER))	0.015832	178.0	2
75	(CLEAR/TRANSPARENT(ADHESIVES), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.015654	176.0	2
81	(CLEAR/TRANSPARENT(ADHESIVES), STAPLES(STAPLING))	0.015565	175.0	2
85	(CLEAR/TRANSPARENT(ADHESIVES), STANDARD INK(BALL PENS))	0.014587	164.0	2
79	(BULLET TIP(MARKERS), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.014142	159.0	2
87	(STANDARD INK(BALL PENS), A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER))	0.013964	157.0	2

Figure 4.51 Result of FP-Growth algorithm to Low Value Customer cluster

Figure 4.51, ‘frequent_itemsets_low_fp’ is assigned the result of applying the FP-Growth algorithm to the dataset ‘basket_sets_low’. The ‘min_support’ parameter is set to 0.01, indicating that only itemsets with a support of at least 1% will be considered frequent. The lambda function is applied to calculate the number of items in each itemset using the ‘len()’ function and assigns this value to the 'size' column for that particular row. After this the ‘size’ is set to 2, selecting only those itemsets that meet the criterion of having a size of 2. This provide insights into which pairs of items are frequently purchased together.

Among the top 10 frequent itemsets extracted using the FP-Growth algorithm, the itemset ‘(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))’ highest support of approximately 2.03%, indicating its presence in a significant portion of transactions, corresponding to 228 unique invoices. The elapsed time, calculated by subtracting the start time from the end time, is approximately 0.13 seconds. This duration quantifies the computational efficiency of the FP-Growth algorithm for the ‘Low Value Customers’ dataset and parameter settings.

4.5.3.3 Low Value Customers cluster – Association Rule

Running an Association Rule algorithm after applying both the Apriori and FP-Growth algorithms is a crucial step in the context of frequent itemset mining and market basket analysis. Association rule mining, specifically using metrics like confidence and lift, helps us uncover meaningful patterns and relationships within the dataset. Frequent itemset mining algorithms, such as Apriori and FP-Growth, identify itemsets that co-occur frequently, but they do not provide information about the strength or significance of these associations. Association rules, on the other hand, quantify the relationships between items in terms of how often they appear together and how likely one item is to occur when another is present in a transaction.

```
as_low = association_rules(frequent_itemsets_low_ap,metric='support',min_threshold=0.01)
as_low[(as_low['lift']>=3)&(as_low['confidence']>=0.3)].sort_values(by=['support'], ascending=False).head(7)
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
26	(CLEAR/TRANSPARENT(ADHESIVES))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.064307	0.077204	0.020279	0.315353	4.084689	0.015315	1.347842
43	(STAPLES(STAPLING))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.051321	0.077204	0.019301	0.376083	4.871317	0.015339	1.479038
33	(STAPLES(STAPLING))	(CLEAR/TRANSPARENT(ADHESIVES))	0.051321	0.064307	0.015565	0.303293	4.716351	0.012265	1.343023
16	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	0.018678	0.024460	0.013075	0.700000	28.618545	0.012618	3.251801
17	(AA / LR6(BATTERIES / TORCHES / CHARGERS))	(AAA / LR3(BATTERIES / TORCHES / CHARGERS))	0.024460	0.018678	0.013075	0.534545	28.618545	0.012618	2.108308
39	(FLOOR DETERGENTS(DETERGENTS))	(WASTE BAGS(WASTE MANAGEMENT))	0.029174	0.059948	0.012363	0.423780	7.069086	0.010614	1.631412
44	(STICKS(GLUES))	(METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))	0.029885	0.077204	0.011741	0.392857	5.088586	0.009433	1.519900

Figure 4.52 Result of Association Rule to Low Value Customer cluster

Figure 4.52, the provided code is conducting association rule mining to extract association rules based on a minimum support threshold of 0.01. Subsequently, it filters these rules using criteria of a minimum lift value of 3 and a minimum confidence value of 0.3.

