Project Report of

Market Basket Analysis

upGrad

Submitted by:

Name: Divyanshi Maurya

Reg. no.: 12219959

Section: K22UG

Roll no. : RK22UGA33

Submitted to:

Mr. Ved Prakash Chaubey (UpGrad, Instructor)

63892

Subject: CSE 353: EDA PROJECT

In partial fulfillment for the requirements of the award of the degree of

“Bachelors of Technology in Computer Science and Engineering”

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab.

SUPERVISIOR CERTIFICATE

This is to certify that **Divyanshi Maurya, 12219959** worked under my supervision on the project **Market Basket Analysis** from **August 2024 to November 2024,** During this period, they demonstrated excellent skills, dedication, and contributed significantly to the project's success. I highly recommend them for future opportunities.

Supervisor's Name: Mr. Ved Prakash Chaubey, 63892.

Sign:

Date:

ACKNOWLEDGEMENT

I would like to thank all those people involved in the successful completion of this project. I especially would like to thank my supervisor, **Mr. Ved Prakesh Chaubey, 63892, UpGrad Instructor,** for the precious guidance and persisting encouragement that helped me to move forward with the research process. Thanks to your expertise and encouragement, this work has taken shape.

I would like to extend my thanks to my colleagues and peers whose general collaborative spirit and constructive discussions illuminated my understanding of matters in particular. Special thanks to UpGrad modules for help with the data collection and analysis besides sharing their knowledge and resources.

This project would not have been completed without my family and friends who, at any juncture, wished for me and made believe in the potential ability that I possessed.

Then, I would like to thank all the participants and respondents for taking time to share their insights and information regarding this project. It is because your willingness to share experiences that made this research not only possible but meaningful for me.

I thank you all for forming part of this journey. The contributions have been substantive enough in enriching this work, and I look forward to a chance at some future collaboration.

- Divyanshi Maurya

12219959

TABLE OF CONTENTS

PROJECT TITLE

MARKET BASKET ANALYSIS

1. ABSTRACT

The objective of this project is to build up a recommendation system with the help of Apriori analysis applied in the Central Relations of this paper such that the shopping experience for customers enjoys their livelihood through recommending related products based on their purchases. The downloaded data set is from the UCI Machine Learning Repository and contains transactions in the online retail store, which would have captured data including product descriptions, quantities of products, and information on customers. The system's major objective is to apply association rule mining in the content-based recommender system to predict the frequent items purchased in order to recommend the product to the relevant customer. Removal of the outlier from the dataset or replacing outliers, changing High Dimensional Features, addition of derived variables and so on; all time-series data is indexed for relevant transactions. After which, the Apriori Algorithm was implemented in terms of generation of association rules based on Support, Confidence, and Lift metrics as well as FP-Growth algorithm. Some of the most relevant insights gained, including several rules with high confidence, include products often bought in pairs. Also, it has found useful relationships between products, especially types of products, applied in cross-selling and increasing customer activity. More research and goals could be raising the range of transactions gathered in the dataset. I had low resources on my computer so only used 10% of the dataset; that was possible to identify many more associations and apply hybrid or other forms of recommendation systems in order to improve their precision and applicability. This project presents an idea to develop a product recommendation system using Apriori analysis on an online retail dataset. The dataset is composed of transactions related to an e-shop present in the United Kingdom; it consists of each record including invoice number, product name, and quantity, unit price, and customer ID. The purpose of analyzing the purchasing behavior of the customers is generation of meaningful association rules that intend to show recommendations of products. The dataset was first preprocessed by removing outliers and unwanted records. By using the Apriori algorithm, frequent itemsets are generated with a minimum support of 0.01 percentage, meaning that similarity between customers of at least 1%. In these frequent itemsets, association rules are derived and can be viewed in terms of metrics such as confidence and lift, which determine the strength of the relationships between products. My system will enable users to input a product name and retrieve recommendations of related products that correspond to the rules generated. This project creates great inputs for cross-selling strategy development and bringing improvements in customer experience. Through preprocessing and mining on the dataset, I removed outliers and irrelevant or duplicate entries to avoid biases and anomalies. I ran the Apriori algorithm with a minimum support threshold of 0.01 and took a 10% sample of the dataset to generate frequent itemsets that identify

commonly purchased combinations of products. Then, the algorithm creates rules or relationships among products based on metrics such as confidence and lift. Based on that, the user can enter a name of a product, and the system will return recommendations for other products according to the generated rules. This project is very useful in cross-selling and can enrich the customer experience. It explores many aspects of customer behavior by using statistical numerical features through EDA, univariate, bivariate, and multivariate analyses. Besides, the FP-Growth algorithm is used for efficient mining of frequent itemsets to further enhance the recommendation process.

2. PROBLEM STATEMENT:

I am working on a recommendation system project that aims to make the platform user-friendly with personalized product recommendations. The system uses association rule mining techniques, including the Apriori and FP-growth algorithms, to find relationships between products from transactional data. My objective is to help the users discover relevant products by analyzing their behavior and suggesting often purchased combinations.

It will collect data regarding the products purchased and searched by customers, thereby determining which products are bought with other items. In terms of calculation, it uses NumPy libraries for numerical calculations and for even simple data visualization with Matplotlib, complex data visualization through Seaborn, and mlxtend for the Apriori and FP-growth algorithms. The system will provide a personalization experience in shopping by allowing customers to input any product name while automatically generating recommendations for relevant products.

By using efficient algorithms, I can reduce the time needed to calculate recommendations, even when dealing with large datasets, thus optimizing my goal. My hope is to make the user experience better for the users by assisting the users in easily finding products that suit their interests and preferences.

3. SOLUTION:

For building a strong recommendation system with association rule mining, I have taken two of the powerful algorithms, namely Apriori and FP-growth. These techniques learn the hidden patterns in the transaction data so that I can frequently purchase some product combinations and I can recommend it to others.

1. Data Collection and Preprocessing

Relevant transactional data has been used to build up the system. My dataset is transactions from an online retail store, with key features being InvoiceNo, StockCode, Description, Quantity, UnitPrice,

and CustomerID. The data captures types of customer actions such as purchasing or searching for a product. To clean the data, I used a preprocessing step, which included:

- Removal of duplicates and unwanted records: All the duplicate transactions or unwanted descriptions of products that had no value for the analysis were removed.
- Dealing with missing data: All the missing data in vital columns were dealt with either by filling with the right values or by removing them if the data was not sufficient.
- Outlier removal: Transactions with erroneous quantities or prices that one could obviously treat as outliers are removed not to figure into the research.

2. Data Analysis and Visualization:

During data cleaning, I continued with exploratory data analysis (EDA) of the purchasing behavior and trends. I visually described key insights using Matplotlib and Seaborn on the following:

- Product frequency distribution: which products were most often purchased
- Time-based transaction trends: what transactions are seasonable
- Product association: which some types of product buyers prefer to buy together on average

The above was helpful because it provided me with promising products to use later in the recommendation system.

3. Association Rule Mining:

The backbone of the recommendation system relies on association rule mining to generate frequent itemsets, which may be a product list frequently bought together. I implemented both algorithms: Apriori and FP-growth.

- **Apriori Algorithm**

  Apriori generates frequent itemsets by scanning the dataset iteratively, from single items up to larger itemsets. Three key metrics are used in the algorithm:

  Support: The percentage of the number of transactions which include the itemset.

  Confidence: The probability that a product will be purchased when another product is being purchased.

  Lift: This measures how many more times two products are likely to be purchased together than they would have if they were purchased separately.

The algorithm finds frequent product combinations with a minimum support threshold of 0.01-that is, itemsets that appear at least in 1% of the transactions.

- FP-Growth Algorithm

The FP-Growth algorithm is an efficient technique for mining frequent itemsets rather than the Apriori algorithm. It does not require multiple database scans and thus is faster, especially with large datasets. By constructing a Frequent Pattern Tree (FP-tree), it directly finds frequent itemsets without candidate generation, making it computationally efficient.

4. Generating Association Rules:

Following, I tried the Apriori and FP-Growth algorithms to generate association rules that might be associated with metrics such as confidence and lift. For instance, if a customer buys "product A," the system may suggest that "product B" is purchased if the rule has higher confidence; that is, if it means the customers who bought product A also buy product B frequently. The higher the lift, the stronger the relation between the products, and the more reliable the recommendation.

5. Interface for Recommendation System:

The final stage for the project was to create an interface which asks for a name of the product after which recommendations based on the association rules generated would be given back to the user. When the user provides a product name, the system queries the generated rules from the Apriori and FP-growth algorithm related to the items that are often sold together along with that particular product.

6. Optimisation and Further Work:

Although the deployment in the first phase has passed without a hitch, the system can be optimized in the following ways:

- Scaling with more data: Using larger datasets, I may yet find more complicated relationships. Currently, I am working with a 10% sample of the data due to resource limitations.
- Hybrid recommendation models: The marriage of association rules using collaborative filtering or content-based filtering can further improve the accuracy of recommendations and personalize it further.
- Real-time updates: Realtime transaction data will give the system the needed agility and responsiveness towards changing consumer preferences and emerging trends in products.

4. INTRODUCTION:

I am working on a recommendation system project which should help to enhance the experience of customers shopping by recommending personal products. This approach utilizes association rule mining techniques, especially the Apriori and FP-growth algorithms, specifically for transactional data; it identifies high-meaning relationships between products. The goal is to help users easily find the right products through past purchases or current interests to make the shopping journey seamless and enjoyable.This dataset is from the UCI Machine Learning Repository. The data pertains to transactional data reported by an online retail store operating in the UK. The data relates to many main features of invoices: including invoice numbers, product descriptions, quantities, prices, and customer IDs. I expect to get customer buying patterns, frequently purchased product combinations, and actionable recommendations from such analysis. The preprocessing includes outlier removal, removal of duplicate entries, and removal of irrelevant entries to ensure data quality and reliability. Then, I use the Apriori algorithm to generate frequent itemsets and association rules using metrics such as support, confidence, and lift. And finally, the FP-growth algorithm is used to handle bigger databases with a high frequency. This uses an FP-tree for identifying frequent itemsets without scanning it multiple times. Not only would it provide product suggestions to the users, but it would also provide business insight on overall popular product pairing and cross-selling opportunities. The way of approach will be with high-quality data and result in high relevance and accuracy of recommendations, which increases customer satisfaction and engagement. The system can be scaled up to accommodate and service more complex use cases and diverse datasets with potential future enhancements such as integrating real-time data and combining multiple recommendation approaches, thereby ensuring its effectiveness and applicability in different retail environments.

4.1 DATASET DESCRIPTION:

It is a transacational data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail.

Variable Name	Role	Type	Description	Units	Missing Value
InvoiceNo	ID	Categorical	A 6-digit integeral numba 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation		no
StockCode	ID	Categorical	a 5-digit integral number uniquely assigned to each distinct product		no
Description	Feature	Categorical	product name		no
Quantity	Feature	Integer	the quantities of each product (item) per transaction		no
InvoiceDate	Feature	Date	the day and time when each transaction was		no

			generated		
UnitPrice	Feature	Continuous	product price per unit	sterling	no
CustomerID	Feature	Categorical	a 5-digit integral number uniquely assigned to each customer		no
Country	Reature	Categorical	the name of the country where each customer resides		no

Table no. 1 : DataSet Attributes and Features

4.2 SIGNIFICANCE OF EDA

- By EDA we can understand the structure as well as the distributions of the variables present in the data. Knowing things that should first begin with later analyses.
- It helps in identifying outliers-anomalies in the data that skew results-cleaner data to analyze.
- EDA also aids in detecting patterns, trends, and correlations that can portray potential driver variables that would influence the outcomes.
- Visual relationships enable EDA to provide ideas about which variable might drive the effects within the specific context of the analysis. For example if a data analyst generalize a hypothesis, by EDA we can support the hypothesis.
- In the era of technology, there is vast of amount of data So, to clear and have quality data EDA helps assess the quality and completeness of the data, hence guiding the preprocessing steps needed before deeper analysis.

4.3 SIGNIFICANCE OF UNIVARIATE ANALYSIS

Some of the basic reasons to perform univariate analysis are to understand the basic nature of individual variables prevalent in any given dataset. Looking at it this way, because it works on a single variable, we can get a view about its distribution, measures of central tendency, and variability. Key statistics or central tendency measures such as mean, median, mode, range, variance, and standard deviation help summarize essential features of the data. This analysis is important to identify outliers or anomalies, which may skew results in subsequent analysis. Additionally, univariate analysis can be used in assessing data quality by recognizing missing values or inconsistencies. It basically precedes more complex analyses, allowing for preliminary insights and hypotheses concerning the data being analyzed.

4.4 SIGNIFICANCE OF BIVARIATE ANALYSIS

Relationship exploration between two variables plays a central role in bivariate analysis to unveil a correlation, an association, or some form of dependency. My main objective of such analysis is to check hypotheses regarding changes in one variable that may lead to change in another, thus highlighting

probable drivers of outcomes. Techniques of bivariate analysis, including correlation coefficients and scatter plots, enable expressing relationships in quantitative terms thereby demonstrating trends or patterns for predictive modeling and decision-making purposes. Besides, the data visualization techniques applied in bivariate analysis, such as using heatmaps and scatter plots, will help in communicating the findings effectively, thus helping stakeholders understand complex relationships better. All this insight put together will contribute significantly to the overall level of analysis and interpretation of the data.

4.5 SIGNIFICANCE OF MULTIVARIATE ANALYSIS:

- It explores complex relationships that may exist between multiple variables at the same time. Because of this, these types of relationships are likely to be missed by univariate analyses.
- It helps in identification of which variable or variables most influentially impact the dependent variable by seeing which ones have significant interactions.
- The ability to control confounding variables means isolating the effects of specific drivers more precisely.
- Multivariate techniques such as regression aid predictive modeling wherein one assesses how much each variable impacts the outcome.

4.6 SIGNIFICANCE OF APRIORI ANALYSIS:

I used the Apriori algorithm to identify frequent itemsets and generate association rules. It helped in finding hidden patterns from the transactional data, since its straightforwardness and interpretability rendered it a useful tool for analyzing buying behavior. I could recommend those products that had good affinity among them, helping in cross-selling opportunities and customer experience, with the help of metrics such as support, confidence, and lift.

4.7 SIGNIFICANCE OF FP-GROWTH ALGORITHM:

I used the FP-Growth algorithm because it can easily manage big data sets. In contrast to Apriori, it avoids scanning the databases repeatedly since it builds an FP-tree which helps in reducing complexity. The above led to quicker handling of data and easier identification of frequent itemsets, making it very effective for large-scale recommendation systems.

5. LITERATURE REVIEW

Personalized shopping experiences have been the hallmark of modern e-commerce, thus raising customer satisfaction levels. Among all the techniques used, association rule mining is especially famous for its ability to unearth patterns and links hidden in transactional data. This involves frequent itemsets and association rule generation as a means of recommending products that are frequently purchased together, which would consequently give a lot to cross-selling and targeted marketing strategies.

