Class_6: R Functions Lab

M. Ramos

2022-10-14

Table of contents

##Function basics

All functions in R have at least 3 things:

- A **name** (we pick this),
- Input **arguments** (there can be loads comma separated),
- A **body** (the R code that does the work).

Take input of example grade schools

```
# Example input vectors to start with
student1 <- c(100, 100, 100, 100, 100, 100, 100, 90)
student2 <- c(100, NA, 90, 90, 90, 90, 97, 80)
student3 <- c(90, NA, NA, NA, NA, NA, NA, NA)
```

Q1. Write a function grade() to determine an overall grade from a vector of student homework assignment scores dropping the lowest single score.

If a student misses a homework (i.e. has an NA value) this can be used as a score to be potentially dropped. Your final function should be adquately explained with code comments and be able to work on an example class gradebook such as this one in CSV format: "https://tinyurl.com/gradeinput"

Beginning to craft my function

I can use the mean() function to get the average

```
mean(student1)
```

```
[1] 98.75
```

Sort the vector in decreasing order

```
sorted_vector <- sort(student3 , decreasing = T, na.last = T)
sorted_vector
```

```
[1] 90 NA NA NA NA NA NA NA
```

Remove the furthest on the right to drop the lowest score since it is in decreasing order

```
new_vector_drop <- head(sorted_vector, n = -1)
new_vector_drop
```

```
[1] 90 NA NA NA NA NA NA
```

change NA's to zero if still in list like with student 3

```
for_scoring_NAs <- replace(new_vector_drop, is.na(new_vector_drop), 0)
for_scoring_NAs
```

```
[1] 90  0  0  0  0  0  0
```

Now calculate the average test score of the student

```r
grade <- mean(for_scoring_NAs)
grade
```

```
[1] 12.85714
```

Next, make it all a single function where x is the students input score vector

```r
grading_is_fun <- function(x){
   sorted_vector <- sort(x , decreasing = T, na.last = T)
   new_vector_drop <- head(sorted_vector, n = -1)
   for_scoring_NAs <- replace(new_vector_drop, is.na(new_vector_drop), 0)
   grade <- mean(for_scoring_NAs)
   grade
}
```

Try it on an example csv file

```r
test_scores <- read.csv(file = 'class6_files/student_homework.csv')
```

Try to modify the function to work on a df gradebook I am applying the function to a subset of the test_scores dataframe so that it only operates on the scores in the dataframe and not the student id.

```r
gradebook_report <- apply(test_scores[,c(2:6)],1, grading_is_fun)
gradebook_report
```

```
 [1] 91.75 82.50 84.25 84.25 88.25 89.00 94.00 93.75 87.75 79.00 86.00 91.75
[13] 92.25 87.75 78.75 89.50 88.00 94.50 82.75 82.75
```

Each position index is the student number. For example, student1 average score is position 1 with 91.75 in the results. Same with student20 having average score of 82.75.

Q2. Using your grade() function and the supplied gradebook, Who is the top scoring student overall in the gradebook?

Find the student with max score among the student grade averages.

```r
which.max(gradebook_report)
```

```
[1] 18
```

Student 18 has the max score among student grade averages.

Q3. From your analysis of the gradebook, which homework was toughest on students (i.e. obtained the lowest scores overall?

Find out which homework was the most difficult for students.

I need to write a new function to do this on the data frame since that df contains hw number annotation.

First, change NAs to zero so they aren't contagious

```
gradebook_No_NAs <- replace(test_scores, is.na(test_scores), 0)
```

Then, calculate the mean of a homework

```
mean(gradebook_No_NAs$hw2)
```

```
[1] 72.8
```

Make the mean calculation into a function

```
hard_homework <- function(hw){
  mean(hw)
}
```

Now iterate your function over the gradebook by homework columns

```
apply(gradebook_No_NAs[c(1:20),c(2:6)],2,hard_homework )
```

```
  hw1   hw2   hw3   hw4   hw5 
89.00 72.80 80.80 85.15 79.25 
```

So, homework 2 was the most challenging.

```
hw_averages <- apply(gradebook_No_NAs[c(1:20),c(2:6)],2,hard_homework )
```

Alternatively, we could have R output the hardest homework by embedding the answer in a text section: The most difficult homework was homework 2

Q4. Optional Extension: From your analysis of the gradebook, which homework was most predictive of overall score (i.e. highest correlation with average grade score)?

Use the correlation function in r to correlate the gradebook values we have for each homework with the gradebook report we created which we generated by calculating the average scores of each student.

```r
cor(gradebook_No_NAs$hw1, gradebook_report)
```

```
[1] 0.4250204
```

```r
cor(gradebook_No_NAs$hw2, gradebook_report)
```

```
[1] 0.176778
```

```r
cor(gradebook_No_NAs$hw3, gradebook_report)
```

```
[1] 0.3042561
```

```r
cor(gradebook_No_NAs$hw4, gradebook_report)
```

```
[1] 0.3810884
```

```r
cor(gradebook_No_NAs$hw5, gradebook_report)
```

```
[1] 0.6325982
```

Homework 5 is the highest correlate with average grade score.

```r
cor(gradebook_No_NAs$hw5, gradebook_report)
```

```
[1] 0.6325982
```