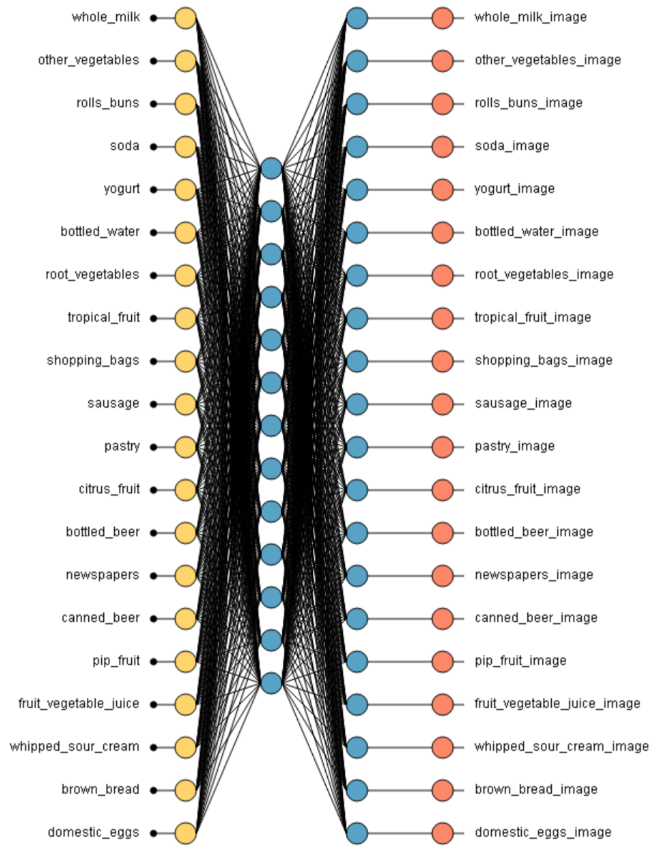
Market Basket Analysis

Apriori Association Rules

Pankaj Shah

1/31/2019



1 Problem:

Purpose :
A Marketer is interested in knowing what product is purchased with what product or if certain products are purchased together as a group of items which they can use to strategize on the cross selling activities.

Steps we will take to tackle above problem.

- First, we listen through data and understand the concept.
- Then, we learn the relationship between the variables.
- then we lead by developing better algortithm.

We know that nowadays, recommendation systems are highly based on machine learning methods that can learn the behavior, e.g., purchasing patterns, of data behaviors.

2 Introduction

Market basket analysis is the reasoning behind the art of arranging items in a store. Product placements should be done in such a way that the items frequently bought together are kept next to each other, so that customers are encouraged to buy them and so that this results in a boost in sales. If we love Shopping or have bought some products either online or anywhere we should have definetly heard about Market Basket Analysis term. When you go through McDonalds, Burger King, Taco Bell or any fast food chain they usually ask you if you would like to get french fries, sundae, or some other things that go well with the products you purchase. If you go for grocery shopping and bought milk and bread then you are more likely to buy eggs. When shopping online in Amazon, Walmart or any other retail store you couldn't have missed the screen that says people who have bought ABC have also bought product XYZ. All these is nothing but Market Basket way of selling more products to consumer and make their shopping experience more enjoyable adding more revenue to the company. So what is Market Basket Analysis truely based upon. How does Netflix knows What kind of Movies I would like. When two or more products are purchased, Market Basket Analysis is done to check whether the purchase of one product increases the likelihood of the purchase of other products. This knowledge is a tool for the marketers to bundle the products or strategize a product cross sell to a customer.

Market Basket Analysis is a modelling technique based upon the theory that if you buy a certain group of items, you are more (or less) likely to buy another group of items. Market Basket Analysis is one of the key techniques used by large retailers to uncover associations between items.

If you eager to know about the model or Algorithm behind the Market Basket Analysis it is the APRIORI Algorithm. Below I will try to explain how retailers or any business personal help themself to boost their business by predicting what items customers buy together by learning historical past data and predicting the future.

Let me explain Couple of other terms that you are more likely to come across while going through my project below.

1. Association Rule Mining:

Association Rules There are many ways to see the similarities between items. These are techniques that fall under the general umbrella of association. The outcome of this type of technique, in simple terms, is a set of rules that can be understood as “if this, then that”

- When do we use Association Rule Mining while doing Market Basket Analysis?

Simple and plain answer is When we are trying to find an association between different objects in our given datasets. For Model to do better we do need bigger sample of datasets. The more the size of datasets and more the frequency of the repeated items that occur. It is more likely to predict accurately. What we see in Market Basket Analysis while doing Clustering, retailing or Classification is applicaion of Association Rule Mining. All the Data Scientist or Data Analyst are trying to find is the association between different consumers what they are buying it together. Simple terms trying to see repeated chains by generating set of rules.

Enough of Explaining Technical term. Lets take a real world grocery shopping example. If you go to any super market. If you have Bread, Milk, Flour in our basket then it is more likely to have Egg in our basket rather than a bottle of Shampoo.

- How can Retailer benfit from these knowledge?

By building up the Architecture of the store to keep the products close to each other or far apart. Sometimes we think why don’t they have milk/ diary product right next to Egg . But as Store models they want you to spend more times they are kept far apart. One thing I have notice in Market Basket Grocery in couple places in Boston. As soon I enter the size of basket is hugely large. So that Pshycologically my goal is to fill by the time I walk out of grocery store. I am greated with Breakfast item like bagel, muffins, egg, bannana. I believe as we start our morning with these things. Pshycologically they are creating in back of my mind what products should I look when I fill my basket. Its all persusian that is built so that I spend more times looking for the things around in Chronological order. Most of the times we don’t think all of these but these is how most of the times we are persuased and spend more than what we want.

Data Scietiest/ Analyst cannot predict the future until and unless they have train themself with the past. In historical datasets all they are doing is finding the association chain rule between different objects in a set of transcation that we have made. All these transactional database can be used to train a model so that Model learns all these chains and predicts the likelyhood what the next person will buy if they bought product ABC and XYZ.

Lets get little deeper to understand the componets of Market Basket Analysis.

Lets say we have some datasets where we have two sets of item. They are A and B. To make it easy lets take our grocery example Milk => Bread [Support= 30%, confidence=60%]

So what does above code even mean?

- It means that 30% historical transaction have shown that Bread is bought with purchase of a Milk
- 60% of customers who purchase Milk have also bought with purchase of a Bread.

Generally association rules are written in "IF-THEN" format. We can also use the term "antecedent" for IF and "Consequent" for THEN.Milk is refered as Antecedent and Bread over here will be refered as Consequent.

3 Key Term and Things to know:

- Market Basket Analysis
- Apriori algorithm
- Association rule learning
- support
- confidence
- lift and
- conviction

Some more terms people who have learnt Market Basket Analysis also have known :

1. *Itemset* : Collection of one or more items. n-item-set means a set of n items.
2. *Support Count* : Frequency of occurrence of an item-set.
3. *Support* : Fraction of transactions that contain the item-set.

We can measure the rule by measuring these two famous terms Support and Confidence. We can set for any datasets what would be our minimum support and what would be our minimum Confidence Tresholds.

Frequent Itemsets : Item-sets whose support is greater or equal than minimum support threshold (min_sup).

Strong rules If a rule A=>B[Support, Confidence] satisfies min_sup and min_confidence then it is a strong rule. **Good Models have strong rules.**

Lift Lift gives the correlation between A and B in the rule A=>B. Correlation shows how one item-set A effects the item-set B. A and B are independent if: P(AUB)=P(A)P(B) otherwise dependent.

Two Golden Rules of Association Rule Mining - Support greater than or equal to min_support - Confidence greater than or equal to min_confidence

Association Rule Mining is viewed as a two-step approach:

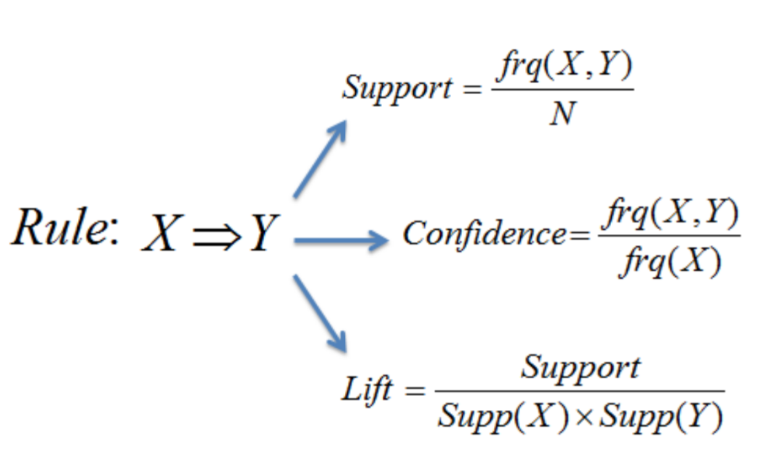
- **Frequent Itemset Generation** Find all frequent item-sets with support >= pre-determined min_support count

- Rule Generation
  - List all Association Rules from frequent item-sets.

- Calculate Support and Confidence for all rules.
- Prune rules that fail min_support and min_confidence thresholds.

4 Measuring rule importance by using support and confidence.

Support and confidence are the two criteria to help us decide whether a pattern is “interesting”. By setting thresholds for these two criteria, we can easily limit the number of interesting rules or item-sets reported.



Support :

$$supp(X \Rightarrow Y) = \frac{|X \cup Y|}{n}$$

For item-sets X and Y, the `support` of an item-set measures how frequently it appears in the data:

$$support(X) = \frac{count(X)}{N},$$

Confidence:

$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

For a rule $X \rightarrow Y$, the `rule's confidence` measures the relative accuracy of the rule:

$$confidence(X \rightarrow Y) = \frac{support(X, Y)}{support(X)}$$

Things to remember
Higher the confidence , stronger the rule is.
As a general rule, Lift ratio of greater than one suggests some usefulness in the rule.

- Frequent Itemset Generation: Most Computionally Expensive, full database scan
- Frequent item set: High frequency Item in Transcations
- Support: Impact in terms of overall size.
- Confidence: Operational usefulness of a rule, conditional probability that customer buy product A will also buy product B.

- Lift ratio : how efficient in the rule is in finding consequences, compared to random selection of transaction. Information about the change in probability of Item A in presence of Item B.

