2022년 한국생명공학연구원 연구데이터 분석과정 R

합성생물학전문연구단 김하성

2022-05-18

Contents

Chapter 1

Introduction

1.1 강의 개요

- 목표: 생물 데이터 분석을 위한 R 사용법과 (Rstudio, Tidyverse, Bioconductor 포함) 프로그래밍 기술을 습득함
- 장소: 코빅 3층 전산교육장(1304호)
- 강사: 한국생명공학연구원 합성생물학전문연구단 김하성
- 연락처: 042-860-4372, haseong@kribb.re.kr
- 강의자료: https://greendaygh.github.io/kribb2022R/

1.2 강의 계획

1. R 사용법 및 데이터 분석 기초 5.19(목), 5.26(목)
2. R/Tidyverse 데이터 분석 중급 6.9(목), 6.16(목)
3. R/Tidyverse 활용 데이터 가시화 7.7(목), 7.14(목)
4. R/Bioconductor 활용한 바이오데이터 분석 기초 8.4(목), 8.11(목)
5. R/Bioconductor 활용한 NGS 데이터 분석 기초 9.1(목), 9.15(목)
6. R/Bioconductor 활용한 NGS 데이터 분석 및 Workflow 10.6(목), 10.13(목)

1.3 참고 자료

- R 홈페이지
- Rstudio 홈페이지
- Bioconductor
- R 기본 문서들
- R ebooks
- Cheat Sheets

- RStudio Webinars
- Shiny
- Hadley github
- R for Data Science
- Using R for Introductory Statistics by John Verzani
    - Free version of 1st Edition
    - Second edition
- Bioinformatics Data Skills by Vince Buffalo
- Introductory Statistics with R by Dalgaard
- 일반통계학 (영지문화사, 김우철 외)

Chapter 2

R/Rstudio basics

2.1 What is R / Rstudio

R은 통계나 생물통계, 유전학을 연구하는 사람들 사이에서 널리 사용되는 오픈소스 프로그래밍 언어 입니다. Bell Lab에서 개발한 S 언어에서 유래했으며 많은 라이브러리 (다른 사람들이 만들어 놓은 코드)가 있어서 쉽게 가져다 사용할 수 있습니다. R은 복잡한 수식이나 통계 알고리즘을 간단히 구현하고 사용할 수 있으며 C, C++, Python 등 다른 언어들과의 병행 사용도 가능합니다. R은 IEEE에서 조사하는 Top programming languages에서 2018년 7위, 2019년 5위, 2020년 6위, 2021년 7위로 꾸준히 높은 사용자를 확보하며 빅데이터, AI 시대의 주요한 프로그래밍 언어로 사용되고 있습니다.

R은 데이터를 통계분석에 널리 사용되는데 이는 데이터를 눈으로 확인하기 위한 visualization 이나 벡터 연산 등의 강력한 기능 때문에 점점 더 많은 사람들이 사용하고 있습니다. 기존에는 속도나 확장성이 다른 언어들에 비해 단점으로 지적되었으나 R 언어의 계속적인 개발과 업데이트로 이러한 단점들이 빠르게 보완되고 있습니다. R 사용을 위해서는 R 언어의 코어 프로그램을 먼저 설치하고 그 다음 R 언어용

Rank	Language	Type	Score
1	Python		100.0
2	Java		95.4
3	C		94.7
4	C++		92.4
5	JavaScript		88.1
6	C#		82.4
7	R		81.7
8	Go		77.7
9	HTML		75.4
10	Swift		70.4

Figure 2.1: https://spectrum.ieee.org/top-programming-languages/

IDE(Integrated Development Environment)인 RStudio 설치가 필요합니다.

Rstudio는 R 언어를 위한 오픈소스 기반 통합개발환경(IDE)으로 R 프로그래밍을 위한 편리한 기능들을 제공해 줍니다. R언어가 주목을 받고 두터운 사용자 층을 확보할 수 있게된 핵심 동력이 Rstudio 입니다. 자체적으로 최고수준의 오픈소스 개발팀이 있으며 `tidyverse`, ‘,shiny‘ 등의 데이터 분석 관련 주요 패키지를 개발하였고 정기적으로 conference 개최를 하면서 기술 보급의 핵심 역할을 하고 있습니다.

