CUKUROVA UNIVERSITY

ENGINEERING FACULTY

Computer Engineering

Data Mining for the Online Retail Industry

İbrahim Gürbüz - 2021555401

Hatice Kar - 2021555402

Ahmetcan Uzunaslan – 2020555064

Laababsi Fatima Ezzahra- 2024555E01

Github link: https://github.com/Ahmetcan2727/Data-mining

16.12.24

Abstract

The study, explores the transformative role of data mining techniques in addressing critical challenges within the e-commerce sector. By leveraging a one-year transactional dataset comprising attributes such as InvoiceNo, StockCode, Quantity, UnitPrice, and Country, the study demonstrates the potential of advanced data analytics to uncover actionable insights that enhance business decision-making and operational efficiency. The dataset also includes derived metrics such as TotalPrice and normalized features, providing a comprehensive foundation for analysis. Key methodologies employed include association rule mining (ARM) for identifying frequent product relationships, K-Means clustering for customer segmentation, and Random Forest classification for predicting customer loyalty. The dataset underwent extensive preprocessing, including the removal of missing and duplicate entries, handling of negative values, normalization of numerical features, and encoding of categorical data. These steps ensured reliability, consistency, and improved analytical accuracy. ARM revealed both strong and weak product associations, offering valuable guidance for cross-selling strategies, product bundling, and inventory optimization. K-Means clustering provided granular customer segmentation based on Recency, Frequency, and Monetary (RFM) metrics, enabling targeted marketing campaigns, re-engagement strategies for dormant users, and loyalty initiatives for high-value customers. Random Forest classification demonstrated high accuracy in categorizing customers into loyalty tiers, further refining strategies for customer engagement and retention. Results underscore the strategic value of data mining in the online retail industry, showcasing its ability to extract meaningful patterns and relationships from complex datasets. Practical applications include personalized marketing, inventory management, and customer relationship optimization. While addressing limitations such as data imbalance, the study highlights future directions, including the integration of additional features, expanded datasets, and exploration of alternative algorithms to enhance the robustness and applicability of findings. This work emphasizes the critical role of data mining in driving innovation and growth in the competitive e-commerce landscape.

Keywords: Data Mining, E-commerce, Customer Segmentation, Predictive Analytics

List of Contents

ABSTRACT I

LİST OF CONTENTS II

LIST OF FIGURES IV

LİST OF TABLES V

LİST OF CODES VI

1. INTRODUCTION 1

1.1 Subject of the Project 1

1.2 Purpose of the Project 1

1.3 Scope of the Project 2

2. MATERİALS AND METHODS 4

2.1 Dataset Characteristics 4

2.1.1 Overview of Attributes 4

2.1.2 Scope and Size of the Dataset 4

2.2 Data Cleaning and Feature Engineering 5

2.2.1 Handling Missing Values in the Dataset 5

2.2.2 Handling Duplicate Rows 6

2.2.3 Handling Negative Values in the Dataset 7

2.2.4 Adding a New Feature to the Dataset 7

2.3 Understanding Normalization 8

2.3.1 Types of Normalization Techniques 8

2.3.2 Encoding Categorical Data: Country Column 11

2.4.2 Customer Segmentation Using K-Means Clustering 13

2.4.3 Customer Classification 14

3. EXPERİMENTAL WORK 16

3.1 Association Rule Mining: Insights into Product Relationships 16

3.1.1 Strongest Product Relationships 17

3.1.2 Weaker Product Relationships 18

3.2 RFM-based clustering to identify customer segments 18

3.2.1 Loading the Dataset 18

3.2.2 Calculating RFM Metrics 19

3.2.3 Outlier Detection 20

3.2.4 K-Means Algorithm and Implementation 21

3.3 Random Forest classifier for customer loyalty prediction 24

4. RESULTS 25

4.1 Regional Variations in Product Relationships 25
4.2 Cluster visualizations and customer segmentation patterns 27
4.3 Classification metrics 30
5. DİSCUSSİON 32
5.1 Insights from Association Rule Mining 32
5.2 Strategic Customer Segmentation Using K-Means Clustering 33
5.3 Random Forest in Customer Loyalty Classification: Insights and Future Directions 34
6. CONCLUSİON AND FUTURE WORK 35
6.1 Conclusion 35
6.2 Future Work 36
REFERENCES 37

List of Figures

Figure 1 How random forest algorithm works 16
Figure 2 Cluster Visualization for Recency vs. Monetary 27
Figure 3 Cluster Size Distribution for K=3 and K=5 28
Figure 4 3D RFM Cluster Visualization for K=3 and K=5 29
Figure 5 Graph of the final result 32

List of Tables

Table 1 Min and Max Values Before Normalization *10*
Table 2 Metrics Comparison Before and After Normalization *10*
Table 3 Country Code Mapping *12*
Table 4 Strongest Product Relationships Identified by ARM *17*
Table 5 Weaker Product Relationships Identified by ARM *18*
Table 6 RFM Metrics Dataset for Customer Segmentation *20*
Table 7 Regional Product Relationship Metrics and Most Related Items *26*
Table 8 Result of Random Forest Algorithm 1 ...... *30*
Table 9 Result of Random Forest Algorithm 2 ...... *30*
Table 10 Results of Classification Algorithms *31*
Table 11 Result of Random Forest Algorithm 3 ...... *31*

List of Codes

Code 1 Reading and Displaying the Excel Dataset using Pandas *19*
Code 2 Calculating Customer Metrics for RFM Analysis in Retail Data *20*
Code 3 Outlier Removal Using Z-Scores for RFM Metrics *21*
Code 4 Implementation of K-Means Clustering Algorithm with Utility Functions *22*
Code 5 Centroid Calculation and Convergence Check in K-Means Clustering *23*
Code 6 RFM Scoring and Customer Loyalty Classification *24*
Code 7 Customer Loyalty Classification with Random Forest *25*

1. Introduction

1.1 Subject of the Project

The online retail industry has experienced unprecedented expansion over the last two decades, fueled by advancements in technology and the global proliferation of internet access. E-commerce platforms have emerged as a dominant force in commerce, breaking geographical barriers and enabling retailers to reach diverse consumer bases worldwide. This growth, while presenting enormous opportunities, also brings significant challenges, such as deciphering complex customer behavior, managing dynamic inventory demands, and fostering customer loyalty in a competitive market (Muralidhar & Lakkanna, 2024).

To navigate these challenges effectively, businesses must harness and analyze the vast volumes of transactional data generated daily. Such data holds the key to understanding purchasing trends, identifying customer preferences, and tracking product performance. However, the sheer size and complexity of these datasets make manual analysis impractical. This is where data mining techniques become indispensable, providing the means to uncover patterns, correlations, and actionable insights embedded within the data (Temirov & Dongxiao, 2018).

Data mining is not merely a tool but a transformative approach that enables businesses to move beyond surface-level observations. By leveraging sophisticated algorithms, businesses can predict customer needs, optimize stock levels, and design targeted marketing campaigns. The insights gained through data mining empower retailers to make strategic, data-driven decisions that enhance operational efficiency, improve customer satisfaction, and maintain a competitive edge in the fast-paced online retail landscape (Rajeshkumar & Rajakumari, 2023).

1.2 Purpose of the Project

This project seeks to highlight the transformative potential of data mining techniques in converting vast volumes of raw transactional data into actionable insights within the online retail industry. By systematically analyzing purchasing behaviors and product interactions, the project aims to provide retailers with strategic tools to optimize decision-making processes and enhance operational efficiency. The ultimate goal is to address critical industry challenges, such as improving customer engagement, driving revenue growth, and streamlining inventory management.

A key focus of the project is the application of association rule mining to uncover hidden relationships between products. This technique enables retailers to identify patterns in customer purchases, such as frequently co-purchased items, which can be leveraged to design effective cross-selling and upselling strategies. These insights help businesses recommend complementary products, increase basket sizes, and optimize inventory placement to meet customer demands more effectively.

Additionally, the project employs K-Means clustering and Random Forest classification to segment and classify customers based on their purchasing behaviors and loyalty. By grouping customers into distinct segments using K-Means, retailers can tailor marketing strategies to specific needs, such as targeting high-value customers with personalized offers or re-engaging dormant users. Random Forest further enhances this approach by categorizing customers by their value to the business, enabling focused efforts to nurture loyalty and maximize lifetime value. Together, these techniques underscore the crucial role of data mining in addressing industry complexities and driving both operational and strategic success.

1.3 Scope of the Project

The scope of this project is divided into three main areas, beginning with the dataset. The dataset used spans one year of transactional data, from December 2010 to December 2011, encompassing attributes such as InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerID, and Country. In addition to these core attributes, derived metrics such as TotalPrice (calculated as Quantity × UnitPrice) and normalized features are included to enhance the analytical process. These metrics provide a comprehensive foundation for evaluating customer behavior and product performance.

The methodological scope involves robust preprocessing and the application of advanced data mining techniques. Preprocessing steps include cleaning missing data, eliminating duplicate entries, and addressing negative quantities to ensure the dataset's reliability and accuracy. This is followed by the implementation of key techniques: the Apriori algorithm is used for association rule mining to uncover frequent product pairings; K-

Means clustering is employed to segment customers based on Recency, Frequency, and Monetary (RFM) metrics; and Random Forest is utilized for classifying customers by loyalty and value, facilitating targeted marketing strategies.

The analytical focus of the project is to extract actionable insights that inform business strategies. By identifying frequent product pairings, the project highlights opportunities for cross-selling and inventory optimization. Customer segmentation results in distinct groupings such as high-value customers, dormant users, and occasional buyers, which enable tailored engagement approaches. Additionally, classification insights guide the development of marketing campaigns aimed at boosting customer retention and revenue. Together, these analyses demonstrate the practical impact of data mining in addressing operational and strategic challenges in the online retail industry.

2. Materials and Methods

2.1 Dataset Characteristics

2.1.1 Overview of Attributes

The dataset utilized in this project captures transactional data from an online retail platform, offering a view of customer purchasing behavior and product performance. Key attributes include:

- **InvoiceNo**: A unique identifier for each transaction, which helps track individual sales and group related purchases.
- **StockCode**: A distinct code assigned to each product, enabling detailed performance analysis and inventory tracking.
- **Description**: A textual description of the product, useful for identifying items and performing text-based analyses, such as keyword extraction.
- **Quantity**: The number of units purchased in each transaction, providing insights into demand patterns and inventory requirements.
- **InvoiceDate**: A timestamp for each transaction, critical for conducting time-series analyses to identify seasonal trends and patterns.
- **UnitPrice**: The price per unit of a product, essential for calculating revenue and understanding pricing strategies.
- **CustomerID**: A unique anonymized identifier for each customer, enabling segmentation and the analysis of purchasing behavior.
- **Country**: The geographic location of the customer, useful for regional trend analysis and market segmentation.
- **Derived Metrics**: The dataset includes calculated attributes such as TotalPrice (Quantity × UnitPrice) and normalized values for attributes like Quantity and TotalPrice to improve comparability across items and transactions.

2.1.2 Scope and Size of the Dataset

The dataset spans a full year of retail transactions, from December 1, 2010, to December 9, 2011, providing a comprehensive view of customer activity and product performance over time. With several thousand unique transactions, the dataset serves as a robust foundation for statistically reliable analyses. Each entry documents essential details such as products purchased, transaction timestamps, and customer identifiers, allowing for in-depth studies of purchasing patterns, product relationships, and customer behaviors. The inclusion of multiple attributes and calculated metrics further enhances the dataset's analytical potential.

The dataset also covers a wide geographic range, with transactions recorded from customers in multiple countries. This diversity facilitates cross-regional comparisons, helping to identify market-specific trends and opportunities for business expansion. Furthermore, the dataset's temporal coverage captures seasonal variations and significant shopping events, such as holiday surges, which are critical for understanding time-dependent buying behaviors. The combination of multivariate data, extensive coverage, and sequential organization makes this dataset exceptionally suitable for implementing advanced data mining techniques aimed at optimizing business strategies in the online retail sector.

2.2 Data Cleaning and Feature Engineering

Upon reviewing the dataset initially, several tasks were conducted to understand its structure and identify issues requiring attention before further analysis. The first step involved examining the first five rows of the dataset to gain an overview of its contents and structure. The dataset included several columns related to transaction details, such as CustomerID, Description, Quantity, UnitPrice, StockCode, and Country. Subsequently, the data types of these columns were reviewed to ensure consistency. It was observed that the data types were appropriate for their expected values—for instance, numerical columns like Quantity and UnitPrice had numerical data types, while CustomerID and Description were stored as categorical or string types.