The most prevalent itemset in our analysis is '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS)),' which has a support of around 2.03%. This support value signifies how frequently this itemset appears in our dataset. Moreover, this association is supported by a lift value of 4.08, indicating that the co-occurrence of these items is significantly higher than what would be expected by chance. The confidence value of 0.31 suggests that there's a 31% likelihood of customers purchasing 'METAL(FIXING ACCESSORIES, CLIPS & FASTENERS)' when they've bought CLEAR/TRANSPARENT(ADHESIVES).' These figures comfortably meet our stringent criteria for strong associations, which require a minimum lift value of 3 and a minimum confidence value of 0.3.

4.5.3.4 Low Value Customers cluster – Summary

Upon conducting association rule mining to validate and justify the outcomes of both the Apriori and FP-Growth algorithms, we find a consistency in the results. Notably, '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' emerges as the most frequent itemset, exhibiting a robust support value of approximately 2.03% or 228 times these items appear together out of 11,243 transactions. This observation aligns seamlessly with the stringent criteria we had set, demanding a minimum lift value of 3 and a minimum confidence value of 0.3, both of which are comfortably met. The convergence of findings from multiple methodologies reinforces the significance of this association, affirming it as a reliable and substantial pattern within the dataset.

```
print("Apriori Elapsed time: ", elapsed_time_low_ap, " seconds")
print("FP-Growth Elapsed time: ", elapsed_time_low_fp, " seconds")
```

```
Apriori Elapsed time:  0.22039508819580078  seconds
FP-Growth Elapsed time:  0.1329944133758545  seconds
```

Figure 4.53 Comparing algorithms runtime of Low Value Customer cluster

Figure 4.53, the provided elapsed times indicate the runtime of Apriori and FP-Growth algorithms, applied to the same dataset. Apriori took approximately 0.22 seconds to complete its execution, whereas FP-Growth achieved the same task significantly faster, with a runtime of approximately 0.13 seconds. This comparison suggests that FP-Growth outperformed Apriori in terms of computational efficiency for the given dataset and parameter settings, making it a potentially more favorable choice for association rule mining tasks when runtime is a critical factor. The elapsed time results are in accordance with the findings of Swaminathan and Shavanas (2013) as well as Patil (2022), both of whom observed that the FP-Growth algorithm exhibits superior performance compared to the Apriori algorithm when applied to datasets of similar size and criteria.

4.6 Summary

Across the three customer value segments—high-value, medium-value, and low-value—upon conducting association rule mining to validate and justify the outcomes of both the Apriori and FP-Growth algorithms, we find variations in the results. For high-value customers, a consistent and robust result emerges with the itemset '(STAPLES(STAPLING), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' aligning seamlessly with stringent criteria. In contrast, medium-value customers exhibit an inconsistency, with the itemset '(A4 WHITE B GRADE/MULTIPURPOSE(WHITE OFFICE PAPER), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' not meeting the predefined criteria. Finally, low-value customers demonstrate consistency, with the itemset '(CLEAR/TRANSPARENT(ADHESIVES), METAL(FIXING ACCESSORIES, CLIPS & FASTENERS))' meeting the stringent criteria. These findings underscore the importance of segment-specific analysis when interpreting

association patterns, revealing both reliable and divergent insights within each customer segment.

In examining the computational efficiency of the Apriori and FP-Growth algorithms across three customer value segments—high-value, medium-value, and low-value—it becomes evident that FP-Growth consistently outperformed Apriori in terms of runtime. For high-value customers, FP-Growth completed its execution in approximately 2.87 seconds, significantly faster than Apriori's 8.66 seconds. Similarly, for medium-value customers, FP-Growth achieved a runtime of around 0.64 seconds compared to Apriori's 1.20 seconds. In the case of low-value customers, FP-Growth again demonstrated superior efficiency, taking approximately 0.13 seconds, while Apriori required 0.22 seconds. These findings echo prior research by Swaminathan and Shavanas (2013) and Patil (2022), reinforcing that FP-Growth is a more expedient choice for association rule mining tasks when runtime is a critical consideration, irrespective of customer value segments.