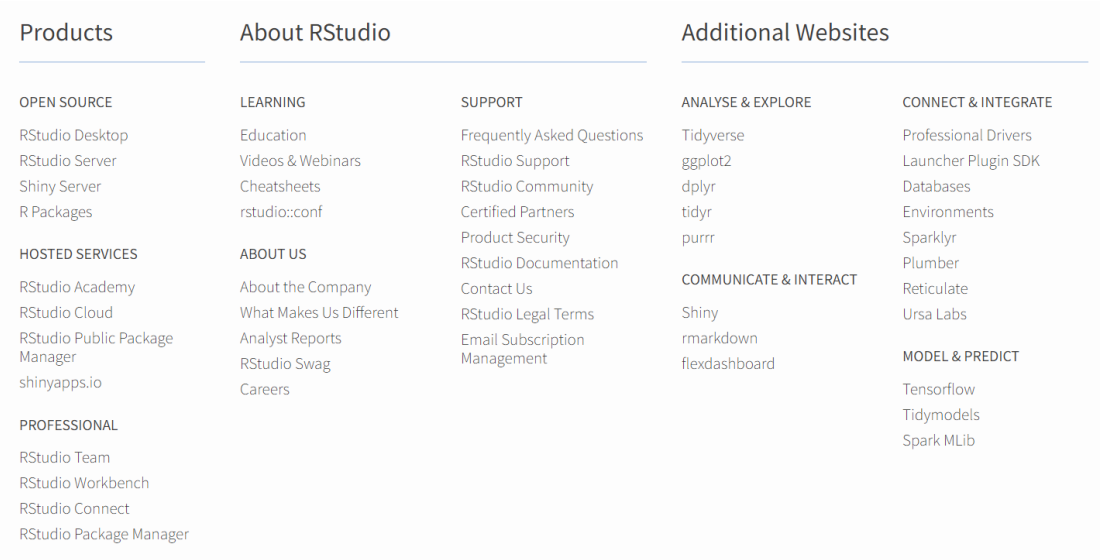


Figure 2.2: https://www.rstudio.com/

2.2 R / Rstudio Installation

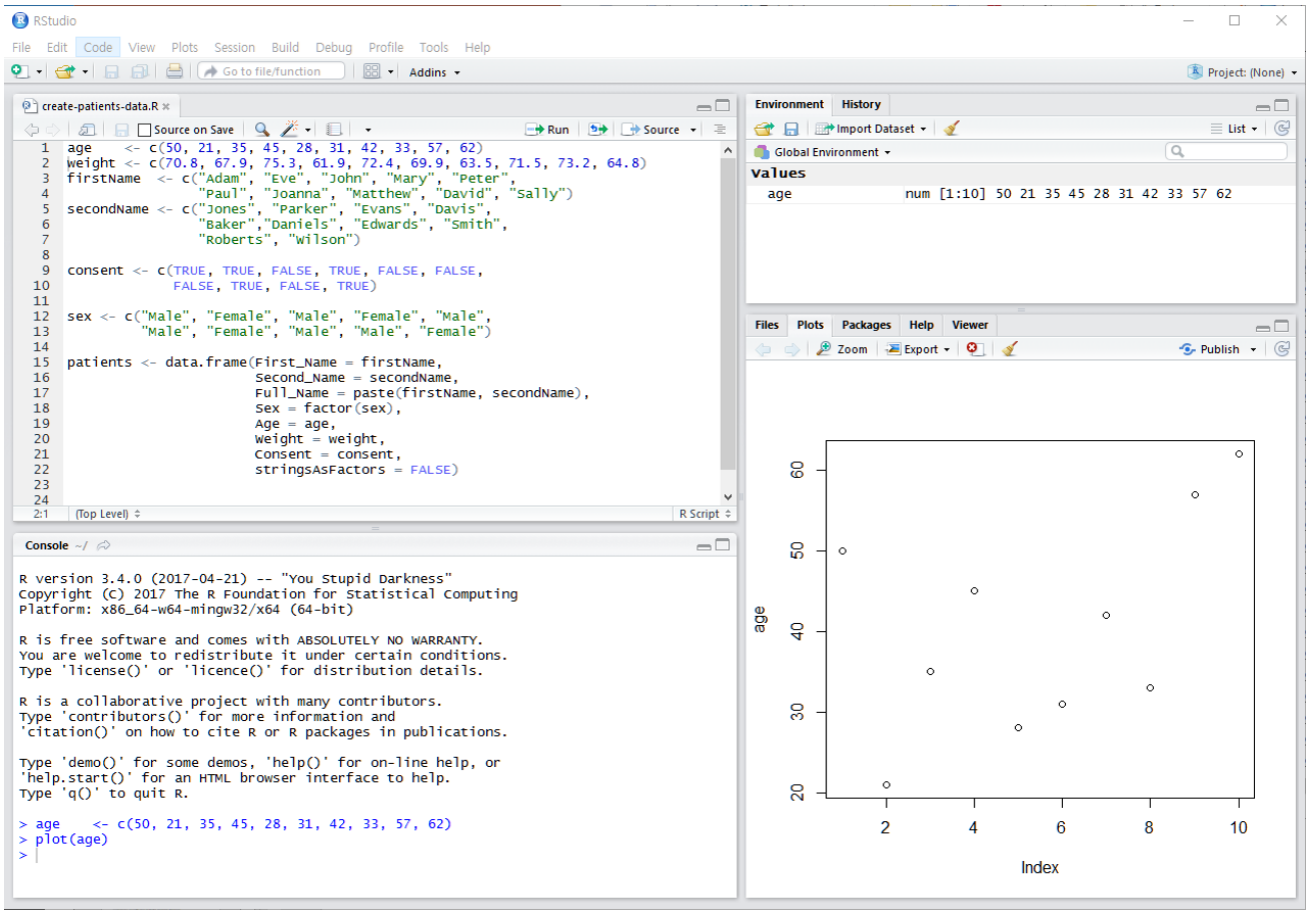
2.2.1 R 설치

- R 사이트에 접속 후 (https://www.r-project.org/) 좌측 메뉴 상단에 위치한 CRAN 클릭.
- 미러 사이트 목록에서 Korea의 아무 사이트나 들어감
- Download R for Windows를 클릭 후 base 링크 들어가서
- Download R x.x.x for Windows 링크 클릭으로 실행 프로그램 다운로드
- 로컬 컴퓨터에 Download 된 R-x.x.x-win.exe 를 실행 (2022.5 현재 R 버전은 4.2.0).
- 설치 프로그램의 지시에 따라 R 언어 소프트웨어 설치를 완료

2.2.2 Rstudio 설치

- 사이트에 접속 (https://www.rstudio.com/), 상단의 Products > RStudio 클릭
- RStudio Desktop 선택
- Download RStudio Desktop 클릭
- RStudio Desktop Free 버전의 Download를 선택하고
