

最終更新日：2025年9月29日

Rの基本的な使い方 (OMU学生向け)

作成者：武藤 拓之

※この資料の二次配布はしないようにしてください。

統計ソフト「R」と統合開発環境「RStudio」

■ R

- 統計解析に特化したプログラミング言語およびソフトウェア。
- 心理学だけでなく様々な分野・業界で広く利用されている。
→ ネット上に資料も豊富
- オープンソース・フリーウェアであり開発が盛んに行われている。
- 美しいグラフが描ける。

■ RStudio

- Rを使いやすくするための統合開発環境。



Tips:

R以外でよく使われる統計ソフトおよびプログラミング言語としては、
エクセル、SPSS、SAS、Statistica、Python、Julia等があるが、どれも一長一短。

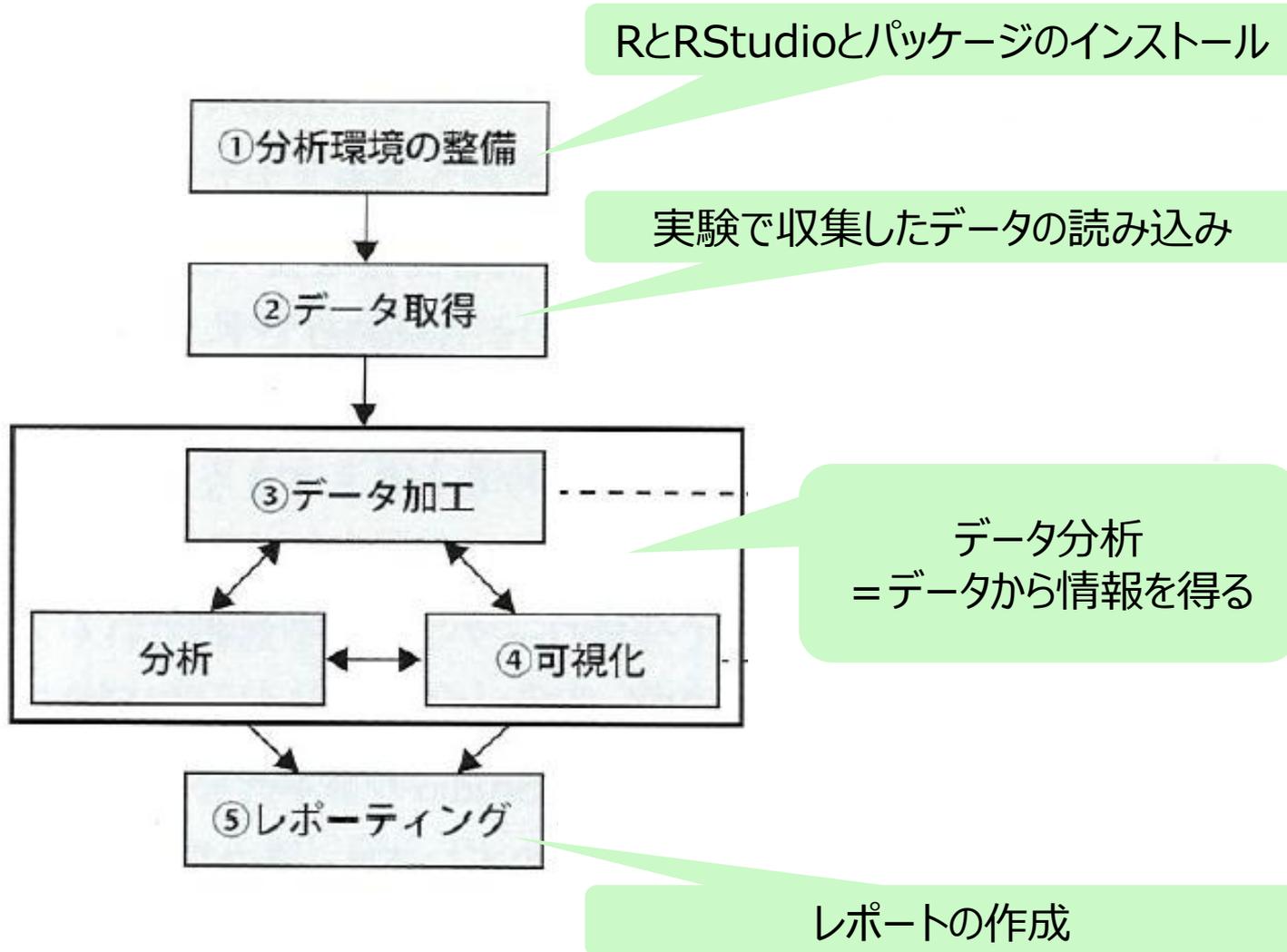
なぜRを使うのか（エクセルじゃだめ？）