- Lift > 1
  - A lift greater than 1 indicates that the presence of A has increased the probability that the product B will occur on this transaction.
- Lift < 1
  - A lift smaller than 1 indicates that the presence of A has decreased the probability that the product B will occur on this transaction

Lift: The ratio of the observed support to that expected if X and Y were independent.

$$lift(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)supp(Y)}$$

The first step of Apriori is to count up the number of occurrences, i.e., the support, of each member item separately. By scanning the database for the first time.

Market Basket Analysis with XLMINER in Excel.After installing the XLMINER you should be able to find it as an Add-in in your MS Excel.

A Brief intro to XLMINER:

XLMINER is a Excel Add-in which can be used for performing data mining works like neural nets, classification, regression and much more.

Interpretation of the output:

- The item set should exceed minimum support determined based on the business need.
- Should exceed the minimum confidence.
- Should have greater Lift Ratio.
- % increase of chance of buying other product(s) = (Lift - 1) * 100
- A lift value of 1.25 implies that chance of buying product B (on the right hand side) would increase by 25%.

Practical Application

Lift indicates the strength of an association rule over the random co-occurrence of Item A and Item B, given their individual support.

Drawback of Confidence

- Confidence does not measure if the association between A and B is random or not.
- Whereas, Lift measures the strength of association between two items.

5 Apriori & FP growth Algorithm.

Mining association rules and frequent item sets allows for the discovery of interesting and useful connections or relationships between items.

The objectives of the study are the following:

Most of the Market Basket Analysis are done - to obtain association rules - analyze them for better decision support - better understanding of data association - increasing company profit using the Apriori Algorithm and FP-Growth Algorithm - to analyze association rules based on relevance, interestingness, and correlation, - Use lift, Imbalance Ratio (IR), and Kulczynski (Kulc) measure as correlation measures.

Transaction	Items
T1	{Milk, Egg, Bread}
T2	{Milk, Coffee}
T3	{Coffee, Butter}
T4	{Milk, Egg, Coffee}
T5	{Milk, Egg, Sugar, Coffee, Bread}
T6	{Egg, Sugar, Bread}
T7	{Egg, Bread, Sugar}

$$I = \{i_1, i_2, i_3, \ldots, i_n\}$$

In our case it corresponds to:

$$I = \{T\text{-}Milk, Egg, Bread, Coffee, Sugar, Butter\}$$

- Item set : No. of individaul items in above each Transactions. [A-Z]
    - $$I = \{T\text{-}Milk, Egg, Bread, Coffee, Sugar, Butter\}$$
- Transaction : Individual transaction happen every time. e.g [AB, DE, KJ, LOY, POK]
    - Example:
        - T1={Milk, Egg, Bread}
- Association Rule :
    - X⇒Y , where X⊂I, Y⊂I and X∩Y=0
    - {T-Milk, Egg} ⇒ {Bread}
- If combination of AB will Result to C, combination of something should result to something.
- In Simple terms if we have to define support,it is nothing but an indication of how frequently the item set appears in the data set.
    - number of transactions with both X and Y divided by the total number of transactions.
    - not useful for low support values
- For a rule X⇒Y, confidence shows the percentage in which Y is bought with X.
- It's an indication of how often the rule has been found to be true.
- For example, the rule Milk ⇒ Egg has a confidence of 3/4, which means that for 75% of the transactions containing a t-shirt the rule is correct (75% of the times a customer buys a t-shirt, trousers are bought as well)

Conviction

$$conv(X \Rightarrow Y) = \frac{1 - supp(Y)}{1 - conf(X \Rightarrow Y)}$$

- It can be interpreted as the ratio of the expected frequency that X occurs without Y if X and Y were independent divided by the observed frequency of incorrect predictions.
- A high value means that the consequent depends strongly on the antecedent.

A **transaction** is represented by the following expression:

$$T = \{t_1, t_2, \ldots, t_n\}$$

Then, an **association rule** which is defined as an implication of the form:

$$X \Rightarrow Y\text{, where } X \subset I,\ Y \subset I \text{ and } X \cap Y = 0$$

For example,

$$\{T\text{-}Milk, Egg\} \Rightarrow \{Bread\}$$

6 Loading Libraries

```
library(tidyverse) # helpful in Data Cleaning and Manipulation
library(arules) # Mining Association Rules and Frequent Itemsets
library(arulesViz) # Visualizing Association Rules and Frequent Itemsets
library(gridExtra) #  low-level functions to create graphical objects
library(ggthemes) # For cool themes like fivethirtyEight
library(dplyr) # Data Manipulation
library(readxl)# Read Excel Files in R
library(plyr)# Tools for Splitting, Applying and Combining Data
library(ggplot2) # Create graphics and charts
library(knitr) # Dynamic Report generation in R
library(lubridate) # Easier to work with dates and times.
library(kableExtra) # construct complex tables and customize styles
library(RColorBrewer) # Color schemes for plotting
```

7 Information about Datasets:

Implementing MBA/Association Rule Mining using R

In this project, we will use a dataset from the UCI Machine Learning Repository. The dataset is called Online-Retail, and we can download it from here (http://archive.ics.uci.edu/ml/datasets/online+retail).

- The dataset contains transaction data from 01/12/2010 to 09/12/2011 for a UK-based registered non-store online retail.

8 Data Preparation

```
#read excel into R dataframe
retail <- read_excel('~/Desktop/R_markdown/Market_Basket_Analysis/Online Retail.xl
  sx')
retail <- retail[complete.cases(retail), ] # will clean up the non missing values.
```

Let’s get an idea of what we’re working with.

```
glimpse(retail)
```

```
## Observations: 406,829
## Variables: 8
## $ InvoiceNo   <chr> "536365", "536365", "536365", "536365", "536365", ...
## $ StockCode   <chr> "85123A", "71053", "84406B", "84029G", "84029E", "...
## $ Description <chr> "WHITE HANGING HEART T-LIGHT HOLDER", "WHITE METAL...
## $ Quantity    <dbl> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2...
## $ InvoiceDate <dttm> 2010-12-01 08:26:00, 2010-12-01 08:26:00, 2010-12...
## $ UnitPrice   <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1....
## $ CustomerID  <dbl> 17850, 17850, 17850, 17850, 17850, 17850, 17850, 1...
## $ Country     <chr> "United Kingdom", "United Kingdom", "United Kingdo...
```

Dataset Description - Number of Rows: 406,829 - Number of Attributes: 8

Attribute Information:

- InvoiceNo: Invoice number, Nominal,6-digit unique transcation number. 'c'- cancellation.
- StockCode: Product (item) code, Nominal, 5-digit distinct product Number.
- Description: Description about Product Name, Nominal.
- Quantity: The quantities of each product (item) per transaction, Numeric.
- InvoiceDate: Invoice Date and time, Numeric
- UnitPrice: Unit price,Numeric, Product price per unit in pound sterling not to be confused with Dollar.
- CustomerID: Customer number,Nominal, a 5-digit integral number uniquely assigned to each customer.
- Country: Country name,Nominal, the name of the country where each customer resides.

9 Data Cleaning:

First step lets clean up the class variables for the datasets.

```
retail$Description <- as.factor(retail$Description)
retail$Country <- retail$Country
retail$Date <- as.Date(retail$InvoiceDate)
retail$InvoiceNo <- as.numeric(as.character(retail$InvoiceNo))
retail$Time <- format(retail$InvoiceDate,"%H:%M:%S")
```

```
#ddply(dataframe, variables_to_be_used_to_split_data_frame, function_to_be_applie
  d)
transaction_data <- ddply(retail,c("InvoiceNo","Date"),
                       function(df1)paste(df1$Description,
                       collapse = ","))
# paste() concatenates vectors to character and separated results using collapse=
  [any optional charcater string ]. Here ',' is used
```

```
## Remove redundancies
transaction_data$InvoiceNo <- NULL # set column InvoiceNo of dataframe transaction
  Data
transaction_data$Date <- NULL # set column Date of dataframe transactionData
colnames(transaction_data) <- c("items") # Rename column to items
```

10 Write CSV

SAVE THE FILE AS OUTPUT

```
write.csv(transaction_data,'~/Desktop/R_markdown/Market_Basket_Analysis/market_bas
  ket_transactions.csv', quote = FALSE, row.names = TRUE)
# Quote : TRUE "character or factor column with double quotes."
# Quote : 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.
```

Transaction data file which is in basket format let's convert it into an object of the transaction class.

```
# Will get lots of EOF within quoted string in your output
tr <- read.transactions('~/Desktop/R_markdown/Market_Basket_Analysis/market_basket
  _transactions.csv', format = 'basket', sep=',')
# sep tell how items are separated.
```

transactions as itemMatrix in sparse format with
18839 rows (elements/itemsets/transactions) and
26725 columns (items) and a density of 0.0007046267