Chapter 5

CONCLUSION

The utilization of segmentation and marketing basket analysis has emerged as a formidable strategy for companies seeking to enhance their sales performance. Through this comprehensive case study of an office supplier company, we have witnessed the potential of these techniques. The implementation of segmentation allowed for the identification of distinct customer groups with unique behaviors and preferences, enabling tailored marketing approaches. Simultaneously, marketing basket analysis uncovered valuable insights into cross-selling and upselling opportunities, facilitating the maximization of revenue from existing clients. As we conclude this study, it is evident that segmentation and marketing basket analysis are not mere tools but strategic imperatives for companies operating in the B2B space. By embracing these methodologies, businesses can unlock a new level of customer understanding, optimize their marketing efforts, and ultimately drive sustainable growth in an increasingly competitive marketplace.

5.1 Conclusion

This research begins by looking at past sales data, focusing on specific criteria and a certain period. To distill meaningful customer segmentation, we employ the K-Means clustering algorithm, leveraging four key variables derived from RFM (Recency, Frequency, Monetary) analysis and the specific product sections customers have bought during the study period. Notably, this multifaceted approach is designed to comprehensively encapsulate customer behavior, offering a nuanced perspective on their preferences and engagement patterns. The delineation of the optimal number of clusters is accomplished through the application of the elbow method. This pivotal step ensures that our segmentation framework achieves the delicate balance between granularity and interpretability.

Each of the selected variables and clusters undergoes further analysis through the analysis of variance (ANOVA) and the Tukey post hoc test. These statistical methodologies serve as our measure in gauging the extent of dissimilarity and statistical significance between variables and clusters. Such analyses illuminate the true distinctions that exist among these segments, affirming the validity of our segmentation strategy. In essence, this multi-step process provides empirical evidence to substantiate the discernible differences inherent within our identified clusters.

After segmenting our customers into distinct groups, the next step involves employing three different market basket analysis methods within each cluster. Specifically, we apply the Apriori algorithm and the FP-Growth algorithm to investigate each cluster's purchasing patterns. This enables us to not only measure the support for various patterns but also to make a comparative assessment of the speed and efficiency of both algorithms.

Subsequently, we employ an Association rule analysis to justify the results obtained from the Apriori and FP-Growth algorithms. This validation step seeks to determine whether the most frequently identified rules by both algorithms hold up under further scrutiny. In this context, we utilize measures such as confidence and lift values, employing predefined minimum criteria values as benchmarks to assess the validity of these rules. By doing so, we ensure that the patterns identified by the Apriori and FP-Growth algorithms are not only frequent but also possess additional criteria that lend credibility to their significance in our market basket analysis.

The Apriori and FP-Growth algorithms have consistently produced congruent results across all three clusters, both in terms of the appearance of itemsets and their computational runtimes. It is noteworthy that when subjected to the same dataset and criteria, the FP-Growth algorithm consistently outperforms Apriori in terms of speed and efficiency. Among our high-value customers, the most frequently co-purchased itemsets within this cluster notably include staple filling and metal clip/fastener. In contrast, both

our medium and low-value customer clusters exhibit a recurring trend, where the most frequent itemsets for both clusters consist of clear tape and metal clip/fastener.

5.2 Benefits

The study unveils distinct customer segments within the B2B field sales customer base, spanning from low-value, medium-value and high-value customers, each with its unique preferences and purchasing behaviors. Market basket analysis uncover valuable product associations, providing insights for cross-selling and upselling opportunity. Consequently, the research may propose tailored actions to enhance sales, such as targeted promotions, product bundling, or personalized approaches, catering to the diverse needs of these segments. These strategies aim to boost metrics like sales revenue, customer retention, average order value, and customer satisfaction, providing measurable outcomes for the implemented tactics.

5.3 Suggestions

- To enhance the approach, contemplate delving into dynamic or real-time customer segmentation, as the ever-evolving nature of customer behaviors and preferences necessitates the utilization of machine learning techniques to consistently update and adapt segmentation models in response to evolving data.

- To examine the influence of external factors, such as economic conditions or industry trends, on customer segmentation and purchasing patterns, as this analysis can yield valuable insights essential for businesses navigating dynamic environments.

- To utilize the established research framework and methodologies across diverse industries or sectors to evaluate the generalizability of the findings, enabling the adaptation of strategies tailored to specific business environments based on the insights garnered.

