

class06HW

Ilyas Darif A16577084

2022-04-18

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

A silly function

Lets write a function to add numbers. we can call `add()`

```
x <- 10  
y <- 1  
x + y
```

```
[1] 11
```

```
add <- function(x, y=10){  
  x + y  
}
```

can i just use the function?

```
add(10)
```

```
[1] 20
```

```
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)
```

start with student1

```
mean(student1)
```

```
[1] 98.75
```

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

```
[1] 91
```

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

```
[1] 90
```

ok lets try to work with student1 and find (and drop) the lowest score.

```
min(student1)
```

```
[1] 90
```

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

```
[1] 8
```

```
student1[8]
```

```
[1] 90
```

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

```
[1] 90
```

```
student1[-8]
```

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

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

```
[1] 100
```

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

```
[1] NA
```

our approach to the na problem (missing hws): we can replace to na w 0
first task is find na

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

```
y <- 1:5  
y
```

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

```
y[y>3] <- 0  
y
```

```
[1] 1 2 3 0 0
```

i want to combine the `na.is(x)` with making these elements equal to zero and then take this “masked” (vector of student scores with na values of zero) and drop the lowest and get the mean

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

```
[1] 12.85714
```

```
grade <- function(scores) {
  #make na missing work equal to 0
  scores[is.na(scores)] <- 0
  # drop lowest score and get mean
  mean(scores[-which.min(scores) ])
}
```

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

mean() is.na() min() which.min() apply(.) apply(gradebook, 1 #rows 2 #columns, FUN)

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 adequately 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]

```
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
```

the `apply()` func in r is super useful but can be a little confusing to begin with

```
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
```

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

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

```
student-18
18
```

```
max(ans)
```

```
[1] 94.5
```

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

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

hw3

3

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]

```
#ans  
cor(gradebook$hw1, ans)
```

[1] 0.4250204

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

[1] NA

```
gradebook$hw5
```

```
[1] 79 78 77 76 79 77 100 100 77 76 100 100 80 76 NA 77 78 100 79  
[20] 76
```

```
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

```

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

```
[1] 0.6325982
```

now we can use `apply()` to examine the correlation of

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

```

      hw1      hw2      hw3      hw4      hw5
0.4250204 0.1767780 0.3042561 0.3810884 0.6325982

```

`mean()` `is.na()` `min()` `which.min()` `apply()` `apply(x (dataframe name)gradebook, 1 #rows or 2 #columns, FUN(function, extra arguments for fun))`

Q5. Make sure you save your Quarto document and can click the “Render” (or Rmarkdown”Knit”) button to generate a PDF format report without errors. Finally, submit your PDF to gradescope. [1pt]