

## Assignment 6

Create the following functions. You don't have to save the functions to a separate file. After creating the functions, test them with at least **two** sample inputs to the functions.

1. Create a function that takes in a numeric vector. The output should be a vector with running mean values. The  $i^{\text{th}}$  element of the output vector should be the mean of the values in the input vector from 1 to  $i$ . (worth 20 points)

2. Create a function that takes in a numeric vector. The objective is to forecast using exponential smoothing. The formula is ( $Y_t$  is the actual value and  $\hat{Y}_t$  is the forecasted value) -

$$\hat{Y}_{t+1} = \hat{Y}_t + \alpha(Y_t - \hat{Y}_t)$$

The output of the function should be a dataframe with two columns – actual and predicted values. Set a default value of 0.8 for  $\alpha$ . (worth 30 points)

*Note: for the first row, the predicted and the actual values are same.*

3. There is a package with a function that enables you to check if a number is prime.

```
install.packages("schoolmath")
```

```
library(schoolmath)
```

```
is.prim(3)
```

Create a function that takes in two integers (set default values of 1 to both). The function should calculate the number of prime numbers between the two values. (worth 20 points)

4. Simulate a function to roll a dice. Note that a dice turns up with numbers 1, 2, 3, 4, 5 or 6. The function should do the following: you roll the dice twice, and if both the numbers are the same then return 'You Win' otherwise return 'You Lose' (worth 20 points)

Hint: use the function `sample(vector, size, replace=T/F)`. The argument `vector` indicates the vector of values to draw from. The argument `size` indicates how many values to draw, and `replace` argument indicates whether you sample with replacement. Also note that this function does not take in any parameters, it returns a string, i.e. if the result of two rolling are 3 and 6, the function should return `"You Lose"`

5. Create a function called `Missing` that takes in a data frame as the input and outputs another data frame with column names, number of missing values in each column, percentage of missing values in each column, and the number of unique values in each column. An example output is:

```
> Missing(survey)
  Column Name # Missing Values % Missing Values # Unique Values
1      Sex      1      0.004219409           3
2    wr.Hnd      1      0.004219409          61
3    NW.Hnd      1      0.004219409          69
4      W.Hnd      1      0.004219409           3
5      Fold      0      0.000000000           3
6      Pulse     45      0.189873418          44
7      Clap      1      0.004219409           4
8      Exer      0      0.000000000           3
9      Smoke      1      0.004219409           5
10     Height     28      0.118143460          68
11       M.I     28      0.118143460           3
12      Age      0      0.000000000          88
```

Hint: If `x` is the name of a data frame, then `ncol(x)` gives you the number of columns in the data frame `x` and `sum(is.na(x[,i]))` gives you the number of missing values in the  $i^{\text{th}}$  column of `x`. (worth 50 points)

Save all the code above in a script file called *A6yourname*. Make sure to include ample comments to describe your code. Upload your script to submit the assignment.

**(A significant portion of your grade will depend on adequate commenting of the code. Make sure to include your name in the code as a comment).**

**(Please read: By submitting the assignment, you are confirming that this represents your own work, and that no part of the work submitted has been copied from others. Consequences for cheating can be quite serious. It can lead to not only failing the course but also dismissal from the program)**