Data Analysis with R - Day 4

ylee@dongguk.edu // dryl@icloud.com

2019-5-28

1. 문제 풀이
2. 패키지
3. 벡터의 이해: 심화

문제 풀이

패키지

R Packages

지금까지 다룬 것은 R 기본. 하지만 R의 강점은 제삼자가 제작해서 보급하는 R 패키지에 있다고 보아도 무방하다. 남이 작성한 코드를 읽을 때 난점의 하나로 꼽힌다.

- From CRAN repository

```
install.packages("package-name")
library(package-name)
```

- From GitHub repository

```
# https://cran.r-project.org/web/packages/githubinstall/vignettes/githubinstall.h
library(devtools)
install_github(repo="repository-name/package-name")
library(package-name)
```

- From Zip file

```
install.packages("path/package-name.zip")
library(package-name)
```

R 패키지의 다양성

- CRAN: https://www.r-project.org
- 14,299 available packages (May 27, 2019)
- Another scores of thousand packages hosetd by Github

```
x <- available.packages()
dim(x)

## [1] 14266    17

x[1:10, 1:4]

##             Package       Version  Priority
## A3          "A3"          "1.0.0"  NA
## abbyyR      "abbyyR"      "0.5.4"  NA
## abc         "abc"         "2.1"    NA
## abc.data    "abc.data"    "1.0"    NA
## ABC.RAP     "ABC.RAP"     "0.9.0"  NA
## ABCanalysis "ABCanalysis" "1.2.1"  NA
## abcdeFBA    "abcdeFBA"    "0.4"    NA
## ABCoptim    "ABCoptim"    "0.15.0" NA
## ABCp2       "ABCp2"       "1.2"    NA
## abcrf       "abcrf"       "1.7.1"  NA
##             Depends
## A3          "R (>= 2.15.0), xtable, pbapply"
## abbyyR      "R (>= 3.2.0)"
## abc         "R (>= 2.10), abc.data, nnet, quantreg, MASS, locfit"
## abc.data    "R (>= 2.10)"
## ABC.RAP     "R (>= 3.1.0)"
## ABCanalysis "R (>= 2.10)"
## abcdeFBA    "Rglpk,rgl,corrplot,lattice,R (>= 2.10)"
## ABCoptim    NA
## ABCp2       "MASS"
## abcrf       "R(>= 3.1)"
```

Package: magrittr

파이프 연산자를 써서 왼쪽에서 오른쪽으로 (읽는 방향) 객체 또는 객체의 값을 넘겨 가며 연산하는 방식

```
x <- rnorm(10, mean = 17)
x
```

```
##  [1] 16.90894 17.20944 16.97401 16.39184 15.58436 15.78340 17.58796
##  [8] 16.95277 17.02220 16.96581
```

```
mean(x)
```

```
## [1] 16.73807
```

```
require(magrittr)
x %>% mean
```

```
## [1] 16.73807
```

```
# 다음 벡터 y에서 0보다 큰 숫자의 개수는?

y <- c(2, 4, 0, -3, -2, NA, 10, -1, NA)
sum(y > 0)
```

```
## [1] NA
```

```
y %>% ">"(0) %>% sum
```

```
## [1] NA
```

```
# y에서 결측값의 개수는?
y %>% is.na %>% sum
```

```r
# 0도에서 180도까지의 각도를 30도 간격으로 변수 d에 담고 radian으로 변환한 뒤
# 다시 sin 값을 구해서 소수점 두 자리까지 절삭하려면?

(d <- seq(0, 180, by = 30))

## [1]   0  30  60  90 120 150 180

d %>% "*"(pi) %>% "/"(180) %>% sin %>% round(2)

## [1] 0.00 0.50 0.87 1.00 0.87 0.50 0.00

#
# 그렇게 처리한 sin 값을 다시 d에 담으려면?
#

d %<>% "*"(pi) %>% "/"(180) %>% sin %>% round(2)

#
# 이 과정을 pipe 연산자 없이 처리하려면?
#

(d <- seq(0, 180, by = 30))

## [1]   0  30  60  90 120 150 180

(d <- round(sin(d * pi / 180), 2))

## [1] 0.00 0.50 0.87 1.00 0.87 0.50 0.00
```

파이프 연산자에서 함수 용례 대비표

일반	파이프 연산
sin(x), cos(x)	`x %>% sin, x %>% cos`
rep(c("M", "F"), 2)	`c("M", "F") %>% rep(2)`
5 + 3, 6 * 7	`5 %>% "+"(3), 6 %>% "*"(7)`
s <- s + 1	`s %<>% "+"(1)`
r <- d * pi / 180	`r <- d %>% "*"(pi) %>% "/"(180)`
	`d %>% "*"(pi) %>% "/"(180) -> r`
x[1:2]	`x %>%"["(1:2)`

