MARKET BASKET INSIGHTS

MEMBER: J.FARHATH NASEEM (922121106014)

PHASE 3 SUBMISSION DOCUMENT: DEVELOPMENT PART 1

PROJECT: Market basket insights

Phase 3: Development Part 1

In this part we will begin building your project by loading and preprocessing the dataset. We start the market basket insights project by loading and preprocessing the transaction data.Load the transaction dataset and preprocess the data for association analysis.

Dataset Link: https://www.kaggle.com/datasets/aslanahmedov/market basket-analysis

About Dataset

Market Basket Analysis

Introduction

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.

An Example of Association Rules

Assume there are 100 customers, 10 of them bought Computer Mouth, 9 bought Mat for Mouse and 8 bought both of them.

- bought Computer Mouth => bought Mat for Mouse
- support = P(Mouth & Mat) = 8/100 = 0.08
- confidence = support/P(Mat for Mouse) = 0.08/0.09 = 0.89
- lift = confidence/P(Computer Mouth) = 0.89/0.10 = 8.9

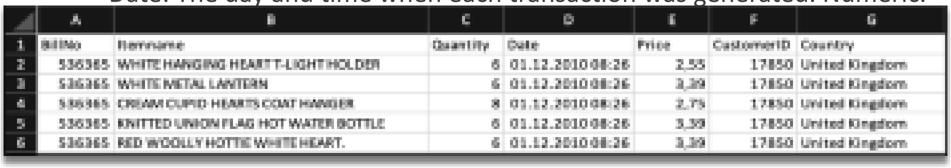
This just simple example. In practice, a rule needs the support of several hundred transactions, before it can be considered statistically significant, and datasets often contain thousands or millions of transactions.

Strategy

- Data Import
- Data Understanding and Exploration
- Transformation of the data – so that is ready to be consumed by the association rules algorithm • Running association rules
- Exploring the rules generated
- Filtering the generated rules
- Visualization of Rule

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.



	A	B	C	D	E	F	G
1	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
2	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	01.12.2010 08:26	2,55	17850	United Kingdom
3	536365	WHITE METAL LANTERN	6	01.12.2010 08:26	3,39	17850	United Kingdom
4	536365	CREAM CUPID HEARTS COAT HANGER	8	01.12.2010 08:26	2,75	17850	United Kingdom
5	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	01.12.2010 08:26	3,39	17850	United Kingdom
6	536365	RED WOOLLY HOTTIE WHITE HEART.	6	01.12.2010 08:26	3,39	17850	United Kingdom

- Price: Product price. Numeric.
- CustomerID: 5-digit number assigned to each customer. Nominal.
- Country: Name of the country where each customer resides. Nominal.

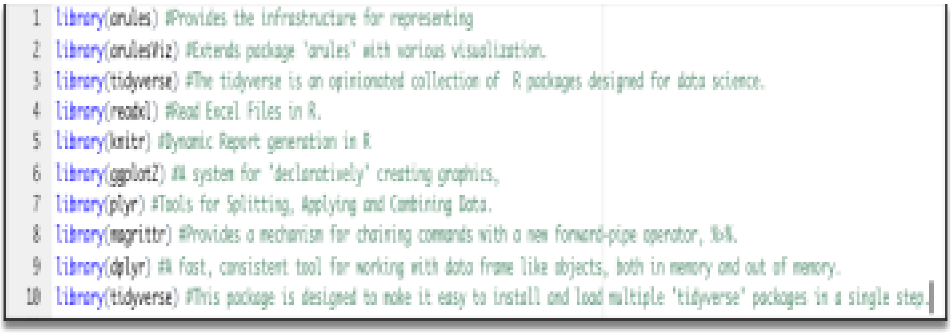
Libraries in R

First, we need to load required libraries. Shortly I describe all libraries.

