

Lab6: R Function Lab

Xiaoxuan Teng (PID: A69028742)

2024-01-28

```
# 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)
```

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]

```
# For student 2
x <- student2
# Find lowest value
ind <- which.min(x)
# Exclude lowest value and find mean
mean(x[-ind], na.rm = T)
```

```
[1] 92.83333
```

```
# For student 3
student3
```

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

```
x <- student3
# Find lowest value
ind <- which.min(x)
# Exclude lowest value and find mean
mean(x[-ind], na.rm = T)
```

[1] NaN

```
# There are 7 NA values in this vector
# There is no value left when exclude lowest and NA values
# So assign 0 to NA value
x <- student3
x[is.na(x)] <- 0
mean(x[-which.min(x)])
```

[1] 12.85714

```
# grade() function: get an overall grade (mean grade) after dropping the lowest score
grade <- function(x) {
  x[is.na(x)] <- 0 # assign 0 to NA value in x
  x <- x[-which.min(x)] # Exclude lowest value
  mean(x) # find mean
}
```

```
# import a csv file named "gradebook"
url <- "https://tinyurl.com/gradeinput"
gradebook <- read.csv(url, row.names = 1)
```

```
# Use the grade() function to grade each students (row) in "gradebook"
results <- apply(gradebook, 1, FUN = grade)
results
```

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]

```
# Determine the index of student number with highest average scores
which.max(results)
```

```
student-18
18
```

So **student 18** is the top scoring student overall in the gradebook.

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

```
# Apply mean() function to each column to get the average score for each homework
# Exclude all NA values
toughest <- apply(gradebook, 2, mean, na.rm = T)
toughest
```

```
      hw1      hw2      hw3      hw4      hw5
89.00000 80.88889 80.80000 89.63158 83.42105
```

```
# Find the homework index with the lowest average score using which.min()
which.min(toughest)
```

```
hw3
3
```

So **hw3** was the toughest on 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]

```
# Generate a data.frame called mask where the NA is replaced by 0
mask <- gradebook
mask[is.na(mask)] <- 0
# Apply cor() function to find the correlation of average grade and grade for each hw
correlation <- apply(mask, 2, cor, y = results)
```

```
correlation
```

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

```
# Find the hw index with the highest correlation using which.max()  
which.max(correlation)
```

```
hw5  
5
```

So **hw5** was the most predictive of overall score.