**Market Basket Analysis:**

**TEAM MEMBER: DHIVYA.N(922121106010)**

**PHASE 1 DOCUMENT SUBMISSION:**

**PROJECT: MARKET BASKET ANALYSIS**

****

**OBJECTIVE:**

Market basket analysis with Apriori algorithm

The retailer wants to target customers with suggestions on itemset that a customer is most likely to purchase .I was given dataset contains data of a retailer; the transaction data provides data around all the transactions that have happened over a period of time. Retailer will use result to grove in his industry and provide for customer suggestions on itemset, we be able increase customer engagement and improve customer experience and identify customer behavior. I will solve this problem with use Association Rules type of unsupervised learning technique that checks for the dependency of one data item on another data item.

**DATA SOURCE: :**[**https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis**](https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis)

**Data Pre-processing**

Next, we need to upload Assignment-1\_Data. xlsx to R to read the dataset.Now we can see our data in R.After we will clear our data frame, will remove missing values.To apply Association Rule mining, we need to convert dataframe into transaction data to make all items that are bought together in one invoice will be in one row. Below lines of code will combine all products from one BillNo and Date and combine all products from that BillNo and Date as one row, with each item, separated by (,)

We don’t need BillNo and Date, we will make it as Null.  
Next, you have to store this transaction data into .csv

This how should look transaction data before we will go to next step.

At this step we already have our transaction dataset, and it shows the matrix of items which bought together. We can’t see here any rules and how often it was purchase together. Now let’s check how many transactions we have and what they are. We will have to have to load this transaction data into an object of the transaction class. This is done by using the R function read.transactions of the arules package. Our format of Data frame is basket.Let’s have a view our transaction object by summary(transaction)We can see 18193 transactions (rows) and 7698 items (columns). 7698 is the product descriptions and 18193 transactions are collections of these items.

The summary gives us some useful information:

* Density tells the percentage of non-zero cells in a sparse matrix. In other words, total number of items that are purchased divided by a possible number of items in that matrix. You can calculate how many items were purchased by using density: 18193x7698x0.002291294=337445
* Summary will show us most frequent items.
* Element (itemset/transaction) length distribution: It will gave us how many transactions are there for 1-itemset, 2-itemset and so on. The first row is telling you a number of items and the second row is telling you the number of transactions.  
  For example, there is only 1546 transaction for one item, 860 transactions for 2 items, and there are 419 items in one transaction which is the longest.

Let’s check item frequency plot, we will generate an itemFrequencyPlot to create an item Frequency Bar Plot to view the distribution of objects based on itemMatrix (e.g., >transactions or items in >itemsets and >rules) which is our case.In itemFrequencyPlot(transaction,topN=20,type="absolute") first argument - our transaction object to be plotted that is tr. topN is allows us to plot top N highest frequency items. type can be as type="absolute" or type="relative". If we will chouse absolute it will plot numeric frequencies of each item independently. If relative it will plot how many times these items have appeared as compared to others. As well I made it in colure for better visualization.

**FEATURE SELECTION:**

Association Rule is most used when you are planning to build association in different objects in a set. It works when you are planning to find frequent patterns in a transaction database. It can tell you what items do customers frequently buy together and it allows retailer to identify relationships between the items.

* **Dataset Description**
* File name: Assignment-1\_Data
* List name: retaildata
* File format: . xlsx
* Number of Row: 522065
* Number of Attributes: 7
  + BillNo: 6-digit number assigned to each transaction. Nominal.
  + Itemname: Product name. Nominal.
  + Quantity: The quantities of each product per transaction. Numeric.
  + Date: The day and time when each transaction was generated. Numeric.
  + Price: Product price. Numeric.
  + CustomerID: 5-digit number assigned to each customer. Nominal.
  + Country: Name of the country where each customer resides. Nominal.

**MODEL SELECTION:**

We have thousands of rules generated based on data, we will need a couple of ways to present our findings. We will use ItemFrequencyPlot to visualize association rules.

**Scatter-Plot:**

A straight-forward visualization of association rules is to use a scatter plot using plot() of the arulesViz package. It uses Support and Confidence on the axes. In addition, third measure Liftis used by default to color (grey levels) of the points.

**Interactive Scatter-Plot:**

We can have a look for each rule (interactively) and view all quality measures (support, confidence and lift).

**Graph - Based Visualization and Group Method:**

Graph plots are a great way to visualize rules but tend to become congested as the number of rules increases. So, it is better to visualize a smaller number of rules with graph-based visualizations. We can see as well group method for top 10 items.

## 1. | Loading and Cleaning data

### 1-1. | Loading data

Out[2]:

|  | BillNo | Itemname | Quantity | Date | Price | CustomerID | Country |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 536365 | WHITE HANGING HEART T-LIGHT HOLDER | 6 | 01.12.2010 08:26 | 2,55 | 17850.0 | United Kingdom |
| 1 | 536365 | WHITE METAL LANTERN | 6 | 01.12.2010 08:26 | 3,39 | 17850.0 | United Kingdom |
| 2 | 536365 | CREAM CUPID HEARTS COAT HANGER | 8 | 01.12.2010 08:26 | 2,75 | 17850.0 | United Kingdom |
| 3 | 536365 | KNITTED UNION FLAG HOT WATER BOTTLE | 6 | 01.12.2010 08:26 | 3,39 | 17850.0 | United Kingdom |
| 4 | 536365 | RED WOOLLY HOTTIE WHITE HEART. | 6 | 01.12.2010 08:26 | 3,39 | 17850.0 | United Kingdom |

