Chapter 3. R Basics

if () {} else {}

which.min 최솟값지는 인덱스 반환

ifelse(if 조건, if일 때 반환값, else일 때 반환값)

data.frame(= ifelse()) 가능

is.na() NA 체크 -> sum(is.na()) NA 개수 (ifelse(is.na(), ,) 가능)

any() 1개 이상 TRUE면 TRUE

all() 모두 TRUE면 TRUE

함수 이름 <- function(x) {} (argument에 default 값 지정 가능)

identical(a, b) TRUE, FALSE 반환

sum() 합, prod() 곱

for (n in 1:m) {}

plot(x, y)

sqrt()

sapply(함수를 적용할 (vector, matrix, data frame, list), 함수)

비슷한 함수: apply lapply tapply mapply vapply replicate

Chapter 4. The tidyverse Package

mutate(data = , column =) 열을 추가하는 함수

filter(data = , 조건) 부분 집합화

select(data =, column1, column2, ...) 열을 선택하는 함수

%>% pipe 연산자, 함수의 결과를 전송하는 역할

log2(), log10, log(base =)

%in% 포함 연산자

sort(a, decreasing = , na.last =)

rank(a, na.last = T, F, 'keep', NA) 순위의 인덱스를 나타내기

order() 정렬했을 때의 인덱스, 데이터프레임에서 사용 가능

summarize() 통계적 요약

pull() 데이터에 담긴 값에 접근

group_by() 데이터를 분리 -> summarize()와 연계 용이

arrange(), arrange(desc()), Nested sorting도 가능 (argument 여러 개)

top_n(data, n, variable) filter 역할, 정렬된 것은 아님

tibble() data.frame처럼 사용 가능

.$a pull로 접근하지 못하는 데이터에 접근할 때

purrr 패키지 map() -> sapply()와 유사, 항상 list 반환

map_dbl() vector 반환

map_df() tibble 반환

case_when(case ~ return, case ~ return, TRUE ~ return)

between(x, a, b) == (x >= a & x <= b)

Chapter 5. Importing Data

system.file(filename, package =), directory 반환

file.path(dir, filename)

file.copy(path, filename)

read_csv(filename)

list.files(path =)

setwd(), getwd()

```
library(readr)
```

- The following functions are available to read-in spreadsheets:

Function	Format	Typical suffix
read_table	white space separated values	txt
read_csv	comma separated values	csv
read_csv2	semicolon separated values	csv
read_tsv	tab delimited separated values	tsv
read_delim	general text file format, must define delimiter	txt

filename, fullpath 모두 가능

read_lines(filename, n_max =)

download.file(url, filename)

R-base: 인자로 stringAsFactors =

read.table()

read.csv()

read.delim()

scan(file.path(path, filename), sep = , what =)

```
library(readxl)
```

- The package provides functions to read-in Microsoft Excel formats:

Function	Format	Typical suffix
read_excel	auto detect the format	xls, xlsx
read_xls	original format	xls
read_xlsx	new format	xlsx

Chapter 6. Data Visualization with ggplot2

ggplot(data) 캔버스 만들기

ggplot(data, aes(x = , y =)) 축 만들기

geom_point(size = , color =) 점 찍기

geom_label(), geom_text(aes(x = , y = , label =), nudge_x = , hjust = , check_overlap = TF)

scale_x_continuous(trans =), scale_y_continuous(trans =)

scale_x_log10(), scale_y_log10()

xlab(), ylab(), ggtitle()

geom_abline(intercept =)

scale_color_discrete(name =)

theme_economist()

qplot(x, y)

grid.arrange(qplot(), qplot(), ncol =)

Chapter 7. Visualizing Data Distributions

geom_bar(stat = , show.lengnd =)

stat_ecdf()

geom_histogram(binwidth = , color = , alpha =)

geom_density(alpha = , fill = , color =)

geom_line(stat = , adjust =)

geom_area(aes(x = , y =), data = , alpha = , fill =) // data = filter(tmp, between())

ggplot(fill =)

scale(x)

pnorm(x), qnorm(x, mean = , sd =)

quantile(x, p)

ggplot(aes(sample =))

geom_qq(dparams =)

geom_boxplot()

expand_grid(x = , y =)

ggplot(aes(x, y, fill = z))

geom_raster()

scale_fill_gradient(color = terrain.colors(10))

qplot(x, bins = , color = I('black'), geom =)

Chapter 8. Data Visualization in Practice

geom_bar(stat = , show.lengnd =)

ggplot(aes(color = , group = , col =))

facet_grid(a ~ b)

facet_wrap(~a)

facet_wrap(. ~ year, scales = 'free')

geom_text(data = , aes(x, y, label =), size =)

theme(legend.position = 'none')

ggplot(aes(log2())) ggplot(aes(log10()))

scale_x_continuous(trans = '', breaks = c())

mutate(region = reorder(region, dollars_per_day, FUN = median))

mutate(group = case_when())

mutate(group = factor(group, levels = c()))

theme(axis.text.x = element_text(angle = 90, hjust = 1))

geom_point(alpha =)

library(ggridges)

geom_density_ridges(jittered_points = TRUE, position = position_points_hitter(height = 0),
                    point_shape = '|', point_size = 3,
                    point_alpha = 1, alpha = 0.7)

intersect(a, b) 두 벡터의 교집합

..count.. 실제로 count가 아니지만 count라고 불리는 변수에 접근

geom_density(alpha = 0.2, bw = 0.75, position = 'stack')