2.2.1 Handling Missing Values in the Dataset

One crucial aspect of data cleaning involves addressing missing values, as they can significantly affect analysis. The dataset was thoroughly assessed to identify such gaps in each column. It was found that the CustomerID column had a considerable number of missing entries. Since this column serves as a reference key for customer-related data, its absence posed significant challenges for customer segmentation and transaction analysis.

Meanwhile, the Description column showed fewer missing values. Though its impact on the dataset was less pronounced, attention was still needed to address these omissions effectively (Qi et al., 2018).

To handle these missing values, various strategies, such as imputation or removal, were considered. Given the CustomerID column's critical role, imputing values was avoided to prevent inconsistencies or inaccuracies. Instead, rows with missing CustomerID entries were removed to maintain data integrity. For the Description column, the missing values were also removed since their proportion was small, and excluding these entries did not significantly affect the dataset's overall structure or utility.

After addressing the missing values, the dataset was reduced to 4,068,828 rows and 8 columns. This adjustment resulted in a cleaner dataset, ensuring greater reliability for subsequent analysis while maintaining consistency and accuracy in the data.

2.2.2 Handling Duplicate Rows

To ensure the dataset's reliability and accuracy, a detailed investigation into the presence of duplicate rows was carried out during the data cleaning process. Duplicate rows, defined as entries where all column values were identical, can significantly distort analytical results if left unaddressed. The focus of this step was on identifying and removing complete duplicates rather than partial similarities, such as matching prices or stock codes across different transactions, as these do not indicate true redundancy in the context of this dataset (Subramaniyaswamy & Pandian, 2012).

The analysis revealed a total of 5,225 duplicate rows, which were promptly removed. This step was necessary to maintain the dataset's integrity and ensure that the results of subsequent analyses would not be skewed by redundant information. Removing these duplicates also helped optimize the dataset, reducing unnecessary noise and enhancing its overall quality. This process not only eliminated redundancies but also ensured that the remaining data provided an accurate representation of the transactional information.

After addressing the issue of duplicate rows, the dataset was reduced to 4,016,604 rows, while the number of columns remained unchanged at 8. This refined version of the dataset is now more efficient, well-organized, and prepared for further analytical tasks, including advanced techniques such as classification and clustering. By eliminating redundant data, the dataset is better suited to deliver meaningful insights and facilitate more precise modeling and decision-making.

2.2.3 Handling Negative Values in the Dataset

The dataset was carefully examined for the presence of negative values in numerical columns, as these can significantly impact analysis. Special attention was given to the UnitPrice and Quantity columns, which play a vital role in understanding sales and revenue patterns. Negative values in these fields can indicate anomalies, such as data entry errors or the recording of returns, which need to be addressed to ensure the dataset's reliability (Learn Statistics Easily, 2024).

No negative values were identified in the UnitPrice column, which aligned with expectations since prices in a retail context are inherently non-negative. Consequently, no adjustments were necessary for this column. However, the Quantity column revealed 8,905 rows with negative values. These entries likely stemmed from product returns or inconsistencies in data entry. To maintain the integrity of the dataset, all rows with negative quantities were removed.

After addressing these anomalies, the dataset was reduced to 3,979,924 rows, while retaining its 8 columns. By removing transactions with negative quantities, the dataset was refined to more accurately reflect sales and revenue data, making it suitable for further analysis and ensuring the reliability of insights derived from it.

2.2.4 Adding a New Feature to the Dataset

To improve revenue calculations and facilitate more effective analysis, a new feature called TotalPrice was added to the dataset. This column was created by multiplying the values in the Quantity and UnitPrice columns, allowing for a straightforward calculation of the total revenue generated by each transaction. By including this feature, the dataset offers a clearer perspective on revenue trends and enables the identification of patterns, such as total sales by specific products or customers.

With the addition of the TotalPrice column, the dataset now consists of 3,927,732 rows and 9 columns. This enhancement not only improves the dataset's usability for revenue-focused analysis but also ensures that it is well-prepared for advanced analytical methods such as exploratory data analysis, classification, and clustering. The refined dataset, with its improved structure and enriched features, is now more reliable and ready for in-depth modeling and decision-making tasks.

2.3 Understanding Normalization

Normalization is the process of adjusting the scales of numerical features to ensure uniformity and improve computational efficiency. It plays a critical role in addressing variations in feature scales, which can otherwise complicate data analysis and negatively impact model performance. By scaling data, normalization reduces the influence of outliers, stabilizing statistical calculations and improving the reliability of results. Additionally, it enhances model performance by accelerating the training process and achieving higher accuracy. This process is particularly important for optimizing the performance of analytical models. Furthermore, normalization works alongside encoding techniques for categorical variables to ensure clarity and compatibility with machine learning models, addressing their specific input requirements (Wang et al., 2020).

2.3.1 Types of Normalization Techniques

Normalization is a crucial step in data preprocessing, aimed at transforming numerical features to a consistent scale. This process ensures that features contribute equally to the analysis and prevents dominance by features with larger magnitudes. Below are three widely used normalization techniques (Jaiswal, 2024):

1. **Min-Max Normalization:**

   This technique scales data to a predefined range, typically between 0 and 1. It preserves the relationships between values by adjusting them proportionally within the specified range. The formula for this method is:

   $$X = (X-Xmin)/(Xmax-Xmin)$$

Min-Max Normalization is intuitive and computationally efficient, making it suitable for datasets with features that have comparable distributions and without significant outliers.

2. **Z-Score Normalization:**
   Also known as standardization, this method centers the data around the mean and scales it based on the standard deviation. The formula is:
   $$Z=(X-\mu)/\sigma$$
   This technique is particularly effective when features follow a normal distribution and ensures the mean becomes zero and the standard deviation is one. It is commonly used in scenarios where relative deviations from the mean are significant.

3. **Decimal Scaling:**
   Decimal scaling normalizes values by dividing them by a power of 10, determined by the maximum absolute value in the dataset. This method is less commonly used but is useful when the dataset's numerical range varies significantly across features.

Selected Approach: Min-Max Normalization

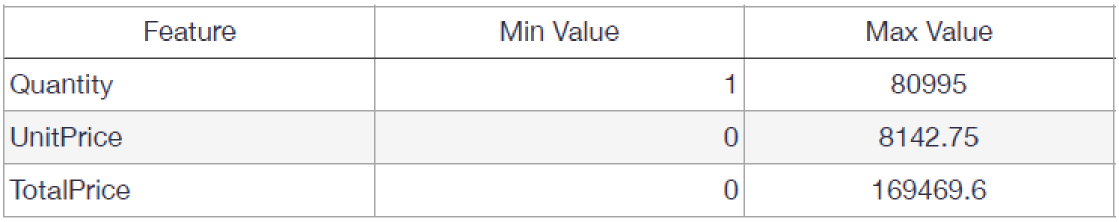
Min-Max Normalization was chosen for this dataset due to its simplicity and efficiency, making it a particularly suitable method for preprocessing. This technique ensures uniform scaling across key features, such as Quantity, UnitPrice, and TotalPrice, allowing these variables to be directly comparable. By limiting the range of values to a predetermined scale, typically between 0 and 1, Min-Max Normalization simplifies computations and reduces the complexity of operations on large datasets. Additionally, this approach minimizes the impact of outliers, preventing extreme values from disproportionately affecting the overall analysis. Importantly, the method preserves the relationships between values, maintaining the integrity of the data while enhancing its usability. Its practicality and effectiveness make it an ideal choice for the dataset under consideration.

Application of Min-Max Normalization

The dataset contains three primary numerical features critical for analysis:

- Quantity, ranging from 1 to 80,995
- UnitPrice, spanning from 0.00 to 8,142.75
- TotalPrice, calculated from the other two features, extending from 0.00 to 169,469.60

Applying Min-Max Normalization transformed these features to a uniform scale between 0 and 1. This transformation retained the relative relationships among values while ensuring they operated within a consistent range. For example, a Quantity value of 80,995 was scaled to 1, while the lowest value, 1, was scaled to 0. Similarly, all UnitPrice and TotalPrice values followed this range adjustment.

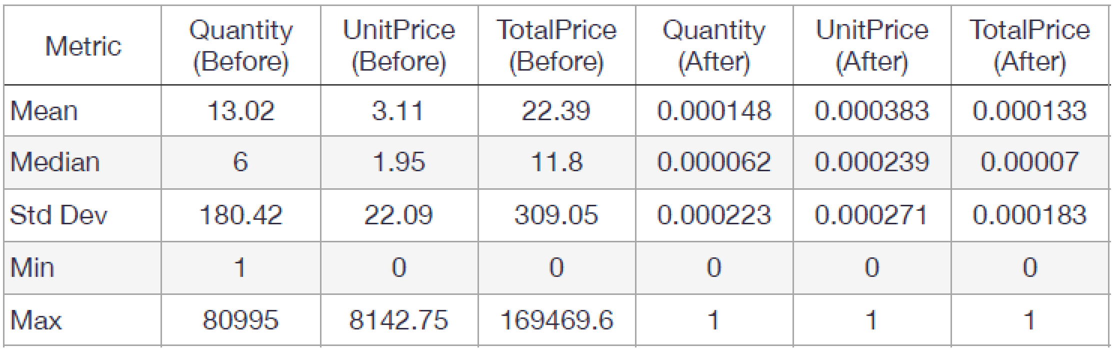


Feature	Min Value	Max Value
Quantity	1	80995
UnitPrice	0	8142.75
TotalPrice	0	169469.6

Table 1 Min and Max Values Before Normalization

Normalization Results and Enhanced Dataset Utility

After applying normalization, all numerical values in the dataset were scaled to a uniform range between 0 and 1. This transformation ensured consistency across features, making them directly comparable and easier to analyze. The results of normalization demonstrated a uniform scale, which not only streamlined the data for analysis but also enhanced the performance of machine learning models by mitigating issues related to varying feature magnitudes and improving computational efficiency.



Metric	Quantity (Before)	UnitPrice (Before)	TotalPrice (Before)	Quantity (After)	UnitPrice (After)	TotalPrice (After)
Mean	13.02	3.11	22.39	0.000148	0.000383	0.000133
Median	6	1.95	11.8	0.000062	0.000239	0.00007
Std Dev	180.42	22.09	309.05	0.000223	0.000271	0.000183
Min	1	0	0	0	0	0
Max	80995	8142.75	169469.6	1	1	1

Table 2 Metrics Comparison Before and After Normalization

2.3.2 Encoding Categorical Data: Country Column

The dataset's Country column, which contains 37 unique entries, was encoded into numeric codes to enhance readability and compatibility with machine learning algorithms. Each country was assigned a unique three-digit code, such as 001, 002, and so on, up to 037. This encoding approach simplifies data operations, such as filtering and grouping, while preserving the dataset's clarity. By transforming textual entries into numerical values, the encoded column becomes more suitable for computational tasks and ensures seamless integration into analytical models.

Country	Code
United Kingdom	001.
France	002.
Australia	003.
Netherlands	004.
Germany	005.
Norway	006.
EIRE	007.
Switzerland	008.
Spain	009.
Poland	010.
Portugal	011.
Italy	012.
Belgium	013.
Lithuania	014.
Japan	015.
Iceland	016.
Channel Islands	017.
Denmark	018.
Cyprus	019.
Sweden	020.
Austria	021.
Israel	022.
Finland	023.
Greece	024.
Singapore	025.
Lebanon	026.
United Arab Emirates	027.
Saudi Arabia	028.
Czech Republic	029.
Canada	030.
Unspecified	031.
Brazil	032.
USA	033.
European Community	034.
Bahrain	035.
Malta	036.
RSA	037.

Table 3 Country Code Mapping

The encoding process was implemented systematically to avoid duplication and maintain accuracy. First, all unique country names were identified using Excel tools. Then, sequential numeric codes were assigned to each country, ensuring no duplicates. Finally, the Country column was updated by replacing the textual country names with their corresponding numeric codes. This transformation not only streamlined the dataset but also made it more efficient for operations like grouping and filtering, while keeping the data unambiguous and easy to interpret.

2.4 Algorithms and Techniques

2.4.1 Association Rule Mining

Overview of the Apriori Algorithm

The Apriori algorithm is a robust and widely used technique in association rule mining, designed to uncover patterns and relationships in large datasets. It leverages the principle of downward closure, which states that if a specific itemset is frequent, all of its subsets must also be frequent. This characteristic allows the algorithm to efficiently reduce the search space, focusing only on relevant itemsets. In retail data, the Apriori algorithm is particularly effective for identifying frequently purchased product combinations, enabling businesses to create data-driven marketing strategies, optimize store layouts, and refine inventory management practices (Jadhav et al., 2023).

