Market Basket Analysis

INTRODUCTION

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.

In this phase the design to innovation and data flow of market business analysis is going to be done.

DATASET

The data is obtained from https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis

COLUMNS USED

From `Assignment-1_Data` data the following columns are used

- Bill no
- Item name
- Quantity
- Date
- Price
- Customer id
- Country

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.

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

Untitled1 * | itemslist

Filter

	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	536367	HOME BUILDING BLOCK WORD	3	2010-12-01 08:34:00	5.95	13047	United Kingdom
19	536367	LOVE BUILDING BLOCK WORD	3	2010-12-01 08:34:00	5.95	13047	United Kingdom
20	536367	RECIPE BOX WITH METAL HEART	4	2010-12-01 08:34:00	7.95	13047	United Kingdom
21	536367	DOORMAT NEW ENGLAND	4	2010-12-01 08:34:00	7.95	13047	United Kingdom
22	536368	JAM MAKING SET WITH JARS	6	2010-12-01 08:34:00	4.25	13047	United Kingdom

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.

- **LOAD THE TRANSACTION DATASET AND PREPROCESS THE DATA FOR ASSOCIATION ANALYSIS.**

retaildata (522k rows)

Detail Compact Column

7 of 7 columns

About this table

- File name: Assignment-1_Data
- List name: retaildata
- File format: . xlsx
- Number of Row: 522065

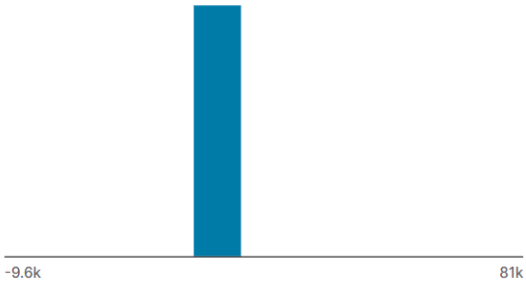
# BillNo	A Itemname	# Quantity	Date	# Price	# CustomerID	A Country
6-digit number assigned to each transaction. Nominal.	Product name. Nominal.	The quantities of each product per transaction. Numeric.	The day and time when each transaction was generated. Numeric.	Product price. Numeric.	5-digit number assigned to each customer. Nominal.	Name of the co where each cu resides. Nomin
536k 582k	4186 unique values	-9.6k 81k	1Dec10 9Dec11	-11.1k 13.5k	12.3k 18.3k	United Kingdo Germany Other (25400)
536365	WHITE HANGING HEART T-LIGHT HOLDER	6	12/01/2010 08:26:00	2.55	17850	United Kingc
536365	WHITE METAL LANTERN	6	12/01/2010 08:26:00	3.39	17850	United Kingc
536365	CREAM CUPID HEARTS COAT HANGER	8	12/01/2010 08:26:00	2.75	17850	United Kingc
536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/01/2010 08:26:00	3.39	17850	United Kingc
536365	RED WOOLLY HOTTIE	6	12/01/2010 08:26:00	3.39	17850	United Kingc

retaildata (522k rows)

Detail Compact Column 7 of 7 columns

Quantity

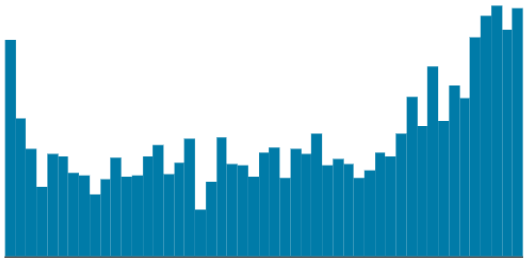
The quantities of each product per transaction. Numeric.



Valid	522k	100%
Mismatched	0	0%
Missing	0	0%
Mean	10.1	
Std. Deviation	161	
Quantiles	-9.6k	Min
	1	25%
	3	50%
	10	75%
	81k	Max

Date

The day and time when each transaction was generated. Numeric.



