Homework 3: Processing indoor location data
Exploratory Data Analysis

Prof: Yannet Interian

Due: July 27th 2015

1 Introduction

In this homework you will transform raw data in *json* format into a data set without missing values and in a tabular format ready for further data analysis. The work is divided into the following parts:

1. Transforming data in *json* format to data frame format. (two R scripts).
2. Define metrics to rank and select wifi sensors (two scripts)
3. Replacing missing data (NA)

To get started with the exercise, you will need to download the datasets from canvas. These are the files included in this exercise:

```
test.json    # small data set for testing
indoor-location-training.json    # data set for indoor location json format
indoor-location-test.csv    #  data set for indoor location csv format
```

You will be submitting five R scripts with the following names. Please write comments in your code.

```
get_wifi.R   # for you to complete
indoor2dataframe.R # for you to complete
select_wifi_mean.R    # for you to complete
select_wifi_coverage.R    # for you to complete
replace_NAs.R    # for you to complete
```

1.1 Data

This is how the beginning of your json file look like. You may want to read about json format here `http://www.json.org`

```
cat test.json | less
```


```
[
  {
    "Data": [
      {
        "Properties": [
          {
            "Value": "OFFSITE",
            "Name": "room"
          }
        ],
        "Quantities": [
          {
            "Name": "c0:83:0a:60:07:31",
            "Number": -88,
            "Unit": "dB"
          },
          {
            "Name": "60:c3:97:10:d4:f1",
            "Number": -92,
            "Unit": "dB"
          },
          {
            "Name": "00:19:e3:fa:36:57",
            "Number": -95,
            "Unit": "dB"
          }
        ],
        "Source": "wifi",
        "MTime": "2013-07-29T04:00:00.245Z"
      },
....
```


You are able to read this data by using the following code.

```
library(rjson) # you may need to install this
json_data <- fromJSON(file="test.json", method='C')
## looking at the data
class(json_data)
[1] "list"
> names(json_data[[1]])
[1] "Data"       "Properties" "App"        "Build"      "User"
[6] "MTime"
> class(json_data[[1]]$Data)
[1] "list"
> names(json_data[[1]]$Data[[1]])
[1] "Properties" "Quantities" "Source"     "MTime"
....
```

2 Transforming *json* into tabular format

2.1 Part 1 (5 points)

In this exercise you write the function get_wifi (save the function in $get\_wifi.R$) that given a $json$ file with the format described above returns a vector of wifi names sorted alphabetically. The wifi names are found when "Source" has value "wifi". In this small measurement the names are "c0:83:0a:60:07:31", "60:c3:97:10:d4:f1", "00:19:e3:fa:36:57". For the file test.json you will get these names:

```
source("get_wifi.R")
> get_wifi("test.json")
[1] "00:19:e3:fa:36:57" "00:26:f2:fc:36:c4" "28:16:2e:9d:a8:89"
[4] "60:c3:97:10:d4:f1" "c0:83:0a:60:07:31" "c8:d7:19:35:fd:3d"

> get_wifi("indoor-location-training.json")
[1] "00:15:ff:45:c1:99" "00:1c:b3:ad:7a:25" "00:21:29:86:9d:57"
 [4] "00:21:29:b9:26:b3" "00:22:a4:c2:a5:e1" "00:26:f3:60:4f:48"
 [7] "20:4e:7f:0b:ff:2a" "28:16:2e:3a:e5:69" "36:46:9a:09:cf:74"
[10] "56:04:a6:ca:65:2b" "ac:5d:10:92:a0:b9" "b0:e7:54:52:d1:c1"
[13] "b0:e7:54:64:62:f9" "c0:14:3d:8b:71:72" "c8:bc:c8:fd:54:6d"
[16] "f8:d1:11:5a:a0:54" "f8:d1:11:5a:c4:8c" "f8:d1:11:5a:c8:c3"
[19] "f8:d1:11:5b:52:c9"
```

A prototype of the function is as follows

```
get_wifi <- function(json_file) {

}
```

2.2 Part 2 (5 points)

Note: this is a challenging exercise you should leave it for the end.
Write a function that reads the input file and outputs a data frame with the following data (you will need to use the function for Part 1). The first column is the "Source", which in this case is "wifi". The second column is the value in "Properties" (the header is "Room"). The third column is the value from "MTime" (the header is Time). All the other columns have as a header the name of the wifi ("Name" in "Quantities") and the strength of the signal ("Number" in "Quantities"). If a particular wifi is not present give the value NA. The output of running your code for test.json should be as follows:

```
source("indoor2dataframe.R")
indoor2dataframe("test.json")
Source    Room                      Time 00:19:e3:fa:36:57 00:26:f2:fc:36:c4
1   wifi OFFSITE 2013-07-29T04:00:00.245Z               -95                NA
2   wifi OFFSITE 2013-07-29T04:00:04.264Z                NA              -102
3   wifi OFFSITE 2013-07-29T04:00:08.282Z                NA               -98
4   wifi OFFSITE 2013-07-29T04:00:12.308Z               -99                NA
5   wifi OFFSITE 2013-07-29T04:00:16.337Z               -92                NA
6   wifi OFFSITE 2013-07-29T04:00:20.377Z                NA               -97
7   wifi OFFSITE 2013-07-29T04:00:24.458Z                NA               -97
8   wifi OFFSITE 2013-07-29T04:00:28.446Z                NA                NA
  28:16:2e:9d:a8:89 60:c3:97:10:d4:f1 c0:83:0a:60:07:31 c8:d7:19:35:fd:3d
1                NA               -92               -88                NA
2               -97               -89               -87               -96
3                NA               -93               -87                NA
4               -98               -93               -85                NA
5               -96               -91               -85                NA
6                NA                NA               -86                NA
7                NA               -90               -88                NA
8                NA                NA               -87                NA
```

A prototype of the function is as follows

```
indoor2dataframe <- function(json_file) {
```

```
}
```

3 Define metrics to rank and select wifi sensors

For this exercise you will write the functions $select\_wifi\_mean$ and $select\_wifi\_coverage$ and save them in the files **select_wifi_mean.R** and **select_wifi_coverage.R**. These functions select a subset of the wifi based on different criteria. You will use the file **indoor-location-test.csv** to test your code.