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

**RangeIndex: 522064 entries, 0 to 522063**

**Data columns (total 7 columns):**

**# Column Non-Null Count Dtype**

**--- ------ -------------- -----**

**0 BillNo 522064 non-null object**

**1 Itemname 520609 non-null object**

**2 Quantity 522064 non-null int64**

**3 Date 522064 non-null object**

**4 Price 522064 non-null object**

**5 CustomerID 388023 non-null float64**

**6 Country 522064 non-null object**

**dtypes: float64(1), int64(1), object(5)**

**memory usage: 27.9+ MB**

Out[4]:

**BillNo 0**

**Itemname 1455**

**Quantity 0**

**Date 0**

**Price 0**

**CustomerID 134041**

**Country 0**

**dtype: int64**

### 1-2. | Dropping data with negative or zero quantity

In [6]:

**df**=**df**.**loc[df['Quantity']**>**0]**

### 1-3. | Dropping data with zero price

In [8]:

**df**=**df**.**loc[df['Price']**>**'0']**

### 1-4. | Dropping Non-product data.

In [10]:

**df**=**df**.**loc[(df['Itemname']**!=**'POSTAGE')**&**(df['Itemname']**!=**'DOTCOM POSTAGE')**&**(df['Itemname']**!=**'Adjust bad debt')**&**(df['Itemname']**!=**'Manual')]**

### 1-5. | Filling null data

In [12]:

**df**=**df**.**fillna('-')**

**df**.**isnull()**.**sum()**

Out[12]:

**BillNo 0**

**Itemname 0**

**Quantity 0**

**Date 0**

**Price 0**

**CustomerID 0**

**Country 0**

**dtype: int64**

### 1-6. | Splitting data into year and month

In [13]:

**df['Year']**=**df['Date']**.**apply(**lambda **x:x**.**split('.')[2])**

**df['Year']**=**df['Year']**.**apply(**lambda **x:x**.**split(' ')[0])**

**df['Month']**=**df['Date']**.**apply(**lambda **x:x**.**split('.')[1])**

**df**.**head()**

Out[13]:

### 1-7. | Creating a Total price column

In [14]:

**df['Price']**=**df['Price']**.**str**.**replace(',','.')**.**astype('float64')**

**df['Total price']**=**df**.**Quantity**\***df**.**Price**

**df**.**head()**

Out[14]:

### 1-8. | Checking the Total price in each month.

In [15]:

**df**.**groupby(['Year','Month'])['Total price']**.**sum()**

Out[15]:

**Year Month**

**2010 12 778386.780**

**2011 01 648311.120**

**02 490058.230**

**03 659979.660**

**04 507366.971**

**05 721789.800**

**06 710158.020**

**07 642528.481**

**08 701411.420**

**09 981408.102**

**10 1072317.070**

**11 1421055.630**

**12 606953.650**

**Name: Total price, dtype: float64**

**It is appropriate to look at 12-month increments to implement data analytics properly, so I'll drop the data for 2020 Dec.**

In [16]:

**df**=**df**.**loc[df['Year']**!=**'2010']**

| Category

### Top 10 highest sales amount items

Out[20]:

|  | Itemname | Price |
| --- | --- | --- |
| 0 | REGENCY CAKESTAND 3 TIER | 24653.67 |
| 1 | PARTY BUNTING | 9416.13 |
| 2 | SET OF 3 CAKE TINS PANTRY DESIGN | 7621.05 |
| 3 | CREAM SWEETHEART MINI CHEST | 6836.38 |
| 4 | SET/4 WHITE RETRO STORAGE CUBES | 6714.75 |
| 5 | ENAMEL BREAD BIN CREAM | 6585.93 |
| 6 | WHITE HANGING HEART T-LIGHT HOLDER | 6563.80 |
| 7 | DOORMAT KEEP CALM AND COME IN | 6385.09 |
| 8 | SPOTTY BUNTING | 6262.40 |
| 9 | RED RETROSPOT CAKE STAND | 6035.29 |

### Top 10 most purchased items

Out[21]:

|  | Itemname | Quantity |
| --- | --- | --- |
| 520583 | PAPER CRAFT , LITTLE BIRDIE | 80995 |
| 59999 | MEDIUM CERAMIC TOP STORAGE JAR | 74215 |
| 405138 | WORLD WAR 2 GLIDERS ASSTD DESIGNS | 4800 |
| 198929 | SMALL POPCORN HOLDER | 4300 |
| 94245 | EMPIRE DESIGN ROSETTE | 3906 |
| 260928 | ESSENTIAL BALM 3.5g TIN IN ENVELOPE | 3186 |
| 51228 | FAIRY CAKE FLANNEL ASSORTED COLOUR | 3114 |
| 154834 | FAIRY CAKE FLANNEL ASSORTED COLOUR | 3114 |
| 416997 | SMALL CHINESE STYLE SCISSOR | 3000 |
| 280572 | ASSORTED COLOUR BIRD ORNAMENT | 2880 |

### Top 10 most frequently purchased items

**EVALUVATION**:

Based on the results of these calculations can be used as a recommendation for retail owners to arrange the arrangement of product catalogs and take strategic steps to improve product marketing.. By utilizing the association rules which are discovered as a result of the analyses, the retailer can apply effective marketing and sales promotion strategies, he will be able increase customer engagement and improve customer experience and identify customer behavior.