Week 5: R Functions

Meg Robinson A59010583

2/5/2022

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]

```
# 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)
```

Average of student 1

```
student1
```

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

```
mean(student1)
```

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

Use 'min()' to find lowest score

```
min(student1)
```

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

Find index at which min occured using which.min()

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

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

Get everything except lowest score using minus ("-") and caclulate mean (as long as no "NA" in vector)

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

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

Try it on student 2

```
mean(student2[-which.min(student1)])
```

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

It does not work because mean() function doesn't allow for **NA** values. Now find **NA** values

```
is.na(student2)

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

Replace *NA* with zero

```
student.prime = student2
student.prime[is.na(student.prime)] = 0
student.prime

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

Now get mean()

```
mean(student.prime[-which.min(student.prime)])

## [1] 91
```

Which we can see is the value of *student2*

```
mean(c(100,90,90,90,90,97,80))

## [1] 91
```

So now do the above with *student3*

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

## [1] 12.85714
```

So we can write our function

```
#' Calculate avg scores for a vector hw socres
#' Drop lowest homework score
#' Missing values treated as 0
#'
#' @param x Numeric vector of homework scores
#'
#' @return average score
#' @export
#'
#' @examples
#' student = c(100,NA,90, 80)
#' grade(student)
#'
grade = function(x){
  # Map NA missing hw vals to 0
  # Assign hw scores 0
```

```
  x[is.na(x)] = 0
  # Drop the lowest score
  mean(x[-which.min(x)])
}
```

And use it

```
grade(student1)
```

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

```
grade(student2)
```

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

```
grade(student3)
```

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

Now grade entire class

```
url = "https://tinyurl.com/gradeinput"
gradebook = read.csv(url, row.names=1)
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
## 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  NA  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  NA
## student-16  92 100  74  89  77
## student-17  88  63 100  86  78
## student-18  91  NA 100  87 100
## student-19  91  68  75  86  79
## student-20  91  68  76  88  76
```

Use **apply()** to grade all of the students using our **grade()** function

```
apply(gradebook,1,grade)
```

```
##  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]

```
finalgrades <- apply(gradebook,1,grade)
which.max(finalgrades)

## student-18
##         18

max(finalgrades)

## [1] 94.5
```

student 18 is the top scoring student with an average score of 94.5

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

Average hw's

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

## hw3
##   3
```

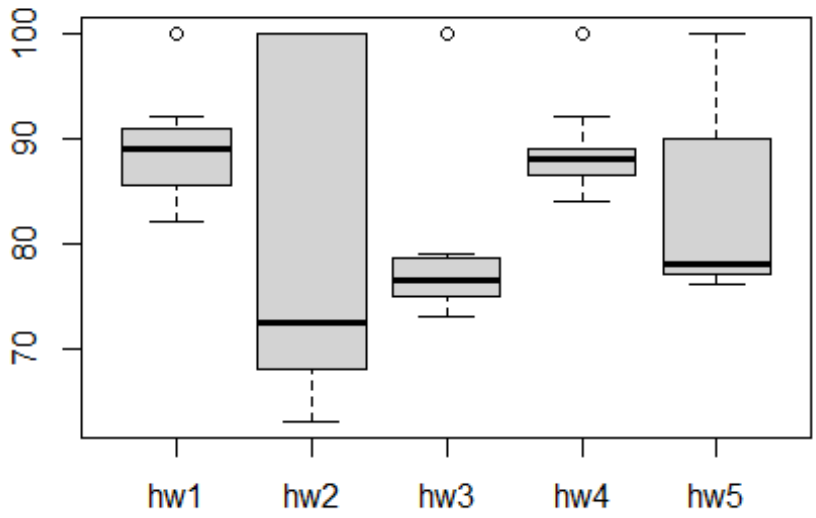
Median of hw's

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

## hw2
##   2
```

Since results were different look at plot of gradebook

```
boxplot(gradebook)
```



It looks like **homework 2** is the toughest.

> 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]

Use the **cor()** function

```
gradebook[is.na(gradebook)] = 0
cor(finalgrades, gradebook$hw1)

## [1] 0.4250204

apply(gradebook, 2, cor, x=finalgrades)

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

Homework 5 was most predictive of a studen't overall score