Data Science - Practice 6 (Data Preparation II)

Make sure you not only just "write down" the R code but also "explain the answer with your own language". All answers without explanation will not be accepted.

Problem

Import 'UserConsump.RData' and 'AirBnB_data.csv'. The prior data is about user's OTT consumption in 2019. Here, the users are classified in one of three groups, Low (0), Medium (1) and High Consumption (2). The latter data is about the AirBnB's host data. The latter data is AirBnB service records obtained from Netherlands.

< Question 1 – UserConsump data>

Is there an issue of duplicated "identifier"? If so, write down R code that finds the list of duplicated identifiers.

Expected Result

(4)

```
  [1] 3232266502 3232266504 3232266505 3232266511 3232266514 3232266526 3232266528 3232266532 3232266549 3232266568 3232266573 3232266584 3232266586
 [14] 3232266591 3232266596 3232266602 3232266616 3232266660 3232266683 3232266694 3232266721 3232266725 3232266746 3232266759 3232266787 3232266794
 [27] 3232266799 3232266802 3232266808 3232266809 3232266811 3232266813 3232266816 3232266819 3232266824 3232266840 3232266845 3232266846 3232266847
 [40] 3232266849 3232266856 3232266863 3232266879 3232266883 3232266886 3232266888 3232266889 3232266897 3232266898 3232266899 3232266901 3232266902
 [53] 3232266905 3232266908 3232266928 3232266933 3232266996 3232267018 3232267023 3232267035 3232267052 3232267100 3232267115 3232267143 3232267165
 [66] 3232267171 3232267174 3232267181 3232267198 3232267222 3232267248 3232267249 3232267259 3232267303 3232267368 3232267369 3232267394 3232267418
 [79] 3232267504 3232267534 3232267535 3232267542 3232267543 3232267545 3232267548 3232267552 3232267556 3232267559 3232267560 3232267562 3232267568
 [92] 3232267569 3232267573 3232267575 3232267576 3232267578 3232267579 3232267581 3232267582 3232267588 3232267590 3232267591 3232267592 3232267594
[105] 3232267605 3232267606 3232267607 3232267610 3232267611 3232267612 3232267616 3232267617 3232267618 3232267622 3232267624 3232267626 3232267633
[118] 3232267638 3232267642 3232267645 3232267670 3232267674 3232267688 3232267691 3232267693 3232267694 3232267697 3232267701 3232267702 3232267706
[131] 3232267715 3232267717 3232267727 3232267734 3232267755 3232267768 3232267770 3232267771 3232267805 3232267834 3232267842 3232267919 3232268026
[144] 3232268031 3232268045 3232268052 3232268054 3232268056 3232268058 3232268061 3232268063 3232268065 3232268071 3232268074 3232268077 3232268078
[157] 3232268080 3232268081 3232268088 3232268092 3232268095 3232268101 3232268106 3232268107 3232268119 3232268121 3232268127 3232268143 3232268146
[170] 3232268229 3232268236 3232268290 3232268295 3232268300 3232268302 3232268307 3232268313 3232268320 3232268329 3232268337 3232268339 3232268340
[183] 3232268342 3232268344 3232268347 3232268349 3232268356 3232268357 3232268362 3232268364 3232268370 3232268380 3232268381 3232268383 3232268392
[196] 3232268403 3232268406 3232268407 3232268408 3232268420 3232268433 3232268437 3232268439 3232268440 3232268444 3232268449 3232268453 3232268455
[209] 3232268456 3232268459 3232268460 3232268462 3232268463 3232268464 3232268465 3232268466 3232268467 3232268470 3232268472 3232268476 3232268479
[222] 3232268482 3232268490 3232268493 3232268507
```

< Question 2 – UserConsump data>

It has been found that duplicated identifier issue happened because of receiving multiple records from the same users. For this case, multiple records of the same identifier should be all "added", except "cluster" variable. In case of cluster, the largest value should be selected. Knowing the reason of this issue, solve it and show the fixed data set. Below is the expected result when all operations are successfully completed. (Hint: %in% is an operation used for checking whether the multiple elements are included in another vector or not. For example "x %in% y" will check whether elements of y are included in x or not.)

Expected Result

```
> length(unique(UserConsump$src_ip_numeric))
[1] 973
> nrow(UserConsump)
[1] 973
```

< Question 3 – UserConsump data>

Use str_detect() to create a new variable called "UserConsump.Google", which only contains Google services' data consumptions. First row of the UserConsump.Google is shown below.

