Class06

AUTHOR
Joseph Kesler

PUBLISHED
October 16, 2022

This week we are introducing R functions and how to write our own.

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" [3pts]

```
#example 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)
```

Making a working snippet of code:

```
mean(student1)
```

```
[1] 98.75
```

But we need to drop the lowest score. First we need to identify the lowest score.

```
min(student1)
```

```
[1] 90
```

Is there another way?

```
#which element of the vector is the lowest?
which.min(student1)
```

```
[1] 8
```

What I want is to now drop this lowest score from my mean calculation.

```
#This will return everything but the 8th element of the vector
student1[-8]
```

```
[1] 100 100 100 100 100 100 100
```

Now we can use which.min() to return all the lowest elements.

```
#This is our first working snippet
mean(student1[-which.min(student1)])
```

```
[1] 100
```

We are now able to easily score 1 student. Will it work with the other students? In short, no because we have to do something with the NA's

```
x <- student2
is.na(x)
```

```
[1] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
which(is.na(x))
```

```
[1] 2
```

Now that we have identified the NA's, we want to mask 'em. Replace them with 0?

```
x[is.na(x)] <- 0
x
```

```
[1] 100   0  90  90  90  90  97  80
```

```
mean(x)
```

```
[1] 79.625
```

Remember, we should drop the lowest score though 2. So lets combing the above bit with our fist snippet

```
mean(x[-which.min(x)])
```

```
[1] 91
```

Lets try it for student 3 now.

```
x <- student3
x[is.na(x)] <- 0
mean(x[-which.min(x)])
```

```
[1] 12.85714
```

##now we make our heading

Take the snippet and turn it into the function. Gotta give it a name, input arguments, and a body.

I will select code in the Rstudio dashboard then extract funtion in the drop down

```
grade <- function(x) {
  x[is.na(x)] <- 0
  mean(x[-which.min(x)])
}
```

LETS RUN THE FUNCTION!

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

```
#This function calculates the average score for a vector of student scores dropping the lowest score. Missing values will be treate

#' @param x A numeric vector of HW scores
#'
#' @return Average score
#' @export
#'
#' @examples
#' student <- c(100,NA,90,97)
#' grade(student)

grade <- function(x) {
  x[is.na(x)] <- 0
  mean(x[-which.min(x)])
}
```

Now we can finally use the function on 'real' class data

```
url <-  "https://tinyurl.com/gradeinput"
gradebook <- read.csv(url, row.names = 1)
```

```
apply(gradebook, 1, grade)
```

```
 student-1  student-2  student-3  student-4  student-5  student-6  student-7
     91.75      82.50      84.25      84.25      88.25      89.00      94.00
 student-8  student-9 student-10 student-11 student-12 student-13 student-14
     93.75      87.75      79.00      86.00      91.75      92.25      87.75
student-15 student-16 student-17 student-18 student-19 student-20
     78.75      89.50      88.00      94.50      82.75      82.75
```

> Q2. Using your grade() function and the supplied gradebook, Who is the top scoring student overall in the gradebook? [3pts]

```
results <- apply(gradebook, 1, grade)
sort(results,decreasing = TRUE)
```

```
student-18  student-7  student-8 student-13  student-1 student-12 student-16
     94.50      94.00      93.75      92.25      91.75      91.75      89.50
 student-6  student-5 student-17  student-9 student-14 student-11  student-3
     89.00      88.25      88.00      87.75      87.75      86.00      84.25
 student-4 student-19 student-20  student-2 student-10 student-15
     84.25      82.75      82.75      82.50      79.00      78.75
```

```
which.max(results)
```

```
student-18
        18
```

Student 18 is the top scorer

> Q3. From your analysis of the gradebook, which homework was toughest on students (i.e. obtained the lowest scores overall? [2pts]

```
apply(gradebook, 2, mean, na.rm = TRUE)
```

```
     hw1      hw2      hw3      hw4      hw5
89.00000 80.88889 80.80000 89.63158 83.42105
```

##HW 3 is the worst scoring HW