11 Summary

```
summary(tr)
```

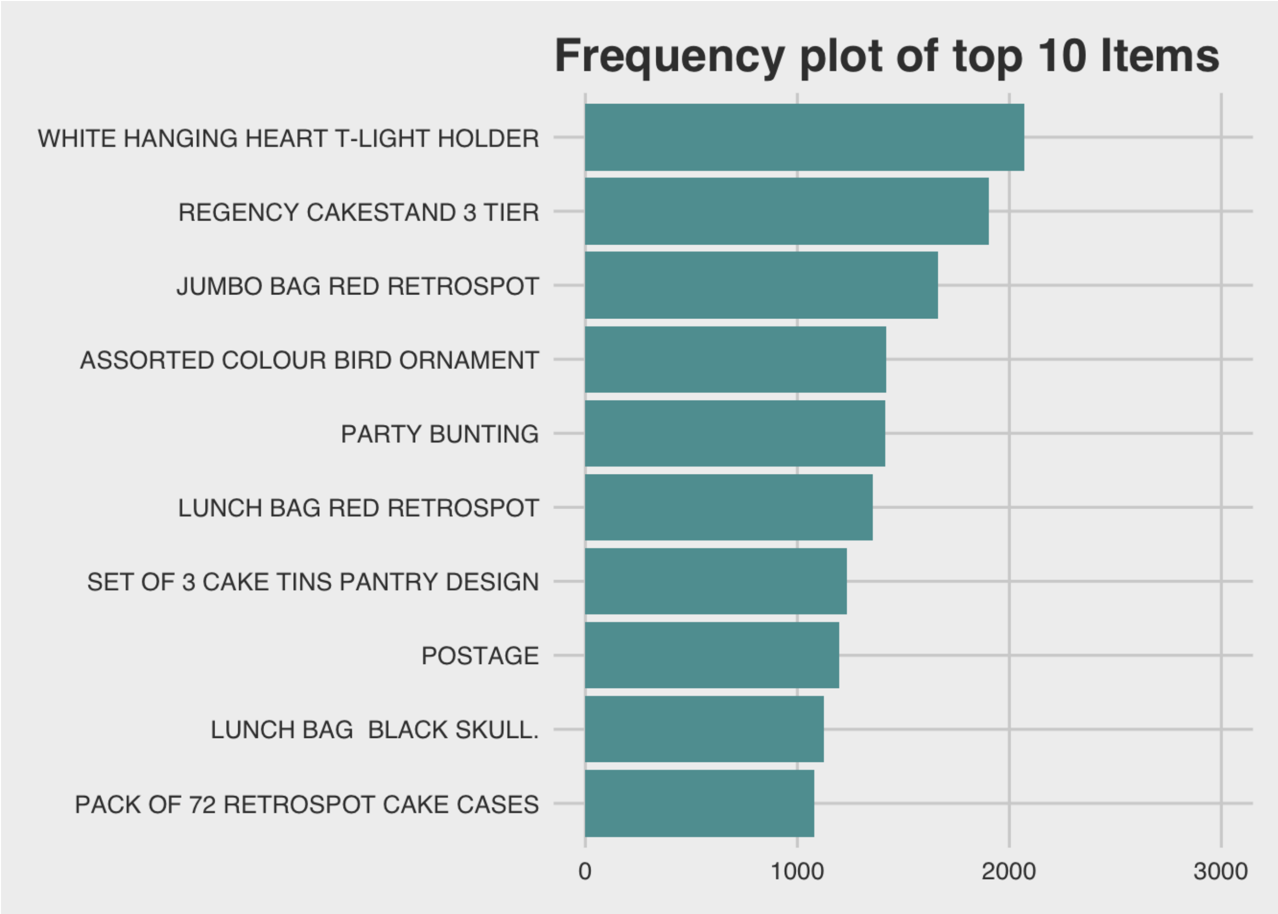
```
## transactions as itemMatrix in sparse format with
##  18839 rows (elements/itemsets/transactions) and
##  26725 columns (items) and a density of 0.0007046267
##
## most frequent items:
## WHITE HANGING HEART T-LIGHT HOLDER           REGENCY CAKESTAND 3 TIER
##                               1798                               1644
##            JUMBO BAG RED RETROSPOT                      PARTY BUNTING
##                               1450                               1282
##      ASSORTED COLOUR BIRD ORNAMENT                            (Other)
##                               1249                             347337
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
##    1 1577  867  762  773  768  721  660  652  648  586  621  532  510  532
##   16   17   18   19   20   21   22   23   24   25   26   27   28   29   30
##  555  525  470  442  483  425  396  319  310  276  241  255  230  218  223
##   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45
##  215  173  163  143  146  139  112  118   89  117   96   97   89   93   67
##   46   47   48   49   50   51   52   53   54   55   56   57   58   59   60
##   66   68   65   61   64   53   67   43   42   50   43   37   31   40   30
##   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75
##   27   28   18   26   25   20   27   25   25   15   20   20   13   16   16
##   76   77   78   79   80   81   82   83   84   85   86   87   88   89   90
##   12   16   12    7    9   14   15   12    8    9   11   11   14    8    6
##   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105
##    5    6   12    6    4    4    3    6    5    2    4    2    5    4    3
##  106  107  108  109  110  111  112  113  114  115  117  118  119  121  122
##    2    2    6    3    4    3    2    1    3    1    4    3    3    1    2
##  123  124  126  127  128  132  133  134  135  141  142  143  144  146  147
##    2    1    3    2    2    1    1    2    1    1    2    2    1    1    2
##  148  151  155  158  169  172  178  179  181  203  205  229  237  250  251
##    1    1    3    2    2    2    1    1    1    1    1    1    1    1    1
##  286  321  401  420
##    1    1    1    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    6.00   14.00   18.83   24.00  420.00
##
## includes extended item information - examples:
##     labels
## 1        1
## 2 1 HANGER
## 3       10
```

12 Frequency plot of top 10 Items:

```
top_items<-retail %>%
  dplyr::group_by(Description) %>%
  dplyr::summarise(count=n()) %>%
  dplyr::arrange(desc(count))

summary(retail)
```

```
##    InvoiceNo       StockCode
##  Min.   :536365   Length:406829
##  1st Qu.:549234   Class :character
##  Median :561893   Mode  :character
##  Mean   :560617
##  3rd Qu.:572090
##  Max.   :581587
##  NA's   :8905
##                               Description        Quantity
##  WHITE HANGING HEART T-LIGHT HOLDER:  2070   Min.   :-80995.00
##  REGENCY CAKESTAND 3 TIER          :  1905   1st Qu.:     2.00
##  JUMBO BAG RED RETROSPOT           :  1662   Median :     5.00
##  ASSORTED COLOUR BIRD ORNAMENT     :  1418   Mean   :    12.06
##  PARTY BUNTING                     :  1416   3rd Qu.:    12.00
##  LUNCH BAG RED RETROSPOT           :  1358   Max.   : 80995.00
##  (Other)                           :397000
##   InvoiceDate                    UnitPrice          CustomerID
##  Min.   :2010-12-01 08:26:00   Min.   :    0.00   Min.   :12346
##  1st Qu.:2011-04-06 15:02:00   1st Qu.:    1.25   1st Qu.:13953
##  Median :2011-07-31 11:48:00   Median :    1.95   Median :15152
##  Mean   :2011-07-10 16:30:57   Mean   :    3.46   Mean   :15288
##  3rd Qu.:2011-10-20 13:06:00   3rd Qu.:    3.75   3rd Qu.:16791
##  Max.   :2011-12-09 12:50:00   Max.   :38970.00   Max.   :18287
##
##    Country               Date                 Time
##  Length:406829      Min.   :2010-12-01   Length:406829
##  Class :character   1st Qu.:2011-04-06   Class :character
##  Mode  :character   Median :2011-07-31   Mode  :character
##                     Mean   :2011-07-10
##                     3rd Qu.:2011-10-20
##                     Max.   :2011-12-09
##
```

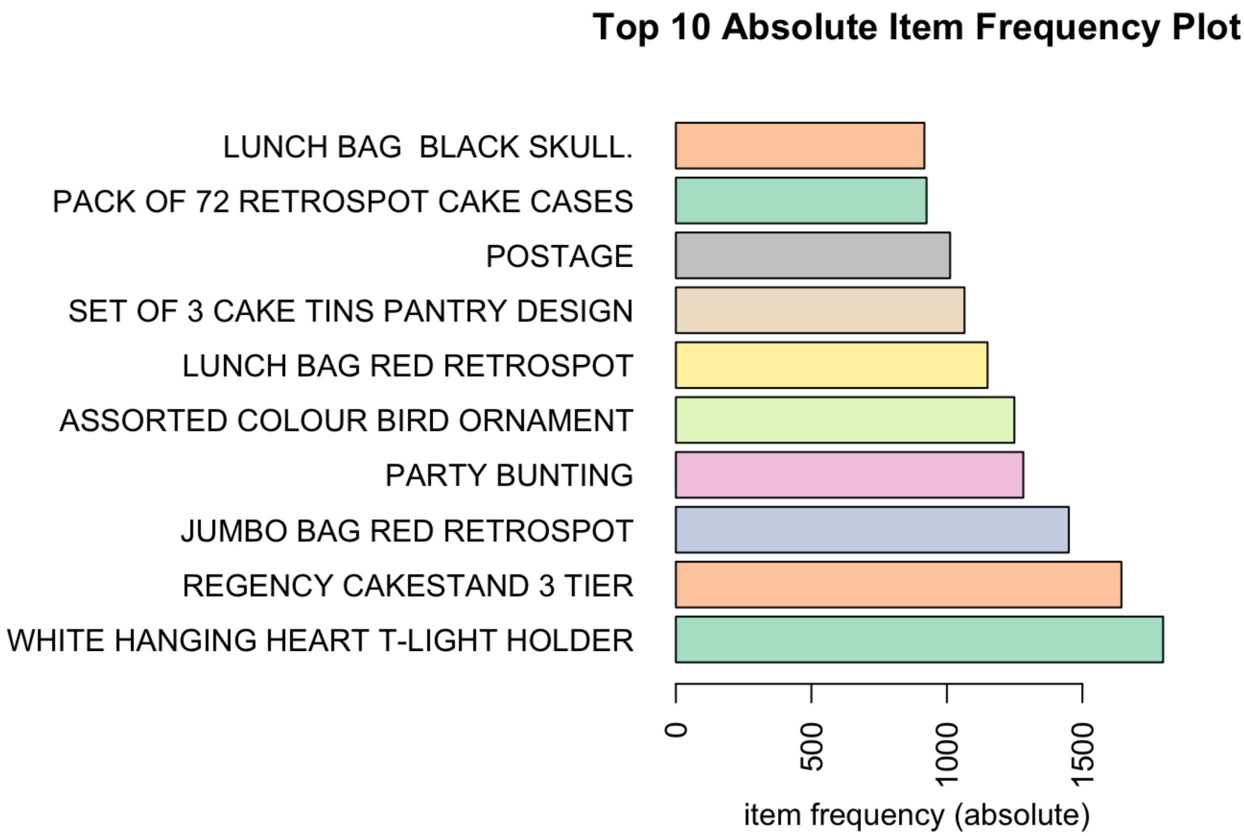
```
top_items<-head(top_items,10)

ggplot(top_items,aes(x=reorder(Description,count), y=count))+
  geom_bar(stat="identity",fill="cadetblue")+
  coord_flip()+
  scale_y_continuous(limits = c(0,3000))+
  ggtitle("Frequency plot of top 10 Items")+
  xlab("Description of item")+
  ylab("Count")+
  theme_fivethirtyeight()
```