파이프 연산자가 항상 더 보기 쉬운 것은 아니고 어쩌면 R의 효율을 낮출 가능성까지 내포하고 있지만, 분명한 점은 괄호의 내포로 인한 오류(매우 흔함!)를 근원적으로 제거할 수 있다. 미지의 숫자벡터로부터 짝수만 걸러내는 R구문은 이렇게 다르다.

```
# 파이프
x <- rnorm(100, mean = 100) %>% as.integer
x %>% "["(x %>% "%%"(2) %>% "=="(0))

##  [1] 102 100 100  98 100  98 102 100 100 100 100  98  98 100  98  98 100
## [18] 100  98 100 100 100 100  98 100  98 100  98  98 100 100 100 100 100
## [35] 100 100 100 100 100  98  98 100  98  98  96 100  98 100 100 100

# 일반
x[x %% 2 == 0]

##  [1] 102 100 100  98 100  98 102 100 100 100 100  98  98 100  98  98 100
```

Package: ggplot2

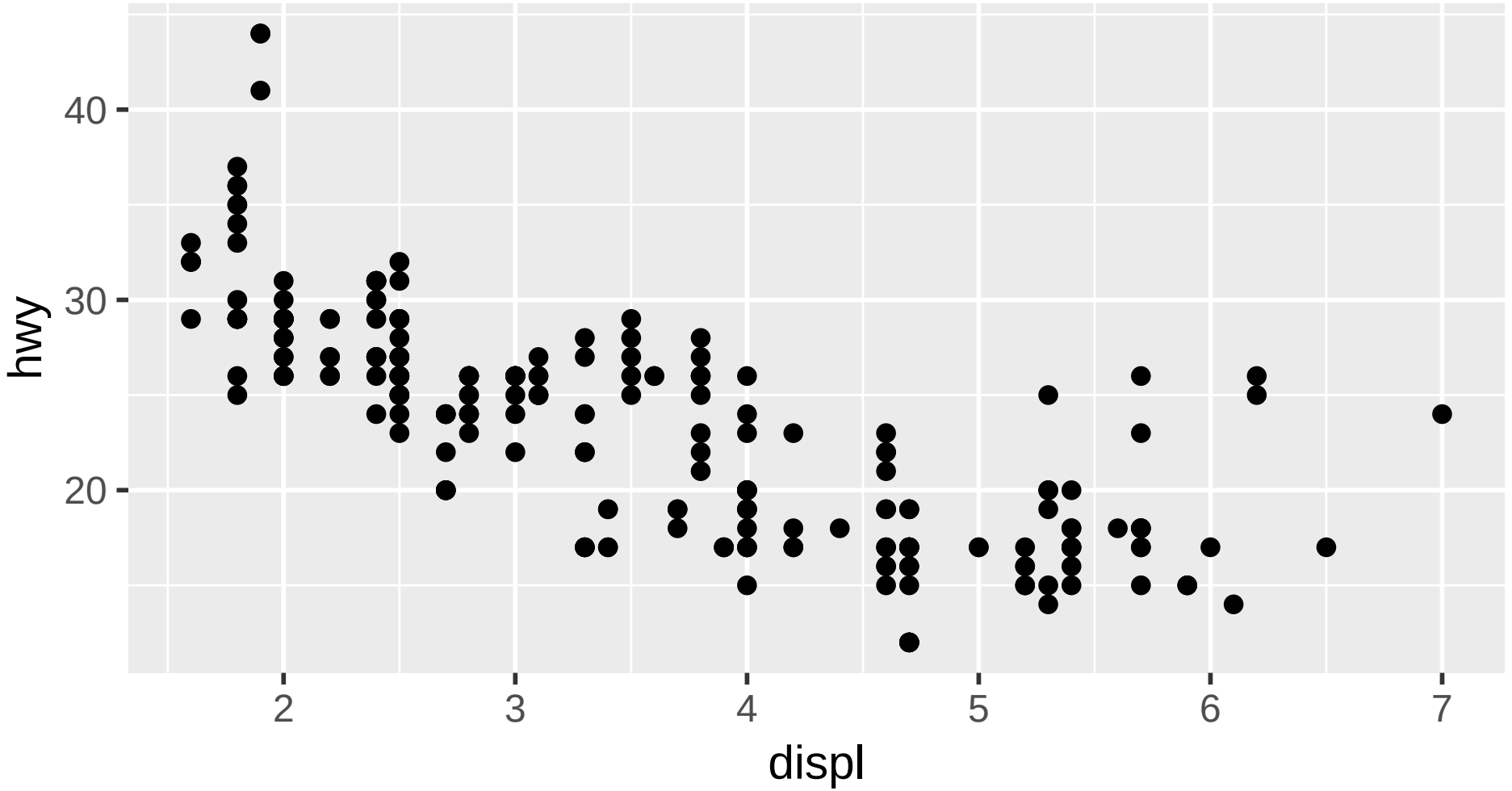
R에서 grammar of graphics를 명시적으로 구현함.

```
require(ggplot2)
str(mpg)
```

> 함수 `require()`와 `library()`는 공히 패키지의 내용을 메모리에 올리는 데 쓰이는데 내부 동작이 미묘하게 다르다. 만약 설치되지 않은 패키지를 부를 때 library()는 에러를 내지만 `require()`는 `FALSE`를 반환할 뿐이다. 이 차이는 다양한 실전에서 달리 응용될 수 있다.

