SWE sample data - Q3 data

Jae Ha

import data

```
setwd("D:/")
data <- read.csv("SWE sample data - Q3 data.csv",header = T)
d <- data.frame(data)
library(dplyr)
```

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

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

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

Consider only the rows with country_id = “BDV” (there are 844 such rows). For each site_id, we can compute the number of unique user_id’s found in these 844 rows. Which site_id has the largest number of unique users? And what’s the number?

```
#find the site with the most number of users
bdv <- filter(d, country_id == "BDV")
summary(bdv)
```

```
##                       ts          user_id     country_id    site_id
##  2019-02-01 20:54:24:  2   LCC346 : 15   BDV:844    5NPAU  :717
##  2019-02-06 12:41:15:  2   LC3DB6 : 12   HVQ:  0    NOOTG  :122
##  2019-02-01 00:47:58:  1   LC3973 : 11   K1R:  0    3POLC  :  5
##  2019-02-01 02:46:15:  1   LCCAC3 :  9   NVV:  0    EUZ/Q  :  0
##  2019-02-01 06:44:42:  1   LC31A6 :  7   QLT:  0    GVOFK  :  0
##  2019-02-01 07:36:56:  1   LC3FFF :  7   TL6:  0    JSUUP  :  0
##  (Other)            :836   (Other):783   XA7:  0    (Other):  0
```

```
#returns the number of unique users for the site 5NPA
NPAU <- filter(bdv, site_id == "5NPAU")
length(unique(NPAU$user_id))
```

```
## [1] 544
```

Between 2019-02-03 00:00:00 and 2019-02-04 23:59:59, there are four users who visited a certain site more than 10 times. Find these four users & which sites they (each) visited more than 10 times. (Simply provides four triples in the form (user_id, site_id, number of visits) in the box below.)

```r
#create a subset of the data that only include data within the specified timeframe
data$ts <- as.POSIXct(data$ts)
feb3to4 <- filter(data, ts >= "2019-02-03 00:00:00 EST", ts < "2019-02-05 00:00:00 EST")
#return the users that have logs of more than 10
summary(feb3to4)[1:4,2]
```

```
##
## "LC06C3 : 26  " "LC3A59 : 26  " "LC3C9D : 18  " "LC3C7E : 15  "
```

```r
LC06C3 <- filter(feb3to4, user_id == "LC06C3")
LC3A59 <- filter(feb3to4, user_id == "LC3A59")
LC3C9D <- filter(feb3to4, user_id == "LC3C9D")
LC3C7E <- filter(feb3to4, user_id == "LC3C7E")
#returns the site_id of the site the users visited the most
summary(filter(feb3to4, user_id == "LC06C3"))[1,4]
```

```
## [1] "N0OTG  :25  "
```

```r
summary(filter(feb3to4, user_id == "LC3A59"))[1,4]
```

```
## [1] "N0OTG  :26  "
```

```r
summary(filter(feb3to4, user_id == "LC3C9D"))[1,4]
```

```
## [1] "N0OTG  :17  "
```

```r
summary(filter(feb3to4, user_id == "LC3C7E"))[1,4]
```

```
## [1] "3POLC  :15  "
```

For each site, compute the unique number of users whose last visit (found in the original data set) was to that site. For instance, user “LC3561”’s last visit is to “N0OTG” based on timestamp data. Based on this measure, what are top three sites? (hint: site “3POLC” is ranked at 5th with 28 users whose last visit in the data set was to 3POLC; simply provide three pairs in the form (site_id, number of users).)

```r
# Grabs the list of unique user IDs
user <- unique(d$user_id)

#Fliters the data to get all visitation data for first 2 users on the list
site1 <- filter(d, d$user_id == user[1])
site2 <- filter(d, d$user_id == user[2])
#gets the last visited sites for the first 2 users
last1 <- site1[nrow(site1),]
last2 <- site2[nrow(site2),]
#binds the rows together to get a merged data frame
last_site <- bind_rows(last1,last2)

#for loop binds the rest of the user's last visited site to the previously merged dataframe
for(i in seq(3, length(user))){
  site <- filter(d, d$user_id == user[i])
  last <- site[nrow(site),]
  last_site <- bind_rows(last_site,last)
}
#shows the top 3 most popular last visited sites
summary(last_site)[1:3,4]
```

```
##
## "5NPAU  :992  " "N0OTG  :561  " "QGO3G  :289  "
```

For each user, determine the first site he/she visited and the last site he/she visited based on the timestamp data (Please include users who visited the site only once.). Compute the number of users whose first/last visits are to the same website. What is the number?

```
#Fliters the data to get all visitation data for first 2 users on the list
site1 <- filter(d, d$user_id == user[1])
site2 <- filter(d, d$user_id == user[2])
#gets the first visited sites for the first 2 users
fist1 <- site1[1,]
fist2 <- site2[1,]
#binds the rows together to get a merged data frame
first_site <- bind_rows(last1,last2)

#for loop binds the rest of the user's first visited site to the previously merged dataframe
for(i in seq(3, length(user))){
  site <- filter(d, d$user_id == user[i])
  first <- site[1,]
  first_site <- bind_rows(first_site,first)
}
#combines the list of first visited sites and last visited sites
comb_fl <- bind_cols(first_site, last_site)

#returns the list of IDs that have the same first and last visited sites
f_equal_l <- filter(comb_fl, comb_fl$site_id == comb_fl$site_id1)

#the number of users whose first/last visits are to the same website including users who visited the si
length(f_equal_l)
```

```
## [1] 8
```

For each site, count the following numbers: (A) the number of unique users who have visited at least two different countries (B) the number of all unique users. (For example, user “LC3450” has visited 3 countries, “BDV”, “QLT”, and “TL6” with “5NPAU”, “5NPAU”, and “N0OTG”, respectively. When counting (A), both “5NPAU” and “N0OTG” need to consider the user.) Please calculate the ratio B/A for each site and list top three sites and the corresponding ratio.

```
#finds the list of users that have been to more than 2 countries
user2_total <- user
for(i in seq(length(user))){
  if(length(unique(filter(d, d$user_id == user[i])$country_id))<2){
    user2_total <- user2_total[-i]
  }
```

```
}
#finds the list of unique sites
site <- unique(d$site_id)

#stores the number of site users that have been to more than 2 countries in a vector
num_site_users_2countries <- c()
num_site_users_total <- c()
for(n in 1:length(site)){
  sitelog <- filter(d, d$site_id == site[n])
  users_in_site <- unique(sitelog$user_id)
  #returns the number of site users that have been to more than 2 countries
  num_site_users_2countries <- c(num_site_users_2countries, length(intersect(users_in_site, user2_total
  num_site_users_total <- c(num_site_users_total, length(users_in_site))
}

A <- num_site_users_2countries
B <- num_site_users_total
#calulate B/A and show with correspoonding site_id
bdiva <- data.frame(B/A, row.names = site)
bdiva
```

```
##            B.A
## N0OTG 1.788043
## QGO3G 1.928962
## GVOFK 2.034483
## 3POLC 1.384615
## 5NPAU 1.722309
## RT9Z6 1.000000
## JSUUP      Inf
## EUZ/Q 1.000000
```