■ Rを使うメリット（エクセルやSPSS等との比較）

- データの前処理や分析を効率よく実行するための関数を利用できる。
→ 有志による便利な関数の詰め合わせ（パッケージ）もある。
- 綺麗なグラフや情報の多いグラフを簡単かつ自由に描画できる。
- 理論上はどんな分析でも実行できる（最新の手法も！）
- 分析スクリプト（＝分析の手順）を完璧に残すことができる。
→ 分析の**再生可能性** (reproducibility) を担保できる。
i.e., 同じデータを同じ方法で分析したら同じ結果になる。
- 誰でも無料で利用できる & 透明性が高い (i.e., オープンソース)
- データの読み込みからドキュメント化までの全ての工程をRだけで完結させられる。(R Markdown) ※この授業では扱わない
→ ヒューマンエラーが減る

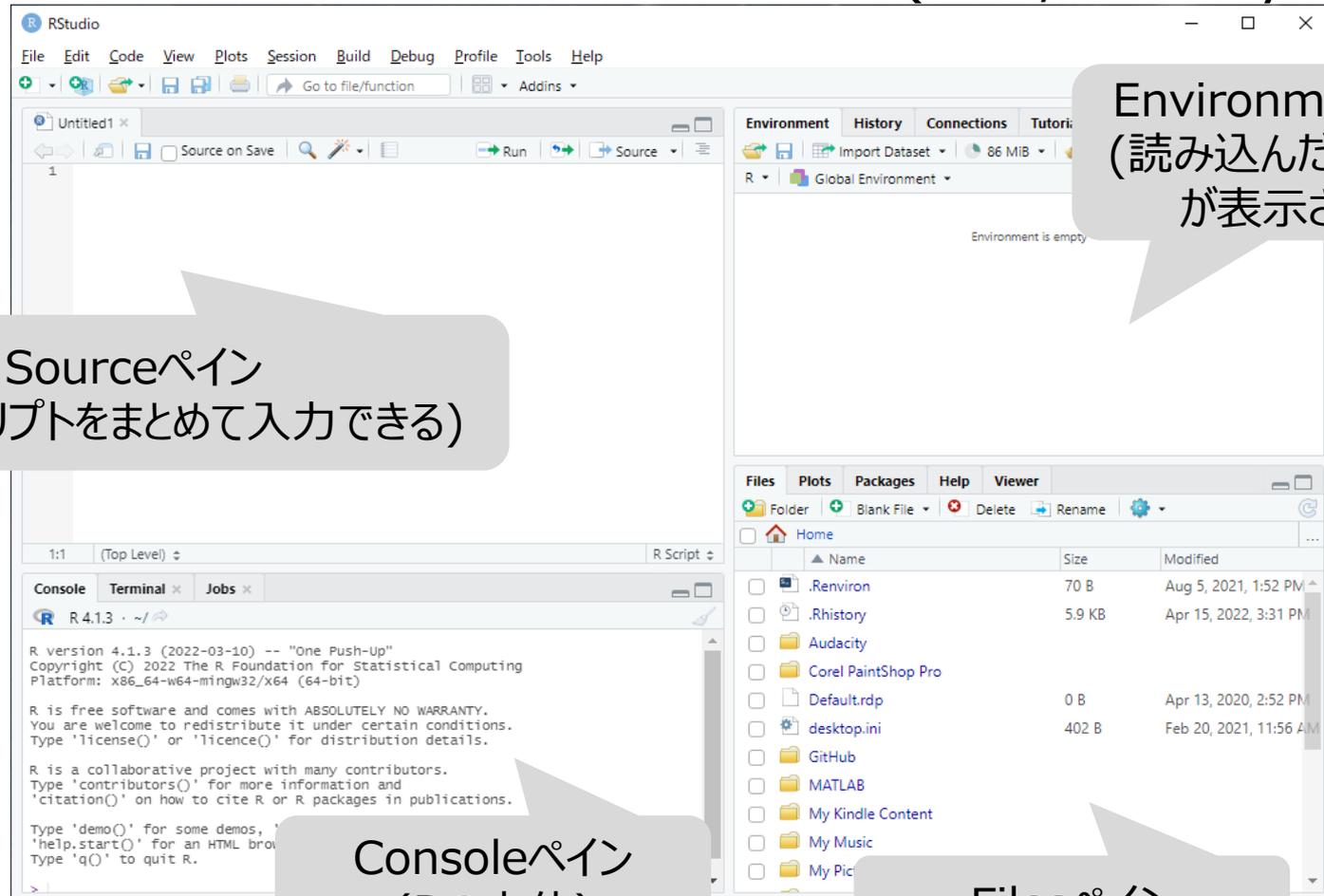
etc...