The algorithm known as Apriori, introduced by Agrawal and Srikant in 1994, is one of the most commonly used approaches to association rule mining. Its efficiency in generating frequent itemsets based on the principle of downward closure (if an itemset is frequent, all its subsets are frequent) has made this algorithm a foundational tool in market basket analysis. Aggarwal et al. (2013) point out the utility of Apriori to understand better customer behavior and actionable insight in inventory management as well as product recommendation. Although successful, the drawbacks in the Apriori algorithm lie in its high computational complexity as well as scans on the database when large data are considered.

To overcome these disadvantages, the FP-Growth algorithm is developed as a more efficient algorithm. FP-Growth does this by building a compact frequent pattern tree, which can summarize the data in a way that will reduce the number of scans needed, and thus produce frequent itemsets faster. In comparison studies conducted between FP-Growth and Apriori, as Han et al. (2000) show, the former outperforms the latter significantly with regard to speed efficiency and memory usage, particularly in large-scale environments. It therefore finds applications in any environment where datasets are massive and in real-time analysis.

Other than retail, association rule mining has been practiced in nearly all fields. For instance, in the health sector, it is used in prescribing drugs according to patient histories, as in Patil et al. (2018), in which Apriori discovered some relationships between symptoms and medicines. Similarly, in the finance sector, association rules help post portfolios in terms of client preferences. And in the entertainment industry, both the movies and series of Netflix and the songs of Spotify are recommended according to patterns that were practiced by the users.

Alternative approaches to recommendation include content-based filtering and collaborative filtering, alongside association rule mining. Content-based filtering uses the features of products to recommend similar products. Collaborative filtering uses the tastes of other users with preferences similar to the target audience's tastes. Hybrid systems, which combine content-based filtering and collaborative filtering

approaches with association rule mining, have been proposed to improve recommendation accuracy and overcome individual limitations.

Many Kaggle projects support these applications of Apriori and FP-Growth algorithms. For instance, the Market Basket Analysis project demonstrated how association rule mining can be used in order to identify the most frequently purchased itemsets in retail data. The Online Retail Dataset Analysis also justified the use of FP-Growth based on a UK online store transactional file, thus pointing out how efficiently the algorithm helps in producing actionable insights.

Despite this, the mentioned methods have some drawbacks: Apriori got worse as the data size is increased. The efficiency of FP-Growth algorithm relies on the structure of FP-tree; however, rapid development in hardware and development of hybrid algorithms could reduce such obstacles, making these algorithms applicable on a wider scale.

This was achieved by combining literature insights and practical applications, whereby the resultant design for a hybrid Apriori and FP-Growth-based recommendation system could be effectively implemented to support friendly user development in terms of generating personalized product recommendations.

6. LIBRARIES USED:

The libraries used in this project played a significant role in implementing the recommendation system and effectively analyzing the data.

- **NumPy** was a fundamental library used to create arrays and perform numerical operations, thus allowing for efficient handling of large-sized datasets and needed array-based computations required for preprocessing and data transformations. This library ensured smooth integration with other libraries used in the project because of its capability to handle multi-dimensional arrays and matrices.

- **Pandas** was an absolute necessity for data manipulation and analysis. It offered tools to clean, transform, and explore the dataset effectively: how to handle missing values, remove duplicates, and create derived variables. The DataFrame and Series structures offered by the library make data handling intuitive for association rule mining, while structuring online retail data.

- **Matplotlib and Seaborn** were used to visually understand the data and uncover trends and patterns. Though Matplotlib supported a wide variety of plots, Seaborn was helpful because it allowed us to create more fashionable, detailed statistical visualizations like heatmaps and bar charts and paired

plots, which played a crucial role during the EDA phase while identifying key insights into customer behavior.

- A library named **Mlxtend** extended the functionality of machine learning and data science workflows. It implemented the Apriori and FP-growth algorithms. Its module for association rule mining made generating frequent itemsets simple and deriving meaningful association rules based on support, confidence, and lift metrics of easy computation. Thereby making it possible to deploy efficient as well as scalable recommendation algorithms.

- Finally, the support in preprocessing steps and performance evaluation were both offered by **Scikit-learn**. These crucial tools include splitting datasets and standardizing values so that the whole data is analyzed in a uniform manner. All these libraries together combined to form an all-inclusive toolkit, supporting each phase of the project-from data preprocessing to generating insightful recommendations.

7. METHODOLOGY

7.1 DATA COLLECTION

The dataset for this project was sourced from the UCI Machine Learning Repository, which is recognized to source quality datasets for research and analysis. This dataset is highly relevant as it originates from real-world data extracted from online retail transactions, making it ideal for developing a recommendation system.

Some of the attributes linked with the dataset are:

- InvoiceNo: A number used to uniquely identify a transaction.
- StockCode: A unique stock code for each product.
- Product Name: Product name.
- Quantity: Number of products per transaction.
- Invoice Date: Date and time for a single transaction.
- Unit Price: Each product unit price.
- Customer ID: Unique ID for each customer.
- Country: Country to which each customer belongs.

I have chosen this dataset because it gives a nice, rich snapshot of customer purchasing behavior-thus very important to look into patterns and the right kinds of product recommendations. Based on its

structure, this is fitted perfectly to apply the Apriori algorithm and association rule mining for useful findings.

Libraries I have used:

- google drive: to import google drive to my jupyter notebook
- pandas: to convert the given excel file into dataframes for easier manipulation of data.

7.2 DATA EXPLORATORATION

print(data.head())

Telling us about 5 rows in the dataset.

	InvoiceNo	StockCode	Description	Quantity	/
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	
1	536365	71053	WHITE METAL LANTERN	6	
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	

	InvoiceDate	UnitPrice	CustomerID	Country
0	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

print(data.tail())

Last 5 rows of the dataset

	InvoiceNo	StockCode	Description	Quantity	\
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	

	InvoiceDate	UnitPrice	CustomerID	Country
541904	2011-12-09 12:50:00	0.85	12680.0	France
541905	2011-12-09 12:50:00	2.10	12680.0	France
541906	2011-12-09 12:50:00	4.15	12680.0	France
541907	2011-12-09 12:50:00	4.15	12680.0	France
541908	2011-12-09 12:50:00	4.95	12680.0	France

data.info():

The function data.info() will display, a summary of the whole dataset, the number of entries, column names, non-null entries and data type for each column, so it's helpful to really get the structure of the dataset, pick out missing values, and know what type of data the variables are using.

```
<class 'pandas.core.frame.DataFrame'>

RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 33.1+ MB
None
```

data.describe()

The data.describe() function will generate descriptive statistics of the dataset and includes measures such as count, mean, standard deviation, min, and max values for the numerical columns. It has a summary of the distribution of the data that can be used in finding any possible outliers or anomalies. This will help understand the central tendency, spread, and general shape of the numerical features.

	Quantity	InvoiceDate	UnitPrice	CustomerID
count	541909.000000	541909	541909.000000	406829.000000
mean	9.552250	2011-07-04 13:34:57.156386048	4.611114	15287.690570
min	-80995.000000	2010-12-01 08:26:00	-11062.060000	12346.000000
25%	1.000000	2011-03-28 11:34:00	1.250000	13953.000000
50%	3.000000	2011-07-19 17:17:00	2.080000	15152.000000
75%	10.000000	2011-10-19 11:27:00	4.130000	16791.000000
max	80995.000000	2011-12-09 12:50:00	38970.000000	18287.000000
std	218.081158	NaN	96.759853	1713.600303

Table no. 2: Dataset Description

7.3 DATA PREPROCESSING

During this project, I took these following major steps towards data cleaning to prepare the dataset for analysis:

7.3.1 MISSING VALUE PREPROCESSING

There existed many missing values in the dataset, mainly in the `CustomerID` and `Description` columns. I addressed this by:

The total numbers were:

Total Numbers:

```
Description     1454
CustomerID    135080
dtype: int64
```

Percentage Total:

```
Description     0.268311
CustomerID     24.926694
dtype: float64
```

By Percentage I am dropping rows with missing CustomerID, which is critical to distinguishing between different behavior by customer and retaining the Description dataset as it is used while identifying products and making suggestions.

7.3.2 DATA TYPE CONVERSION

This is the process of changing data types of columns of a dataset so that there is homogeneity and efficiency in terms of memory usage or in computer's resource use. Numerical values stored as strings can be converted into integers or floats, and also date strings into `datetime` objects for proper data analysis. This is the only way to ensure that the data is ready for arithmetic operations or comparison operations.

As for my project, I have converted my customerID column from float to string.

```
data['CustomerID'] = data['CustomerID'].astype(str)
```

7.3.3 REMOVING DUPLICATES

For eliminating the repetition in the dataset, I made use of the drop_duplicates() method from the pandas library. This function also provides control on whether to check duplicates on all columns or a few columns, and hence only the truly duplicate entries are removed. After doing this, I checked the cleanliness of the dataset where I looked at the number of entries before and after the removal and comparing these two numbers. This activity not only helps in cleaning a dataset but also improves the

quality of the analysis as it ensures that there is no repeated information in each entry which further helps in improving the credibility of the results.

Number of duplicate rows: 5268

Duplicates dropped.

New total number of rows: 535187

7.3.4 REMOVING CANCELLATIONS IN INVOICE

I ensured that cancellations are removed by filtering out the rows whose `InvoiceNo` starts with the letter ‘C’ because such transactions are those that were cancelled. This is important in this scenario because one is able to concentrate on the genuine sales transactions only, hence providing a better foundation for analysis and insights.

```
# Filtering out rows where InvoiceNo starts with 'C'
data = data[~data['InvoiceNo'].str.startswith('C')]
```

By this, we are removing the unwanted data.

7.3.5 OUTLIER ANALYSIS:

As outliers creates bias in the dataset, they needed to be removed.

There are two methods I have used to remove the outliers:

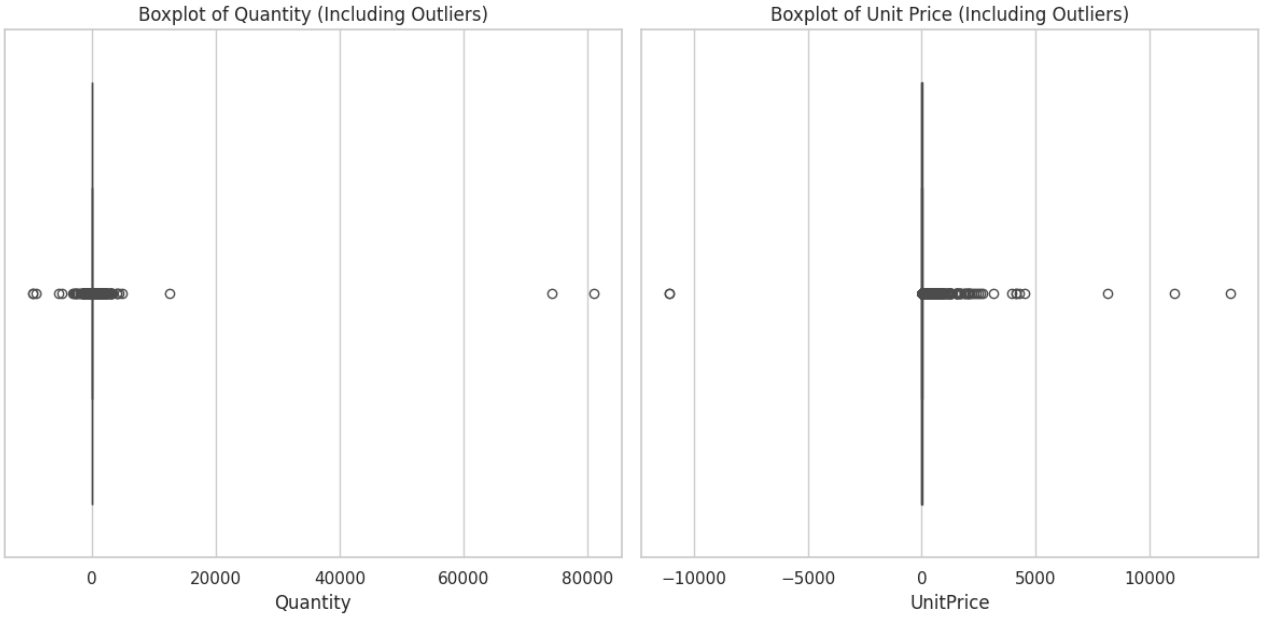
1. Z-score method
2. IQR Method

```
Number of outliers detected using Z-score:
Quantity     525
UnitPrice    680
dtype: int64
```

Box plot bounds:

Quantity Bounds: -14.0 26.0

Unit Price Bounds: -3.0700000000000003 8.45



After Removing the outliers:

Original data shape: (525936, 8)

Data shape after removing outliers: (460801, 8)

7.3.6 FEATURE ENGINEERING:

I have created a new feature named: Total price

And divided the time into year, month and day

```
data_no_outliers['InvoiceDate_day'] = data_no_outliers['InvoiceDate'].dt.day
data_no_outliers['InvoiceDate_month'] = data_no_outliers['InvoiceDate'].dt.month
data_no_outliers['InvoiceDate_year'] = data_no_outliers['InvoiceDate'].dt.year
```

index	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	TotalPrice
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	15.299999999999
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	20.34
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	22.0
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	20.34
4	536365	84029E	RED	6	2010-12-01	3.39	17850.0	United	20.34

index	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	TotalPrice
			WOOLLY HOTTIE WHITE HEART.		08:26:00			Kingdom	
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850.0	United Kingdom	15.3

Table no. New Features added

7.3.7 DATA VISUALIZATION:

Data visualization is an essential part of data analytics, a powerful tool for interpreting complex data and effectively communicating insights. By turning raw data into graphical representations, such as charts, graphs, and interactive dashboards, data visualization enhances our ability to identify patterns, trends, and outliers that might be obscured in traditional number formats It enables researchers and decision makers can understand relationships between variables and understand the story behind data. Additionally, effective data visualization facilitates good decision-making by providing a clear, intuitive, and engaging visual narrative that can be presented to a variety of audiences from technical teams to stakeholders. As the amount and complexity of data grows, data visualization becomes increasingly important in deriving meaningful insights and driving actions In today's data-driven world, it's not like improved data visualization techniques not only support analysis but improve communication, leading to a deeper understanding of data and its meaning.

7.3.7.1 UNIVARIANT ANALYSIS:

Here, the univariate analysis approach was employed to analyze the individual features of this dataset, giving indispensable insights into transactions made by customers. These included primary variables like product quantity, unit price, and customer country, for instance; an analysis of which can point out trends and outliers. For example, from the distribution of product quantities, one can see the items most frequently purchased; unit price analysis will emphasize pricing patterns and points of anomaly. Customer country analysis provided insight into geographic purchasing behavior; this was done to ensure a thorough understanding of the data set and would also assist in preprocessing and association rule mining, since patterns important for generating accurate and meaningful recommendations would be identified.