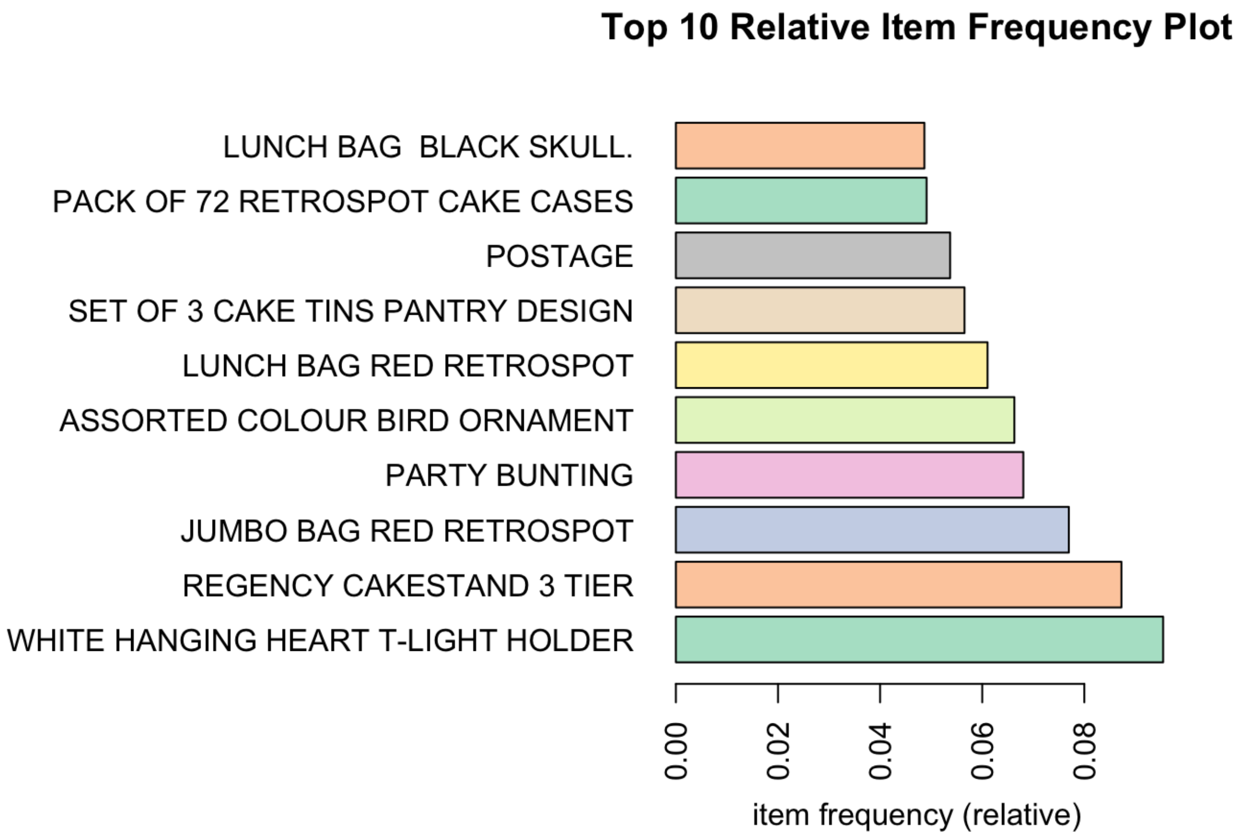
- Lets Plot Item Frequency Bar Plot to view distribution.

We can plot either Relative or Absolute Values. - Absolute: plot numeric frequencies of each item independently - Relative: how many times these items have appeared as compared to others.

```
itemFrequencyPlot(tr,topN=10,type="absolute",col=brewer.pal(8,'Pastel2'), main="To
 p 10 Absolute Item Frequency Plot", horiz = TRUE)
```




```
itemFrequencyPlot(tr,topN=10,type="relative",col=brewer.pal(8,'Pastel2'),main="Top
   10 Relative Item Frequency Plot", horiz = TRUE)
```



`WHITE HANGING HEART T-LIGHT HOLDER` and `REGENCY CAKESTAND 3 TIER`,

This plot shows that `WHITE HANGING HEART T-LIGHT HOLDER` and `REGENCY CAKESTAND 3 TIER` have the most sales. U can see at the bottom two of the chart.So to increase the sale of `SET OF 3 CAKE TINS PANTRY DESIGN` the retailer can put it near `REGENCY CAKESTAND 3 TIER`.

Next we will mine the rules using the APRIORI algorithm. The function apriori() is from package arules.

```
# Parameter Spec: min_sup=0.001, min_confidence=0.8 values with 10 items max items
   in rule.
association_rules <- apriori(tr, parameter = list(supp=0.001, conf=0.8,maxlen=10))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target   ext
##      10  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 18
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[26725 item(s), 18839 transaction(s)] done [0.18s].
## sorting and recoding items ... [2455 item(s)] done [0.01s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 7 8 9 10
```

```
## Warning in apriori(tr, parameter = list(supp = 0.001, conf = 0.8, maxlen =
## 10)): Mining stopped (maxlen reached). Only patterns up to a length of 10
## returned!
```

```
##  done [0.59s].
## writing ... [116493 rule(s)] done [0.05s].
## creating S4 object  ... done [0.05s].
```

- minval: minimum value of the support an itemset should satisfy to be a part of a rule.
- smax: maximum support value for an itemset.
- AREM(Additional Rule Evaluation Parameter): constrained the number of rules using Support & Confidence. There are several other ways to constrain the rules
- AVAL: logical indicating whether to return the additional rule evaluation measure selected with arem.
- originalSupport: The traditional support value only considers both LHS and RHS items for calculating support. If you want to use only the LHS items for the calculation then you need to set this to FALSE.
- maxtime : maximum amount of time allowed to check for subsets.
- minlen : minimum number of items required in the rule.
- maxlen : maximum number of items that can be present in the rule.

- The apriori will take tr as the transaction object on which mining is to be applied.
- Parameter will allow you to set min_sup and min_confidence.
- The default values:
  - minimum support of 0.1, the minimum confidence of 0.8, maximum of 10 items (maxlen).

```
summary(association_rules) #shows the following:
```

```
## set of 116493 rules
##
## rule length distribution (lhs + rhs):sizes
##     2     3     4     5     6     7     8     9    10
##   111  3378 10947 29980 39875 23872  6860  1249   221
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   5.000   6.000   5.826   7.000  10.000
##
## summary of quality measures:
##     support           confidence          lift              count
##  Min.   :0.001009   Min.   :0.8000   Min.   :  8.382   Min.   : 19.00
##  1st Qu.:0.001062   1st Qu.:0.8333   1st Qu.: 18.897   1st Qu.: 20.00
##  Median :0.001168   Median :0.8750   Median : 23.917   Median : 22.00
##  Mean   :0.001323   Mean   :0.8870   Mean   : 48.813   Mean   : 24.92
##  3rd Qu.:0.001380   3rd Qu.:0.9310   3rd Qu.: 39.552   3rd Qu.: 26.00
##  Max.   :0.022453   Max.   :1.0000   Max.   :607.710   Max.   :423.00
##
## mining info:
##  data ntransactions support confidence
##    tr         18839   0.001        0.8
```

- Total number of rules: The set of 116493 rules
- Distribution of rule length:
  - A length of 6 items has the most rules: 39875 &
  - length of 2 items have the lowest number of rules: 111
- Summary of Quality measures: Min and max values for Support, Confidence and, Lift.
- Information used for creating rules: The data, support, and confidence we provided to the algorithm.

Since there are 116493 rules, let's print only top 10:

```
inspectDT(head(sort(association_rules, by = "confidence"), 3))
```

Show 10 **entries** **Search:**

	LHS	RHS	support	confidence	lift	count
	All	All		All		
[1]	{WOBBLY CHICKEN}	{METAL}	0.001	1.000	376.780	28.000
[2]	{WOBBLY CHICKEN}	{DECORATION}	0.001	1.000	376.780	28.000
[3]	{DECOUPAGE}	{GREETING CARD}	0.001	1.000	330.509	23.000

Showing 1 to 3 of 3 entries Previous 1 Next

Limiting the number and size of rules.

- If we want stronger rules, you can increase the value of conf and for more extended rules give higher value to maxlen.

```
shorter_association_rules <- apriori(tr, parameter = list(supp=0.001, conf=0.8,max
  len=3))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target   ext
##       3  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 18
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[26725 item(s), 18839 transaction(s)] done [0.17s].
## sorting and recoding items ... [2455 item(s)] done [0.01s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 done [0.26s].
## writing ... [3489 rule(s)] done [0.03s].
## creating S4 object  ... done [0.01s].
```

Removing redundant rules You can remove rules that are subsets of larger rules.

```
# Use the code below to remove such rules:
subset_rules <- which(colSums(is.subset(association_rules, association_rules)) > 1
  ) # get subset rules in vector
length(subset_rules)  #> 107755
```

```
## [1] 107755
```

```
subset_association_rules <- association_rules[-subset_rules] # remove subset rule
  s.
```

- which() returns the position of elements in the vector for which value is TRUE.
- colSums() forms a row and column sums for dataframes and numeric arrays.
- is.subset() Determines if elements of one vector contain all the elements of other
- Appearance gives us options to set LHS (IF part) and RHS (THEN part) of the rule.

Sometimes, we want to work on a specific product. If we want to find out what causes influence on the purchase of item X we can use appearance option in the apriori command.

For example, to find what customers buy before buying 'METAL'. Lets look into that.

```r
metal.association.rules <- apriori(tr, parameter = list(supp=0.001, conf=0.8),appe
  arance = list(default="lhs",rhs="METAL"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target   ext
##      10  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 18
##
## set item appearances ...[1 item(s)] done [0.00s].
## set transactions ...[26725 item(s), 18839 transaction(s)] done [0.18s].
## sorting and recoding items ... [2455 item(s)] done [0.01s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 7 8 9 10 done [0.59s].
## writing ... [5 rule(s)] done [0.06s].
## creating S4 object  ... done [0.02s].
```

```r
# Here lhs=METAL because you want to find out the probability of that in how many
   customers buy METAL along with other items
inspectDT(head(metal.association.rules))
```

Show 10 entries Search:

	LHS	RHS	support	confidence	lift	count
	All			All		
[1]	{WOBBLY CHICKEN}	{METAL}	0.001	1.000	376.780	28.000
[2]	{WOBBLY RABBIT}	{METAL}	0.002	1.000	376.780	34.000
[3]	{DECORATION}	{METAL}	0.003	1.000	376.780	50.000
[4]	{DECORATION,WOBBLY CHICKEN}	{METAL}	0.001	1.000	376.780	28.000
[5]	{DECORATION,WOBBLY RABBIT}	{METAL}	0.002	1.000	376.780	34.000

Showing 1 to 5 of 5 entries Previous 1 Next

Similarly, to find the answer to the question Customers who bought METAL also bought.... we will keep METAL on lhs:

```
metal.association.rules <- apriori(tr, parameter = list(supp=0.001, conf=0.8),appe
  arance = list(lhs="METAL",default="rhs"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target   ext
##      10  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 18
##
## set item appearances ...[1 item(s)] done [0.00s].
## set transactions ...[26725 item(s), 18839 transaction(s)] done [0.21s].
## sorting and recoding items ... [2455 item(s)] done [0.01s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [1 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