- arules - Provides the infrastructure for representing, manipulating and analyzing transaction data and patterns (frequent itemsets and association rules).
- arulesViz - Extends package 'arules' with various visualization. techniques for association rules and item-sets. The package also includes several interactive visualizations for rule exploration.
- tidyverse - The tidyverse is an opinionated collection of R packages designed for data science. •

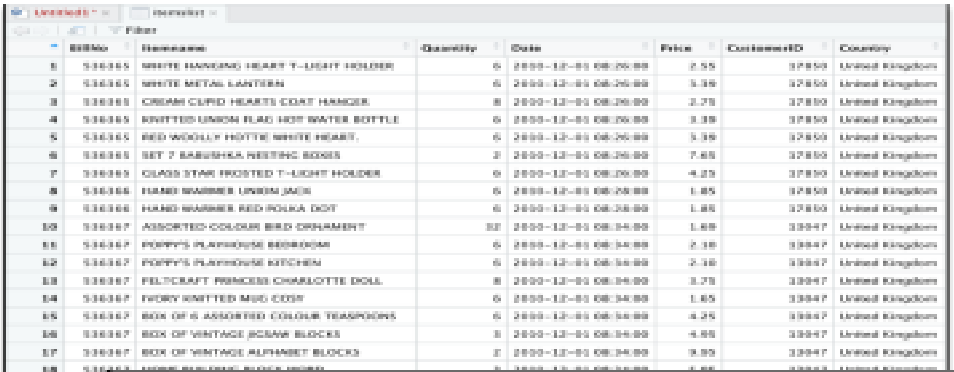
readxl - Read Excel Files in R.

- plyr - Tools for Splitting, Applying and Combining Data.
- ggplot2 - A system for 'declaratively' creating graphics, based on "The Grammar of Graphics". You provide the data, tell 'ggplot2' how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.
- knitr - Dynamic Report generation in R.



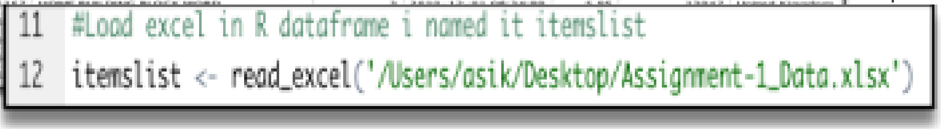
```
library(arules) #Provides the infrastructure for representing
library(arulesViz) #Extends package 'arules' with various visualization.
library(tidyverse) #The tidyverse is an opinionated collection of  R packages designed for data science.
library(readxl) #Read Excel Files in R.
library(knitr) #Dynamic Report generation in R
library(ggplot2) #A system for 'declaratively' creating graphics,
library(plyr) #Tools for Splitting, Applying and Combining Data.
library(magrittr) #Provides a mechanism for chaining commands with a new forward-pipe operator, %>%.
library(dplyr) #A fast, consistent tool for working with data frame like objects, both in memory and out of memory.
library(tidyverse) #This package is designed to make it easy to install and load multiple 'tidyverse' packages in a single step.
```

- magrittr- Provides a mechanism for chaining commands with a new forward-pipe operator, %>%. This operator will forward a value, or the result of an expression, into the next function call/expression. There is flexible support for the type of right-hand side expressions.



	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
1	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850	United Kingdom
2	536365	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850	United Kingdom
3	536365	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850	United Kingdom
4	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850	United Kingdom
5	536365	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850	United Kingdom
6	536365	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850	United Kingdom
7	536365	GLASS STAR FROSTED T-LIGHT HOLDER	6	2010-12-01 08:26:00	4.25	17850	United Kingdom
8	536366	HAND WARMER UNION JACK	6	2010-12-01 08:28:00	1.85	17850	United Kingdom
9	536366	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850	United Kingdom
10	536367	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047	United Kingdom
11	536367	POPPY'S PLAYHOUSE BEDROOM	6	2010-12-01 08:34:00	2.10	13047	United Kingdom
12	536367	POPPY'S PLAYHOUSE KITCHEN	6	2010-12-01 08:34:00	2.10	13047	United Kingdom
13	536367	FELTCRAFT PRINCESS CHARLOTTE DOLL	8	2010-12-01 08:34:00	3.75	13047	United Kingdom
14	536367	IVORY KNITTED MUG COSY	6	2010-12-01 08:34:00	1.65	13047	United Kingdom
15	536367	BOX OF 6 ASSORTED COLOUR TEASPOONS	6	2010-12-01 08:34:00	4.25	13047	United Kingdom
16	536367	BOX OF VINTAGE JIGSAW BLOCKS	3	2010-12-01 08:34:00	4.95	13047	United Kingdom
17	536367	BOX OF VINTAGE ALPHABET BLOCKS	2	2010-12-01 08:34:00	9.95	13047	United Kingdom
18	[illegible]	HOME BUILDING BLOCK WORD	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
19	[illegible]						
20	[illegible]						
21	[illegible]						
22	[illegible]						

