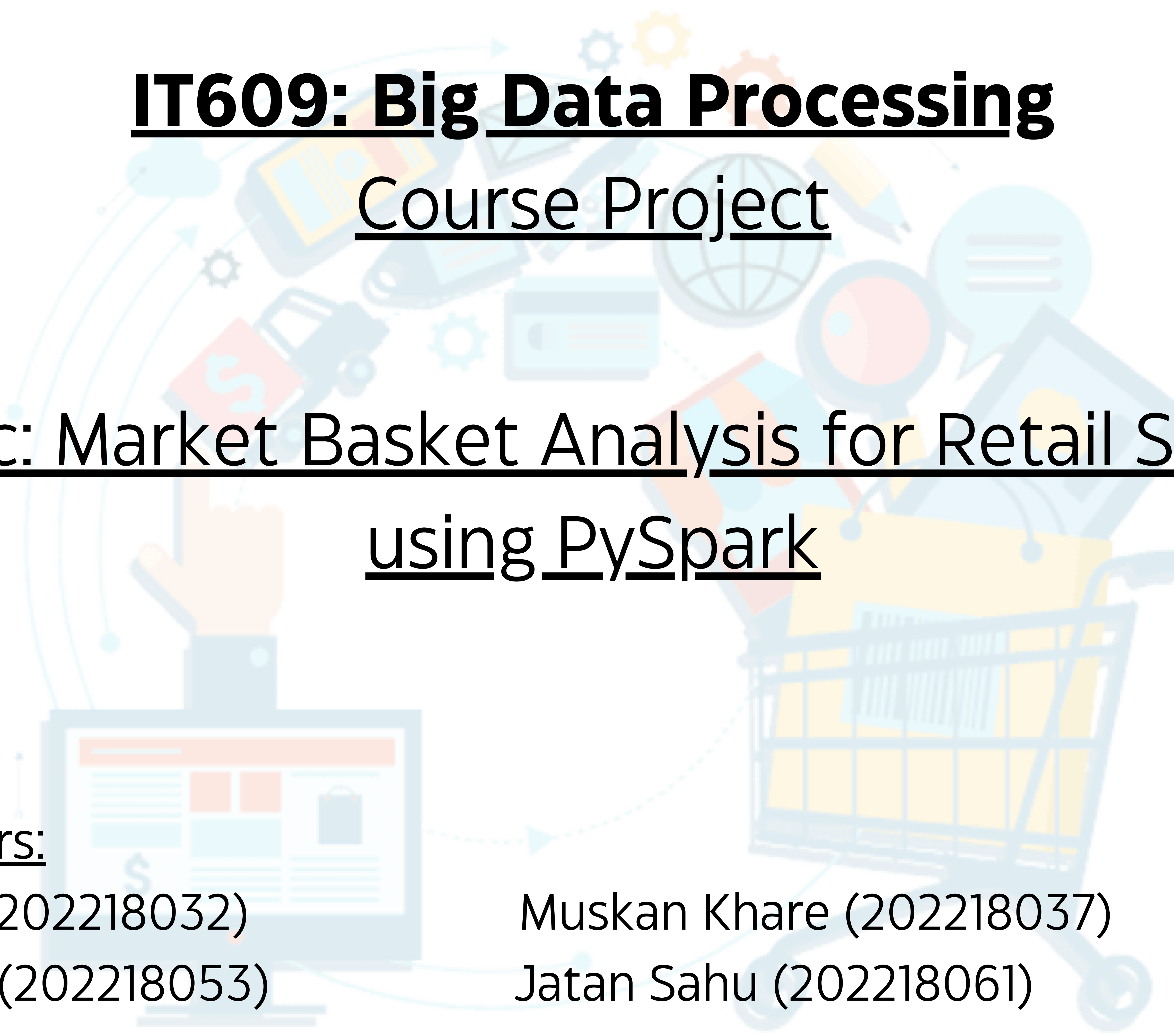


IT609: Big Data Processing

Course Project

Topic: Market Basket Analysis for Retail Stores using PySpark

Group Members:

Shreya Arora (202218032)	Muskan Khare (202218037)
Dhruv Solanki (202218053)	Jatan Sahu (202218061)

Problem Statement

Market basket analysis is a technique used to identify the relationship between the products that are frequently purchased together. The problem involves identifying the frequent itemsets and association rules from a large dataset of transactions. The target is to identify the frequent patterns within a reasonable time frame.

The main goal of this problem statement is to analyse the market basket, from large volumes of transactional data efficiently in lesser time than traditional database handling tools and libraries, and provide insights into the relationships between the products.

Use-Case of Market Basket Analysis

By analyzing customer purchase behavior and identifying patterns of frequently purchased items, retailers can gain valuable insights that can help them optimize their product offerings and increase sales.

Here are some specific use cases of Market Basket Analysis:

1. **Cross-selling and Upselling:** By analyzing customer purchase history, businesses can identify complementary products that are frequently purchased together and use this information to cross-sell or upsell customers.

2. **Product Bundling:** Market Basket Analysis can help businesses identify which products should be bundled together to create attractive packages for customers. This can increase sales and help clear out excess inventory.