```
# Here lhs=METAL because you want to find out the probability of that in how many
   customers buy METAL along with other items
inspectDT(head(metal.association.rules))
```

Show 10 entries Search:

	LHS	RHS	support	confidence	lift	count
		All		All		
[1]	{METAL}	{DECORATION}	0.003	1.000	376.780	50.000

Showing 1 to 1 of 1 entries Previous 1 Next

13 Vizulatization

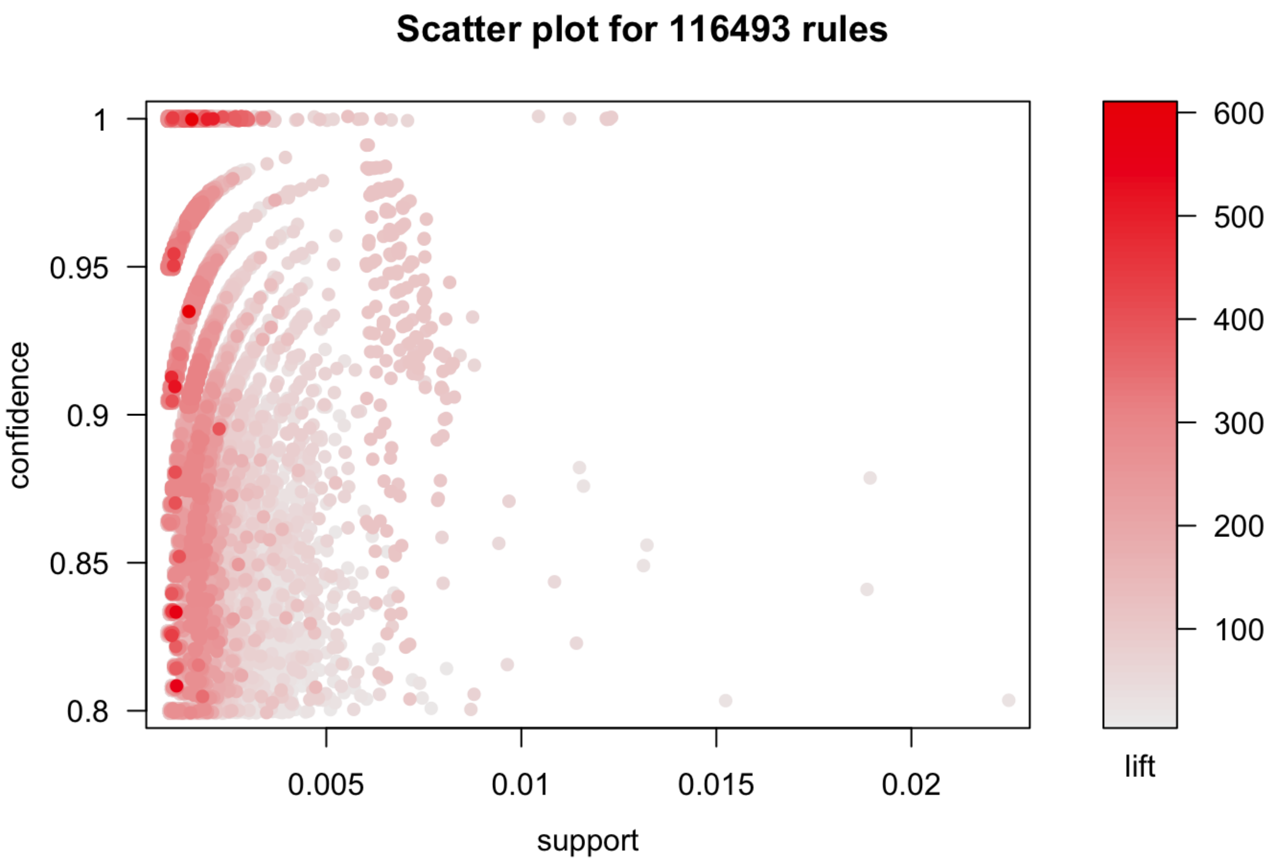
Some of the Vizualization Option:

- Scatter-Plot
- Interactive Scatter-plot
- Individual Rule Representation
- Scatter-Plot :

- straight-forward visualization of association rules
- uses Support and Confidence on the axes.
- Lift is used by default to color (grey levels) of the points.

13.1 Scatterplot

```
# Filter rules with confidence greater than 0.4 or 40%
subRules<-association_rules[quality(association_rules)$confidence>0.4]
#Plot SubRules
plot(subRules)
```



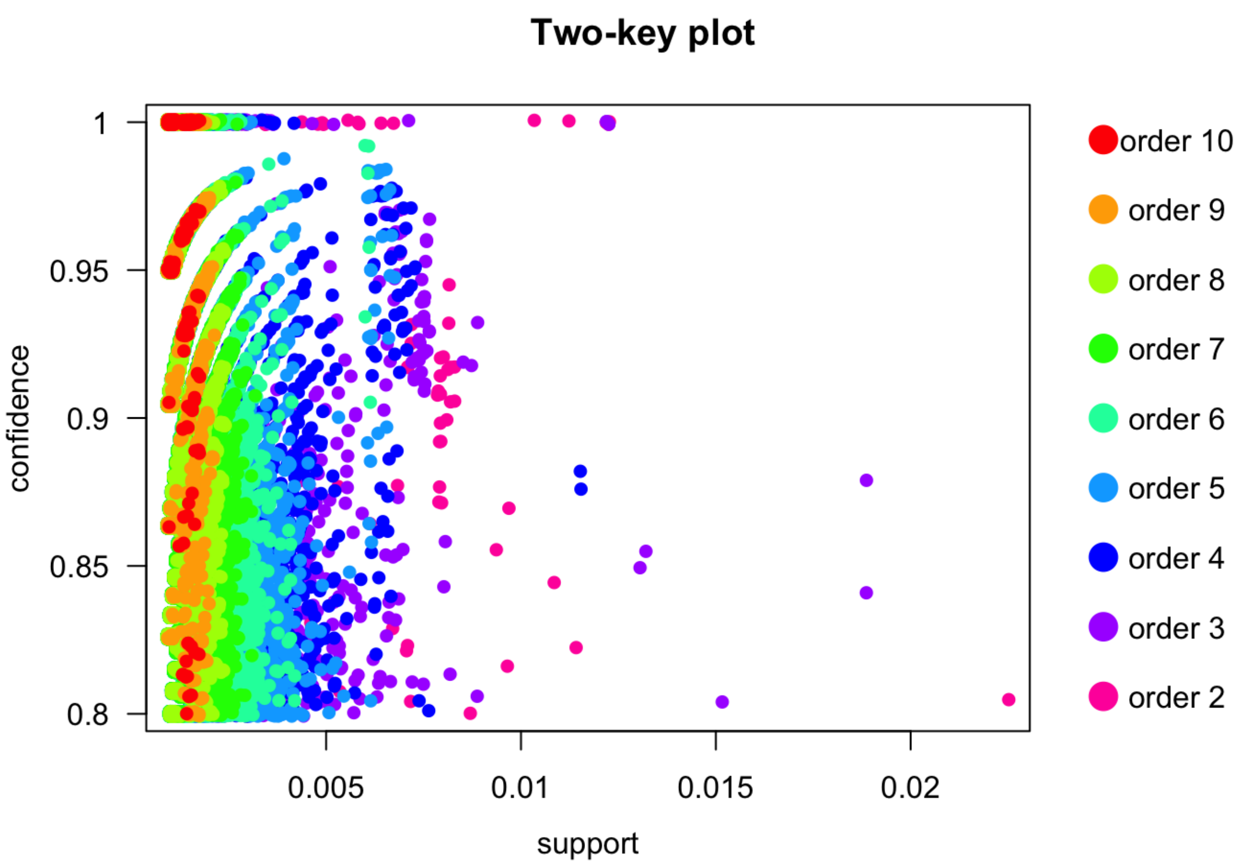
The above plot shows that rules with high lift have low support. We can use the following options for the plot: plot(rulesObject, measure, shading, method)

- rulesObject: the rules object to be plotted
- measure: Measures for rule interestingness.
  - Can be Support, Confidence, lift or combination of these depending upon method value.
- shading: Measure used to color points (Support, Confidence, lift). The default is Lift.
- method: Visualization method to be used (scatterplot, two-key plot, matrix3D).

13.2 The two-key plot

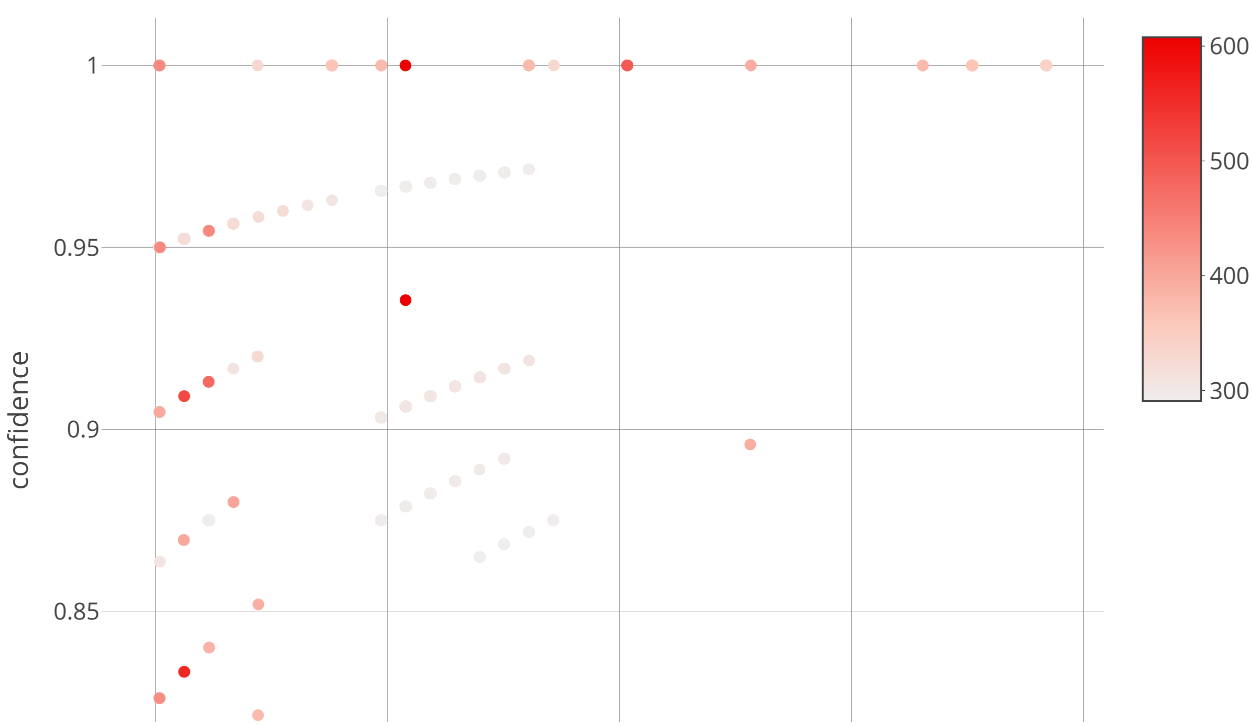
- uses support and confidence on x and y-axis respectively.
- uses order for coloring. The order is the number of items in the rule.

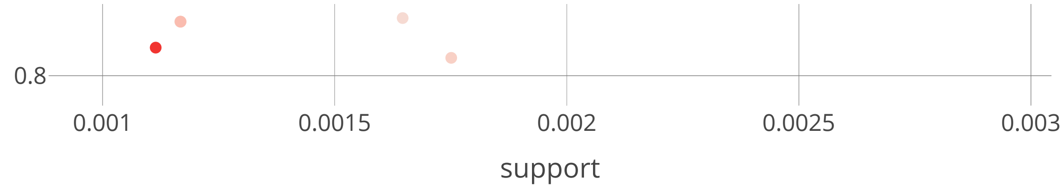
```
plot(subRules,method="two-key plot")
```



Interactive Scatter-Plot : Plotly