Steps for Identifying Frequent Itemsets and Generating Rules

The algorithm begins by transforming the dataset into a transactional format, where each record represents a set of purchased items. In the first phase, it generates candidate itemsets of length one and calculates their support values to identify those meeting the minimum threshold. Subsequent iterations extend the itemsets by combining frequent ones from the previous iteration, iteratively narrowing down the candidates until no further frequent itemsets are found. Once frequent itemsets are established, association rules are generated by partitioning them into antecedents and consequents, which are evaluated for strength and relevance (Al-Maolegi & Arkok, 2014).

The final step involves applying metrics such as Support, Confidence, and Lift to filter and rank the rules. Support quantifies how often an itemset appears in the dataset, while

Confidence measures the likelihood that a consequent will occur given the presence of an antecedent. Lift, on the other hand, evaluates the strength of an association relative to random chance, with values greater than one indicating a strong positive relationship. Together, these metrics ensure that only the most impactful and actionable rules are retained for practical application in retail strategies (Aditya, 2023).

2.4.2 Customer Segmentation Using K-Means Clustering

Customer segmentation is an essential technique in marketing and business strategies. By dividing customers into distinct groups based on their behavior, businesses can target each segment with tailored offers and strategies. In this project, we use clustering techniques to segment customers of an online retail business based on their purchasing behavior. The clustering process is driven by three metrics: Recency, Frequency, and Monetary Value (RFM). Recency refers to the number of days since a customer's last purchase, Frequency measures the number of unique transactions made by a customer, and Monetary Value represents the total amount spent by a customer. We employed the K-Means clustering algorithm to group customers into similar clusters for two values of K: K=3 and K=5, and analyzed the resulting segments.

→ But Why Clustering ?
Since customer segmentation often doesn't have predefined labels (i.e., we don't already know the customer categories in our dataset), so this task typically involves **unsupervised learning** techniques like **clustering** (in our case : K-Means)

→ Why K-Means ?
K-Means clustering is a popular and effective choice for customer segmentation due to its simplicity, scalability, and ability to handle numerical data like RFM. It groups customers into clusters based on their similarities, making it ideal for identifying distinct customer segments.

Main reasons :

1. **Efficiency**: K-Means is computationally efficient and works well with large datasets, such as the customer transaction data used in this project.
2. **Intuitive Results**: The algorithm assigns customers to clusters based on proximity to centroids, providing clear and interpretable groupings.

3. **Flexibility**: By adjusting the number of clusters (K), it can provide both broad (K=3) and detailed (K=5) segmentations.
4. **Numerical Data Handling**: K-Means performs particularly well with continuous variables like RFM metrics, making it suitable for this application.

2.4.3 Customer Classification

Classification is a technique in data mining that involves categorizing or classifying data objects into predefined classes, categories, or groups based on their features or attributes. It is a supervised learning technique that uses labelled data to build a model that can predict the class of new, unseen data. It is an important task in data mining because it enables organizations to make informed decisions based on their data. For example, a retailer may use data classification to group customers into different segments based on their purchase history and demographic data. This information can be used to target specific marketing campaigns for each segment and improve customer satisfaction.

In this project, we classify customers as high, medium, or low in terms of their loyalty to the store by analyzing their recency (when they last visited the store),frequency (how often they shop at the store), and monetary value (how much money they spend there). Here, we assumed that all purchases in the dataset were made from the same store. The aim of the project is to enable the store owner to use this classification to understand the level of customer satisfaction, take appropriate measures, and ultimately grow their business. I used Random Forest algorithm for that classification.

Before applying the Random Forest algorithm to the dataset, let's briefly review what metrics like precision, recall, F1 score and support mean in the classification results.

TP=True Positive
FP=False Positive
FN=False Negative

- Precision indicates how many of the instances predicted as positive are actually positive. High precision means that the model makes fewer false positive errors, predicting positives accurately.

$$\text{Precision} = TP / TP + FP$$

- Recall indicates how many of the actual positive instances the model correctly identified. High recall means that the model identifies a high proportion of the actual positives.

$$\text{Recall} = \text{TP} / \text{TP} + \text{FN}$$

- F1-Score is the harmonic mean of precision and recall. It is used when both high precision and high recall are important. F1-Score provides a balance between precision and recall, especially when the class distribution is imbalanced.

$$\text{F1} = 2 \text{ x } [(\text{Precision x Recall}) / \text{Precision} + \text{Recall}]$$

- Support refers to the number of actual occurrences of each class in the dataset. It indicates the number of instances belonging to each class. This helps understand the model's performance in the context of the number of examples for each class.

These metrics are often used together to evaluate the accuracy and reliability of a classification model.

Random Forest

Random forest is a technique used in modeling predictions and behavior analysis and is built on decision trees. It contains many decision trees representing a distinct instance of the classification of data input into the random forest. The random forest technique considers the instances individually, taking the one with the majority of votes as the selected prediction.

Each tree in the classifications takes input from samples in the initial dataset. Features are then randomly selected, which are used in growing the tree at each node. Every tree in the forest should not be pruned until the end of the exercise when the prediction is reached decisively. In such a way, the random forest enables any classifiers with weak correlations to create a strong classifier.

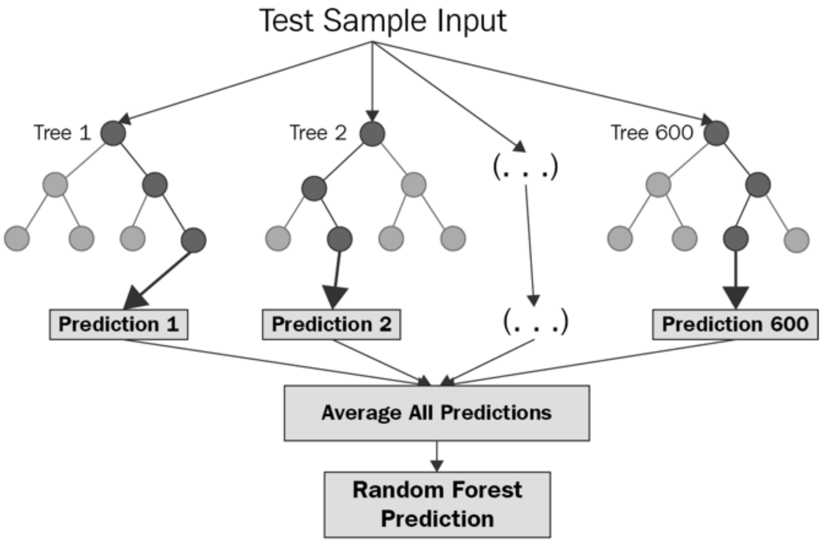


Figure 1 How random forest algorithm works

Random Forest is an excellent choice for classifying due to its key advantages:

High Accuracy: Using several decision trees, each trained on a distinct subset of the data, Random Forest aggregates their predictions. Random Forest lessens the variation associated with individual trees, resulting in predictions that are more accurate, by averaging (for regression) or voting (for classification) the predictions of these trees. When using an ensemble approach instead of a single decision tree model, accuracy is typically higher.

Overfitting Resistance: Random Forest reduces the likelihood of overfitting by combining multiple decision trees. This helps the model to generalize better and achieve improved performance on unseen data.

Parallel Processing Capability: Since decision trees are built independently, Random Forest supports parallel processing, providing a speed advantage when working with large datasets.

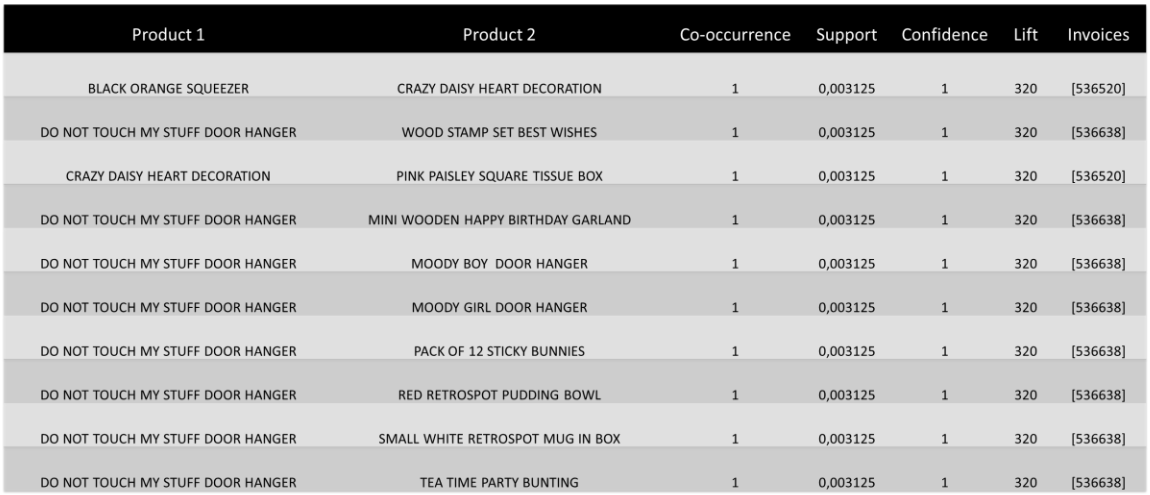
3. Experimental Work

3.1 Association Rule Mining: Insights into Product Relationships

Association Rule Mining (ARM) was applied using the Apriori algorithm to uncover meaningful product relationships within the transactional dataset. This method identifies frequent product combinations, which can inform cross-selling strategies, product bundling, and inventory optimization. The dataset included one year of transactional data with detailed attributes such as InvoiceNo, StockCode, and Quantity, providing a robust foundation for ARM. Key metrics used to evaluate the rules included **Support**, **Confidence**, and **Lift**, which measure the strength, reliability, and significance of product associations.

3.1.1 Strongest Product Relationships

Using the Apriori algorithm, high-confidence rules were generated, highlighting strong product relationships. For instance, the combination of "DO NOT TOUCH MY STUFF DOOR HANGER" and "WOOD STAMP SET BEST WISHES" demonstrated high values for Support (0.003125), Confidence (1), and Lift (320). These results signify a strong, positive correlation between these products, suggesting they are frequently purchased together. This association provides actionable insights for designing product bundles or targeted marketing campaigns. Other combinations, such as "BLACK ORANGE SQUEEZER" and "CRAZY DAISY HEART DECORATION," displayed similarly high metrics, further emphasizing opportunities for cross-selling and improved inventory planning.

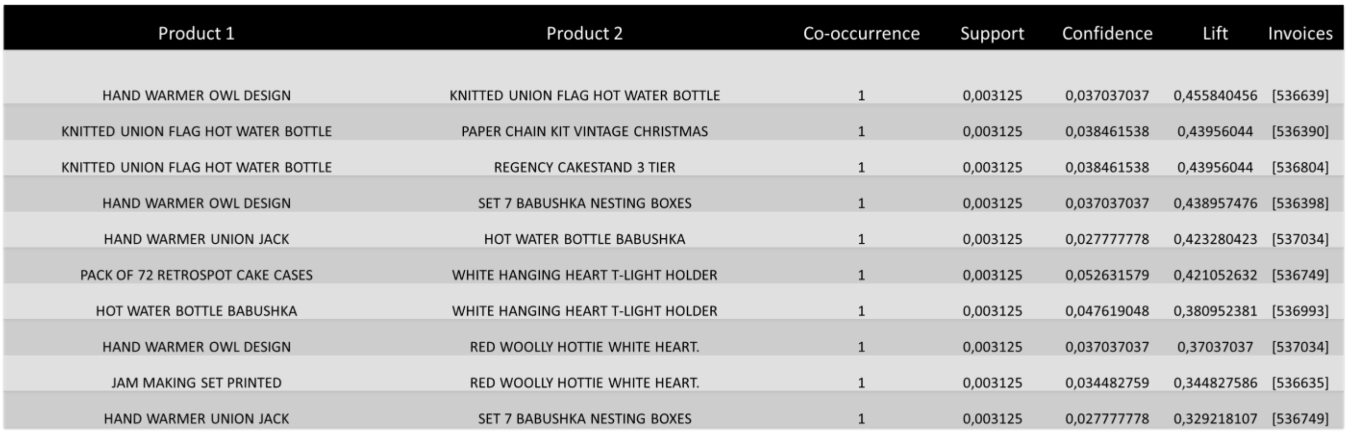


Product 1	Product 2	Co-occurrence	Support	Confidence	Lift	Invoices
BLACK ORANGE SQUEEZER	CRAZY DAISY HEART DECORATION	1	0,003125	1	320	[536520]
DO NOT TOUCH MY STUFF DOOR HANGER	WOOD STAMP SET BEST WISHES	1	0,003125	1	320	[536638]
CRAZY DAISY HEART DECORATION	PINK PAISLEY SQUARE TISSUE BOX	1	0,003125	1	320	[536520]
DO NOT TOUCH MY STUFF DOOR HANGER	MINI WOODEN HAPPY BIRTHDAY GARLAND	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	MOODY BOY DOOR HANGER	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	MOODY GIRL DOOR HANGER	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	PACK OF 12 STICKY BUNNIES	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	RED RETROSPOT PUDDING BOWL	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	SMALL WHITE RETROSPOT MUG IN BOX	1	0,003125	1	320	[536638]
DO NOT TOUCH MY STUFF DOOR HANGER	TEA TIME PARTY BUNTING	1	0,003125	1	320	[536638]

Table 4 Strongest Product Relationships Identified by ARM

3.1.2 Weaker Product Relationships

ARM also identified weaker associations, such as the pair "HAND WARMER UNION JACK" and "SET 7 BABUSHKA NESTING BOXES," which exhibited a Confidence of 0.02778 and a Lift of 0.3292. These values indicate a relatively weaker relationship, suggesting these items are infrequently purchased together. Such findings are useful in determining which product pairs may not benefit from bundling or promotional efforts. Similarly, "PACK OF 72 RETROSPOT CAKE CASES" and "WHITE HANGING HEART T-LIGHT HOLDER" showed moderate values, which can guide inventory strategies and marketing decisions with caution.



