class06

Edward Vo (A16215241)

All about functions in R

Functions are the way we get stuff done in R. We call a function to read data, compute stuff, plot stuff, etc

R makes writing functions accessible but we should always start by trying to get a working snippet of code first before we write our function.

Today's lab

We will grade a whole class of student assignments. We will always try to start with a simplified version of the problem.

Load data

```
# 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 we want the average, we can use the `mean()` function

```
mean(student1)
```

```
[1] 98.75
```

Let's be nice instructors and drop the lowest score so the answer here should be 100. I found the `which.min()` function that may be useful here.

```
min(student1)
```

```
[1] 90
```

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

```
[1] 8
```

```
student1 [8]
```

```
[1] 90
```

```
# same as
student1[which.min(student1)]
```

```
[1] 90
```

I can use the minus syntax trick to get everything but the element with the min value.

```
student1 [-8]
```

```
[1] 100 100 100 100 100 100 100
```

```
# or generically
student1[-which.min(student1)]
```

```
[1] 100 100 100 100 100 100 100
```

I have my first working snippet of code Average of dropped student

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

```
[1] 100
```

Let's test on the other students

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

```
[1] NA
```

Where is the problem?

```
mean(student2)
```

```
[1] NA
```

`mean()` with NA input returns NA by default but I can change this Remove NA

```
mean(student2, na.rm = TRUE)
```

```
[1] 91
```

```
mean(student3)
```

```
[1] NA
```

```
mean(student3, na.rm = TRUE)
```

```
[1] 90
```

I want to stop working with `student1` `student2` `student3` and typing it out every time so let's instead work with an input called `x`

```
x <- student2
```

We want to overwrite the NA values with zero - if you miss a homework you score zero on this homework

`is.na` function?

```
x
```

```
[1] 100  NA  90  90  90  90  97  80
```

```
is.na(x)
```

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

We can use logicals to index a vector

```
y <- 1:5
y
```

```
[1] 1 2 3 4 5
```

```
y>3
```

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

```
y[y>3]
```

```
[1] 4 5
```

```
y[y>3] <- 1000
```

This is my working snippet of code that solves the problem for all my example student inputsr

```
x <- student3

# overwrite NA values to 0
x[is.na(x)] <- 0
x
```

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

```
# remove lowest score
mean(x)
```

```
[1] 11.25
```

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

```
[1] 12.85714
```

```
grade <- function(x) {
  # overwrite NA values to 0
  x[is.na(x)] <- 0
  # remove lowest score
  mean(x[-which.min(x)])
}
```

Use this function:

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

Question 1

```
grade <- function(x) {
  # overwrite NA values to 0
  x[is.na(x)] <- 0
  # remove lowest score
  mean(x[-which.min(x)])
}
```

Question 2

Read the gradebook

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

```r
ans<-apply(gradebook, MARGIN = 1, FUN = "grade")
which.max(apply(gradebook, MARGIN = 1, FUN = "grade"))
```

```
student-18
        18
```

Student 18 is top score

Question 3

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

apply(mask, MARGIN = 2, FUN = "mean")
```

```
  hw1   hw2   hw3   hw4   hw5
89.00 72.80 80.80 85.15 79.25
```

```
which.min(apply(mask, MARGIN = 2, FUN = "mean"))
```

```
hw2
  2
```

Homework 2 is toughest

Question 4

```
apply(mask, MARGIN = 2, cor, y=ans)
```

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

```
which.max(apply(mask, MARGIN = 2, cor, y=ans))
```

```
hw5
  5
```

HW 5 most predictive