Customer Segmentation. Application of Unsupervised Learning Methods for Trend Exploration

by Ketao Li, Kush Halani, Josue Romain, Juan Peña

Abstract Customer segmentation is the process of dividing customers into groups based on common characteristics so companies can market to each group effectively and appropriately.

Background

Without a deep understanding of how a company's best current customers are segmented, a business often lacks the market focus needed to allocate and spend its precious human and capital resources efficiently. Furthermore, a lack of best current customer segment focus can cause diffused go-to-market and product development strategies that hamper a company's ability to fully engage with its target segments. Together, all of those factors can ultimately impede a company's growth.
RFM (recency, frequency, monetary) analysis is a marketing technique used to determine quantitatively which customers are the best ones by examining how recently a customer has purchased (recency), how often they purchase (frequency), and how much the customer spends (monetary).

Objective

The objective of customers segment according to their purchase history, is to turn them into loyal customers by recommending products of their choice.

Apply the Ethical ML framework

1 Data risk awareness

We commit to develop and improve reasonable processes and infrastructure to ensure data and model security are being taken into consideration during the development of machine learning systems. We commit to prepare for security risks through explicit efforts, such as educating relevant personnel, establishing processes around data, and assess implications of ML backdoor.

2 Trust by privacy

We commit to build and communicate processes that protect and handle data with stakeholders that may interact with the system directly and/or indirectly. One key way to establish trust with users and relevant stakeholders is by showing the right process and technologies are in place to protect personal data. We should make explicit effort to understand the potential implications of metadata involved, and whether the metadata can expose unexpected personal information from relevant users or stakeholders. Fortuantely in our dataset there are not sensitive data.There is no explicit personal information.The only feature related to personal information is CustomerID.We should not put any file related to the detailed person to the server.

3 Displacement strategy

We commit to identify and document relevant information so that business change processes can be developed to mitigate the impact towards workers being automated. When planning the rollout of a new technology to automate a process, there are a number of people who's role or at least responsibilities will be automated. If this is not taken into consideration, these people will not have a transition plan and it won't be possible to fully benefit from the time and resources gained from the automation.

We should make sure they are able to raise the relevant concerns when business change or operational transformation plans are being set up, as this would make a significant positive impact in the rollout of the technology.

4 Bias evaluation

As developer it is important to obtain an understanding of how potential biases might arise. Once the different sub-categories for bias are identified it's possible to evaluate the results on a breakdown based on precision, recall and accuracy for each of the potential inference groups. We checked through our system, there is no bias in our system.

Data Analysis

Typically e-commerce datasets are proprietary and consequently hard to find among publicly available data. However, The UCI Machine Learning Repository has made this dataset containing actual transactions from 2010 and 2011.The data set used for this research contains 540k of transaction from UK retailer. The data has been sourced from Kaggle.

Data Dictionary

Column Name	Column Description
InvoiceNo	Unique ID to identify each Invoice
StockCode	Unique ID for each item in stock
Description	A short description for each item
Quantity	Number of items bought
InvoiceDate	Invoice Date
UnitPrice	The price of each item
CustomerID	Unique ID for each custumer
Country	The country were the custumer lives

Data Exploration

Firstly we are going to load and examine content and statistics of the data set

```
data = read.csv("../data/data.csv", header = T,
                na.strings = c("NA","","#NA"),sep=",")
```

Table 2: Online Retail Dataset Summary

No	Variable	Stats / Values	Freqs (% of Valid)	Missing
1	InvoiceNo [factor]	1. 536365 2. 536366 3. 536367 [25897 others]	7 (0.0%) 2 (0.0%) 12 (0.0%) 541888 (100.0%)	0 (0%)
2	StockCode [factor]	1. 10002 2. 10080 3. 10120 [4067 others]	73 (0.0%) 24 (0.0%) 30 (0.0%) 541782 (100.0%)	0 (0%)
3	Description [factor]	1. ·4 PURPLE FLOCK DINNER CA 2. ·50'S CHRISTMAS GIFT BAG 3. ·DOLLY GIRL BEAKER [4220 others]	41 (0.0%) 130 (0.0%) 181 (0.0%) 540103 (99.9%)	1454 (0.27%)
4	Quantity [integer]	Mean (sd) : 9.6 (218.1) min < med < max: -80995 < 3 < 80995 IQR (CV) : 9 (22.8)	722 distinct values	0 (0%)
5	InvoiceDate [factor]	1. 1/10/2011 10:04 2. 1/10/2011 10:07 3. 1/10/2011 10:08 [23257 others]	1 (0.0%) 1 (0.0%) 1 (0.0%) 541906 (100.0%)	0 (0%)
6	UnitPrice [numeric]	Mean (sd) : 4.6 (96.8) min < med < max: -11062.1 < 2.1 < 38970 IQR (CV) : 2.9 (21)	1630 distinct values	0 (0%)
7	CustomerID [integer]	Mean (sd) : 15287.7 (1713.6) min < med < max: 12346 < 15152 < 18287 IQR (CV) : 2838 (0.1)	4372 distinct values	135080 (24.93%)

No	Variable	Stats / Values	Freqs (% of Valid)	Missing
8	Country [factor]	1. Australia 2. Austria 3. Bahrain [35 others]	1259 (0.2%) 401 (0.1%) 19 (0.0%) 540230 (99.7%)	0 (0%)

From the above summary, we can find that there are some negative values for Quantity and UnitPrice.These values don't make sense, so we'll delete them directly.

```
customerData <- data %>%
  mutate(Quantity = replace(Quantity, Quantity<=0, NA),
         UnitPrice = replace(UnitPrice, UnitPrice<=0, NA))


customerData = customerData %>%filter(complete.cases(.))
```

##Missing data

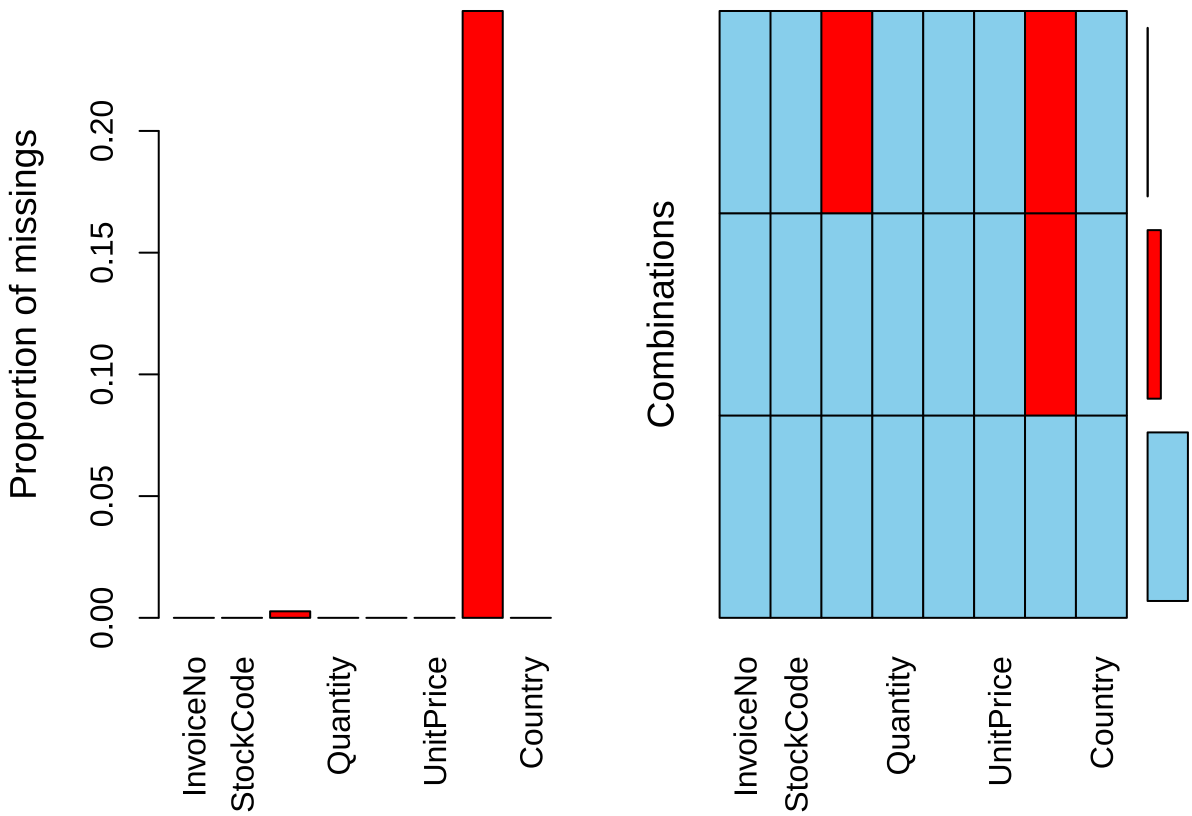


Figure 1: Missing data

```
summary(a)


 Missings per variable:
    Variable  Count
   InvoiceNo      0
   StockCode      0
 Description   1454
    Quantity      0
 InvoiceDate      0
   UnitPrice      0
  CustomerID 135080
     Country      0
```

```
Missings in combinations of variables:
   Combinations  Count    Percent
0:0:0:0:0:0:0:0 406829 75.0733057
0:0:0:0:0:0:1:0 133626 24.6583836
0:0:1:0:0:0:1:0   1454  0.2683107
```

There are some missing data for CustomerID and Desciption, we just remove them directly considering we have enough data.

Table 3: Online Retail Dataset Summary

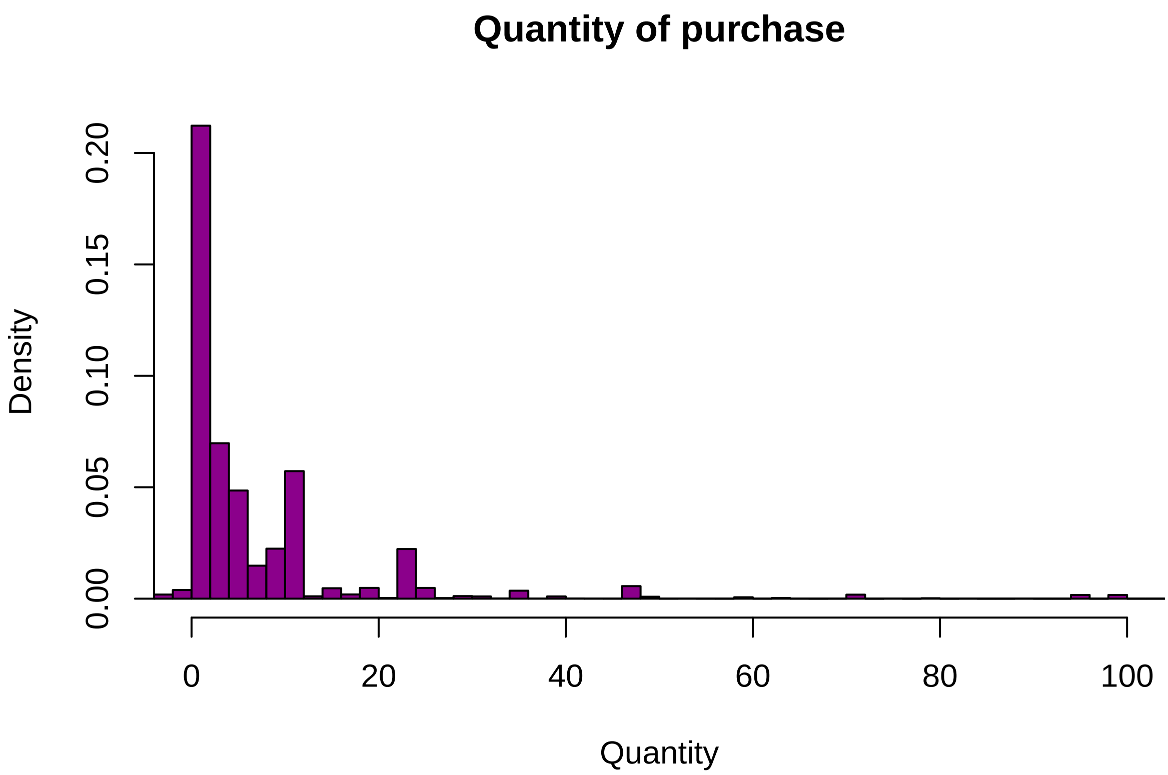
No	Variable	Stats / Values	Freqs (% of Valid)	Missing
1	InvoiceNo [factor]	1. 536365 2. 536366 3. 536367 [25897 others]	7 (0.0%) 2 (0.0%) 12 (0.0%) 397863 (100.0%)	0 (0%)
2	StockCode [factor]	1. 10002 2. 10080 3. 10120 [4067 others]	49 (0.0%) 21 (0.0%) 30 (0.0%) 397784 (100.0%)	0 (0%)
3	Description [factor]	1. ·4 PURPLE FLOCK DINNER CA 2. ·50'S CHRISTMAS GIFT BAG 3. ·DOLLY GIRL BEAKER [4220 others]	39 (0.0%) 109 (0.0%) 138 (0.0%) 397598 (99.9%)	0 (0%)
4	Quantity [integer]	Mean (sd) : 13 (179.3) min < med < max: 1 < 6 < 80995 IQR (CV) : 10 (13.8)	301 distinct values	0 (0%)
5	InvoiceDate [factor]	1. 1/10/2011 10:04 2. 1/10/2011 10:07 3. 1/10/2011 10:08 [23257 others]	0 (0.0%) 0 (0.0%) 0 (0.0%) 397884 (100.0%)	0 (0%)
6	UnitPrice [numeric]	Mean (sd) : 3.1 (22.1) min < med < max: 0 < 2 < 8142.8 IQR (CV) : 2.5 (7.1)	440 distinct values	0 (0%)
7	CustomerID [integer]	Mean (sd) : 15294.4 (1713.1) min < med < max: 12346 < 15159 < 18287 IQR (CV) : 2826 (0.1)	4338 distinct values	0 (0%)
8	Country [factor]	1. Australia 2. Austria 3. Bahrain [35 others]	1182 (0.3%) 398 (0.1%) 17 (0.0%) 396287 (99.6%)	0 (0%)