Expected Result

```
head(UserConsump.Google,1)
Google_data_occupation GoogleDocs_data_occupation GoogleDrive_data_occupation
               9847.76                          0                           0
GoogleHangoutDuo_data_occupation GoogleMaps_data_occupation GooglePlus_data_occupation
                               0                          0                          0
GoogleServices_data_occupation
                      3910.738
```

< Question 4 – UserConsump data>

Below is the summary of UserConsump.Google's GoogleHanghoutDuo_data_consumption. Here, most of values are 0. Is this make sense? Explain.

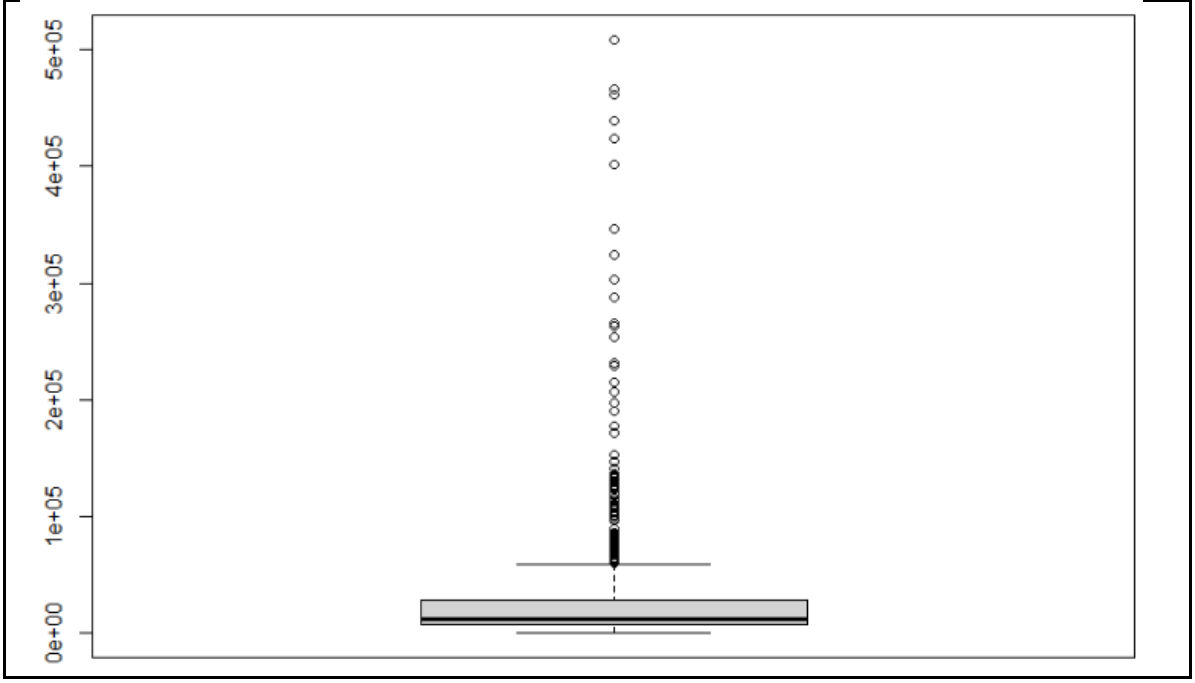
Expected Result

```
> summary(UserConsump.Google$GoogleHangoutDuo_data_occupation)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0       0       0    8324       0 5717984
```

< Question 5 – UserConsump data>

Below is the boxplot of UserConsump.Google's GoogleServices_data_occupation. Explain what sort of information we can notice from this. Anything strange or special?

Expected Result



< Question 6 – AirBnB data>

Below is the first 6 rows of our data. What are the number of rows and the number of unique host_id (host's ID)? Is this acceptable? Explain with your own language.

