Class 06: R Functions

Clarissa Savko (PID: A69028482)

All about functions in R

Every function in R has at least 3 things: - name (you pick it) - arguments (the input(s) to your function) and - the body

Today we will write a function to grade a class of student assignment scores (homework) First we will work with a simplified vector input where we know what the answer should be

Example input vectors to start with

```r
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)
```

```r
mean(student1)
```

```
[1] 98.75
```

How can we drop the lowest score?

```r
min(student1)
```

```
[1] 90
```

Let's try the function `which.min`

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

```
[1] 8
```

Alternatively… this will give the score in the position that `which.min()` returned.

```
student1[8]
```

```
[1] 90
```

This will return everything besides the score in the position of the minimum.

```
student1[-8]
```

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

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

```
[1] 100
```

Will this work for student 2?

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

```
[1] NA
```

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

```
[1] 100
```

```
student3
```

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

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

```
[1] 90
```

We can mask the NA or change them to 0. We can use the `is.na()` function to find where the missing homeworks are in the input vector.

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

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

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

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

```r
#Mask NA to zero
x <- student3
x[is.na(x) ] <- 0
#Find the mean dropping the lowest score.
mean(x[-which.min(x)])
```

```
[1] 12.85714
```

Turn this into a function.

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

```r
grade(student2)
```

```
[1] 91
```

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]

I need to read the gradebook csv file

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

A very useful function we are being forced to use is the `apply()` function. How can we use it for our `grade()` function and apply it over the full grade book?

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

```
which.max(ans)
```

```
student-18 
        18 
```

Student 18 is the top scoring student.

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

```
hw3 
  3 
```

Let's mask the NA values to 0.

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

```
           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   0  73 100  76
student-5   88 100  75  86  79
student-6   89  78 100  89  77
student-7   89 100  74  87 100
student-8   89 100  76  86 100
student-9   86 100  77  88  77
```

```
student-10  89  72  79   0  76
student-11  82  66  78  84 100
student-12 100  70  75  92 100
student-13  89 100  76 100  80
student-14  85 100  77  89  76
student-15  85  65  76  89   0
student-16  92 100  74  89  77
student-17  88  63 100  86  78
student-18  91   0 100  87 100
student-19  91  68  75  86  79
student-20  91  68  76  88  76
```

```
apply(mask, 2, mean)
```

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

```
which.min(apply(mask, 2, mean))
```

```
hw2 
  2 
```

```
which.min(apply(mask, 2, mean))
```

```
hw2 
  2 
```

HW 2 was the toughest for the students.

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]

```
cor(mask$hw2, ans)
```

```
[1] 0.176778
```

```
cor(mask$hw5, ans)
```

```
[1] 0.6325982
```

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

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

HW5 is most predictive of overall scores because it has the highest correlation value.