ggplot 패키지는 plot = data + geometry + aesthetic으로 그래프를 구상화한다:

```
ggplot(data = mpg) + geom_point(aes(x = displ, y = hwy))
```




```
# ggplot(mpg, aes(x = displ, y = hwy)) + geom_point()
```

중간과제 1

3차 실습에서 다룬 Crampton의 음식물 내 아스코르브산 동물실험 그래프 작성과 동등한 geometry와 aesthetic을 ggplot2로 표현할 수 있다.

```
# do not run
# jump to the script
str(ToothGrowth)

## 'data.frame':^^I60 obs. of  3 variables:
##  $ len : num  4.2 11.5 7.3 5.8 6.4 10 11.2 11.2 5.2 7 ...
##  $ supp: Factor w/ 2 levels "OJ","VC": 2 2 2 2 2 2 2 2 2 2 ...
##  $ dose: num  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
```

Component	Content
DATA	ToothGrowth
ELEMENT	schema (position (bin.quantile.letter(dose * len)), color(supp))
GUIDE	axis (dim(1), limit(0.5, 2), discrete.tick(0.5, 1, 2), label = "dose of ascorbic acid (mg)")
GUIDE	axis (dim(2), limit(0, 35), tick(5), label = "tooth length (mm) ")

```r
ggplot(data = ToothGrowth, aes(x = factor(dose), y = len, fill = supp)) + geom_boxpl

tG <- ToothGrowth
tG$supp <- factor(tG$supp, levels = c("VC", "OJ"),
    labels = c("Vitamin capsule", "Orange juice")) #

ggplot(data = tG, aes(x = factor(dose), y = len,
    fill = supp)) + geom_boxplot() #
```

```r
# graph can be stored as an object

g <- ggplot(data = tG,
    aes(x = factor(dose, labels = c("0.5", "1.0", "2.0")),
            y = len, fill = supp)) + geom_boxplot() #

# Try google "how to change legend of ggplot2"

g + scale_fill_manual(values = c("yellow", "orange"))
g + scale_fill_manual(values = c("yellow", "orange")) +
    xlab("dose of ascorbic acid (mg)" ) +
        ylab("Tooth Length (mm)") #
g + scale_fill_manual(values = c("yellow", "orange")) +
    xlab("dose of ascorbic acid (mg)" ) +
        ylab("Tooth Length (mm)") +
        theme(legend.title = element_blank(), legend.position = c(0.8, 0.2)) #
```

패키지 car

John Fox's (Now, J.Fox and S. Weisberg's) book; Companion to Applied Regression

```
data(Orthodont, package = "nlme")
colnames(Orthodont)

## [1] "distance" "age"      "Subject"  "Sex"

leveneTest(Orthodont$distance, Orthodont$Sex, center = median) # invokes error

## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   1  1.4056 0.2384
##       106

library(car)
leveneTest(Orthodont$distance, Orthodont$Sex, center = median)

## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   1  1.4056 0.2384
##       106
```

패키지: nlme and lme4

Pinheiro and Bates's book "Mixed effect modelling in S and S+" vs. Bates's extension for nlme

```
library(nlme)
m1 <- lme(distance ~ Sex, random = ~ 1 | Subject, data = Orthodont)
print(m1)
```

```
## Linear mixed-effects model fit by REML
##   Data: Orthodont
##   Log-restricted-likelihood: -252.9359
##   Fixed: distance ~ Sex
## (Intercept)   SexFemale
##   24.968750   -2.321023
##
## Random effects:
##  Formula: ~1 | Subject
##         (Intercept) Residual
## StdDev:    1.595839 2.220312
##
## Number of Observations: 108
## Number of Groups: 27
```

```
library(lme4)
m2 <- lmer(distance ~ Sex + (1 | Subject), data = Orthodont)
print(m2)

## Linear mixed model fit by REML ['lmerMod']
## Formula: distance ~ Sex + (1 | Subject)
##    Data: Orthodont
## REML criterion at convergence: 505.8718
## Random effects:
##  Groups   Name        Std.Dev.
##  Subject  (Intercept) 1.596
##  Residual             2.220
## Number of obs: 108, groups:  Subject, 27
## Fixed Effects:
## (Intercept)    SexFemale
##      24.969       -2.321
```

패키지의 판권 정보: 논문에서 인용하려면?

https://ekja.org/journal/view.php?number=8303

```
citation("car")

##
## To cite the car package in publications use:
##
##   John Fox and Sanford Weisberg (2011). An {R} Companion to
##   Applied Regression, Second Edition. Thousand Oaks CA: Sage. URL:
##   http://socserv.socsci.mcmaster.ca/jfox/Books/Companion
##
## A BibTeX entry for LaTeX users is
##
##   @Book{,
##     title = {An {R} Companion to Applied Regression},
##     edition = {Second},
##     author = {John Fox and Sanford Weisberg},
##     year = {2011},
##     publisher = {Sage},
##     address = {Thousand Oaks {CA}},
##     url = {http://socserv.socsci.mcmaster.ca/jfox/Books/Companion},
##   }
```

```
citation("lme4")
```

```
##
## To cite lme4 in publications use:
##
##   Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015).
##   Fitting Linear Mixed-Effects Models Using lme4. Journal of
##   Statistical Software, 67(1), 1-48. doi:10.18637/jss.v067.i01.
##
## A BibTeX entry for LaTeX users is
##
##   @Article{,
##     title = {Fitting Linear Mixed-Effects Models Using {lme4}},
##     author = {Douglas Bates and Martin M{\"a}chler and Ben Bolker and Steve Walke
##     journal = {Journal of Statistical Software},
##     year = {2015},
##     volume = {67},
##     number = {1},
##     pages = {1--48},
##     doi = {10.18637/jss.v067.i01},
##   }
```