Figure 1: Distribution of total Price

By the graphs , it shows the distribution of the Total Price in the dataset. The x-axis is the Total Price and the y-axis is the Frequency i.e., how many transactions exist for a given price range. It uses a histogram, bars, and a kernel density estimate, a smooth curve, to describe the distribution.

- Skewness :

  The data is skewed to the right, which suggests most transactions are at lower price levels, and higher prices are less frequent. This is typical in retail, which is here, It is because low-priced goods often have higher volume sales.

- Peak (Mode):

  The highest frequency (peak of the bars) is at the lower end of the total price, roughly around 1 to 5 units, indicating that most transactions are inexpensive.

- Multi-modal Distribution:

  More than one small peaks on the curve indicates the presence of Total Price Clusters, likely because of the pricing groups or popular product bundles.

- Long Tail:

  The right hand side of the curve (prices above 20) indicates a non-zero value at that price point, although its frequency continues to decline, thus indicating some transactions at high prices. These might be wholesale purchases or high-value items.

- Density Curve:

  The density curve is a smoother version of a histogram. It will remove any sort of fluctuation and show the general trend in the distribution of prices.

This analysis suggests that most customers purchase lower-priced items, in keeping with the usual patterns of an e-commerce site. The presence of some high-value transactions may be useful for finding

premium customers or exclusive product categories. This distribution is crucial for designing recommendation systems as it indicates the need to focus on frequently bought lower-priced items while considering strategies to target high-value transactions.

Figure 2: Word Cloud

Word clouds represent the frequency of words in a text; the more frequent the word, the bigger it would be in the picture. This word cloud uses color and sizes for fonts to get out the words. Some of the words for which "Bag" is the most common are "Jumbo Bag," "Lunch Bag," "Retrospot," "Light," "Metal," and "Christmas." This may imply it's an image of product listings or advertisement material for a bag specifically related to bigger bags perhaps designed in a retro or vintage theme, and perhaps with a Christmas theme. There are numerous adjective words like Vintage, Pink, Red, and Dolly.

Words like "cake," "bottle," "holder," "clock," "tin," "cases, and "drawer knob" may imply that the kind of product it is might be varied.

In other words, the word cloud is probably a set of words probably taken from product descriptions or tags associated with a selection of items probably in some online shop catalog, likely focusing on a certain type of bag with a retro or vintage style.

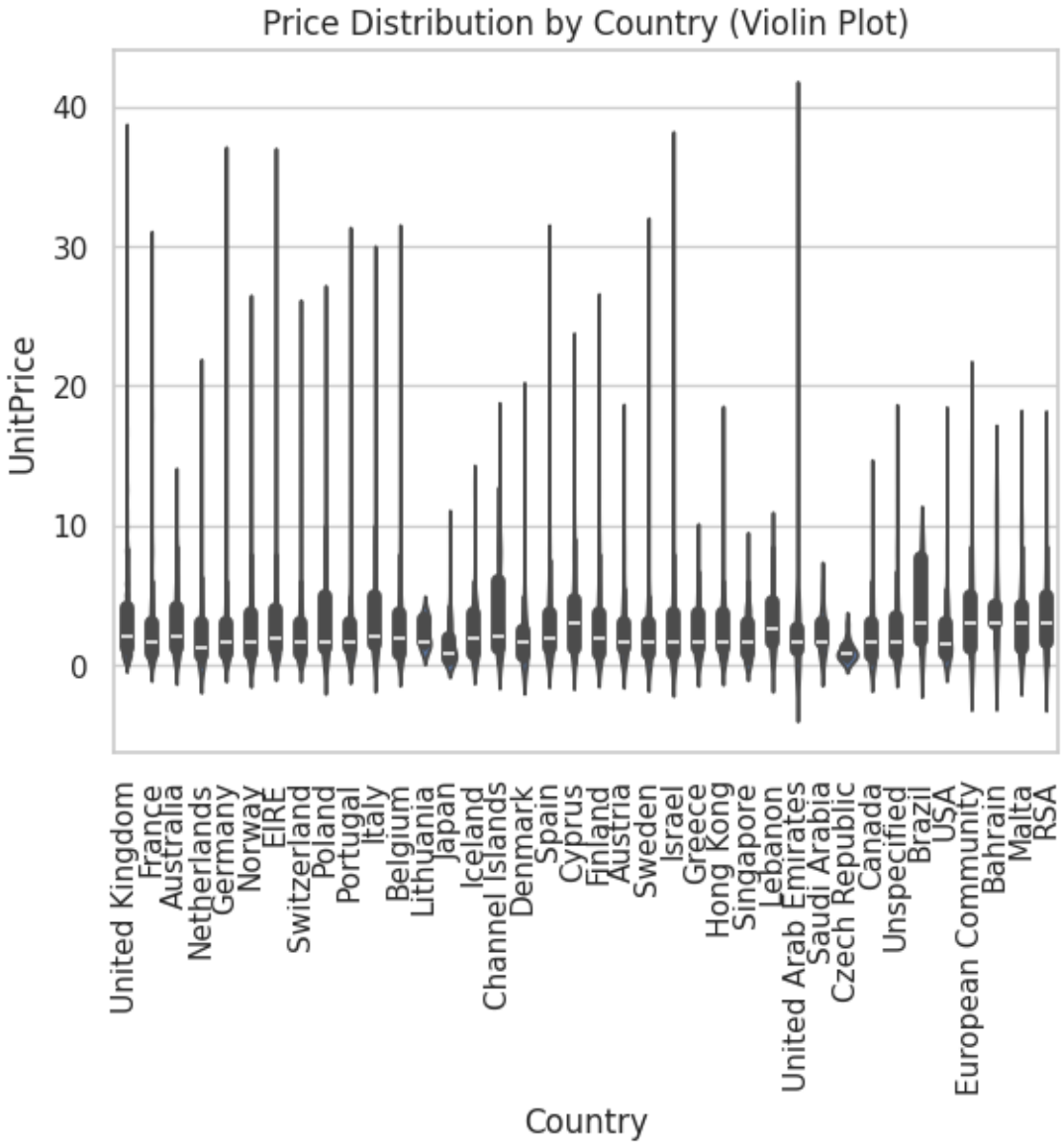


Figure 3: Violin Plot price distribution across countries

The violin plot is a display that characterizes the distribution of unit prices across different countries while giving an overview of variability and central tendencies for each country. Each country's unit price distribution is represented as a "violin," the combination of a box plot and a kernel density estimate. Wider areas of a violin suggest where the data points are concentrated, meaning the range of prices that prevails for each country. This actually gives a finer view of price distribution than a basic box plot, as the latter can only provide for mean, quartiles, and outliers.

On this plot, the X-axis lists various countries versus the horizontal axis. The Y-axis range is unit prices - nearly zero to 40 units. Within the violin, each piece affords insight into the range and variability of unit prices for that country. The middle central white dot in each violin represents the median unit price. The

length of the black bar, called interquartile range, or IQR, indicates how spread out the middle 50% of the data is. Whiskers extending from the IQR indicate the range of the data, excluding any outliers.

Just by looking at the plot, it should be easy to see that the countries vary in the changes in the unit prices. Narrowly clustered prices have some countries, as indicated by their narrow violins, meaning less variability in prices. Their prices are more spread out, while in some countries, the violins are broader. Whiskers extended upwards in certain countries reveal that they reached more remarkable maximum prices than others. In addition, a few violins have asymmetrical shapes, indicating that the price distribution in those countries is maybe skewed, with more lower-priced or higher-priced products.

The violin plot helps to effectively compare distributions between multiple countries. It can reveal patterns in the variability of price, central tendencies, and outliers, which otherwise provide less useful information on how unit prices differ across countries. Such analysis can prove useful in understanding regional pricing differences and making informed recommendations about products based on these regions.

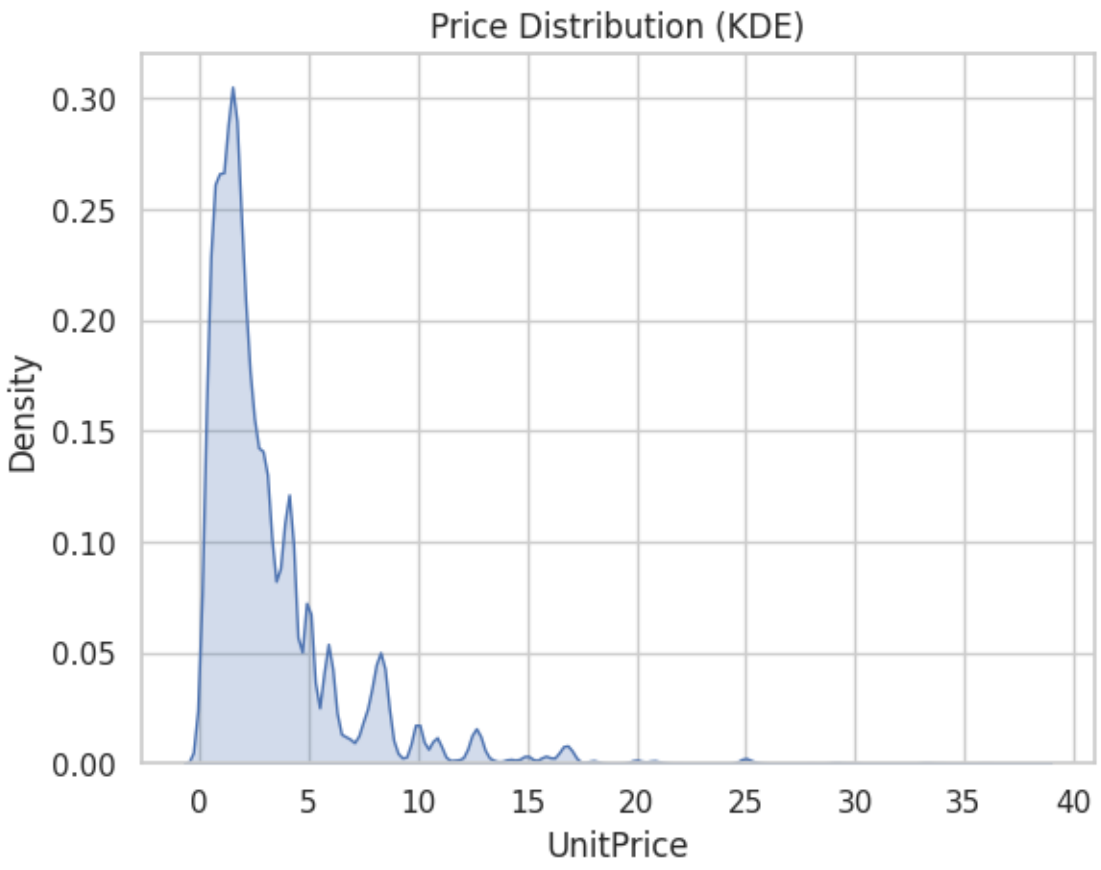


Figure 4: Price Distribution

The KDE plots a smooth curve defining the distribution of unit prices, indicating how prices are spread among different products. For example, in this graph, the x-axis refers to different unit prices from around 0 to approximately 40, while the y-axis shows how dense or concentrated products are at those levels of price. The plot is actually designed to offer more of an understanding of the underlying distribution, particularly when compared with a simple histogram or bar chart.

The plot clearly shows a strongly right-skewed distribution, with the majority of products priced on the low end of the spectrum, around a very prominent peak in the range from 2-3 units. This peak shows that most items are bound within this range, therefore with high concentration of rather inexpensive items. Increasing unit price depresses the products' density gradually forming a long tail that goes up into high price. This is a tail of less important products and at higher prices; fewer items were sold at higher price points.

The plot further illustrates different minor peaks and troughs that spread out across the distribution. These might be associated with product clustering at specific price points, thus possibly indicating that it refers to distinct groups of products or a specific price range in the market. The region under the curve would represent the probability density graphically. Balanced by the shape, in general, the KDE plot reflects a scenario in which most of the products are low-priced and trail off as the price increases. This can provide useful insights on the pricing trends and customer preferences in your dataset.

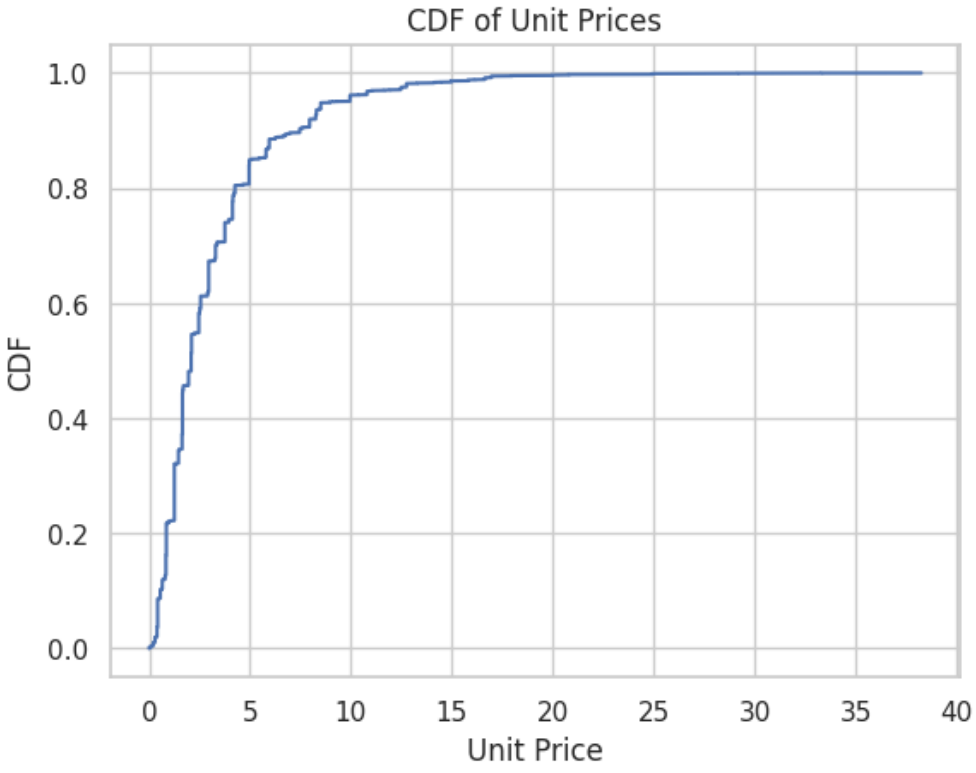


Figure 5: Cumulative distribution of Price

This plot will present how unit prices are distributed in the dataset based on CDF, showing the cumulative probability for prices to be below or equal to a given number of value. The x-variable represents the unit price, from approximately 0 to 38, while the y-variable is the cumulative probability, which is the proportion of unit prices that fall below or equal to a given value on the x-axis. If the y-value equals 1.0, then 100% of the unit prices in the dataset are less than or equal to the associated x-value.

