class06

Jacob Gil

```
# 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)

## removes lowest value from student1
student1[-which.min(student1)]
```

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

```
## changes any NA values to 0
student1[is.na(student1)] <- 0

#' Title
#' drops the lowest value of a vector and averages the remaining values, missing values wi
#' @param x a numerical value of homework scores
#'
#' @return average score
#' @export
#'
#' @examples
#' studentA <- c(100, 90, 100, NA)
#' grade(studentA)
#'
grade <- function(x){
  # treat missing values as 0
  x[is.na(x)] <- 0
  # exclude lowest score from average
  mean(x[-which.min(x)])
}
```

```
grade(student1)
```

```
[1] 100
```

```
## importing data from url
url <- "https://tinyurl.com/gradeinput"

## reads csv data from website
gradebook <- read.csv(url, row.names = 1)

## runs grade function over the imported csv
results <- apply(gradebook, 1, grade)
results
```

```
 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
```

```
## finds the highest scorer in a csv/gradebook
which.max(results)
```

```
student-18
        18
```

```
## finds the hardest hw assignment
ave.scores <- apply(gradebook, 2, mean, na.rm = T)
which.min(ave.scores)
```

```
hw3
  3
```

```
ave.scores
```

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

```r
## changes NA values to 0
masked.gradebook <- gradebook
masked.gradebook[is.na(masked.gradebook)] <- 0

## finds the hw assignment that best determines overall grades
apply(masked.gradebook, 2, cor, x = results)
```

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

```r
which.max(apply(masked.gradebook, 2, cor, x = results))
```

```
hw5
  5
```