Tools for data import and analysis

Total 8,037 PMIDs were batch-queried from PubMed, using rentrez package (rentrez: Entrez in R. David Winter., R package version 1.0.4). Matched DOIs were queried using the PMIDs as key variables using the ID Converter application programming interface (API) ([7]).

The article-based 3 timestamps; 'received date,' 'accepted date,' and 'epublished date' were obtained through XML-parsing and mining of the custom metadata field of the Crossref article page, scraped in the form of XML using author-developed R code that utilized API calling procedures provided by Crossref ([8]).

Randomness in selecting journal names was assured using dplyr package (dplyr: A Grammar of Data Manipulation. Hadley Wickham and Romain Francois., R package version 0.5.0). Throughout the data acquisition and analyses; API procedures for web scraping, data handling, graphing, and statistical analyses were powered by R software version 3.3.2 (R: A language and environment for statistical computing; R Foundation for Statistical Computing, Vienna, Austria) added on GNU Emacs version 25.1.1 (Free Software Foundation, Inc., Boston, MA, USA; 2016). Linear mixed-effects models were constructed using the lme4 package (lme4: R package for linear mixed-effects models. Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker, R package version 1.0.+) ([9]), with maximum likelihood method. Since the authors planned 2 inferential tests separately on 2 dependent variables (acceptance and lead lag), each inference was targeted to α value of 0.025, keeping overall α value, 0.05. So, CIs in this report were within 97.5%, as well. Subsidiary *P* values were attained by performing the likelihood ratio test against a null model. The journal names were set in italicized International Organization for Standardization (ISO)-abbreviation format.

벡터의 이해: 심화

$\sum_{i=1}^{20} (x1_i - \bar{x1})^2$ 을 R로 계산한다.

```
x1 <- c(5, 4, 9, 3, 7, 5, 8, 10, 11, 6,
    8, 11, 7, 7, 10, 9, 12, 8, 9, 10)

# 1. loop operation
m <- mean(x1)
s <- 0
for (i in 1:20) s <- s + (x1[i] - m)^2
print(s)

## [1] 114.95

# 2. complete vector operation
sum((x1 - mean(x1))^2)

## [1] 114.95

# 3. 2 + pipe
x1 %>% "-"(x1 %>% mean) %>% "^"(2) %>% sum

## [1] 114.95
```

```r
x2 <- c(11, 4, 10, 10, 8, 7, 10, 16, 8, 9, 16, 12, 10, 10, 11, 3, 11, 11, 10, 14) #

# 10보다 큰 자료 찾기

x2 > 10 # 벡터 내 위치를 T/F로 찾아서
```

```
##  [1]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE
## [12]  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE  TRUE
```

```r
x2[x2 > 10] # x2 내에서 찾아서 출력
```

```
## [1] 11 16 16 12 11 11 11 14
```

```r
(x2 > 10) %>% "["(x2) # compare with: x > 10 %>% "["(x2)
```

```
##  [1]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE
## [12]  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
```

```r
sum(x2 > 10) # x2 > 10에서 TRUE의 합 = 정확하게는 1의 개수
```

```
## [1] 8
```

```r
length(x2[x2 > 10]) # 또는 x2 내에서 그 값을 찾아서 그 벡터의 길이로 측정
```

```
## [1] 8
```

```r
table(x2 > 10) # 10이 넘는가를 단지 T/F로 합
```

```
##
## FALSE  TRUE
##    12     8
```

```
x2
```

```
##  [1] 11  4 10 10  8  7 10 16  8  9 16 12 10 10 11  3 11 11 10 14
```

```
# 두 번 이상 나오는 숫자 찾기
table(x2)
```

```
## x2
##  3  4  7  8  9 10 11 12 14 16
##  1  1  1  2  1  6  4  1  1  2
```

```
table(x2) > 2
```

```
## x2
##     3     4     7     8     9    10    11    12    14    16
## FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE
```

```
rbind(table(x2) > 2, table(x2))
```

```
##      3 4 7 8 9 10 11 12 14 16
## [1,] 0 0 0 0 0  1  1  0  0  0
## [2,] 1 1 1 2 1  6  4  1  1  2
```

```
# 벡터 내 어느 요소라도 10보다 큰 게 있는가?
any(x2 > 10)
```

```
## [1] TRUE
```

```
# 벡터 내 모든 요소가 10보다 큰가?
all(x2 > 10)
```

```
## [1] FALSE
```

