COMP 364 - Tools for the Life Sciences

Assignment #1: Learning R
Due: February 25th, 2016
10% of total grade

The goal of the assignment is to help you explore the basics of R, and to ease the transition into dealing with large -omic datasets and statistical methods. At the end, the assignment focuses on performing small tasks on the `huc` *data structure that was described in class. It is available through the code in the GIT repository.*

Question 1: [20% of assignment] Practicing the basics

Create a new file called `Assignment1.R` in RStudio.

Put a comment (line starting with #) with your name and student number.

Add a line the your `Assignment.1.R` file stating *Question 1.*

(i) Suppose x and y are variables that you have assigned values to. Show how to swap the values in x and y so that afterwards, y has the value x had, and x has the value y had.

(ii) Consider the following code:

```
M <- sample( 1:10000, size = sample(1:10000, size = 1) )
```

Describe in a sentence or two what M is. Is it always the same?

(iii) Using M as in (ii), show R code that finds the maximum value in M. (Don’t use the built-in R function for finding the maximum … this is just for your practice using loops.)

(iv) Write a function as follows:

```
findMax <- function( myData ) {
        blah blee blah blah

        }
```

and fill in the “blah blee blah blah” in order to return the maximum value in the vector of data myData. (This question asks you to simply turn your solution from (iii) into a function.)

Question 2: [20% of assignment] More basics

This question builds on our discussion in class on Feb 8.
Add a line the your `Assignment.1.R` file stating *Question 2.*

(i) Define a matrix M of size n by m (n and m can be any integers and they don’t necessarily have to be equal!).
(ii) Now any diagonal elements of the form M[i,i] (as long as 1 <= i <= n and 1 <= i <= m) have an 0.
(iii) Any elements of M[i,j] where i < j have value -1.
(iv) Any elements of M[i,j[where i > j have value +1.

Now make this into a function called “initializeM” that takes as an argument a matrix M of dimensions n, m, and returns a modified version of M according to conditions (ii-iv).

So…

```
initializeM <- function( M ) {
        blah blah blah
        return(???)
}
```

Question 3: [20% of assignment] Ya, more basics

This builds on our discussion in class on Feb 10.
Add a line the your `Assignment.1.R` file stating *Question 3.*

Define a list `myList` with the following three elements:
(i) A list of all constants in lower case;
(ii) A list of odd numbers between 0:100;
(iii) A list that contains the birthdays of you and your siblings (so it might have only one element if you are an only child), where the birthday is of the form c(DD, MM, YY) where DD is the day, MM is the month, and YY is the year in integer format).

(iv) Show R code to select only the 1st and 3rd element of `myList`.

Question 4: [20% of assignment] The HuC.

Add a line the your `Assignment.1.R` file stating *Question 4.*

In this file, write the R scripts necessary to accomplish the following.

Set the working directory in your R session to `~/repo/comp364`.

Then load the hucMini source file (`src/hucMini.R`). This file contains the `huc.load()` function.

Using this function, load the following datasets

```
dataset.collection <- c( “miniTCGA”, “nki”, “vanvliet” )
```

and assign them to a variable called `huc`.

Now print out the names of all the objects under huc.

For each of these objects, now print out the names of all of its sub-objects (so in other words, all the objects below each dataset).

Try to use a `for`-loop or other type of looping structure to do this.

Hint: If you do something like

```
x <- dataset.collection[1]
names(huc$x)
```

you will get an error.

However if you instead do

```
names(huc[[x]])
```

it will work.

Question 5: [20% of assignment] The `clinical` object

Add a line the your `Assignment.1.R` file stating *Question 5.*

This question tries to help familiarize you with the `clinical` object associated with each gene expression dataset in the HuC. In the following questions, if the an entry in the `clinical` data.frame is `NA`, this is to be ignored. For example, in the 'er' frame, the patient is neither considered ER+ nor ER-.

(Note: where appropriate, use the `na.rm = TRUE` flag in function calls. This removes (rm) the NA entries from the calculation.)

Write R code to determine the following:

a. How many patients are there in each of the five datasets?
b. What fraction of patients are ER+ in each of the five datasets?
c. What is the ratio of observed good to poor outcome in each of the 5 datasets? (This is the `event.5` variable where `TRUE` means there was an distant metastasis and therefore poor outcome.)

d. What fraction of **all** patients (all datasets combined) are ER+?
e. What fraction of **all** patients (all datasets combined) are HER2+?
f. ER+ and HER2+?
g. ER-, HER2-, lymph node positive, and under 50 years of age?
h. ER-, HER2-, lymph positive, < 50 years, no event at 5 years (event.5)?

Submit your code and a table to organize all the actual numbers you obtained for parts (a-h).