Part 1. R 프로그래밍 (데이터 분석 전문가 양성과정)

06

데이터 프레임

경북대학교 배준현 교수
(joonion@knu.ac.kr)

06. 데이터 프레임

- **데이터 프레임**: data.frame
  - R에서 **2차원 테이블** 형태로 데이터셋을 저장하는 가장 기본적인 자료구조
  - **변수**는 열(*column*)로, **관측값**은 행(*row*)으로 저장

```
   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1           5.1         3.5          1.4         0.2  setosa
2           4.9         3.0          1.4         0.2  setosa
3           4.7         3.2          1.3         0.2  setosa
4           4.6         3.1          1.5         0.2  setosa
5           5.0         3.6          1.4         0.2  setosa
6           5.4         3.9          1.7         0.4  setosa
7           4.6         3.4          1.4         0.3  setosa
8           5.0         3.4          1.5         0.2  setosa
9           4.4         2.9          1.4         0.2  setosa
10          4.9         3.1          1.5         0.1  setosa
```

06. 데이터 프레임

- **데이터 프레임은 벡터의 리스트처럼 생각할 수 있음**

```
> v1 <- 1:7
> v2 <- c('홍길동', '전우치', '주니온', '아사달', '아사녀', '연오랑', '세오녀')
> v3 <- factor(c('M', 'M', 'M', 'M', 'F', 'M', 'F'))
> df <- data.frame(no = v1, name = v2, sex = v3)
> str(df)
'data.frame':  7 obs. of  3 variables:
 $ no  : int  1 2 3 4 5 6 7
 $ name: chr  "홍길동" "전우치" "주니온" "아사달" ...
 $ sex : Factor w/ 2 levels "F","M": 2 2 2 2 1 2 1
> head(df)
  no   name sex
1  1 홍길동   M
2  2 전우치   M
3  3 주니온   M
4  4 아사달   M
5  5 아사녀   F
6  6 연오랑   M
```

06. 데이터 프레임

- 데이터 프레임은 행렬처럼 2차원으로 인덱싱을 할 수 있음

```
> iris[1:5, ]
Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa

> iris[1:5, 1:4]
Sepal.Length Sepal.Width Petal.Length Petal.Width
1          5.1         3.5          1.4         0.2
2          4.9         3.0          1.4         0.2
3          4.7         3.2          1.3         0.2
4          4.6         3.1          1.5         0.2
5          5.0         3.6          1.4         0.2

> iris[1:5, -5]
.....(위와 동일)
```

06. 데이터 프레임

- **데이터 프레임은 리스트처럼 $ 기호로 열 벡터를 가져올 수 있음**

```
> iris$Sepal.Length
  [1] 5.1 4.9 4.7 4.6 5.0 5.4 4.6 5.0 4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4
 [18] 5.1 5.7 5.1 5.4 5.1 4.6 5.1 4.8 5.0 5.0 5.2 5.2 4.7 4.8 5.4 5.2 5.5
.....(이하 생략)
> iris[, 1]
.....(위와 동일)
> iris[, "Sepal.Length"]
.....(위와 동일)

> iris$Species
  [1] setosa     setosa     setosa     setosa     setosa     setosa
  [7] setosa     setosa     setosa     setosa     setosa     setosa
.....(이하 생략)
> iris[, 5]
.....(위와 동일)
> iris[, "Species"]
.....(위와 동일)
```

06. 데이터 프레임

- 하나의 열은 벡터로 다루지만, 하나의 행은 데이터 프레임(인덱싱의 경우에도 마찬가지)

```
> iris[1, ]
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa

> class(iris[1, ])
[1] "data.frame"


> class(iris[, 1])
[1] "numeric"

> class(iris[, 5])
[1] "factor"
```

06. 데이터 프레임

- **데이터 프레임에서도 조건식을 이용하여 필터링할 수 있음**

```
> summary(iris$Sepal.Length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  4.300   5.100   5.800   5.843   6.400   7.900

> iris[iris$Sepal.Length < 5.1, ]
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
2            4.9         3.0          1.4         0.2     setosa
3            4.7         3.2          1.3         0.2     setosa
......(이하 생략)

> iris[iris$Sepal.Length < 5.1 & iris$Species == 'versicolor', ]
   Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
58          4.9         2.4          3.3           1 versicolor
61          5.0         2.0          3.5           1 versicolor
94          5.0         2.3          3.3           1 versicolor

> with(iris, iris[Sepal.Length < 5.1 & Species == 'versicolor', ])
......(위와 동일)
```

- 데이터 프레임에 새로운 변수(열 벡터)를 추가할 수도 있음

```
> df <- iris
> head(df, n = 3)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa

> df$Sepal.Sum <- df$Sepal.Length + df$Sepal.Width

> head(df, n = 3)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species Sepal.Sum
1          5.1         3.5          1.4         0.2  setosa       8.6
2          4.9         3.0          1.4         0.2  setosa       7.9
3          4.7         3.2          1.3         0.2  setosa       7.9
```

- **데이터 객체의 자료형 확인과 변환:**
  - is.xxx() 함수: 데이터 구조의 자료형 확인
  - as.xxx() 함수: 데이터 구조의 자료형 변환
  - 행렬은 데이터 프레임으로 자료형 변환이 가능함

- state.x77: **행렬 형태로 제공되는 R의 내장 데이터셋**

```
> ?state.x77
> str(state.x77)
num [1:50, 1:8] 3615 365 2212 2110 21198 ...
 - attr(*, "dimnames")=List of 2
  ..$ : chr [1:50] "Alabama" "Alaska" "Arizona" "Arkansas" ...
  ..$ : chr [1:8] "Population" "Income" "Illiteracy" "Life Exp" ...

> class(state.x77)
[1] "matrix" "array"

> is.matrix(state.x77)
[1] TRUE
> is.data.frame(state.x77)
[1] FALSE
```