The plot represents a stair-step curve in relation to the nature of the data being cumulative. There is a steeply rising curve around the unit price of 2, indicating that a large percentage of the items in the dataset have prices in this range. This shows most of the products are rather cheap. As the unit price rises, the curve tappers down on reduction in numbers of products with increases in price. That is the flattening shows that high-priced items are not numerous in the given dataset. The curve peaks at a value of 1.0 at the highest price so that all prices have been covered and there are no prices above the maximum unit price.

The CDF plot successfully demonstrates the way unit prices are distributed in the dataset, having such visual cues as a clear presence of price concentration in lower price ranges and a dimming-off number of products at higher price points. It is useful in general for understanding the overall pricing structure, where most items are priced, and how the distribution of prices varies across the set.

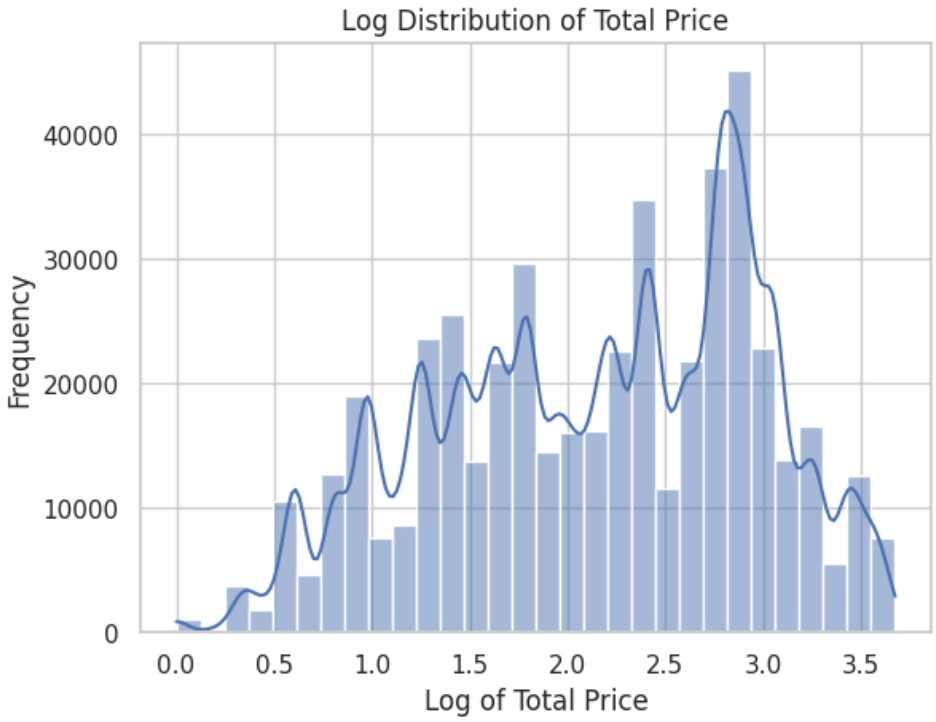


Figure 6: Log Distribution of Total Price

The histogram of the logtotal price lets us understand how total prices would be distributed after a logarithmic transformation. A histogram alone provides bars representing the frequency of observations within certain ranges, but in combination with an added kernel density estimate curve, it can serve as a smoothed realization of the underlying probability density function.

In the x-axis, the total price is represented with logarithmic transformation, possibly base 10 or natural logarithm. Logarithmic transformations are often used to reduce data skewness so it is more normally distributed and thus easier in the analysis. Such transformation compresses the scale of large values and may bring out underlying trends in data otherwise obscured by extreme values. The y-axis is the frequency of counts in each bin corresponding to various ranges of log-transformed prices.

The histogram exhibits a multimodal distribution, meaning that the data can't be represented with a single peak, and the dataset includes several distinct ranges or groups according to total price. The log transformation notwithstanding, the distribution looks slightly skewed on this scale, though the transformation typically helps to mitigate such skewness. The sharp peak at about a log value of 2.5-3 suggests that there actually is a notably prevalent price range in this portion of the data, marking a concentration of products in that range.

The KDE curve, a smooth blue line placed above the histogram, helps visualize the general shape of the distribution. It then gives a clearer view into the probability density, and by it all smoothing out the variability in the histogram bar, it is easier to understand the general trend of the data. The log transformation of the total price data, according to the visual representation, is not evenly distributed because it has visible marked distinct clusters of prices with some degree of skewness left.

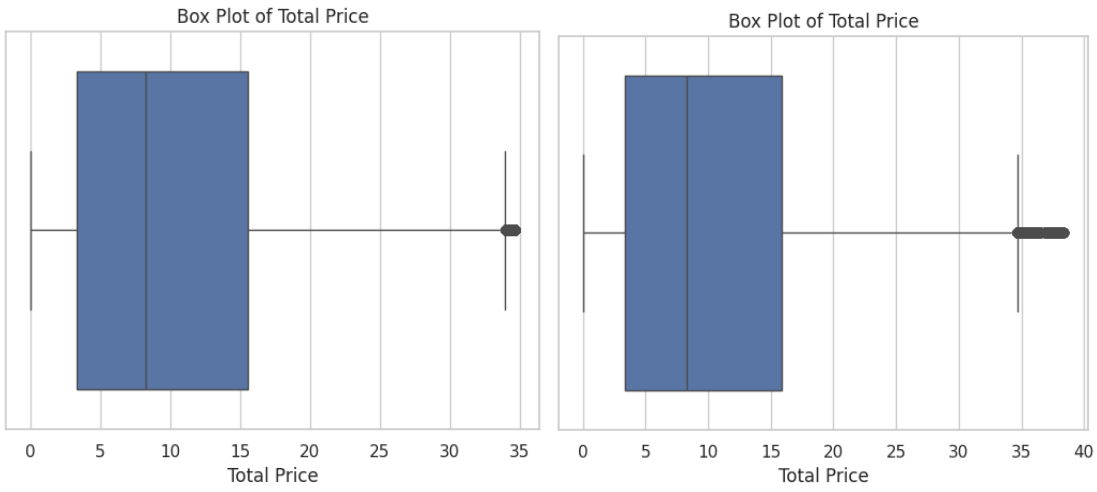


Figure 7: Box Plot of Total Price

The box plot for the sum of price values clearly visualizes how the distribution and variability of the sum of price values are in the given set. The plot can be divided into crucial parts: Q1, which is 25th percentile, followed by the median or 50th percentile, and then Q3, which is the 75th percentile; whiskers are going out from the box representing the range of data.

In the following boxplot, Q1 (25th percentile) is 3 that means 25% of all price values are smaller than 3. The median or middle value in this case is 8. This means: Half of all price values are less than 8 and half of them are greater than 8. A 75th percentile, Q3=16: it represents that 75 % of all values of total price are less than 16. The interquartile range, IQR equals Q3-Q1=16-3=13, and illustrates the spread of the middle 50% of data.

The whiskers of the boxplot expand the limits of the interquartile range to indicate the range of the values, excluding outliers. In the first boxplot, the right whiskers are longer, and there are several data points visible outside of the whisker. It suggests that, in this case, higher total prices in general have a higher range, with some extreme values that lie well above the usual price range. Outliers show that there do exist few very high total prices in the dataset-the infrequent occurrences could be representing a few very rare or high price items.

The whiskers of the second boxplot are shorter and the number of outliers on the right-hand side much smaller, which suggests that the total prices in that data set are more tightly clustered with fewer extreme values. This would indicate that most of the items have a more consistent total price, and less frequently unusually high prices appear.

From a general overview, the two boxplots reveal important information about the total prices. The first plot appears to have more spread and outliers, while the second plot presents relatively concentrated distribution with fewer extreme values. This could be done in order to conclude on which set of products had extreme differences in terms of price.

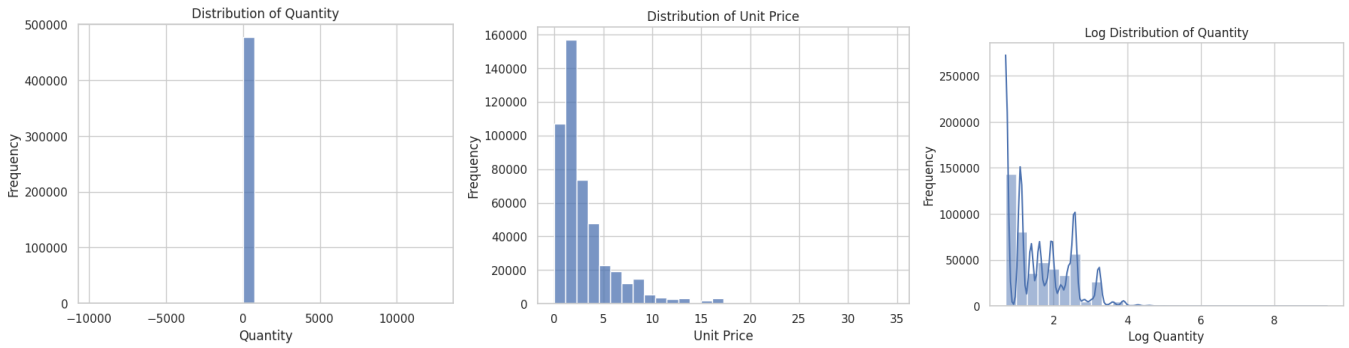


Figure 8: Distribution of numerical Values

The distribution of quantity, unit price, and log-transformed quantity in the dataset is depicted with some unique features relating to each variable's data pattern.

The distribution of quantity is represented by a single bar, the y-axis denotes the Quantity and X-axis denotes the frequency of the the dataset which is near 50,000. And the quantity is a single bar which suggests that most products have a fixed or very consistent quantity associated with the transactions. This is a reflection of very little variation in the data, which could indicate uniformity in the data of quantity because it could be standard packaging or predefined purchase quantities in the dataset.

On the other hand, the unit price is skewed to the right. This results in most of the unit prices being concentrated at the lower end of the spectrum, but few products at higher price ranges. Skewness arises in data when most items are affordably priced, while fewer premium or more expensive items contribute to a long tail in the distribution.

As can be seen above, the logged quantity, too, is skewed; nonetheless, the log-compression of large values makes the distribution appear to be more normalized than the original data. Log-transformation is also useful in order to reduce the impacts of extreme values and provide greater clarity over the trend of the data. However, skewness in log-transformed quantity means that original data contains notable variation in quantities even if the extreme values are less conspicuous after transformation.

Summarily, the distribution of quantities is very uniform, the unit price is right-skewed and at lower concentration value, and despite the normalization by log transformation, log-transformation of quantity will retain some degree of skewness by nature, as original data does contain variability. These patterns can provide necessary insights into the structure and nature of the dataset and inform further analysis and decision-making.

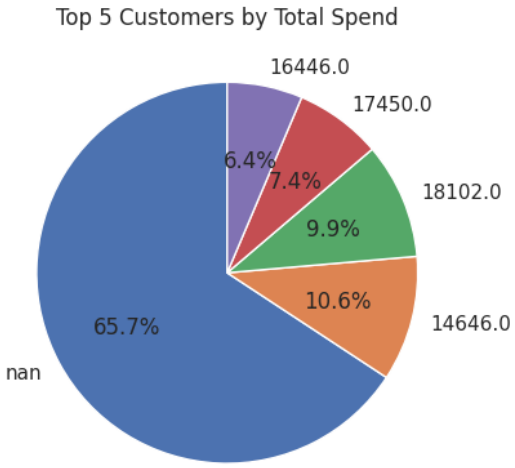


Figure 9: Pie Distribution of top 5 customers by total spend

A pie chart illustrating the break down of the total spend from the top five customers can be very beneficial in showing the concentration of the purchasing in the dataset. The first significant piece of information is that "nan" dominates the category with 65.7% of the total spend, and this most probably represents missing or unassigned customer data. This high percentage reveals that most of the dataset lacks clear identification of customers, which might adversely affect the analytical capability and segmentation of customers.

The largest number of identified customers belongs to customer 14646, which amounts to a share of 10.4% of total spend, thus establishing the status of a key buyer with values far over the rest of customers. Customers 18102, 17450, and 16446 occupy 9.9%, 7.4%, and 6.4%, respectively. These four big spenders-although they are significant as an individual, together-they amount to a vastly smaller proportion than the "nan" column, which actually illustrates the predominance of a few big contributors in the data set.

In terms of analysis, this distribution seems to indicate that most of the revenue is generated by a few major spenders based on the Pareto principle, in which it is often evident that a smaller customer base accounts for the majority of sales. However, this reality has only uncovered that more efforts should be dedicated toward seeking better collection and management practices for customer data. This imbalance may uncover additional concealed valuable customers and enhance segmentation efforts.

Dependency on just a few key customers was also suggestive of the risk of concentration of revenues; in other words, if one or more of the above top customers reduced their spending or leave, it will somewhat affect overall sales. Diversifying the customer base and finding other potential high-value customers could possibly mitigate this risk.

In the nutshell, the pie chart indicates that urgent attention is required toward missing customer data while using the skewness of expenditure as a way of pointing towards the reason for maintaining relational relationships with key buyers and spreading this customer base thereby minimizing revenue dependency on one small group.

Figure 10: Q-Q plot of Unit Price

A Q-Q plot gives a graphical representation of the distribution of unit prices since it compares the actual data to a theoretical distribution. The vertical axis shows the ordered values of the unit prices while the horizontal axis represents the theoretical quantiles that were assumed for the model. Data points are plotted against a reference line that corresponds to the perfect alignment of real and theoretical distributions. The numeric range varies between about -10 and 40. It represents the spread on the scale of unit prices in the dataset. Points that closely follow the reference line indicate that the observed data fits the theoretical distribution well there. Departures from the line indicate areas where the observed data deviates in some places where skewness, outliers, or other distributional differences could occur. This plot is specifically very useful for assessing normality or some other assumed distributions of unit prices. Significant deviations, especially those in the tails, could suggest extreme values or a non-normal nature to the data.

7.3.7.2 BIVARIANT ANALYSIS:

Bivariate analysis in this project explores relationships between two variables, providing insights into patterns and associations within the dataset. For example, analyzing Quantity and UnitPrice may reveal trends, such as higher quantities being associated with lower unit prices, indicating bulk discounts. Similarly, examining TotalPrice against Country can highlight regional spending behaviors, identifying key markets. Another important relationship would be between InvoiceDate and TotalPrice, which may potentially reveal seasonal or temporal purchasing trends. Such an analysis not only identifies dependencies between variables but also helps in predictive modeling, customer segmentation, and strategic decision making through actionable insights emanating from the data.