Product 1	Product 2	Co-occurrence	Support	Confidence	Lift	Invoices
HAND WARMER OWL DESIGN	KNITTED UNION FLAG HOT WATER BOTTLE	1	0,003125	0,037037037	0,455840456	[536639]
KNITTED UNION FLAG HOT WATER BOTTLE	PAPER CHAIN KIT VINTAGE CHRISTMAS	1	0,003125	0,038461538	0,43956044	[536390]
KNITTED UNION FLAG HOT WATER BOTTLE	REGENCY CAKESTAND 3 TIER	1	0,003125	0,038461538	0,43956044	[536804]
HAND WARMER OWL DESIGN	SET 7 BABUSHKA NESTING BOXES	1	0,003125	0,037037037	0,438957476	[536398]
HAND WARMER UNION JACK	HOT WATER BOTTLE BABUSHKA	1	0,003125	0,027777778	0,423280423	[537034]
PACK OF 72 RETROSPOT CAKE CASES	WHITE HANGING HEART T-LIGHT HOLDER	1	0,003125	0,052631579	0,421052632	[536749]
HOT WATER BOTTLE BABUSHKA	WHITE HANGING HEART T-LIGHT HOLDER	1	0,003125	0,047619048	0,380952381	[536993]
HAND WARMER OWL DESIGN	RED WOOLLY HOTTIE WHITE HEART.	1	0,003125	0,037037037	0,37037037	[537034]
JAM MAKING SET PRINTED	RED WOOLLY HOTTIE WHITE HEART.	1	0,003125	0,034482759	0,344827586	[536635]
HAND WARMER UNION JACK	SET 7 BABUSHKA NESTING BOXES	1	0,003125	0,027777778	0,329218107	[536749]

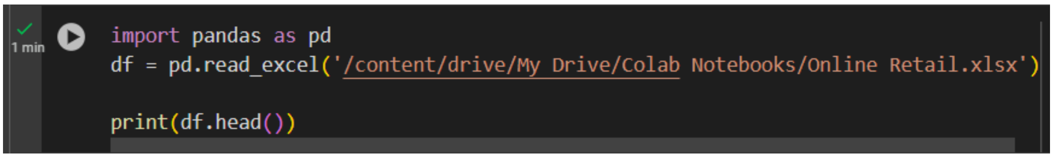
Table 5 Weaker Product Relationships Identified by ARM

By applying Association Rule Mining, the analysis highlights both strong and weak product relationships, enabling data-driven recommendations for product placement, bundling, and targeted promotions. The results provide a valuable basis for enhancing customer purchasing experiences and operational efficiency in the online retail sector.

3.2 RFM-based clustering to identify customer segments

3.2.1 Loading the Dataset

We loaded the dataset using Pandas, a powerful Python library for data analysis. The dataset was in Excel format, so we used the read_excel function to read the file into a DataFrame. This structure allows for efficient manipulation and analysis of tabular data.



```
import pandas as pd
df = pd.read_excel('/content/drive/My Drive/Colab Notebooks/Online Retail.xlsx')

print(df.head())
```

Code 1 Reading and Displaying the Excel Dataset using Pandas

The next step in the process involved creating three essential metrics—Recency, Frequency, and Monetary Value (RFM)—that summarize customer purchasing behavior. These metrics serve as the foundation for clustering and allow for meaningful segmentation of customers.

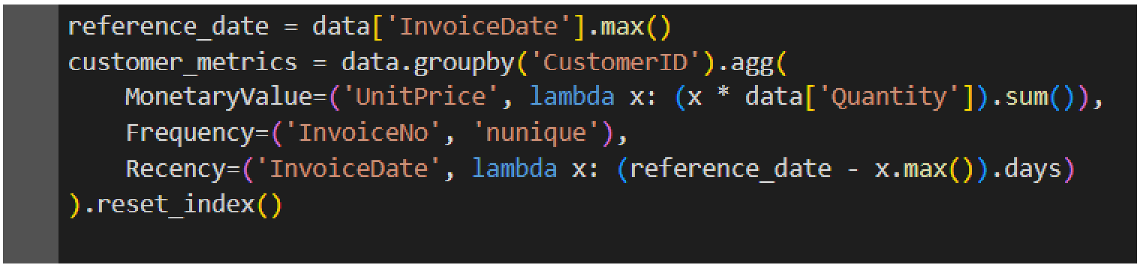
3.2.2 Calculating RFM Metrics

Recency: Recency measures the time since the customer's last transaction. To calculate this, we used the latest transaction date in the dataset as a reference point. For each customer, the number of days between their most recent purchase and the reference date was computed. This provides an indication of how recently a customer has been active.

Frequency: Frequency captures how often a customer makes purchases. It was calculated as the number of unique invoices associated with each customer. A higher frequency indicates a more regular buying pattern.

Monetary Value: Monetary Value represents the total spending by each customer. For each transaction, the unit price was multiplied by the quantity of items purchased, and the sum was aggregated for each customer. This metric identifies the financial contribution of each customer to the business.

Implementation in Code: We used the Pandas groupby function to calculate RFM metrics for each customer based on their CustomerID. The lambda functions within the agg method allowed us to compute the custom aggregations required for each metric.



```python
reference_date = data['InvoiceDate'].max()
customer_metrics = data.groupby('CustomerID').agg(
    MonetaryValue=('UnitPrice', lambda x: (x * data['Quantity']).sum()),
    Frequency=('InvoiceNo', 'nunique'),
    Recency=('InvoiceDate', lambda x: (reference_date - x.max()).days)
).reset_index()
```

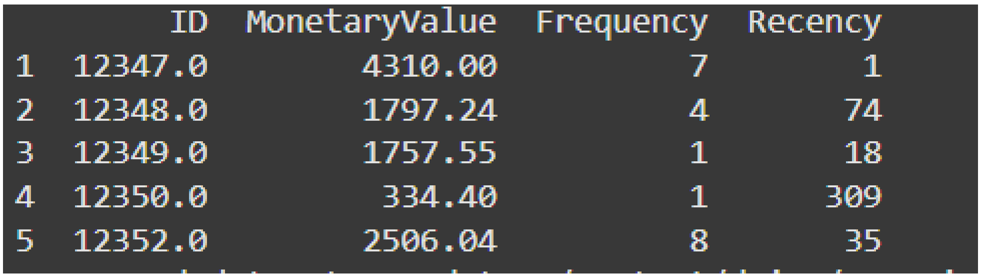
Code 2 Calculating Customer Metrics for RFM Analysis in Retail Data

These metrics provide a compact yet comprehensive summary of each customer's behavior. Recency helps identify recently active customers, Frequency highlights the regularity of their transactions, and Monetary Value showcases the profitability of each customer. By using RFM, we transform raw transactional data into actionable features that can be effectively used for clustering.

The resulting dataset contains the following columns for each customer:

- **CustomerID**: A unique identifier for each customer.
- **Recency**: Number of days since the last purchase.
- **Frequency**: Number of unique transactions.
- **Monetary Value**: Total spending amount.

This RFM dataset is now ready for further analysis and clustering.



```
        ID  MonetaryValue  Frequency  Recency
1  12347.0        4310.00          7        1
2  12348.0        1797.24          4       74
3  12349.0        1757.55          1       18
4  12350.0         334.40          1      309
5  12352.0        2506.04          8       35
```

Table 6 RFM Metrics Dataset for Customer Segmentation

3.2.3 Outlier Detection

Outliers are extreme data points that deviate significantly from other observations and can distort clustering results. For this project, we used the Z-Score method to detect and remove outliers in the RFM metrics. By filtering out these extreme values, we ensured that the clusters represent typical customer behavior without being skewed by anomalies.

To identify outliers, we calculated Z-scores for each of the RFM metrics. A Z-score indicates how many standard deviations a value is from the mean of its distribution. Typically, a Z-score greater than 3 (or less than -3) is considered an outlier, as such values fall in the extreme tails of a normal distribution. For each metric:

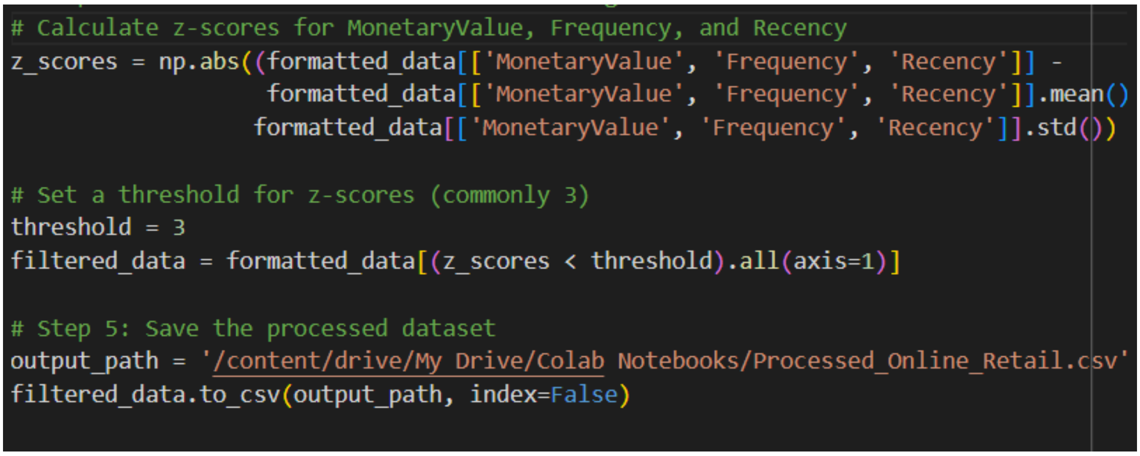
1. The mean and standard deviation were calculated.
2. The Z-score was computed for each data point as:

   Where:

   - is the data point.
   - is the mean of the metric.
   - is the standard deviation of the metric.
3. Data points with Z-scores exceeding the threshold of 3 for any metric were flagged as outliers.

