**Functions and Loops with R**

The files you need to submit are your **R script and this document in PDF**. Paste in your code and output into this document where it is specifically requested. If directions are given but nothing is asked directly, an “OK” response will suffice. Please name your files following this convention:

epid799b\_hw2\_lastname.pdf (Homework document with answers filled in)

epid799b\_hw2\_lastname.r (R script).

For this assignment, 10 points will be taken away for failing to follow all instructions.

**Readability.** Both your graders, and in “real” life, your colleagues and future-you will appreciate greatly your well-written, structured and documented code. Please continue to explore readability habits as you code.

You may need certain packages that are not pre-installed. If needed, install the packages first, and load the packages by using the command **library(*functionname*)**.

Complete the follow steps in R/RStudio:

1. **Read & subset the file** (15 points)This assignment will continue to use the dataset births.csv from Sakai.
   1. Read in births.csv file into a data.frame called “births,” as we have done in class and homework 1. Do not read strings in as factors. (5 points)
   2. Again, as before, let’s subset – in this case to these variables: "MAGE", "MDIF", "VISITS", "WKSGEST", "MRACE", “SEX”, and “CIGDUR”. Keep all the records. (5 points)
   3. Just to avoid typing caps all the time, change data.frame elements to lowercase. Hint: You can do this many (inefficient) ways, such as one by assigning each variable to a new, lower-case variable, then using subsetting to drop the capital ones. Instead, you can directly assign to the names(births). Try using the names() and tolower() functions to do this all in one line. (5 points)
2. **Functions A: My dog Lima** (10 points)  
   We have been using many functions already, like read.csv(), summary() and str(). Now let’s explore creating our own.
   1. Copy this (silly) function into your code and run the function block. What does running the function block do? (5 points)

my\_dog = function (bean){

cat("My dog", bean, "likes to roam.\n")

cat("One day", bean, "left his home!\n")

cat("He came back all nice and clean.\n")

cat("Where oh where had", bean, "been?\n")

cat(bean, "been!\n", bean, "been!\n")

cat("Where oh where has", bean, "bean?\n\n")

}

* 1. Now run the function with a dog name of your choice. If you’re not feeling creative, you might consider the below list: Lima, Coffee, Pinto, Espresso, Bean, Bean-Bean. How did you call that function with your dog name? (5 points)

1. **Loops & vectors: Numeric** (30 points)
   1. Write a for-loop that iterates through the length of the vector a=1:10, adds 1 to the element at that position, and prints it out. (10 points)
   2. Write a for-loop that iterates through the letters of the alphabet *by numeric position* and prints out “The letter \_.”, where \_ is replaced by the letter of the alphabet. Use the length() function and the R character array constant “letters.” (10 points)
   3. Write the same for-loop as before, but loop through the letters themselves without using their numerical index. Hint: you won’t need 1:X or the length() function. (10 points)
2. **Loops & vectors: cigdur** (35 points)Let’s investigate different ways to work with the cigdur variable.
   1. Use a function of your choice to explore the current contents of the (now lowercase) cigdur variable. Some options include (but are not limited to): (1) the table() function with the useNA=”always” parameter, (2) the unique() function, (3)the str() function. (5 points)
   2. **SAS style**: create a new variable smoke1 in the births dataset and assign it to zero. Loop through the length of the births dataset using the position integer i. Depending on whether the ith cigdur variable is Y, N or U, recode smoke1 to 1, 0 or NA respectively. (10 points)
   3. Create an integer variable smoke2 from cigdur using three vector assignment statements (likely the same as you did in HW1). (10 points)
   4. Create an integer variable smoke3 using the ifelse() function and vector assignment. (10 points)  
      *Note: (1) You will need to nest them and (2) use the ifelse() function, not an if(){}else{} block!*
   5. Optional challenge: A quick google search of “recode R” suggests a few helpful packages. We’ll explore dplyr later, but for now: spend a few minutes looking at the recode() function of the car package. Give a few sentences of what car::recode() does.
3. **Write the file** (10 points)Let’s save our work on this birth dataset using the write.csv() function. Open the help file for this function and explore the arguments.
   1. The write.csv() function is an instance of a more generic function. What is that function?
   2. What is the first argument by position?
   3. What does the row.names argument default to and what does setting it do? (you might consider running write.csv() with each version and opening the csv file in excel to see these differences).
   4. Write the births data.frame to births\_v2.csv, setting row.names=F.
4. **Project** (Pass/Fail)  
   Last homework you described the background for your project. This time, let’s work on a more specific, but high-level plan.
   1. Briefly describe the steps to your analysis at a high level. This will include the methods you’re hoping to use, but also steps like reading in your file, recoding your data, creating table 1, graphing, etc. Most projects will have at least 5-10 big picture steps.
   2. Now, in an R file, pseudo-code your analysis plan using comments. Each step might have a comment-block or just a comment line.