- To supplement quantitative findings with qualitative research methodologies, such as conducting customer interviews or surveys, to acquire a more profound understanding of the underlying reasons behind specific purchasing patterns within each distinct customer segment.

REFERENCE

An, J. Kwak, H. Jung, S.G. Salminen, J. and Jansen, J. 2018. “Customer Segmentation Using Online Platforms: Isolating Behavioral and Demographic Segments for Persona Creation via Aggregated User Data”. Social Network Analysis and Mining. 8. DOI: 10.1007/s13278-018-0531-0.

Brunner, T. and Siegrist, M. 2011. “A Consumer-Oriented Segmentation Study in The Swiss Wine Market”. British Food Journal. 113. 353-373. DOI: 10.1108/00070701111116437.

Buttle, F. 2008. “Customer Relationship Management: Concepts and Technologies”. DOI: 10.4324/9780080949611.

Chen, M.C. Chiu, A.L. and Chang, H.H. 2005. “Mining changes in customer behavior in retail marketing”. Expert Systems with Applications. 28. 773-781. DOI: 10.1016/j.eswa.2004.12.033.

Gomes, M. and Meisen, T. 2023. “A Review on Customer Segmentation Methods for Personalized Customer Targeting in E-Commerce Use Cases”. Information Systems and e-Business Management. 1-44. DOI: 10.1007/s10257-023-00640-4.

Grewal, R and Lilien, G. 2012. “Business-To-Business Marketing: Looking Back, Looking Forward”. Handbook of Business-to-Business Marketing. 3-12. DOI: 10.4337/9781849801423.00008.

Griva, A. Kotsopoulos, D. Karagiannaki, A. and Zamani, E. 2021. “What Do Growing Early-Stage Digital Start-Ups Look Like? A Mixed-Methods Approach”. International Journal of Information Management. 69. DOI: 102427. 10.1016/j.ijinfomgt.2021.102427.

Hal, G. J. and Åke F. 1980. “Some Factors in Industrial Market Segmentation”. Industrial Marketing Management, Volume 9, Issue 3, DOI: 10.1016/0019-8501(80)90003-6

Han, J. Pei, J. and Yin, Y. 2004. “Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach”. Data Mining and Knowledge Discovery 8, 53–87. DOI: 10.1023/B:DAMI.0000005258.31418.83.

Hsu, P.Y., Huang, C.W. 2020. “A Methodology for Identifying Critical Products Using Purchase Transactions”. Appl Soft Comput. DOI: 10.1016/j.asoc.2020.106420.

Jain, D. Singh, M. and Sharma, A.K. (2017). “Comparative Study of Density Based Clustering Algorithms for Data Mining”. International Journal of Computer Applications. 8. 9-13. DOI:10.5120/3341-4600.

Josan, M., 2018. “B2B vs. B2C: A Comparative Analysis”. International Journal of Research and Analytical Reviews. 5(4). ISSN 2349-5138.

Kaur, M. and Kang, S. 2016. “Market Basket Analysis: Identify the Changing Trends of Market Data Using Association Rule Mining”. Procedia Computer Science. 85: 78-85. DOI: 10.1016/j.procs.2016.05.180.

Liu, H.B. McCarthy, B. and Chen, T. 2013. “The Chinese Wine Market: A Market Segmentation Study”. Asia Pacific Journal of Marketing and Logistics. 26. 450-471. DOI: 10.1108/APJML-07-2013-0089.

Martinez, M. and Escobar, M. B. 2021. “Market Basket Analysis with Association Rules in The Retail Sector Using Orange. Case Study: Appliances Sales Company”. CLEI Electronic Journal. 24(2). DOI: 10.19153/cleiej.24.2.12.

McDonald, M. and Dunbar, I. 2012. “Market Segmentation: How to Do It and How to Profit from It”. 1-19. Wiley. DOI: 10.1002/9781119207863.

Mulhern, F. J., & Leone, R. P. 1991. “Implicit Price Bundling of Retail Products: A Multiproduct Approach to Maximizing Store Profitability”. Journal of Marketing. *55*(4), 63–76. DOI: 10.2307/1251957.