分析のワークフロー



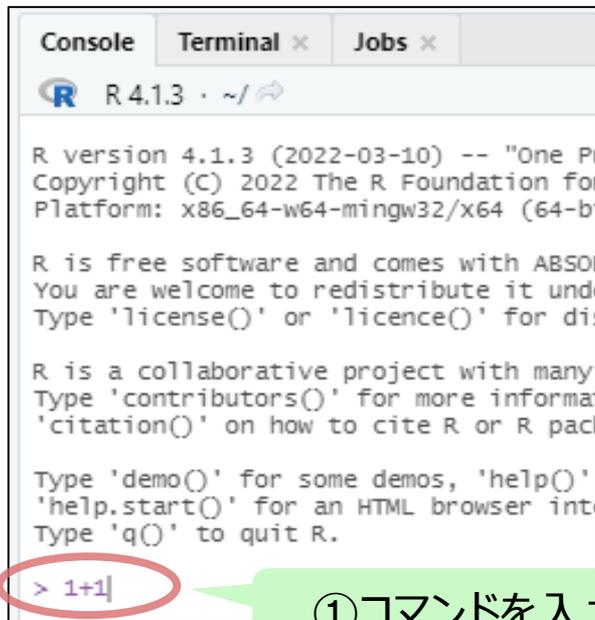
RStudioのインターフェース

- 複数の画面 (ペイン, pane) で区切られている。
- タブを切り替えることで表示されるペインもある (Plot, History等)



Consoleペイン

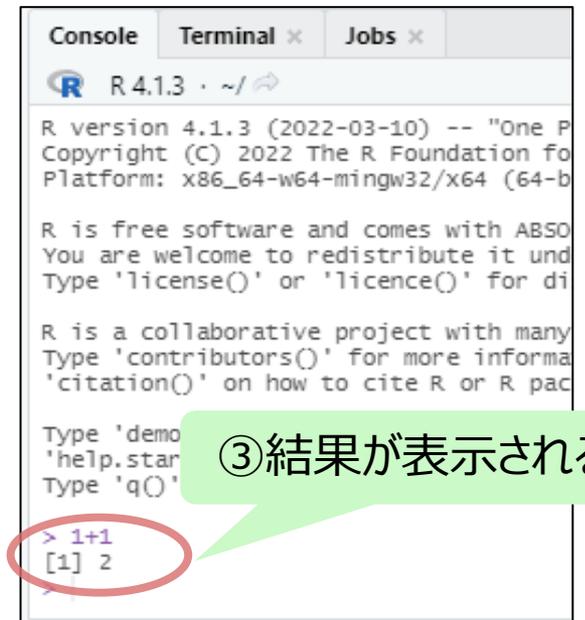
- Rの本体。ここにコマンドを入力すると実行されて結果が表示される。
- ただし、入力したコマンドと結果はどんどん流れていってしまう。
- ちょっとした計算をしたいときにのみ使う。
- 試しに「1+1」「22/7」「0.1*9」など適当な計算式を入力してみよう。



```
Console Terminal x Jobs x  
R 4.1.3 · ~/ ↻  
R version 4.1.3 (2022-03-10) -- "One P  
Copyright (C) 2022 The R Foundation fo  
Platform: x86_64-w64-mingw32/x64 (64-b  
R is free software and comes with ABSO  
You are welcome to redistribute it und  
Type 'license()' or 'licence()' for di  
R is a collaborative project with many  
Type 'contributors()' for more informat  
'citation()' on how to cite R or R pac  
Type 'demo()' for some demos, 'help()'  
'help.start()' for an HTML browser int  
Type 'q()' to quit R.  
> 1+1
```

