Class 6: R Functions Lab

Nicholas Chiu

2024-01-25

R Functions

Functions are how we get stuff done. We call functions to do everything useful in R.

All functions in R have at least 3 things:

- A **name** (we get to pick this)
- One or more **input arguments** (parameters)
- The **body** (lines of code that do the work)

```
funname <- function(input1, input2) {
  # The body with R code
}
```

Silly function to add 2 numbers

```
x <- 5
y <- 1
x + y
```

```
[1] 6
```

```
addme <- function(x,y=1) {
  x+y
}
```

```
addme(x,y)
```

```
[1] 6
```

```
addme(10)
```

```
[1] 11
```

Lab for Today

```
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)

grade <- function(vec) {
  #vec[is.na(vec)] <- 0
  mean(vec, na.rm = T)
}

grade(student1)
```

```
[1] 98.75
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 90
```

```
# remove lowest score and find the mean
ind <- which.min(student1)
mean(student1[-ind])
```

```
[1] 100
```

Q1: Writing grade()

```
##create function
grade <- function(x) {
  #change NA to 0
  x[is.na(x)] <- 0

  #calculate mean
  mean(x[-which.min(x)])
}

grade(student3)
```

```
[1] 12.85714
```

```
# Name: grades
# Parameters: CSV file containing a vector of student grades
# Output:
# grade <- function(url) {
#   # Read in CSV
#   gradebook <- read.csv(url, row.names = 1)
#   x[is.na(x)] <- 0
#   mean(x[-which.min(x)])
# }


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

```
results <- apply(gradebook,1,grade)
gradebook$avg <- apply(gradebook,1,grade)
head(gradebook)
```

```
          hw1 hw2 hw3 hw4 hw5   avg
student-1 100  73 100  88  79 91.75
student-2  85  64  78  89  78 82.50
student-3  83  69  77 100  77 84.25
student-4  88  NA  73 100  76 84.25
student-5  88 100  75  86  79 88.25
student-6  89  78 100  89  77 89.00
```

Q2: Top Scoring Student

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

```
student-18
        18
```

Student 18 scored the highest.

Q3: Toughest HW

```
# Lowest score based on mean
which.min(apply(gradebook, 2, mean, na.rm=T))
```

```
hw3
  3
```

```
# Lowest score based on sum
which.min(results2 <- apply(gradebook, 2, sum, na.rm=T))
```

```
hw2
  2
```

HW2 had the lowest scores overall.

Q4: Most predictive

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

cor(mask$hw1, results)
```

```
[1] 0.4250204
```

```r
cor(mask$hw5, results)
```

```
[1] 0.6325982
```

```r
# Use 'apply()' function to run 'cor()' cover the whole course (ie. masked gradebook)
correlation <- apply(mask, 2, cor, results)
which.max(correlation)
```

```
avg
  6
```

HW6 had the most predictive overall score.