We need do some some data transformation and add one new variant total.

```
customerData <- customerData %>%
  mutate( InvoiceDate=as.Date(InvoiceDate, '%m/%d/%Y %H:%M'),
          CustomerID=as.factor(CustomerID),
          Country = as.character(Country))

customerData <- customerData %>%
  mutate(total = Quantity*UnitPrice)

glimpse(customerData)

Observations: 397,884
Variables: 9
$ InvoiceNo   <fct> 536365, 536365, 536365, 536365, 536365, 536365, 536365,...
$ StockCode   <fct> 85123A, 71053, 84406B, 84029G, 84029E, 22752, 21730, 22...
```

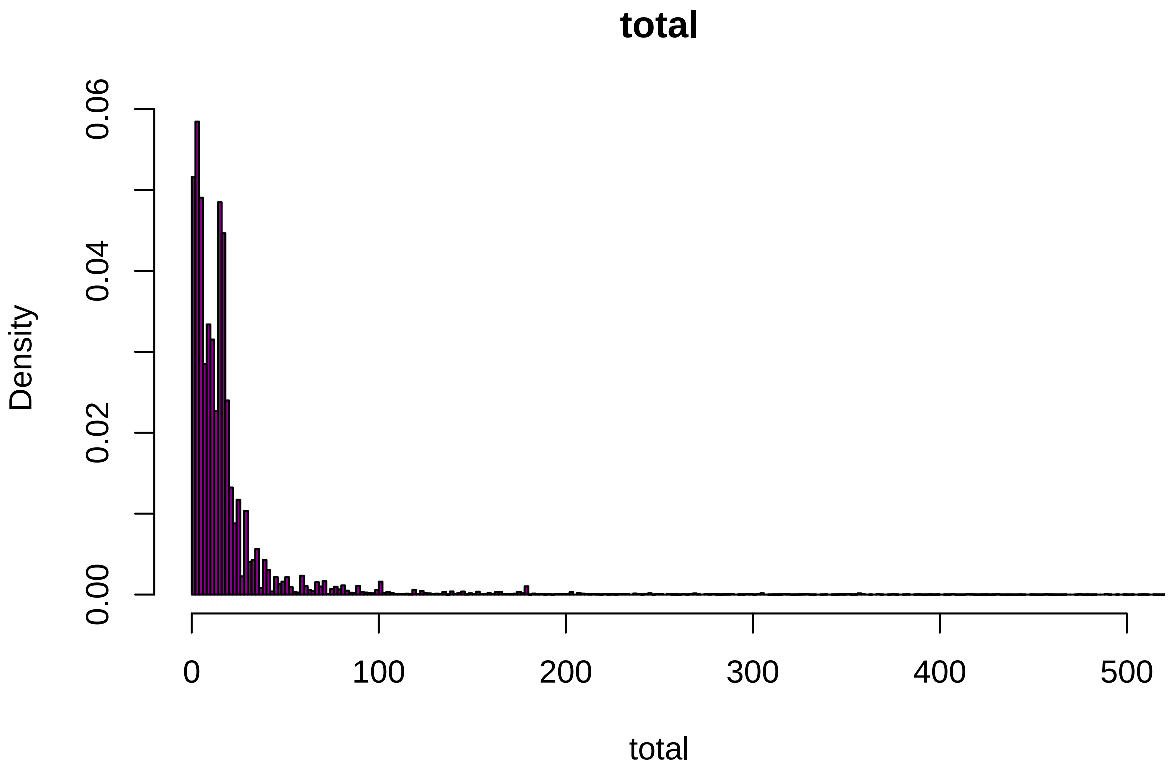
```
$ Description <fct> WHITE HANGING HEART T-LIGHT HOLDER, WHITE METAL LANTERN...
$ Quantity    <int> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2, 3, ...
$ InvoiceDate <date> 2010-12-01, 2010-12-01, 2010-12-01, 2010-12-01, 2010-1...
$ UnitPrice   <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1.85, 1...
$ CustomerID  <fct> 17850, 17850, 17850, 17850, 17850, 17850, 17850, 17850,...
$ Country     <chr> "United Kingdom", "United Kingdom", "United Kingdom", "...
$ total       <dbl> 15.30, 20.34, 22.00, 20.34, 20.34, 15.30, 25.50, 11.10,...
```

```
# histogram with added parameters
hist(data$Quantity,
main="Quantity of purchase",
xlab="Quantity",
breaks=100000,
xlim=c(0,100),
col="darkmagenta",
freq=FALSE
)
```




```
# histogram with added parameters
hist(data$UnitPrice,
main="UnitPrice of purchase",
xlab="UnitPrice",
breaks=100000,
xlim=c(0,100),
col="darkmagenta",
freq=FALSE
)
```


```
# histogram with added parameters
hist(customerData$total,
main="total",
xlab="total",
breaks=100000,
xlim=c(0,500),
col="darkmagenta",
freq=FALSE
)
```



Descriptive Features. *Description* is free-text features that might provide additional insights about the customer shopping. We are going to take a close look at this feature and decide if we could utilize it.

Lets' begin with the *Description*

Figure 2: Most Common Words in Description

From the word cloud, we can get some highly frequently used words such as set,red,box,lunch,blue,box,paper,glass.

Unfortunately *Description* feature does not provide more knowledge to what the others features already supply. Thus it will be dropped.

Country gives information about the country were the customer lives.

```
length(unique(data$Country))
```

```
[1] 38
```

The custumers are from 38 different countries. Lets visualize this.

```
33 codes from your data successfully matched countries in the map
5 codes from your data failed to match with a country code in the map
210 codes from the map weren't represented in your data
```

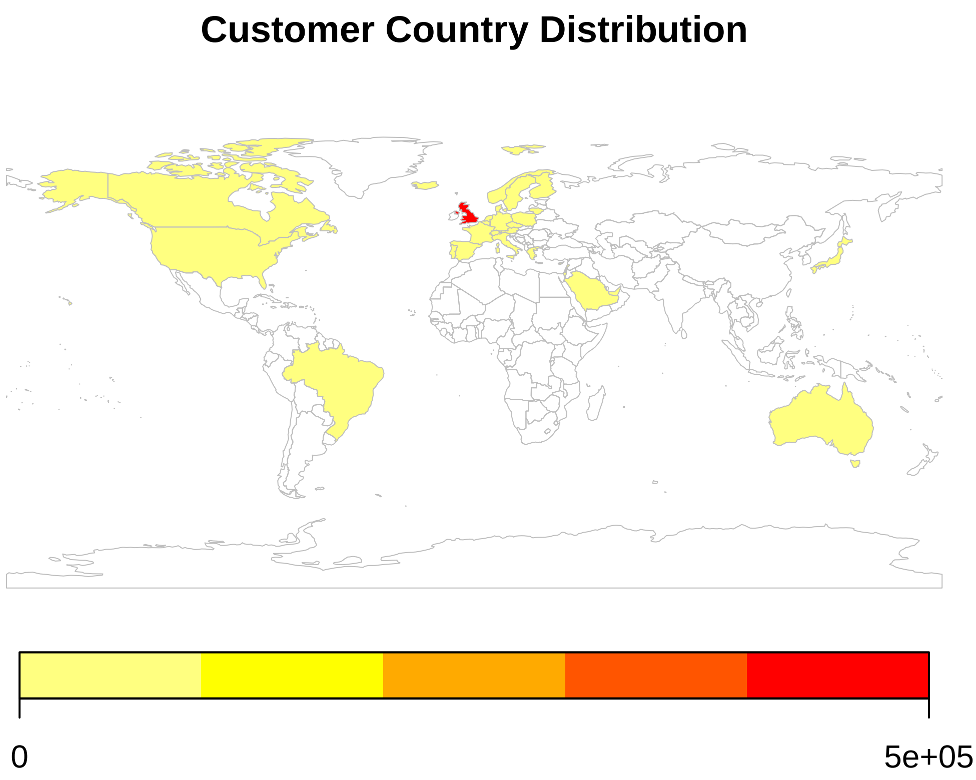


Figure 3: customer country distribution

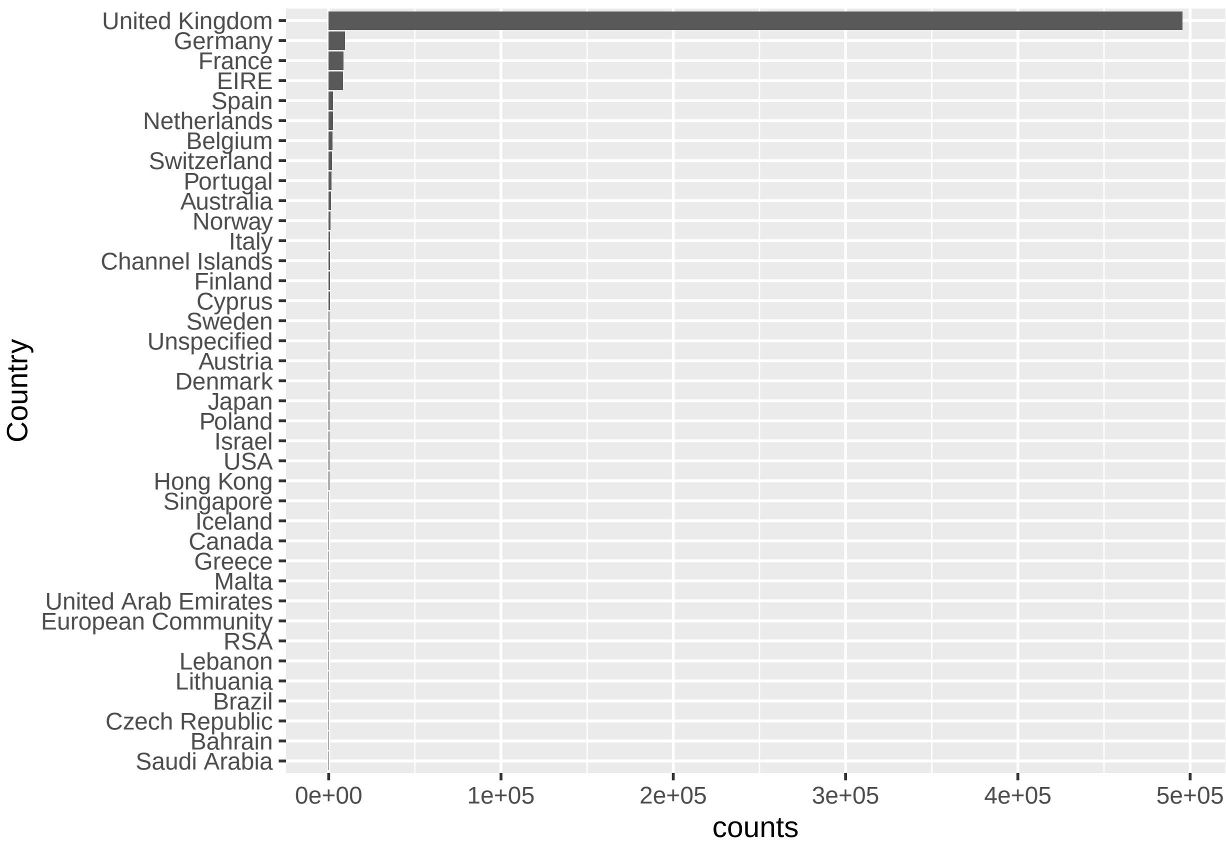


Figure 4: Countries Description

```
data1 <- data
data1$InvoiceDate <- mdy_hm(data$InvoiceDate)


head(data1)

  InvoiceNo StockCode                         Description Quantity
1    536365    85123A  WHITE HANGING HEART T-LIGHT HOLDER        6
2    536365     71053                 WHITE METAL LANTERN        6
3    536365    84406B      CREAM CUPID HEARTS COAT HANGER        8
4    536365    84029G KNITTED UNION FLAG HOT WATER BOTTLE        6
5    536365    84029E      RED WOOLLY HOTTIE WHITE HEART.        6
6    536365     22752        SET 7 BABUSHKA NESTING BOXES        2
          InvoiceDate UnitPrice CustomerID        Country
1 2010-12-01 08:26:00      2.55      17850 United Kingdom
2 2010-12-01 08:26:00      3.39      17850 United Kingdom
3 2010-12-01 08:26:00      2.75      17850 United Kingdom
4 2010-12-01 08:26:00      3.39      17850 United Kingdom
5 2010-12-01 08:26:00      3.39      17850 United Kingdom
6 2010-12-01 08:26:00      7.65      17850 United Kingdom
```

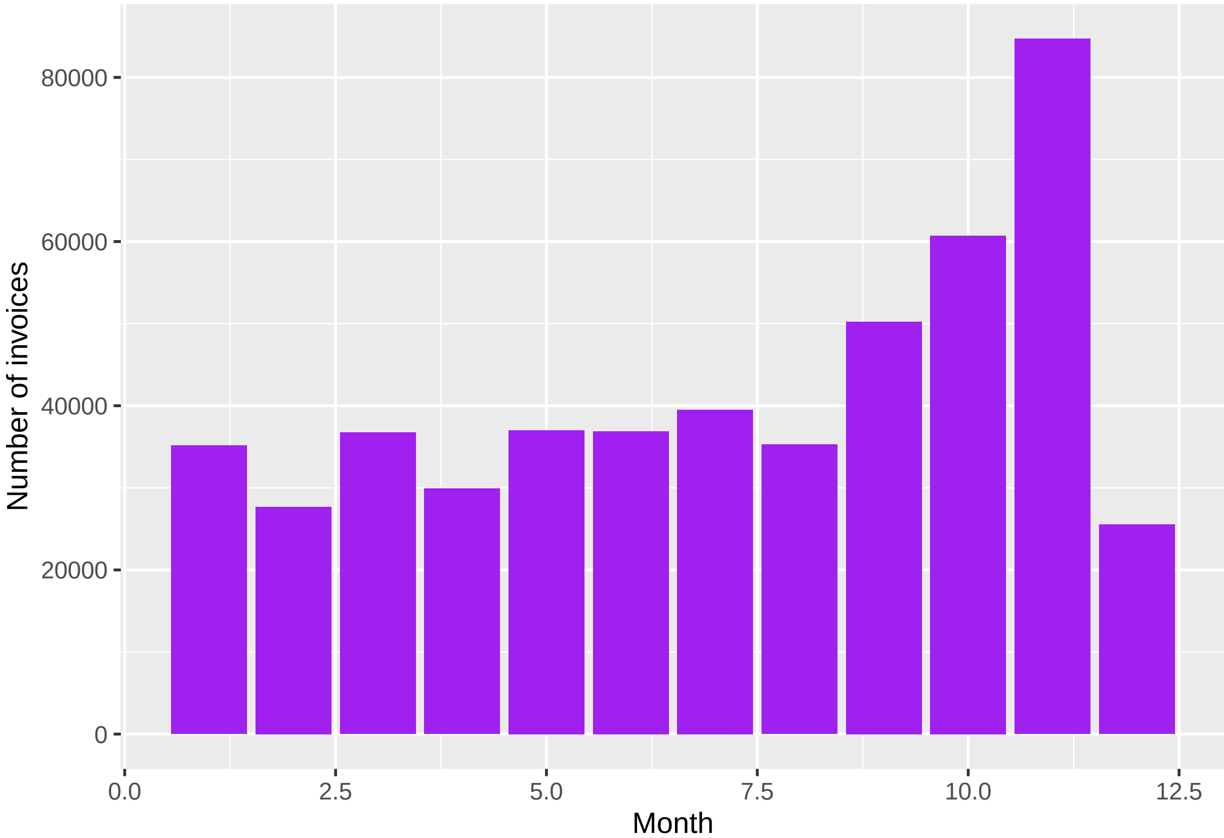
We now have the data transformed into datetime data. From the variable InvoiceDate we can extract the year, month, day and time.

```
data1$InvoiceYear <- year(data1$InvoiceDate)
data1$InvoiceMonth <- month(data1$InvoiceDate)
data1$InvoiceWeekday <- wday(data1$InvoiceDate)
data1$InvoiceHour <- hour(data1$InvoiceDate)
```