As we discussed in class we want select a set of wifi sensors that is going to help our prediction problem. These are some of the properties with want.

- High signal in some rooms
- High level of coverage

3.1 Part 1 (5 points)

Write a function in R (**select_wifi_mean**) that selects wifi using the following criteria.

- Compute the mean value (removing NA's) for each wifi and each room.
- For every wifi, count the number of rooms in which the mean for the value of the signal is at least -90.
- Return wifi with at least 7 rooms with mean greater than -90.

A prototype of the function is as follows

```
# computes mean for each wifi in each room
# uses this to calculate # of rooms in which wifis have avg signals > -90
# returns the wifis with more than 7 rooms with avg > -90
select_wifi_mean <- function(file_csv) {

}
### Results from running
source("select_wifi_mean.R")
select_wifi_mean("indoor-location-test.csv")
[1] "b0.e7.54.64.62.f9" "c0.14.3d.8b.71.72" "f8.d1.11.5a.a0.54"
[4] "f8.d1.11.5a.c4.8c" "f8.d1.11.5a.c8.c3" "f8.d1.11.5b.52.c9"
```

3.2 Part 2 (5 points)

Write a function (**select_wifi_coverage**) that selects wifi using the following criteria.

- Compute percent coverage (percent of values that are not NA) for each wifi and each room.
- For every wifi, count the number of rooms with greater than 70% coverage.
- Return a vector of wifi names with at least 7 rooms with greater than 70% coverage.

A prototype of the function is as follows

```
# Select wifi with at least 7 rooms with at least 70\% coverage.
select_wifi_coverage <- function(file_csv) {

}
###
source("select_wifi_coverage.R")
select_wifi_coverage("indoor-location-test.csv")
[1] "b0.e7.54.64.62.f9" "f8.d1.11.5a.a0.54" "f8.d1.11.5a.c4.8c"
[4] "f8.d1.11.5a.c8.c3" "f8.d1.11.5b.52.c9"
```

4 Replacing missing data (NA) (5 points)

For this exercise you will write your function in **replace_NAs.R**. The function takes as an input a csv file (use the file **indoor-location-test.csv** to test your code) and a vector with a subset of the wifi columns and returns a data frame with no missing values. The function replaces missing values following the heuristic described below.

Why do we have NAs? In class we discussed two theories:

a. The room is very far away from the access point (router) so the signal is lost (very weak).

b. The room is reasonably close to the access point (router) but the signal was lost for some other reason.

How can we figure out if it is a. or b.? How do we handle the two cases?

Write a function (**replace_NAs.R**) that given a data frame with NAs replaces the NAs using the following heuristic.

- Compute the mean value (removing NA's) for each wifi and each room.
- If the mean value is larger than -90 substitute with the mean value otherwise substitute with -100. (Hint: do one room at a time)

A prototype of the function is as follows

```
# Compute the mean value (removing NA's) for each wifi and each room.
# If the mean value is larger than -90 substitute with the mean value
# otherwise substitute with -100.

replace_NAs <-function(file_csv, wifi=c("b0.e7.54.64.62.f9","f8.d1.11.5a.a0.54")) {

}
###
a <- replace_NAs("indoor-location-test.csv",
                 c("b0.e7.54.64.62.f9","f8.d1.11.5a.a0.54"))
> summary(a)
  Source           Room                              Time       b0.e7.54.64.62.f9
 wifi:10264   BED5   :6312   2013-06-19T03:24:37.593Z:    1   Min.   :-103.00
              BED1   :1673   2013-06-19T03:24:41.592Z:    1   1st Qu.:-100.00
              KITCHEN: 969   2013-06-19T03:24:45.592Z:    1   Median : -95.00
              LIVING : 629   2013-06-19T03:24:49.592Z:    1   Mean   : -90.37
              BATH1  : 185   2013-06-19T03:24:53.592Z:    1   3rd Qu.: -82.00
              DINING : 158   2013-06-19T03:24:57.592Z:    1   Max.   : -64.00
              (Other): 338   (Other)                 :10258
 f8.d1.11.5a.a0.54
 Min.   :-102.00
 1st Qu.: -97.00
 Median : -91.00
 Mean   : -90.51
 3rd Qu.: -85.00
 Max.   : -34.00
```

5 Testing your code

I will deduct 10% of your grade if your code is not executable exactly in the fashion mentioned in this document. Please use the test code (test_hw2.R)

provided with this assignment and make sure your code passes the tests before submitting your homework. I will be testing your code in a unix machine (a mac) please make sure your code runs properly under linux.

Before submitting, place your R files, "test_hw3.R" and the data dir "exp-auc" in your working directory and run the following commands in R.

```
library("testthat")
test_dir(".")
# You should get this output
Running unit tests

Testing get_wifi.R
.
Testing indoor2dataframe.R
...
Testing select_wifi_mean.R
.
Testing select_wifi_coverage.R
.
Testing replace_NAs.R
.
```