①コマンドを入力

②Enterキー



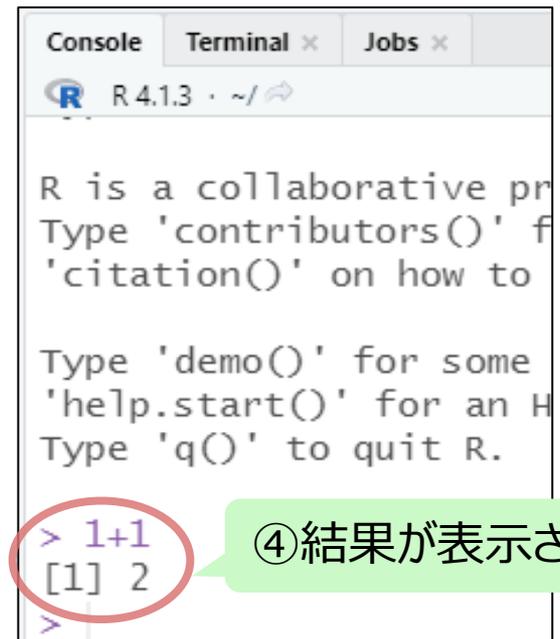
```
Console Terminal x Jobs x  
R 4.1.3 · ~/ ↻  
R version 4.1.3 (2022-03-10) -- "One P  
Copyright (C) 2022 The R Foundation fo  
Platform: x86_64-w64-mingw32/x64 (64-b  
R is free software and comes with ABSO  
You are welcome to redistribute it und  
Type 'license()' or 'licence()' for di  
R is a collaborative project with many  
Type 'contributors()' for more informat  
'citation()' on how to cite R or R pac  
Type 'demo()  
'help.start  
Type 'q()  
> 1+1  
[1] 2
```

③結果が表示される

Sourceペイン

- 分析スクリプト(実行したいコードの集まり)を記述する領域。
 - ※ Sourceペインが表示されていない場合は、メニューの「File」→「New File」→「R script」をクリックする。
- 実行したい部分を選択し、右上の「Run」を押すと Consoleペインに結果が出力される。複数行でも行の一部だけでもOK。(Windowsなら「Run」をクリックする代わりに「**Ctrl + Enter**」を押すと楽。)
- 一度スクリプトを作成すれば、何度でも同じ分析を実行できる。

③「Run」をクリック
(または、WindowsならCtrl + Enter)



スクリプトの保存

- 左上の「File」から「Save As...」を選択する。
(日本語で言う「名前を付けて保存」のこと。二回目以降は「Save」で上書き保存すればOK。)
- 文字コードを聞かれるので一番上のデフォルトを選択して「OK」
(基本的には一番上を選べばよい。)
- 自分のパソコンの適当なフォルダに「test.R」といった名前で保存

The image illustrates the process of saving an R script in three stages:

- RStudio File Menu:** The 'File' menu is open, and 'Save As...' is highlighted with a red circle.
- Choose Encoding Dialog:** A dialog box titled 'Choose Encoding' is shown. 'CP932 (System default)' is selected and circled in red. The 'OK' button is also circled in red.
- Save File Dialog:** A Windows 'Save File' dialog box is shown. The folder path 'ILAS2022work' is circled in red. The filename 'test.R' is entered in the 'ファイル名' field and circled in red. The 'Save' button is circled in red.

