RFM-Analysis.R

jyiwu

2022-02-17

```
# # ==================================================
# Title: RFM analysis on CDNOW data
# Author: Jingyi Wu
# Instructor: Yufeng Huang
# Data: CDNOW customer data (this time full data)
# Source: provided by Professor Bruce Hardie on
#   http://www.brucehardie.com/datasets/CDNOW_sample.zip
# ==================================================


# ====== CLEAR EVERYTHING ======
rm(list = ls())



# ====== READ TRIAL DATA =======

url <- 'https://dl.dropboxusercontent.com/s/xxfloksp0968mgu/CDNOW_sample.txt'
if (!file.exists('CDNOW_sample.txt')) {     # check whether data exists in local folder
 (prevents downloading every time)
    download.file(url, 'CDNOW_sample.txt')
}
df.raw <- read.fwf('CDNOW_sample.txt', width = c(6, 5, 9, 3, 8), stringsAsFactors = F)
# load data

# ====== Section 2: loading the data ======

df.raw[[1]] <- NULL # drop old id
names(df.raw) <- c("id", "date", "qty", "expd")

head(df.raw)
```

```
##   id     date qty  expd
## 1  1 19970101   2 29.33
## 2  1 19970118   2 29.73
## 3  1 19970802   1 14.96
## 4  1 19971212   2 26.48
## 5  2 19970101   3 63.34
## 6  2 19970113   1 11.77
```

```
# a) generate year and month


df.raw$date <- as.Date(as.character(df.raw$date), format = "%Y%m%d")
df.raw$year <- as.numeric(format(df.raw$date, "%Y"))
df.raw$month <- as.numeric(format(df.raw$date, "%m"))
head(df.raw)
```

```
##   id       date qty  expd year month
## 1  1 1997-01-01   2 29.33 1997     1
## 2  1 1997-01-18   2 29.73 1997     1
## 3  1 1997-08-02   1 14.96 1997     8
## 4  1 1997-12-12   2 26.48 1997    12
## 5  2 1997-01-01   3 63.34 1997     1
## 6  2 1997-01-13   1 11.77 1997     1
```

```
# b) aggregate into monthly data with number of trips and total expenditure

individual_month <- aggregate(.~ id + year + month, data = df.raw, FUN = sum)
head(individual_month)
```

```
##   id year month  date qty  expd
## 1  1 1997     1 19741   4 59.06
## 2  2 1997     1 19736   4 75.11
## 3  3 1997     1  9862   1  6.79
## 4  4 1997     1  9862   1 13.97
## 5  5 1997     1  9862   2 23.94
## 6  6 1997     1 19734   2 68.98
```

```
num_trips <- aggregate(qty ~ id + year + month, data = df.raw, FUN = length)
colnames(num_trips)[4] <- "trips"
df <- subset(cbind(individual_month,num_trips),
             select = c("id","year","month","qty","expd","trips"))
head(df)
```

```
##   id year month qty  expd trips
## 1  1 1997     1   4 59.06     2
## 2  2 1997     1   4 75.11     2
## 3  3 1997     1   1  6.79     1
## 4  4 1997     1   1 13.97     1
## 5  5 1997     1   2 23.94     1
## 6  6 1997     1   2 68.98     2
```

```
# c) generate a table of year-months, merge, replace no trip to zero.
# Hint: how do you deal with year-months with no trip? These periods are not in the orig
inal data,
#   but you might need to have these periods when you calcualte RFM, right?
# Consider expanding the time frame using expand.grid() but you do not have to.

df <- df[
  with(df, order(id)),
]

head(df)
```

```
##      id year month qty  expd trips
## 1     1 1997     1   4 59.06     2
## 1965  1 1997     8   1 14.96     1
## 2225  1 1997    12   2 26.48     1
## 2     2 1997     1   4 75.11     2
## 3     3 1997     1   1  6.79     1
## 4     4 1997     1   1 13.97     1
```

```
ym <- expand.grid(year = 1997:1998,
                  month = 01:12,
                  id = unique(df$id))
ym <- ym[!(ym$year == 1998 & ym$month >6),]
head(ym)
```

```
##   year month id
## 1 1997     1  1
## 2 1998     1  1
## 3 1997     2  1
## 4 1998     2  1
## 5 1997     3  1
## 6 1998     3  1
```

```
df <- merge(df, ym, by = c("id","year","month"), all = TRUE)
miss.rw <- is.na(df$qty)
df[miss.rw, 4:6] <- 0
head(df)
```

```
##   id year month qty  expd trips
## 1  1 1997     1   4 59.06     2
## 2  1 1997     2   0  0.00     0
## 3  1 1997     3   0  0.00     0
## 4  1 1997     4   0  0.00     0
## 5  1 1997     5   0  0.00     0
## 6  1 1997     6   0  0.00     0
```

```
# now we should have the dataset we need; double check to make sure that every consumer
 is in every period


# ====== Section 3.1: recency ======
# use repetition statement, such as a "for-loop", to generate a recency measure for each
consumer
#   in each period. Hint: if you get stuck here, take a look at Example 3 when we talked
about "for-loops"
#   call it df$recency

df$start <- ifelse(df$qty != 0, 1, 0)
df$recency = NA

suppressWarnings(for (i in 1:nrow(df)) {
  temp = max(which(df$start[1:i-1] == 1 & df$id[1:i-1] == df$id[i]))
  df$recency[i] = i - temp
})
options(warn = -1)
warnings(5)
df$recency[df$year == 1997 & df$month == 1] <- NA
head(df)
```

```
##   id year month qty  expd trips start recency
## 1  1 1997     1   4 59.06     2     1      NA
## 2  1 1997     2   0  0.00     0     0       1
## 3  1 1997     3   0  0.00     0     0       2
## 4  1 1997     4   0  0.00     0     0       3
## 5  1 1997     5   0  0.00     0     0       4
## 6  1 1997     6   0  0.00     0     0       5
```