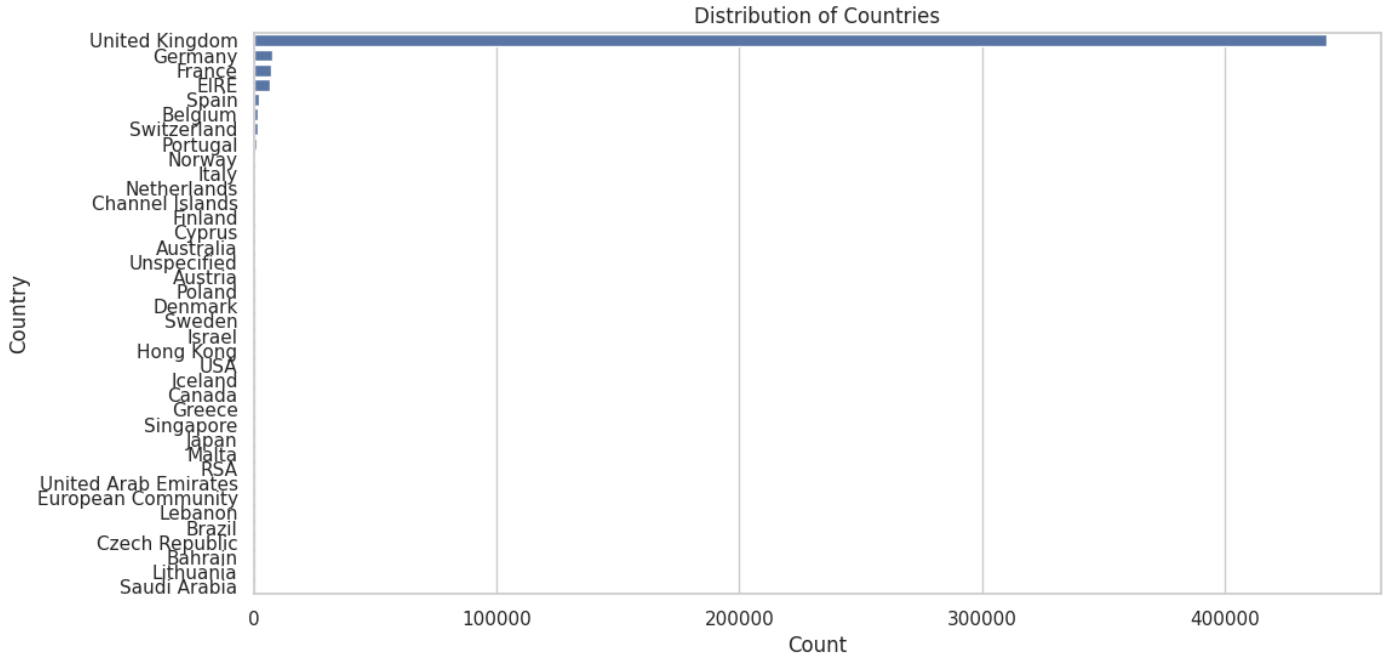


Figure 11: Country v/s Frequency

The UK bar is significantly taller than the others, which presents dominance as a leading market. From here, it shows that most of the transactions in the dataset are held through this country, mainly because this is where the business has harnessed its operations or due to having a larger customer base.

The other countries' bars are demonstrably shorter and follow a downward trend, indicative of a long tail. Transactions are international in scope, though far less frequent than in the UK. The visualization underlines how important the UK is for revenue and provides an opportunity to explore and expand into markets where activity is lower. It can help tailor strategies to maintain strong performance in the UK while targeting growth within the represented areas.

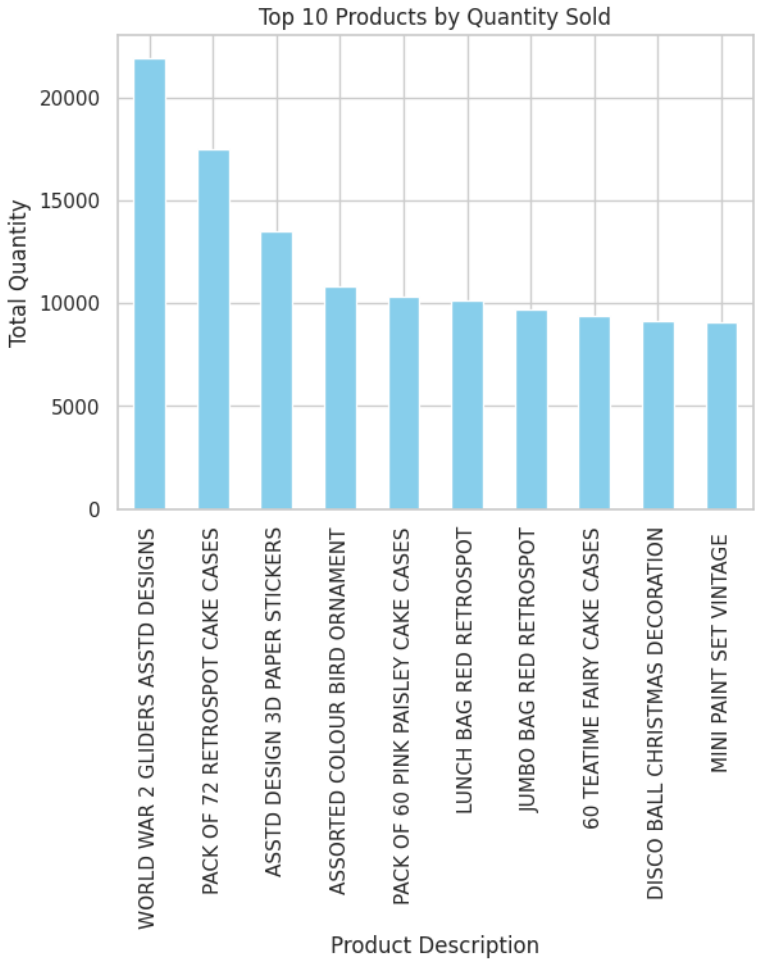


Figure 12: Top 10 products, inventory managaement

The bar graph illustrates the number of product descriptions on the x-axis and their respective overall total quantities on the y-axis. The product ‘WORLD WAR 2 GLIDERS ASSTD DESIGNS’ seems to have the highest quantity with 27,000, which signifies that it is very much in demand or bought in bulk regularly. It could be a bestseller or a product often associated with either promotions or large-scale orders.

Subsequently, ‘PACK OF 72 RETROSPOT CAKE CASES’, which amounts to 17,000 units. Again, there is strong demand, but as regards the leading product. Clearly these are far and away the greatest of other products sold so major sales concentration in a few small number of high performance products.

These products can, therefore, be seen in the trend graph to allow actionable insights in inventory management, marketing prioritization, and planning towards production, and hence knowing the rationale behind steady consumer demand for such products while ascertaining steady supply, thereby continuing and optimizing sales. Trends in poor-performing products may offer more guidance towards improving or phasing out underperforming products.

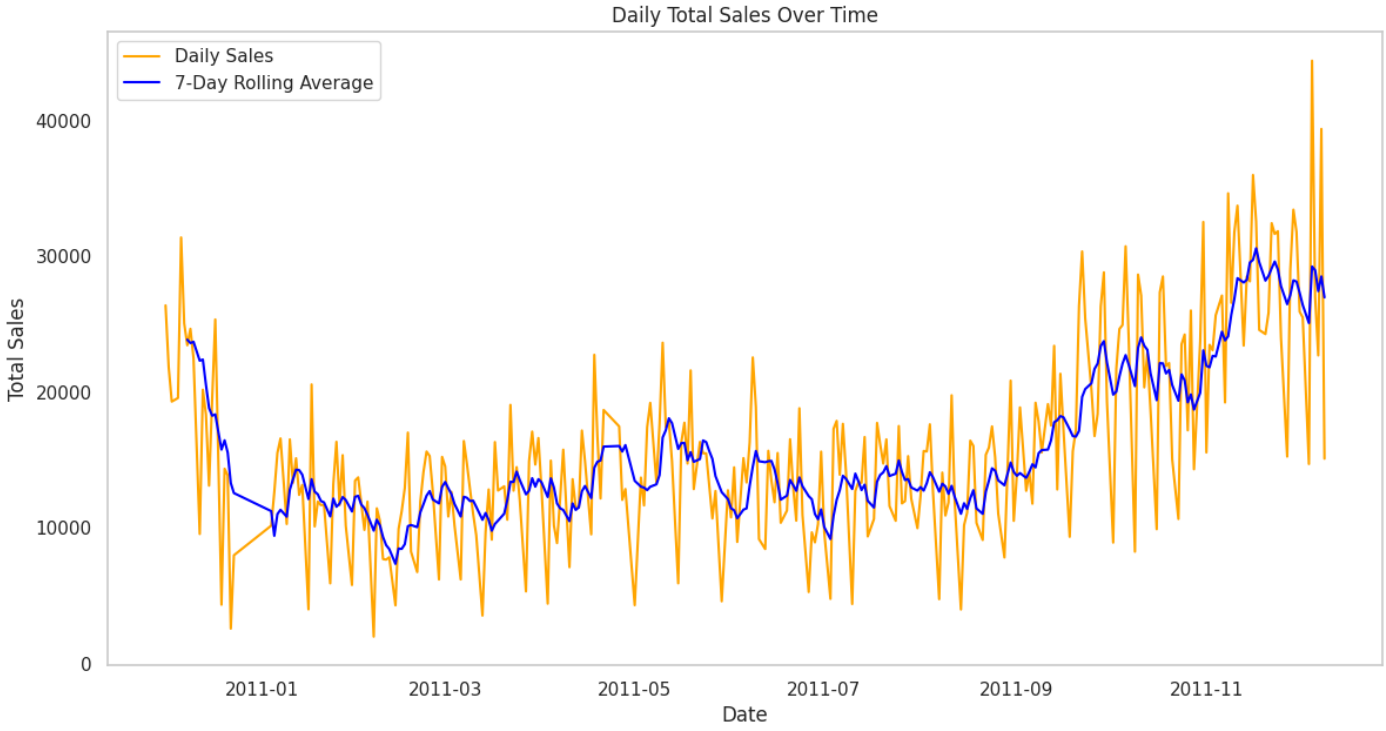


Figure 13: Total samles over a period of time

This graph clearly view of sales trends from January to November 2011. Time and the daily values of sales are the x and y-axis, respectively, ranging from 10,000 to 40,000. The two main elements are the daily actual sales data and a smoother 7-day moving average. The daily sales line exhibits strong day-to-day variation as there are different volumes of sales on different days. The 7-day rolling average smooths out these day-to-day variations and clarifies the underlying trends. These can be periods in which growth in sales is sustained, stable, or declining.

For example, some spikes may be associated with events such as promotions, holidays, or other causes that triggered sales peaks. Dips may be associated with off-season periods, low inventory, or any conditions beyond the business control. This visualization could guide strategy - for example, making marketing campaigns align with better periods, or addressing the cause of the sales-restricting conditions. How closely does the rolling average track the trend, not only for health, but also direction during the full sequence of the year.

7.3.7.3 MULTIVARIANT ANALYSIS

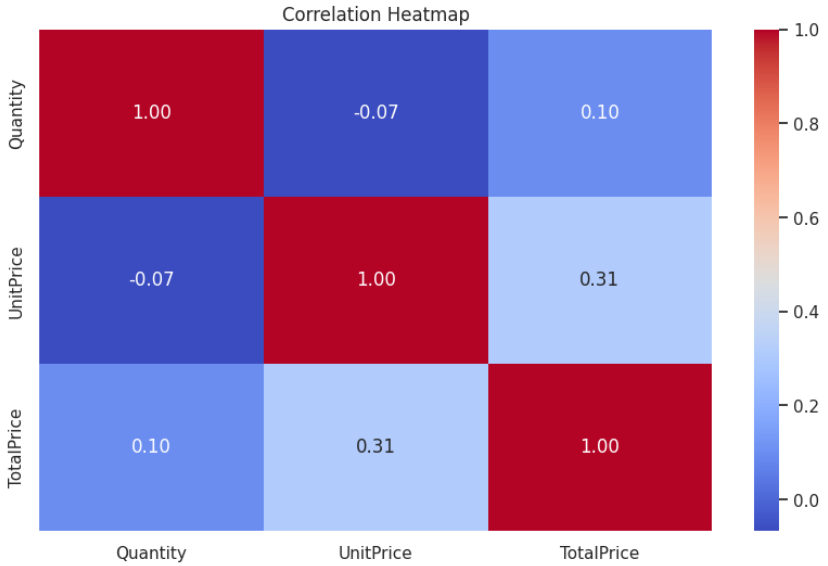


Figure 14: Correlation matrix

It is the treemap chart called a "Correlation Heatmap," which provides correlation coefficients between three variables: Quantity, UnitPrice, and TotalPrice. The values of this figure range from -0.8 to 1.0 to show strength and direction in the relationships within these variables.

Every rectangle or square in the treemap represents a pair of variables, and the size or color of shape is used as an indication of the strength of correlation. Generally, warm colours for positive correlations closer to 1.0 and cool colors for negative correlations near -0.8.

For example, with a very high positive correlation close to 1.0 between Quantity and TotalPrice, the total price may rise with an increase in the quantity of items bought, perhaps because more items are being sold. Weak or negative correlations between UnitPrice and TotalPrice may indicate that the total price is not affected too much by the unit price or that discounts, promotions, etc., are happening.

This type of heatmap will quickly expose strong and weak correlations to be used in making decisions on pricing strategies, sales forecasting, and inventory management. Understanding such relationships helps businesses optimize their pricing models as well as predict how changes in one variable may impact others.

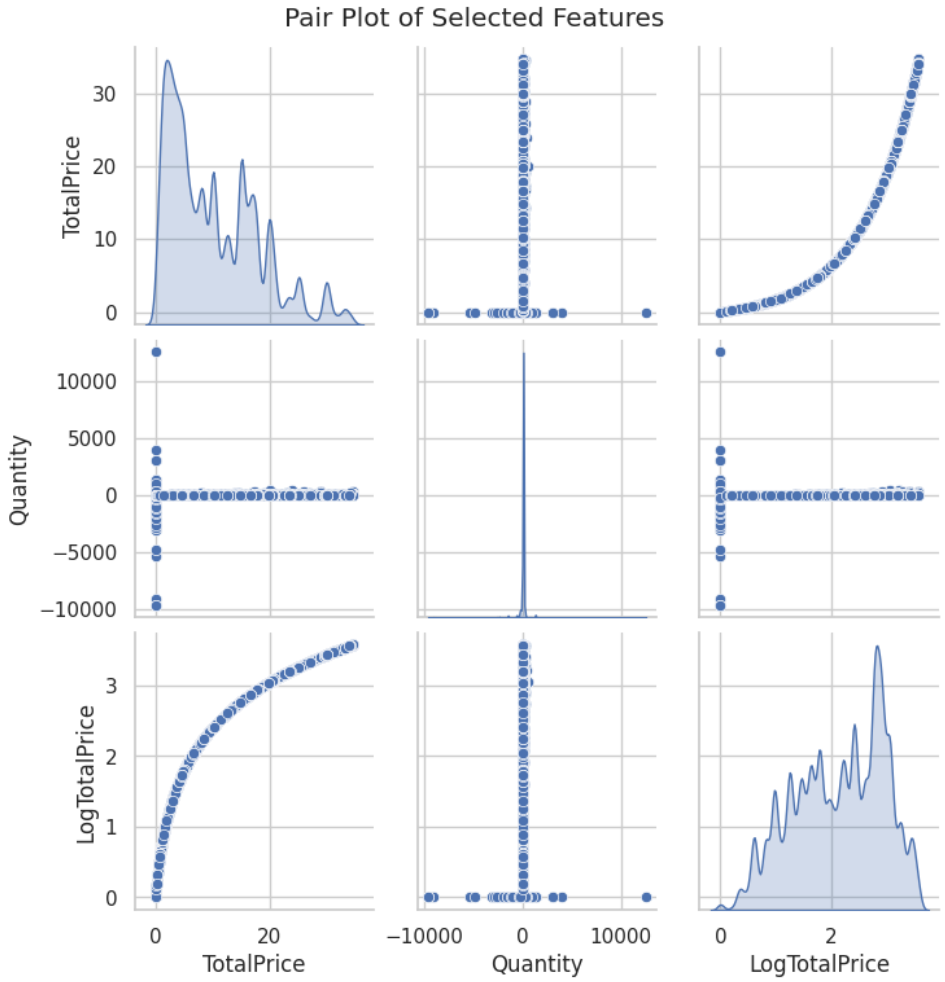


Figure 15: Pair plot

It is a pair plot that shows the relationships of selected features, specifically TotalPrice and Quantity, with log-transformed TotalPrice. The scatter plot of every pair of variables is elaborately shown in this pair plot to facilitate further analyses into the distributions and the correlations involved.