Omran, M. Engelbrecht, A. and Salman, A. 2007. “An Overview of Clustering Methods”. Intell. Data Anal. 11. 583-605. DOI: 10.3233/IDA-2007-11602.

Patil, B. and Khot, L. 2022. “A Study on Market Basket Analysis Using Apriori Algorithm”. DOI: 10.13140/RG.2.2.19506.48328.

Richards, K. A., and Jones, E. 2008. “Customer Relationship Management: Finding Value Drivers”. Journal of Collaborative Customer Relationship Management, 37, 120-130. DOI: 10.1016/j.indmarman.2006.08.005.

Ruswati, R. Gufroni, A. and Rianto, R. 2018. “Associative Analysis Data Mining Pattern Against Traffic Accidents Using Apriori Algorithm”. Scientific Journal of Informatics. 5. 91-104. DOI: 10.15294/sji.v5i2.16199.

Samboteng, L. Kasmad, R. Kasmad, M. Basit, M. and Rahim, R. 2022. “Market Basket Analysis of Administrative Patterns Data of Consumer Purchases Using Data Mining Technology”. Journal of Applied Engineering Science. 20. 1-7. DOI: 10.5937/jaes0-32019.

Sanjeev V. Rohit S. Subhamay D. and Debojit M. 2021. “Artificial Intelligence in Marketing: Systematic Review and Future Research Direction”. International Journal of Information Management Data Insights, 1(1). DOI: 10.1016/j.jjimei.2020.100002.

Sara D. 2002. “A Review of Data-Driven Market Segmentation in Tourism, Journal of Travel & Tourism Marketing”, 12(1), 1-22, DOI: 10.1300/J073v12n01_01

Sergey, B. Rajeev, M. Jeffrey, D. and Shalom, T. 1997. “Dynamic Itemset Counting and Implication Rules for Market Basket Data”. SIGMOD. 26(2). 255–264. DOI: 10.1145/253262.253325.

Stormi, K. Lindholm, A. and Laine, T. 2020. “RFM Customer Analysis for Product-Oriented Services and Service Business Development: An Interventionist Case Study of Two

Machinery Manufacturers". J Manag Gov. 24, 623–653. DOI: 10.1007/s10997-018-9447-3.

Swaminathan, M. and Shanavas, M. 2013. "Performance Evaluation of Apriori and FP-Growth Algorithms". International Journal of Computer Applications. 79. 34-37. DOI: 10.5120/13779-1650.

Zimmerman, A. and Blythe, J. 2013. "Business to Business Marketing Management". Routledge. 19-26. DOI: 10.4324/9780203067581.

APPENDIX A

PYTHON LIBRARIES UTILIZED IN THE STUDY

This section provides an overview of the Python libraries that played a pivotal role in the execution and analysis of the research presented in this thesis. Python, a versatile and widely adopted programming language, offers an extensive ecosystem of libraries that facilitate various aspects of data collection, processing, analysis, and visualization. The selection of libraries was tailored to the specific needs and objectives of this study, ensuring robustness and reproducibility.

```
import pandas as pd
import datetime as dt
import numpy as np
import plotly.express as px
import seaborn as sns
import matplotlib.pyplot as plt

from kneed import KneeLocator
from sklearn.cluster import KMeans
from statsmodels.stats.multicomp import pairwise_tukeyhsd
from scipy.stats import f_oneway, zscore, pearsonr
from IPython.display import display, HTML

import time
from mlxtend.frequent_patterns import apriori, fpgrowth, association_rules
```

Figure A.1 Python libraries used in this study.

Figure A.1 shows Python libraries used in the study this include:

- Pandas: pandas, a versatile Python library for data manipulation and analysis, played a pivotal role in this study. It provided a robust framework for handling structured data through its two primary data structures: DataFrames and Series. DataFrames,
- Datetime: The datetime module, a core component of Python's standard library, played a crucial role in managing date and time-related operations within this

study. It provided the means to create, manipulate, and format date and time objects.