```
# ====== Section 3.2: frequency ======
# first define quarters and collapse/merge data sets
#   quarters should be e.g. 1 for January-March, 1997, 2 for April-June, 1997, ...
#   and there should be 6 quarters in the 1.5-year period
#   Next, let's define frequency purchase occasions in PAST QUARTER
#   Call this df$frequency

df$quarter <- ifelse(df$year > 1997, 4 + ceiling(df$month/3), ceiling(df$month/3) )
head(df)
```

```
##   id year month qty  expd trips start recency quarter
## 1  1 1997     1   4 59.06     2     1      NA       1
## 2  1 1997     2   0  0.00     0     0       1       1
## 3  1 1997     3   0  0.00     0     0       2       1
## 4  1 1997     4   0  0.00     0     0       3       2
## 5  1 1997     5   0  0.00     0     0       4       2
## 6  1 1997     6   0  0.00     0     0       5       2
```

```
for (i in 1:1000) {
  for (q in 2:6) {
    df$frequency[df$id == i & df$quarter == q] <-
    sum(df$trips[df$id == i & df$quarter == q-1])
  }
}

head(df)
```

```
##   id year month qty  expd trips start recency quarter frequency
## 1  1 1997     1   4 59.06     2     1      NA       1        NA
## 2  1 1997     2   0  0.00     0     0       1       1        NA
## 3  1 1997     3   0  0.00     0     0       2       1        NA
## 4  1 1997     4   0  0.00     0     0       3       2         2
## 5  1 1997     5   0  0.00     0     0       4       2         2
## 6  1 1997     6   0  0.00     0     0       5       2         2
```

```
# ====== Section 3.3: monetary value ======
# average monthly expenditure in the months with trips (i.e. when expenditure is nonzer
o)
#   for each individual in each month, find the average expenditure from the beginning t
o
#   the PAST MONTH. Call this df$monvalue

df$exp_month = ifelse(df$expd == 0, 0, 1)
head(df)
```

```
##   id year month qty  expd trips start recency quarter frequency exp_month
## 1  1 1997     1   4 59.06     2     1      NA       1        NA         1
## 2  1 1997     2   0  0.00     0     0       1       1        NA         0
## 3  1 1997     3   0  0.00     0     0       2       1        NA         0
## 4  1 1997     4   0  0.00     0     0       3       2         2         0
## 5  1 1997     5   0  0.00     0     0       4       2         2         0
## 6  1 1997     6   0  0.00     0     0       5       2         2         0
```

```
for (i in 1:nrow(df)) {
  sum_exp = NA
  sum_mon = NA
  sum_exp <- sum(df$expd[which(df$exp_month[1:i-1] == 1 & df$id[1:i-1] == df$id[i])])
  sum_mon <- sum(df$exp_month[which(df$exp_month[1:i-1] == 1 & df$id[1:i-1] == df$id
[i])])

  df$monval[i] = sum_exp / sum_mon
}
df$monval[df$year == 1997 & df$month == 1] <- NA
head(df)
```

```
##   id year month qty  expd trips start recency quarter frequency exp_month
## 1  1 1997     1   4 59.06     2     1      NA       1        NA         1
## 2  1 1997     2   0  0.00     0     0       1       1        NA         0
## 3  1 1997     3   0  0.00     0     0       2       1        NA         0
## 4  1 1997     4   0  0.00     0     0       3       2         2         0
## 5  1 1997     5   0  0.00     0     0       4       2         2         0
## 6  1 1997     6   0  0.00     0     0       5       2         2         0
##   monval
## 1     NA
## 2  59.06
## 3  59.06
## 4  59.06
## 5  59.06
## 6  59.06
```

