1. Which of the following is a common use of unsupervised clustering?

a) Detect outliers b) Determine a best set of projection for supervised learning c) Evaluate the likely performance of a supervised learner model d) Determine if meaningful relationships can be found in a dataset e) All of the above

**Ans.** e

2. Which statement is true about the K-Means algorithm?

a) All attribute values must be categorical b) The output attribute must be categorical c) Attribute values may be either categorical or numeric d) All attributes must be numeric

**Ans.** d

3. Amongst below data transformation technique which works well when minimum and maximum values for a real-valued attribute are known.

a) min-max normalization b) decimal scaling c) z-score normalization d) logarithmic normalization

**Ans.** a

4. This technique uses mean and standard deviation scores to transform real-valued attributes.

a) decimal scaling b) min-max normalization c) z-score normalization d) logarithmic normalization

**Ans.** c

5. This unsupervised clustering algorithm terminates when mean values computed for the current iteration of the algorithm are identical to the computed mean values for the previous iteration.

a) agglomerative clustering b) conceptual clustering c) K-Means clustering d) expectation maximization

**Ans.** c

6. What is the minimum no. of variables/features required to perform clustering?

a) 0 b) 1 c) 2 d) 3

**Ans.** b

7. Which of the following algorithm is most sensitive to outliers?

a) K-means clustering algorithm b) K-medians clustering

algorithm c) K-modes clustering algorithm d) K-medoids clustering algorithm

**Ans.** a

8. The most popularly used dimensionality reduction algorithm is Principal Component Analysis (PCA).

1. PCA is an unsupervised method 2. It searches for the directions that data have the largest variance 3. Maximum number of principal components <= number of features 4. All principal components are orthogonal to each other Which is above is true.

A. 1 and 2

B. 1 and 3

C. 2 and 3

D. 1, 2 and 3

E. 1,2 and 4

F. All of the above

**Ans.** f

Answer the following using TRUE /FALSE

9. Given historical weather records, can we predict if tomorrow's weather will be sunny or rainy using K-means.

**Ans.** FALSE

10.Given a set of news articles from many different websites, using k-means can you find out what topics are the main topics covered.

**Ans.** TRUE

11.Dimensionality reduction algorithms are one of the possible ways to reduce the computation time required to build a model.

**Ans.** TRUE

12.PCA can be used for projecting and visualizing data in lower dimensions.

**Ans.** TRUE

Q13 => 6 mark

13.Point out pros and cons (at least one) for the following unsupervised algorithms

a) K-Means Clustering

|  |  |
| --- | --- |
| Pros | Cons |
| It is fast in finding the clusters.  It can detect outliers in multiple dimensions. | It suffers from multicollinearity.  It is very sensitive to more/new data. |
| It is simple a flexible. | It gives unstable solutions, depends on initialization. |
|  | It is mandatory to give number of clusters, which if not chosen properly, then it may produce poor clusters. |

b) Scatter Plots

|  |  |
| --- | --- |
| Pros | Cons |
| They are easy to draw. | They are helpful to measure the precise extent of correlation. |
| They are easily understood and interpreted. | It doesn’t measure the relationship between the variables. It is only expression of quantitative change. |

c) Principal Components Analysis

|  |  |
| --- | --- |
| Pros | Cons |
| Very helpful for reduction of data dimensionality. | It fails when the assumption of linearity and orthogonality fails. |
| It helps in better visualization of data. | These are cases when PCA fails. |
| It remove noise in feature. |  |

14. MARKET BASKET ANALYSIS: The dataset called “Online Retail” from UCI Machine Learning repository contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered online retailer. Perform market basket analysis in python/R with your preference of tool to obtain following results.

• What time do people often purchase online? [1 Mark]

Ans. Around 11:00 am to 2:00 am, people purchase online.

• How many items each customer buy? [1 Mark]

Ans. Mostly in 15 items each customer buy.

• Top 10 best sellers [1 Mark]