```
plotly_arules(subRules)
```





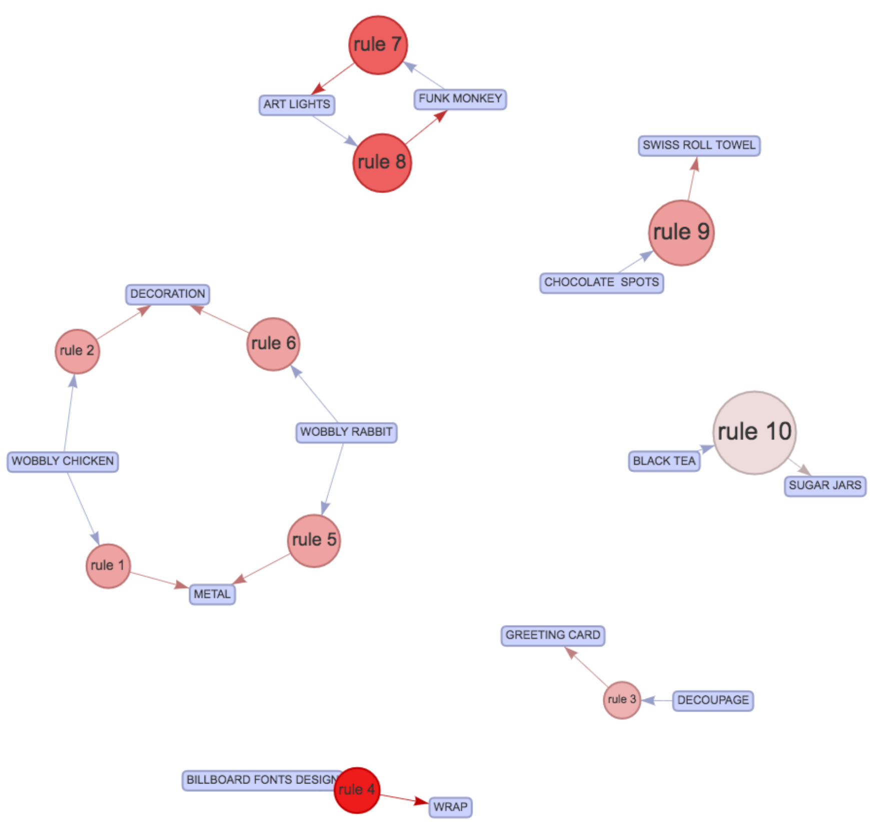
13.3 Graph-Based Visualizations

Graph-based techniques visualize association rules using vertices and edges - Vertices are labeled with item names, and item sets or rules are represented as a second set of vertices. Items are connected with item-sets/rules using directed arrows. - Arrows pointing from items to rule vertices indicate LHS items and an arrow from a rule to an item indicates the RHS. - The size and color of vertices often represent interest measures.

```
#10 rules from subRules having the highest confidence.
top10subRules <- head(subRules, n = 10, by = "confidence")
```

```
plot(top10subRules, method = "graph",  engine = "htmlwidget") #interactive plot en
  gine=htmlwidget
```

Select by id



From arulesViz graphs for sets of association rules can be exported in the GraphML format or as a Graphviz dot-file to be explored in tools like Gephi. For example, the 1000 rules with the highest lift are exported by:

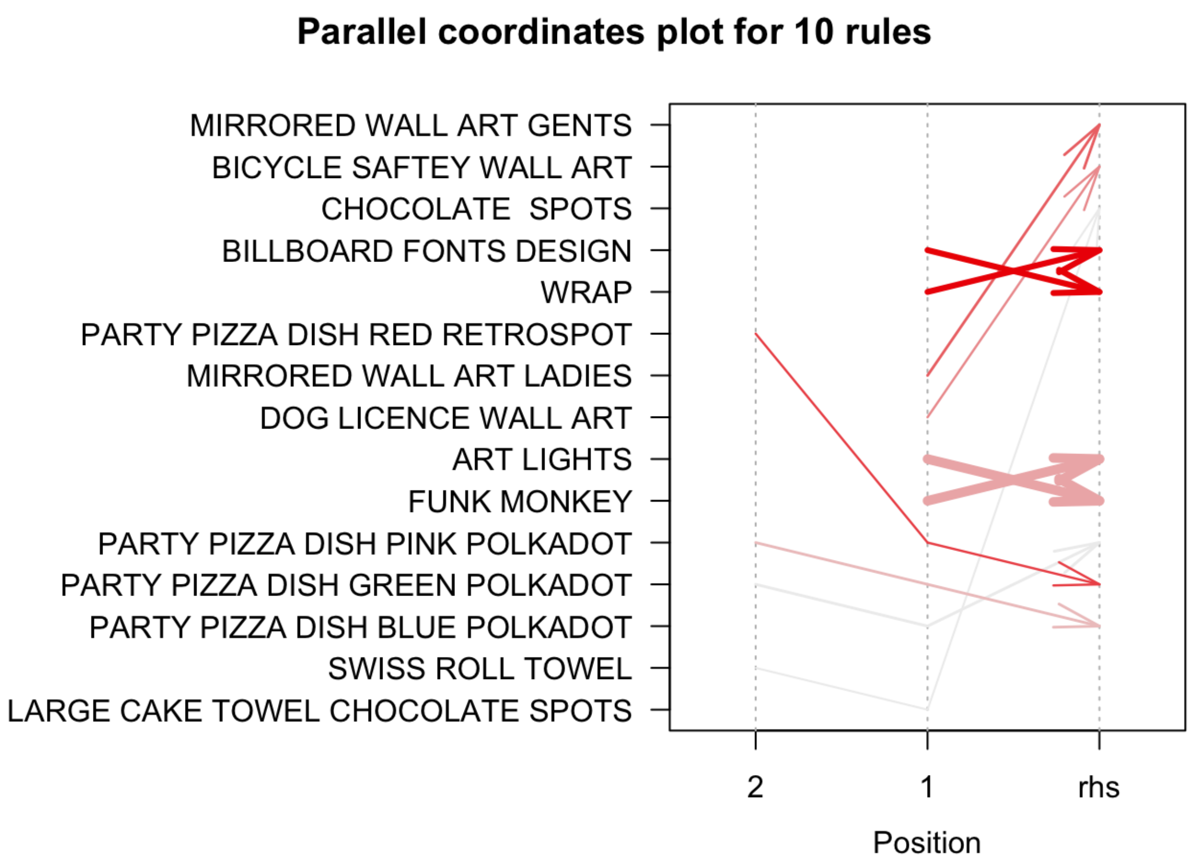
```
saveAsGraph(head(subRules, n = 1000, by = "lift"), file = "rules.graphml")
```

13.4 Individual Rule Representation

- also called as Parallel Coordinates Plot.
- Useful to visualized which products along with which items cause what kind of sales.

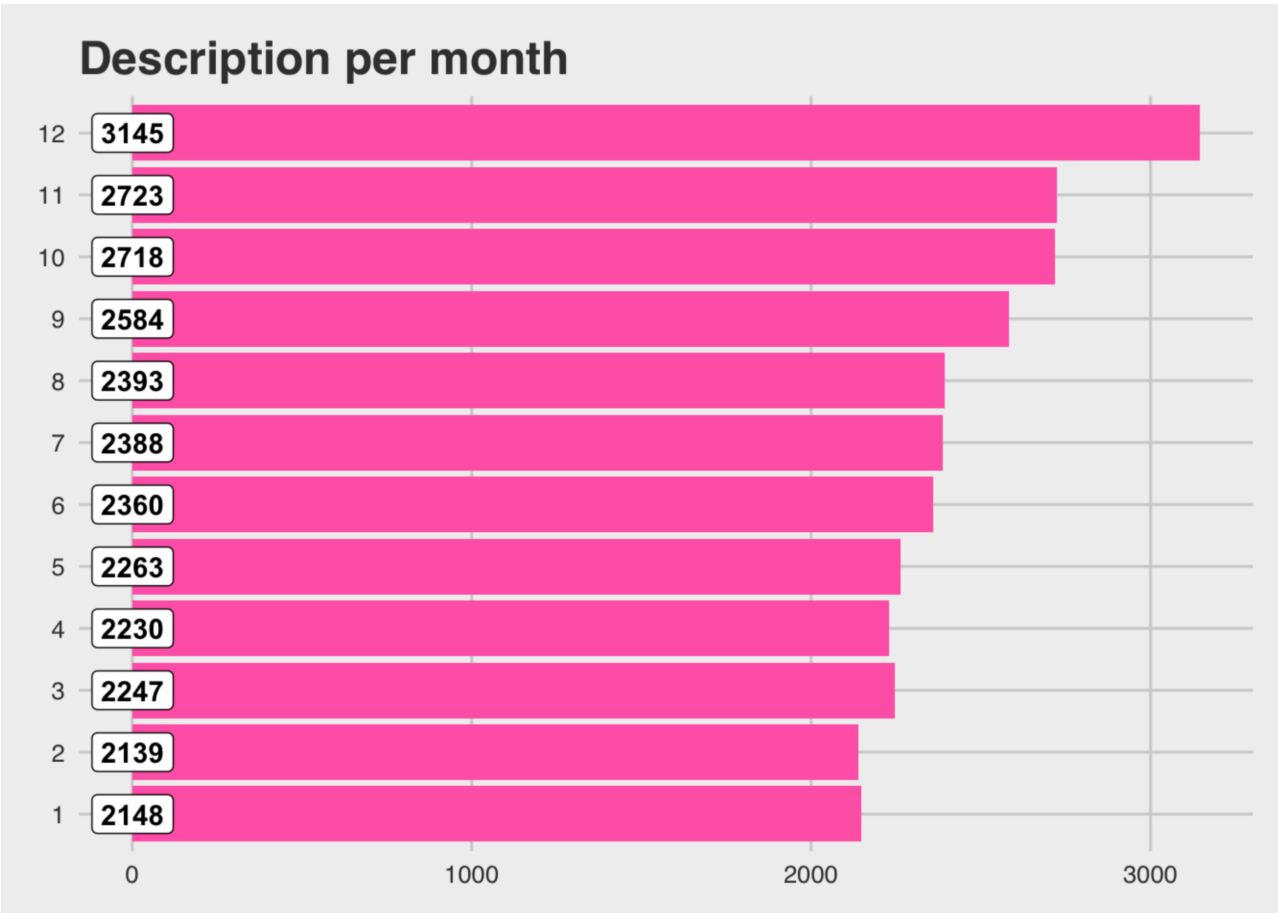
As mentioned above, the RHS is the Consequent or the item we propose the customer will buy; the positions are in the LHS where 2 is the most recent addition to our basket and 1 is the item we previously had.

