Introduction to Bioinformatics (236523)

HW 1 – Spring 2021

General instructions:

- Dead Line: 26/04/21 23:55.
- Submission according to published pairs only.
- The submission is via the course website.
- You should submit the markdown file (.Rmd) and its word/pdf version in a .ZIP format named according to next format:

<HW#>_<ID1>_<ID2>.zip . Please decide which ID number of the two partners will be first and keep this convention for the rest of the semester. For example: 398837676_234543234.zip.

Detailed instruction for the HW:

In this HW you will generate your own markdown file for the first time using the information regarding markdown at: https://r4ds.had.co.nz/r-markdown.html or/and at R Markdown: The Definitive Guide (bookdown.org).

The answer to the questions that involve code should be submitted as a single markdown file. Please make sure that the output of the markdown file as doc or pdf (applying the Knit) option will give us the question number in bold large font.

You are more than encouraged to use the tutorials' markdowns as a template.

We had heard that sometimes there is a temptation to use other students work and to submit it instead of investing the time of your own. Please pay attention that the home work is a very important step in acquiring qualification that you'll have to apply in your final project. Not only the knowledge and understanding of material, but also hands on with the RStudio, learning from mistakes, making your first steps in research etc. Therefore, we strongly encourage you to take some time for learning and to handle the HW mission together with your partner.

Question 1:

1.1. Construct a 4x6 matrix named mat1. The matrix should be filled with random numbers from uniform distribution at the range [1,20]. Use the runif() function to generate the numbers.
1.2. Sum each row using the apply() function.
1.3. Write a short function named tentative.normalization(). This function must be written without the "for" loops usage. The function receives a matrix and normalization parameter that defines whether the normalization will be by row or by column. The function will normalize by rows when the function gets npar=1 and by columns when it gets npar=2.
Normalization is performed by division of either row values (in normalization by row) by the maximal value in the row, or by division of column (in normalization by columns) by the maximal value in the column.
The function returns the normalized matrix.

Additional notes:
max() is a function from base R.
Pay attention what you divide when you divide a matrix by vector.
Pay attention that the normalized function dimensions are the same as the input matrix.
Hint: You can use apply() for various user defined functions. As very naïve example, if you want to apply square on the matrix rows you can use the **apply(data, 2, function(x){x^2})** (although you can just do data^2).

1.4. Run the tentative.normalization() function on the mat1 rows and append the result into the mat1.norm.rows variable. Do the same for the mat1 columns and append the result into the mat1.norm.columns variable.

1.5. Run the heatmap() function on the mat1.norm.rows and mat1.norm.columns. Make sure that the heatmap plot appears in the output when you run the code chunk at the markdown.

Question 2:

2.1. Download the "data1.csv" and "data2.csv" file from the course site.

The data1.csv file contains information regarding blood sugar levels in human (males and females) in two groups – the control group that received the placebo and following treatment by the HF123 drug. The data2.csv file contains subset of female patients that were tested for the drug effects.

Set the downloaded files path by file.path() function that allow to construct the path to a file from components in a platform-independent way.

Assign the file path to data1.file and data2.file variable in the form of:

data1.file <-file.path("path as you set it according the file.path function requirements")

data2.file <-file.path("path as you set it according the file.path function requirements")

Please read the table given in the data1.file into the "data1" variable using the read.csv() function. Preform the same operation for data2.file, assigning it into the "data2" variable.

2.2. What is the type of data structure of data1? Search for appropriate function that can provide you this information.

2.3. From the data2 create 2 variables – control and treatment, that will contain the blood sugar levels. Use the filter() function from dplyr package for this purpose.

2.4. Calculate the mean difference between the groups. Does this mean difference significant?

2.5. Build a single histogram for both treatment and control.

2.6. Build a boxplot for both treatment and control. Please make sure that your boxplot includes group names.

2.7. Perform the t-test on the treatment vs control group. What can you say about the treatment?

Now let's compare to data1 dataset.

2.7. Build a boxplot and perform the t-test on the treatment vs control group on the data1 dataset. Perform all the preliminary data manipulations that will allow you to perform the t-test. What can you say about the treatment effect?

2.8. When you compare between the t-tests of data1 vs t-test of data2 which result will you trust more and why?

Question 3:

In this question you will work with data provided in the CLIP.BED file. This file contains protein binding data stored in a BED format. Each range contains a genomic location to which the protein binding sites were mapped.

3.1. Generate a GRanges object which will contain a data from this file. For this matter read the file by the CLIP_data<-read.table(file.path("path as you set it according the file.path function requirements"),header=TRUE)
3.2. Generate a GRanges object that will contain the flanking regions of the end of each range. The width of the flanking region should be 200. The flanking region should not overlap the end coordinate.