- Download RStudio for Windows 클릭, 다운로드
- 로컬 컴퓨터에 다운로드된 RStudio-x.x.x.exe 실행 (2022.5 현재 RStudio Desktop 2022.02.2+485)
- 설치 가이드에 따라 설치 완료

2.3 Rstudio interface



- 기본 화면에서 좌측 상단의 공간은 코드편집창, 좌측 하단은 콘솔창
- 각 위치를 기호에 따라서 바꿀 수 있음 (View -> Pane)

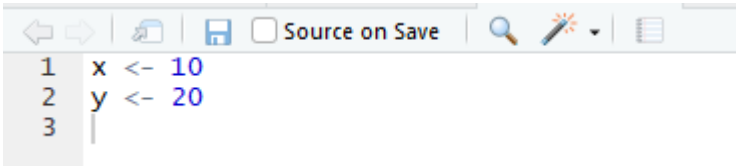
2.3.1 Keyboard shortcuts

- 참고사이트
  - https://support.rstudio.com/hc/en-us/articles/200711853-Keyboard-Shortcuts
  - Tools -> Keyboard shortcut Quick Reference (`Alt + Shift + K`)
- 코드편집창 이동 (`Ctrl + 1`) 콘솔창 이동(`Ctrl + 2`)
- 한 줄 실행 (`Ctrl + Enter`)
- 저장 (`Ctrl + S`)
- 주석처리 (`Ctrl + Shift + C`)
  - 또는 #으로 시작하는 라인