Implementation in Code

We used NumPy to calculate Z-scores for the RFM metrics and filtered out rows where any metric exceeded the threshold:



```
# Calculate z-scores for MonetaryValue, Frequency, and Recency
z_scores = np.abs((formatted_data[['MonetaryValue', 'Frequency', 'Recency']] -
                   formatted_data[['MonetaryValue', 'Frequency', 'Recency']].mean()
                  formatted_data[['MonetaryValue', 'Frequency', 'Recency']].std())

# Set a threshold for z-scores (commonly 3)
threshold = 3
filtered_data = formatted_data[(z_scores < threshold).all(axis=1)]

# Step 5: Save the processed dataset
output_path = '/content/drive/My Drive/Colab Notebooks/Processed_Online_Retail.csv'
filtered_data.to_csv(output_path, index=False)
```

Code 3 Outlier Removal Using Z-Scores for RFM Metrics

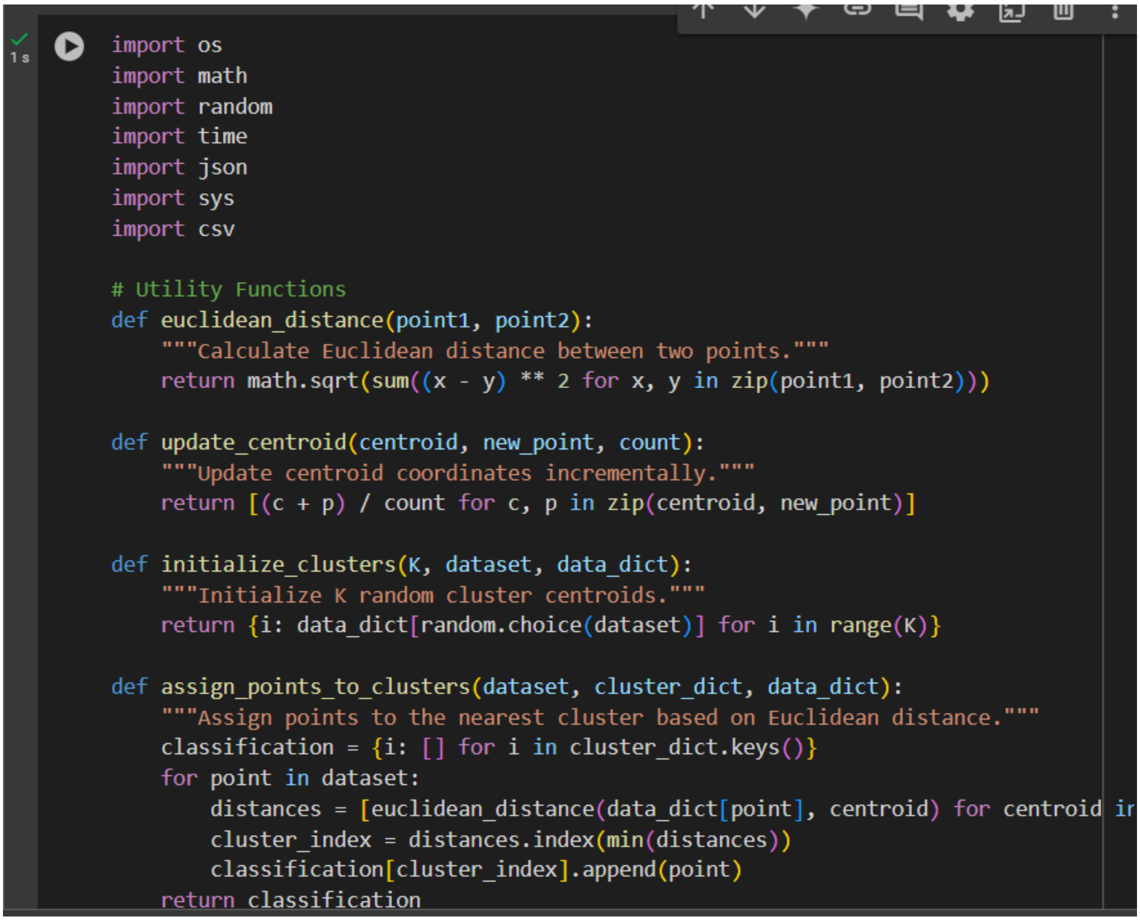
By removing these outliers, we ensured that the clusters are representative of the majority of customers and provide actionable insights.

3.2.4 K-Means Algorithm and Implementation

The K-Means algorithm is a widely used clustering method that partitions data points into distinct clusters. It iteratively assigns data points to clusters based on their proximity to centroids (cluster centers) and recalculates the centroids until convergence. This algorithm was chosen for its simplicity, efficiency, and suitability for numerical data such as RFM metrics.

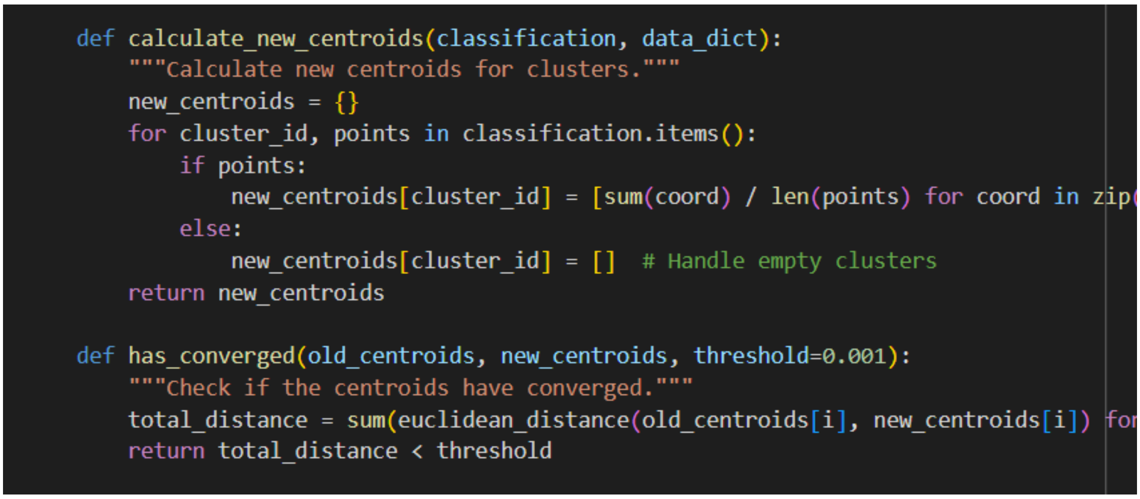
Steps in K-Means Algorithm

1. **Initialization**: Randomly select data points as the initial centroids.
2. **Assignment**: Assign each data point to the nearest centroid based on Euclidean distance.
3. **Update**: Recalculate the centroids as the mean of all points assigned to each cluster.
4. **Convergence**: Repeat steps 2 and 3 until the centroids stabilize or a predefined number of iterations is reached.



```
import os
import math
import random
import time
import json
import sys
import csv

# Utility Functions
def euclidean_distance(point1, point2):
    """Calculate Euclidean distance between two points."""
    return math.sqrt(sum((x - y) ** 2 for x, y in zip(point1, point2)))

def update_centroid(centroid, new_point, count):
    """Update centroid coordinates incrementally."""
    return [(c + p) / count for c, p in zip(centroid, new_point)]

def initialize_clusters(K, dataset, data_dict):
    """Initialize K random cluster centroids."""
    return {i: data_dict[random.choice(dataset)] for i in range(K)}

def assign_points_to_clusters(dataset, cluster_dict, data_dict):
    """Assign points to the nearest cluster based on Euclidean distance."""
    classification = {i: [] for i in cluster_dict.keys()}
    for point in dataset:
        distances = [euclidean_distance(data_dict[point], centroid) for centroid ir
        cluster_index = distances.index(min(distances))
        classification[cluster_index].append(point)
    return classification
```

Code 4 Implementation of K-Means Clustering Algorithm with Utility Functions



```
def calculate_new_centroids(classification, data_dict):
    """Calculate new centroids for clusters."""
    new_centroids = {}
    for cluster_id, points in classification.items():
        if points:
            new_centroids[cluster_id] = [sum(coord) / len(points) for coord in zip
        else:
            new_centroids[cluster_id] = []  # Handle empty clusters
    return new_centroids

def has_converged(old_centroids, new_centroids, threshold=0.001):
    """Check if the centroids have converged."""
    total_distance = sum(euclidean_distance(old_centroids[i], new_centroids[i]) for
    return total_distance < threshold
```

Code 5 Centroid Calculation and Convergence Check in K-Means Clustering

The algorithm initializes centroids randomly, assigns data points to clusters based on the shortest distance to the centroids, and recalculates centroids as the mean of points in each cluster. This process repeats until convergence or the maximum number of iterations is reached.

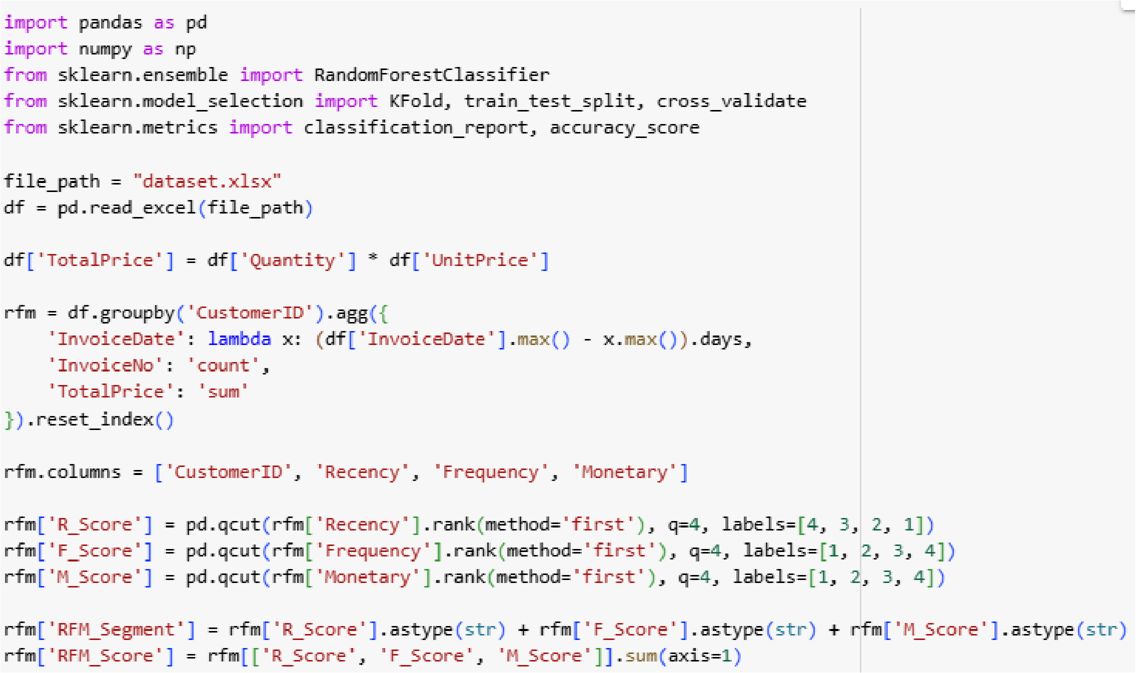
Why Two Values of K Were Chosen

Choosing the number of clusters, , is a critical step in applying K-Means. In this project, we used and to explore different levels of segmentation granularity:

- **K=3**: This simpler configuration provides broad segmentation, grouping customers into three main categories: low-value, moderate-value, and high-value customers. It is useful for businesses seeking an overview of their customer base with fewer distinctions.
- **K=5**: This configuration adds granularity, creating more detailed customer groups. For example, high-value customers can be further split into frequent high-spenders and occasional high-spenders, while low-value customers can be differentiated into dormant and occasional buyers. This level of detail supports targeted marketing strategies and niche campaigns.