```
subRules2<-head(subRules, n=10, by="lift")
plot(subRules2, method="paracoord")
```



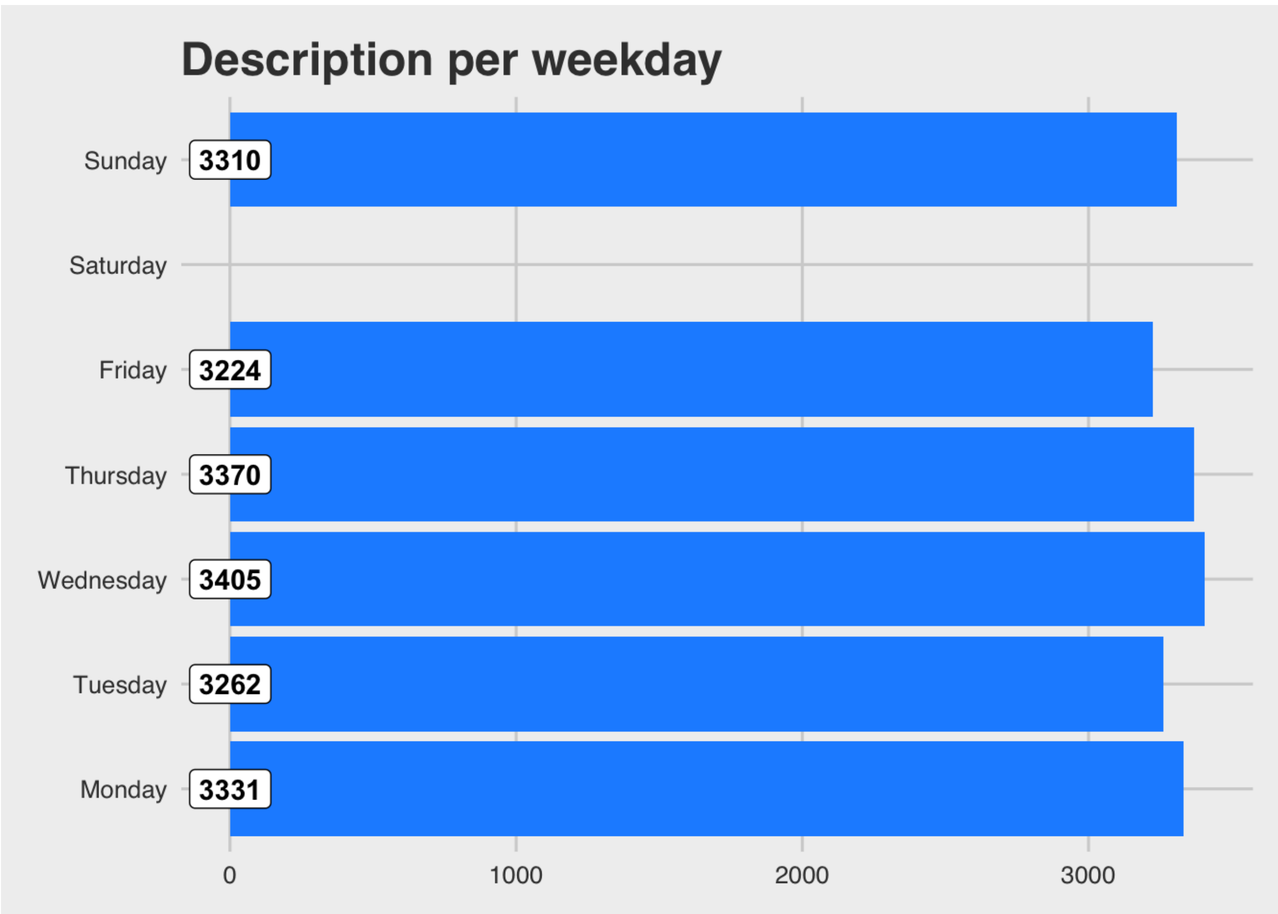
14 Transactions per month

```
# Transactions per month
retail %>%
  mutate(Month=as.factor(month(Date))) %>%
  group_by(Month) %>%
  dplyr::summarize(Description=n_distinct(Description)) %>%
  ggplot(aes(x=Month, y=Description)) +
  geom_bar(stat="identity", fill="#FF69B4", show.legend=FALSE) +
  geom_label(aes(label=Description, y= 1, fontface = 'bold')) +
  labs(title="Description per month") +
  theme_fivethirtyeight()+
  coord_flip()
```



15 Transactions per weekday

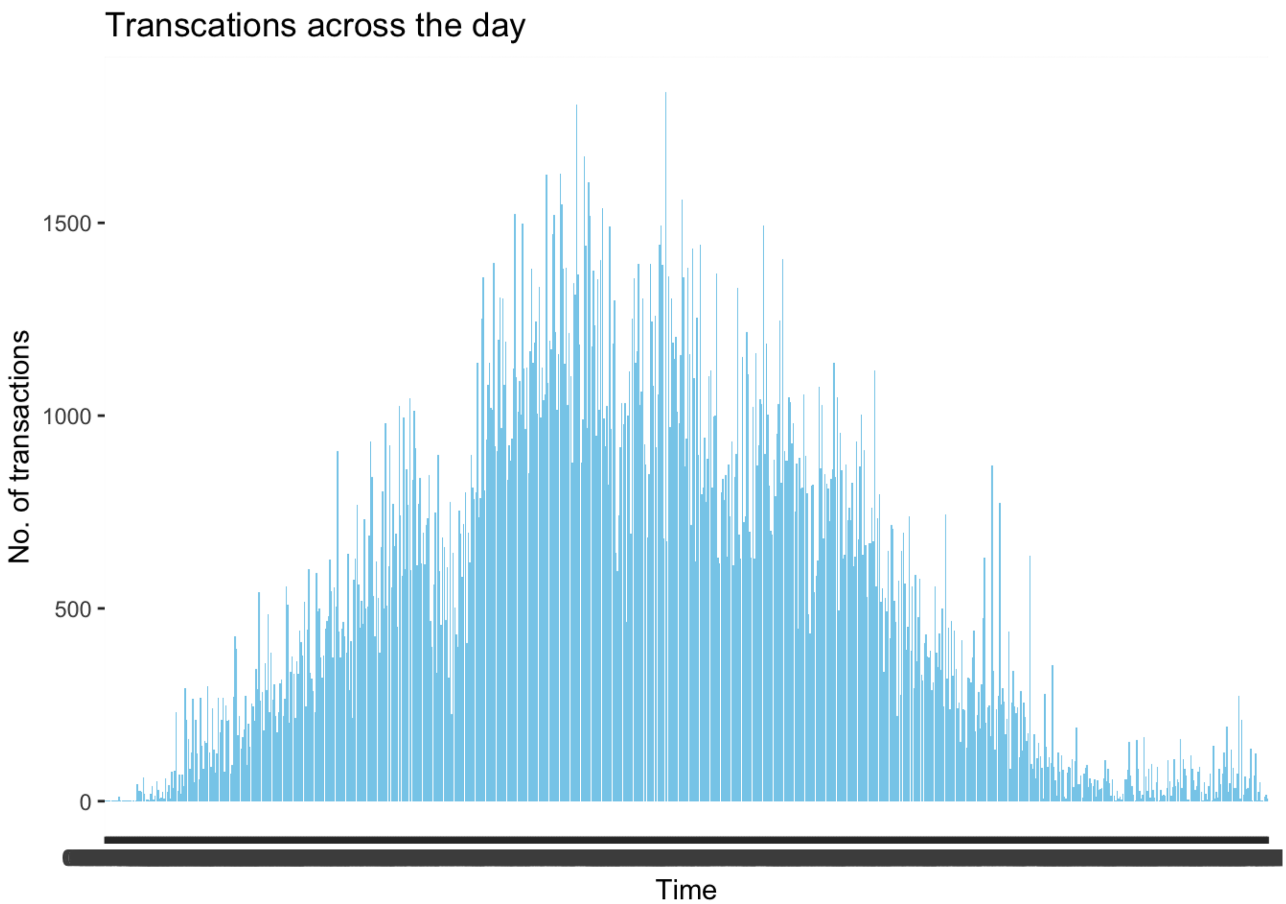
```
# Description per weekday
retail %>%
  mutate(WeekDay=as.factor(weekdays(as.Date(Date)))) %>%
  group_by(WeekDay) %>%
  dplyr::summarize(Description=n_distinct(Description)) %>%
  ggplot(aes(x=WeekDay, y=Description)) +
  geom_bar(stat="identity", fill="dodgerblue", show.legend=FALSE) +
  geom_label(aes(label=Description, y =1, fontface = 'bold')) +
  labs(title="Description per weekday") +
  scale_x_discrete(limits=c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday"
  , "Saturday", "Sunday")) +
  theme_fivethirtyeight()+
  coord_flip()
```



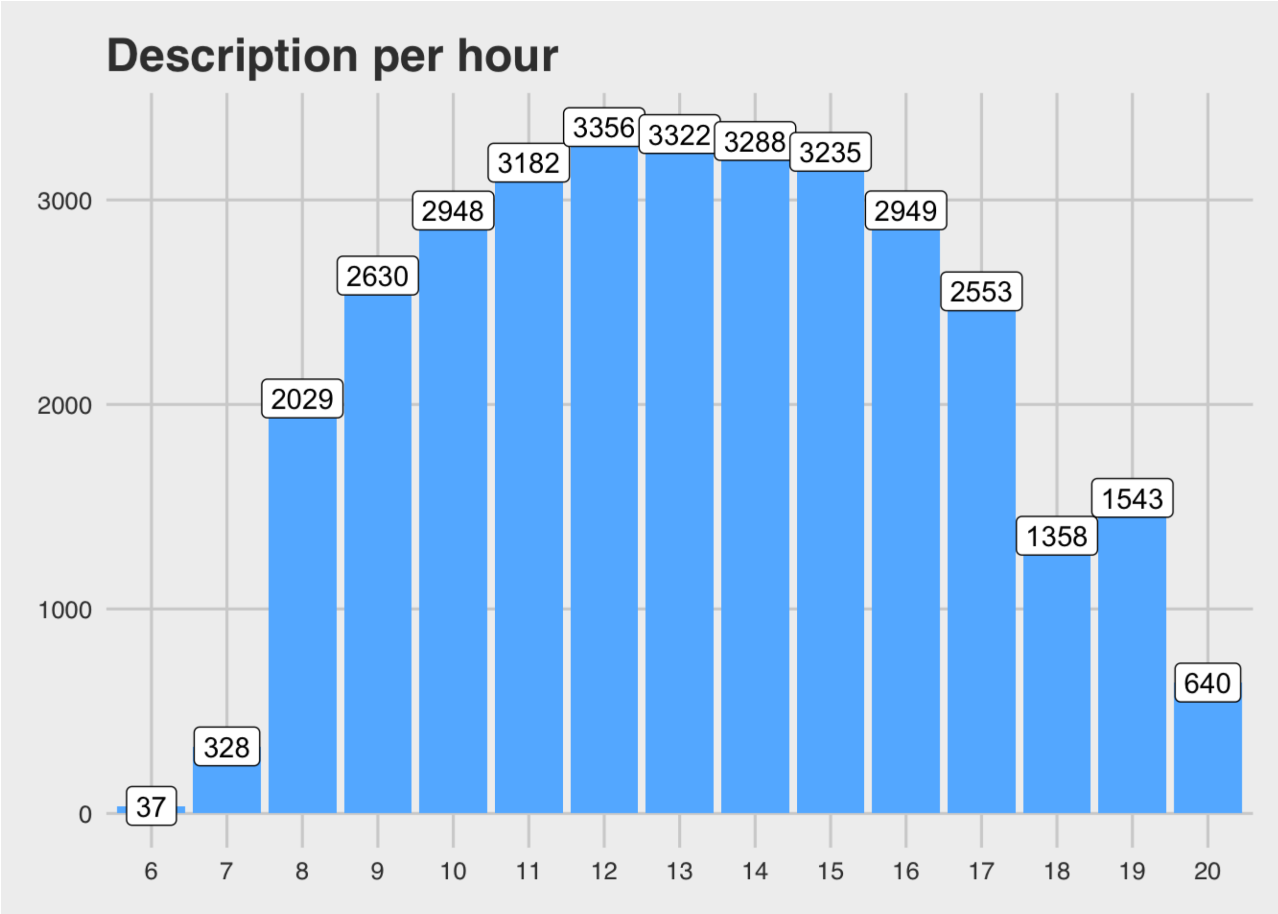