- 텝 이동 (`Ctrl + F11`, `Ctrl + F12`)
- 코드편집창 확대 (`Shift + Ctrl + 1`) 콘솔창 확대 (`Shift + Ctrl + 2`)
- 컬럼 편집 (`Alt +`)

2.3.2 Exercise

1. 코드편집창에서 다음을 입력/실행하고 단축키를 사용하여 주석을 넣으시오



- 단축키 `Ctrl + enter`로 코드 실행
- 단축키 `Ctrl + 2`로 커서 콘솔창으로 이동
- `x`값 `x+y`값 확인
- 단축키 `Ctrl + 1`로 코드편집창 이동
- 단축키 `Ctrl + Shift + C` 사용

```
# x <- 10
# y <- 20
```

2.3.3 Set a project

프로젝트를 만들어서 사용할 경우 파일이나 디렉토리, 내용 등을 쉽게 구분하여 사용 가능합니다. 아래와 같이 임의의 디렉토리에 `kribbR` 이라는 디렉토리를 생성하고 `lecture1` 프로젝트를 만듭니다.

> File > New Project > New Directory > New Project > “kribbR” > Create Project

시작할 때는 해당 디렉토리의 `xxx.Rproj` 파일을 클릭합니다. Rstudio 오른쪽 상단 프로젝트 선택을 통해서 빠르게 다른 프로젝트의 작업공간으로 이동할 수 있습니다.

2.3.4 Terminology

- Session: R 언어 실행 환경
- Console: 명령어 입력하는 창
- Code: R 프로그래밍 변수/제어문 모음
- Object types:
  - vector: 값들의 모임 combine function `c()` EX: c(6, 11, 13, 31, 90, 92)
  - matrix: 2D 형태 값들의 모임
  - array: 1D, 2D, 3D, ⋯ 형태 값들의 모임
  - factor: 범주형 데이터
  - data frame: 2D 형태 값들의 모임 (다른 타입 값 가능)

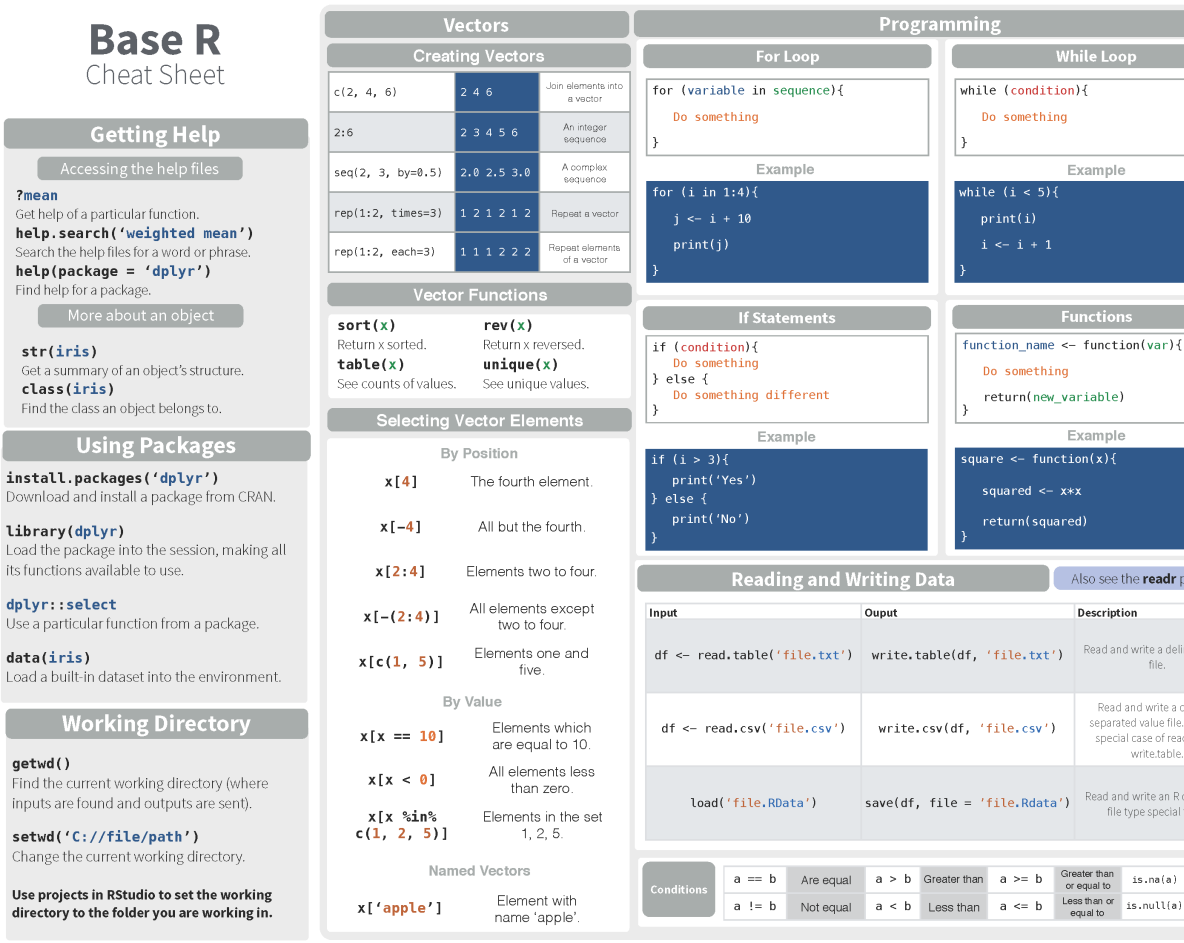
    - list:
    - function: 특정 기능 수행, [함수이름, 입력값 (arguments), 출력값 (return)] 으로 구성
- Data (value) types:
    - Integers
    - doubles/numerics
    - logicals
    - characters.
- Conditionals (조건, 제어):
    - `if`, `==`, `&` (AND), | (OR) Ex: `(2 + 1 == 3) & (2 + 1 == 4)`
    - `for`, `while`: 반복 수