3.3 Random Forest classifier for customer loyalty prediction

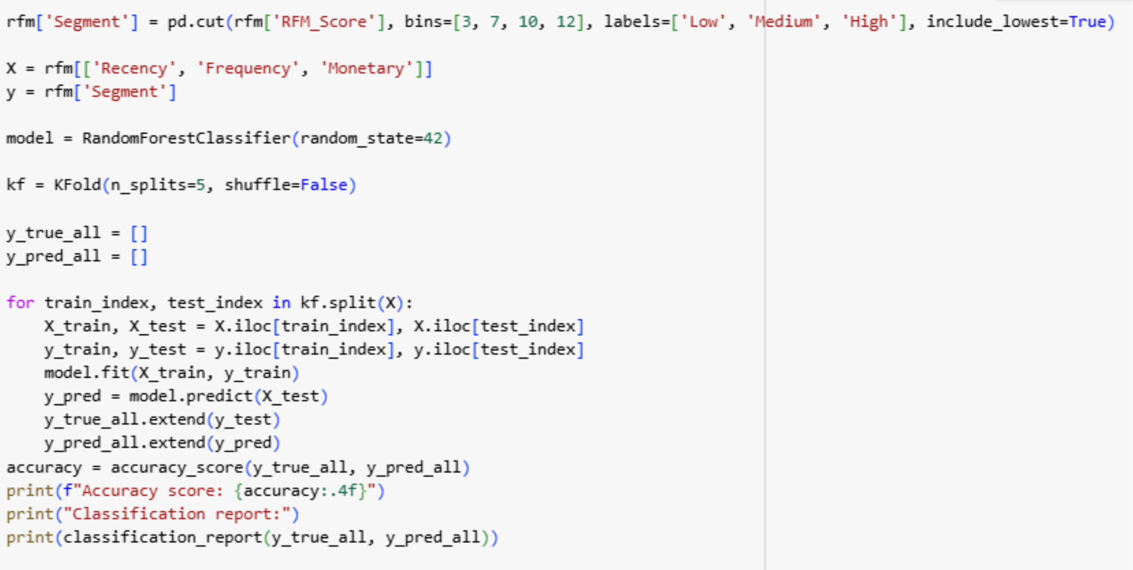
I adjusted the Recency, Frequency, and Monetary values to range between 1 and 4. For example, a customer who visited the store very recently (4), spent a lot of money (4), and shopped very frequently (4) would score 12 points, classifying them as a high loyalty customer. On the other hand, a customer who visited the store a long time ago (1), shops infrequently (2), and spends little money (2) would score 5 points, classifying them as a low loyalty customer.



```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import KFold, train_test_split, cross_validate
from sklearn.metrics import classification_report, accuracy_score

file_path = "dataset.xlsx"
df = pd.read_excel(file_path)

df['TotalPrice'] = df['Quantity'] * df['UnitPrice']

rfm = df.groupby('CustomerID').agg({
    'InvoiceDate': lambda x: (df['InvoiceDate'].max() - x.max()).days,
    'InvoiceNo': 'count',
    'TotalPrice': 'sum'
}).reset_index()

rfm.columns = ['CustomerID', 'Recency', 'Frequency', 'Monetary']

rfm['R_Score'] = pd.qcut(rfm['Recency'].rank(method='first'), q=4, labels=[4, 3, 2, 1])
rfm['F_Score'] = pd.qcut(rfm['Frequency'].rank(method='first'), q=4, labels=[1, 2, 3, 4])
rfm['M_Score'] = pd.qcut(rfm['Monetary'].rank(method='first'), q=4, labels=[1, 2, 3, 4])

rfm['RFM_Segment'] = rfm['R_Score'].astype(str) + rfm['F_Score'].astype(str) + rfm['M_Score'].astype(str)
rfm['RFM_Score'] = rfm[['R_Score', 'F_Score', 'M_Score']].sum(axis=1)
```

Code 6 RFM Scoring and Customer Loyalty Classification

Customers between 3 and 7 are low loyalty, customers between 7 and 10 are medium loyalty, and customers between 10 and 12 are high loyalty. I used 80% of my dataset for training and 20% for testing. I split the dataset into 5 folds (KFold(n_splits=5)), using each fold as a test set once, while the remaining folds serve as the training set. This method helps to evaluate the model's performance more reliably.



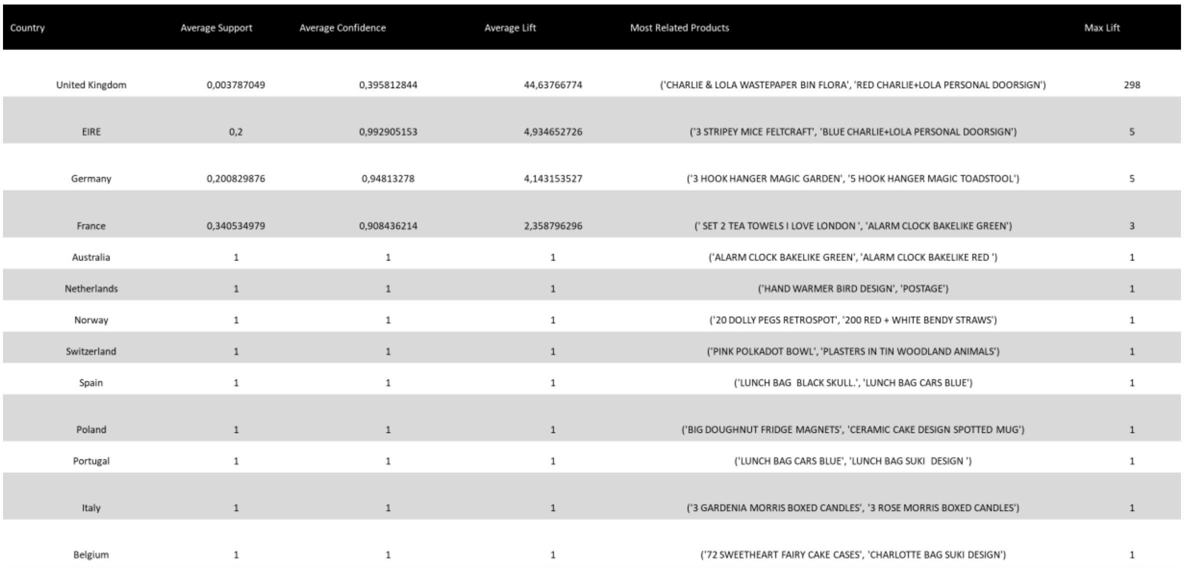
```
rfm['Segment'] = pd.cut(rfm['RFM_Score'], bins=[3, 7, 10, 12], labels=['Low', 'Medium', 'High'], include_lowest=True)

X = rfm[['Recency', 'Frequency', 'Monetary']]
y = rfm['Segment']

model = RandomForestClassifier(random_state=42)

kf = KFold(n_splits=5, shuffle=False)

y_true_all = []
y_pred_all = []

for train_index, test_index in kf.split(X):
    X_train, X_test = X.iloc[train_index], X.iloc[test_index]
    y_train, y_test = y.iloc[train_index], y.iloc[test_index]
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    y_true_all.extend(y_test)
    y_pred_all.extend(y_pred)
accuracy = accuracy_score(y_true_all, y_pred_all)
print(f"Accuracy score: {accuracy:.4f}")
print("Classification report:")
print(classification_report(y_true_all, y_pred_all))
```

Code 7 Customer Loyalty Classification with Random Forest

4. Results

4.1 Regional Variations in Product Relationships

Country-wise analysis of product relationships, highlighting key metrics such as average support, confidence, and lift values, along with the most strongly related product pairs. These metrics help identify variations in customer purchasing behaviors across regions and offer valuable insights for tailoring marketing strategies and inventory decisions. For instance, high lift values signify strong product associations, indicating opportunities for cross-selling or bundling, while lower values suggest minimal correlation between items.



Country	Average Support	Average Confidence	Average Lift	Most Related Products	Max Lift
United Kingdom	0,003787049	0,395812844	44,63766774	('CHARLIE & LOLA WASTEPAPER BIN FLORA', 'RED CHARLIE+LOLA PERSONAL DOORSIGN')	298
EIRE	0,2	0,992905153	4,934652726	('3 STRIPEY MICE FELTCRAFT', 'BLUE CHARLIE+LOLA PERSONAL DOORSIGN')	5
Germany	0,200829876	0,94813278	4,143153527	('3 HOOK HANGER MAGIC GARDEN', '5 HOOK HANGER MAGIC TOADSTOOL')	5
France	0,340534979	0,908436214	2,358796296	(' SET 2 TEA TOWELS I LOVE LONDON ', 'ALARM CLOCK BAKELIKE GREEN')	3
Australia	1	1	1	('ALARM CLOCK BAKELIKE GREEN', 'ALARM CLOCK BAKELIKE RED ')	1
Netherlands	1	1	1	('HAND WARMER BIRD DESIGN', 'POSTAGE')	1
Norway	1	1	1	('20 DOLLY PEGS RETROSPOT', '200 RED + WHITE BENDY STRAWS')	1
Switzerland	1	1	1	('PINK POLKADOT BOWL', 'PLASTERS IN TIN WOODLAND ANIMALS')	1
Spain	1	1	1	('LUNCH BAG BLACK SKULL.', 'LUNCH BAG CARS BLUE')	1
Poland	1	1	1	('BIG DOUGHNUT FRIDGE MAGNETS', 'CERAMIC CAKE DESIGN SPOTTED MUG')	1
Portugal	1	1	1	('LUNCH BAG CARS BLUE', 'LUNCH BAG SUKI DESIGN ')	1
Italy	1	1	1	('3 GARDENIA MORRIS BOXED CANDLES', '3 ROSE MORRIS BOXED CANDLES')	1
Belgium	1	1	1	('72 SWEETHEART FAIRY CAKE CASES', 'CHARLOTTE BAG SUKI DESIGN')	1

Table 7 Regional Product Relationship Metrics and Most Related Items

In the United Kingdom, the pair "CHARLIE & LOLA WASTEPAPER BIN FLORA" and "RED CHARLIE+LOLA PERSONAL DOORSIGN" demonstrated the highest lift value of 298, reflecting an exceptionally strong positive correlation. This indicates that these items are frequently purchased together, making them ideal candidates for bundling or promotional offers. Similarly, in EIRE, the pair "3 STRIPEY MICE FELTCRAFT" and "BLUE CHARLIE+LOLA PERSONAL DOORSIGN" achieved a lift value of 4.93, showing a significant association. These findings suggest the potential for highly targeted marketing campaigns in these regions to enhance sales.

France and Germany displayed moderate but meaningful product relationships. In France, the pair "SET 2 TEA TOWELS I LOVE LONDON" and "ALARM CLOCK BAKELIKE GREEN" recorded a lift value of 2.35, suggesting that while the association is less strong, it remains statistically significant. Similarly, in Germany, "3 HOOK HANGER MAGIC GARDEN" and "5 HOOK HANGER MAGIC TOADSTOOL" achieved a lift value of 4.14, highlighting notable purchasing patterns. These insights provide a foundation for regional bundling strategies, particularly during promotional periods or seasonal campaigns.

In contrast, regions like Australia, Netherlands, and Norway exhibited weaker or unremarkable associations, with lift values of 1 for all analyzed product pairs. This suggests that product relationships in these countries are equivalent to random chance, offering limited immediate potential for cross-selling or bundling strategies. However, these results present an opportunity for further exploratory research to uncover latent patterns or emerging trends in purchasing behaviors that might not be immediately apparent.

4.2 Cluster visualizations and customer segmentation patterns

Scatter Plots: Recency vs. Monetary Value

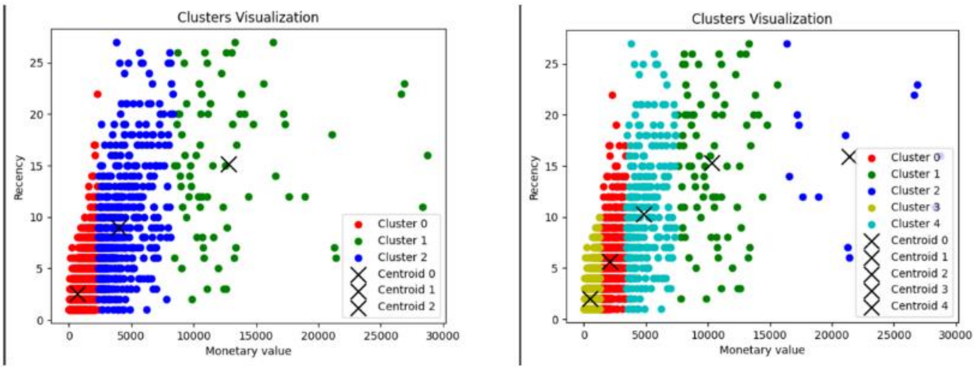


Figure 2 Cluster Visualization for Recency vs. Monetary

- **K=3:**
  - **Cluster 0 (Red):** Represents customers with very low monetary value but high recency (long time since their last purchase). These are likely **dormant or lost customers**.
  - **Cluster 1 (Blue):** Represents customers with medium monetary value and moderate recency. These could be **active, moderate spenders**.
  - **Cluster 2 (Green):** Represents customers with high monetary value and low recency (recent transactions). These are the **high-value, loyal customers**.

- **K=5:**
  - **Cluster 0 (Red):** Same as Cluster 0 in K=3, but further refined.
  - **Cluster 1 (Yellow):** Represents customers with very low monetary value but moderate recency. These could be **occasional buyers**.
  - **Cluster 2 (Blue):** Represents consistent medium-value customers, similar to Cluster 1 in K=3.
  - **Cluster 3 (Green):** Represents customers with high monetary value and very low recency, same as Cluster 2 in K=3.

- **Cluster 4 (Teal):** Represents customers with medium monetary value and higher recency. These are **inactive or declining customers**.