Here we have the number of transactions per month for 2011.

```
timedata <- data1 %>%
  filter(InvoiceYear==2011) %>%
```

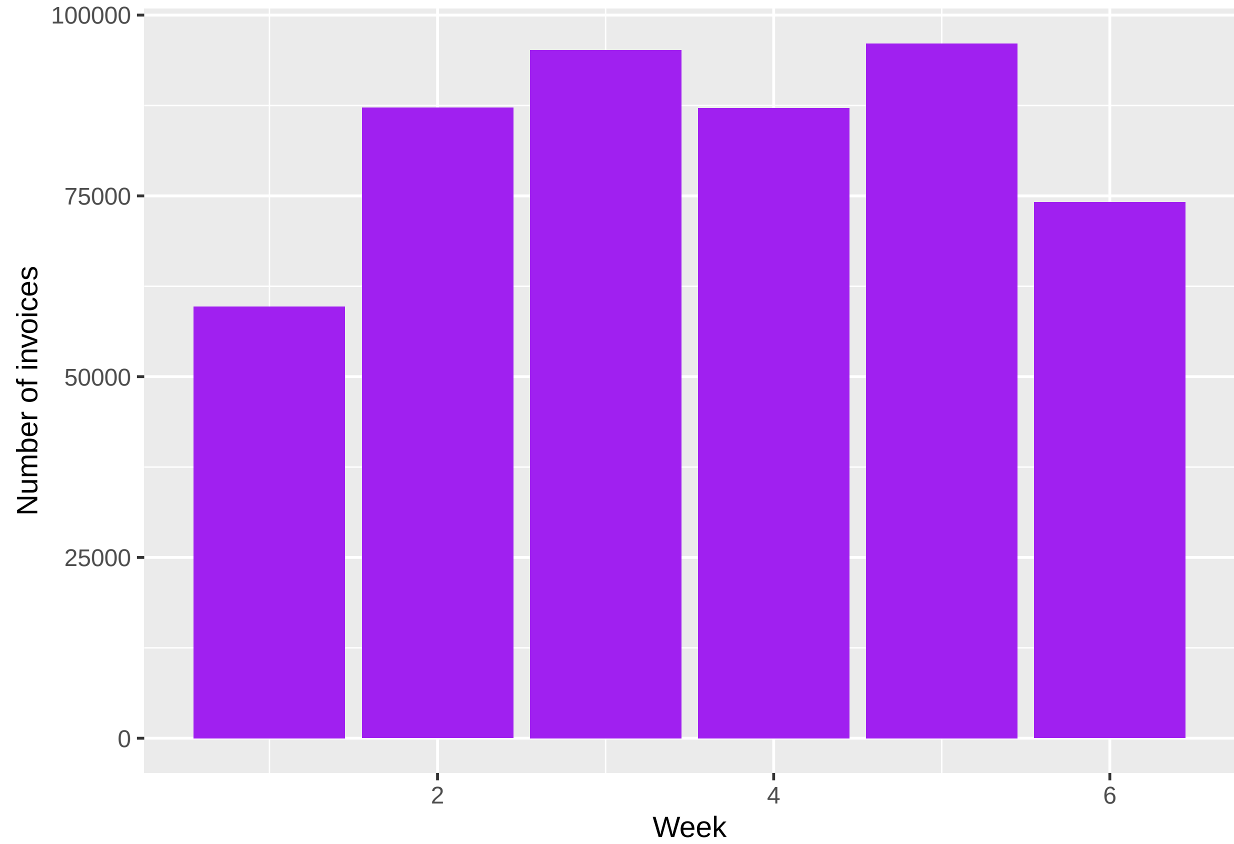
```
  count(InvoiceMonth)  #count the number of invoices per month for 2011

ggplot(timedata, aes(InvoiceMonth, n)) +  #plot the number of invoices per day
  geom_col(fill = "purple") +
  labs(x="Month", y="Number of invoices")
```



It seems that the number of transactions is rising from September and the highest in November. In december the lowest number of transactions is performed.

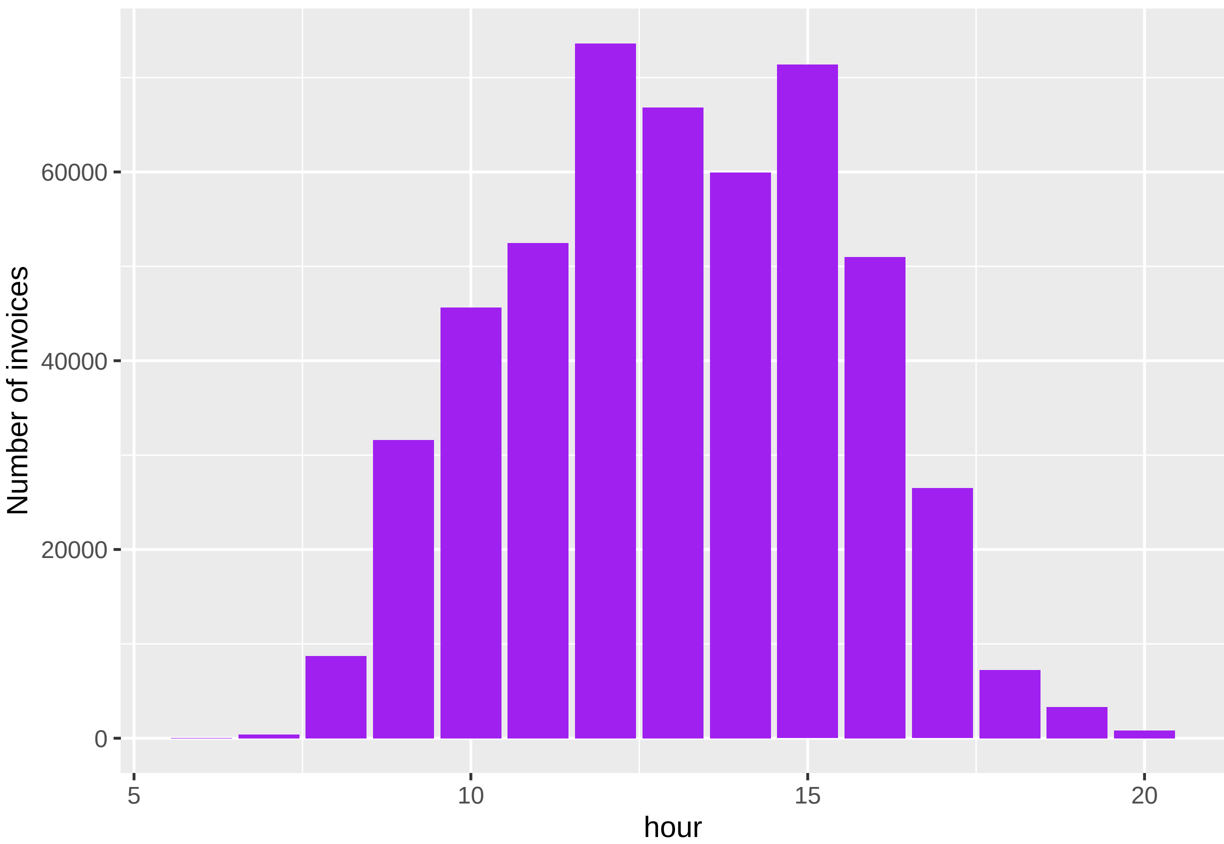
Lets explore which days are the most busy ones

```
timedata <- data1 %>%
  filter(InvoiceYear==2011) %>%
  count(InvoiceWeekday)

ggplot(timedata, aes(InvoiceWeekday, n)) +  #plot the number of invoices per day
    geom_col(fill = "purple") +
  labs(x="Week", y="Number of invoices")
```



Most transactions are placed on monday, tuesday, wednesday and thursday.

```
timedata <- data1 %>%
 filter(InvoiceYear==2011) %>%
  count(InvoiceHour)

ggplot(timedata, aes(InvoiceHour, n)) +  #plot the number of invoices per day
    geom_col(fill = "purple") +
  labs(x="hour", y="Number of invoices")
```



Data Preparation

Calculate RFM

To implement the RFM analysis, we need to take steps to get the rfm values:

1. Find the most recent date for each customer ID and calculate the days to the 2012-01-01, to get the recency data.
2. Calculate the quantity of transactions of a customer, to get the frequency data
3. Sum the amount of money a customer spent and divide it by frequency, to get the amount per transaction on average, that is the monetary data.

```
cd_RFM <- customerData %>%
  group_by(CustomerID) %>%
  summarise(recency=as.numeric(as.Date("2012-01-01")-max(InvoiceDate)),
            frequenci=n_distinct(InvoiceNo), monitery= sum(total)/n_distinct(InvoiceNo),
            country = max(Country)

            )

summary(cd_RFM)

head(cd_RFM)
```

```
   CustomerID       recency         frequenci          monitery
 12346  :    1   Min.   : 23.0   Min.   :  1.000   Min.   :    3.45
 12347  :    1   1st Qu.: 40.0   1st Qu.:  1.000   1st Qu.:  178.62
 12348  :    1   Median : 73.0   Median :  2.000   Median :  293.90
 12349  :    1   Mean   :115.1   Mean   :  4.272   Mean   :  419.17
 12350  :    1   3rd Qu.:164.8   3rd Qu.:  5.000   3rd Qu.:  430.11
 12352  :    1   Max.   :396.0   Max.   :209.000   Max.   :84236.25
 (Other):4332
   country
 Length:4338
```

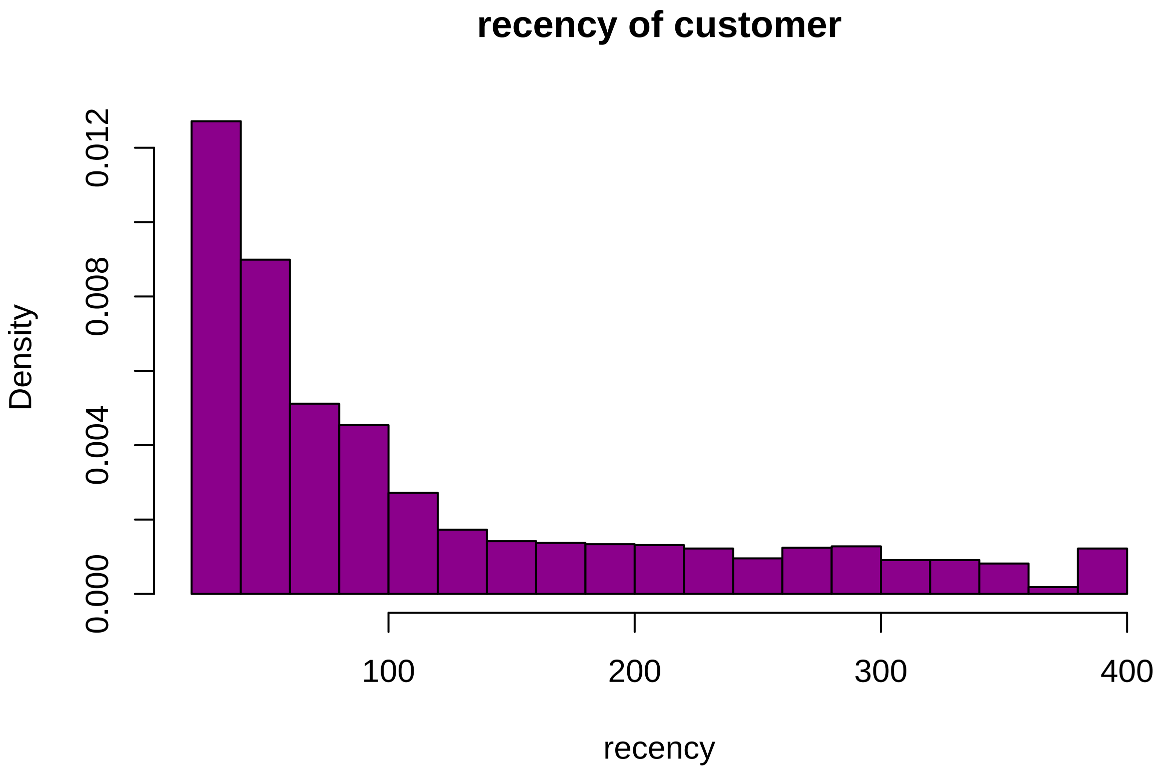
```
 Class :character
 Mode  :character
```

```
# A tibble: 6 x 5
  CustomerID recency frequenci monitery country
  <fct>        <dbl>     <int>    <dbl> <chr>
1 12346          348         1   77184. United Kingdom
2 12347           25         7     616. Iceland
3 12348           98         4     449. Finland
4 12349           41         1    1758. Italy
5 12350          333         1     334. Norway
6 12352           59         8     313. Norway
```