3. **Inventory Management:** By understanding which products are frequently purchased together, businesses can optimize their inventory management by stocking the right amount of each product.

4. **Pricing Strategy:** Market Basket Analysis can help businesses determine the optimal price for products by analyzing the relationships between different products and how they are priced.

5. **Store Layout Optimization:** By analyzing customer purchasing patterns, businesses can optimize their store layout to encourage customers to purchase complementary products or increase sales of slow-moving products.

Tools and Technologies

In order to perform market basket analysis on a large dataset of transactions, a scalable and efficient approach is required. This can be achieved through PySpark.

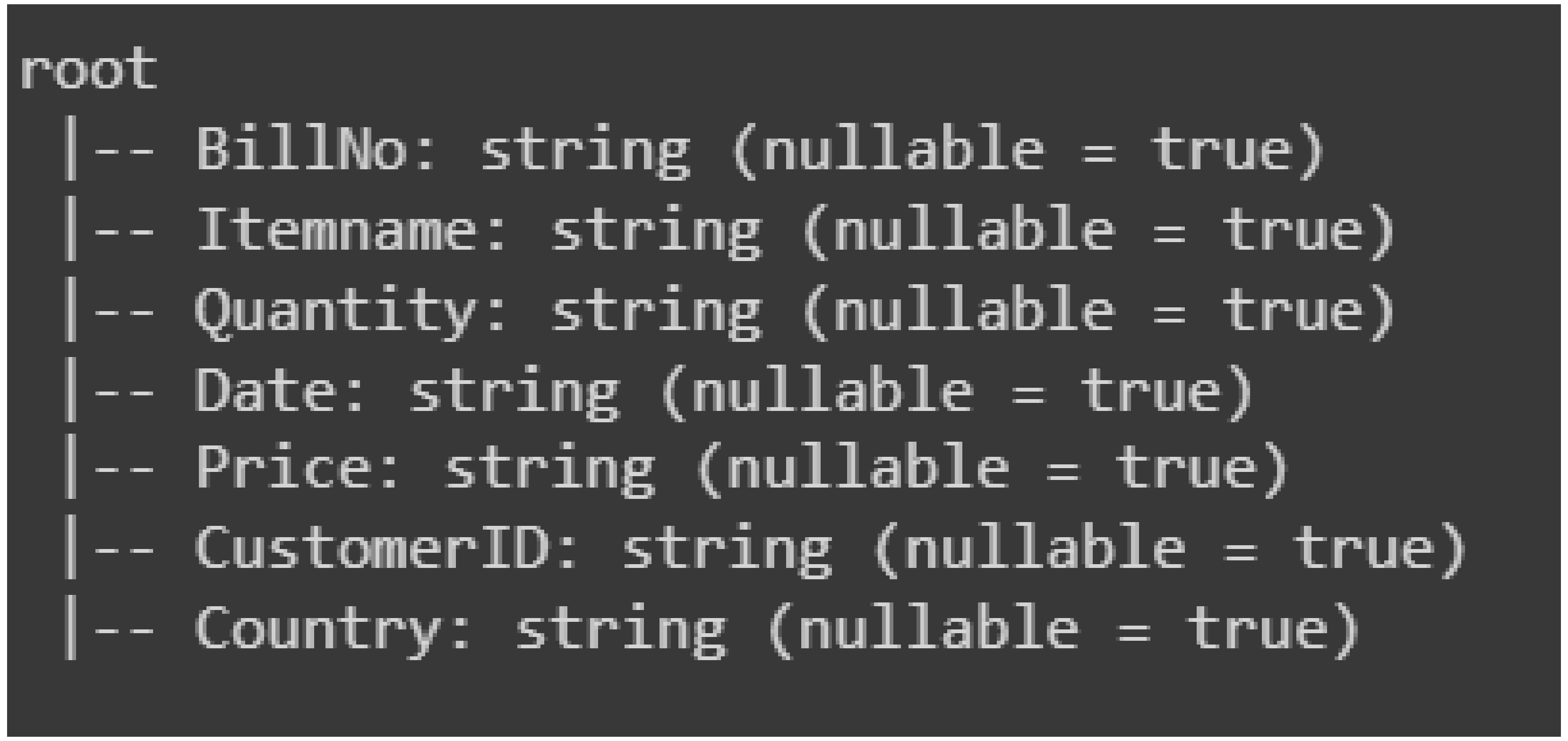
PySpark is a Python library that provides an interface to Apache Spark, a powerful open-source distributed computing system. Apache Spark provides a scalable and efficient platform for processing large amounts of data, making it an ideal choice for performing Market Basket Analysis on retail transaction data.

Dataset Description:

The Market Basket Analysis dataset publicly available on Kaggle has been used for this project.

