Class 06 Bimm 143

Nickolas Beam

Table of contents

Function basics

All functions in R consist of at least 3 things:

- A **name** (we can pick this but it must start with a character)
- Input **arguments** (there can be multiple comma separated inputs)
- The **body** (where woork actually happens)

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)
```

```
mean(student1)
```

```
[1] 98.75
```

```
min(student1)
```

```
[1] 90
```

Looking at the "See Also" section of the `min()` help page I found out about `which.min()`

```
which.min(student1)
```

```
[1] 8
```

```
student1[1:7]
```

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

I can get the same vector without the 8th element

```
index1 <- which.min(student1)
index2 <- which.min(student2)
index3 <- which.min(student3)
student1[-index1]
```

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

```
student2[-index2]
```

```
[1] 100  NA  90  90  90  90  97
```

```
student3[-index3]
```

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

```
mean1 <- mean(student1)
mean2 <- mean(student2)
mean3 <- mean(student3)
```

```
  top_score <- max(mean1, mean2, mean3)
```

line 44 and 47 and 50 can be combined as such

```
  mean( student1[-which.min(student1)])
```

```
[1] 100
```

```
  mean( student2[-which.min(student2)], na.rm = TRUE)
```

```
[1] 92.83333
```

na.rm = TRUE allows to ignore the NA's is.(na) -> 0 replaces NA's with 0

```
  mean(student1[-which.min(student1)])
```

```
[1] 100
```

```
  student2[is.na(student2)] <- 0
  mean(student2[-which.min(student2)])
```

```
[1] 91
```

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

```
[1] 12.85714
```

^ averages of students scores without the lowest score and with NA -> 0

I now have a working snippet of code that will work for each student

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

```
[1] 12.85714
```

turning this into a function grade()

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

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

Question 2

Q2. Using your grade() function and the supplied gradebook, Who is the top scoring student overall in the gradebook? [3pts] Student 1 is the top scoring student overall

```
url <- "https://tinyurl.com/gradeinput"
gradebook <- read.csv(url, 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
```

Learn apply() function

```
results <- apply(gradebook, 1, grade)
```

Which student did the best overall??

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

```
student-18
        18
```

```
results[which.max(results)]
```

```
student-18
      94.5
```

Student 18 is the top scoring student

```
which.min(apply(gradebook, 2, sum, na.rm = TRUE))
```

```
hw2
  2
```

Question 3

```
low_assignment <- apply(gradebook, 2, grade)
low_assignment[which.min(low_assignment)]
```

```
     hw2
76.63158
```

Question 4

> 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)? [1pt]

```r
mask <- gradebook
mask[is.na(mask)] <- 0

cor(mask, results)
```

```
        [,1]
hw1 0.4250204
hw2 0.1767780
hw3 0.3042561
hw4 0.3810884
hw5 0.6325982
```

Question 5

```r
apply(mask, 2, cor, y=results)
```

```
      hw1       hw2       hw3       hw4       hw5
0.4250204 0.1767780 0.3042561 0.3810884 0.6325982
```