

Note: Please submit your R program via Blackboard by the due date. The instruction for submitting the assignment is written on the course syllabus. Please read *AssignmentFormat.pdf* file for correct assignment format.

Grading

The first assignment is only graded based on completion. (The second and the third assignments will be graded on correctness.) I will give you full credit for the first assignment even for incorrect answer; however, you will not receive any credits for any missing problems. You will also be deducted points if you do not follow the correct format that is written on the *AssignmentFormat.pdf*. I will provide my solution after the assignment due date.

Format: 3 points

Problem 1: 9 points

1. Generate the following vectors. Using the `seq` function to create vector **a** and **b**. Using the `rep` to create **d** and **e**.
 - Vector **a**: 2, 3, 4, ..., 10
 - Vector **b**: 15, 12, 9, 6, 3
 - Vector **d**: repeating vector **a** twice
 - Vector **e**: repeat the first element of vector **b** 5 times, the second element of **b** 4 times, the third element 3 times, the fourth element twice, and the last element once
2. Write R commands to answer the following questions (using the vectors that you created in previous problem).
 - How many numbers in vector **d** are equal to 5?
 - Are any elements of vector **e** < 1 ?
 - How many numbers are greater than 9 in both vectors **a** and **b** combined?
 - How many missing values are in vector **f**?
`f = c(1, 4, 5, 9, -1, NA, 2, NA, 3, NA, 9, 3)`
 - Calculate the sum of **f** (Hint: use the `na.rm` option).

Problem 2: 9 points

To generate a random number that follows standard normal distribution, we can use the `rnorm` function. For example, to generate 5 random numbers, simply type `rnorm(5)`.

- Create a 4 by 5 matrix containing 20 randomly generated numbers that follow standard normal distribution. Use two ways to create this matrix. One matrix will be called **x1**, the other one **x2**.
- Create a matrix, **smallx**, by taking the last three rows and first and last columns of **x1**.
- Write R commands to answer the following questions:
 - What is the dimension of **smallx**?
 - How would one change **smallx** to a vector?

Problem 3: 9 points

Consider the following data set:

Name	Sex	Age	Height	Weight	Smoke
Alfred	M	23	72	160.3	TRUE
Barbara	F	35	61	125.4	NA
John	M	25	NA	175.0	FALSE
Kerry	F	19	66	130.2	FALSE

- Create 6 vectors, `name`, `sex`, `age`, `height`, `weight`, and `smoke`, one for each of the variables above.
- Add the `names` attribute for the `age` vector by using the `name` vector.
- Write an R command to find out whose weight is over 150 pounds?
- Create a list, `example.list`, based on these 6 vectors. Use the names of the vector as the names of component of the list.
- Create a vector, `bmi`, based on vectors `weight` and `height`, according to the following formula: $bmi = 100weight/height^2$. Then concatenate `bmi` to `example.list`. Make sure `bmi` is a list before you concatenate it.
- Create a list, named `small.list`, based on `example.list` that only contains the `name` and `sex` components.
- Convert `example.list` to a data frame, named `example.data`.
- Create a data frame, `female`, based on the data frame `example.data` by only keeping the female subjects. When you create this data frame, only keep variables `name`, `sex` and `age`.
- Change the variable names of the `female` data set from `name`, `sex` and `age` to `f.name`, `f.sex`, and `f.age`.
- Change the default row names of `female` to A01, A02.