2.4 Getting help

R은 방대한 양의 도움말 데이터를 제공하며 다음과 같은 명령어로 특정 함수의 도움말과 예제를 찾아볼 수 있습니다.

```
help("mean")
?mean
example("mean")
help.search("mean")
??mean
help(package="MASS")
```

또한 https://www.rstudio.com/resources/cheatsheets/ 에서는 다양한 R언어의 기능을 한 눈에 알아볼 수 있게 만든 cheatsheet 형태의 문서를 참고할 수 있습니다.



Base R
Cheat Sheet

Getting Help

Accessing the help files

?mean
Get help of a particular function.
help.search('weighted mean')
Search the help files for a word or phrase.
help(package = 'dplyr')
Find help for a package.

More about an object

str(iris)
Get a summary of an object's structure.
class(iris)
Find the class an object belongs to.

Using Packages

install.packages('dplyr')
Download and install a package from CRAN.

library(dplyr)
Load the package into the session, making all its functions available to use.

dplyr::select
Use a particular function from a package.

data(iris)
Load a built-in dataset into the environment.

Working Directory

getwd()
Find the current working directory (where inputs are found and outputs are sent).

setwd('C://file/path')
Change the current working directory.

Use projects in RStudio to set the working directory to the folder you are working in.

Vectors

Creating Vectors

c(2, 4, 6)	2 4 6	Join elements into a vector
2:6	2 3 4 5 6	An integer sequence
seq(2, 3, by=0.5)	2.0 2.5 3.0	A complex sequence
rep(1:2, times=3)	1 2 1 2 1 2	Repeat a vector
rep(1:2, each=3)	1 1 1 2 2 2	Repeat elements of a vector

Vector Functions

sort(x)
Return x sorted.
rev(x)
Return x reversed.
table(x)
See counts of values.
unique(x)
See unique values.

Selecting Vector Elements

By Position

x[4]	The fourth element.
x[-4]	All but the fourth.
x[2:4]	Elements two to four.
x[-(2:4)]	All elements except two to four.
x[c(1, 5)]	Elements one and five.