Bar Charts: Cluster Size Distribution

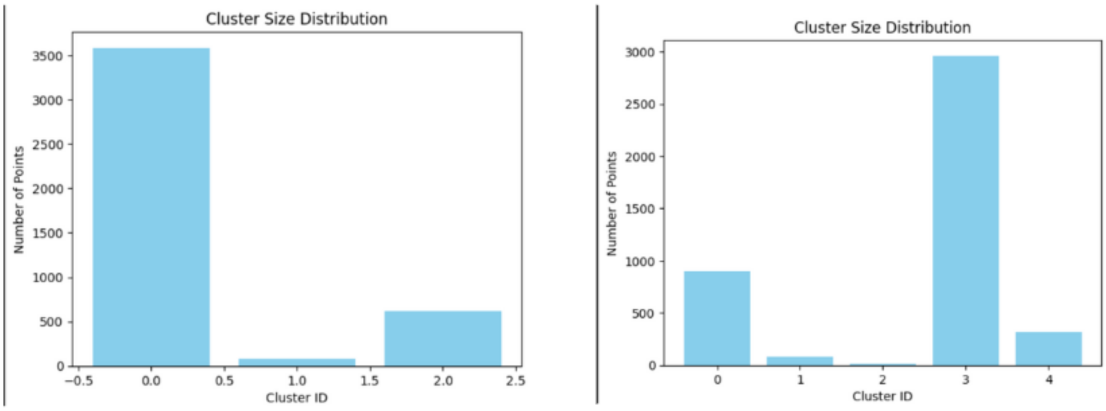


Figure 3 Cluster Size Distribution for K=3 and K=5

- **K=3:**
  - **Cluster 0:** Dominates in size, indicating a large proportion of dormant or low-value customers.
  - **Cluster 1:** Moderate size, indicating a decent number of active customers contributing medium value.
  - **Cluster 2:** Smallest cluster, showing fewer high-value customers.
- **K=5:**
  - The distribution is more spread out, with clusters representing finer customer groups:
    - **Cluster 3 (High-value customers)** remains a smaller cluster but still contributes significantly.
    - **Cluster 0 and Cluster 4 (Dormant and declining customers)** remain the largest.

3D Scatter Plots: RFM Metrics

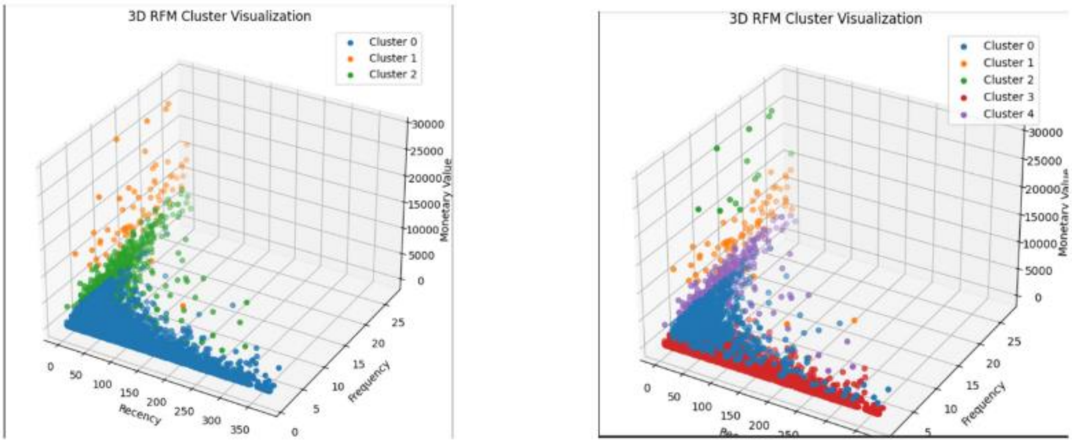


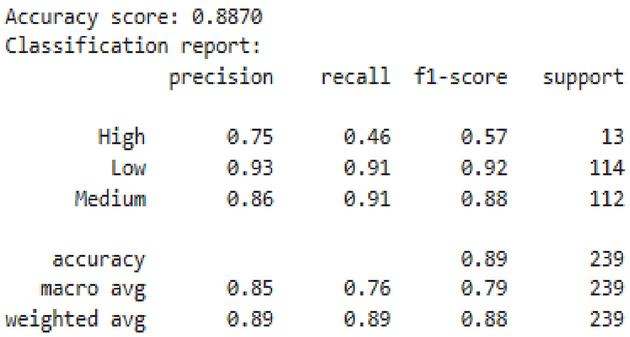
Figure 4 3D RFM Cluster Visualization for K=3 and K=5

- **K=3:**
  - The 3D visualization reinforces the separation between the clusters:
    - **Cluster 0 (Blue):** Concentrated at low monetary value and high recency.
    - **Cluster 2 (Green):** Distinctly clustered at high monetary value, low recency, and moderate frequency.
    - **Cluster 1 (Orange):** Positioned in between, representing medium-value customers.

- **K=5:**
  - Greater granularity in the 3D space:
    - **Cluster 0 and Cluster 4:** Both represent low-value customers, but with differences in recency and frequency.
    - **Cluster 3 (High-value customers):** Clearly visible as a compact, high-value group.
    - **Cluster 1 (Yellow):** Refined to represent occasional, low-frequency buyers.

4.3 Classification metrics

At the beginning, I performed classification on a dataset with 5000 rows and got the following results:

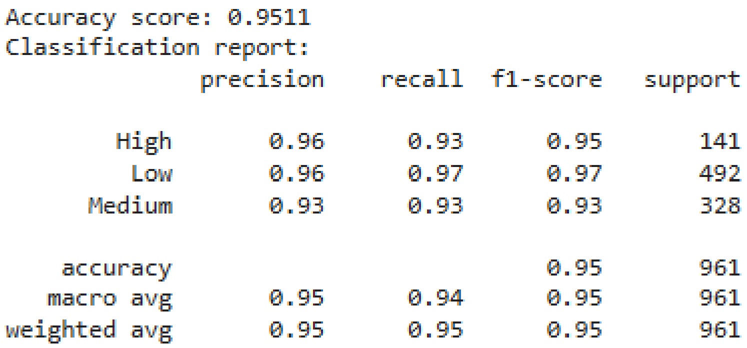


```
Accuracy score: 0.8870
Classification report:
              precision    recall  f1-score   support

        High       0.75      0.46      0.57        13
         Low       0.93      0.91      0.92       114
      Medium       0.86      0.91      0.88       112

    accuracy                           0.89       239
   macro avg       0.85      0.76      0.79       239
weighted avg       0.89      0.89      0.88       239
```

Table 8 Result of Random Forest Algorithm 1

In this graph, we can see that the accuracy for the high (loyalty) is very low. To address this imbalance, I used the SMOTE algorithm, but since the data for the high-loyalty customer class was very limited, I couldn't achieve the desired result from there. Therefore, I decided to expand my dataset. If we increase the size of our dataset, the number of high loyalty customers will also increase, which will enlarge the testing set of our model. As a result, the model will classify more accurately for testing.

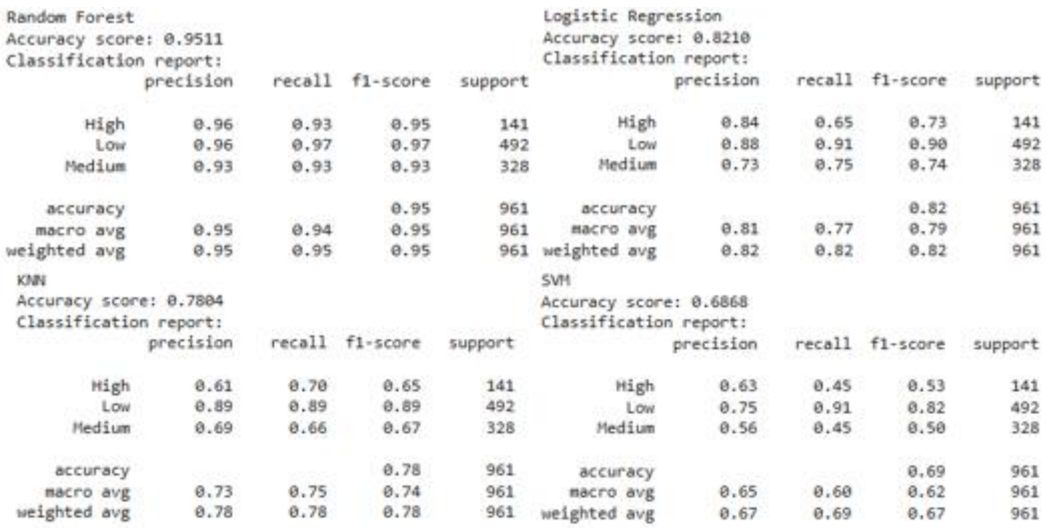
Here is the result of the graph with bigger dataset (30000 rows):



```
Accuracy score: 0.9511
Classification report:
              precision    recall  f1-score   support

        High       0.96      0.93      0.95       141
         Low       0.96      0.97      0.97       492
      Medium       0.93      0.93      0.93       328

    accuracy                           0.95       961
   macro avg       0.95      0.94      0.95       961
weighted avg       0.95      0.95      0.95       961
```

Table 9 Result of Random Forest Algorithm 2

I also classified my dataset using different classification methods and observed that the Random Forest algorithm provided the best performance(Dataset=30000 rows):



```
Random Forest
Accuracy score: 0.9511
Classification report:
              precision    recall  f1-score   support

        High       0.96      0.93      0.95       141
         Low       0.96      0.97      0.97       492
      Medium       0.93      0.93      0.93       328

    accuracy                           0.95       961
   macro avg       0.95      0.94      0.95       961
weighted avg       0.95      0.95      0.95       961
```

```
Logistic Regression
Accuracy score: 0.8210
Classification report:
              precision    recall  f1-score   support

        High       0.84      0.65      0.73       141
         Low       0.88      0.91      0.90       492
      Medium       0.73      0.75      0.74       328

    accuracy                           0.82       961
   macro avg       0.81      0.77      0.79       961
weighted avg       0.82      0.82      0.82       961
```

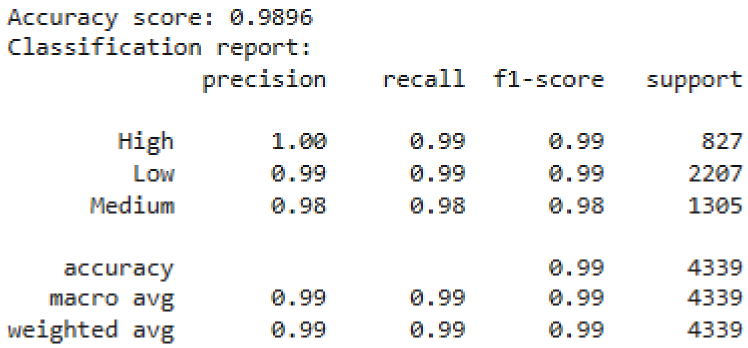
```
KNN
Accuracy score: 0.7804
Classification report:
              precision    recall  f1-score   support

        High       0.61      0.70      0.65       141
         Low       0.89      0.89      0.89       492
      Medium       0.69      0.66      0.67       328

    accuracy                           0.78       961
   macro avg       0.73      0.75      0.74       961
weighted avg       0.78      0.78      0.78       961
```

```
SVM
Accuracy score: 0.6868
Classification report:
              precision    recall  f1-score   support

        High       0.63      0.45      0.53       141
         Low       0.75      0.91      0.82       492
      Medium       0.56      0.45      0.50       328

    accuracy                           0.69       961
   macro avg       0.65      0.60      0.62       961
weighted avg       0.67      0.69      0.67       961
```

Table 10 Results of Classification Algorithms

As you can see above, it provided better results compared to other classification algorithms.

We are gonna take better results if we use whole dataset(390000 rows). Here is the result:



```
Accuracy score: 0.9896
Classification report:
              precision    recall  f1-score   support

        High       1.00      0.99      0.99       827
         Low       0.99      0.99      0.99      2207
      Medium       0.98      0.98      0.98      1305

    accuracy                           0.99      4339
   macro avg       0.99      0.99      0.99      4339
weighted avg       0.99      0.99      0.99      4339
```

Table 11 Result of Random Forest Algorithm 3

As you see, our classification accuracy is very high because we used the entire dataset. After analyzing the results of the classification, we can easily understand that using larger datasets makes our classification more reliable and accurate.