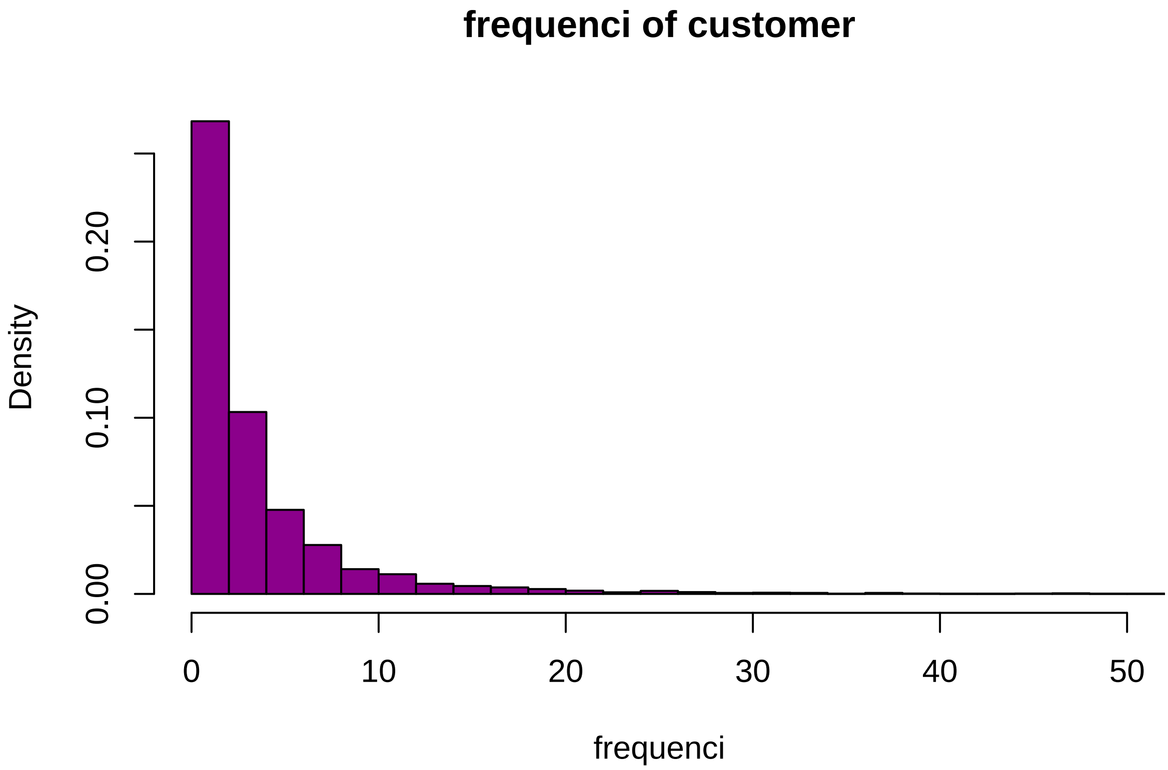
Table 4: Online Retail Dataset Summary

No	Variable	Stats / Values	Freqs (% of Valid)	Missing
1	CustomerID [factor]	1. 12346 2. 12347 3. 12348 [4335 others]	1 (0.0%) 1 (0.0%) 1 (0.0%) 4335 (99.9%)	0 (0%)
2	recency [numeric]	Mean (sd) : 115.1 (100) min < med < max: 23 < 73 < 396 IQR (CV) : 124.8 (0.9)	304 distinct values	0 (0%)
3	frequenci [integer]	Mean (sd) : 4.3 (7.7) min < med < max: 1 < 2 < 209 IQR (CV) : 4 (1.8)	59 distinct values	0 (0%)
4	monitery [numeric]	Mean (sd) : 419.2 (1796.5) min < med < max: 3.5 < 293.9 < 84236.2 IQR (CV) : 251.5 (4.3)	4249 distinct values	0 (0%)
5	country [character]	1. United Kingdom 2. Germany 3. France [34 others]	3920 (90.4%) 94 (2.2%) 87 (2.0%) 237 (5.5%)	0 (0%)

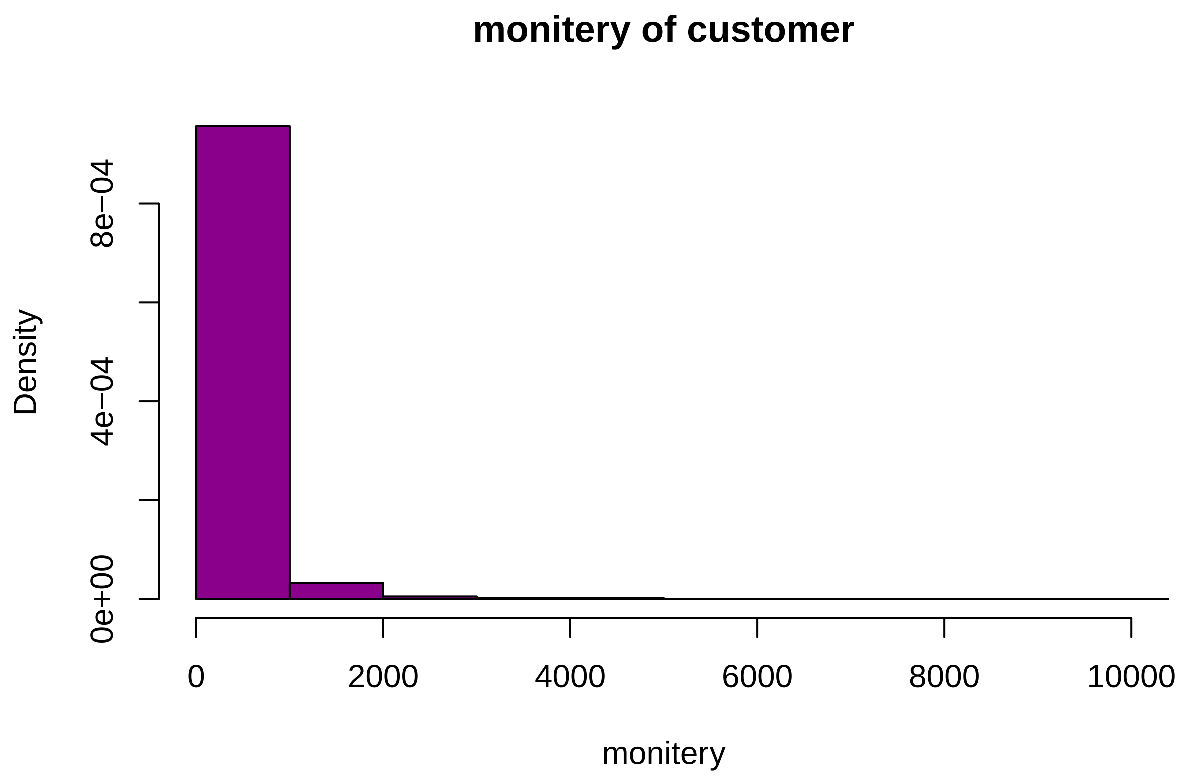
```
# histogram with added parameters
hist(cd_RFM$recency,
main="recency of customer",
xlab="recency",
xlim=c(20,400),
col="darkmagenta",
freq=FALSE
)
```




```
# histogram with added parameters
hist(cd_RFM$frequenci,
main="frequenci of customer",
xlab="frequenci",
breaks=100,
xlim=c(0,50),
col="darkmagenta",
freq=FALSE
)
```

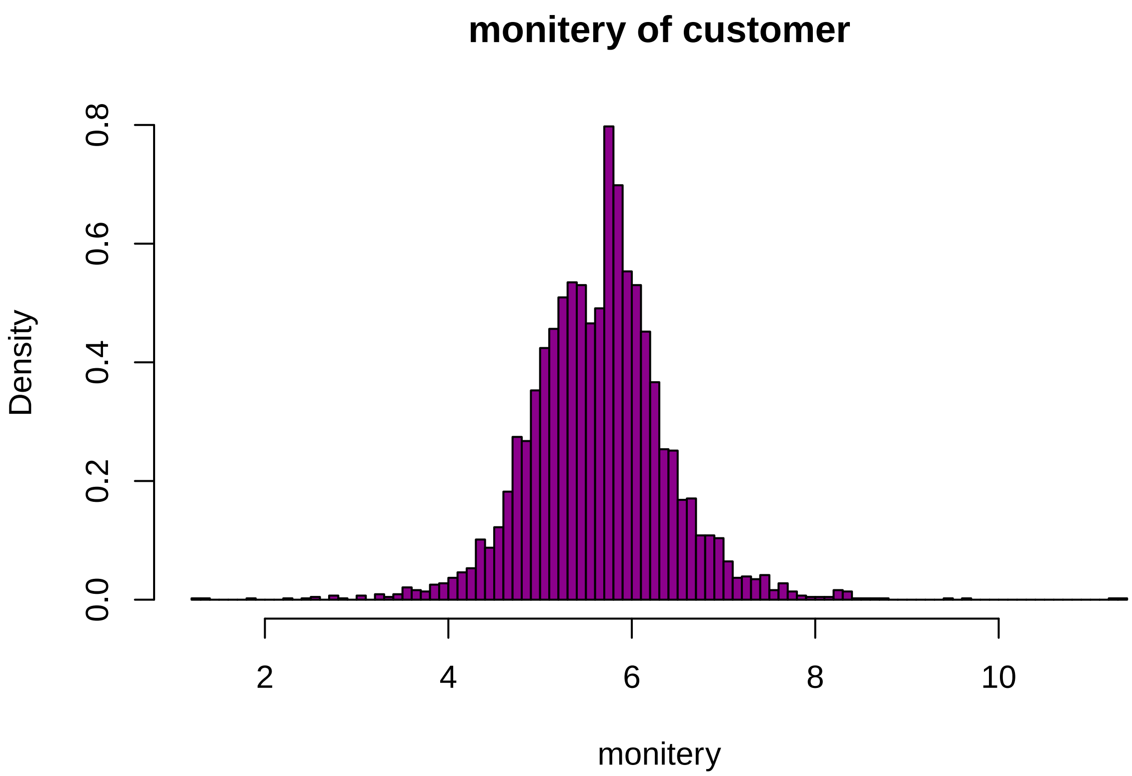



```
# histogram with added parameters
hist(cd_RFM$monitery,
main="monitery of customer",
xlab="monitery",
breaks=100,
xlim=c(0,10000),
col="darkmagenta",
freq=FALSE
)
```



Because the data is realy skewed, we use log scale to normalize

```
cd_RFM$monitery <- log(cd_RFM$monitery)
hist(cd_RFM$monitery,
main="monitery of customer",
xlab="monitery",
breaks=100,
col="darkmagenta",
freq=FALSE
)
```




```
cd_RFM1 = cd_RFM%>%
dplyr::select(-CustomerID,-country)

summary(cd_RFM1)

    recency        frequenci         monitery
 Min.   : 23.0   Min.   :  1.000   Min.   : 1.238
 1st Qu.: 40.0   1st Qu.:  1.000   1st Qu.: 5.185
 Median : 73.0   Median :  2.000   Median : 5.683
 Mean   :115.1   Mean   :  4.272   Mean   : 5.646
 3rd Qu.:164.8   3rd Qu.:  5.000   3rd Qu.: 6.064
 Max.   :396.0   Max.   :209.000   Max.   :11.341

cd_RFM2 <- cd_RFM1 %>%
mutate(recency = scale(recency),
       frequenci = scale(frequenci),
       monitery = scale(monitery)

)

summary(cd_RFM2)

      recency.V1         frequenci.V1         monitery.V1
 Min.   :-0.9204819   Min.   :-0.425048   Min.   :-5.883231
 1st Qu.:-0.7505027   1st Qu.:-0.425048   1st Qu.:-0.615310
 Median :-0.4205432   Median :-0.295144   Median : 0.049302
 Mean   : 0.0000000   Mean   : 0.000000   Mean   : 0.000000
 3rd Qu.: 0.4968443   3rd Qu.: 0.094568   3rd Qu.: 0.557567
 Max.   : 2.8090607   Max.   :26.594965   Max.   : 7.601186
```

Modeling and Evalutation

In this section we will apply various clustering methods to cluster the customer based rfm. We will use Partitioning clustering and Hierarchical clustering approaches.

Before we apply clustering models to the dataset we should assess clustering tendency. In order to do so we will employ **Hopkins** statistics.

Hopkins Statistics

Hopkins statistic is used to assess the clustering tendency of a dataset by measuring the probability that a given dataset is generated by a uniform data distribution.(Ref: Jiawei Han (2012)). Let's calculate Hopkins (**H**) statistics for cd_RFM2:

The **H** value close to one indicates very good clustering tendency. The **H** value around or greater than 0.5 denotes poor clustering tendency(Ref: Alboukadel Kassambara).

```
H =  get_clust_tendency(cd_RFM2,n = 100, graph = F, seed = 6709)
print(H[["hopkins_stat"]])
```

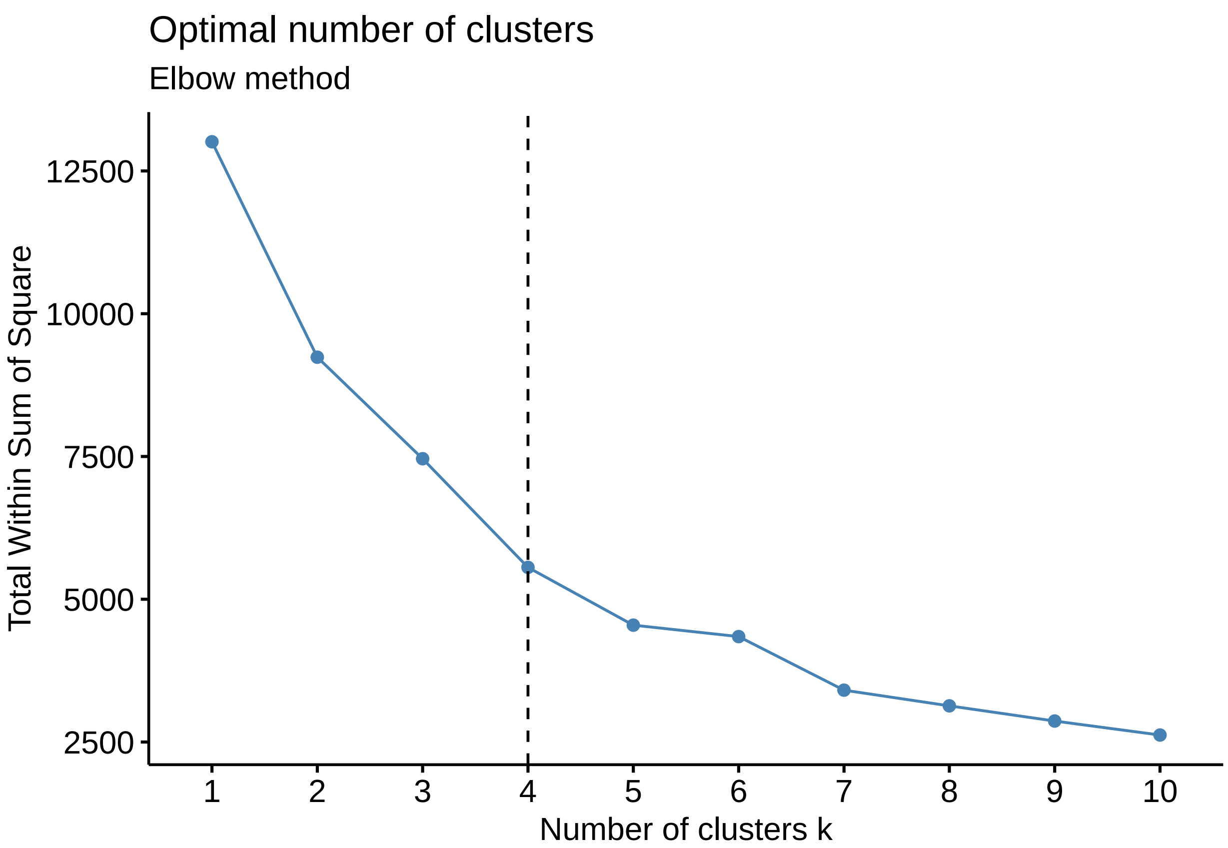
```
[1] 0.9816015
```

Perfect! H value is very close to 1. The dataset is clustrable.

