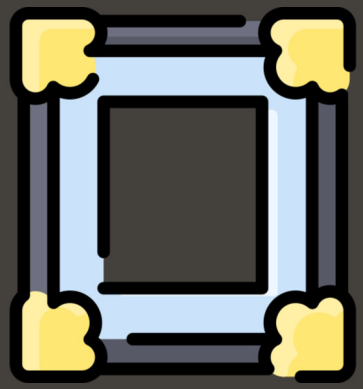
R 프로그래밍 기초다지기

5강 - 데이터 프레임 가지고 놀기

슬기로운통계생활

Issac Lee

Data Frame를 배워보자.



데이터 프레임(Data frame)이란 무엇일까?

2차원 모양의 프레임!

- 행렬의 경우 구성원들이 모두 같은 타입이여야 함.

```
matrix(c(as.character(c(1:5)), 6
       ncol = 2)
```

```
##      [,1] [,2]
## [1,] "1"  "6"
## [2,] "2"  "7"
## [3,] "3"  "8"
## [4,] "4"  "9"
## [5,] "5"  "10"
```

- 데이터 프레임은 각 열의 `mode`가 다를 수 있음.

```
data.frame(col1 = c("one", "two"
           col2 = c(6:10))
```

```
##    col1 col2
## 1   one    6
## 2   two    7
## 3 three    8
## 4  four    9
## 5  five   10
```

데이터 프레임 만들기

특징

- 데이터 분석에서 가장 많이 쓰이는 형태의 자료저장 방법

선언방법

- `data.frame()` 함수를 사용하여 선언
- 각 열에 들어갈 벡터들을 차례대로 넣어줌.

```
name <- c("issac", "bomi")
birthmonth <- c(5, 4)

my_df <- data.frame(name,
                    birthmonth)
my_df
```

```
##    name birthmonth
## 1 issac          5
## 2  bomi          4
```

원소 접근 방법

유연한 접근 방식을 제공

- 열 이름을 `$` 연산자를 사용해서 접근
- 리스트의 특성처럼 각 열을 `[[]]` 기호를 사용해서 접근 가능
- 행렬 형태로 접근 가능함.

```
my_df$name
```

```
## [1] "issac" "bomi"
```

```
my_df[[1]]
```

```
## [1] "issac" "bomi"
```

```
my_df[, 1]
```

```
## [1] "issac" "bomi"
```

csv 파일로 읽어오기

중간고사 성적 데이터

- 링크를 클릭해서 파일을 다운 받아주세요.

```
mydata <-
  read.csv("examscore.csv",
           header = TRUE)
```

- url을 사용해서 바로 읽어오는 것도 가능

```
mydata <- read.csv("https://www.
```

```
head(mydata)
```

```
##   student_id gender midterm f
## 1          1      F      38
## 2          2      M      42
## 3          3      F      53
## 4          4      M      48
## 5          5      M      46
## 6          6      M      51
```

```
dim(mydata)
```

```
## [1] 30  4
```

데이터 프레임 인덱싱(indexing)

행렬 접근 방법 사용하기

- 행렬 접근 방식과 동일하게 `[]`을 이용
- `drop` 옵션을 사용해서 형식을 유지

```
mydata[1:4, 2]
```

```
## [1] "F" "M" "F" "M"
```

```
class(mydata[1:4, 2])
```

```
## [1] "character"
```

```
mydata[1:4, 2, drop = FALSE]
```

```
##   gender
## 1      F
## 2      M
## 3      F
## 4      M
```

```
class(mydata[1:4, 2, drop = FALS
```

```
## [1] "data.frame"
```

NA에 대처하는 우리들의 자세

완전한 표본 체크

```
mydata[1, 2] <- NA
```

- complete.cases(): 모든 열이 꽉 채워져있는 완전한 행들만을 TRUE로 반환
- NA가 제거된 꽉찬 데이터 프레임을 얻기 위해서는 어떻게 해야할까?

```
sum(complete.cases(mydata$gender
```

```
## [1] 29
```

```
complete.cases(mydata)
```

```
##  [1] FALSE  TRUE  TRUE  TRUE
## [13]  TRUE  TRUE  TRUE  TRUE
## [25]  TRUE  TRUE  TRUE  TRUE
```

구성원소 추가/삭제/변경

변경 및 추가

- `$` 기호를 사용하여 새로운 열을 만들기

```
mydata$total <-
  mydata$midterm +
  mydata$final
mydata[1:3, 4:5]
```

```
##   final total
## 1    46    84
## 2    67   109
## 3    56   109
```

- `cbind` 함수 사용

```
mydata <- cbind(mydata,
                mydata$total/2)
names(mydata)[6]
```

```
## [1] "mydata$total/2"
```

```
names(mydata)[6] <- "average"
mydata[1:3, 4:6]
```

```
##   final total average
## 1    46    84    42.0
## 2    67   109    54.5
## 3    56   109    54.5
```