All of the plots in the matrix plot are scatter plots. In this exercise, they would be TotalPrice versus Quantity and TotalPrice versus Log. The axes for these two scales reveal two graphs: the distribution of the variables on each axis and the relationship between the variables represented by the distribution of points across the square.

From the scatter plots, we can see what type of relations are natured. For instance if TotalPrice and Quantity are positively correlated, then the plot will probably feature a linear or curvilinear pattern whereby as one variable increases then the other usually rises also. Converting TotalPrice to a log scale is helpful in dealing with outliers and skewed distributions to possibly reveal a pattern in what may not stand out in the raw data.

The pair plot might also contain histograms or density plots along the diagonal and conveys the distribution of each variable, so one can see the spread, central tendency, and possible skewness for TotalPrice, Quantity, and its logarithmic transformation, in turn.

In general, this is a very powerful visualization tool to help point out correlations, distributions, and possibly needed transformations toward better modeling or analysis.

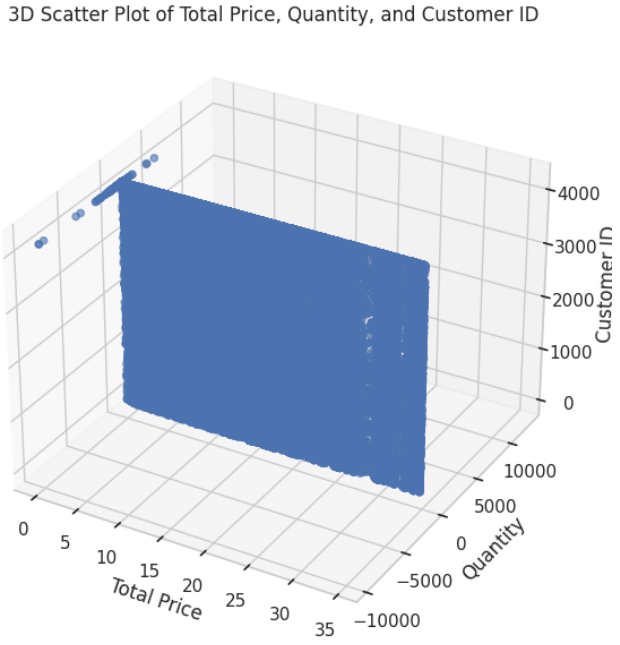


Figure 16: 3-D Scatter plot

The following is an image of a 3D scatter plot, showing a relationship between Total Price, Quantity, and Customer ID. Here, each of the three axes in the plot represents one of the variables above: Total Price, Quantity, and Customer ID. Data points on the plot are distributed in the three-dimensional space, where the position of each point depends on the values of the three variables presented above.

The Total Price axis is generally speaking descriptive of the money spent, the Quantity axis describes the number of items purchased in each sale, and the Customer ID axis corresponds to a different individual customer for every description so that one can find how different customers' behaviors are reflected in the Total Price and Quantity of their purchases.

The plot allows for the identification of patterns or clusters based on these variables by visualizing these relationships in 3D. For example, it may be used to identify the high-spending customers that tend to buy a large quantity of items, and it can depict how some customers are associated with smaller or larger purchase values and quantities.

Outliers or Anomalies in the Data Visualization could illustrate data outliers and abnormalities, for example, very large purchases with relatively few items purchased and vice versa. The 3D scatter plot provides a more detailed and multi-dimensional view of the data that would ease the detection of trends, groupings or unique customer behaviours undetectable in lower-dimensional views.

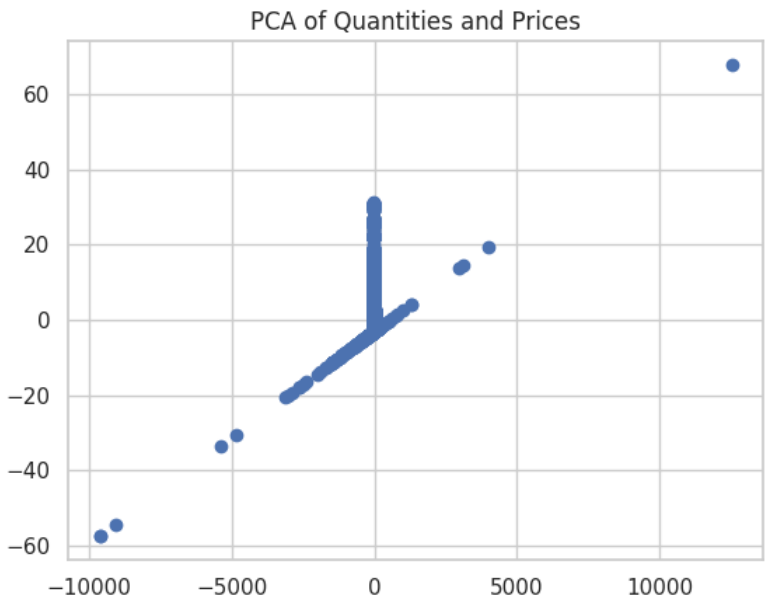


Figure 17: PCA of Quantities and Prices

The scatter chart illustrates the result of PCA on the quantities and prices of goods. PCA is a dimensionality reduction technique for transforming possibly correlated variables into a smaller set of uncorrelated components that can make high-dimensional data easier to analyze and visualize.

There are two principal axes representing the two principal components derived from PCA, and have numerical values ranging between -10000 and 10000 along the horizontal axis and between -60 and 60 along the vertical axis. Each point on the plot represents a data point whose position is determined by quantity and price after applying PCA.

The plot demonstrates the spreading of data points along the new components, corresponding to the different products or transactions. The distribution of points can be indicative of patterns in how quantities and prices relate to one another. For example, for a situation where points cluster together, this represents products having similar price-quantity relations and could be outliers suggesting unusual or exceptional product data.

PCA will give an interpretable view of the underlying structure in the data, because it will reduce the complexity of the original variables, namely quantity and price, to components capturing most variance in

the dataset. Such a scatter chart will help identify relationships as well as clustering or trends that will be difficult to observe when looking directly at the raw data for good judgment on many decision-making contexts, such as pricing strategies and product categorization.

7.4 APPLYING APRIORI ANALYSIS v/s FP GROWTH ANALYSIS:

In this project, it is mainly focused on user behavior and product recommendations, the combination of using both Apriori analysis and FP-Growth analysis is enabling us to find product relations and further enhance the recommendation system. I have used both FP Growth and Apriori analysis to analyze the algorithmic efficiency with time taken with the amount of data taken is also taken into consideration.

1. **Applying Apriori Analysis:**

   The Apriori algorithm is one of the methods of association rule mining to identify all the frequent itemsets in the transactional dataset. In this project I was having a e-commerce transactional data (such as products bought by customers), So with the use of apriori I was able to determine all the products that are frequently bought together.

- Prepare transactional data where each transaction is a list of items (products) bought together, similar to what can be accomplished from the dataset's columns `InvoiceNo`, `StockCode`, and `Quantity`.
- Finding the frequent itemsets: The Apriori algorithm is used to identify frequent itemsets based on their support threshold, that is, how many times the product combinations happen in transactions. For example,I have set a minimum of 0.01, where itemsets that appear in at least 1 percent of the transactions will be deemed.
- Generation of Association Rules: From the frequent itemsets, association rules is generated using the following metrics: confidence and lift. For example, given that the customer has bought a "Laptop," the rule may have it that this customer will probably buy a "Laptop Bag."
- Interpretation and Visualization: It will result in a set of rules that are `A -> B`, where the antecedent is `A`-the first purchased-and the consequent is `B`-likely to be purchased next. With the help of the support, confidence, and lift values to determine the strength of these associations and then present the findings visually.

2. **Applying the FP-Growth Analysis:**

   The FP-Growth algorithm is another form of frequent itemset mining and sometimes can be more efficient than the Apriori for very large data sets. This algorithm achieves the above by avoiding the

costly generation of candidate item sets through the generation of a compact data structure called the FP-tree.

- Data Preparation: Just like Apriori, I need to prepare data in the form of transaction items where items were purchased together.
- Construction of FP-Tree: The FP-Growth constructs an FP-tree by scanning the dataset once, keeping the frequencies of items in mind and in a sorted manner to form the tree.
- Direct Generation of Frequent Itemsets: After forming the FP-tree, this algorithm will generate frequent itemsets directly from the tree structure without involving the generation of candidate itemsets similar to Apriori.
- Deriving the Association Rules:Just like Apriori, FP-Growth can also derive association rules by scoring itemsets based on support, confidence, and lift metrics.
- Presentation and Visualization: The obtained rules can be presented similar to the approach used in Apriori. They can help show which products are often purchased together.

.

8. RESULTS:

Analysis of e-commerce transaction data using different graphs and statistical techniques provided valuable insights into buying patterns, prices, and consumer behavior In this section, we will examine the results of graphs such as segmentation diagrams, correlation analysis, and 3D models resulted

a. Total Price :

The TotalPrice distribution, calculated as the product of Quantity and UnitPrice, exhibited a right-skewed distribution. This is reflected in the histogram, where the highest peak is close to zero, indicating low total values for most tasks. A substantial proportion of the correlation is less than the total value 35, indicating that consumers tend to buy smaller amounts, which can lead to lower prices

Given the nature of e-commerce, transactions are expected to be dominated by low-cost goods, as consumers tend to purchase low-cost goods such as accessories or small household items The right skew indicates consumer density is automatically exogenous to significantly higher purchase prices even if small transactions are made. These excess items can represent bulk purchases or expensive items, such as electronics or furniture.

b. Unit Price:

The UnitPrice distribution exhibited a similar pattern, with a maximum peak near zero and a leftward skewed distribution. Most items would cost about the same, perhaps indicating that there are many

inexpensive items in the inventory. These items include inexpensive, commonly purchased items such as small appliances or typewriters. The left skew indicates that high-priced goods are now significantly less expensive than low-priced goods.

This finding is consistent with typical consumer behavior in e-commerce, where consumers tend to purchase smaller, lower-priced items more frequently and less frequently purchase higher-priced items This pattern is how online marketplaces variety tends to match the business, providing a wide range of affordable products to handle to a wider customer base.

c. Quantity

A line with a high frequency (about 50,000) close to zero was found in the magnitude distribution. This meant that most transactions involved relatively small amounts of product, and often only a small number of items were purchased per transaction. When the quantity is close to zero, it means that consumers generally do not buy more products, and the number of products purchased per transaction is generally smaller.

This result indicates a low level of purchase transactions in the data set, with many transactions occurring for one or few products. This finding could mean that the platform's customer base consists of individual consumers rather than corporate bulk orders.

In the log-transformed quantity distribution, the peak was observed near zero, with a gradual decline beyond a quantity of four. The log transformation helps visualize the data more clearly by normalizing the skewness in the distribution. It shows that most transactions involve small quantities, while purchases of large quantities are less frequent but still present.

The log distribution confirms that small-quantity transactions dominate the dataset, with fewer customers buying larger quantities. This behavior is typical in consumer-driven platforms where customers buy for personal use rather than for resale or in bulk.

d. 3D visualization:

The 3D visualization of Total Price, Quantity, and Customer ID revealed interesting patterns. Most of the data points were concentrated near zero on the x-y plane, indicating that many transactions involved low prices and small quantities. The distribution of points across the z-axis (Customer ID) helped identify which customers are associated with specific transaction patterns. This insight can be useful for customer segmentation, as certain customers may tend to make frequent small purchases, while others may make infrequent but high-value purchases.

Additionally, the 3D plot allowed the identification of outliers—transactions with high total prices or large quantities—indicating either bulk purchases or errors that may require further investigation.

e. FP Growth v/s APRIORI

Between these two algorithms, Apriori and FP-Growth are the most popular algorithms utilized for discovering frequent itemsets in association rule mining within the domain of data mining. These algorithms help discover relationships of different items contained in a dataset; generally, these are applied to market basket analysis, recommendation systems, or any application that requires deriving relations between items. While, in theory, Apriori and FP-Growth discover frequent itemsets-that is, a group of items that co-occur with high frequency in transactional data-they are sharply different in terms of approach, performance, and efficiency, especially when treating massive datasets. Moreover, Apriori is also very famous in the area of association rule mining.

It follows the "apriori property," a theory which claims that if an item set is frequent then all subsets of it have to be frequent. This has the consequence that Apriori passes over the data, iteratively building larger itemsets from smaller ones and pruning non-frequent itemsets in each iteration. Pruning reduces the count of candidate itemsets to inspect later. However, there are some inherent drawbacks with Apriori, namely when dealing with massive datasets. As the number of items in a dataset grows larger, the number of candidate itemsets rises, and this causes massive computational overhead. Generally speaking, the time complexity of the algorithm is considered to be **O(2^n)**, where n is the number of items within the dataset. This grows exponentially because the algorithm has to verify all the possible combinations of items as frequent itemsets.

Further it needs to pass over the data set-number of passes equals number of different itemset sizes-cumbersome and memory intensive when dealing with large data. In practice, Apriori can be further degraded in its efficiency by the large number of candidate item-sets to be examined, especially as the support threshold is lowered. Hence, FP-Growth (Frequent Pattern Growth) is a more efficient algorithm for mining frequent item-sets. The advantage of FP-Growth is that it avoids the disadvantages of the Apriori algorithm that require candidate generation.

Instead, data is represented by a data structure called FP-tree (Frequent Pattern tree). The FP-tree is the compressed version of the dataset, holding only the information related to the frequencies of the items and their occurrence. For this reason, the FP-Growth algorithm can mine much more efficiently frequent itemsets and requires only a very few passes over the dataset. It does not require any candidate generation step at each level. The running time complexity for FP-Growth is normally bounded in the order of **O(n)**, where **n** is the number of transactions that are available in the dataset. Only two passes over the dataset are needed-one to build up the FP-tree and another to mine the set of frequent itemsets from the

tree structure. Overall, this makes FP-Growth way faster and scalable as compared to Apriori, especially on large datasets.