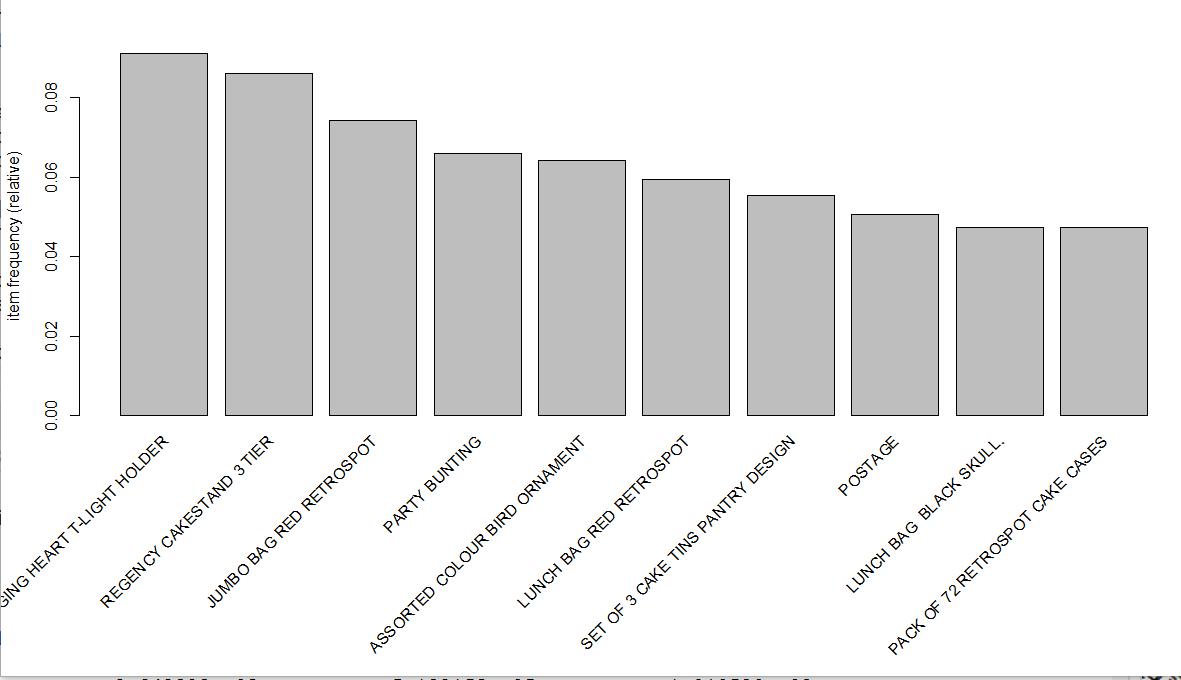


Fig: Showing Top 10 items from the company

• Share your insights which can help retailers to increase his profits and few association rules [4 Marks]

Ans.

I read the .csv file and read it.

transactions as itemMatrix in sparse format with

19297 rows (elements/itemsets/transactions) and

27165 columns (items) and a density of 0.0006701659

most frequent items:

WHITE HANGING HEART T-LIGHT HOLDER REGENCY CAKESTAND 3 TIER JUMBO BAG RED RETROSPOT

1758 1660 1434

PARTY BUNTING ASSORTED COLOUR BIRD ORNAMENT (Other)

1271 1237 343943

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

1 2263 1189 851 768 725 662 618 597 582 554 572 506 487 508 504 503 449 413 477 420 383

23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44

304 313 270 237 253 223 204 222 216 171 147 138 147 130 111 116 89 104 96 92 85 94

45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66

61 67 73 67 64 52 49 59 50 41 53 50 35 24 40 35 29 27 23 21 21 17

67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88

27 31 24 16 24 18 19 18 13 14 17 14 7 9 18 17 11 10 8 13 10 14

89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110

6 7 9 6 7 8 5 4 5 5 3 3 3 4 5 5 2 3 3 7 4 6

111 112 113 114 115 116 117 118 119 120 121 122 123 124 126 127 128 132 133 134 135 140

3 4 1 2 2 1 3 4 3 1 2 1 3 2 4 1 1 1 1 3 1 1

141 142 143 144 146 147 148 150 151 155 158 162 167 169 172 178 179 181 199 200 203 205

1 1 2 1 1 3 1 1 1 2 2 1 1 1 2 1 1 1 1 1 1 1

206 210 230 237 250 251 287 322 402 421

1 1 1 1 1 1 1 1 1 1

Min. 1st Qu. Median Mean 3rd Qu. Max.

1.00 5.00 13.00 18.21 24.00 421.00

includes extended item information - examples:

labels

1 1

2 1 HANGER

3 10

There are 19297 rows and 27165 columns in the dataset.

I build an Apriori following models based on different parameters:

1. Using Apriori function,a sparse matrix object is created with specific support and confidence values. It means that the by default Apriori algorithm runs for 10% support and 80% confidence.

There 568 rules, so these are the rules that

satisfies the condition that support should be 0.006 and confidence = 0.6(60%).

transrules <- apriori(trans, parameter = list(support = 0.006, confidence = 0.25))

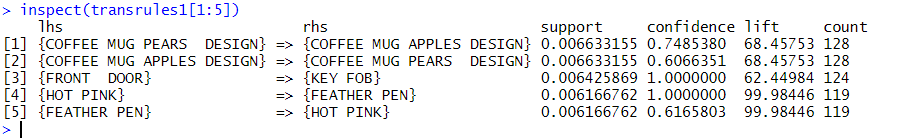


Fig: inspect of transition rule

From the figure we can say that:

* 74.8% customers coffee mug pears design also buy coffee mug apples design.
* 100% people buy front door , also buy key fob and hot pink with feather pen.

1. There are many rules in this model.

transrules <- apriori(trans, parameter = list(support = 0.006, confidence = 0.25, minlen = 2))

1. There are many rules in this model.

transrules <- apriori(trans, parameter = list(support = 0.006, confidence = 0.25, minlen = 3))

Conclusion:

Based on the above model, we can find out what itemsets are being sold with other itemsets and what are not selling in the store. So, these models are helpful to decide whether owner should two or three items closure or give recommendations to the customers based on the buying habit/ pattern observed from these model.