문자열벡터: string, character

```
# 세 개의 요소를 가진 문자열벡터 c1에서 "John"을 포함한 요소를 찾는다
c1 <- c("Stephen King", "John Snow", "John Steinbeck")
grep("John", c1)

## [1] 2 3

c1[grep("John", c1)]

## [1] "John Snow"      "John Steinbeck"

# 대소문자 구분은 필수적이지만
grep("john", c1)

## integer(0)

# 필요에 따라 대소문자 하나로 통일해서 검색할 수 있다.
grep("john", tolower(c1))

## [1] 2 3

grep("JOHN", toupper(c1))

## [1] 2 3
```

```r
# 문자열벡터의 길이는 length()로 구하지만 문자열의 길이는 nchar()로 구한다.
c1
```

```
## [1] "Stephen King"   "John Snow"      "John Steinbeck"
```

```r
length(c1)
```

```
## [1] 3
```

```r
nchar("John Snow")
```

```
## [1] 9
```

```r
# 문자열은 규칙적 방식으로 붙일 수 있다.
paste("Stephen", "King")
```

```
## [1] "Stephen King"
```

```r
paste("Stephen", "King", sep = "")
```

```
## [1] "StephenKing"
```

```r
paste("Johns,", "Snow", "and", "Steinbeck")
```

```
## [1] "Johns, Snow and Steinbeck"
```

```r
paste("Johns,", "Snow", "and", "Steinbeck") %>% length
```

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

```r
# 문자열에서 문자 골라내기

substr("Rebecca Solnit", 3, 5)

## [1] "bec"

substr("Rebecca Solnit", 1, nchar("Rebecca Solnit"))

## [1] "Rebecca Solnit"

# 특정 분리마커에 따른 문자열 분리. 분리 후 처리
strsplit("Rebecca Solnit", split = " ")

## [[1]]
## [1] "Rebecca" "Solnit"

"2019-05-28" %>% strsplit(split = "-") -> D
typeof(D)

## [1] "list"

str(D)

## List of 1
##  $ : chr [1:3] "2019" "05" "28"

D %>% unlist %>% paste(collapse = "/")

## [1] "2019/05/28"
```

중간과제 2

theKorea 자료철은 2011년부터 2016년 연간에 영국의 Times Higher Education이 선발하는 우수대학 리스트의 상위권에 오른 국내대학의 역량을 순위(ranking), 교수법(teaching), 국제적전도유망성(international.outlook), 연구역량(research), 피인용(citations), 산학재원(industry.income), 전반(overall)으로 나누어 기재하였다.

```
theKorea <- read.csv("https://raw.githubusercontent.com/ylee03/r_for_students/master/theKorea
str(theKorea)

## 'data.frame':^^I61 obs. of  9 variables:
##  $ year                 : int  2016 2016 2016 2016 2016 2016 2016 2016 2016 2016 ...
##  $ ranking              : Factor w/ 28 levels "109","116","124",..: 27 2 4 5 12 14 14 16 1
##  $ name                 : Factor w/ 26 levels "Ajou University\nSouth Korea",..: 18 15 11
##  $ teaching             : num  66.5 49.1 43.3 51.4 43.9 35.2 37.2 33.9 23.9 26.1 ...
##  $ international.outlook: num  30.9 33.7 33.9 36.7 40.2 36.2 41.5 54.3 34.3 46.4 ...
##  $ research             : num  70.5 47.1 40.5 53.5 43.4 42 34.1 32.6 18.3 27.6 ...
##  $ citations            : num  50 76.7 75.9 53.8 41.8 43.7 39.4 31.1 50.4 39.2 ...
##  $ industry.income      : num  85.4 100 100 97.5 99.8 45.2 75.9 87.9 57.6 89.4 ...
##  $ overall              : num  60.5 56.9 53 52.8 NA NA NA NA NA NA ...
```

이 자료에는 작성자의 뜻과는 달리 국내대학이 아닌 대학이 끼어들어 있다. 오염을 찾아서 제거하시오.

```
grep("Korea", theKorea$name)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
## [24] 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 47
## [47] 48 49 51 52 53 54 55 56 57 58 59 60 61
```

```
theKorea[grep("Korea", theKorea$name), ] -> newKorea
```

자료 모으고(gather) 펼치기(spread)

Table: Wide form

subject	sex	control	cond1	cond2
1	M	7.9	12.3	10.7
2	F	6.3	10.6	11.1
3	F	9.5	13.1	13.8
4	M	11.5	13.4	12.9

Table: Long form

subject	sex	condition	measurement
1	M	control	7.9
1	M	cond1	12.3
1	M	cond2	10.7
2	F	control	6.3
2	F	cond1	10.6
2	F	cond2	11.1
3	F	control	9.5
3	F	cond1	13.1
3	F	cond2	13.8
4	M	control	11.5
4	M	cond1	13.4
4	M	cond2	12.9

subject	sex	control	cond1	cond2
1	M	7.9	12.3	10.7
2	F	6.3	10.6	11.1
3	F	9.5	13.1	13.8
4	M	11.5	13.4	12.9

subject	sex	condition	measurement
1	M	control	7.9
1	M	cond1	12.3
1	M	cond2	10.7
2	F	control	6.3
2	F	cond1	10.6
2	F	cond2	11.1
3	F	control	9.5
3	F	cond1	13.1
3	F	cond2	13.8
4	M	control	11.5
4	M	cond1	13.4
4	M	cond2	12.9