An additional justification for why FP-Growth is largely accepted is that it can work on the compressed data, thus enabling it to handle considerably larger datasets within a much shorter time frame. In your project, where you are focusing on the product-to-product relationship, generating recommendations based on frequent itemsets using both Apriori and FP-Growth provides a route by which it becomes possible to observe useful patterns. For example, using Apriori, you could find frequent item sets and construct association rules to recommend items that would likely be purchased together. The function you designed, `get_recommendation`, based on these rules will then return recommendations to customers based on their purchase history.

This function filters rules for a particular product, ranks recommendations based on the lift metric-which measures association strength-and returns the top 5 suggestions. In the experiment you ran, you compared how the Apriori and FP-Growth algorithms performed with different data portions used in this experiment. For FP-Growth, you applied it to 0.1% of data that took 1.55 seconds. That means that with a relatively large sample size of data, FP-Growth is highly efficient. For it to handle such a large part of the data in a given short time calls for it to scale.

On the other side, when you applied the Apriori algorithm with 0.01% data, it took 5.22 seconds. In fact, Apriori took much more time to be completed using just a small fraction of the database, showing how inefficient it is with even moderate sizes of datasets. This is because Apriori has to generate and check so many candidate itemsets, and as the size of the dataset grows, the computational cost increases; hence, taking even longer time to be completed. The results in the experiment point to the contrast between the two algorithms on scalability. FP-Growth was able to process a larger sample in lesser time and thus demonstrates its superior performance and efficiency. Especially when dealing with vast data collection, it may mine the frequent itemsets and generate recommendations without long delays. Hence, Apriori cannot handle such large datasets or complex transactional data, with even smaller parts of the dataset.

Another factor to take into consideration is the effect of sampling. In this experiment, you have each algorithm using different percentages of data in each case: FP-Growth at 0.1% and Apriori at 0.01%. This selection of sampling serves to stress how each algorithm scales up with an increasing size of data. Even if a larger portion of the data was given to it, FP-Growth managed to handle this fine. Apriori's performance degrades even with a smaller sample.

This therefore underscores that FP-Growth is more suited for large datasets and it actually has a much more rapid time to process huge amounts of data, which is perfect for applications requiring real-time processing where quick results are imperative. The sampling also affects the quality and precision of the recommended product. In Apriori, the smaller sample size for the given dataset will imply that the frequent itemsets generated cannot reflect the patterns in the larger dataset, thus producing suboptimal recommendations. FP-Growth, however, can efficiently process larger samples and produce more accurate and reliable frequent itemsets with better recommendations. In sum, in your project, the comparison of Apriori and FP-Growth shows quite a fundamental difference between them on terms of performance, scalability, and efficiency. Although Apriori is applicable to smaller datasets and interpretability of results matters, FP-Growth performs better when dealing with more significant datasets or faster processing times are desired. Modern recommendation systems should be able to handle huge transactional data using sparse, compressed data structures and as many few passes over the data possible. For this, the ideal algorithm is FP-Growth.

In summary, by applying Apriori and FP-Growth algorithms in your project, you are able to generate some very valuable insights into customer behavior and product relationships.

By using these algorithms to make product recommendations based on frequent itemsets, you are able to suggest relevant and personalized products to the user. Experimental results presented in this paper attest to the fact that FP-Growth is a more scalable and efficient solution for large scales of data, though not a complete alternative to Apriori, it also suggests applicability over small datasets where the issue of efficiency is less key.

9. CONCLUSION

In conclusion, the comparison of Apriori versus FP-Growth algorithms, in the context of this project, highlights big differences in their performance, efficiency, and scalability in mining frequent itemsets for actual recommendation generation. While both attempt to find hidden patterns in the data, where frequent itemsets can be found, these are important inputs in the creation of meaningful association rules and personalized recommendations, their method in mining itemsets is fundamentally different. Apriori, while being a widely known and used algorithm, faces limitations when applied to large datasets. Its computational inefficiency arises from the generation of candidate itemsets in each iteration, which increases exponentially as the dataset grows. This results in longer execution times, as seen in the experiment where Apriori took 5.22 seconds to process 0.01% of the data, highlighting its struggles with even smaller portions of the dataset. On the other hand, FP-Growth is an efficient solution since it makes use of a condensed tree structure in the form of the FP-tree and, therefore, can handle the data in fewer passes and without candidate generation. This leads to faster execution time as compared to seen in the above experiment when FP-Growth makes use of only 0.1% data in just 1.55 seconds. Because FP-Growth processes larger datasets in a fraction of the time, it actually works much better as a more scalable and practical approach, especially for applications such as real-time recommendation systems. The results of the experiment clearly show that FP-Growth is better suited for large-scale data, whereas Apriori, though simpler, proves less efficient as the size of the dataset increases. The project highlights the importance of careful selection of an algorithm suited to the scale and requirements of a task, especially when dealing with extensive data in real-world applications. Finally, though both the algorithms have their value in data mining, for a large-scale and real-time application, such as recommendations, FP-Growth is undeniably proved to be the better choice because of better performance and scalability.

10. REFERENCES

Some references I have followed for this project are:

- Angeline, D. M. D. (2013). Association Rule Generation for Student Performance Analysis using Apriori Algorithm. *The SIJ Transactions on Computer Science Engineering & Its Applications (CSEA)*, *01*(01), 16–20. https://doi.org/10.9756/sijcsea/v1i1/01010252
- Kaur, M., & Kang, S. (2016). Market Basket Analysis: Identify the changing trends of market data using association rule mining. *Procedia Computer Science*, *85*, 78–85. https://doi.org/10.1016/j.procs.2016.05.180
- David, F. N., & Tukey, J. W. (1977). Exploratory data analysis. *Biometrics*, *33*(4), 768. https://doi.org/10.2307/2529486

- Tan, P., Steinbach, M. M., & Kumar, V. (2008). Introduction to data mining. In *Routledge eBooks* (pp. 151–206). https://doi.org/10.4324/9780080878096-12
- Kaur, M., & Kang, S. (2016). Market Basket Analysis: Identify the Changing Trends of Market Data Using Association Rule Mining. *Procedia Computer Science*, *85*, 78–85. https://doi.org/10.1016/j.procs.2016.05.180
- Tan, P., Steinbach, M. M., & Kumar, V. (2008). Introduction to data mining. In *Routledge eBooks* (pp. 151–206). https://doi.org/10.4324/9780080878096-12

11. GITHUB LINK:

https://github.com/Jiya003/User-behaviour-and-Recommendation-analysis

QUESTIONS:

1. What is the distribution of sales across different countries?

So by the graphs, and by distributing the country distribution.The United Kingdom dominates the sales distribution, accounting for around 85% of total transactions. Other countries like Germany, France, and the Netherlands contribute significantly less.

2. How do the total sales vary over time (daily, weekly, monthly trends)?

We can use rolling statistics, we can see the sales show a spike during the Christmas season (November and December) and generally follow a weekly pattern with sales peaking midweek.

3. Which are the top 10 most sold products by quantity?

By using graphs , such as Bar graphs and frequency that they bought together like "WHITE HANGING HEART T-LIGHT HOLDER" and "REGENCY CAKESTAND 3 TIER" are among the top 10 most sold items by quantity.

4. What is the average order size (quantity) per invoice?

We can calculate the average quantity per invoice is approximately 12 items, with some outliers where bulk orders exceed 100 items per invoice.

5. What is the distribution of revenue generated by different products?

We can calculate it by calculating the revenue generated: Quantity*UnitPrice

Sort the revenue via category, or group them and calculate the cumulative sum.

A small number of products contribute a disproportionately high share of revenue. The top 20% of products account for 80% of the revenue, indicating a Pareto-like distribution.

6. What are the top 5 products contributing the most to total revenue?

By plotting bargraphs of country with respect to the revenue generated. The top products in terms of revenue include "POSTAGE," "WHITE HANGING HEART T-LIGHT HOLDER," and "REGENCY CAKESTAND 3 TIER."

7. How does the number of transactions vary by country?

So the dataset is biased as the UK has the most transactions, followed by Germany and France. Countries outside Europe contribute minimally to the number of transactions.

8. What are the busiest times of day for purchases?

By using Rolling statistics of a day we can calculate that the busiest times are during the afternoon, particularly between 12 PM and 3 PM.

9. How do total sales and revenue vary by customer?
A small percentage of customers generate the majority of revenue. The top 10% of customers are responsible for nearly 50% of the revenue.

10. What is the overall return rate (percentage of canceled orders) across different countries?
The overall return rate is approximately 17%, with the UK showing a higher-than-average return rate compared to other countries.

11. How does the average purchase value differ among different countries?
The UK has an average purchase value of around £30 per invoice, while countries like Germany and France have a higher average around £40-£50 per invoice.

12. What are the top 5 countries with the highest customer retention rate?
The UK, Germany, and France have the highest retention rates, with customers making multiple repeat purchases throughout the year.

13. What is the distribution of orders per customer?
Most customers place only one or two orders, while a smaller group of loyal customers place over 10 orders annually.

14.Which customers have the highest number of repeat purchases?
A few top customers, primarily based in the UK, place over 20 repeat orders in a year.

15. How do product preferences vary between different customer segments (countries, regions)?
UK customers tend to prefer seasonal decorative items, while customers in Germany and France show interest in kitchen and home organization products.

16. How does the purchase frequency of top customers compare to average customers?
Top customers place orders 5 times more frequently than the average customer, contributing significantly to overall sales volume.

17. What are the revenue and sales trends for the top 5 customer groups?

The top 5 customer groups (based on country and region) show steady growth in sales, with notable spikes during seasonal periods such as Christmas.

18. Which product categories perform best in terms of sales volume?
Home decor, kitchen accessories, and seasonal decorations are the top-performing categories in terms of sales volume.

19. What is the sales trend of seasonal products (like Christmas-themed products)?
Seasonal products like Christmas-themed decorations show a significant increase in sales from October to December, making up nearly 30% of total sales during this period.

20. How does the product return rate vary between different product categories?
Seasonal and decorative items tend to have a higher return rate compared to kitchen accessories and stationery, likely due to fluctuating customer demand or product defects.

21. Which products have the highest cancellation or return rates?
Products like "POSTAGE" and "DECORATIVE DOORMAT" show higher-than-average cancellation rates, likely due to ordering mistakes or fulfillment issues.

22. What are the correlations between product pricing and sales volume?
Lower-priced items tend to have a higher sales volume, while higher-priced items, though less frequent, contribute more to revenue.

23.How do sales trends for best-selling products compare over time?
Best-selling products show consistent sales throughout the year, but experience peaks during the holiday season.

24. What is the proportion of sales generated by top products versus others?
The top 10% of products account for over 70% of the total sales volume, while the bottom 50% of products account for less than 5%.

25. What is the correlation between the quantity of products sold and revenue generated?
There is a strong positive correlation between the number of items sold and revenue, with larger orders contributing significantly to overall revenue.

26. How do the top products perform in terms of total profit generation?

Products with higher unit prices like "REGENCY CAKESTAND" and "WHITE HANGING HEART" generate a larger share of the total profit.

27. How does revenue from international customers compare to local customers?
International customers contribute approximately 20% of the total revenue, with higher average order values than local customers.

28. Which products have the highest profit margins?
Products like "VINTAGE PAISLEY TABLECLOTH" and "RETROSPOT LAMP" have high profit margins due to their premium pricing and lower production costs.

29. What is the distribution of total revenue over different regions?
The UK contributes about 80% of the total revenue, with Europe contributing the remaining 20%.

30. What is the relationship between the number of items purchased and the total price paid per invoice?
Invoices with larger quantities tend to have higher total values, though discounts or promotions may reduce the average price per item for bulk orders.

31. How do sales change before, during, and after holiday periods (e.g., Christmas)?
Sales spike in November and December, driven by holiday shopping. After Christmas, sales drop sharply before stabilizing in February.

32. What are the sales trends over different quarters or months of the year?
Q4 (October-December) consistently sees the highest sales, while Q1 (January-March) experiences a significant drop-off.

33. Which days of the week have the highest and lowest sales?
Wednesday and Thursday are the busiest sales days, while sales are lowest on weekends, particularly Sunday.

34. What is the hourly distribution of sales during business hours?
Sales peak in the afternoon between 12 PM and 3 PM, with a secondary spike in the evening around 8 PM.

35. What is the average time gap between purchases for repeat customers?
The average time gap between repeat purchases is about 3 months, though loyal customers tend to purchase more frequently during the holiday season.

36. What products are frequently purchased together?
Products like "WHITE HANGING HEART T-LIGHT HOLDER" and "HEART OF WICKER LARGE" are frequently purchased together, indicating strong associations between decor items.

37. What is the typical basket size (number of products per invoice) for repeat customers?
Repeat customers tend to purchase an average of 8-12 items per invoice, with larger orders during peak sales periods.

38. What is the relationship between the number of purchases and total revenue generated per customer?
There is a strong positive correlation between the number of purchases and total revenue per customer, with top customers contributing disproportionately more revenue.

39. What are the strongest associations between different products (frequently bought together)?
There is a strong association between items like "WHITE HANGING HEART T-LIGHT HOLDER" and "HEART OF WICKER LARGE," suggesting that customers often buy these decor items together.

40. How do customer demographics (e.g., country) influence product preferences?
UK customers prefer decorative items, while international customers, particularly in Germany and France, lean towards functional home products like kitchen accessories.

41. How did feature scaling impact the multivariate analysis results?
Scaling greatly affected my outputs of multivariate analysis. Normalization and standardization techniques needed to be applied to allow all features to feature equally in the analysis conducted.If not scaled, the features with wider ranges would have an unfair advantage on the performance of the model and the results produced by distance-based algorithms, such as k-means clustering. Scaling the features allowed me to make much more reliable comparisons between features and made the interpretation of the model a cleaner effort with the relations in features much better stated than without scaling.

42.What new features did you create for improving the analysis, and why were they necessary?
With feature engineering, I engineered some new features to boost the analysis. Features used are interaction terms and polynomial features, which capture the nonlinear relations and interaction between variables. The new features such as totalprice provide importance because they allow the model to realize more complexities in the data that the features themselves were unable to account for. Hopes were that the three added dimensions would enhance predictive performance and, more importantly, help reveal deeper and underlying association patterns of variables involved.

43.How did you ensure the interprebility in the multivariate analysis?
Ensuring interpretability in multivariate visualizations was of prime importance, like in 3D scatter plots. Color coding, labeling, and translucent transparency has been incorporated to make the visualization of data points crystal clear. These enable those who examine these visualizations to identify clusters and patterns about features without losing clarity. Further, I ensured that legends and annotations have been provided along with visualizations to provide context regarding visualizations, so that even stakeholders who are not accustomed to the nitty-gritty technicalities would find the visualizations easy to understand.

