

class06

Izabelle Querubin

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>”.

```
# Importing the example class gradebook in CSV format
gradebook <- read.csv("https://tinyurl.com/gradeinput", header = TRUE, sep = ",",
  quote = "\"", dec = ".", fill = TRUE, comment.char = "")
gradebook
```

	X	hw1	hw2	hw3	hw4	hw5
1	student-1	100	73	100	88	79
2	student-2	85	64	78	89	78
3	student-3	83	69	77	100	77
4	student-4	88	NA	73	100	76
5	student-5	88	100	75	86	79
6	student-6	89	78	100	89	77
7	student-7	89	100	74	87	100
8	student-8	89	100	76	86	100
9	student-9	86	100	77	88	77
10	student-10	89	72	79	NA	76
11	student-11	82	66	78	84	100
12	student-12	100	70	75	92	100
13	student-13	89	100	76	100	80
14	student-14	85	100	77	89	76
15	student-15	85	65	76	89	NA
16	student-16	92	100	74	89	77
17	student-17	88	63	100	86	78
18	student-18	91	NA	100	87	100

```
19 student-19 91 68 75 86 79
20 student-20 91 68 76 88 76
```

```
#Defining grade() function
grade <- function(scores) {
  # Removing "NA" scores
  scores <- scores[!is.na(scores)]

  # Returning NA if all scores are "NA"
  if(length(scores) == 0) {
    return(NA)
  }

  # Finding the lowest score
  min_score <- min(scores)

  # Removing the lowest score
  scores <- scores[scores != min_score]

  # Computing the average of the remaining scores
  mean(scores)
}

# Applying the grade() function to every row of the example class gradebook
gradebook$overall.grade <- apply(gradebook[,2:6], 1, grade)
gradebook$overall.grade
```

```
[1] 91.75000 82.50000 84.25000 88.00000 88.25000 89.00000 94.00000 93.75000
[9] 91.33333 81.33333 86.00000 91.75000 92.25000 87.75000 83.33333 89.50000
[17] 88.00000 97.00000 82.75000 82.75000
```

Q2. Using your grade() function and the supplied gradebook, who is the top scoring student overall in the gradebook?

```
# Finding the student with the highest overall grade
top_student <- gradebook[which.max(gradebook$overall.grade), 1]
top_student
```

```
[1] "student-18"
```

The top scoring student overall is student-18.

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

```
# Computing the average score for each homework
avg_hw <- apply(gradebook[,2:6], 2, mean)

# Identifying the homework with the lowest average score
toughest_hw <- names(avg_hw)[which.min(avg_hw)]
toughest_hw
```

```
[1] "hw3"
```

Homework 3 had the lowest overall score and therefore, it was the toughest homework for students.

Q4. From your analysis of the gradebook, which homework was most predictive of overall score (i.e. highest correlation with average grade score)?

```
# Computing the correlation between each homework and the overall grades
hw_corr <- sapply(gradebook[,2:6], function(x) cor(x, gradebook$overall.grade, use =
  "everything"))

# Identifying the homework with the highest correlation
predictive_hw <- names(hw_corr)[which.max(hw_corr)]
predictive_hw
```

```
[1] "hw1"
```

Homework 1 was the most predictive of the overall score.