By Value

x[x == 10]	Elements which are equal to 10.
x[x < 0]	All elements less than zero.
x[x %in% c(1, 2, 5)]	Elements in the set 1, 2, 5.

Named Vectors

x['apple']	Element with name 'apple'.

Programming

For Loop

```
for (variable in sequence){
   Do something
}
```

Example

```
for (i in 1:4){
   j <- i + 10
   print(j)
}
```

While Loop

```
while (condition){
   Do something
}
```

Example

```
while (i < 5){
   print(i)
   i <- i + 1
}
```

If Statements

```
if (condition){
   Do something
} else {
   Do something different
}
```

Example

```
if (i > 3){
   print('Yes')
} else {
   print('No')
}
```

Functions

```
function_name <- function(var){
   Do something
   return(new_variable)
}
```

Example

```
square <- function(x){
   squared <- x*x
   return(squared)
}
```

Reading and Writing Data

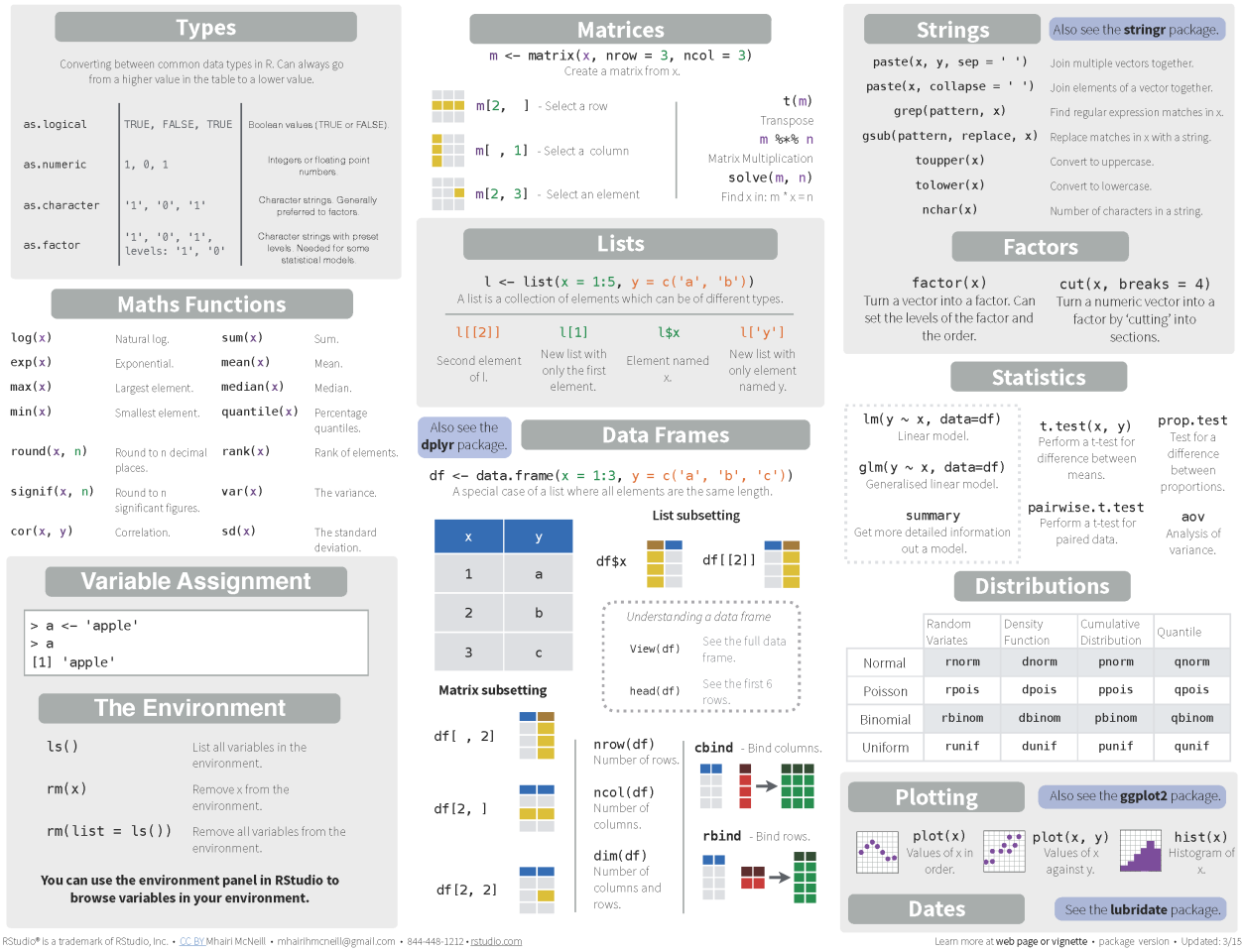
Also see the **readr** p

Input	Ouput	Description
df <- read.table('file.txt')	write.table(df, 'file.txt')	Read and write a deli file.