Partitioning Clustering Approach

At first, we use Elbow method to get optimal number of clusters for k-means clustering:

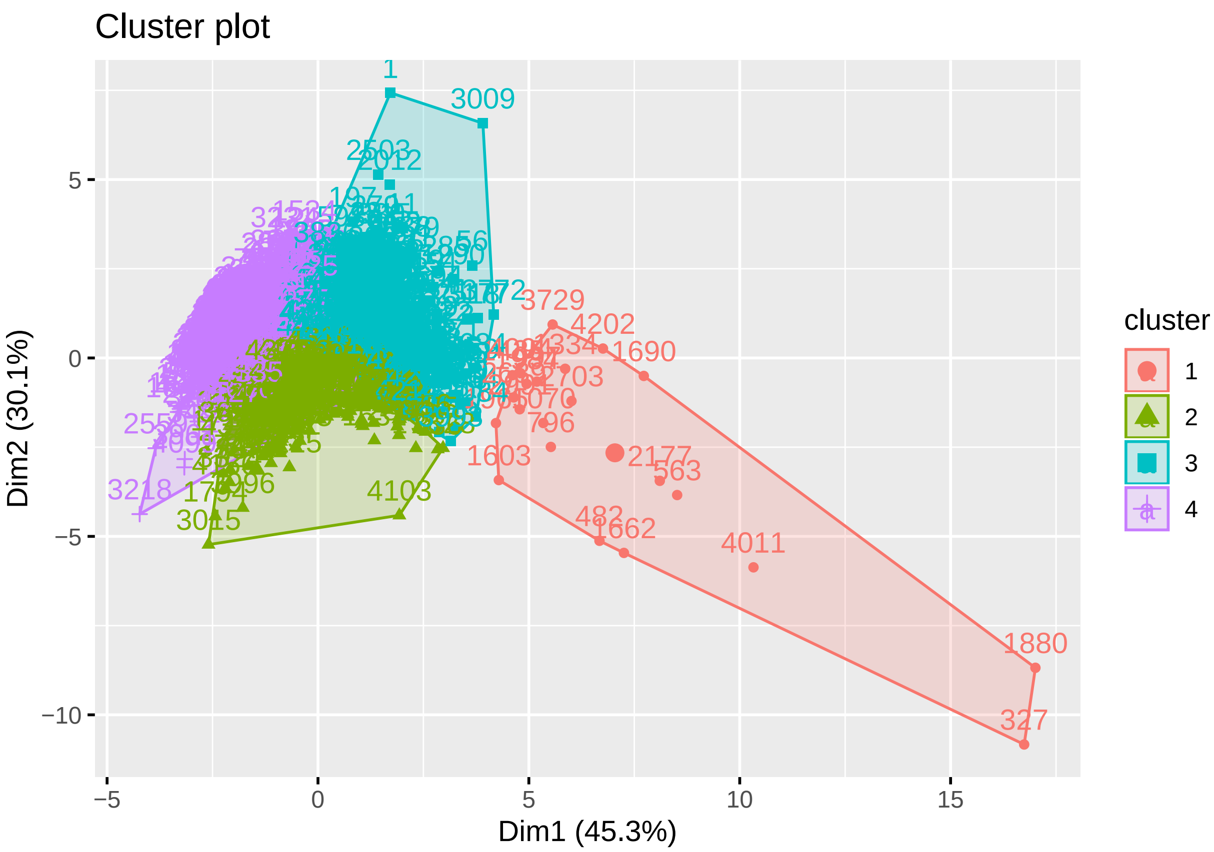
```
set.seed(123)
# Elbow method
fviz_nbclust(cd_RFM2, kmeans, method = "wss") +
    geom_vline(xintercept = 4, linetype = 2)+
  labs(subtitle = "Elbow method")
```



It seems that the optimal number of clusters is 4.

Let's use kmeans to cluster the dataset.

```
set.seed(123)
k2 <- kmeans(cd_RFM2, centers = 4, nstart = 25)
k2
fviz_cluster(k2, data = cd_RFM2)
```




```
K-means clustering with 4 clusters of sizes 22, 1504, 1840, 972

Cluster means:
     recency   frequenci   monitery
1 -0.8695790  9.50669672  1.2750038
2 -0.4700318 -0.07282135 -0.7293078
3 -0.4715478  0.13311570  0.7398225
4  1.6396157 -0.35448242 -0.3008688

Clustering vector:
   [1] 3 3 3 3 4 3 4 4 4 3 3 3 3 3 4 3 2 3 4 2 3 3 3 4 3 2 4 3 3 3 3 4 2 4 3 3 3
  [38] 3 3 2 3 3 2 4 4 3 3 3 3 3 3 4 3 2 4 3 3 3 3 2 2 2 3 3 4 2 3 3 2 3 3 3 3 3
  [75] 3 3 3 4 2 3 2 3 4 3 3 2 3 2 3 3 3 3 3 3 3 3 2 3 4 3 3 3 3 3 3 3 3 3 3 3 3
 [112] 3 3 3 4 3 3 2 4 2 3 2 3 4 3 2 4 3 2 4 3 2 3 4 4 3 3 3 3 3 3 2 2 3 3 2 3 3
 [149] 2 3 3 3 3 3 3 3 3 3 3 3 4 2 3 4 2 3 3 3 2 4 3 4 3 4 4 3 3 2 2 3 4 4 2 2 3
 [186] 3 4 2 4 3 2 4 2 2 2 3 3 4 2 2 3 3 3 3 2 2 4 2 3 2 3 3 3 3 3 3 4 3 3 3 2 3
 [223] 3 3 4 4 3 3 3 3 2 3 3 3 3 3 3 4 3 2 3 3 3 3 3 3 4 2 3 4 4 3 2 3 3 3 2 2 3
 [260] 3 4 3 3 3 3 4 3 3 2 3 2 2 3 3 3 3 3 4 3 3 4 2 2 3 2 3 3 3 2 2 3 3 4 3 3 3
 [297] 3 3 3 3 3 3 2 3 3 3 2 2 2 4 3 3 2 3 3 4 4 4 4 4 4 3 3 2 3 3 1 3 2 3 3 4 4
 [334] 2 3 3 3 3 2 4 3 4 3 3 2 2 3 3 3 3 2 4 2 2 4 4 4 4 3 3 4 3 4 2 4 4 2 2 3 2
 [371] 4 4 2 4 3 3 3 2 2 2 4 3 4 2 3 4 3 4 3 3 3 2 3 2 2 4 3 3 2 4 3 3 4 3 4 3 2
 [408] 3 4 3 4 2 4 4 2 3 2 4 2 4 3 2 3 3 3 4 3 4 2 2 4 3 4 2 3 4 3 3 3 3 2 3 3 2
 [445] 2 3 4 2 2 3 3 4 2 3 2 3 2 3 2 3 3 2 3 4 2 3 3 3 2 3 3 3 4 3 2 2 3 2 4 2 2
 [482] 1 4 4 2 3 2 4 2 3 2 4 3 2 2 2 3 2 3 4 2 3 4 2 3 4 2 4 4 2 3 3 3 2 2 3 3 3
 [519] 2 3 3 3 3 3 4 3 3 3 3 3 4 4 3 3 3 3 3 3 4 3 2 4 4 2 4 2 4 2 2 2 2 4 4 3 2
 [556] 3 2 2 3 3 2 4 1 3 3 2 4 2 2 3 3 2 4 3 2 2 2 3 4 3 2 3 3 3 2 2 4 4 3 3 2 2
 [593] 2 3 4 4 3 3 3 3 2 3 2 3 3 4 2 2 2 3 3 4 3 2 3 3 3 2 4 3 3 4 2 3 3 2 2 2 3
 [630] 3 2 3 2 3 2 4 3 4 2 2 3 2 2 3 3 3 2 2 2 2 3 3 3 2 3 4 2 2 2 4 4 3 3 4 4 3
```