Number of Attributes: 7

- **BillNo:** 6-digit number assigned to each transaction
- **Itemname:** Product name
- **Quantity:** The quantities of each product per transaction
- **Date:** The day and time when each transaction was generated
- **Price:** Product price
- **CustomerID:** 5-digit number assigned to each customer
- **Country:** Name of the country where the retail store is situated



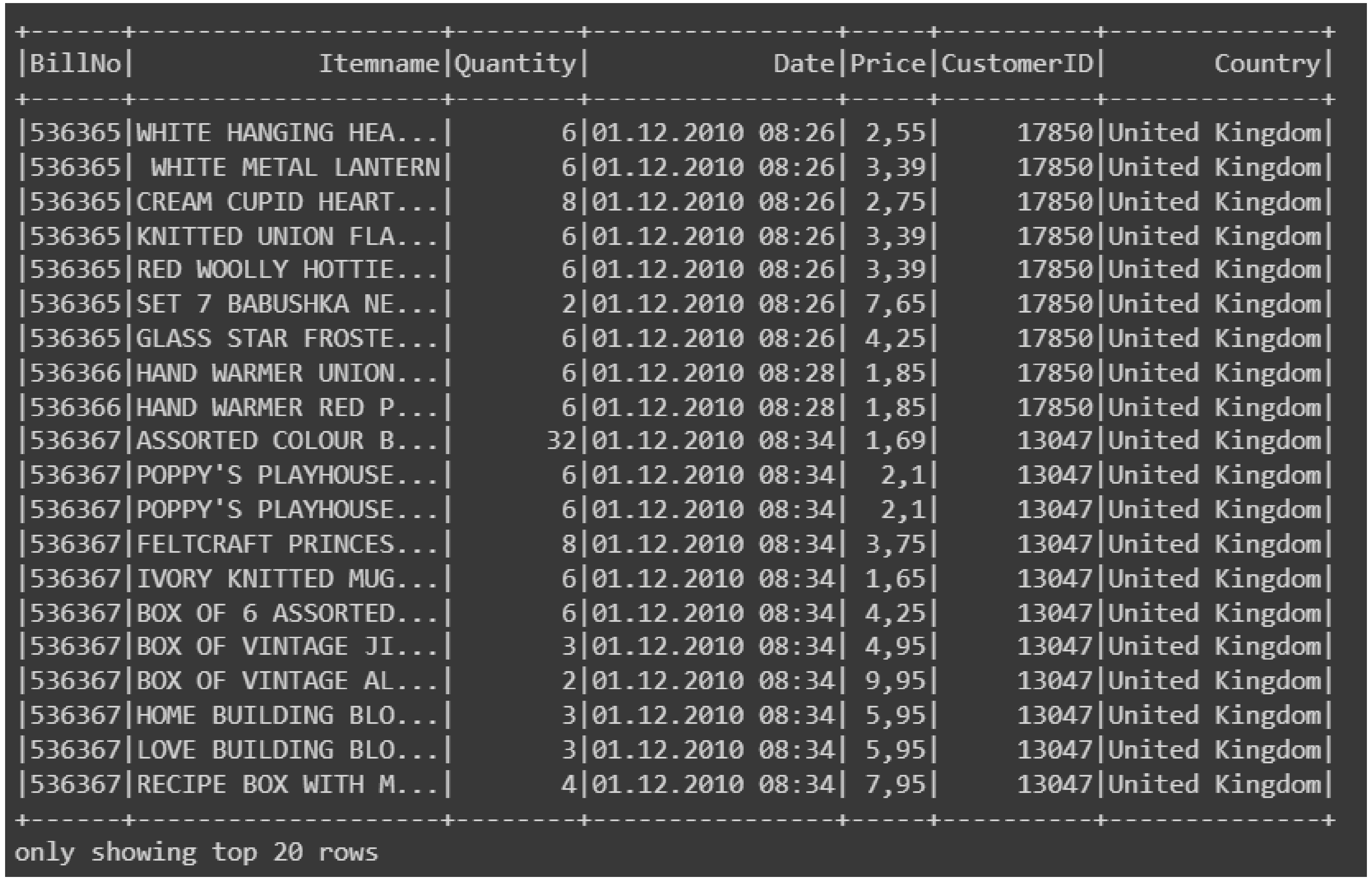
```
root
 |-- BillNo: string (nullable = true)
 |-- Itemname: string (nullable = true)
 |-- Quantity: string (nullable = true)
 |-- Date: string (nullable = true)
 |-- Price: string (nullable = true)
 |-- CustomerID: string (nullable = true)
 |-- Country: string (nullable = true)
```

Dataset Description:

The dataset has **5,35,260** tuples.

The number of unique values in each column are as follows:

- Bill No.: 21663
- Itemname: 4187
- Quantity: 691
- Date: 19642
- Price: 1268
- CustomerID: 2499
- Country: 31



```
+------+--------------------+--------+----------------+-----+----------+--------------+
|BillNo|            Itemname|Quantity|            Date|Price|CustomerID|       Country|
+------+--------------------+--------+----------------+-----+----------+--------------+
|536365|WHITE HANGING HEA...|       6|01.12.2010 08:26| 2,55|     17850|United Kingdom|
|536365| WHITE METAL LANTERN|       6|01.12.2010 08:26| 3,39|     17850|United Kingdom|
|536365|CREAM CUPID HEART...|       8|01.12.2010 08:26| 2,75|     17850|United Kingdom|
|536365|KNITTED UNION FLA...|       6|01.12.2010 08:26| 3,39|     17850|United Kingdom|
|536365|RED WOOLLY HOTTIE...|       6|01.12.2010 08:26| 3,39|     17850|United Kingdom|
|536365|SET 7 BABUSHKA NE...|       2|01.12.2010 08:26| 7,65|     17850|United Kingdom|
|536365|GLASS STAR FROSTE...|       6|01.12.2010 08:26| 4,25|     17850|United Kingdom|
|536366|HAND WARMER UNION...|       6|01.12.2010 08:28| 1,85|     17850|United Kingdom|
|536366|HAND WARMER RED P...|       6|01.12.2010 08:28| 1,85|     17850|United Kingdom|
|536367|ASSORTED COLOUR B...|      32|01.12.2010 08:34| 1,69|     13047|United Kingdom|
|536367|POPPY'S PLAYHOUSE...|       6|01.12.2010 08:34|  2,1|     13047|United Kingdom|
|536367|POPPY'S PLAYHOUSE...|       6|01.12.2010 08:34|  2,1|     13047|United Kingdom|
|536367|FELTCRAFT PRINCES...|       8|01.12.2010 08:34| 3,75|     13047|United Kingdom|
|536367|IVORY KNITTED MUG...|       6|01.12.2010 08:34| 1,65|     13047|United Kingdom|
|536367|BOX OF 6 ASSORTED...|       6|01.12.2010 08:34| 4,25|     13047|United Kingdom|
|536367|BOX OF VINTAGE JI...|       3|01.12.2010 08:34| 4,95|     13047|United Kingdom|
|536367|BOX OF VINTAGE AL...|       2|01.12.2010 08:34| 9,95|     13047|United Kingdom|
|536367|HOME BUILDING BLO...|       3|01.12.2010 08:34| 5,95|     13047|United Kingdom|
|536367|LOVE BUILDING BLO...|       3|01.12.2010 08:34| 5,95|     13047|United Kingdom|
|536367|RECIPE BOX WITH M...|       4|01.12.2010 08:34| 7,95|     13047|United Kingdom|
+------+--------------------+--------+----------------+-----+----------+--------------+
only showing top 20 rows
```

Pre-processing

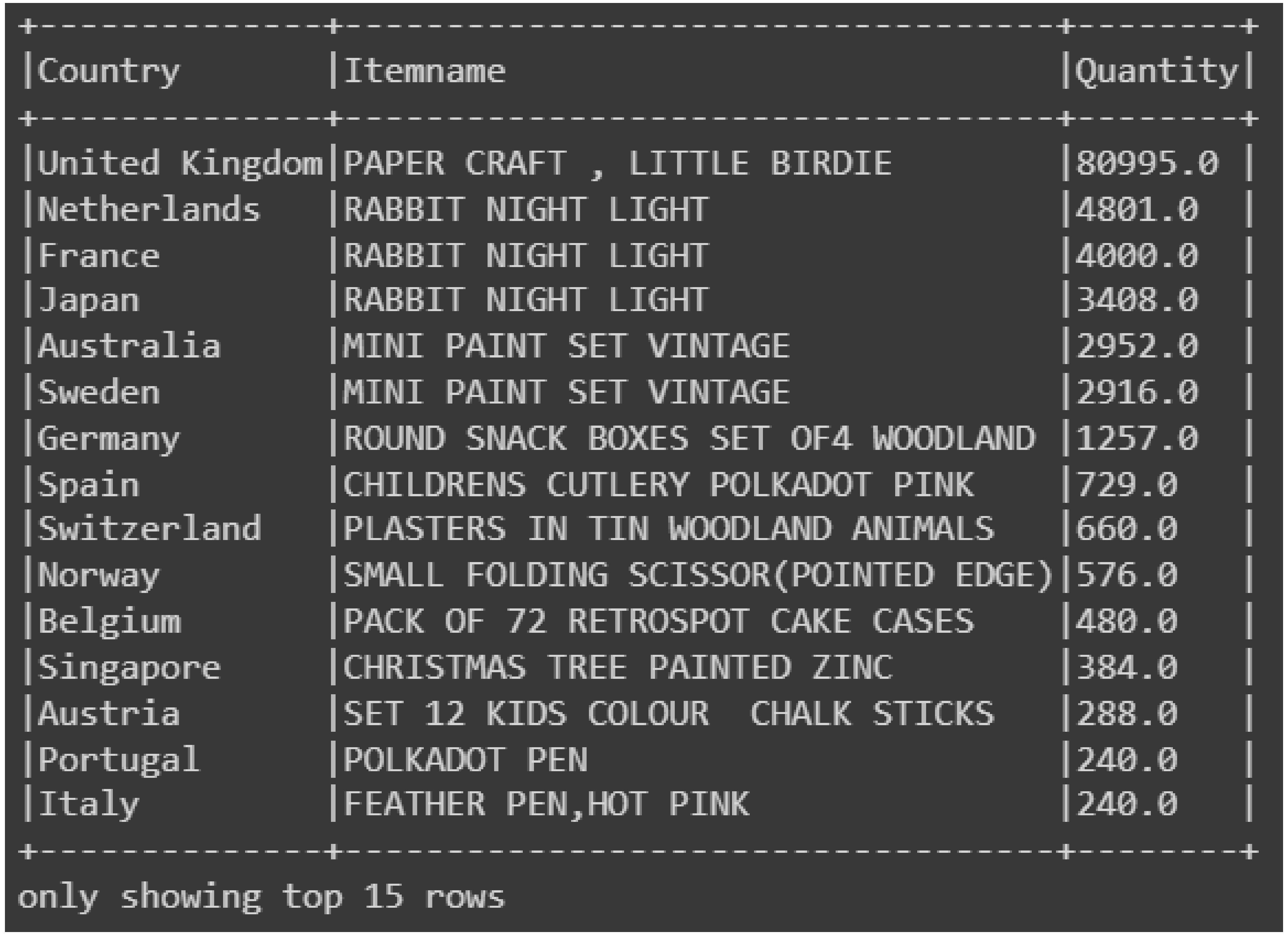
Data preprocessing is an important step in the data analysis pipeline that involves transforming raw data into a format that can be used for analysis. This is done in order to improve data quality and model processing.