df <- read.csv('file.csv')	write.csv(df, 'file.csv')	Read and write a c separated value file. special case of rea write.table.
load('file.RData')	save(df, file = 'file.Rdata')	Read and write an R file type special

Conditions

a == b	Are equal	a > b	Greater than	a >= b	Greater than or equal to	is.na(a)	
a != b	Not equal	a < b	Less than	a <= b	Less than or equal to	is.null(a)	

Learn more at web page or vignette • package version

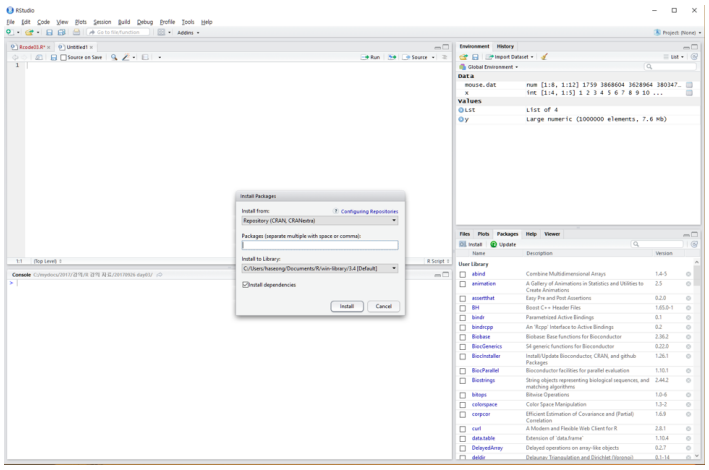


2.5 R packages and Dataset

R 패키지는 함수와 데이터셋의 묶음으로 다른 사람들이 만들어 놓은 코드나 기능을 가져와서 사용하므로써 코드 작성의 수고로움을 줄이고 편리하고 검증된 함수(기능)를 빠르게 도입하여 사용할 수 있다는 장점이 있습니다. 예를 들어 `sd()` 함수는 `stats` package에서 제공하는 함수로써 표준편차 계산을 위한 별도의 함수를 만들어서 사용할 필요가 없이 바로 (stats 패키지는 R 기본 패키지로) 별도 설치 없이 바로 사용 가능합니다.

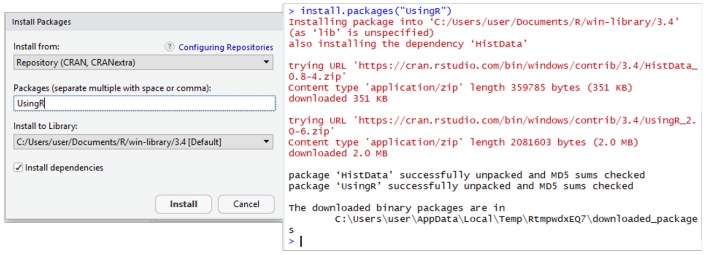
패키지를 구할 수 있는 가장 대표적인 사이트는 (저장소는) The Comprehensive R Archive Network (CRAN) http://cran.r-project.org/web/views/ 이며 우리가 사용할 Bioconductor - http://www.bioconductor.org/packages/release/bioc

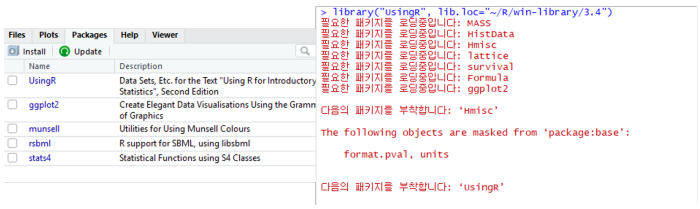
/ 저장소도 체계적으로 관리되고 많은 사람들이 활용하고 있는 사이트 중 하나입니다.