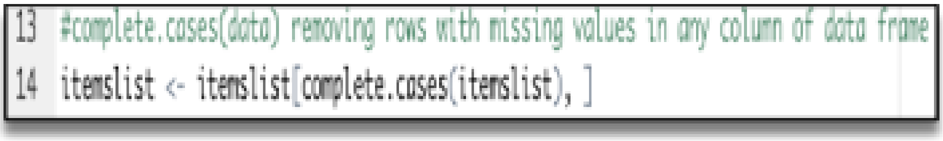
- dplyr - A fast, consistent tool for working with data frame like objects, both in memory and out of memory.
- tidyverse - This package is designed to make it easy to install and load multiple 'tidyverse' packages in a single step.



```
#Load excel in R dataframe i named it itemslist
itemslist <- read_excel('/Users/asik/Desktop/Assignment-1_Data.xlsx')
```

Data Pre-processing

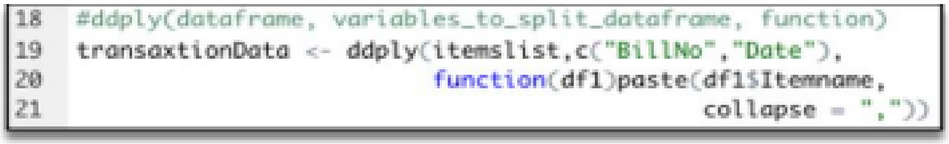
Next, we need to upload Assignment-1_Data. xlsx to R to read the dataset.Now we can see our data in R.



```
#complete.cases(data) removing rows with missing values in any column of data frame
itemslist <- itemslist[complete.cases(itemslist), ]
```

After we will clear our data frame, will remove missing values.

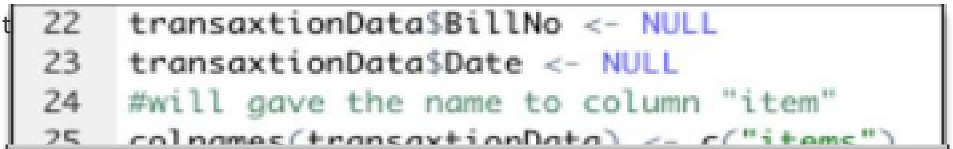
To apply Association Rule mining, we need to convert dataframe into transaction data to make all items that are bought together in one invoice will be in one row. Below lines of code will combine all



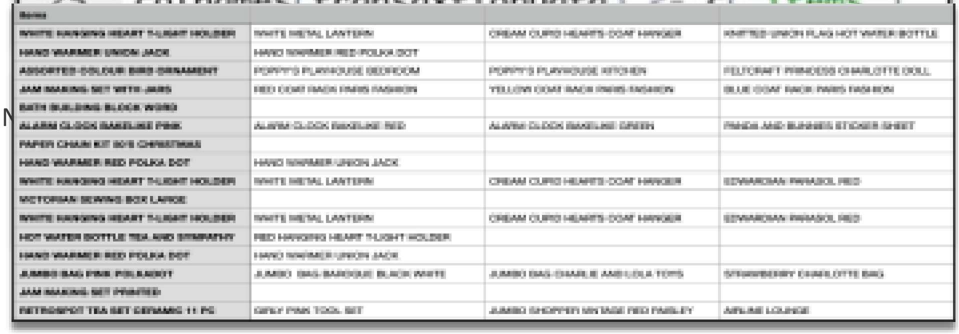
```
#ddply(dataframe, variables_to_split_dataframe, function)
transactionData <- ddply(itemslist,c("BillNo","Date"),
                         function(df1)paste(df1$Itemname,
                                            collapse = ","))
```

products from one BillNo and Date and combine all products from that BillNo and Date as one row, with each

it



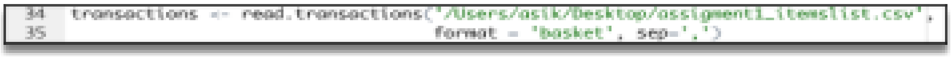
```
transaxtionData$BillNo <- NULL
transaxtionData$Date <- NULL
#will gave the name to column "item"
colnames(transaxtionData) <- c("items")
```



