

class06

Michael Preston

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]

```
# Assign scores to start
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)

# define scoring funcs

grade_student <- function(scores){
  # check for NA
  na_count <- sum(is.na(scores))

  # if no NA, drop lowest
  if (sum(is.na(scores)) == 0) {
    top_scores = scores[scores != min(scores)]
  }

  # if one NA, drop it
  else if (na_count == 1) {
    top_scores = scores[!is.na(scores)]
  }

  # if multiple NA, drop first
  else {
    top_scores = scores[-(which(is.na(scores))[1])]
  }
}
```

```

    }

# set remaining NAs to 0
top_scores[is.na(top_scores)] <- 0

# compute mean score
overall_grade = mean(top_scores)

return(overall_grade)
}

# create wrapper function for table
grade <- function(table){
  apply(table[, sapply(table, is.numeric)], 1, grade_student)
}

# score each
for (scores in list(student1, student2, student3)) {
  print(grade_student(scores))
}

```

```

[1] 100
[1] 91
[1] 12.85714

```

Q2.

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

```

# import data
table <- read.csv("C:/Users/micha/Downloads/student_homework.csv", row.names = 1)
table

      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
```

```
# score each
overall_grade <- grade(table)
which.max(overall_grade)
```

```
student-18
18
```

Student 8 scored the highest

Q3.

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

```
# average each column and calc min
average_score <- colMeans(table, na.rm = TRUE)
which.min(average_score)
```

```
hw3
3
```

HW3 was the hardest

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]

```
# correlate each

# set NAs to 0
table_clean <- table
table_clean[is.na(table_clean)] <- 0

corrs = NULL
for (i in 1:ncol(table_clean)) {
  corrs = c(corrs, cor(table_clean[,i], overall_grade))
}
which.max(corrs)
```

[1] 5

HW5 has the highest correlation with average grade score

Q5.

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