Mod 3.3: R Workshop on RFM Analysis

Eric Tham

2 May 2018

R Markdown

This is a workshop on computing RFM modified from the Kaggle website. https://www.kaggle.com/hendraherviawan/customer-segmentation-using-rfm-analysis-r RFM stands for the three dimensions: 1. Recency - How recently did the customer purchase? 2. Frequency - How often do they purchase? 3. Monetary Value - How much do they spend?

Reading in the data that is downloaded from Kaggle website.

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 3.4.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
#library(stringr)
#library(DT)
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.4.2
```

```
library(knitr)
library(rmarkdown)

df_data <- fread('customerdata.csv')
glimpse(df_data)
```

```
## Observations: 541,909
## 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    <int> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2...
## $ InvoiceDate <chr> "12/1/2010 8:26", "12/1/2010 8:26", "12/1/2010 8:2...
```

```
## $ UnitPrice   <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1....
## $ CustomerID  <int> 17850, 17850, 17850, 17850, 17850, 17850, 17850, 1...
## $ Country     <chr> "United Kingdom", "United Kingdom", "United Kingdo...
```

Cleaning data

```
## Warning: package 'bindrcpp' was built under R version 3.4.2
```

Recode variables

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

df_data <- df_data %>%
  mutate(total_dolar = Quantity*UnitPrice)

glimpse(df_data)
```

```
## Observations: 397,884
## Variables: 9
## $ InvoiceNo   <fctr> 536365, 536365, 536365, 536365, 536365, 536365, 5...
## $ StockCode   <fctr> 85123A, 71053, 84406B, 84029G, 84029E, 22752, 217...
## $ Description <chr> "WHITE HANGING HEART T-LIGHT HOLDER", "WHITE METAL...
## $ Quantity    <int> 6, 6, 8, 6, 6, 2, 6, 6, 6, 32, 6, 6, 8, 6, 6, 3, 2...
## $ InvoiceDate <date> 2010-12-01, 2010-12-01, 2010-12-01, 2010-12-01, 2...
## $ UnitPrice   <dbl> 2.55, 3.39, 2.75, 3.39, 3.39, 7.65, 4.25, 1.85, 1....
## $ CustomerID  <fctr> 17850, 17850, 17850, 17850, 17850, 17850, 17850, ...
## $ Country     <fctr> United Kingdom, United Kingdom, United Kingdom, U...
## $ total_dolar <dbl> 15.30, 20.34, 22.00, 20.34, 20.34, 15.30, 25.50, 1...
```

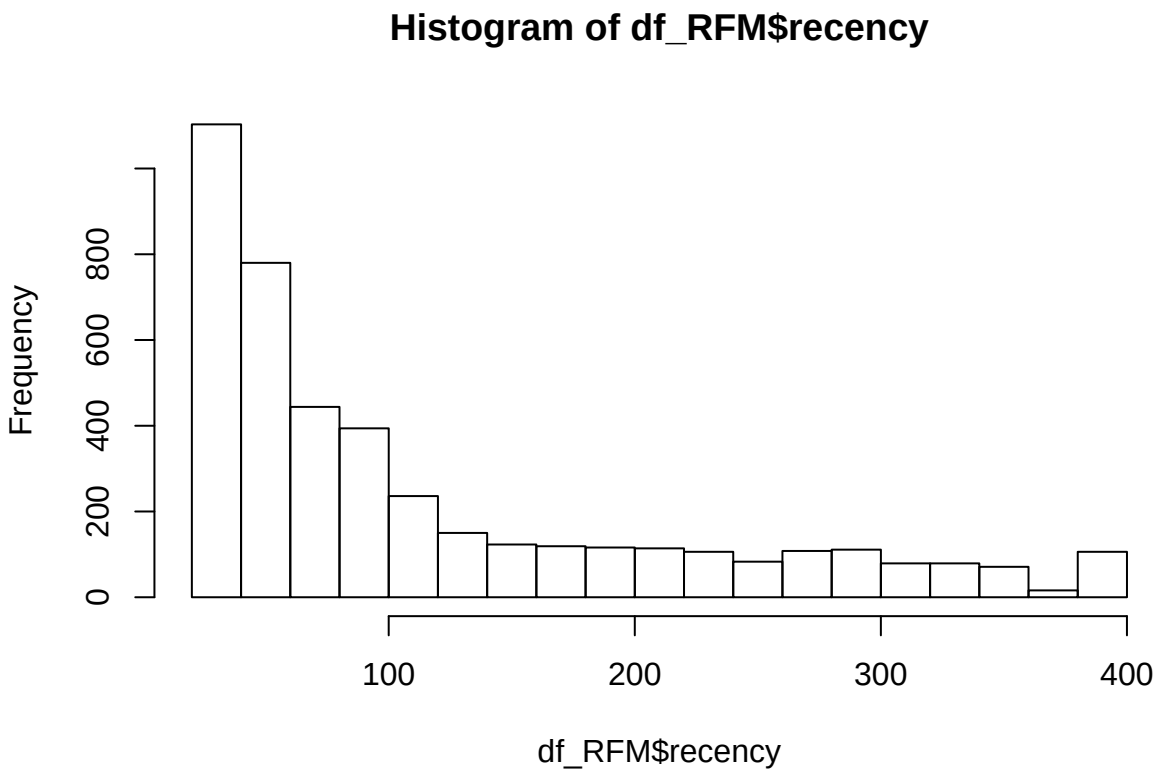
```
df_RFM <- df_data %>%
  group_by(CustomerID) %>%
  summarise(recency=as.numeric(as.Date("2012-01-01")-max(InvoiceDate)),
            frequenci=n_distinct(InvoiceNo), monitery= sum(total_dolar)/n_distinct(InvoiceNo))

summary(df_RFM)
```