Valid	522k	100%
Mismatched	0	0%
Missing	0	0%
Minimum	1Dec10	
Mean	4Jul11	
Maximum	9Dec11	

retaildata (522k rows)

Detail Compact Column 7 of 7 columns

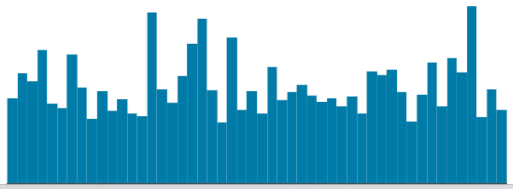
Price

Product price. Numeric.

Valid	522k	100%
Mismatched	0	0%
Missing	0	0%
Mean	3.83	
Std. Deviation	41.9	
Quantiles	-11.1k	Min
	1.25	25%
	2.08	50%
	4.13	75%
	13.5k	Max

CustomerID

5-digit number assigned to each customer. Nominal.



Valid	388k	74%
Mismatched	0	0%
Missing	134k	26%
Mean	15.3k	
Std. Deviation	1.72k	
Quantiles	12.3k	Min
	13.9k	25%
	15.3k	50%
	16.8k	75%

retaildata (522k rows)

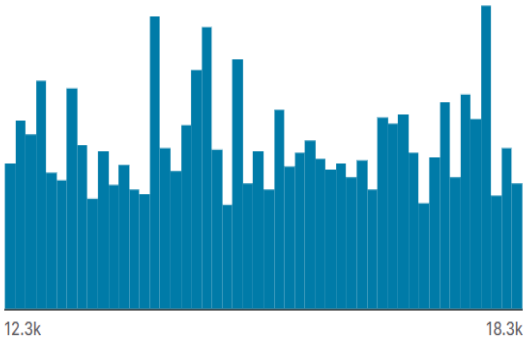
Detail Compact Column

7 of 7 columns

2.08	50%
4.13	75%
13.5k	Max

CustomerID

5-digit number assigned to each customer. Nominal.



Valid	388k	74%
Mismatched	0	0%
Missing	134k	26%
Mean	15.3k	
Std. Deviation	1.72k	
Quantiles	12.3k	Min
	13.9k	25%
	15.3k	50%
	16.8k	75%
	18.3k	Max