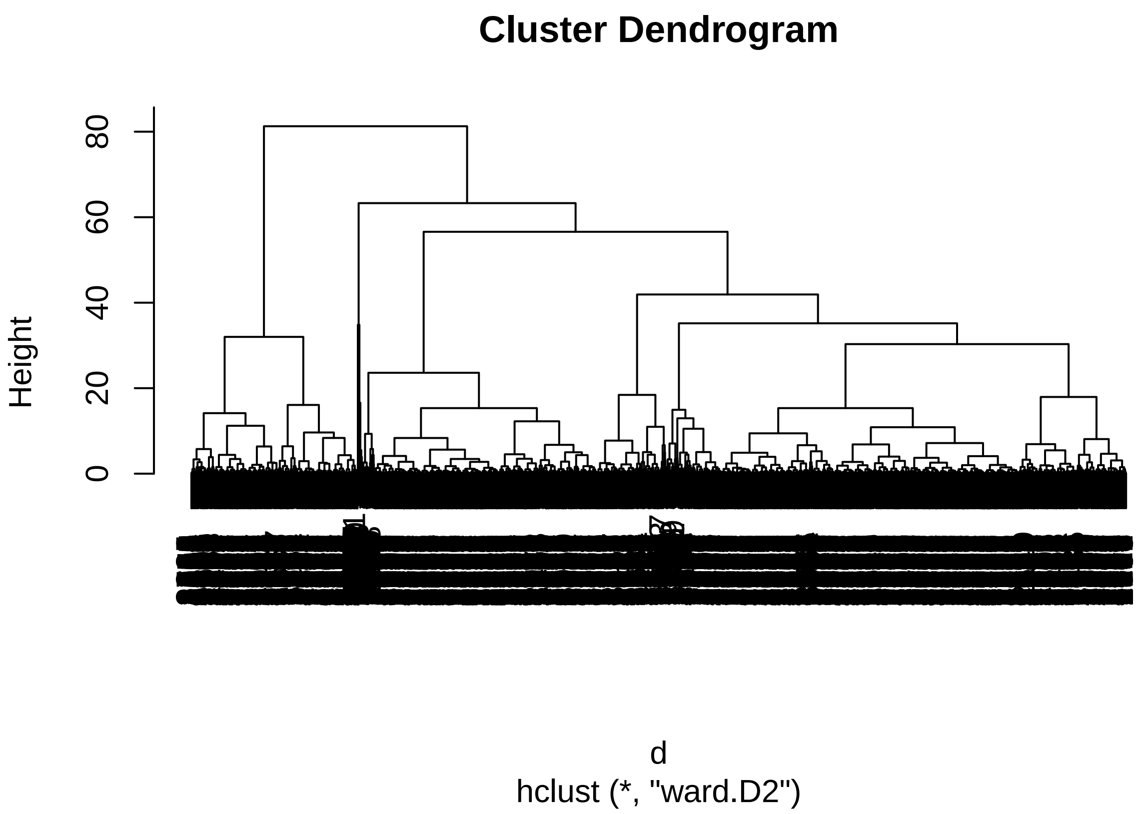
```
 [667] 4 2 2 3 3 3 2 4 2 2 4 2 3 3 2 2 2 3 2 2 2 3 4 4 3 3 3 3 3 3 4 2 2 2 3 2 3
 [704] 2 2 3 2 4 3 4 2 4 4 2 3 3 4 4 2 2 4 2 2 3 3 3 3 3 2 3 3 2 3 3 2 2 2 3 3 3
 [741] 4 3 3 3 3 3 3 3 3 4 4 3 4 3 2 2 3 3 3 3 3 3 4 4 3 3 3 2 3 2 4 4 4 3 3 3 3
 [778] 2 3 4 3 4 3 2 3 4 3 4 3 3 4 3 2 2 2 1 2 2 3 3 3 3 2 2 3 3 3 3 2 2 2 3 3 2
 [815] 3 2 3 4 2 3 3 2 3 3 4 4 3 3 3 2 3 4 3 4 3 2 2 3 2 4 4 3 2 4 3 4 2 4 4 3 2
 [852] 3 2 2 3 4 4 3 3 4 2 3 4 2 2 2 3 4 3 4 2 3 4 2 2 2 3 3 3 2 3 3 2 3 2 3 3 2
 [889] 4 2 4 3 3 2 2 2 2 3 2 4 3 3 2 2 2 4 3 3 3 2 3 4 2 4 2 3 2 2 3 3 2 4 3 4 3
 [926] 3 2 3 3 2 3 3 3 2 3 4 3 3 2 2 3 2 3 3 4 2 4 2 2 4 3 3 3 3 3 3 3 2 2 2 3 4
 [963] 2 3 2 3 4 3 2 3 3 3 4 2 3 4 3 4 2 4 3 2 3 4 2 4 4 3 2 4 4 3 4 3 2 3 1 2 2
[1000] 4 2 4 2 2 4 4 4 3 2 4 4 4 4 2 4 2 2 3 3 4 2 3 2 2 3 4 3 3 3 4 2 4 2 3 2 3
[1037] 4 4 2 3 4 2 4 2 2 3 3 3 2 3 4 3 3 3 3 3 3 3 3 2 2 4 2 2 4 2 3 3 2 1 3 4 3
[1074] 3 4 3 3 2 4 3 4 3 4 2 3 3 3 2 2 4 2 3 3 3 4 2 2 3 2 4 2 4 2 4 3 3 3 2 3 3
[1111] 3 3 3 3 4 2 4 2 3 4 2 2 3 3 3 3 3 2 4 3 2 2 3 2 3 2 3 2 4 2 4 3 3 2 2 4 2
[1148] 3 3 4 2 4 3 2 3 3 3 2 3 2 4 3 3 2 2 3 2 4 2 3 4 3 4 4 4 2 2 2 3 2 2 4 3 2
[1185] 4 3 3 2 2 2 2 2 2 4 2 3 3 2 4 3 3 4 2 3 2 3 2 2 3 3 3 3 4 2 2 3 3 2 3 4 3
[1222] 2 3 3 3 4 4 2 3 3 3 3 3 4 4 2 2 2 2 3 3 3 2 3 4 4 3 2 3 4 2 3 4 2 4 3 3 3
[1259] 2 2 2 3 3 3 4 3 3 2 4 2 4 2 2 3 2 2 2 3 2 4 4 2 3 2 3 2 4 3 3 3 2 3 2 3 2
[1296] 2 4 3 3 2 3 2 3 3 4 2 2 3 3 3 2 2 3 2 4 3 2 3 3 3 3 2 4 2 3 3 2 4 4 3 3 4
[1333] 4 1 2 3 3 4 4 3 3 4 2 3 2 2 3 4 2 2 3 2 3 4 3 3 3 3 3 4 3 2 4 2 2 2 4 2 3
[1370] 3 2 3 2 3 3 2 2 2 4 3 3 3 2 3 3 3 4 3 3 2 3 2 4 4 2 3 4 4 2 4 3 4 4 3 3 4
[1407] 3 3 2 3 3 4 2 3 4 4 2 2 3 3 4 3 3 3 3 2 4 2 3 3 3 2 2 3 1 3 2 3 4 3 3 3 2
[1444] 4 2 2 3 3 4 3 3 4 4 3 2 2 3 3 2 3 4 4 4 3 2 2 2 3 4 2 4 2 4 4 2 4 3 2 3 3
[1481] 2 3 3 4 4 4 4 4 2 2 3 3 2 4 3 3 3 3 4 3 2 2 3 3 3 2 2 3 3 3 2 4 3 4 2 3 2
[1518] 3 2 3 3 3 4 3 3 2 2 4 3 2 2 4 4 4 3 3 4 3 2 2 2 2 4 3 2 4 3 4 3 2 3 2 2 4
[1555] 3 2 2 2 2 3 3 4 3 4 2 4 3 4 3 2 2 3 4 4 3 2 2 4 4 3 2 2 4 2 3 2 3 3 2 2 2
[1592] 4 2 3 3 2 2 2 2 2 2 3 1 3 2 3 4 3 3 3 4 3 2 2 3 2 4 2 3 3 3 4 4 3 2 2 2 3
[1629] 2 3 2 3 3 3 3 3 2 4 3 4 4 2 2 2 2 2 4 2 4 2 4 3 2 2 2 3 3 2 4 2 4 1 3 3 2
[1666] 4 2 4 4 3 4 2 2 4 3 3 3 3 4 4 3 3 3 3 2 2 4 3 4 1 4 3 3 2 3 3 4 2 2 3 2 2
[1703] 2 3 3 2 3 4 4 4 3 3 3 3 2 2 2 3 3 4 2 3 2 4 3 4 2 4 3 3 2 3 2 4 3 3 2 3 3
[1740] 2 2 2 2 3 2 2 2 3 4 4 2 2 3 3 2 3 2 3 4 2 2 2 4 3 3 3 3 3 3 3 3 3 4 2 4 2
[1777] 2 3 2 3 4 2 3 2 3 4 2 4 2 2 3 2 2 2 2 2 4 3 2 2 3 2 4 3 2 2 3 2 4 3 4 4 2
[1814] 2 2 4 4 2 2 3 2 2 3 2 3 4 3 3 3 4 3 4 2 3 2 3 3 3 3 3 2 2 3 2 2 3 4 3 3 4
[1851] 2 4 3 2 2 2 3 2 3 4 3 4 4 4 4 3 3 3 4 3 2 2 3 3 3 4 3 2 2 1 3 3 2 3 3 4 3
[1888] 3 4 2 3 2 4 3 4 3 3 3 2 2 2 3 3 3 2 3 4 2 2 2 3 4 3 4 3 2 3 2 3 3 2 3 2 2
[1925] 2 3 4 2 4 2 4 2 2 4 3 3 3 3 3 4 3 3 2 2 4 2 2 4 2 2 3 2 3 2 2 2 2 2 4 2 3
[1962] 2 2 2 1 4 4 3 3 3 3 4 2 4 3 3 4 2 2 4 2 2 2 1 2 4 3 3 3 2 3 4 3 3 3 3 3 3
[1999] 4 2 4 4 3 3 4 3 4 3 3 2 2 3 3 4 2 3 4 3 3 4 3 4 2 3 3 3 3 3 3 2 4 3 2 3 3
[2036] 2 3 3 3 3 4 2 3 2 4 3 4 3 3 4 3 3 3 4 3 3 3 3 3 2 3 3 3 3 4 2 2 4 4 2 4 3
[2073] 4 2 4 3 4 2 4 2 3 2 3 3 3 3 3 3 3 2 3 2 3 4 3 2 2 4 3 4 4 2 2 2 3 4 3 4 4
[2110] 4 4 4 4 2 3 4 3 4 4 4 3 3 3 3 2 3 3 3 4 3 3 3 3 2 3 3 2 4 4 2 2 4 2 3 3 4
[2147] 3 3 3 2 3 3 3 2 2 4 4 2 2 2 3 3 3 3 4 3 2 3 3 3 3 4 2 3 2 4 1 2 2 4 2 4 3
[2184] 4 2 3 3 2 4 2 3 3 4 3 2 2 2 3 3 3 2 3 4 4 2 4 4 3 4 2 3 3 3 4 4 3 3 3 3 3
[2221] 3 3 2 2 3 3 3 4 3 4 3 4 4 4 3 2 3 4 2 4 3 2 2 4 4 2 2 4 3 3 4 2 4 3 3 2 2
[2258] 4 2 4 3 3 2 2 2 4 2 2 3 4 2 2 4 3 3 4 4 4 3 3 2 2 4 4 4 2 4 2 2 4 3 3 2 2
[2295] 2 3 3 2 3 3 3 3 2 3 3 2 3 3 3 3 3 3 3 2 3 4 2 3 3 4 4 4 2 2 3 3 2 4 2 3 3
[2332] 2 3 3 2 2 3 4 3 3 3 3 2 4 4 3 2 2 4 3 2 3 3 2 3 4 3 4 3 2 3 2 4 3 2 2 2 2
[2369] 3 2 3 3 4 3 2 4 2 3 2 2 4 2 4 3 2 3 2 3 2 3 3 2 2 2 2 3 4 2 4 2 2 2 3 2 3
[2406] 3 3 3 4 2 3 3 3 3 3 2 3 2 3 3 3 4 4 3 2 4 4 2 3 3 3 4 2 2 4 3 2 4 2 3 3 3
[2443] 4 2 4 3 3 3 2 3 3 3 2 3 3 2 3 2 2 4 2 3 4 3 3 4 3 4 3 4 2 2 3 4 2 2 2 4 4
[2480] 3 3 3 3 4 4 3 3 4 3 4 4 3 2 3 3 2 2 2 2 2 3 2 3 3 3 4 2 3 2 3 2 4 2 3 3 3
[2517] 2 3 3 2 4 2 4 4 2 3 2 3 3 3 4 2 3 3 3 3 3 3 3 3 3 3 3 2 3 4 2 3 3 3 4 3
[2554] 3 3 4 2 3 4 3 3 2 3 3 2 4 3 2 2 3 3 3 2 3 4 2 3 2 2 4 2 2 2 3 2 3 2 3 2 2
[2591] 3 3 2 3 4 3 2 3 2 4 4 4 2 2 2 2 4 4 4 2 4 2 4 3 2 3 2 2 3 4 2 4 2 4 4 3 2
[2628] 4 4 4 4 3 3 3 4 2 2 3 4 3 4 4 3 3 2 3 3 2 2 3 3 2 2 2 2 2 4 3 4 3 3 4 3 3
[2665] 3 3 3 3 3 2 2 2 3 2 3 3 2 3 3 3 2 4 3 2 2 4 2 3 1 2 2 2 2 2 3 2 4 2 2 3 2
[2702] 2 1 4 2 3 4 3 4 2 3 3 2 2 3 4 3 4 3 3 3 4 3 3 2 3 4 4 3 3 3 4 4 2 4 3 2 3
[2739] 3 4 2 2 2 2 2 2 4 4 3 2 3 3 2 3 4 2 3 3 4 3 2 4 4 2 4 4 2 3 3 2 3 4 4 3 3
[2776] 3 3 3 3 4 2 2 3 3 4 3 4 3 4 3 4 2 3 4 3 3 2 4 3 3 4 4 2 2 3 3 2 2 2 3 2 3
[2813] 2 2 3 3 3 3 2 2 3 3 2 2 2 3 4 3 3 3 4 2 3 2 2 3 3 3 3 3 3 2 4 2 4 4 3 2 4
[2850] 3 2 4 4 3 4 2 2 2 3 2 3 2 3 2 4 3 4 2 3 4 3 3 4 2 3 3 2 3 2 3 4 3 3 4 4 3
[2887] 3 3 3 4 3 3 4 3 2 3 4 2 2 3 3 4 3 3 3 2 4 3 3 3 4 3 2 2 4 2 2 2 3 2 2 3 2
[2924] 4 4 3 3 2 2 2 3 3 2 4 2 4 2 3 2 4 2 3 2 3 2 2 2 3 2 3 3 2 3 3 2 2 3 4 4 3
[2961] 3 3 2 2 2 4 3 4 2 3 3 3 2 3 3 4 4 3 4 2 2 3 2 2 2 2 2 3 2 2 1 3 4 4 2 4 2
[2998] 3 4 2 2 2 2 3 2 2 2 2 3 2 4 4 4 2 2 2 3 4 2 2 4 2 3 3 2 2 2 2 4 2 2 2 3 2
```

```
[3035] 2 3 2 4 4 3 2 3 3 4 2 3 3 2 4 2 3 3 3 2 4 4 3 2 3 2 4 2 2 4 3 3 4 2 2 4 2
[3072] 3 3 3 3 3 2 4 4 3 3 3 4 3 2 2 3 2 3 3 2 3 4 2 4 2 3 3 3 3 3 4 3 4 3 4 4 2
[3109] 2 3 2 2 3 2 2 4 2 2 3 2 2 3 4 4 3 3 4 2 4 3 4 2 2 4 3 4 3 3 3 2 3 4 3 2 3
[3146] 2 3 4 4 4 2 3 3 3 2 3 3 2 4 2 3 4 2 2 2 3 3 3 3 2 3 3 4 2 3 2 3 2 3 3 2 2
[3183] 4 2 2 4 4 3 3 3 2 3 3 2 3 3 3 4 3 4 3 4 2 2 4 3 3 4 2 4 2 3 2 3 2 3 3 4 4
[3220] 3 3 2 3 3 3 3 2 3 4 2 4 3 4 4 3 2 3 3 4 2 4 4 3 2 4 2 3 2 2 2 3 2 2 3 2 4
[3257] 2 2 3 2 2 2 4 2 2 4 3 3 3 3 2 4 2 3 2 2 4 2 4 2 3 2 2 3 3 4 2 3 3 2 4 3 4
[3294] 2 4 2 2 3 4 3 3 3 3 2 2 2 4 4 2 2 2 2 4 3 2 2 2 2 3 2 3 3 2 2 2 3 2 4 2 2
[3331] 2 4 2 2 4 2 4 2 2 3 3 3 2 3 2 2 3 2 2 4 2 2 2 4 2 2 2 3 2 4 2 2 2 2 2 2 2
[3368] 3 2 3 3 2 4 2 4 2 2 3 3 2 2 2 4 3 4 2 2 2 4 4 3 4 2 2 3 4 3 3 4 3 3 3 3 3
[3405] 2 2 2 4 3 2 4 2 2 3 3 3 2 2 2 2 2 3 3 3 3 2 2 2 4 2 3 2 2 3 2 4 3 2 3 3 2
[3442] 3 3 2 3 3 3 3 3 4 3 3 2 3 3 4 3 4 2 2 2 3 3 3 3 2 2 2 2 2 2 3 3 3 3 3 3 3
[3479] 3 3 4 3 2 2 3 3 4 4 3 2 4 2 2 2 4 4 3 4 2 4 4 2 2 4 3 3 2 2 4 3 4 4 3 2 3
[3516] 2 4 2 3 2 2 3 3 3 2 2 3 4 2 3 4 3 3 4 3 4 4 2 2 3 2 4 3 2 2 3 2 4 3 2 3 3
[3553] 4 3 2 3 4 3 2 4 2 2 2 2 3 4 4 2 2 2 4 3 2 2 4 2 2 2 2 4 3 3 3 4 2 3 2 4 2
[3590] 2 2 2 2 4 2 4 2 2 2 2 3 2 4 4 2 3 2 2 2 2 3 2 4 2 2 2 2 2 2 2 4 3 2 3 2 3
[3627] 3 2 3 3 3 2 2 4 2 2 2 3 4 3 3 2 3 4 4 2 2 2 2 2 4 3 2 3 3 4 2 4 2 2 2 4 3
[3664] 3 4 2 2 2 2 2 4 4 2 2 3 3 2 2 2 2 3 3 3 3 4 4 4 3 3 2 3 2 2 3 3 4 2 4 2 2
[3701] 2 3 3 3 2 3 2 3 3 3 3 2 3 2 2 4 2 3 3 2 4 2 3 4 3 2 2 3 1 2 2 4 4 4 3 3 3
[3738] 2 2 3 3 4 3 2 3 2 4 2 3 3 2 4 2 3 3 4 2 4 2 4 3 4 2 2 4 3 3 4 4 3 3 3 3 3
[3775] 2 2 2 4 3 3 2 2 2 4 4 3 2 3 2 4 2 4 4 3 2 2 4 2 4 2 4 4 2 3 3 2 4 2 2 2 2
[3812] 2 3 3 4 3 2 2 4 2 4 2 4 3 4 3 2 3 3 3 3 3 3 2 3 3 2 4 3 3 4 2 2 3 3 3 3 2
[3849] 4 3 4 2 3 3 3 3 2 2 4 2 3 2 2 2 4 3 3 4 3 4 2 3 3 2 3 3 3 4 3 3 3 2 2 2 3
[3886] 3 3 2 3 4 3 3 2 2 3 2 3 4 3 2 3 2 3 3 3 4 2 2 4 3 3 4 4 3 4 3 2 4 2 3 4 4
[3923] 4 2 3 4 4 3 4 3 3 3 3 3 3 3 3 3 3 4 3 2 3 2 2 3 3 2 3 4 2 4 2 4 3 2 3 2 2
[3960] 3 4 3 2 2 3 3 4 2 2 3 4 2 2 2 2 2 2 3 3 4 2 2 3 4 2 2 3 2 3 3 3 3 3 2 3 2
[3997] 2 3 3 3 3 4 2 2 2 2 2 2 3 3 1 4 2 3 2 4 4 2 4 4 3 3 3 4 4 2 4 3 2 3 3 2 2
[4034] 2 4 3 4 4 3 4 4 4 2 2 2 3 2 2 4 4 4 2 4 2 2 2 4 2 4 4 2 2 2 4 4 3 4 2 3 2
[4071] 3 3 2 4 3 4 2 2 2 3 2 3 3 2 3 3 2 3 3 2 4 2 3 1 2 3 2 4 4 2 3 2 2 2 2 2 2
[4108] 4 4 4 4 4 3 2 3 4 2 4 2 4 2 3 2 4 2 4 4 2 4 2 2 2 4 2 3 2 4 4 4 2 2 2 3 2
[4145] 2 2 3 4 2 2 4 2 2 2 2 2 2 2 2 2 2 2 4 4 3 3 4 2 2 2 4 3 2 3 3 4 3 4 4 3 4
[4182] 2 2 4 3 3 2 3 2 4 4 3 2 3 3 3 4 3 3 2 2 1 3 2 4 4 3 2 2 4 4 2 3 3 4 4 4 3
[4219] 3 2 2 2 4 3 4 2 2 4 3 3 3 2 3 2 2 2 2 3 2 2 3 2 4 3 3 3 4 3 4 2 3 2 3 3 2
[4256] 4 2 3 3 3 3 4 2 4 3 3 4 4 3 4 3 3 3 2 4 4 3 2 2 3 2 4 4 3 3 3 4 2 4 2 3 3
[4293] 4 3 3 4 3 3 3 4 2 4 3 2 2 4 3 2 3 3 3 3 2 4 3 2 2 3 3 3 2 2 3 3 2 4 2 3 2
[4330] 2 3 2 2 4 4 2 2 3

Within cluster sum of squares by cluster:
[1]  803.7674 1227.1171 2239.1390 1288.0246
 (between_SS / total_SS =  57.3 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