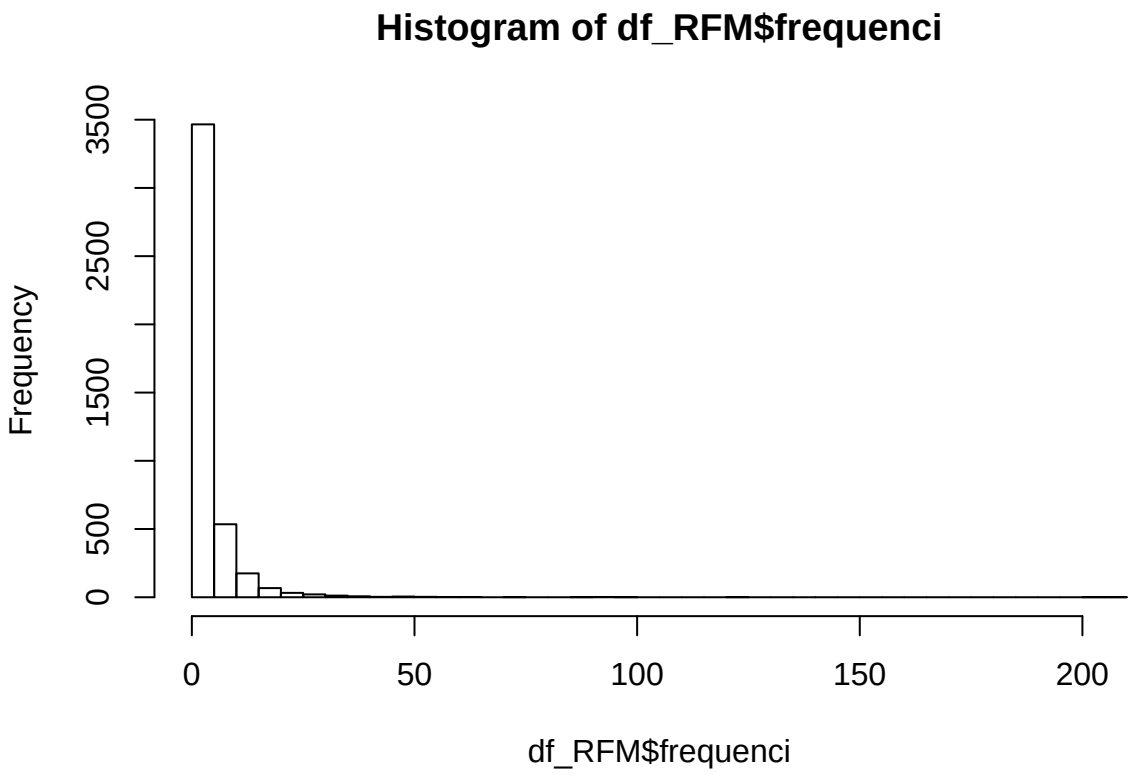
```
##    CustomerID      recency        frequenci         monitery
##  12346  :   1   Min.   : 23.0   Min.   :  1.000   Min.   :    3.45
##  12347  :   1   1st Qu.: 40.0   1st Qu.:  1.000   1st Qu.:  178.62
##  12348  :   1   Median : 73.0   Median :  2.000   Median :  293.90
##  12349  :   1   Mean   :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
```

Histogram of the RFM

```
hist(df_RFM$recency)
```




```
hist(df_RFM$frequenci, breaks = 50)
```



Applying hurdle Only the rows with frequency greater than 1 are filtered.

```
df_RFM_Hurdle <- df_RFM[df_RFM$frequenci > 1,]
```

Cutting dataframe by quartiles of R, F or M

Sorting the dataframe cut by the cut quartiles, and examining the individual group properties.

```
df_RFM$q_rec <- ntile(df_RFM$frequenci,4)  # part of the dplyr library
df_RFM <- df_RFM[order(df_RFM$q_rec),]     # sorting the cut rows of the RFM

# f1 <- function(x) c(Mean = mean(x), Max = max(x), SD = sd(x))
df_RFM_grouppty <- aggregate(df_RFM[,c("monitery","recency")],list(df_RFM$q_rec), mean)
```

The properties of the quartile group by frequency for say the monetary can be examined as well.

```
library(plyr)
```

```
## ------------------------------------------------------------------------
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## ------------------------------------------------------------------------
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
ddply(df_RFM, ~q_rec, summarise, mean=mean(monitery), sd=sd(monitery))
```

```
##   q_rec     mean        sd
## 1     1 450.1686 2375.4662
## 2     2 419.0710 2572.8033
## 3     3 386.0922  679.9241
## 4     4 421.3352  439.2508
```