We performed the following steps to make our dataset suitable for modelling:

1. Removing and filling null values as required
2. Removing unnecessary rows or noise in the data
3. Formatting certain columns in the required format

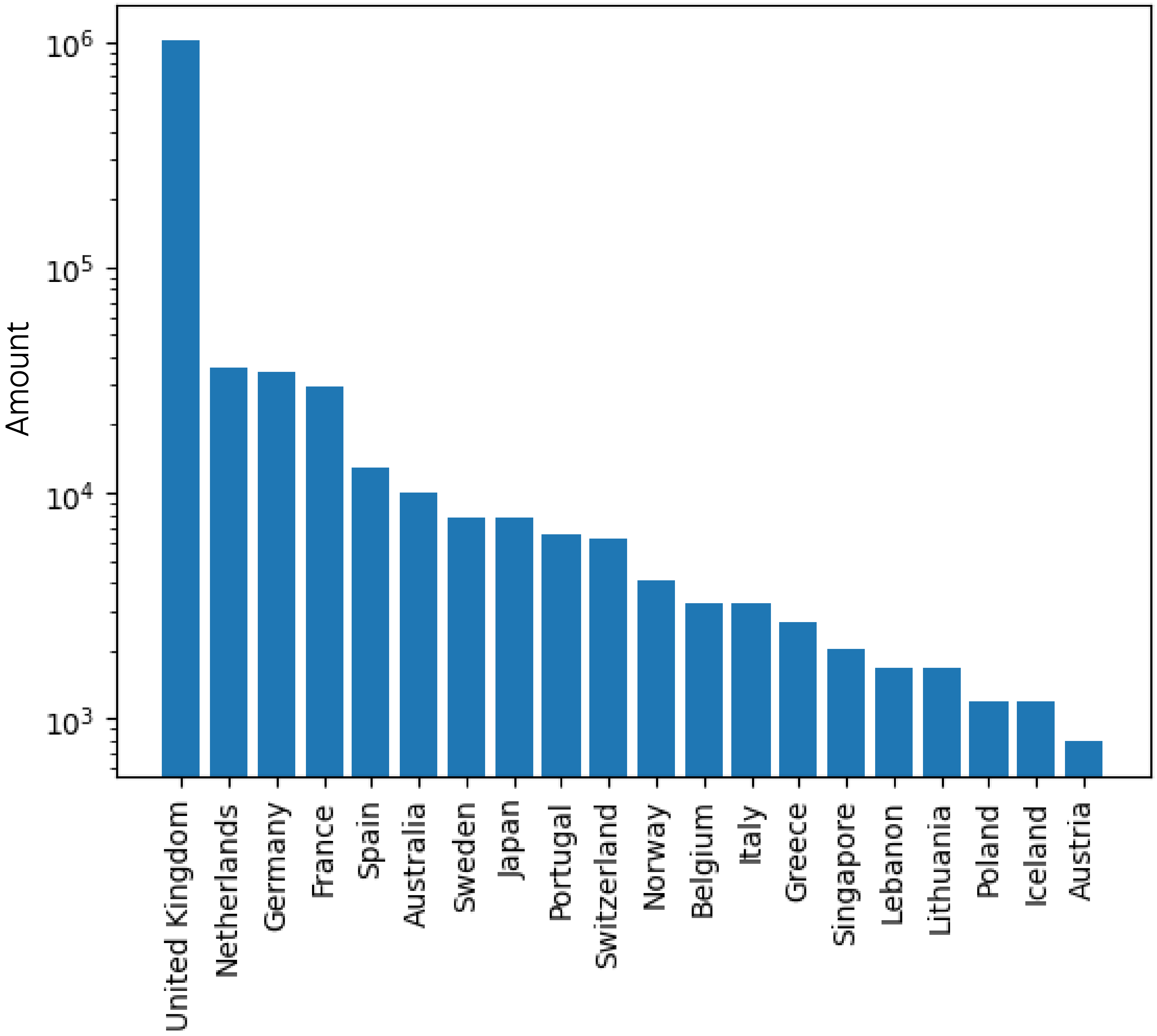
Exploratory Data Analysis

Best Selling Items in a Country

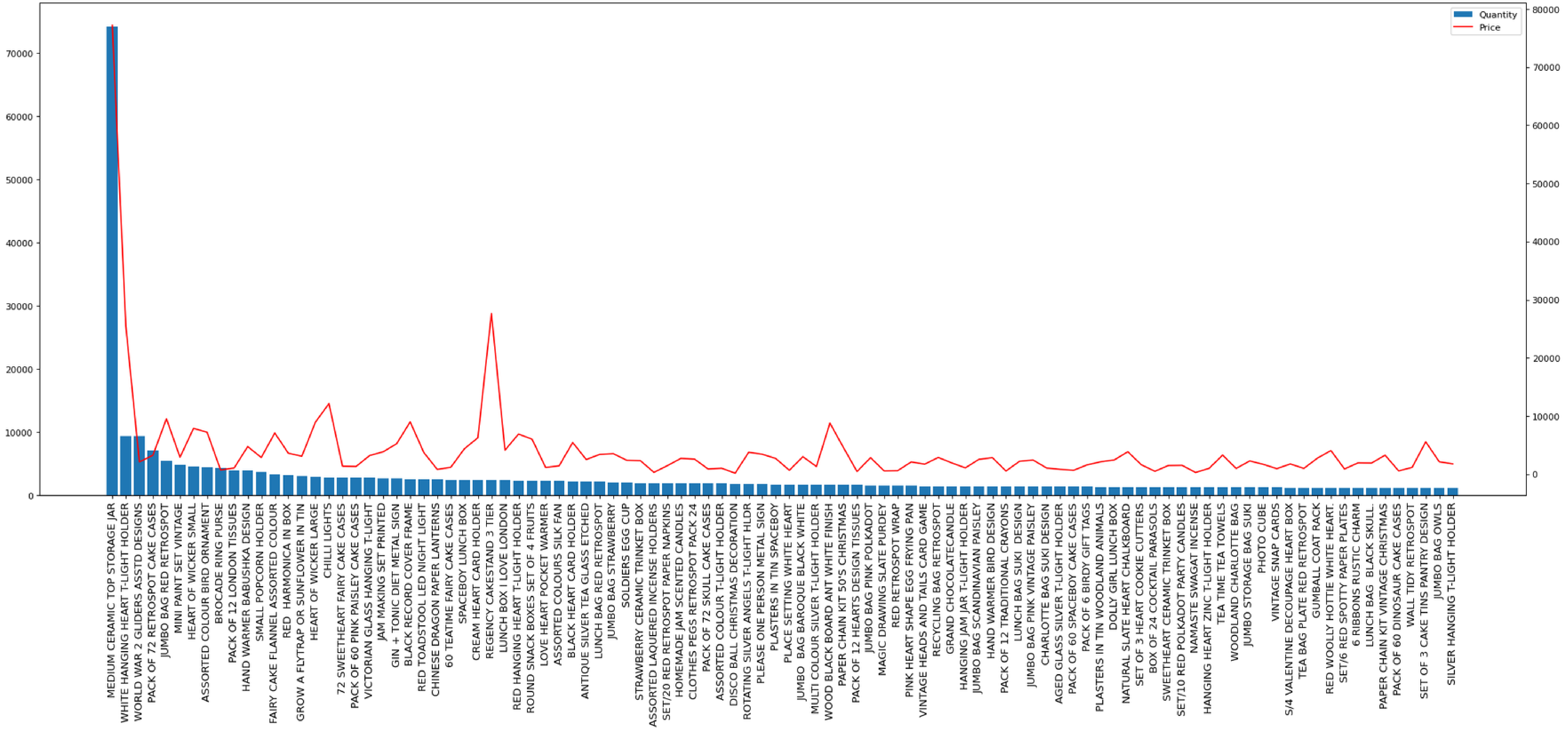


```
+--------------+-----------------------------------+--------+
|Country       |Itemname                           |Quantity|
+--------------+-----------------------------------+--------+
|United Kingdom|PAPER CRAFT , LITTLE BIRDIE        |80995.0 |
|Netherlands   |RABBIT NIGHT LIGHT                 |4801.0  |
|France        |RABBIT NIGHT LIGHT                 |4000.0  |
|Japan         |RABBIT NIGHT LIGHT                 |3408.0  |
|Australia     |MINI PAINT SET VINTAGE             |2952.0  |
|Sweden        |MINI PAINT SET VINTAGE             |2916.0  |
|Germany       |ROUND SNACK BOXES SET OF4 WOODLAND |1257.0  |
|Spain         |CHILDRENS CUTLERY POLKADOT PINK    |729.0   |
|Switzerland   |PLASTERS IN TIN WOODLAND ANIMALS   |660.0   |
|Norway        |SMALL FOLDING SCISSOR(POINTED EDGE)|576.0   |
|Belgium       |PACK OF 72 RETROSPOT CAKE CASES    |480.0   |
|Singapore     |CHRISTMAS TREE PAINTED ZINC        |384.0   |
|Austria       |SET 12 KIDS COLOUR  CHALK STICKS   |288.0   |
|Portugal      |POLKADOT PEN                       |240.0   |
|Italy         |FEATHER PEN,HOT PINK               |240.0   |
+--------------+-----------------------------------+--------+
only showing top 15 rows
```