- UsingR package installation

Packages → Install






```
install.packages("UsingR")
```

- UsingR package loading

```
library(UsingR)
help(package="UsingR")
```

일반적으로 패키지 안에 관련된 데이터도 같이 저장되어 있으며 `data()` 함수를 이용해서 패키지 데이터를 사용자 작업공간에 복사해서 사용 가능합니다.

```
head(rivers)
length(rivers)
class(rivers)
```

```
data(rivers)
data(package="UsingR")
library(HistData)
head(Cavendish)
str(Cavendish)
head(Cavendish$density2)
```

2.6 R programming

2.6.1 Console calculator

```
2 + 2
((2 - 1)^2 + (1 - 3)^2 )^(1/2)
2 + 2; 2 - 2
```

2.6.2 Exercise

다음 공식들을 계산하는 R 코드를 작성하시오

$$\sqrt{(4+3)(2+1)}$$

$$2^3 + 3^2$$

$$\frac{0.25 - 0.2}{\sqrt{0.2(1-0.2)/100}}$$

2.6.3 Variables and values

- 프로그래밍 언어의 공통적 개념 , , , ,
- Assignment operator (<- OR =)
  - Valid object name <- value
  - 단축키: `Alt + -` (the minus sign)

```
x <- 2
y <- x^2 - 2*x + 1
y
x <- "two"
some_data <- 9.8
```

- 내장 변수 Built-in variables

```
pi
```

- 변수이름 작명법
  - 문자, 숫자, “_”, “.” 등으로 구성
  - 대소문자 구분
  - 가독성, 의미있는 변수 이름
  - 길이 제한 없음

```
i_use_snake_case <- 1
otherPeopleUseCamelCase <- 2
some.people.use.periods <- 3
And_aFew.People_RENOUNCEconvention <- 4
```

- 자동 완성 기능 (Tab completion) in RStudio
- 이전 명령은 콘솔에서 위 아래 화살표

2.6.4 Exercise

1. 변수 x에 1, 3, 5, 7, 9를, 변수 y에 2, 4, 6, 8, 10을 저장하는 코드를 작성하시오
2. 앞서 변수 x와 y를 더한 값을 z에 저장하는 코드를 작성하시오
3. 변수 x에 “hello world!” 를 저장하고 x의 값을 출력하는 코드를 작성하시오

2.6.5 Functions

함수(Function)란 사용자가 원하는 기능을 수행하는 코드의 모음으로서 반복적으로 쉽게 사용할 수 있도록 만들어 놓은 코드 입니다. 특정 데이터를 입력으로 받아 원하는 기능을 수행한 후 결과 데이터를 반환하는 구조를 가집니다. 함수는 일반적으로 다음과 같은 포멧으로 구현할 수 있습니다.

```
my_function_name <- function(parameter1, parameter2, ... ){
  ##any statements
  return(object)
}
```

예를 들어 다음과 같은 my_sine 함수를 만들 수 있으며 parameter (매개변수)는 x이고 y는 반환값을 저장하는 지역변수 입니다.

```
my_sine <- function(x){
    y <- sin(x)
    return(y)
}
```

- 내장 함수 (Built-in functions)

```
x <- pi
sin(x)
```

```
sqrt(x)
log(x)
log(x, 10)
x <- c(10, 20, 30)
x + x
mean(x)
sum(x)/length(x)
```

2.6.6 Exercise

1. 변수 x에 1, 3, 5, 7, 9를, 변수 y에 2, 4, 6, 8, 10을 저장하는 코드를 작성하시오
2. x와 y를 더한 값을 z에 저장하는 코드를 작성하시오
3. mysum 이라는 이름의 함수를 작성하되 두 변수를 입력으로 받아 더한 후 결과를 반환하는 코드를 작성하시오
4. mymean 이라는 이름의 함수를 작성하되 두 변수를 입력으로 받아 평균을 구한 후 결과를 반환하는 코드를 작성하시오