class6

Hetian Su

Q1 grade() function

```r
# define function
grade <- function(s){

  # store the overall scores in a new vector
  overall <- c()

  # drop the lowest score in which student's grades and compute mean of the rest
  for (i in 1:nrow(s)){
    #extract row as vector and drop the first entry which is the student's number
    grades <- as.numeric(as.vector(s[i,])[-1])

    #first convert NA to 0
    grades[is.na(grades)] = 0

    #sort in increasing order and drop the first entry
    grades = sort(grades)[-1]

    #append the overall score to the storing vector
    overall <- append(overall, mean(grades))
  }

  return(overall)
}
```

Q2

```r
gradeBook <- read.csv('student_homework.csv')

#call the function to compute overall grade of students
overall <- grade(gradeBook)

#find the max score and its index
Max <- max(overall)
pos <- which(overall==Max)

cat('The student with the best overall score is student',pos)
```

The student with the best overall score is student 18

Q3

```r
# compute the mean score of each homework and assign to new vector
hw <- c()

for(i in 2:ncol(gradeBook)){
  hw <- append(hw, mean(gradeBook[,i], na.rm = TRUE))
}

#find the homework that has the min overall score
Min <- min(hw)
hwPos <- which(hw==Min)

cat('The hardest homework by overall score is homework', hwPos)
```

The hardest homework by overall score is homework 3

Q4

```r
#apply cor function to each hw (column) of the gradebook
pred <- apply(gradeBook[,2:6], 2, cor, y=overall, use='complete.obs')
pred
```

```
       hw1          hw2          hw3          hw4          hw5
 0.42502036   0.61142768   0.30425610  -0.09644108   0.60398041
```

As the correlations show, the second homework is the most predictive one.

Q5

Render as PDF and submit to gradescope.