

# Homework 4

For this homework you will create an R Markdown file and output (PDF) and upload both to wolflare. Be sure to include text explaining your thought process/what you are doing with your questions.

The purpose of this homework is to get practice writing functions. For part of the assignment, we'll also use the titanic data set from <https://www4.stat.ncsu.edu/~post/ST558/datasets/>. Code is given to pull this data set (just make sure you have a connection to the internet when you go to knit or run the code).

1. Create a function called `count_my_change` that adds up the total amount of money you input.

- Include 5 arguments - dollars, quarters, dimes, nickels, and pennies
- Set the default value of ALL the arguments to 0
- Calculate a total value by multiplying each argument by its dollar value
- Create and return a string that says "Your total is \$amount" with the total

Test the function for on a few values to check it works correctly.

2. Explain what the difference is between an infix and prefix function.
3. Write your own infix function that can be used in replace of the `cbind` (or column bind) function (feel free to call the `cbind` function in your function :) Check that the function works on the two data frames below (`df1` & `df2`) and then all three objects (`df1`, `df2`, `vec1`)!

```
df1 <- data.frame(unif = runif(10), norm = rnorm(10))
df2 <- data.frame(gamma = rgamma(10, shape = 1), beta = rbeta(10, shape1 = 5, shape2 = 1))
vec1 <- rnorm(10, mean = 10)
```

4. Write a basic function (call it `getIQR`) that takes in a vector of numbers and returns the interquartile range (75th percentile - 25th percentile). You can use the `quantile` function. Remove the name on the returned vector (i.e. make sure the returned item doesn't have a name like 75% or something like that). Allow your function to pass unnamed arguments to the `quantile` function.

Have your function check if the passed value is numeric and if it is appropriate for this argument (i.e. is a vector or a data frame/tibble with only one column - don't allow for matrices) - if not, stop the function and return an appropriate message.

Note: trying to deal with data frames/tibbles of 1 dimension was a bit tricky for me. I first checked if it was a data frame, if so, I checked if it had only one column. If those were both true I converted that column to a vector (using `[[` or `pull`). I then checked to see if the value was not numeric or not a vector.

5. Read in the titanic data set and use the above function twice - once to the age column and once to the fare column. Check that it works with `select(titanicData, age)` as well as `titanicData$age`. Note: you need to be able to handle the NA values.

```
titanicData <- readr::read_csv("https://www4.stat.ncsu.edu/~post/ST558/datasets/titanic.csv")
```

6. Write another function (call it `getCoefVar`) that returns the sample coefficient of variation, defined as the sample standard deviation divided by the absolute value of the sample mean. Allow the function to pass unnamed arguments to `sd` and `mean`.

As before, check that the given argument is valid for use with this function - if not return a message.

7. Use the above function twice on the same columns of the titanic data set. Note: you need to be able to handle the `NA`. Again, check if this works on a one dimensional tibble (as returned by `dplyr::select`)
8. Create a function (call it `getSummaries`) that takes in a vector and returns both the IQR and the coefficient of variation. Do not rewrite your above two functions, call them. When returning your values, give them appropriate names.
9. Apply your new function to the same two columns of data.
10. Why can you name a variable `sd` in a function and not cause any issues with the `sd` function?