

# Lab06

Xiaoyan Wang(A1645055)

This week we are introducing **R functions** and how to write our own functions.

```
# 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)
```

## Questions to answer:

**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]

### Student 1:

Initial grade:

```
#Stright forward mean()
mean(student1)
```

```
[1] 98.75
```

corrected grade:

```
#Drop the lowest score

##Find which element of the vector is the lowest
lowest_score_pos <- which.min(student1)
```

```
##Exclude this lowest score from mean
new_student1 <- student1[-lowest_score_pos]

##Calculate new mean()
mean(new_student1)
```

[1] 100

### Student 2:

Initial grade:

```
#Stright forward mean()
mean(student2, na.rm=TRUE)
```

[1] 91

corrected grade:

```
#Drop the lowest score
new_student2 <- student2

##Find NA
is.na(new_student2)
```

[1] FALSE TRUE FALSE FALSE FALSE FALSE FALSE

```
which(is.na(new_student2))
```

[1] 2

```
##Replace NA with 0
new_student2[is.na(new_student2)] <- 0
mean(new_student2[-which.min(new_student2)])
```

[1] 91

### Student 3:

Initial grade:

```
#Straight forward mean()
mean(student3, na.rm=TRUE)
```

```
[1] 90
```

corrected grade:

```
new_student3 <- student3
new_student3[is.na(new_student3)] <- 0
mean(new_student3[-which.min(new_student3)])
```

```
[1] 12.85714
```

### Creating the function

Take the snippet and turn into a function

Every function has 3 parts

- A name: 'grade()'
- Input arguments: vector of student scores
- The body: our working snippet of code

Using RStudio instill select 'Code > Extract Function'

```
#' Calculate the average score for a vector of homework scores, dropping the lowest score.
#' Missing values will be treated as 0.
#' @param x A numeric vector of homework scores
#'
#' @return Average score
#' @export
#'
#' @examples
#' student <- c(100,NA,90,96)
#' grade(student)
#'
grade <- function(x) {
  #mask NA with 0, treat missing values as 0
  x[is.na(x)] <- 0
  #Exclude the lowest score from mean
  mean(x[-which.min(x)])
}
```

Try grade()

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

**Q2.** Using your grade() function and the supplied gradebook, Who is the top scoring student overall in the gradebook? [3pts]

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

```
#apply the grade function to the gradebook
results <- apply(gradebook,1,grade)
#Arremhe scores from highest to lowest
sort(results, decreasing=TRUE)
```

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

```
#Finding the top student
which.max(results)
```

```
student-18
18
```

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

```
#calculate base on lowest mean
avg_score<-apply(gradebook,2,mean, na.rm= TRUE)
avg_score
```

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

```
which.min(avg_score)
```

```
hw3
3
```

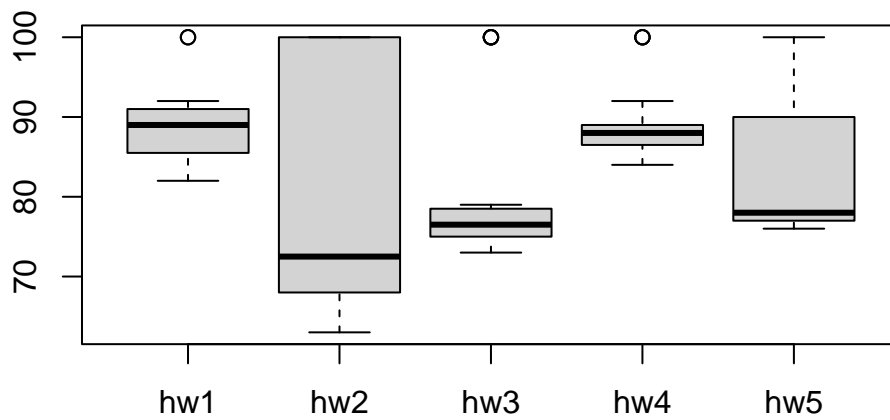
```
#calculate base on lowest median
med_score<-apply(gradebook,2,median, na.rm= TRUE)
med_score
```

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

```
which.min(med_score)
```

```
hw2
2
```

```
boxplot(gradebook)
```



**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]

```
gradebook_copy <- gradebook
gradebook_copy[is.na(gradebook_copy)]<-0
which.max(apply(gradebook_copy,2,cor,results))
```

hw5  
5

**Q5.** Make sure you save your Quarto document and can click the “Render” (or Rmarkdown”Knit”) button to generate a PDF format report without errors. Finally, submit your PDF to gradescope. [1pt]