```
# ====== Section 4: Targeting using RFM ======
# now combine these and construct an RFM index
#   You only need to run this section.

b1 <- -0.05
b2 <- 3.5
b3 <- 0.05

df$index <- b1*df$recency + b2*df$frequency + b3*df$monval
head(df)
```

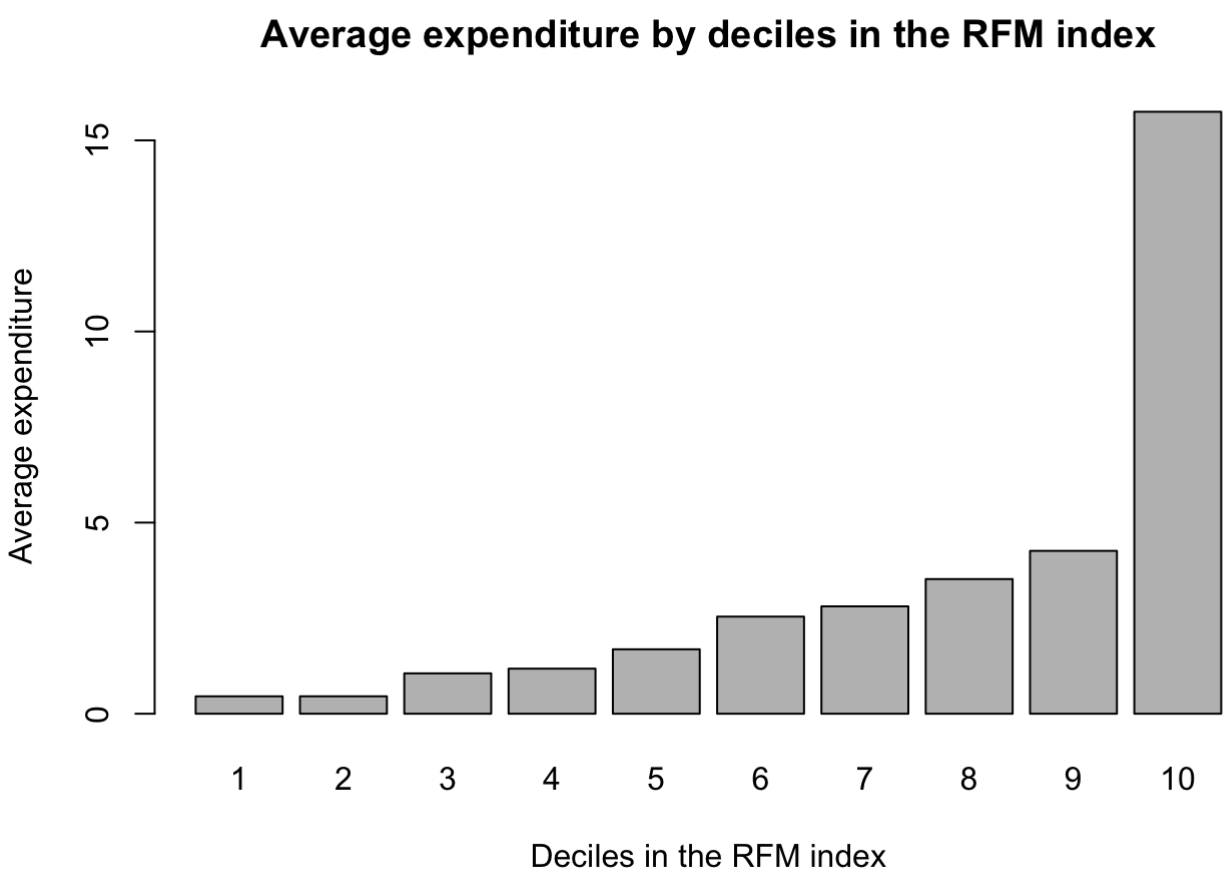
```
##   id year month qty  expd trips start recency quarter frequency exp_month
## 1  1 1997     1   4 59.06     2     1      NA       1        NA         1
## 2  1 1997     2   0  0.00     0     0       1       1        NA         0
## 3  1 1997     3   0  0.00     0     0       2       1        NA         0
## 4  1 1997     4   0  0.00     0     0       3       2         2         0
## 5  1 1997     5   0  0.00     0     0       4       2         2         0
## 6  1 1997     6   0  0.00     0     0       5       2         2         0
##   monval index
## 1     NA    NA
## 2  59.06    NA
## 3  59.06    NA
## 4  59.06 9.803
## 5  59.06 9.753
## 6  59.06 9.703
```

```
# validation: check whether the RFM index predict customer purchase patterns
# Order your sample (still defined by keys of consumer-year-month) based on the RFM inde
x.
#   Split your sample into 10 groups. The first group is top 10% in terms of
#   the RFM index; second group is 10%-20%, etc.
# Make a bar plot on the expected per-trip revenue that these consumers generate and com
ment on
#   whether the RFM index help you segment which set of customers are "more valuable"

df_check <- df[order(df$index),]
df_check$qtl <- as.numeric(cut(df_check$index,quantile(df_check$index, seq(0,1,0.1), na.
rm = T)))
ave_expd <- aggregate(expd~qtl, df_check, FUN = mean)
head(ave_expd)
```

```
##   qtl      expd
## 1   1 0.4552292
## 2   2 0.4560691
## 3   3 1.0561685
## 4   4 1.1792426
## 5   5 1.6857564
## 6   6 2.5403748
```

```
barplot(ave_expd[,2],
        main = "Average expenditure by deciles in the RFM index",
        xlab = "Deciles in the RFM index",
        ylab = "Average expenditure",
        names.arg = ave_expd[,1])
```


Average expenditure by deciles in the RFM index
Average expenditure
0
5
10
15
1
2
3
4
5
6
7
8
9
10
Deciles in the RFM index