- Gather: wide one into LONG form
- Spread: long one into WIDE form

```
thisWideData <- read.csv("https://raw.githubusercontent.com/ylee03/
r_for_students/master/thisWideData.csv", header = TRUE) #
```

```r
# solution 1: manual
longData <- data.frame(
        subject = 1:4 %>% rep(3) %>% factor,
        sex = thisWideData$sex %>% rep(3),
        condition = c("control", "cond1", "cond2") %>% rep(each = 4),
        measurement = thisWideData[, 2:4] %>% as.matrix %>% as.vector)
longData <- longData[order(longData$subject), ]
longData %>% head
```

```
##    subject sex condition measurement
## 1        1   M   control         7.9
## 5        1   M     cond1        12.3
## 9        1   M     cond2        10.7
## 2        2   F   control         6.3
## 6        2   F     cond1        10.6
## 10       2   F     cond2        11.1
```

```r
# solution 2: gather from the tidyr package
library(tidyr)
gather(thisWideData, key = condition, value = measurement,
    control, cond1, cond2) #
```

```
##    sex condition measurement
## 1    M   control         7.9
## 2    F   control         6.3
## 3    F   control         9.5
## 4    M   control        11.5
## 5    M     cond1        12.3
## 6    F     cond1        10.6
## 7    F     cond1        13.1
## 8    M     cond1        13.4
## 9    M     cond2        10.7
## 10   F     cond2        11.1
## 11   F     cond2        13.8
## 12   M     cond2        12.9
```

```r
# gather(thisWideData, key = condition, value = measurement, 2:4) #
# gather(thisWideData, condition, measurement, 2:4) #
```

```
# solution: long to wide

(m <- longData$measurement %>%  matrix(byrow = TRUE, ncol = 3))

##      [,1] [,2] [,3]
## [1,]  7.9 12.3 10.7
## [2,]  6.3 10.6 11.1
## [3,]  9.5 13.1 13.8
## [4,] 11.5 13.4 12.9

(h <- longData[0:3 * 3 + 1, 1:2])

##   subject sex
## 1       1   M
## 2       2   F
## 3       3   F
## 4       4   M

w <- cbind(h, m)
colnames(w)[3:5] <- c("control", "cond1", "cond2")
w %>% print

##   subject sex control cond1 cond2
## 1       1   M     7.9  12.3  10.7
## 2       2   F     6.3  10.6  11.1
## 3       3   F     9.5  13.1  13.8
## 4       4   M    11.5  13.4  12.9

# solution: tidyr::spread
# spread(longData, key = condition, value = measurement)
```

중간과제 3

Orthodont 자료철은 피험자(남자 16, 여자 11) 당 4회 측정된 해부학적 거리(distance)가 종형(long form)으로 저장되어 있음을 기억하라. 기본 R 문형을 써서 이 자료철을 한 피험자 당 한 행을 차지하는 횡형(wide form)으로 변환하라.

```
data(Orthodont, package = "nlme")
Orthodont %>% head(5)

## Grouped Data: distance ~ age | Subject
##   distance age Subject  Sex
## 1     26.0   8     M01 Male
## 2     25.0  10     M01 Male
## 3     29.0  12     M01 Male
## 4     31.0  14     M01 Male
## 5     21.5   8     M02 Male

d <- Orthodont$distance %>% matrix(nrow = 4) %>% t
colnames(d) <- 4:7 * 2
d %<>% as.data.frame()
d$Sex <- c(rep("M", 16), rep("F", 11))
d %>% head

##      8   10   12   14 Sex
## 1 26.0 25.0 29.0 31.0   M
## 2 21.5 22.5 23.0 26.5   M
## 3 23.0 22.5 24.0 27.5   M
## 4 25.5 27.5 26.5 27.0   M
## 5 20.0 23.5 22.5 26.0   M
## 6 24.5 25.5 27.0 28.5   M
```

벡터 연산 심화: apply(), tapply()

```
# 위의 d에서 특정 연령대 전체 피험자의 측정값의 평균은?
# loop로? --- no!
apply(d[, 1:4], MARGIN = 1, FUN = mean) # MARGIN = 1 (row operation)
```

```
##  [1] 27.750 23.375 24.250 26.625 23.000 26.375 23.750 23.875 25.125 29.500
## [11] 23.625 24.250 24.250 24.875 25.875 23.000 21.375 23.000 23.750 24.875
## [21] 22.625 21.125 23.000 23.375 21.125 18.500 26.375
```

```
# 그렇다면 d에서 피험자의 6년 동안 치아 크기의 평균은?
apply(d[, 1:4], 2, mean) # MARGIN = 2 (col operation)
```

```
##        8       10       12       14
## 22.18519 23.16667 24.64815 26.09259
```

```
# 피험자 별 평균을 남녀로 구분하려면?
m <- apply(d[, 1:4], 1, mean)
tapply(m, d$Sex, mean)
```

```
##        F        M
## 22.64773 24.96875
```

```r
# 평균과 표준편차를 한꺼번에 구하려면?
meanAndSD <- function(x) c(mean(x), sd(x))
tapply(m, d$Sex, meanAndSD)
```