Green callout boxes provide instructions:

- 保存先のフォルダを選択 (Select the folder to save to)
- ファイル名の欄に「test.R」と入力 (Enter 'test.R' in the filename field)
- 「Save」をクリック (Click 'Save')

変数の代入

- 「<-」という演算子を使うと変数に値を代入できる。
例えば「a <- 1」を実行すると、「a」という名前の変数に「1」が代入される
- 作成した変数は右上のEnvironmentペインに表示される。

※「#」を使うとその後ろのコードは実行されないのでメモや見出しに便利（「コメント」と呼ぶ）

The screenshot shows the RStudio interface. The left pane contains R code with annotations. The right pane shows the Environment window with a table of variables.

Code in the Editor:

```
1 1+1
2
3 # 変数に数値や文字列を代入する
4 a <- 2
5 b <- 3
6 mojiretsu <- "ILASセミナー"
7
8 # 変数の中身の確認
9 a
10 b
11 mojiretsu
12
13 # 変数を使った計算
14 a+b
```

Environment Pane:

Values	
a	2
b	3
mojiretsu	"ILASセミナー"

Annotations:

- ① これを入力して実行 (points to lines 3-6)
- ② 変数の情報が表示される (points to the Environment pane)
- ③ 変数名を入力して実行するとコンソールに中身が表示される。 (points to lines 8-11)
- ④ 変数同士の演算も可能 (結果はコンソールに表示される) (points to lines 13-14)

baseパッケージ

- 数値や文字列などの値や変数を変換するための関数がRにはデフォルトで多数用意されている。(baseパッケージ)
 - 実践上は、baseだけでなく外部から便利なパッケージもインストールして使う。
- 和や平均, 分散, 対数, 平方根のような単純なものだけでなく, データの抽出, 乱数生成, 仮説検定, グラフ作成などさまざま。
- 関数名の後ろの括弧内に引数 (ひきすう) を入れて使う。
※2つ以上の引数をとることも多い。
- 以下のコードを入力して実行してみよう。
(#から始まるコメント部分は入力しなくてよい)

```
16 # base関数の例
17
18 sqrt(a) # 平方根を計算する関数:  $\sqrt{a}=\sqrt{2}=1.414\dots$ になるはず
19
20 vec <- c(1,2,3,4) # 値を連結してベクトルを作る関数
21
22 sum(vec) # ベクトルの要素の和を計算する関数:  $1+2+3+4=10$ になるはず
23
24 mean(vec) # ベクトルの要素の平均値を計算する関数:  $(1+2+3+4)/4=2.5$ になるはず
25
26 log(8, base = 2) # 底(base)が2の対数を計算する関数:  $\log_2(8)=3$ 
```

RStudioを閉じる

- 今回作成したスクリプトを上書き保存しておく。
 - 「File」→「Save」または上部にある  をクリック。
- 「File」→「Quit Session」でRStudioを終了する。
 - 閉じるボタンで普通に閉じてもOK
- 閉じるときにワークスペースイメージを保存するかどうかを聞かれたら、**「Don't Save」を選択すること。**
 - ワークスペースを保存すると、次回RStudioを起動したときに、今回作成した変数などが自動的に復元される。不具合の原因となることが多いので基本的には保存しないほうがよい。

データを外部から読み込む方法

- 一般的には、外部ファイル (csvなど) としてデータを用意しておき、Rスクリプト側で関数を使ってファイルのパスを指定して読み込むとよい。(クリップボードから読み込む方法や、Filesペインから手動で読み込む方法は、**再生可能性**を担保できないので非推奨。)
- Rスクリプトとデータファイルは同じフォルダに入れておくと楽。(慣れている人は、適当なサブフォルダを作ってそこにデータを入れるなどしてもOK。)
- 外部ファイルを読み込む前に、メニューの「Session」→「Set Working Directory」→「To Source File Location」をクリックして、Rスクリプトを置いてあるフォルダをワーキングディレクトリに指定しておく。(これをしないと、ファイルの場所を正しく指定できず、エラーとなる。)

