

Class 6: R functions lab (Grades function)

Devanshi

2022-10-14

Table of contents

Function basics	1
Question 1: Writing the Grade Function	1
Question 2: Finding the top scoring student in class	3
Question 3: To find which homework was the toughest on students	3
Question 4: To find which homework predicted the student's overall score	4

Function basics

All functions in R have three basic parts:

- A **name** which we can assign
- **Input arguments** within the function
- Regular R code for the **body** of the function

Question 1: Writing the Grade Function

Using this as an example data set to first write the function.

```
# 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)
```

Writing the function first before importing the actual data set.

We can use `'is.na()'` to convert all NA values to zeroes. We can use `'which.min()'` to find the position of the lowest score and drop that from our data set. And finally, we can use the `'mean()'` function to find the mean of this data set.

Writing this in a function:

```
grade1 <- function(x) {  
  #assigning NAs to be 0s  
  x[ is.na(x)] <- 0  
  #dropping the lowest score in this array  
  x <- x[-which.min(x)]  
  #finding the mean score from this data set  
  mean(x)  
}
```

The above function should be able to sort through the data set, assign NA scores to 0s and drop one of them or the minimum score, and then calculate the mean.

The overall grade of student 1 is: 100

The overall grade of student 2 is: 91

The overall grade of student 3 is: 12.8571429

Now, we can import our `'csv'` file, and run the `'grade1()'` function on the data set in this file.

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

	hw1	hw2	hw3	hw4	hw5
student-1	100	73	100	88	79
student-2	85	64	78	89	78
student-3	83	69	77	100	77
student-4	88	NA	73	100	76
student-5	88	100	75	86	79
student-6	89	78	100	89	77

Use the `'apply()'` function to take another function e.g. the previous `'grade1()'` function, and apply it to an entire data set.

An `'apply()'` function consists of:

1. the matrix name, in this case “gradebook”
2. whether we want to sort it by rows or columns, 1 for rows, 2 for columns
3. the function, either signs or the name in quotes

```
#saving the apply matrix as a variable 'results'
results <- apply(gradebook, 1, "grade1")
results
```

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	

The grades of the students in this data set are: 91.75, 82.5, 84.25, 84.25, 88.25, 89, 94, 93.75, 87.75, 79, 86, 91.75, 92.25, 87.75, 78.75, 89.5, 88, 94.5, 82.75, 82.75

Question 2: Finding the top scoring student in class

Now that we have our ‘results’ saved as a matrix, we can use the ‘which.max()’ function to find the highest scoring student in the class.

```
mres <- which.max(results)
```

The highest scoring student in this class is: Student 18

Question 3: To find which homework was the toughest on students

We can use the ‘apply()’ function in this case to take the average of each column, and choose the lowest average.

```
#writing another apply function to choose the columns
#and remove NAs
hwresults <- apply(gradebook, 2, sum, na.rm=TRUE)
```

The lowest overall homework was: Homework 2

Question 4: To find which homework predicted the student's overall score

This is helpful to know to get an idea of how to model tests and write future homework.

We can use the 'cor()' function to correlate the student's weekly homework scores with their overall scores.

```
#to convert NA scores in 'gradebook' to 0s
mask <- gradebook
mask[is.na(mask)] <- 0

#use apply function to compare results with each
#column in mask
correlation <- apply(mask, 2, cor, y=results)

#find the highest correlation of all the homework
hcor <- which.max(correlation)
```

The homework that best predicted the students' overall score was: Homework 5