Expected Result

```
> head(AirBnB)
  host_id host_name host_since_year host_since_anniversary      id                neighbourhood_cleansed      city
1    1662     Chloe            2008                  8/11  304958                          Westerpark Amsterdam
2    3159    Daniel            2008                  9/24    2818 Oostelijk Havengebied - Indische Buurt Amsterdam
3    3718    Britta            2008                 10/19  103026               De Baarsjes - Oud-West Amsterdam
4    4716    Stefan            2008                 11/30  550017                        Centrum-Oost Amsterdam
5    5271     Tyler            2008                 12/17 4728389                       Centrum-West Amsterdam
6    5271     Tyler            2008                 12/17 5500954                       Centrum-West Amsterdam
          state zipcode     country latitude longitude property_type       room_type accommodates bathrooms bedrooms
1 North Holland    1053 Netherlands 52.37302  4.868461     Apartment Entire home/apt            4         2        2
2 North Holland         Netherlands 52.36575  4.941419     Apartment    Private room            2         1        1
3 Noord-Holland    1053 Netherlands 52.36939  4.866972     Apartment Entire home/apt            4         1        1
4 North Holland    1017 Netherlands 52.36191  4.888050     Apartment Entire home/apt            2         1        1
5 Noord-Holland 1016 AM Netherlands 52.37153  4.887057     Apartment Entire home/apt            6         1        2
6            NH 1016 AM Netherlands 52.37136  4.888072     Apartment    Private room            4         1        1
  beds bed_type price guests_included extra_people minimum_nights host_response_time host_response_rate
1    2 Real Bed   130              4           10              4       within a day                0.8
2    2 Real Bed    59              1           10              3     within an hour                  1
3    1 Real Bed    95              2           25              3 within a few hours                  1
4    1 Real Bed   100              1           10              2       within a day                  1
5    2 Real Bed   250              2           25              2       within a day               0.89
6    1 Real Bed   140              2           25              2       within a day                0.9
  number_of_reviews review_scores_rating review_scores_accuracy review_scores_cleanliness review_scores_checkin
1                11                   98                     10                       10                      9
2               108                   97                     10                       10                     10
3                15                   92                      9                        9                     10
4                20                   97                     10                       10                     10
5                 1                  100                      8                       10                      8
6                 0                   NA                     NA                       NA                     NA
  review_scores_communication review_scores_location review_scores_value
1                          10                      10                   10
2                          10                       9                   10
3                          10                       9                    9
4                          10                      10                   10
5                          10                      10                    6
6                          NA                      NA                   NA
```

< Question 7 – AirBnB data>

To solve this issue of "duplicated" identifiers, let's create a new variable called "host_cust", which combines host_id and id with a separator "-" (ex. 1662-304958). Will this new variable solve this issue of "duplicated" identifier? If not, is it acceptable? Explain.

< Question 8 – AirBnB data>

As you know, the history of AirBnB service started very recently. To check the annual trend of AirBnB service, write down a R code that counts the number of advent of new AirBnB service providers in each year. Describe the expected result shown below. Is there anything strange? If so, is it acceptable? Explain.

Expected Result

```
2008 2009 2010 2011 2012 2013 2014 2015
   5   19  129  455 1306 2006 2095  363
```

< Question 9 – AirBnB data>

host_since_year and host_since_anniversary tells us the year and date of the service launch. Using this information, create a new variable called anni_date, which is a date type variable, and a new variable called age_days, which measure the number of days passed ever since the service launch (Hint: use

Sys.Date() function to obtain the today's date). Below is the expected result obtained in April 14th, 2022. (age_days should return the number of days based on the "current" date that code is running.)

Expected Result

```
> head(AirBnB[,c("host_id","host_since_year","host_since_anniversary",
+                "anni_date","age_days")])
  host_id host_since_year host_since_anniversary  anni_date  age_days
1    1662            2008                  8/11 2008-08-11 4994 days
2    3159            2008                  9/24 2008-09-24 4950 days
3    3718            2008                 10/19 2008-10-19 4925 days
4    4716            2008                 11/30 2008-11-30 4883 days
5    5271            2008                 12/17 2008-12-17 4866 days
6    5271            2008                 12/17 2008-12-17 4866 days
```

< Question 10 – AirBnB data>

(1) Find the number of missing values for columns related to "review". (2) We decided that we will keep the samples only if any of the "review" columns are not missing (In other words, if the value of one of the review columns is missing, that sample is excluded.) Create a new Data.Frame called AirBnB_1 and write down the number of observations.

Expected Result

- before

```
        number_of_reviews        review_scores_rating      review_scores_accuracy
                        0                        1698                        1709
review_scores_cleanliness       review_scores_checkin review_scores_communication
                     1709                        1708                        1711
   review_scores_location         review_scores_value
                     1709                        1711
```

- after

```
        number_of_reviews        review_scores_rating      review_scores_accuracy
                        0                           0                           0
review_scores_cleanliness       review_scores_checkin review_scores_communication
                        0                           0                           0
   review_scores_location         review_scores_value
                        0                           0
```