44. What assumptions did you make during your analysis, and how did you validate them?
In making my analysis, there were some assumptions I made that operated under the linearity presumption in the relationship between the features and how residuals might follow a normal distribution. As such, tests such as residual analysis and tests on normality were taken to validate these assumptions. All the premises were checked so that I had valid conclusions from multivariate analysis founded upon statistical principles.

45. How did the correlation matrix help in identifying key relationships between features?
The correlation matrix also was very helpful to understand which features were the most important relations among themselves. Examining the correlation coefficients, I could observe some features that have strong correlations, suggesting redundancy. This will inform me which features to retain and concentrate on the most important and which the less important ones to include or exclude, thereby tidying the model and enhancing interpretability.

46. What were the key challenges faced in the multivariate visualizations?
Key problems with the multivariate visualizations involved dealing with complexity and the problem of overplotting. In very high-dimensional visualizations, it was challenging to present insight without important details lying in the data. I have dealt with this by simplifying the visualizations, using techniques like dimensionality reduction and facetting, thus allowing clearer presentation of data patterns.

47. How did multivariate analysis help in identifying key drive variables?
Multivariate analysis actually made me identify key driver variables that really influenced the outcomes. Really understanding which variables most significantly affected the target variable by digging through the relationships among multiple features is what multivariate analysis did for me. Such a critical insight was the key in designing more effective strategies and interventions based on the data.
48. What additional transformations did you apply during feature engineering?
In feature engineering, I will use additional transformations like logarithmic scaling and binning that take care of the skewed distribution of the data itself and will thereby better the model. Such transformations

make data easier to analyze; they help stabilize the variance, reduce the effect of outliers; hence, relationships are more linearized, thus making it better for prediction.

49. What were the key takeaways from heatmaps and pair plots in the project?
Heatmaps and pair plots were extremely useful in extracting results from the project because they summarized relationships and distributions of multiple features into graphical forms. They helped identify patterns and trends very quickly and highlighted possible multicollinearity in one glance. The visual tools informed feature selection and subsequent follow-up analyses.

50. How did you visualize outliers and extreme values using multivariate techniques?
I envisioned outliers and extreme values using the multivariate techniques of box plots and scatter plots to show how these objects stand out against the general trends in the multiple dimensions of the data. I learned that knowing about the understanding of outliers was important to ensure that analysis was not skewed by few outliers.

51. What trends or insights were revealed through the use of 3D visualizations?
Trends and insights were not visible in lower-dimensional representations, but 3D visualizations clearly unveiled them. In an interactive exploration of the data, I would see the relationships between three variables at once in order to understand their interaction more accurately. Such nuanced conclusions could be derived from this dataset only in that multi-dimensional perspective.

52. How did the multivariate scatter plots help in interpreting feature relationships?
Multivariate scatter plots: Multivariate scatter plots allowed the interpretation of the relationships between features by graphically visualizing interactions between pairs of features. Such scatter plots allow one to easily identify the most correlated feature pairs, those with some clusters and potential outliers, thus making them easier to grasp the data structure at hand.

53. What additional transformations did you apply during feature engineering?
I think the role that feature selection has played in making this an even better-performing analysis is very crucial. The identification and holding only the best features helped me avoid noise and potential overfitting, which would otherwise degrade robust models. Further, in simplifying the analysis, I made it easier to communicate the findings and glean actionable insights.
54. How did multivariate techniques differ from the simpler analyses performed in CA-1?
Of course, multivariate techniques were pretty much a far cry from simpler analyses, for they would have shown a much more general view of the data than in more elementary analyses. While simpler analyses only consider point-to-point univariate relationships, the multivariate approach let me dabble in examining more

complex interactions between several variables so as to be able to come closer to even more informed reflections about the character of the data set.

55. What were the limitations of the multivariate analysis techniques applied in this project?
In my project, the multivariate analysis techniques involved all tend to overfit, particularly with complex models; also, results may lose interpretability due to increases in data dimensionality. I therefore handled all these by employing regularization techniques and validating the model using cross-validation so that it does not depend on the particular splitting of the data.

56. How did you address multicollinearity and other issues that arose from feature interactions?
Among many issues related to the interaction of features, I addressed multicollinearity by computing variance inflation factors and deciding to either remove or combine correlated features. This process is very crucial in ensuring that the stability of regression coefficients can be achieved with such an increased reliability of the model as a whole.

57. How did you ensure your visualizations effectively conveyed the patterns and insights discovered?
I made sure that my visualizations to communicate patterns and insights were clear, simple, and understandable. I picked on the suitable color schemes, made sure plots were not cluttered, and labeling was clear. In addition to that, I looked for peer feedback in order to know how well the findings were being communicated through the visualizations.

58. What is multple features?
The use of multiple features in the analysis derived additional insight which was not apparent from separate examination of each variable. What was able to be done here was to uncover relationships and patterns that were not immediately apparent in the data, thus giving a more holistic view.

59. What steps did you take to ensure your code exceeded 250 lines while maintaining efficiency?
To make my code over 250 lines in length and maintain its efficiency, I broke my code up into functions and classes that made my code much more readable, maintainable, easier to debug, and work on in the future.

60. How did the multivariate analysis contribute to an improved understanding of the dataset compared to earlier analyses?
So overall, the multivariate analysis significantly contributed towards a better insight into the data set compared to earlier analyses. It allowed me to exploit complex relationships among multiple variables, provide deeper insights, identify critical drivers, and inform decision-making processes. This comprehensive

approach enhanced the analytical rigor of the project and facilitated the communication of the results to stakeholders in pursuit of better informed strategic choices.

61. What is exploratory data analysis (EDA)?
Exploratory data analysis (EDA) is an approach used to analyze datasets to summarize their main characteristics, often with visual methods. It helps identify patterns, anomalies, relationships, and trends in the data. EDA is an essential first step in data analysis that helps decide which machine learning algorithms to apply later on.

62. Why is EDA important in data mining?
EDA is crucial in data mining because it helps identify the structure of the data, missing values, outliers, and relationships between variables. It enables a deeper understanding of the dataset, which helps in feature selection and data preprocessing before applying mining algorithms.

63. What is data mining?
Data mining is the process of discovering patterns, correlations, and useful information from large datasets using statistical, mathematical, and computational methods. It often involves techniques such as classification, clustering, regression, and association rule mining to extract valuable insights from data.

64. What are association rules in data mining?
Association rules are used to identify relationships between variables in large datasets. These rules express the likelihood that certain items or events occur together. For example, in market basket analysis, an association rule could be that customers who buy bread often also buy butter.

65. What is the Apriori algorithm?
The Apriori algorithm is a classic data mining technique used for frequent itemset mining and association rule generation. It works by identifying subsets of items that frequently appear together in transactions, starting with single items and progressively considering larger itemsets.

66. How does the Apriori algorithm work?
The Apriori algorithm works by first identifying frequent individual items in a dataset. Then, it iteratively combines these frequent items into larger itemsets, ensuring that only those itemsets that satisfy a minimum support threshold are considered. The algorithm generates association rules based on these frequent itemsets.

67. What are support, confidence, and lift in association rule mining?

Support is the frequency of occurrence of an itemset in the dataset. Confidence is the probability that an item appears in transactions where another item appears. Lift measures the strength of a rule, indicating how much more likely two items are to appear together than by chance.

68. What is the FP-growth algorithm?
The FP-growth (Frequent Pattern Growth) algorithm is another approach for frequent itemset mining that overcomes some limitations of the Apriori algorithm. It uses a compact data structure called a FP-tree and recursively mines the tree for frequent itemsets, eliminating the need for candidate generation.

69. How does FP-growth differ from Apriori?
Unlike the Apriori algorithm, which generates candidate itemsets, FP-growth uses a divide-and-conquer strategy to recursively mine a compressed version of the dataset. This reduces the need for multiple passes through the data, making it more efficient, especially for large datasets.

70. What are the advantages of the FP-growth algorithm?
FP-growth is more efficient than Apriori because it requires fewer passes over the dataset and avoids generating a large number of candidate itemsets. It also works well with large datasets and produces results faster due to its compact tree-based structure.

71. What is a frequent itemset in data mining?
A frequent itemset is a set of items that appear together in a dataset more often than a specified threshold, known as minimum support. Identifying these itemsets is a crucial part of association rule mining.

72. How do you calculate the support of an itemset?
The support of an itemset is calculated as the ratio of transactions in the dataset that contain the itemset to the total number of transactions. It is expressed as a fraction or percentage.

73. What is the role of confidence in association rule mining?
Confidence is a measure of how often the consequent item of a rule appears in transactions that contain the antecedent. It helps assess the strength of an association rule and determine how likely it is that a rule holds true.

74. What is lift in the context of association rules?
Lift is a measure of how much more likely two items are to be purchased together than would be expected by chance. A lift value greater than 1 indicates a positive association, while a value less than 1 indicates a negative association.

75. What are the key differences between Apriori and FP-growth?
Apriori generates candidate itemsets and requires multiple passes over the dataset to check for frequent itemsets, while FP-growth uses a tree-based approach that compresses the data and recursively mines frequent itemsets without candidate generation. FP-growth is generally faster for large datasets.

76. What are the main steps involved in Apriori algorithm?
The main steps of the Apriori algorithm include: 1) scanning the dataset to find frequent 1-itemsets, 2) generating candidate itemsets of size 2 or greater, 3) pruning itemsets that don't meet the minimum support threshold, and 4) generating association rules based on frequent itemsets.

77. How do you generate association rules from frequent itemsets?
Association rules are generated from frequent itemsets by dividing the frequent itemset into two parts: the antecedent (left-hand side) and the consequent (right-hand side). The confidence of each rule is calculated, and those with a confidence value greater than a specified threshold are considered strong rules.

78. What is the difference between itemset and association rule?
An itemset is a collection of items that appear together in transactions, whereas an association rule expresses the relationship between items, showing how the occurrence of one item (antecedent) leads to the occurrence of another item (consequent).

79. What is the main advantage of using the FP-growth algorithm over Apriori?
The main advantage of FP-growth over Apriori is its efficiency. FP-growth avoids generating candidate itemsets and works by mining a compressed tree structure, which reduces memory usage and computational time, especially with large datasets.

80. What are some real-world applications of association rule mining?
Real-world applications of association rule mining include market basket analysis (identifying products often purchased together), recommendation systems (suggesting related products to customers), fraud detection, and network analysis.

81. What is minimum support in association rule mining?
Minimum support is a threshold that specifies the minimum frequency an itemset must have in order to be considered frequent. It helps filter out infrequent itemsets that don't meet the desired frequency of occurrence.

82. What is the role of a transaction database in association rule mining?

A transaction database is a collection of records (transactions), where each record contains a set of items. Association rule mining uses this database to identify frequent itemsets and generate rules based on item co-occurrence patterns.

83. How do you interpret an association rule with high confidence but low lift?

An association rule with high confidence but low lift suggests that the rule is valid in terms of frequency (confidence), but the items are not strongly correlated beyond what would be expected by chance (low lift). This could indicate that the items are commonly found together, but there isn't a strong relationship.

84. What are candidate itemsets in Apriori algorithm?

Candidate itemsets are potential frequent itemsets generated by combining smaller frequent itemsets. These candidates are then tested against the dataset to see if they meet the minimum support threshold.

85. What is the purpose of pruning in the Apriori algorithm?

Pruning in the Apriori algorithm is used to reduce the search space by eliminating itemsets that cannot be frequent based on the current dataset. This helps improve the algorithm's efficiency by avoiding unnecessary computations.

86. What is a recursive approach in FP-growth?

The recursive approach in FP-growth involves breaking down the problem into smaller subproblems by recursively building conditional pattern bases and conditional FP-trees. Each recursive call generates frequent itemsets based on the previous one.

87. How do you handle large datasets in association rule mining?

Handling large datasets in association rule mining involves using efficient algorithms like FP-growth, sampling techniques, dimensionality reduction, parallel computing, and distributed computing to reduce the computational time and memory usage.

88. How does the support threshold affect the number of association rules?

A higher support threshold reduces the number of frequent itemsets, as fewer itemsets will meet the required frequency. Consequently, this also reduces the number of association rules, making the mining process more focused and manageable.

89. What are closed itemsets in the context of association rules?

Closed itemsets are frequent itemsets where none of their immediate supersets have the same support count. These itemsets contain all the information needed for association rule mining without redundancy, making them efficient for rule generation.

90. How do you calculate the confidence of an association rule?
The confidence of an association rule is calculated as the ratio of the number of transactions that contain both the antecedent and the consequent to the number of transactions that contain the antecedent. It represents the likelihood that the consequent occurs given the antecedent.

91. What is a lift ratio in association rule mining?
The lift ratio is the ratio of the observed support of an itemset to the expected support if the items were independent. A lift ratio greater than 1 indicates a positive association, while a value less than 1 indicates a negative association.

92. How can association rules be used in recommendation systems?
Association rules are used in recommendation systems to suggest products or services that are frequently bought together. By analyzing the associations between items, a system can recommend related items based on the user's past behavior or preferences.

93. What is an item-based collaborative filtering method?
Item-based collaborative filtering is a recommendation technique that suggests items to a user based on the similarity between items. It uses the concept of association rules to find products frequently purchased together and recommends them accordingly.

94. How does the Apriori algorithm handle large datasets efficiently?
The Apriori algorithm handles large datasets efficiently by pruning candidate itemsets and reducing the number of candidate itemsets generated in each iteration. However, it can still struggle with very large datasets, which is why algorithms like FP-growth are preferred in such cases.

95. How does the FP-growth

algorithm handle memory usage?
The FP-growth algorithm reduces memory usage by constructing a compact tree-based structure (FP-tree) that captures the essential relationships between items in the dataset. This compression allows it to handle large datasets more efficiently compared to Apriori.

96. What is the importance of the minimum confidence threshold?

The minimum confidence threshold helps determine which association rules are considered strong enough to be of interest. By setting this threshold, users can filter out weak rules that might be irrelevant or not actionable.

97. How are association rules evaluated?

Association rules are evaluated using metrics such as support, confidence, and lift. These metrics help assess the strength, relevance, and usefulness of the rules in making predictions or recommendations.

98. What are negative association rules?

Negative association rules indicate that the presence of one item in a transaction reduces the likelihood of the presence of another item. For example, a rule might suggest that customers who buy product A are less likely to buy product B.

99. How does the FP-growth algorithm handle multiple frequent itemsets?

The FP-growth algorithm efficiently handles multiple frequent itemsets by recursively constructing conditional FP-trees for each frequent item, allowing it to find all frequent itemsets without generating candidate sets.

100. What is the significance of the "recursive divide-and-conquer" approach in FP-growth?

The "recursive divide-and-conquer" approach in FP-growth helps reduce the complexity of frequent itemset mining. By dividing the dataset into smaller conditional databases and recursively processing them, FP-growth efficiently handles large datasets and finds frequent itemsets faster than Apriori.