Each model gives different insights about the likelihood of item being sold together or not. So, support, confidence and lift are all are measures to decide whether there is an item which is very popular among customers or not.

library(tidyverse)

library(readxl)

library(knitr)

library(ggplot2)

library(lubridate)

library(arules)

library(arulesViz)

library(plyr)

getwd()

df = read\_excel('Online Retail.xlsx')

View(df)

df = df[complete.cases(df),]

df = df %>% mutate(Description = as.factor(Description))

df = df %>% mutate(Country = as.factor(Country))

df$Date <- as.Date(df$InvoiceDate)

df$Time <- format(df$InvoiceDate,"%H:%M:%S")

df$InvoiceNo <- as.numeric(as.character(df$InvoiceNo))

summary(df)

head(df)

df$Time <- as.factor(df$Time)

tm <- hms(as.character(df$Time))

df$Time = hour(tm)

df %>%

ggplot(aes(x=Time)) +

geom\_density(stat="count",fill="green")

detach("package:plyr", unload=TRUE)

df %>%

group\_by(InvoiceNo) %>%

summarize(n\_items = mean(Quantity)) %>%

ggplot(aes(x=n\_items))+

geom\_histogram(fill="indianred", bins = 100000) +

geom\_rug()+

coord\_cartesian(xlim=c(0,80))

tmp <- df %>%

group\_by(StockCode, Description) %>%

summarize(count = n()) %>%

arrange(desc(count))

tmp <- head(tmp, n=10)

tmp

tmp %>%

ggplot(aes(x=reorder(Description,count), y=count))+

geom\_bar(stat="identity",fill="indian red")+

coord\_flip()

#Association Rules

df\_sorted <- df[order(df$CustomerID),]

library(plyr)

itemList <- ddply(df,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)

trans = read.transactions("market\_basket.csv",format = 'basket',sep=',')

inspect(trans[1:10])

itemFrequency(trans[,1:3])

itemFrequencyPlot(trans, support = 0.1)

itemFrequencyPlot(trans, topN = 20)

image(trans[1:5])

image(sample(trans, 100))

transrules1 <- apriori(trans)

transrules <- apriori(

trans, parameter = list(

support = 0.006, confidence = 0.25,

minlen = 1))

transrules1

summary(transrules1)

#install.packages("RColorBrewer")

#install.packages("bnlearn")

#library("RColorBrewer")

#library(bnlearn)

library(arulesViz)

arules::itemFrequencyPlot(trans,

topN=20,

col=brewer.pal(8,'Pastel2'),

main='Relative Item Frequency Plot',

type="relative",

ylab="Item Frequency (Relative)")

subrules2 <- head(sort(transrules, by="confidence"),20)

ig <- plot( subrules2, method="graph", control=list(type="items") )

subrules <- subset(transrules, lift>2.5)

subrules

plot(subrules, method="matrix", measure=c("lift", "confidence"))

subrules2 <- head(transrules, n = 10, by = "lift")

plot(subrules2, method="graph", control=list(

layout=igraph::with\_graphopt(spring.const=5, mass=50)))

plot.igraph(subrules2, method = "paracoord")

plot(subrules2, method = "paracoord", control = list(reorder = TRUE))

plot(subrules2, method = "graph", control = list(verbose = TRUE))

#Scatter plot

plot(transrules, engine = "htmlwidget")

#Two key plot

plot(transrules, method = "two-key plot")

#Graph

saveAsGraph(head(subrules2, n = 100, by = "lift"), file = "rules.graphml")

plot(subrules2, method="graph",

nodeCol = grey.colors(10), edgeCol = grey(.7), alpha = 1)

plot(subrules2, method="graph",

nodeCol = grey.colors(10), edgeCol = grey(.7), alpha = 1)

plot(subrules2, method="graph",

layout=igraph::with\_graphopt(spring.const=5, mass=50))

#Graph in circle, interactive

plot(subrules2, method="graph", engine="htmlwidget",

igraphLayout = "layout\_in\_circle")

#Parallel coordinate plot

plot(subrules2, method="paracoord", control = list(reorder=TRUE))

#Itemsets

itemsets <- eclat(trans, parameter = list(support = 0.02, minlen=2))

plot(itemsets)

plot(itemsets, method="graph")

plot(itemsets, method="paracoord", alpha=.5, reorder=TRUE)

#More qualities measure for scatterplot

quality(itemsets) <- interestMeasure(itemsets, trans=trans)

head(quality(itemsets))

plot(itemsets, measure=c("support", "allConfidence"), shading="lift")

plot(transrules)

inspect(transrules[1:3])

plot(head(sort(transrules),10), method = "graph", control = list(type ="items"))

plot(head(sort(transrules),10), method = "grouped")

plot(head(sort(transrules),20), method = "matrix", measure = c("lift", "confidence"))

samplerule <- head(sort(transrules, by = "lift"), 1)

inspect(samplerule)

plot(samplerule, method = "doubledecker", data = trans)

inspect(sort(transrules, by = "lift")[1:5])

itemsets = eclat(trans)[1:5]

inspect(head(eclat(trans)))

support(items(itemsets),trans)