IMT573 Problem Set 1: Basic R and Data Exploration (80pt)

Your name:

Deadline: Wed, Oct 14th 5pm

Instructions

These 80 points will give you 8 points of the final grade.

1. Be sure to include well-documented (e.g. commented) code chucks, figures, tables, and clearly written text explanations as necessary. All figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation.
2. Don't output irrelevant, or too much relevant information. A few figures is helpful. A few thousand figures is noise.
3. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment. In particular, note that Stack Overflow is licensed as Creative Commons (CC-BY-SA). This means you have to attribute any code you pick from SO (a link to the question/answer will normally do).
4. Partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. Attempt each question and document your reasoning process even if you cannot get a good answer! If you would like to include R code to show this process, but it does not run without errors you can do so with the `eval=FALSE` chunk option as follows:

    ```
    ```{r example chunk with a bug, eval=FALSE}
    a + b # these object don't exist
    # if you run this on its own it with give an error
    ```
    ```

5. When you have completed the assignment, check that your code runs and knits correctly when you click 'Knit'.
6. Working together is fun and useful but you have to submit your own work. Discussing the solutions and problems with your classmates is all right but do not copy-paste their solution! First understand, and thereafter create your own solution. Please list all your collaborators on the solution.

1 Basic R Programming (40pt)

This question asks you to perform a few basic programming tasks in R. The first part of it is very similar to the other traditional languages, such as C++, python or java; the second part requires you to handle vectors and vectorized operations.

These topics are mostly covered in Lander's book chapters 4 (computing), 8 (functions), 9.1 (if-else). See also R-notes https://otoomet.bitbucket.io/machinelearning-R.html#21_Base_language.

Unlike in almost every other task, in this you do not have to write text. Just code and its output is probably enough (you may still want to add explanations, comments, and such).

1.1 Computing (5pt)

1. Compute how many seconds are there in a year and assign it to a suitable variable. Thereafter print the result.

   Note: these will be big numbers. Consider using `prettyNum` for formatting these numbers in more human-readable form (see example).

   Example: compute seconds in hour

```
SIH <- 60*60
## standard formatting
cat("There are", SIH, "seconds in hour\n")

## There are 3600 seconds in hour

## better formatting
cat("There are", prettyNum(SIH, big.mark=",", scientific=FALSE), "seconds in hour\n")

## There are 3,600 seconds in hour
```

2. How long is a typical human lifetime in seconds? Use the seconds-in-year variable you created above to compute it.

3. Age of the Universe is 13.7 billion year. How old is the Universe in seconds?

1.2 Functions (15pt)

1. Write a function that takes two arguments: first name, and last name; and returns a sentence like "Hi, my name is *first name last name*, nice to meet you!".

   Note: the function should *return* this string, not print it!

   Hint: check out the function `paste`.

1.3 Vectors, loops, if/else (20pt)

While loops and if-else work in R in a fairly similar fashion as in other languages, vectors (here we use *atomic vectors*) are vectorized data types that are built-in into R but typically need additional libraries elsewhere.

This code snippet creates a vector with both positive and negative numbers:

```
set.seed(1)
v <- sample(10, 20, replace=TRUE) - 5
v

##  [1]  4 -1  2 -4 -3  2 -3 -2 -4  0  0  5  1  5  2  4  0  0  4  4
```

The following questions regard this vector:

1. Use a for-loop to extract only positive numbers from this vector.

   Hint: check out *Creating vectors in loop* in https://otoomet.bitbucket.io/machinelearning-R.html#214_Control_structures, and use if-else.

2. Perform the same task without the loop using logical indexing instead.

   Hint: check out *Logical indexing* https://otoomet.bitbucket.io/machinelearning-R.html#222_Logical_indexing.

3. Finally, consider three vectors:

```
v1 <- 9
v2 <- c(1,2)
v3 <- c(2,3,-4)
```

   Write a function that tests if the vectors have negative elements, and prints an appropriate message. Test the negativity of these three vectors to show the function works correctly.

   Hint: do not use loops. Use `if`/`else`, and check out the functions `any` and `all`.

2 Data Exploration (40pt)

Setup

This question asks you to perform basic exploratory analysis, directed toward a particular question about the dataset (or flying in general).

The data itself lives in package *nycflights13*. You may need to install both.

```
data(flights, package="nycflights13")
```

You may also want to read the help about the data (hint: do `?flights`).

2.1 Exploring the NYC flights data

In this dataset we ask you to perform basic descriptive analysis on actual data. Knowledge of the popular data manipulation and visualization packages like `dplyr` and `ggplot2` are an advantage here, but you can also just consult Lander's book Ch 5.1 for data frames and Ch 7.1 for the basic plotting. Unfortunately Lander does not explain data frame indexing and subsetting, see the R notes https://otoomet.bitbucket.io/machinelearning-R.html (TBD)

1. Import data

   Load the data and describe in a short paragraph how the data was collected and what each variable represents.

2. Perform a basic inspection of the data:

   - How many distinct flights do we have in the dataset?
   - What are the variables (variable names) in the data?
   - How many missing values are there in each variable?

     Hint: you can find number of missings using a construct like `sum(is.na(data$variable))`. Check also out the `summary` function.

- Do you see any unreasonable values?

  Hint: check out `min` and `max` (and `range`) functions.

3. Formulate a reasonable question you may want to ask using this data. The question may be either about the flights, or about the dataset itself. The question should be something this data is well-designed to answer.

   Examples:

   - A question about flights: *Which airport, JFK or LGA, experience more delay?s*
   - A question about the dataset: *Does the dataset contain valid information through the whole year? Is there data missing for a certain period of time?*
   - A bad question: *How many airports are there in the US?* This dataset only contains airport codes where there were scheduled regular passenger flights from NYC in 2013. Some of the airport codes may also be wrong. It may not contain (it almost certainly does not) a comprehensive list of airport codes.

4. Explore data

   Explore the data from the viewpoint of the question you asked. You should do something along these lines (but it depends on the question):

   - Explain which variables are the most important ones from your question's perspective. Inspect these variables. Which values, ranges do you see? Do you see any irregularities, missings, implausible values?
   - Define precisely what is your question about. E.g. what does "more delays" mean? More often delayed at arrival? At departure? More frequent long delays? Longer average delay? How do you intend to handle early departures/arrivals?
   - Split the data into subsets to answer your question if your question contains some sort of comparison (e.g for JFK and LGA airports to compare delays)
   - In case of the time period example, you may e.g. compute the daily number of flights for each day, and look for the outliers. If you identify a few, you should explain what to do and how to inspect the outliers. No flights a particular day may mean that we have a data problem, but perhaps the airport was closed instead? (You are just asked to explain, not to actually check the FAA records).
   - Produce tables/graphs that answer your question. In case of the delays, you may compute average delays for each airport (or whatever measure you choose for delays). In case of time period, you may plot the number of flights for each day, or perhaps just make a small table that shows a suspicious day and a few ordinary days before and after the suspicious one.

     Make sure you explain what your tables/graphs show!

5. Write a conclusion based on your analysis. Can you answer this question? Do you have concerns regarding to your answer? Is additional analysis/different data needed?

   Comment on any ethical and/or privacy concerns you have with your analysis.