```
parsed <- parse_date_time(retail$InvoiceDate, orders = "ymd HMS")
retail$date <- as.character(as_date(parsed))
retail$time <- format(parsed, "%H:%M:%S")
```

16 Transactions per hour

```
ggplot(retail,aes(x=time))+
  geom_bar(fill="skyblue")+
  ggtitle("Transcations across the day")+
  xlab("Time")+
  ylab("No. of transactions")
```



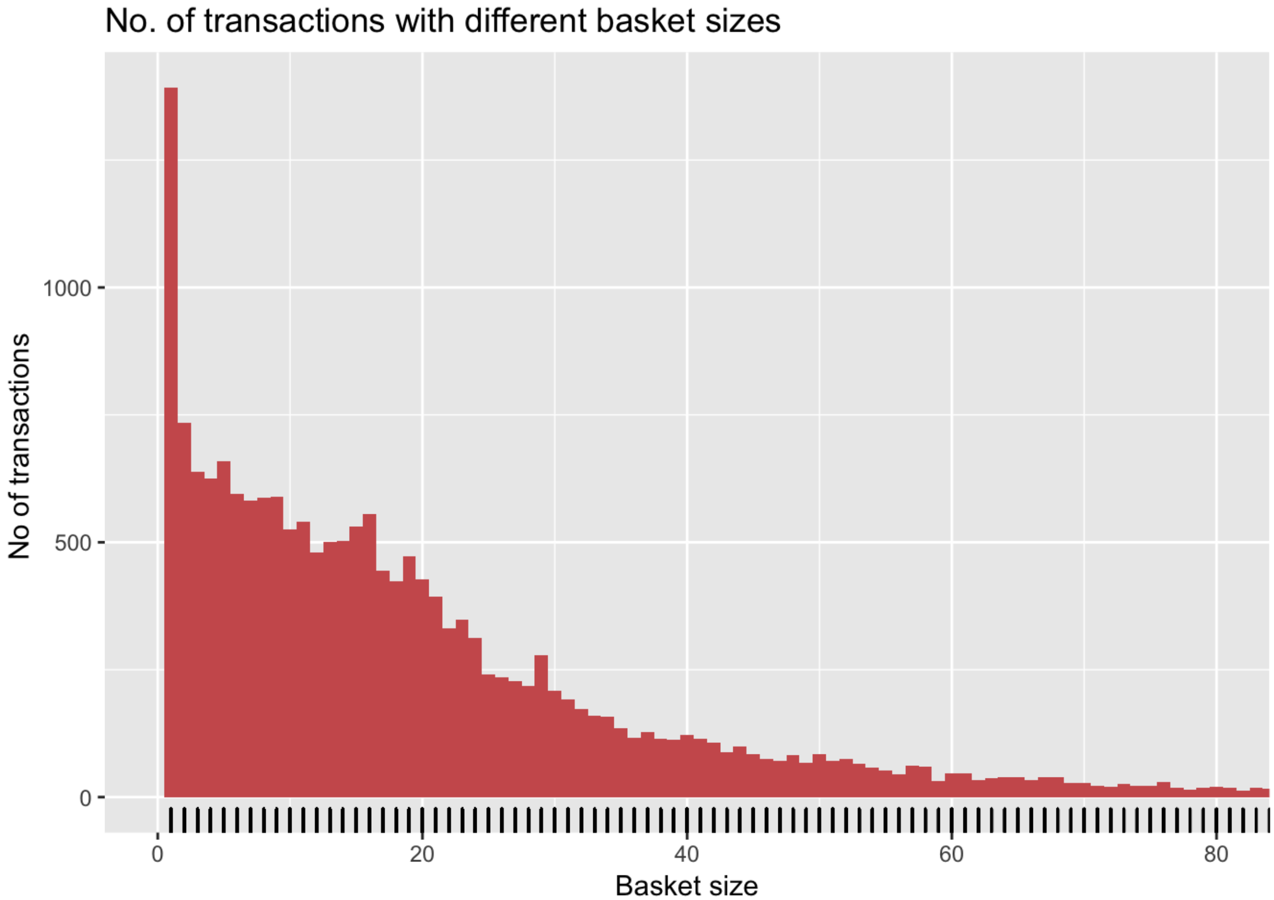

```
# Transactions per hour
retail %>%
  mutate(Hour=as.factor(hour(hms(time)))) %>%
  group_by(Hour) %>%
  dplyr::summarize(Description=n_distinct(Description)) %>%
  ggplot(aes(x=Hour, y=Description)) +
  geom_bar(stat="identity", fill="steelblue1", show.legend=FALSE) +
  geom_label(aes(label=Description)) +
  labs(title="Description per hour") +
  theme_fivethirtyeight()
```



17 No. of transactions with different basket sizes

```
retail$Country<-as.factor(retail$Country)
#retail$Time<-as.factor(retail$Time)
retail$month<-format(retail$Date,"%m")
```

```
items<-retail %>%
  dplyr::group_by(InvoiceNo) %>%
  dplyr::summarise(total=n())

ggplot(items,aes(x=total))+
  geom_histogram(fill="indianred", binwidth = 1)+
  geom_rug()+
  coord_cartesian(xlim=c(0,80))+
  ggtitle("No. of transactions with different basket sizes")+
  xlab("Basket size")+
  ylab("No of transactions ")
```




```
retail_sorted <- retail[order(retail$CustomerID),]
library(plyr)
itemList <- ddply(retail,c("CustomerID","Date"),
                  function(df1)paste(df1$Description,
                                     collapse = ","))

itemList$CustomerID <- NULL
itemList$Date <- NULL
colnames(itemList) <- c("items")
```

```
write.csv(itemList,"market_basket.csv", quote = FALSE, row.names = TRUE)
```

18 Overall quick Snapshot

We Started these projects with question What does the Marketer want? Followed by intrdoucing MBA model, Association Rule Minning. Then we define the key terminology and how can we find out if there is any strong relationship between the variables by looking

- Higher Confidence Value
- Lift Ratio > 1
- Should Exceed Minimum Support and Minimum Confidence.
- 3 Key Terms to take away : Support, confidence, Lift

The first step in order to create a set of association rules is to determine the optimal thresholds for support and confidence. If we set these values too low, then the algorithm will take longer to execute and we will get a lot of rules (most of them will not be useful). We can try different values of support and confidence and see graphically how many rules are generated for each combination.

As we can see, Saturday is the bussiness is closed as we don’t have any transcation day. Rest of the day it does do averge business. The business pickups around 10 AM to 4 PM.There’s not much to discuss with this visualization. The results are logical and expected.