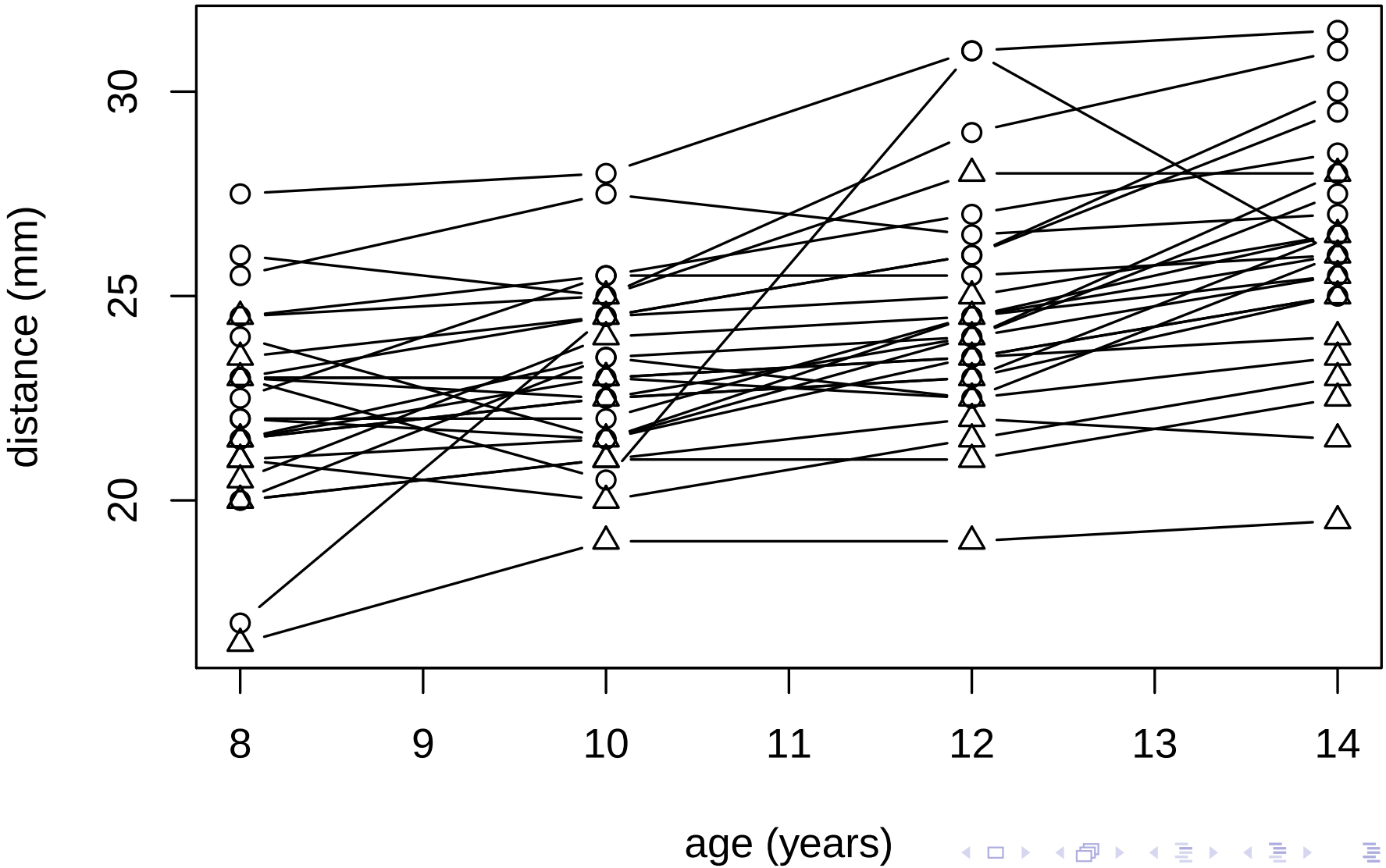
```
## $F
## [1] 22.647727  2.104918
##
## $M
## [1] 24.968750  1.828877
```

```r
# 피험자별 평균값 벡터 m을 d에 붙여 넣고 이름을 붙일 수 있다.
d <- cbind(d, m)
colnames(d)[6] <- "mean"
head(d)
```

```
##      8   10   12   14 Sex   mean
## 1 26.0 25.0 29.0 31.0   M 27.750
## 2 21.5 22.5 23.0 26.5   M 23.375
## 3 23.0 22.5 24.0 27.5   M 24.250
## 4 25.5 27.5 26.5 27.0   M 26.625
## 5 20.0 23.5 22.5 26.0   M 23.000
## 6 24.5 25.5 27.0 28.5   M 26.375
```

```r
matplot(4:7 * 2, d[, 1:4] %>% t,
        type = "b", lty = 1, pch = 1, col = "black",
        xlab = "age (years)",
        ylab = "distance (mm)")

matplot(4:7 * 2, d[, 1:4] %>% t,
        type = "b", lty = 1,
        pch = (d$Sex == "M") %>% ifelse(1, 2),
        col = "black",
        xlab = "age (years)",
        ylab = "distance (mm)")
```


distance (mm)
20
25
30
8
9
10
11
12
13
14
age (years)