I also made a graph to visualize the final result:

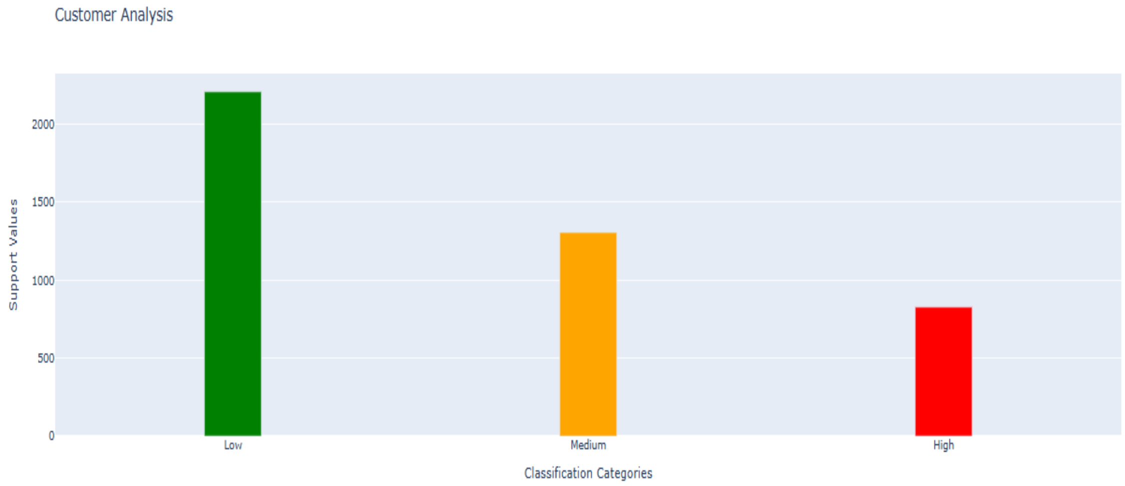


Figure 5 Graph of the final result

5. Discussion

5.1 Insights from Association Rule Mining

The results from ARM provide a detailed understanding of product relationships and customer purchasing patterns, serving as a foundation for strategic decision-making. High-lift product pairs, such as "CHARLIE & LOLA WASTEPAPER BIN FLORA" and "RED CHARLIE+LOLA PERSONAL DOORSIGN" in the United Kingdom, illustrate strong correlations that can be leveraged for targeted marketing campaigns. Similarly, in regions like Germany and France, moderately high-lift pairs like "3 HOOK HANGER MAGIC GARDEN" and "5 HOOK HANGER MAGIC TOADSTOOL" offer opportunities for regional bundling strategies. The lift and confidence values associated with these pairs reflect their consistent co-occurrence, making them ideal for cross-selling initiatives. By focusing on these findings, businesses can better tailor their offerings to meet specific customer demands.

The actionable potential of ARM lies not only in identifying strong product relationships but also in addressing weaker associations. For instance, in regions such as Australia and Norway, where the lift values for analyzed product pairs hover around 1, the relationships appear to be minimal. However, these findings highlight the opportunity for exploratory marketing and product placement strategies. Retailers can experiment with proximity-based placement or bundling promotions to stimulate interest in these combinations, creating potential for future associations that align with customer preferences.

Furthermore, insights from ARM can be utilized to optimize inventory management across different regions. High-support products in regions with strong relationships should be prioritized in stock allocation to prevent stockouts and improve supply chain efficiency. For example, ensuring the availability of "3 STRIPEY MICE FELTCRAFT" and "BLUE CHARLIE+LOLA PERSONAL DOORSIGN" in EIRE can help capitalize on their demonstrated co-purchasing behavior. Conversely, in regions with weaker associations, retailers can adopt a more conservative inventory approach to minimize overstock risks while exploring alternative promotional techniques.

To maximize the utility of ARM findings, businesses should integrate them into broader customer engagement strategies. This includes developing personalized recommendations based on frequently purchased items, offering discounts or promotional bundles for high-confidence pairs, and utilizing regional insights to align marketing campaigns with customer preferences. By doing so, businesses can not only enhance their operational efficiency but also foster customer loyalty, resulting in long-term revenue growth. These data-driven approaches underscore the transformative role of ARM in improving decision-making within the online retail industry (Rana & Enn, 2023).

5.2 Strategic Customer Segmentation Using K-Means Clustering

Customer segmentation is a cornerstone of modern marketing, enabling businesses to deliver tailored strategies to distinct customer groups. This study effectively employs the K-Means clustering algorithm to categorize customers of an online retail business based on their purchasing behaviors, using RFM metrics. By applying K-Means with two values of K (3 and 5), the study demonstrates how varying the number of clusters influences the granularity of insights. For example, K=3 provided broad categorizations such as low, moderate, and high-value customers, whereas K=5 revealed nuanced distinctions, including occasional buyers and declining customers. The thorough preprocessing of the dataset, including outlier removal using the Z-score method, laid a robust foundation for generating actionable insights.

The analysis highlights the strengths of K-Means in customer segmentation, particularly its ability to handle numerical data efficiently and its intuitive clustering outputs. The visualizations—such as scatter plots and 3D representations of the RFM metrics—play a vital role in understanding and interpreting the clusters. The insights derived from the clusters offer practical business implications. For instance, high-value customers identified in both configurations can be targeted with loyalty programs and personalized offers, while dormant customers present opportunities for re-engagement through campaigns or feedback mechanisms. The study emphasizes the strategic utility of K=5 clusters, which enable businesses to target niche groups like occasional buyers with specialized marketing tactics.

However, the study also acknowledges limitations inherent in the K-Means algorithm, such as sensitivity to initial centroids and the necessity of predefining the number of clusters. While K-Means provides valuable segmentation insights, the inclusion of alternative algorithms like DBSCAN or Hierarchical Clustering could enhance the analysis by addressing these constraints. Additionally, incorporating external variables such as demographic or behavioral data could enrich the segmentation further. The study underscores the need for ongoing evaluation of clustering outcomes to adapt to changing customer behaviors, ensuring that businesses stay ahead in a competitive marketplace. By coupling robust data preparation with strategic application of clustering, the work offers a roadmap for businesses aiming to maximize value from their customer segmentation efforts.

5.3 Random Forest in Customer Loyalty Classification: Insights and Future Directions

The findings of this study underscore the importance of leveraging machine learning techniques, particularly the Random Forest algorithm, to classify customers based on loyalty. The discussion below highlights the practical implications of these results, explores potential limitations, and suggests directions for future research.

The classification of customers into loyalty tiers (high, medium, low) provides actionable insights for businesses to optimize their marketing strategies and improve customer satisfaction. For instance, identifying high-loyalty customers enables targeted retention campaigns, while low-loyalty customers can be engaged through customized promotional offers to enhance their shopping frequency and spending. This model empowers decision-makers to allocate resources efficiently and prioritize customer relationship management.

A significant limitation observed during the analysis was the imbalance in the dataset, particularly for high-loyalty customers. The model's accuracy for this category improved as the dataset size increased, underscoring the need for comprehensive and balanced data. While alternative models such as Gradient Boosting Machines (GBMs) or deep learning-based approaches could also be considered, Random Forest offers a balance between accuracy and computational efficiency. GBMs may outperform in certain scenarios but often require more tuning and computational power. Therefore, the use of Random Forest aligns well with the objectives of this study.

Integrating more customer-specific features, such as feedback scores or online behavior, could enhance the model’s predictive capabilities. Investigating hybrid approaches that combine Random Forest with other algorithms might also yield further improvements.

In summary, this study highlights the effectiveness of Random Forest in customer loyalty classification and its potential for driving data-driven business strategies. While challenges such as data imbalance remain, these can be addressed through targeted future research, ensuring more accurate and reliable predictions. This discussion provides a foundation for exploring innovative solutions in customer segmentation and predictive analytics.

6. Conclusion and Future Work

6.1 Conclusion

This study has successfully demonstrated the significant potential of data mining techniques to revolutionize the online retail industry. By analyzing a one-year transactional dataset, key methodologies such as association rule mining, K-Means clustering, and Random Forest classification revealed actionable insights. Association rule mining identified meaningful product relationships that enable strategic initiatives like cross-selling, bundling, and inventory optimization. The comprehensive preprocessing phase, including data cleaning and normalization, ensured that all analyses were based on reliable and accurate data.

Through K-Means clustering, the study achieved granular customer segmentation by utilizing Recency, Frequency, and Monetary (RFM) metrics. This segmentation provided a clear understanding of customer groups, enabling businesses to implement tailored marketing strategies, such as personalized promotions for high-value customers and re-engagement campaigns for dormant users. The findings underscore how clustering can translate abstract purchasing patterns into practical marketing solutions, empowering businesses to address diverse customer needs.

Random Forest classification further strengthened the study's contributions by effectively categorizing customers into loyalty tiers based on their purchasing behaviors. The algorithm's high accuracy not only provided reliable classifications but also highlighted its potential for predicting customer lifetime value. Together, these insights emphasize the transformative role of data mining in improving decision-making, enhancing customer experiences, and driving sustained business growth. The study demonstrates that data-driven strategies are indispensable for maintaining competitiveness in the fast-evolving online retail landscape, paving the way for more informed, effective, and dynamic business practices.

6.2 Future Work

Building upon the findings of this study, future research should prioritize extending the dataset to include multiple years of transactional data. A longer temporal range would enable the identification of long-term trends, seasonality effects, and evolving customer behaviors, providing businesses with a more comprehensive basis for strategic planning. This extended dataset could also enhance the precision of predictive models by incorporating insights from repeated patterns and anomalies over time, offering more robust forecasting capabilities.

The integration of additional data sources presents another avenue for future work, with significant potential to enrich the analytical framework. By incorporating demographic information, customer feedback, and behavioral data such as website interactions or social media engagement, businesses can gain a more nuanced understanding of customer motivations and preferences. This multidimensional approach would not only refine customer segmentation but also support the development of hyper-personalized marketing strategies tailored to individual customer needs, improving both retention and acquisition efforts.

References

Aditya. (2023, März 22). Association Rule Mining Explained With Examples. *Coding Infinite*. https://codinginfinite.com/association-rule-mining-explained-with-examples/

Al-Maolegi, M., & Arkok, B. (2014). An Improved Apriori Algorithm For Association Rules. *International Journal on Natural Language Computing*, *3*(1), 21–29. https://doi.org/10.5121/ijnlc.2014.3103

Jadhav, A., Jadhav, A., & Jadhav, D. R. D. (2023). *Association Rule Mining in Retail: Exploring Market Basket Analysis with Apriori Algorithm* (SSRN Scholarly Paper 4461121). Social Science Research Network. https://doi.org/10.2139/ssrn.4461121

Jaiswal, S. (2024, Januar 4). *What is Normalization in Machine Learning? A Comprehensive Guide to Data Rescaling*. https://www.datacamp.com/tutorial/normalization-in-machine-learning?utm_source=chatgpt.com

Learn Statistics Easily. (2024, März 28). *Data Cleaning Techniques: A Comprehensive Guide*. LEARN STATISTICS EASILY. https://statisticseasily.com/data-cleaning-techniques/

Muralidhar, A., & Lakkanna, Y. (2024). *From Clicks to Conversions: Analysis of Traffic Sources in E-Commerce*. https://doi.org/10.48550/ARXIV.2403.16115

Qi, Z., Wang, H., Li, J., & Gao, H. (2018). Impacts of Dirty Data: And Experimental Evaluation. *ArXiv*. https://www.semanticscholar.org/paper/Impacts-of-Dirty-Data%3A-and-Experimental-Evaluation-Qi-Wang/ed145b646b09d65e856b3a589681afed8974e24e

Rajeshkumar, C., & Rajakumari, K. (2023). Predictive Analytics in E-Commerce Leveraging Data Mining for Customer Insights. *2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 783–787. https://doi.org/10.1109/ICECA58529.2023.10395492

Rana, M. E., & Enn, L. H. (2023, Juli). (PDF) Apriori Algorithm based Association Rule Mining to Enhance Small-Scale Retailer Sales. *ResearchGate*. https://doi.org/10.1109/BDAI59165.2023.10256952

Subramaniyaswamy, V., & Pandian, S. C. (2012). A Complete Survey of Duplicate Record Detection Using Data Mining Techniques. *Information Technology Journal*, *11*(8), 941–945. https://doi.org/10.3923/itj.2012.941.945

Temirov, A., & Dongxiao, R. (2018). *Data Mining Techniques in E-Commerce*. *8*(9).

Wang, Z., She, Q., Zhang, P., & Zhang, J. (2020). Correct Normalization Matters: Understanding the Effect of Normalization On Deep Neural Network Models For Click-Through Rate Prediction. *ArXiv*. https://www.semanticscholar.org/paper/Correct-Normalization-Matters%3A-Understanding-the-of-Wang-She/e11e9db36424ec47ed17f102cfbc43d78e01b19e