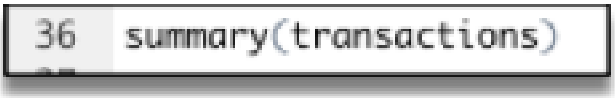
items			
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	KNITTED UNION FLAG HOT WATER BOTTLE
HAND WARMER UNION JACK	HAND WARMER RED POLKA DOT		
ASSORTED COLOUR BIRD ORNAMENT	POPPY'S PLAYHOUSE BEDROOM	POPPY'S PLAYHOUSE KITCHEN	FELTCRAFT PRINCESS CHARLOTTE DOLL
JAM MAKING SET WITH JARS	RED COAT RACK PARIS FASHION	YELLOW COAT RACK PARIS FASHION	BLUE COAT RACK PARIS FASHION
BATH BUILDING BLOCK WORD			
ALARM CLOCK BAKELIKE PINK	ALARM CLOCK BAKELIKE RED	ALARM CLOCK BAKELIKE GREEN	PANDA AND BUNNIES STICKER SHEET
PAPER CHAIN KIT 50'S CHRISTMAS			
HAND WARMER RED POLKA DOT	HAND WARMER UNION JACK		
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	EDWARDIAN PARASOL RED
VICTORIAN SEWING BOX LARGE			
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	EDWARDIAN PARASOL RED
HOT WATER BOTTLE TEA AND SYMPATHY	RED HANGING HEART T-LIGHT HOLDER		
HAND WARMER RED POLKA DOT	HAND WARMER UNION JACK		
JUMBO BAG PINK POLKADOT	JUMBO BAG BAROQUE BLACK WHITE	JUMBO BAG CHARLIE AND LOLA TOYS	STRAWBERRY CHARLOTTE BAG
JAM MAKING SET PRINTED			
RETROSPOT TEA SET CERAMIC 11 PC	GIRLY PINK TOOL SET	JUMBO SHOPPER VINTAGE RED PAISLEY	AIRLINE LOUNGE

We don’t need BillNo and Date, we will make it as Null.

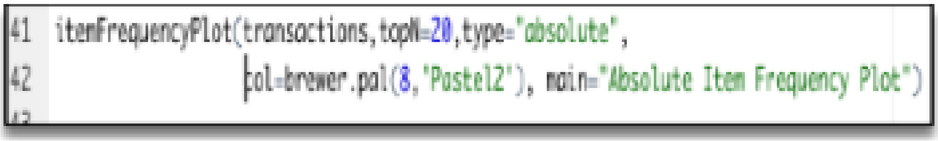
N



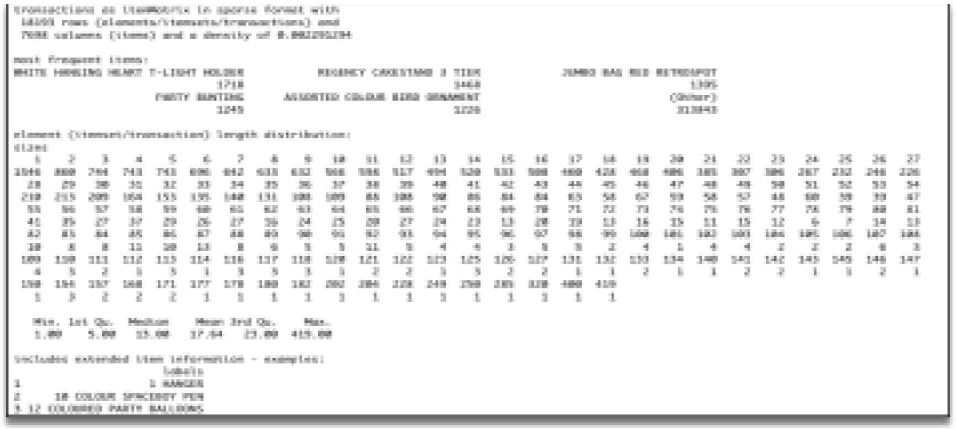
```
transactions <- read.transactions('/Users/asik/Desktop/assigment1_itemslist.csv',
                                  format = 'basket', sep=',')
```



```
summary(transactions)
```