```
group <- k2$cluster
cd_RFM3 <- cbind(cd_RFM, group)
#write.csv(cd_RFM3, "../shiny/www/mydata1.csv")
```

Hierarchical clustering approache

```
set.seed(123)
d <- dist(cd_RFM2)
c <- hclust(d, method = 'ward.D2')

plot(c)
```




```
members <- cutree(c,k = 4)

table(members)
members

members
   1    2    3    4
2452  772 1092   22
   [1] 1 1 1 1 2 1 2 2 2 1 1 1 1 1 2 1 1 1 2 3 1 1 1 2 1 3 2 1 1 1 1 1 1 2 1 1 1
  [38] 1 1 1 1 1 1 2 2 1 1 1 1 1 1 2 1 1 2 1 1 1 1 3 1 3 1 1 1 1 1 1 3 1 1 1 1 1
  [75] 1 1 1 2 3 1 3 1 2 1 1 2 1 3 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1
 [112] 1 1 1 2 1 1 3 1 3 1 3 1 2 1 1 2 1 3 2 1 3 1 2 2 1 1 1 1 1 1 3 1 1 1 3 1 1
 [149] 3 1 1 1 1 1 1 1 1 1 1 1 2 2 1 2 3 1 1 1 1 2 1 2 1 2 2 1 1 1 1 1 2 2 1 3 1
 [186] 1 2 3 1 1 1 2 3 3 3 1 1 2 3 3 1 1 1 1 3 1 2 3 1 3 1 1 1 1 1 1 2 1 1 1 3 1
 [223] 1 1 2 2 1 2 1 1 1 1 1 1 1 1 1 2 1 3 1 1 1 1 1 1 1 3 1 2 2 1 3 1 1 1 3 1 1
 [260] 1 2 1 1 1 1 2 1 1 3 1 1 3 1 1 1 1 1 2 1 1 1 3 3 1 3 1 1 1 3 1 1 1 2 1 1 1
 [297] 1 1 1 1 1 1 1 1 1 1 1 3 1 2 1 1 1 1 1 1 2 2 2 2 2 1 1 1 1 1 4 1 1 1 1 2 2
 [334] 3 1 1 1 1 1 2 1 2 2 1 1 3 1 1 1 1 1 2 3 3 1 2 2 2 1 1 2 1 1 3 1 2 3 3 1 1
 [371] 2 2 3 2 1 1 1 3 3 3 2 1 2 3 1 2 1 2 1 1 1 1 1 1 3 2 1 1 3 2 1 1 2 1 1 1 3
 [408] 1 2 1 1 3 2 2 3 1 1 2 3 2 1 3 1 1 1 2 1 2 3 3 2 1 2 1 1 1 1 1 1 1 3 1 1 3
 [445] 3 1 1 3 1 1 1 2 3 1 3 1 1 1 3 1 1 3 1 2 1 1 1 1 3 1 1 1 2 1 3 3 1 3 2 3 3
 [482] 4 2 1 3 1 3 2 3 1 1 2 1 3 1 3 1 3 1 2 3 1 2 3 1 1 3 2 2 3 1 1 1 1 3 1 1 1
 [519] 3 1 1 1 1 1 2 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 3 2 2 1 1 1 2 1 3 3 1 1 2 1 3
 [556] 1 3 2 1 1 3 1 4 1 1 3 2 1 3 1 1 3 2 1 3 3 2 1 2 1 3 1 1 1 3 3 3 2 1 1 3 1
 [593] 3 1 2 2 1 1 1 1 3 1 3 1 1 2 3 3 3 1 1 2 1 1 1 1 1 3 2 1 1 2 3 1 1 3 3 3 1
 [630] 1 1 1 3 1 3 2 1 2 3 3 1 3 3 1 1 1 3 3 3 3 1 1 1 1 1 2 3 3 1 2 2 1 1 2 1 1
 [667] 2 1 3 1 2 1 3 2 3 3 1 3 1 1 3 3 1 1 3 3 1 1 2 2 1 1 1 1 1 1 2 3 3 3 1 3 1
 [704] 3 1 1 3 2 1 1 1 1 2 2 1 1 2 1 3 2 2 1 3 1 1 1 1 1 1 1 1 3 1 1 1 3 3 1 1 1
 [741] 2 1 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 2 2 1 1 1 3 1 3 2 2 2 1 1 1 1
 [778] 3 1 2 1 2 1 1 1 2 1 1 1 1 1 1 3 3 3 4 3 3 1 1 1 1 3 2 1 1 1 1 3 3 1 1 1 1
 [815] 1 3 1 2 1 1 1 1 1 1 2 1 1 1 1 3 1 2 1 1 1 1 3 1 3 1 1 1 1 2 1 2 3 2 2 1 3
 [852] 1 3 3 1 2 2 1 1 2 3 1 2 3 3 1 1 2 1 2 3 1 2 3 3 1 1 1 1 3 1 1 3 1 3 1 1 3
 [889] 1 3 2 1 1 3 1 3 3 1 3 2 1 1 1 3 3 2 1 1 1 3 1 1 3 1 1 2 3 3 1 1 3 2 1 2 1
 [926] 1 3 1 1 3 1 1 1 3 1 2 1 1 3 3 1 3 1 1 2 3 1 3 3 2 1 1 1 1 1 1 1 3 1 3 1 1
```