06. 데이터 프레임

- state.x77 데이터셋을 행렬에서 데이터 프레임으로 변환

```
> df.x77 <- as.data.frame(state.x77)
> is.data.frame(df.x77)
[1] TRUE

> str(df.x77)
'data.frame':  50 obs. of  8 variables:
 $ Population: num  3615 365 2212 2110 21198 ...
 $ Income    : num  3624 6315 4530 3378 5114 ...
 $ Illiteracy: num  2.1 1.5 1.8 1.9 1.1 0.7 1.1 0.9 1.3 2 ...
 $ Life Exp  : num  69 69.3 70.5 70.7 71.7 ...
 $ Murder    : num  15.1 11.3 7.8 10.1 10.3 6.8 3.1 6.2 10.7 13.9 ...
 $ HS Grad   : num  41.3 66.7 58.1 39.9 62.6 63.9 56 54.6 52.6 40.6 ...
 $ Frost     : num  20 152 15 65 20 166 139 103 11 60 ...
 $ Area      : num  50708 566432 113417 51945 156361 ...
```

데이터 프레임의 저장과 불러오기:

- 연구 데이터의 관리:
  - 엑셀 파일로 저장한 연구 데이터를 데이터 프레임으로 불러오기
  - 데이터 처리를 완료한 연구 데이터를 엑셀 파일로 저장하기
- CSV 파일: Comma Separated Value
  - R에서는 CSV 파일을 읽고 쓰는 함수가 기본적으로 제공됨
    - write.csv(), read.csv()
- 엑셀 파일 직접 읽어오기: readxl 패키지 활용

06. 데이터 프레임

- iris 데이터셋을 csv 파일로 저장하고 다시 불러오기

```
> getwd()
[1] "D:/R/R-for-Research“
> write.csv(iris, file = "iris.csv")
> write.csv(iris, file = "iris2.csv", row.names = F)
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3	1.4	0.1	setosa
14	4.3	3	1.1	0.1	setosa
15	5.8	4	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa

```
> df1 <- read.csv(file = "iris.csv")
> str(df1)
'data.frame':  150 obs. of  6 variables:
 $ X           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : chr  "setosa" "setosa" "setosa" "setosa" ...

> df2 <- read.csv(file = "iris2.csv", stringsAsFactors = T)
> str(df2)
'data.frame':  150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

- **엑셀 파일로 저장한 데이터를 엑셀에서 직접 읽어오기**

```
> install.packages("readxl")
> library(readxl)
> df <- read_excel(path = "mydata.xlsx", sheet = 1)
> str(df)
tibble [3 × 4] (S3: tbl_df/tbl/data.frame)
 $ No    : num [1:3] 1 2 3
 $ Name  : chr [1:3] "홍길동" "전우치" "주니온"
 $ Sex   : chr [1:3] "남" "남" "남"
 $ Height: num [1:3] 158 175 182

> head(df)
# A tibble: 3 × 4
     No Name   Sex   Height
  <dbl> <chr>  <chr>  <dbl>
1     1 홍길동 남       158
2     2 전우치 남       175
3     3 주니온 남       182
```

- **연습문제 6.1:**
  - state.x77 데이터셋에 대하여 R 코드를 작성하시오.
    - state.x77 데이터셋을 st 변수에 저장: 데이터 프레임 형태로 저장할 것
    - st 데이터 프레임의 변수와 관측값의 개수는?
    - 각 주별 소득(Income)의 평균은?
    - 인구(Population)가 10,000보다 큰 주의 인구, 소득은?
    - Florida 주의 인구와 소득은?
      - rownames(st) 는 st 각 주의 이름 벡터를 리턴한다.
    - 인구가 1,000보다 작고, 소득이 4,436보다 작은 주의 모든 정보를 출력하라.
    - 문맹률(Illiteracy)의 평균을
      - 소득이 5,000보다 작은 주에 대해서 구하라.
      - 소득이 5,000보다 큰 주에 대해서 구하라.

- **연습문제 6.2:**
  - state.x77 데이터셋에 대하여 R 코드를 작성하시오.
    - 인구가 1,000보다 작고, 소득이 5,000보다 작은 주의 모든 정보를 출력하라.
    - 문맹률(Illiteracy)의 평균을
      - 소득이 5,000보다 작은 주에 대해서 구하라.
      - 소득이 5,000보다 큰 주에 대해서 구하라.
    - 위의 결과로 다음과 같은 진술이 타당하다고 할 수 있는가?
      - 소득이 높으면 문맹률이 낮아진다.
      - 소득이 낮으면 문맹률이 높아진다.

- **연습문제 6.3:**
  - **아래와 같은 엑셀 파일을 만드시오:** scores.xlsx
    - **엑셀 파일을 R에서 데이터 프레임으로 읽으시오:** read_excel() **사용**
    - **각 학생별로 성적의 합계와 평균을 구하시오.**
      - `df$Sum <- df$Kor + df$Eng + df$Math`
    - **합계와 평균을 포함한 파일을** result.csv **파일로 저장하시오.**

scores.xlsx

No	Name	Kor	Eng	Math
1	A	90	70	60
2	B	60	40	70
3	C	100	50	80

result.csv

No	Name	Kor	Eng	Math	Sum	Mean
1	A	90	70	60	220	73.33333
2	B	60	40	70	170	56.66667
3	C	90	50	80	220	73.33333

Any Questions?