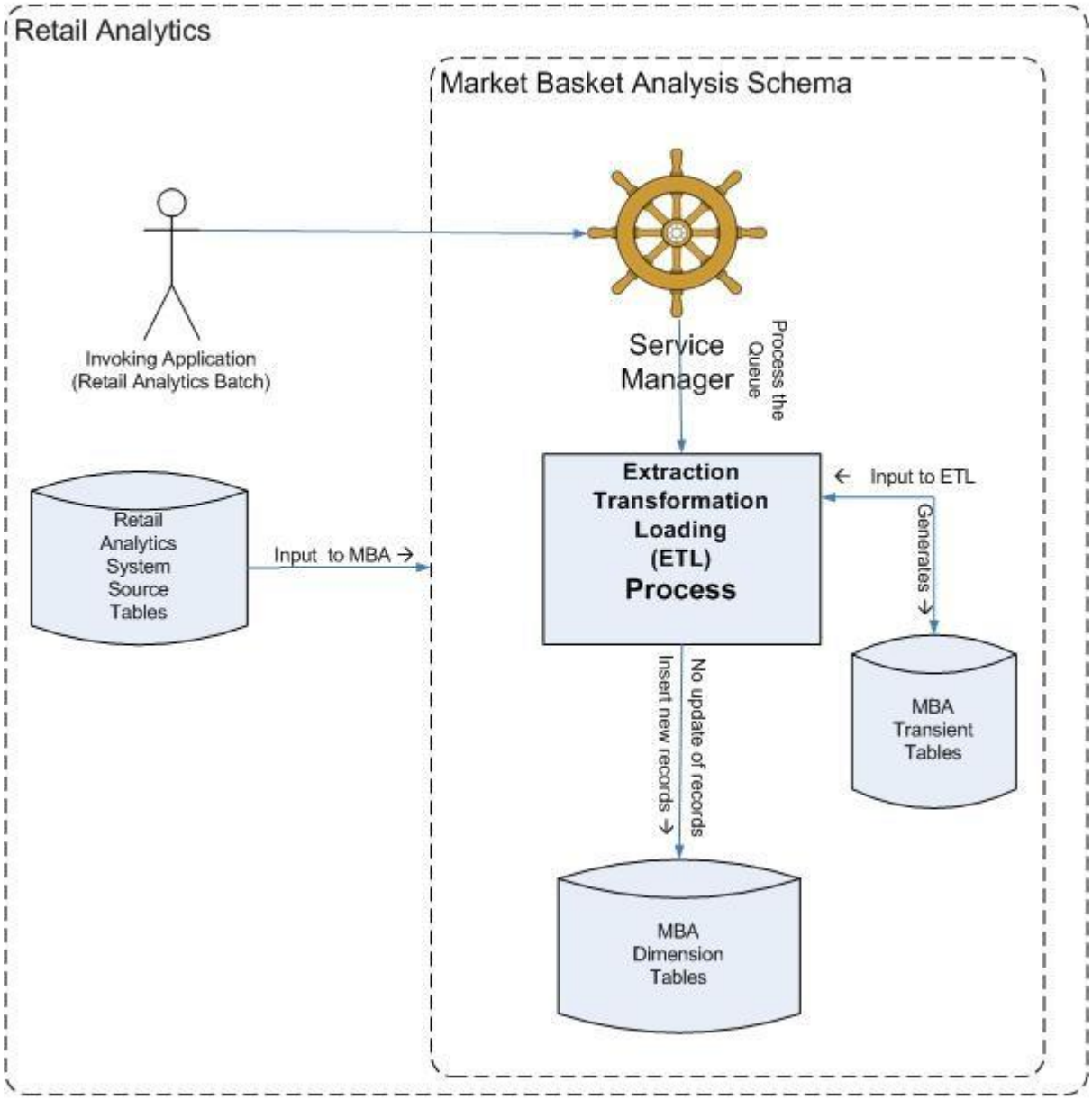
A Country

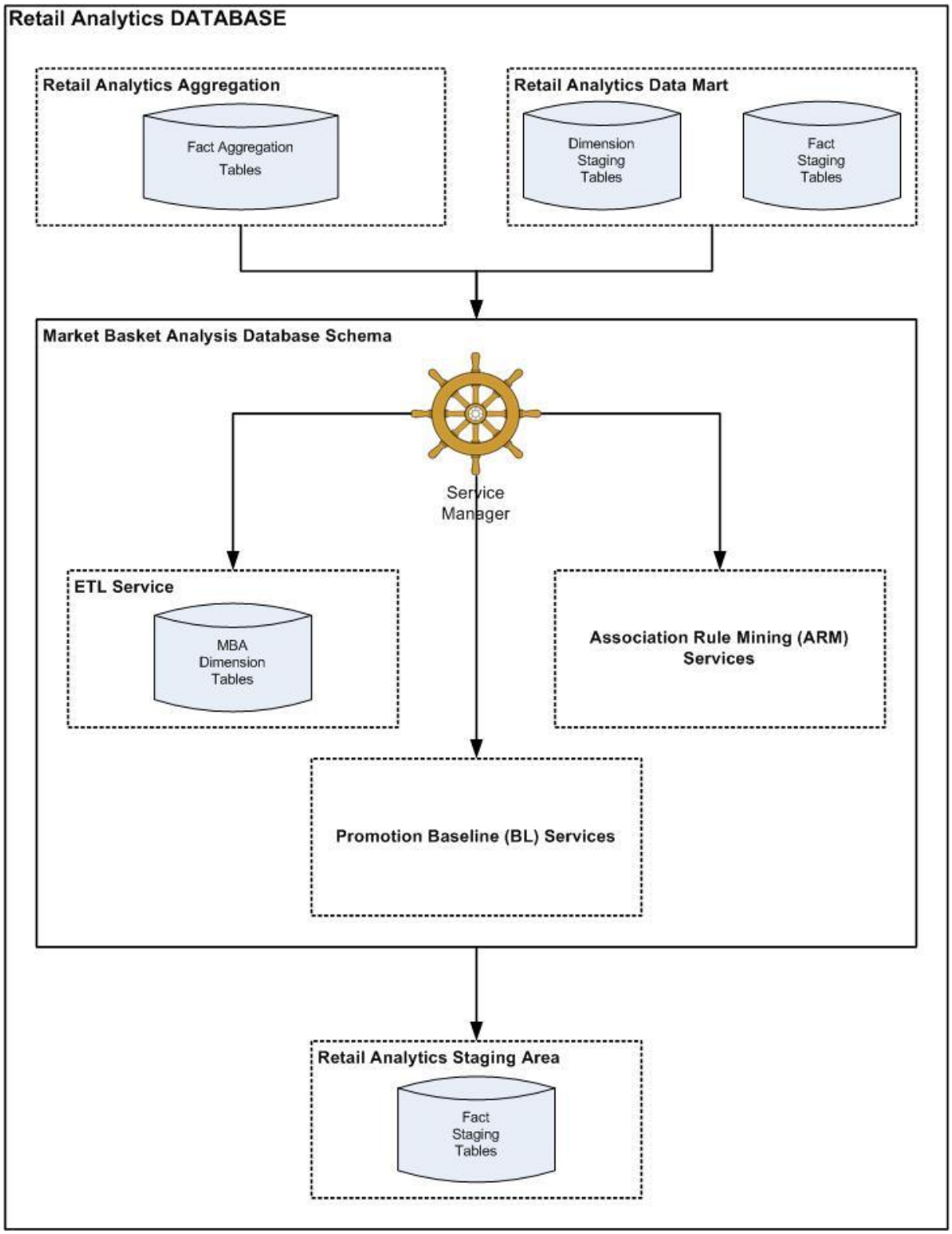
Name of the country where each customer resides. Nominal.

United Kingdom	93%
Germany	2%
Other (25400)	5%

Valid	522k	100%
Mismatched	0	0%
Missing	0	0%
Unique	30	
Most Common	United King...	93%

MBA ETL Process flow Diagram




Retail Analytics DATABASE
Retail Analytics Aggregation
Fact Aggregation Tables
Retail Analytics Data Mart
Dimension Staging Tables
Fact Staging Tables
Market Basket Analysis Database Schema
Service Manager
ETL Service
MBA Dimension Tables
Association Rule Mining (ARM) Services
Promotion Baseline (BL) Services
Retail Analytics Staging Area
Fact Staging Tables

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

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

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 (,)

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

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

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

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

```
#quote: If TRUE it will surround character or factor column with double quotes.
#If FALSE nothing will be quoted
#row.names: either a logical value indicating whether the row names of x are to be
#written along with x, or a character vector of row names to be written.
write.csv(transaxtionData, "assigment1_itemslist.csv", quote = FALSE, row.names = FALSE)
```

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

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.

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

Let's have a view our transaction object by summary(transaction)

```
summary(transactions)
```

We can see 18193 transactions (rows) and 7698 items (columns). 7698 is the product descriptions and 18193 transactions are collections of these items.

```
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                               1395
                      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  860  744  743  743  696  642  633  632  566  598  517  494  520  533  508  460  428  468  406  385  307  306  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
 210  213  209  164  153  135  140  131  108  109   88  108   90   86   84   84   63   58   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   16   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    3    2    1    3    1    3    3    3    1    2    2    1    3    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
```

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.

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

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