구성원소 추가/삭제/변경

`NULL`을 사용한 삭제

```
mydata$gender <- NULL
head(mydata)
```

```
##   student_id midterm final total average
## 1          1      38    46    84    42.0
## 2          2      42    67   109    54.5
## 3          3      53    56   109    54.5
## 4          4      48    54   102    51.0
## 5          5      46    39    85    42.5
## 6          6      51    74   125    62.5
```

`subset()` 함수를 이용한 필터링

행렬 형식 접근

```
mydata[mydata$midterm <= 15,]
```

```
##    student_id midterm final t
## 20         20       9    33
## 22         22      15    12
```

- `subset()` 이용

```
subset(mydata, midterm <= 15)
```

```
##    student_id midterm final t
## 20         20       9    33
## 22         22      15    12
```

데이터 프레임 합치기

두 개의 데이터를 합쳐보자.

```
mydata2 <- data.frame(id = sampl
                      result = c
mydata2
```

```
##   id result
## 1  4   Pass
## 2  5   Pass
## 3 29   Pass
## 4 19   Fail
## 5 28   Fail
```

```
merge(mydata, mydata2,
      by.x = "student_id",
      by.y = "id")
```

```
##   student_id midterm final tc
## 1          4      48    54
## 2          5      46    39
## 3         19      39    16
## 4         28      52    66
## 5         29      65    78
```

- `all` 옵션은 기본적으로 꺼져있음.

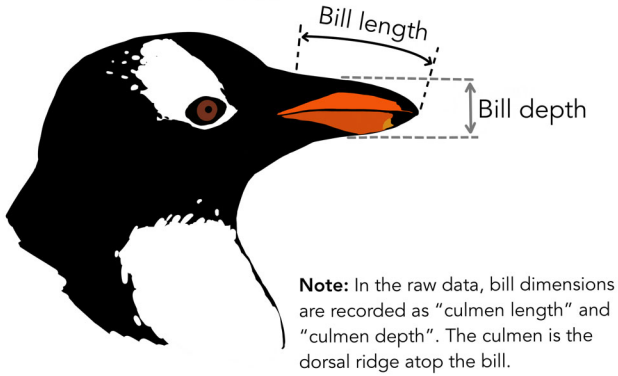
펭귄 데이터셋

데이터 분석 계의 유명인사

- 펭귄 3종 세트


```
# install.packages(palmerpenguir
library(palmerpenguins)
```

- 펭귄 종류별 몸무게와 부리 길이 측정 데이터



- 이제까지 우리가 배운 기술들을 적용해봅시다!

order() 함수를 이용한 정렬

부리 특정 기준을 사용한 분리

```
df_penguins <- data.frame(pengui
str(df_penguins)
```

```
## 'data.frame':    344 obs. of
##  $ species          : Factor
##  $ island           : Factor
##  $ bill_length_mm   : num  39
##  $ bill_depth_mm    : num  18
##  $ flipper_length_mm: int  18
##  $ body_mass_g      : int  37
##  $ sex              : Factor
##  $ year             : int  20
```

```
head(order(df_penguins$bill_leng
```

```
## [1] 143  99  71  93   9  19
```

```
df_penguins[order(df_penguins$bi
```

```
##       species    island bill_
## 143    Adelie     Dream
## 99     Adelie     Dream
## 71     Adelie Torgersen
## 93     Adelie     Dream
## 9      Adelie Torgersen
## 19     Adelie Torgersen
```

`aggregate()` 함수를 사용한 정보 서머리

특정 카테고리 별 수치 요약

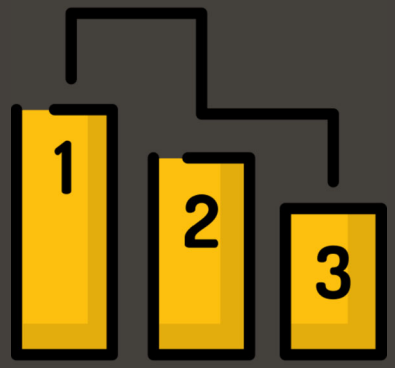
- `aggregate(formula, data, FUN)`
- 수치변수 ~ 카테고리컬 변수
- `+` 연산자와 `.` 연산자 사용가능

- 펭귄 종류별 부리길이

```
aggregate(bill_length_mm ~ speci
          data = df_penguins,
          mean)
```

```
##      species bill_length_mm
## 1     Adelie       38.79139
## 2 Chinstrap       48.83382
## 3     Gentoo       47.50488
```

다음시간



범주형(Factor) 변수

참고자료 및 사용교재

[1] The art of R programming

- R 공부하시는 분이면 꼭 한번 보셔야하는 책입니다.
- 위 교재의 한글 번역본 빅데이터 분석 도구 R 프로그래밍도 있습니다. 도서 제목 클릭하셔서 구매하시면 저의 사리사욕을 충당하는데 도움이 됩니다.

[2] how to download and display an image from an URL in R?