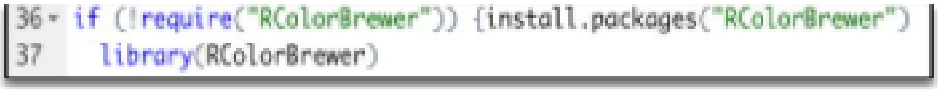


```
itemFrequencyPlot(transactions,topN=20,type="absolute",
                  col=brewer.pal(8,'Pastel2'), main="Absolute Item Frequency Plot")
```



```
transactions as itemMatrix in sparse format with
 18193 rows (elements/itemsets/transactions) and
 7698 columns (items) and a density of 0.002291294

most frequent items:
WHITE HANGING HEART T-LIGHT HOLDER           REGENCY CAKESTAND 3 TIER            JUMBO BAG RED RETROSPOT
                              1718                               1468                               1305
                     PARTY BUNTING      ASSORTED COLOUR BIRD ORNAMENT                            (Other)
                              1245                               1226                             313843

element (itemset/transaction) length distribution:
sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   25   26   27
1546  880  764  743  743  696  642  635  632  566  586  517  494  520  531  500  460  424  468  406  385  387  386  267  232  246  226
  28   29   30   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54
 218  215  209  164  153  135  140  131  108  109   84  108   90   86   84   84   65   56   67   59   58   57   48   60   39   39   47
  55   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80   81
  41   35   27   37   29   26   27   36   24   25   20   27   24   23   13   20   19   13   16   15   11   15   12    6    7   14   13
  82   83   84   85   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105  106  107  108
  10    8    8   11   10   13    8    6    5    5   11    5    4    4    3    5    5    2    4    1    4    4    2    2    2    6    3
 109  110  111  112  113  114  116  117  118  120  121  122  123  125  126  127  131  132  133  134  140  141  142  143  145  146  147
   4    5    2    1    5    1    3    5    3    1    2    2    1    5    2    2    1    1    2    1    1    2    2    1    1    2    1
 150  154  157  168  171  177  178  180  182  202  204  228  249  250  285  320  400  419
   1    3    2    2    2    1    1    1    1    1    1    1    1    1    1    1    1    1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    5.00   13.00   17.64   23.00  419.00

includes extended item information - examples:
                labels
1             1 HANGER
2    10 COLOUR SPACEBOY PEN
3 12 COLOURED PARTY BALLOONS
```

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



```
if (!require("RColorBrewer")) {install.packages("RColorBrewer")
  library(RColorBrewer)
```

The summary gives us some useful information:

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.

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

import warnings
warnings.filterwarnings('ignore')

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
 for filename in filenames:
 print(os.path.join(dirname, filename))
```

```
/kaggle/input/market-basket-analysis/Assignment-1_Data.xls
x
/kaggle/input/market-basket-analysis/Assignment-1_Data.csv
```

[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]:

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country	Year	Month

0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	01.12.2010 08:26	2,55	17850.0	United Kingdom	2010	12
1	536365	WHITE METAL LANTERN	6	01.12.2010 08:26	3,39	17850.0	United Kingdom	2010	12
2	536365	CREAM CUPID HEARTS COAT HANGER	8	01.12.2010 08:26	2,75	17850.0	United Kingdom	2010	12
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	01.12.2010 08:26	3,39	17850.0	United Kingdom	2010	12
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6	01.12.2010 08:26	3,39	17850.0	United Kingdom	2010	12

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]:

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country	Year	Month	Total price
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	01.12.2010 08:26	2.55	17850.0	United Kingdom	2010	12	15.30
1	536365	WHITE METAL LANTERN	6	01.12.2010 08:26	3.39	17850.0	United Kingdom	2010	12	20.34

2	536365	CREAM CUPID HEARTS COAT HANGER	8	01.12.2010 08:26	2.75	17850.0	United Kingdom	2010	12	22.00
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	01.12.2010 08:26	3.39	17850.0	United Kingdom	2010	12	20.34
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6	01.12.2010 08:26	3.39	17850.0	United Kingdom	2010	12	20.34

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']
```

2. | Exploratoty Data Analysis

2-1. | Sales amount and quantity

2468101200.2M0.4M0.6M0.8M1M1.2M1.4M
CountryAustraliaBelgiumFranceGermanyGreeceHong KongIcelandIsraelItalyLebanonNetherlandsPolandPortugalSingaporeSpainSwedenSwitzerlandUnited KingdomAustriaJapanNorwaySaudi ArabiaUnited Arab EmiratesBrazilUSAUnspecifiedBahrainMaltaRSAMonthly sales amount in each country in 2021MonthSales amount

Most of the sales amounts are occupied by the UK.