Total Amount Earned grouped by Country

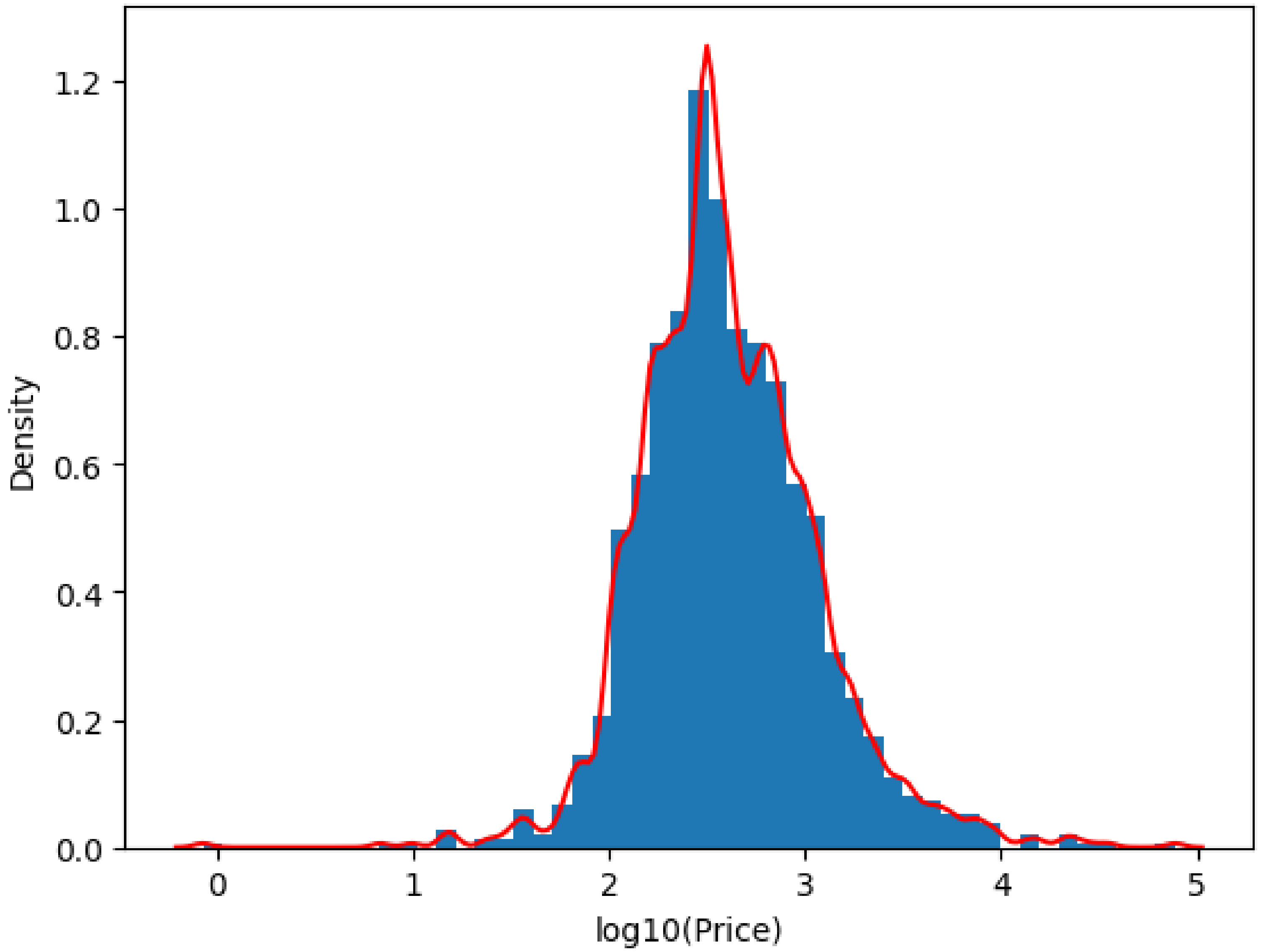


Quantity vs Total Price analysis



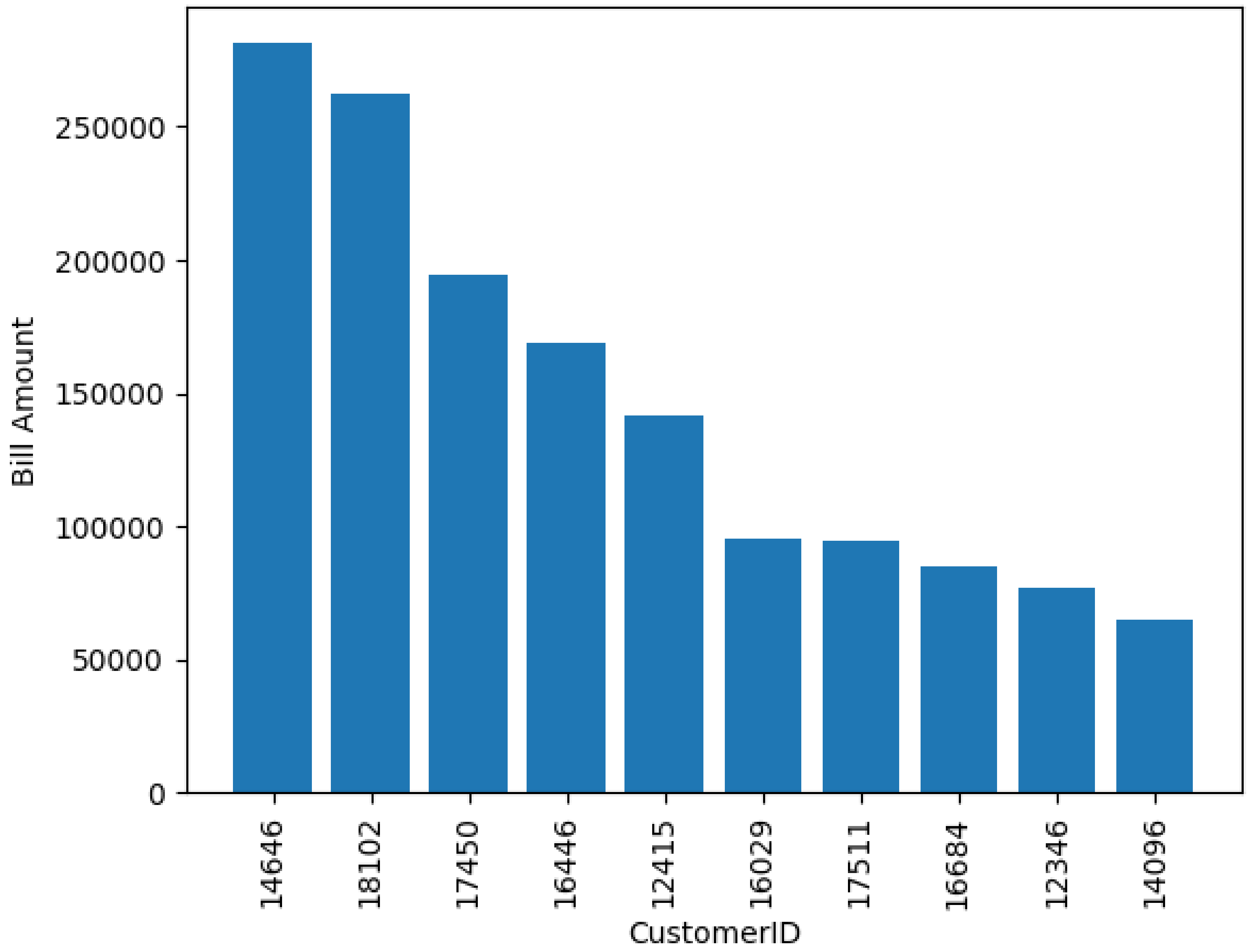
The bars in the above chart represent the quantities sold for each item. The line chart depicts the total gain from that particular item. We can see that there are some expensive items which are sold in large quantities.

Distribution of Expenditure per Customer



From the above chart, we can interpret that the distribution of expenditure by each customer is normal with average expenditure around 315 units. (Log scale is used in the graph.)

Top 10 Bill Amounts



The highest bill value is around 28000 for Customer ID 14646.

Frequent Itemset and Association Rule Generation

There are several algorithms available for frequent itemset and association rule generation in market basket analysis. Some of the most commonly used algorithms are:

Apriori Algorithm

The Apriori Algorithm is one of the most widely used algorithms for frequent itemset and association rule generation. It works by generating a set of frequent itemsets by finding all the sets of items that occur together in transactions with a frequency greater than or equal to a minimum support threshold. The algorithm then generates association rules from the frequent itemsets using metrics such as support, confidence, and lift.

FP-Growth Algorithm

The FP-Growth Algorithm is another popular algorithm for frequent itemset and association rule generation. It builds a frequent pattern tree (FP-tree) from the transactional data and generates a set of frequent itemsets. The association rules are then generated using support and confidence.