- NumPy: NumPy, a cornerstone library for scientific computing in Python, was an essential component of this study. It introduced efficient data structures for handling arrays and matrices, enabling swift and optimized numerical computations.
- Plotly.express: The plotly.express module is a powerful library that was utilized for interactive data visualization in this study. Building upon the Plotly library, plotly.express simplifies the creation of a wide range of interactive charts, graphs, and plots.
- seaborn: The seaborn module, built on top of Matplotlib, played a vital role in enhancing the visual aesthetics and statistical insights of this study's data visualizations.
- Matplotlib.pyplot: The matplotlib.pyplot module, often referred to as pyplot, served as the primary tool for creating static data visualizations in this study. It is a core component of the Matplotlib library, a widely used Python plotting library.
- KneeLocator: The KneeLocator class is part of the kneed library, and it provides a valuable tool for automatically detecting the "knee" point or "elbow" point in a dataset. In the context of data analysis, the knee point often represents a critical change in the data distribution or behavior. This class is particularly useful for determining the optimal number of clusters or groups in clustering algorithms like k-means.
- KMeans: The KMeans class is a fundamental component of the scikit-learn library (sklearn) and is used for performing K-Means clustering, a popular unsupervised machine learning technique. K-Means clustering is employed to group similar data points into clusters based on their similarity, with the number of clusters specified in advance.

- Pairwise_tukeyhsd: The pairwise_tukeyhsd function is a part of the statsmodels library and is used for performing Tukey's Honestly Significant Difference (HSD) test, which is a post hoc test commonly applied after an analysis of variance (ANOVA). Tukey's HSD test is designed to identify which specific groups or treatments differ significantly from each other when you have multiple groups to compare.
- F_oneway: The f_oneway function is part of the scipy.stats module and is used for performing one-way analysis of variance (ANOVA). ANOVA is a statistical test that helps determine if there are statistically significant differences in the means of multiple groups. In the context of this study, it may have been employed to compare the means of several groups to assess whether there are significant differences between them.
- Zscore: The zscore function is also part of the scipy.stats module. It is used for calculating the z-scores of data points in a dataset. Z-scores are a measure of how many standard deviations a data point is away from the mean of the dataset. This function may have been used for data normalization or standardization to compare data points on a common scale.
- Pearsonr: The pearsonr function is used to compute the Pearson correlation coefficient and its associated p-value to assess the strength and direction of a linear relationship between two sets of data. It measures the degree to which two variables are linearly related. In your study, it may have been used to analyze the correlation between two variables and determine if the correlation is statistically significant.
- Display: The display function is used to render and display various types of content, such as dataframes, plots, or multimedia elements, directly within the Jupyter Notebook environment. It allows you to present output or visualizations inline with your code.

- HTML: The HTML class is used to render and display HTML content within a Jupyter Notebook cell. It can be handy when you want to display custom HTML content, including formatted text, tables, or interactive widgets.
- Time module: The time module is part of Python's standard library and provides various time-related functions and methods. It allows you to work with time, measure time intervals, and perform various time-related operations.
- Apriori: apriori is a function used for association rule mining, a technique commonly used in market basket analysis and recommendation systems. It is used to find frequent itemsets in transactional datasets. The function generates a list of itemsets and their associated support values, which can then be used to derive association rules.
- Fpgrowth: fpgrowth is another function for association rule mining, specifically using the FP-growth algorithm. FP-growth is an efficient algorithm for mining frequent itemsets and association rules. Like apriori, it identifies itemsets and their support values, but it does so more efficiently in many cases, especially with large datasets.
- Association_rules: association_rules is a class or function that is typically used after running apriori or fpgrowth. It is used to generate association rules from the frequent itemsets discovered by these algorithms. Association rules describe relationships between items in a dataset and provide insights into patterns and dependencies within the data. These rules are often used for decision-making in various applications.

BIOGRAPHY

Name	Sutiwat Tinburanakul
Date of Birth	7 October 1993
Current Address	250/189 Phaholyotin rd, Anusawaree, Bangkhen, Bangkok 10220
Education	(2017) Bachelor of Engineering – Civil Engineer, Chulalongkorn University
Academic work	None