Class 6 Lab

Darby Patterson

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:

> 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)
#If a student misses 1 homework it can be dropped with NA.
#Now grade all students using the url below.
url <- "https://tinyurl.com/gradeinput"
```

Now we want to find drop the lowest score of each student

```
#calculate average score of vector dropping lowest
#missing values treated as 0
#exclude lowest score from mean
x<- student1
```

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

```
[1] 100
```

Repeat for student 2 and 3

```
  y<- student2
  y[is.na(y)]<- 0
  mean(y[-which.min(y)])
```

```
[1] 91
```

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

```
[1] 12.85714
```

Now we will extract a function using 'Code > Extract Function'

```
  grade <- function(x) {
    x[is.na(x)]<- 0
    mean(x[-which.min(x)])
  }
```

```
  grade(x)
```

```
[1] 100
```

```
  grade(y)
```

```
[1] 91
```

```
  grade(z)
```

```
[1] 12.85714
```

Now we can apply this function to our class data

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

```
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]

```
results <- apply(gradebook, 1, grade)
sort(results)
```

```
student-15 student-10  student-2 student-19 student-20  student-3  student-4
     78.75      79.00      82.50      82.75      82.75      84.25      84.25
student-11  student-9 student-14 student-17  student-5  student-6 student-16
     86.00      87.75      87.75      88.00      88.25      89.00      89.50
 student-1 student-12 student-13  student-8  student-7 student-18
     91.75      91.75      92.25      93.75      94.00      94.50
```

```
which.max(results)
```

```
student-18
        18
```

This shows us student 18 has the top score with 94.50.

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

```
ave.scores <- apply(gradebook, 2, mean, na.rm=TRUE)
ave.scores
```

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

```
which.min(ave.scores)
```

```
hw3
  3
```

We can also compare the homeworks using median.

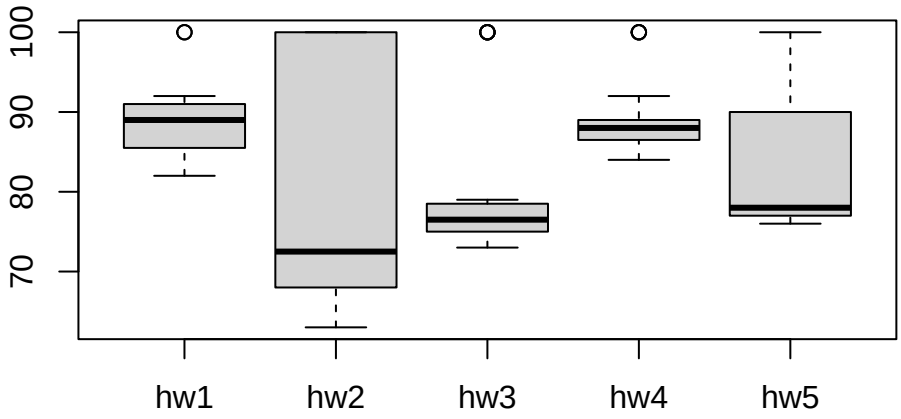
```
med.scores <- apply(gradebook, 2, median, na.rm=TRUE)
med.scores
```

```
 hw1  hw2  hw3  hw4  hw5
89.0 72.5 76.5 88.0 78.0
```

```
which.min(med.scores)
```

```
hw2
  2
```

```
boxplot(gradebook)
```



Because there is so much variation in performance on homework 2, it would be the toughest on students overall. >Q4. From your analysis of the gradebook, which homework was most predictive of overall score (i.e. highest correlation with average grade score)? [1pt]

```
masked.gradebook <- gradebook
masked.gradebook[ is.na(masked.gradebook)] <- 0
masked.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   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
```

```
#and we want to look at the correlation
apply(masked.gradebook,2, cor, x=results)
```

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

```
#the highest correlation would be hw5
```