```
 [963] 3 1 3 1 2 1 3 1 1 1 1 3 1 1 1 1 3 1 1 3 1 2 3 1 1 1 3 2 2 1 2 1 3 1 4 1 3
[1000] 1 1 2 1 3 2 2 2 1 3 2 2 2 2 1 2 3 3 1 1 2 1 1 3 3 1 1 1 1 1 1 3 2 1 1 3 1
[1037] 1 2 3 1 2 3 2 2 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 3 3 2 3 1 2 3 1 1 3 4 1 2 1
[1074] 1 2 1 1 3 2 1 2 1 1 3 1 1 1 1 3 2 3 1 1 1 1 3 3 1 3 1 3 2 3 2 1 1 1 1 1 1
[1111] 1 1 1 1 2 3 2 1 1 2 3 3 1 1 1 1 1 3 2 1 3 3 1 1 1 1 1 3 1 1 1 1 1 1 3 2 3
[1148] 1 1 2 3 1 1 3 1 1 1 3 1 3 2 1 1 3 3 1 3 1 3 1 2 1 2 1 2 3 3 3 1 3 3 2 1 3
[1185] 1 2 1 3 2 3 3 3 3 2 2 1 1 3 2 1 1 1 3 1 3 1 1 3 1 1 1 1 1 3 3 1 1 1 1 1 1
[1222] 1 1 1 1 1 2 3 1 1 1 1 1 1 2 1 3 3 1 1 1 1 3 1 2 2 1 3 1 2 3 1 2 1 2 1 1 1
[1259] 3 3 3 1 1 1 2 1 1 1 2 3 2 3 3 1 3 3 3 1 3 2 2 3 1 3 1 3 2 1 1 1 3 1 3 1 1
[1296] 2 2 1 1 3 1 3 1 1 2 1 2 1 1 1 3 3 1 3 2 1 3 1 1 1 1 3 2 3 1 1 1 2 2 1 1 2
[1333] 2 4 3 1 1 1 2 1 1 2 3 1 3 3 1 2 3 3 1 1 1 2 1 1 1 1 1 1 1 3 2 3 3 3 2 3 1
[1370] 1 1 1 3 1 1 1 3 3 2 1 1 1 1 1 1 1 1 1 1 3 1 1 2 2 1 1 1 2 1 1 1 2 2 1 1 2
[1407] 1 1 3 1 1 2 1 1 2 2 3 3 1 1 2 1 1 1 1 3 2 3 1 1 1 1 3 1 4 1 3 1 2 1 1 1 1
[1444] 2 1 3 1 1 1 1 1 2 1 1 3 1 1 1 3 1 1 2 2 1 3 3 3 1 2 3 2 2 1 2 3 2 1 3 1 1
[1481] 3 1 1 2 2 2 1 1 3 3 1 1 3 2 1 1 1 1 2 1 1 1 1 1 1 1 3 1 1 1 3 2 1 2 1 1 1
[1518] 1 3 1 1 1 1 1 1 3 1 2 1 3 3 2 2 2 1 1 1 1 1 3 3 3 2 1 3 2 1 2 1 3 1 1 3 2
[1555] 1 3 3 1 3 1 1 2 1 2 1 1 1 2 1 3 3 1 2 2 1 3 3 2 2 1 1 3 2 1 1 3 1 1 3 3 1
[1592] 2 3 1 1 3 3 3 1 3 3 1 4 1 3 1 1 1 1 1 2 1 3 1 1 3 2 1 1 1 1 1 2 1 3 2 3 1
[1629] 3 1 1 1 1 1 1 1 3 2 1 1 2 3 3 3 3 3 1 3 2 3 2 1 3 3 3 1 1 3 2 3 2 4 1 1 3
[1666] 2 3 2 1 1 1 3 3 1 1 1 1 1 2 2 1 1 1 1 3 3 1 1 1 4 1 1 1 3 1 1 2 3 3 1 3 3
[1703] 3 1 1 3 1 2 2 2 1 1 1 1 3 3 3 1 1 2 3 1 3 2 1 2 1 2 1 1 1 1 3 2 1 1 3 1 1
[1740] 1 3 3 3 1 3 3 3 1 2 2 3 3 1 1 1 1 3 1 1 3 3 1 2 1 1 1 1 1 1 1 1 1 2 3 2 1
[1777] 3 1 3 1 2 3 1 1 1 2 3 2 3 3 1 3 3 3 3 3 2 1 3 3 1 3 1 1 3 3 1 1 2 1 1 1 1
[1814] 3 3 2 2 1 1 1 3 3 1 3 1 2 1 1 1 1 1 2 1 1 1 1 1 1 1 1 3 3 1 1 3 1 1 1 1 2
[1851] 3 1 1 3 1 1 1 3 1 2 1 1 2 2 1 1 1 1 1 1 3 3 1 1 1 1 1 1 3 4 1 1 3 1 1 1 1
[1888] 1 2 3 1 1 2 1 2 1 1 1 3 3 3 1 1 1 1 1 2 2 3 3 1 2 1 2 1 3 1 3 1 1 3 1 3 1
[1925] 3 1 2 3 2 3 1 3 3 1 1 1 1 1 1 2 1 1 3 3 1 3 3 2 1 3 1 3 1 3 1 3 1 3 2 1 1
[1962] 1 3 3 4 1 2 1 1 1 1 2 3 2 1 1 2 3 3 2 1 3 3 4 1 1 1 1 1 1 1 2 1 1 1 1 1 1
[1999] 1 3 2 2 1 1 2 1 1 1 1 3 1 1 1 2 3 1 1 1 1 2 1 2 1 1 1 1 1 1 1 1 2 1 3 1 1
[2036] 3 1 1 1 1 1 3 1 3 2 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 2 3 1 2 2 3 2 1
[2073] 2 3 3 1 2 1 1 3 1 3 1 1 1 1 1 1 1 3 1 1 1 2 1 3 3 1 1 2 2 3 1 3 1 1 1 2 2
[2110] 2 2 2 2 1 1 1 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 1 2 2 1 1 1 2 3 1 1 2
[2147] 1 1 1 1 1 1 1 3 3 2 2 1 3 1 1 1 1 1 2 1 3 1 1 1 1 2 3 1 3 2 4 1 3 1 1 2 1
[2184] 2 1 1 1 3 2 3 1 1 2 2 2 3 1 1 1 1 3 1 2 2 3 1 2 1 2 3 1 1 1 2 2 1 1 1 1 1
[2221] 1 1 1 3 1 1 1 1 1 1 1 1 2 1 1 3 1 2 1 2 1 1 3 1 1 1 1 2 1 1 1 1 2 1 1 3 1
[2258] 2 3 2 1 1 3 3 3 1 3 3 1 2 3 3 2 1 1 2 2 2 1 1 3 3 2 2 2 3 2 3 3 2 1 1 3 3
[2295] 1 1 1 3 1 1 1 1 2 1 1 3 1 1 1 1 1 1 1 1 1 2 3 1 1 1 1 2 3 2 1 1 3 1 1 1 1
[2332] 3 1 1 1 3 1 2 1 1 1 1 1 2 2 1 3 3 2 1 3 1 1 3 1 2 1 2 1 1 1 3 2 1 3 3 1 1
[2369] 1 3 1 1 1 1 2 2 3 1 3 3 1 1 2 1 3 1 1 1 3 1 1 3 3 3 1 1 2 1 1 3 1 3 1 3 1
[2406] 1 1 1 1 3 1 1 1 1 1 1 1 3 1 1 1 1 1 1 2 1 2 3 1 1 1 2 3 3 1 1 3 2 3 1 1 1
[2443] 2 3 2 1 1 1 3 1 1 1 3 1 1 3 1 3 3 2 3 1 2 1 1 1 1 2 1 2 1 1 1 2 3 3 3 1 2
[2480] 1 1 1 1 2 2 1 1 2 1 2 2 1 3 1 1 3 3 3 1 3 1 3 1 1 1 2 3 1 3 1 3 2 1 1 1 1
[2517] 3 1 1 1 2 1 1 2 3 1 1 1 1 1 2 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 3 1 1 1 1 1
[2554] 1 1 2 1 1 3 1 1 1 1 1 1 2 1 1 3 1 1 1 3 1 2 3 1 3 3 2 3 1 3 1 1 1 3 1 3 3
[2591] 1 1 3 1 2 1 3 1 1 2 2 2 3 3 3 3 1 1 2 3 2 1 2 1 3 1 3 3 1 1 3 2 3 2 2 1 2
[2628] 1 2 2 1 1 1 1 1 3 3 1 2 1 2 2 1 1 3 1 1 3 3 1 1 3 3 3 3 3 2 1 2 1 1 1 1 1
[2665] 1 1 1 1 1 3 3 3 1 3 1 1 3 1 1 1 3 2 1 3 1 2 1 1 4 3 3 3 3 3 1 1 2 1 1 1 3
[2702] 3 4 2 1 1 2 1 2 3 1 1 3 3 1 2 1 2 1 1 1 2 1 1 1 1 2 2 1 1 1 2 2 1 2 1 3 1
[2739] 1 2 3 1 2 1 3 1 2 2 1 1 1 1 3 1 2 3 1 1 1 1 2 2 2 3 1 1 3 1 1 3 1 2 2 1 1
[2776] 1 1 1 1 2 1 3 1 1 2 1 2 1 1 1 2 3 1 2 1 1 3 2 1 1 3 2 1 1 1 1 3 1 3 1 1 1
[2813] 3 3 1 1 1 1 3 1 1 1 1 3 3 1 2 1 1 1 2 3 1 3 3 1 1 1 1 1 1 3 1 1 2 2 1 1 2
[2850] 1 1 2 1 1 1 1 3 3 1 3 1 1 1 3 1 1 2 3 1 2 1 1 2 2 1 1 3 1 3 1 2 1 1 2 1 1
[2887] 1 1 1 2 1 1 2 1 1 1 2 3 3 1 1 2 1 1 1 3 2 1 1 1 2 1 3 3 1 3 3 1 1 3 3 1 3
[2924] 1 2 1 1 3 3 2 1 1 3 2 1 2 1 1 3 2 3 1 1 1 3 3 3 1 3 1 1 1 1 1 3 1 1 2 2 1
[2961] 1 1 1 3 3 2 1 2 1 1 1 1 3 1 1 2 2 1 2 3 1 1 3 3 3 3 3 1 3 3 4 1 2 2 1 2 1
[2998] 1 1 1 3 3 3 1 3 3 3 3 1 1 2 1 1 1 3 1 1 2 1 1 2 3 1 1 3 3 3 3 2 2 1 3 1 3
[3035] 3 1 3 2 1 1 3 1 1 2 3 1 1 2 2 3 1 1 1 3 2 2 1 3 1 3 2 1 3 1 1 1 1 3 1 1 3
[3072] 1 1 1 1 1 3 1 2 1 1 1 2 1 1 3 1 3 1 1 1 1 2 3 1 3 1 1 1 1 1 2 1 2 1 2 1 3
[3109] 3 1 3 3 1 3 3 2 3 3 1 1 1 1 2 2 1 1 1 1 2 1 2 3 3 2 1 2 1 1 1 3 1 2 1 3 1
[3146] 3 1 2 1 2 1 1 1 1 3 1 1 3 2 3 1 2 3 1 3 1 1 1 1 3 1 1 2 3 1 3 1 3 1 1 3 3
[3183] 2 1 3 2 2 1 1 1 3 1 1 1 1 1 1 1 1 2 1 2 3 3 2 1 1 2 1 2 1 1 1 1 3 1 1 3 1
[3220] 1 1 3 1 1 1 1 3 1 2 3 2 1 1 2 1 1 1 1 1 1 2 2 1 3 2 3 1 3 3 1 1 3 3 1 3 2
[3257] 1 3 1 3 1 3 2 3 3 2 1 1 1 1 3 1 3 1 3 3 1 1 2 3 1 3 3 1 1 2 3 1 1 3 1 1 2
[3294] 1 2 3 3 1 1 1 1 1 1 3 3 1 2 2 3 3 3 3 2 1 3 1 2 3 1 1 1 1 3 3 3 1 1 2 3 3
```

```
[3331] 3 1 3 3 2 3 2 3 3 1 1 1 1 1 3 3 1 3 1 2 3 3 1 2 3 1 3 1 1 2 3 1 3 3 1 3 1
[3368] 1 1 1 1 1 1 1 2 3 3 1 1 3 3 1 2 1 1 3 3 3 2 2 1 1 3 3 1 2 1 1 2 1 1 1 1 1
[3405] 3 3 3 2 1 1 2 2 1 1 1 1 3 3 3 1 3 1 1 1 1 3 3 3 2 3 1 3 3 1 1 1 1 3 1 1 3
[3442] 1 1 3 1 1 1 1 1 1 1 1 1 1 1 2 1 2 3 1 1 1 1 1 1 3 1 3 3 3 3 1 1 1 1 1 1 1
[3479] 1 1 2 1 1 3 1 1 3 1 1 3 2 3 3 1 2 2 1 2 3 1 1 3 1 2 1 1 3 3 2 1 2 2 1 1 1
[3516] 3 1 3 1 3 3 1 1 1 3 3 1 2 3 1 2 1 1 2 1 1 1 3 3 1 3 2 1 3 3 1 3 2 1 3 1 1
[3553] 1 1 3 1 2 1 3 2 1 3 1 3 1 2 2 3 3 3 1 1 3 3 2 1 3 3 1 2 1 1 1 2 1 1 3 2 3
[3590] 3 3 3 3 1 1 2 1 3 3 3 1 3 1 1 3 1 1 3 3 3 1 1 2 1 3 3 1 3 3 3 2 1 3 1 3 1
[3627] 1 1 1 1 1 3 1 2 1 3 3 1 2 1 1 3 1 1 2 3 3 1 3 3 2 1 3 1 1 2 3 2 3 3 3 2 1
[3664] 1 2 3 3 3 3 3 1 2 3 1 1 1 3 3 3 1 1 1 1 1 1 2 2 1 1 3 1 1 1 1 1 2 3 1 3 1
[3701] 3 1 1 1 3 1 3 1 1 1 1 3 1 3 3 2 3 1 1 3 2 2 1 1 1 2 1 1 4 3 1 2 2 2 1 1 1
[3738] 3 1 1 1 1 1 3 1 1 1 3 1 1 3 2 3 1 1 2 1 2 1 2 1 2 3 1 1 1 1 2 2 1 1 1 1 1
[3775] 3 3 3 2 1 1 3 3 3 2 2 1 1 1 1 1 3 2 2 1 3 1 2 3 2 3 2 2 1 1 1 1 2 3 1 3 3
[3812] 3 1 1 2 1 3 3 2 1 2 3 2 1 2 1 1 1 1 1 1 1 3 1 1 3 2 1 1 2 3 1 1 1 1 1 1 3
[3849] 1 1 2 3 1 1 1 1 3 3 1 3 1 1 1 1 2 1 1 2 1 1 3 1 1 3 1 1 1 2 1 1 1 1 3 3 1
[3886] 1 1 3 1 2 1 1 1 1 1 3 1 2 1 3 1 3 1 1 1 2 3 3 1 1 1 2 2 1 2 1 1 1 1 1 2 2
[3923] 2 3 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 3 3 1 1 3 1 2 3 1 1 2 1 1 1 1 3
[3960] 1 3 1 2 1 1 1 1 3 3 1 2 3 3 2 3 3 3 1 1 2 1 3 1 2 3 1 1 3 1 1 1 1 1 3 1 3
[3997] 3 1 1 1 1 2 3 3 3 3 3 3 1 1 4 2 3 1 3 2 1 3 2 2 1 1 1 2 2 3 2 1 3 1 1 3 3
[4034] 1 2 1 2 1 1 1 2 2 3 3 3 1 3 3 2 2 1 3 2 3 3 1 1 3 2 2 3 3 1 2 2 1 2 3 1 3
[4071] 1 1 1 2 1 2 3 3 3 1 3 1 1 2 1 1 2 1 1 3 2 3 1 4 3 1 1 2 3 3 1 2 1 2 1 1 1
[4108] 2 2 2 1 2 1 1 1 2 1 2 3 1 3 1 3 1 3 1 3 3 2 1 3 3 2 3 1 1 1 2 2 2 3 3 1 3
[4145] 3 3 1 3 3 3 2 3 3 3 3 3 3 1 3 3 1 3 2 2 1 1 2 3 3 1 1 1 1 1 1 2 1 2 1 1 2
[4182] 1 1 2 1 1 3 1 3 2 2 1 3 1 1 1 2 1 1 3 3 4 1 3 2 2 1 1 3 2 2 3 1 1 2 2 1 1
[4219] 1 1 3 3 2 1 2 3 1 2 1 1 1 1 1 2 3 3 3 1 3 3 1 3 2 1 1 1 1 1 1 3 1 1 1 1 3
[4256] 2 3 1 1 1 1 1 3 2 1 1 2 2 1 1 1 1 1 3 1 2 1 1 3 1 3 2 1 1 1 1 1 1 1 3 1 1
[4293] 2 1 1 2 1 1 1 1 3 2 1 3 3 1 1 1 1 1 1 1 3 2 1 1 3 1 1 1 3 3 1 1 3 2 3 1 3
[4330] 3 1 3 3 2 2 3 1 1
```

According the characteristic of every group, we can give a desription for every group as below.
group 1: Champions
Bought recently, buy often and spend the most!
Reward them. Can be early adopters for new products. Will promote your brand.

group 2: Recent Customers
Bought most recently, but not often.
Provide on-boarding support, give them early success, start building relationship.

group 3: Hibernating Last purchase was long back, low spenders and low number of orders. Offer other relevant products and special discounts. Recreate brand value.

group 4: Promising
Recent shoppers, but haven't spent much.
Create brand awareness, offer free trials

Clustering Method Evaluation

We have applied two different clustering algorithm. Choosing between k-means and hierarchical clustering is not easy. We compare the two kinds of groups with the actual expected result, we decided to adopt k-means.

Model Deployment

Fortunately we can state that the clustering methods were effective for the selected dataset. We do believe it might have a real live application. The model can segment customer successfully.

Conclusion

We selected **e-commerce** dataset hoping to discover the relationship between various attributes, which would segment the customer into different groups.

We spent significant efforts parsing and cleaning the data. Then we separated redundant and useful features. We add new features according to our requirement.

We also processed descriptive features applying data mining techniques. We counted the most frequently used terms to understand the content of the features. We counted the words. we successfully identified the most common words and phrase .

When the data preprocessing was done we measured Hopkins statistics to evaluate cluster tendency of the data set. The result was satisfactory; we proceeded with the clusterization.

We have applied two different clustering algorithm. Choosing between k-means and hierarchical clustering is not easy. We compare the two kinds of groups with the actual expected result, we decided to adopt k-means.

Overall we were able to apply unsupervised learning to reach our goal, and also we develop one shiny app to present our product.

Bibliography

F. M. Alboukadel Kassambara. Extract and visualize the results of multivariate data analyses. URL `https://cran.r-project.org/web/packages/factoextra/factoextra.pdf`. [p19]

J. P. Jiawei Han, Micheline Kamber. *Data Mining. Concepts and Techniques*. The Morgan Kaufmann Series in Data Management Systems, 225 Wyman Street, Waltham, MA 02451, USA, 2012. ISBN 978-0-12-381479-1. [p19]

Note from the Authors

This file was generated using *The R Journal* style article template, additional information on how to prepare articles for submission is here - Instructions for Authors. The article itself is an executable R Markdown file that could be downloaded from Github with all the necessary artifacts.

Ketao Li
York University

`liketao@yahoo.com`

Kush Halani
York University

`kush.halani@ontariotechu.net`

Josue Romain
York University

`josue.rolland.romain@gmail.com`

Juan Peña
York University

`jppena62@my.yorku.ca`