Class06: R Functions

Ari_Fon (PID: A15390446)

2/3/2022

Today we will explore R functions.

We will start with calculating a grade for these example students

```
# 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)
```

We could use the mean() function to calculate an average

```
#Attempt was to find the mean of scores of students
mean(student2, na.rm=TRUE)
```

```
## [1] 91
```

How does the is.na() function work? Attempt with student2

```
student2
```

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

```
is.na(student2)
```

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

We can use this result to get at our NA values (i.e. the TRUE positions)

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

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

```
mean(student2)
```

```
## [1] 79.625
```

```
x <- 1:5
x
```

```
## [1] 1 2 3 4 5
```

```
x[x>2] <- 100
x
```

```
## [1]   1   2 100 100 100
```

Attempt on all students- mean() is.na() functions.

```
mean(student1)
```

```
## [1] 98.75
```

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

```
## [1] 79.625
```

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

```
## [1] 11.25
```

Now we want to find the lowest score

```
student1
```

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

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

```
## [1] 8
```

The which.min() function tells us where the smallest value is.

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

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

My attempt on students mean() is.na() which.min ()

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

```
## [1] 91
```

attempt on student 3

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

```
## [1] 12.85714
```

We are close to our working code snippet that will be the body of our first function

```
#first set NA values to zero
x[ is.na(x)] <-0
#remove the lowest score and calculate average
mean(x[ -which.min(x) ])
```

```
## [1] 92.83333
```

Now we can turn this into our first function. We will call this function 'grade()'

All R functions have 3 things -a name(grade) - input arguments(student scores) -body (does the work!)

```
grade <- function(x) {

#first set NA values to zero
  x[ is.na(x)] <-0
  #remove the lowest score and calculate average
  mean(x[ -which.min(x) ])
}
```

Lets try it out

```
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]

```
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
```

We can use the 'apply()' function to grade the whole class

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

```
##  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]

```
which.max(scores)
```

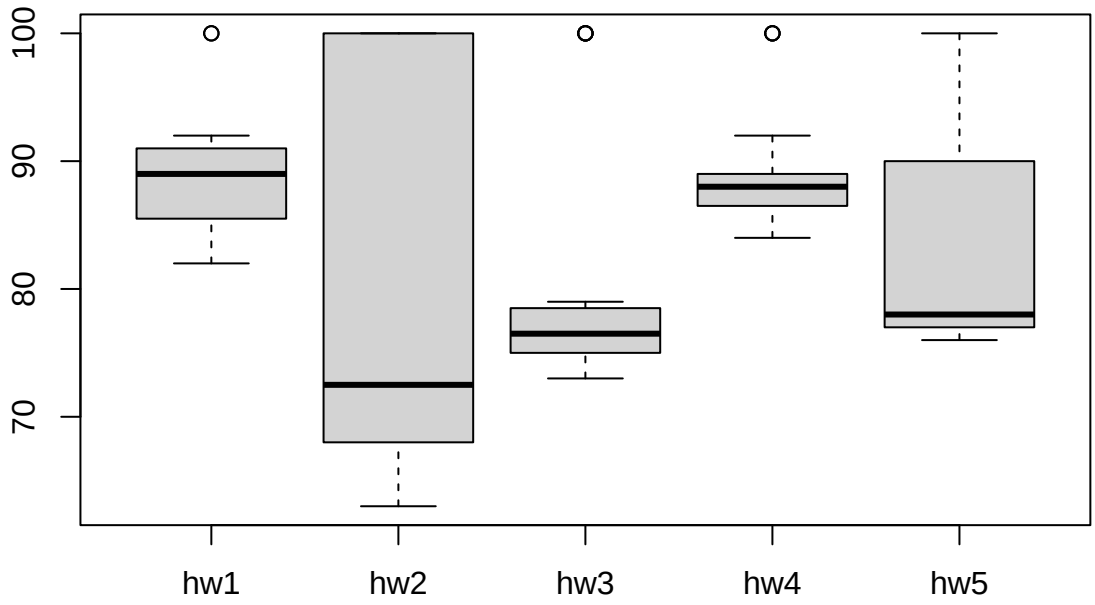
```
## student-18
##         18
```

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

```
avg.hw <- apply(gradebook, 2, median, na.rm=TRUE)
which.min(avg.hw)
```

```
## hw2
##   2
```

```
boxplot(gradebook)
```



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]

Apply over specific HW

```
cor(scores, gradebook$hw1)
```

```
## [1] 0.4250204
```

Apply over all columns (HWs)

```
apply(gradebook, 2, cor, scores)
```

```
##       hw1       hw2       hw3       hw4       hw5
## 0.4250204        NA 0.3042561        NA        NA
```

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

Attempt

```r
apply(mask, 2, cor, scores)
```

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

Looks like HW5 is more predictive of overall score