PortugalItalyHong KongSingaporeAustriaIsraelPolandUnspecifiedGreeceIcelandUSA01M2M3M4M5M6M7M8 M Sales amount in each country in 2021CountrySales amount

2-2. | Category

Top 10 highest sales amount items

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

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

Out[30]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(ALARM CLOCK BAKELIKE GREEN)	(ALARM CLOCK BAKELIKE RED)	0.05	0.05	0.03	0.64	12.41	0.03	2.64	0.97
1	(ALARM CLOCK BAKELIKE RED)	(ALARM CLOCK BAKELIKE GREEN)	0.05	0.05	0.03	0.59	12.41	0.03	2.32	0.97
2	(GARDENERS KNEELING PAD KEEP CALM)	(GARDENERS KNEELING PAD CUP OF TEA)	0.05	0.05	0.03	0.60	13.23	0.03	2.40	0.98

3	(GARDENERS KNEELING PAD CUP OF TEA)	(GARDENERS KNEELING PAD KEEP CALM)	0.05	0.05	0.03	0.72	13.23	0.03	3.39	0.97
4	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.04	0.05	0.03	0.82	15.50	0.03	5.25	0.98

3-2. | The top 5 of the highest `support` value of items(antecedents)

Support(item) = Transactions comprising the item / Total transactions

```
Out[32]:
```

	antecedents	consequents	support
13	frozenset({'JUMBO BAG RED RETROSPOT'})	frozenset({'JUMBO BAG PINK POLKADOT'})	0.05
12	frozenset({'JUMBO BAG PINK POLKADOT'})	frozenset({'JUMBO BAG RED RETROSPOT'})	0.05
16	frozenset({'JUMBO STORAGE BAG SUKI'})	frozenset({'JUMBO BAG RED RETROSPOT'})	0.04
17	frozenset({'JUMBO BAG RED RETROSPOT'})	frozenset({'JUMBO STORAGE BAG SUKI'})	0.04
15	frozenset({'JUMBO SHOPPER VINTAGE RED PAISLEY'})	frozenset({'JUMBO BAG RED RETROSPOT'})	0.04

In the top support value of purchase, it means that "JUMBO BAG PINK RETROSPOT" is present in 5% of all purchases.

3-3. | The top 5 of the highest `confidence` value of items

Confidence = Transactions comprising antecedent and consequent / Transactions comprising antecedent

```
Out[33]:
```

	antecedents	consequents	confidence

4	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	0.82
30	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	0.78
6	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	0.75
7	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	0.73
3	frozenset({'GARDENERS KNEELING PAD CUP OF TEA'})	frozenset({'GARDENERS KNEELING PAD KEEP CALM'})	0.72

In the top confidence value of the purchase, it means that 82% of the customers who bought "PINK REGENCY TEACUP AND SAUCER" also bought "GREEN REGENCY TEACUP AND SAUCER".

3-4. | The top 5 of the highest `lift` value of items

Lift = Confidence (antecedent -> consequent) / Support(antecedent)

In [34]:

```
rules[['antecedents','consequents','lift']].sort_values('lift',ascending=False)[:5].style.background_gradient(cmap=cm).set_precision(2)
```

Out[34]:

	antecedents	consequents	lift
4	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	15.50
5	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	15.50
31	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	14.36
30	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	14.36
6	frozenset({'GREEN	frozenset({'ROSES	13.86

	REGENCY TEACUP AND SAUCER'})	REGENCY TEACUP AND SAUCER'})	

In the top list value of the purchase, it means that customers are 15.5 times more likely to buy "GREEN REGENCY TEACUP AND SAUCER" if you sell "PINK REGENCY TEACUP AND SAUCER".

3-5. | The best combination of the items

Out[35]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
4	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	0.04	0.05	0.03	0.82	15.50	0.03	5.25	0.98
30	frozenset({'PINK REGENCY TEACUP AND SAUCER'})	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	0.04	0.05	0.03	0.78	14.36	0.03	4.24	0.97
6	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	0.05	0.05	0.04	0.75	13.86	0.04	3.78	0.98
7	frozenset({'ROSES REGENCY TEACUP AND SAUCER'})	frozenset({'GREEN REGENCY TEACUP AND SAUCER'})	0.05	0.05	0.04	0.73	13.86	0.04	3.55	0.98

3	frozenset({'GARDENERS KNEELING PAD CUP OF TEA'})	frozenset({'GARDENERS KNEELING PAD KEEP CALM'})	0.05	0.05	0.03	0.72	13.23	0.03	3.39	0.97