PCY (Park-Chen-Yu) Algorithm

The PCY (Park-Chen-Yu) algorithm is a modification of the Apriori algorithm that improves its efficiency by reducing the number of candidate itemsets that need to be generated and stored. The PCY algorithm works by first counting the frequency of individual items in the dataset and storing this information in a hash table. It then uses this hash table to identify pairs of items that occur together frequently, without generating all possible pairs.

FP-Growth Algorithm Implementation

The FP-Growth (Frequent Pattern Growth) algorithm works by constructing an FP-tree, a compact representation of the transactional data, and then generating frequent itemsets from the FP-tree. The steps involved are:

1. Constructing the FP-tree
2. Determining the frequent items by scanning the dataset
3. Generating frequent itemsets by recursively traversing the FP-tree
4. Generating association rules

The FP-Growth algorithm is an efficient algorithm for frequent itemset and association rule generation because it avoids the generation of candidate itemsets.

PCY Algorithm Implementation

The PCY algorithm is an efficient algorithm for frequent itemset and association rule generation because it reduces the number of candidate itemsets that need to be generated and stored. PCY algorithm follows the steps given below:

1. Count individual items and store the value in hash tables
2. Identify frequent item pairs using the hash table
3. Generate candidate itemsets by joining pairs of frequent items that occur together frequently
4. Count the frequency of candidate itemsets
5. Generate association rules using support and confidence