Class 06: R Functions

Nathaniel Lightle (A16669288)

All about functions in R

Functions are the way we get stuff done in R. We call a function to read data, compute stuff, plot stuff, etc.

R makes writing functions accessible, but we should always start by trying to get a working snippet of code first before we write our function.

Todays lab

We will grade a whole class of student assignments

```
# 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)
```

Finding average grade for student 1 If we want to find the average we can use the `mean()` function

```
mean(student1)
```

```
[1] 98.75
```

Now we want to drop the lowest score

```
min(student1)
```

```
[1] 90
```

New function just came out: `which.min()` function. We gonna try it

```
  student1
```

```
[1] 100 100 100 100 100 100 100  90
```

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

```
[1] 8
```

```
  student1[8]
```

```
[1] 90
```

Using the ol' minus syntax trick

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

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

Putting it all together...

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

```
[1] 100
```

Let's try student 2

```
  student2
```

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

```
  mean(student2[-which.min(student2)])
```

```
[1] NA
```

Where is the problem?

```
mean(student2)
```

```
[1] NA
```

Ah it's the `mean()` function. I'm going to fix it using the `na.rm()` function

```
mean(student2, na.rm=TRUE)
```

```
[1] 91
```

```
student3
```

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

```
mean(student3, na.rm=TRUE)
```

```
[1] 90
```

Not good. We need to fix this!

We're going to start writing students into variable `x`

```
x <- student3
x
```

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

We want to overwrite the NA values w/ zero - if you miss a homework you score zero on this homework

```
x
```

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

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

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

```
x[is.na(x)]
```

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

Let's try setting NA = 0

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

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

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

```
[1] 12.85714
```

Here is my final working snippet of code that works for every student!

```
# We can set x to any student
x <- student1
# Mask NA values to zero
x[is.na(x)] <- 0
# Drop lowest score and get the mean
mean(x[-which.min(x)])
```

```
[1] 100
```

Let's make that a function

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

```
[1] 12.85714
```

The function works!

```
  grade(student1)
```

```
[1] 100
```

```
  grade(student2)
```

```
[1] 91
```

```
  grade(student3)
```

```
[1] 12.85714
```

Let's see if it works with a sample class gradebook

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

  apply(gradebook, 1, grade, simplify=TRUE)
```

```
 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
```

Let's condense this into ans

```
  ans <- apply(gradebook,1,grade,simplify=TRUE)
  ans
```

```
 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
```

This is the answer to Q1

QUESTION 2

Let's try to find the top scoring student overall for Q2

```
which.max(apply(gradebook, 1, grade, simplify = TRUE))
```

```
student-18
        18
```

```
max(apply(gradebook, 1, grade, simplify = TRUE))
```

```
[1] 94.5
```

Looks like its Student 18 with a score of 94.5!

QUESTION 3

Now let's see which homework was toughest on students

```
apply(gradebook, 2, grade, simplify=TRUE)
```

```
     hw1      hw2      hw3      hw4      hw5
89.36842 76.63158 81.21053 89.63158 83.42105
```

Looks like it was homework 2