Chapter 9. Data Visualization Principles

geom_bar(stat = , show.lengnd =)

mutate(state = reorder(state, murder_rate))

geom_bar(stat = 'identity')

geom_jitter(width = , alpha =)

geom_boxplot(coef =)

spread(a, b)

geom_text_repel()

geom_abline(lty =)

geom_vline(xintercept = , col =)

scale_fill_gradientn(color = brewer.pal(9, 'Reds'), trans = 'sqrt')

theme_minimal()

theme(panel.grid = element_blank(), legend.position = 'bottom', text = element_text(size =))

geom_tile(color =)

ggtitle(the_disease)

geom_line(mapping = aes(x, y), data = , size =)

geom_line(aes(x, y, group =), color = , show.legend = FALSE, alpha = , size =)

Chapter 10. Probability

rep(c(a, b), time = c(a, b))

replicate(B, sample(beads, 1))

table(events)

prop.table(tab)

sample(beads, B, replace = TRUE)

paste(a, b)

permutations(a, b, v =)

library(gtools)

pnorm(a, m, s)

rnorm(n, m, s)

seq(a, b, length.out =)

dnorm(a, m, s)

Chapter 11. Statistical Inference

replicate(B, {x <- sample(c(0, 1), size = N, replace = TRUE, prob = c(1-p, p)) mean(x)})

// Bernoulli

stat_qq(dparams = list(mean = , sd =))

summarize_all() // 모든 column에 대하여 연산

Chapter 13. Linear Models

stat_qq(aes(sample =))

get_slope <- function(x, y) cor(x, y) * sd(y) / sd(x)

sample_n()

ungroup()

rename()

fit <- lm(y ~ x, data =)

fit$coef

summary(fit)

geom_smooth(method = 'lm', se = TRUE/FALSE)

predict(fit, se.fit = TRUE, newdata =)

library(broom)

tidy(fit, conf.int =)

do(tidy(lm(y ~ x, data = .), conf.int =)) // skillful line

ggplot(aes(ymin = , ymax =))

geom_errorbar()

paste0("", c("", "", ...))

summarize_at(var(), funs())

which.max()

apply(., 1, func) // 1: per each row, 2: per each column

str_to_upper()

str_remove(string = , pattern =)

Chapter 14. Association is not Causation

rep(1:n, each = N)

cor(rank(x), rank(y)) cor(x, y, method = 'spearman')

xtabs(y ~ x1 + x2, data)
prop.table(2) // columnwise standardized
geom_bar(position = 'fill', stat = 'identity')
ggplot(aes(label =))
scale_y_continuous(labels = scales::percent)
geom_bar(position = 'stack')

Chapter 15. Data Wrangling

pivot_longer(wide_data, columns (pivot), names_to = , values_to =)
pivot_wider(tidy_data, names_from = , values_from =)
separate(data, name, c('year', 'variable_name'), '_')
separate(data, name, c('year', 'first_variable_name', 'second_variable_name'), fill = 'right)
separate(data, name, c('year', 'variable_name'), extra = 'merge')
unite(data, variable_name, first_variable_name, second_variable_name, sep =)
left_join(a, b, by =) // table with the same rows as the first table
slice(data, 1:6)
right_join(a, b, by =) // table with the same rows as the second table
inner_join(a, b, by =) // keep only the rows that have information in both tables
full_join(a, b, by =) // keep all the rows and fill the missing parts with NAs
semi_join(a, b, by =)
// keep the part of the first table for which we have information in the second
anti_join(a, b, by =)
// keep the elements of the first table for which there in no information in the second
bind_cols(a = 1:3, b = 4:6)
bind_rows(tab[1:2,], tab[3:4,])
intersect(a, b)
dplyr::intersect(a, b)
union(a, b)
dplyr::union(a, b)
setdiff(a, b)
dplyr::set(a, b) // a – b
setequal(a, b)
dplyr::setequal(a, b) // TRUE or FALSE

Chapter 16. Parsing Dates And Times

as.Date('1970-01-01') %>% as.numeric // [1] 0

library(lubridate)

month(dates, label = TRUE) day(dates) year(dates)

ymd(x) mdy(x) ydm(x) myd(x) dmy(x) dym(x)

Sys.time()

now() now('GMT')

now() %>% hour() // minute() // second()

hms(x) mdy_hms(x)

make_date(2022, 12, 03)

make_date(1980:1989)

round_date(x)

// used to round dates to nearest year, quarter, month, week, day, hour, minutes or seconds

Chapter 17. Smoothing

with(data, ksmooth(x, y, kernel = 'box', bandwidth = span))

with(data, ksmooth(x, y, kernel = 'normal', bandwidth = span))

loess(y ~ x, degree = , span = , data =) // span: proportion

geom_line(lty =)

geom_smooth(span =) // span: proportion

diff(x) range(x) + diff(range(x))

Chapter 18. Cross Validation

knn3(y ~ ., data = , k =)

predict(model, dataset, type = 'class')

confusionMatrix(pred, dataset$y)$overall[['Accuracy']]

```
M <- replicate(B, {
  X <- sample(data, N, replace = TRUE: bootstrap / FALSE: not)
  median(X)
})
quantile(M, c(0.025, 0.975))
```

```
library(purrr)
accuracy <- map_df(ks, function(k) {
  fit <- knn3(y ~ ., data = dataset$train, k = k)

  y_hat <- predict(fit, train set, type = 'class')
  cm_train <- confusionMatrix(y_hat, train$y)
  train_error <- cm_train$overall['Accuracy']

  y_hat <- predict(fit, test set, type = 'class')
  cm_test <- confusionMatrix(y_hat, test$y)
  test_error <- cm_test$overall['Accuracy']

  tibble(train = train_error, test = test_error)
})
```