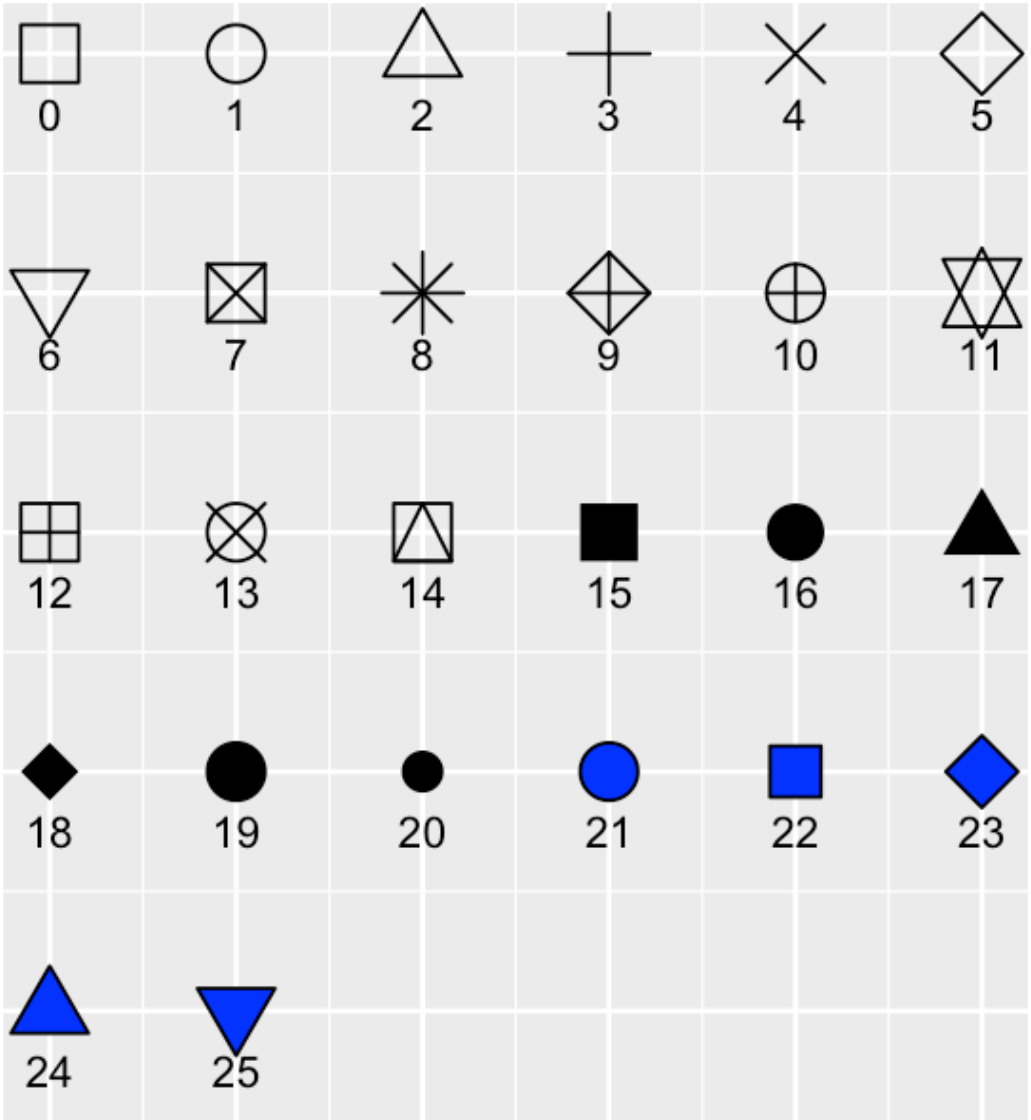


Figure: First 26 Default R Symbols Used in pch = ?

```
spread(Orthodont, key = age, value = distance)
```

```
## Error in df[representative, , drop = FALSE]: incorrect number of dimensions
```

```
# why was the function spread() not working on the Orthodont dataset?
ortho <- Orthodont[, -4]
ortho %>% spread(age, distance) %>% head
```

```
##   Subject    8   10   12   14
## 1     M16 22.0 21.5 23.5 25.0
## 2     M05 20.0 23.5 22.5 26.0
## 3     M02 21.5 22.5 23.0 26.5
## 4     M11 23.0 23.0 23.5 25.0
## 5     M07 22.0 22.0 24.5 26.5
## 6     M08 24.0 21.5 24.5 25.5
```

```
#
library(tibble)
ortho <- Orthodont %>% as.tibble
ortho %>% spread(key = age, value = distance)
```

```
## # A tibble: 27 x 6
##    Subject Sex     `8`   `10`   `12`   `14`
##    <ord>   <fct> <dbl> <dbl> <dbl> <dbl>
##  1 M16     Male   22    21.5  23.5  25
##  2 M05     Male   20    23.5  22.5  26
```