class06

Zavier Annis

Answers to questions:

Q1 (see code comments in work below): grade <- function(x) { x[is.na(x)] <- 0 mean(x[-which.min(x)]) }

Q2: Student 18

Q3: Homework 2

Q4: Homework 2

Inputting grades:

```
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)
```

Testing which.min:

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

```
[1] 8
```

Trying to make it into a function:

```
drop <- function(x) {
  which.min(x)
}
drop(student1)
```

```
[1] 8
```

Calculate the average using 'mean()':

```
mean(student1)
```

```
[1] 98.75
```

Trying 'which.min()' with NA values:

```
drop(student2)
```

```
[1] 8
```

It didn't work very well (did not return the NA position).

Get the average of one student after dropping their lowest score:

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

```
[1] 100
```

Now trying to modify it so NA is counted as the lowest:

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

```
[1] NA
```

```
mean(student2[-which.min(student2)], na.rm=TRUE)
```

```
[1] 92.83333
```

For student 3?

```
mean(student3[-which.min(student3)], na.rm=TRUE)
```

```
[1] NaN
```

How to deal with multiple NAs?

```
mean(student2[-which.min(student2)], na.rm=0)
```

```
[1] NA
```

Replace NAs with 0s:

```
student3
```

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

```
is.na(student3)
```

```
[1] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
student3[is.na(student3)] <- 0
student3
```

```
[1] 90  0  0  0  0  0  0  0
```

Now we can use the old function:

```
mean(student3[-(which.min(student3))])
```

```
[1] 12.85714
```

Way too much copying and pasting! Time for general:

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

```
[1] 12.85714
```

Boom, it works! Now there's a working snippet of simplified code for any student 'x'.

Now turn into a function:

```
grade <- function(x) {
  x[is.na(x)] <- 0
  mean(x[-which.min(x)])
}
grade(student1)
```

```
[1] 100
```

Q2 work below:

Save the gradebook:

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

take a look:

```
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
```

how does 'apply()' work?

```
apply(gradebook, 1, grade, simplify = TRUE)
```

```
 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
```

Store as results:

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

Which student did best overall?

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

```
student-18
        18
```

What was the score?

```
  results [which.max(results)]
```

```
student-18
      94.5
```

Q3 work below:

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

```
hw2
  2
```

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

```
hw2
  2
```

Q4 work below:

```
  cor(gradebook[,5],results)
```

```
[1] NA
```

still need to mask zeros

```
mask <- gradebook
mask[is.na(mask)] <- 0
cor(mask$hw5,results)
```

```
[1] 0.6325982
```

now apply it to all the homeworks:

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

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