csvファイル (データ) の読み込み

```
d <- read.csv("statdata_06_game.csv")
```

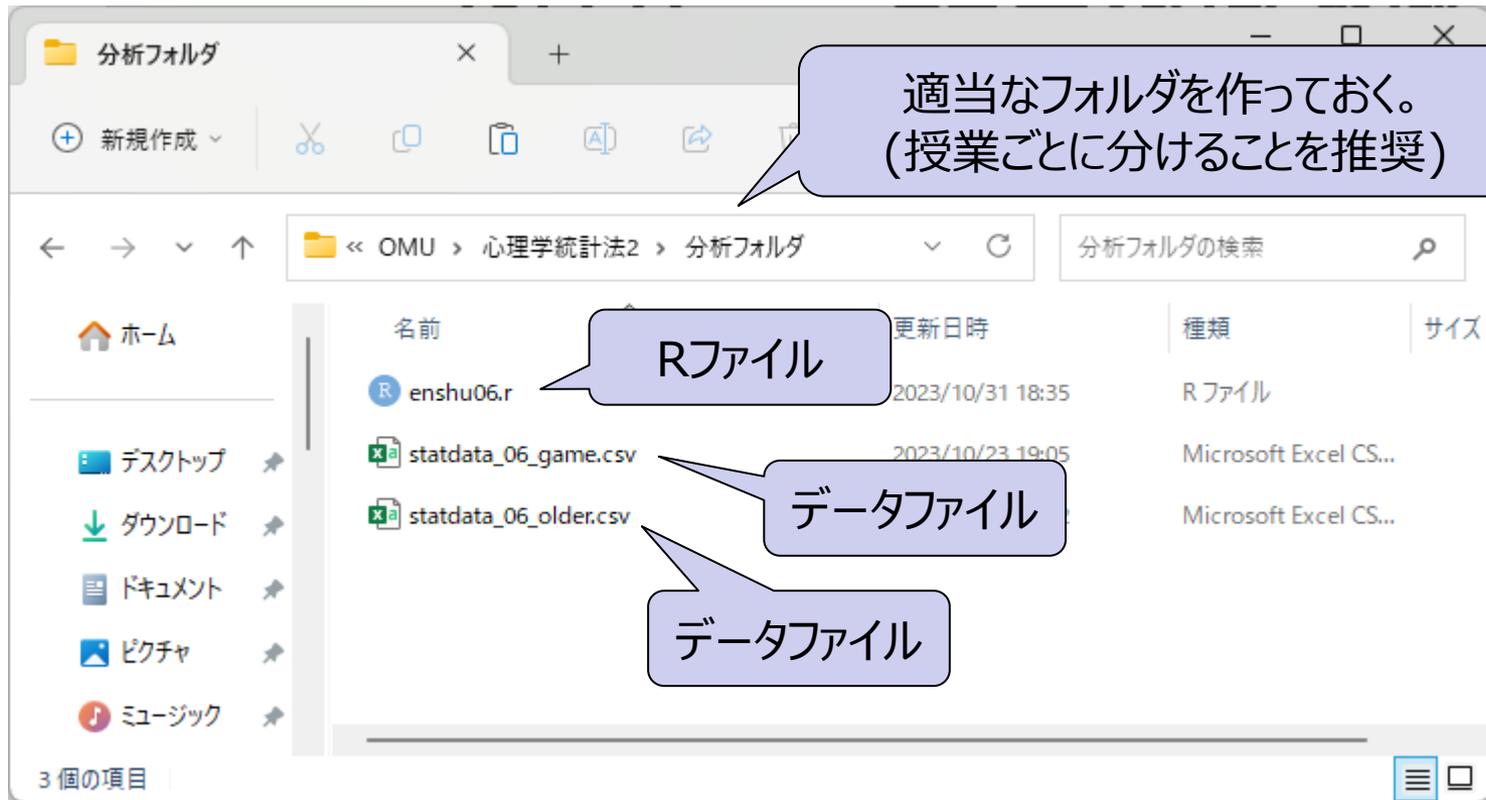
csvファイルを
データフレームとして
読み込むbase関数

引数にファイル名を入れる。
(ダブルクォーテーション""を忘れないこと。)

※ファイルを読み込む関数はread.csv以外にもありますが、初心者はこれを使うのが無難です。

データを置く場所の例

- Rファイルとデータファイルを同じディレクトリに置くと便利。
- Windowsの例 ↓



- パソコンに慣れている人は、データファイルだけを入れるフォルダを作るなどしてもOK。

外部パッケージ

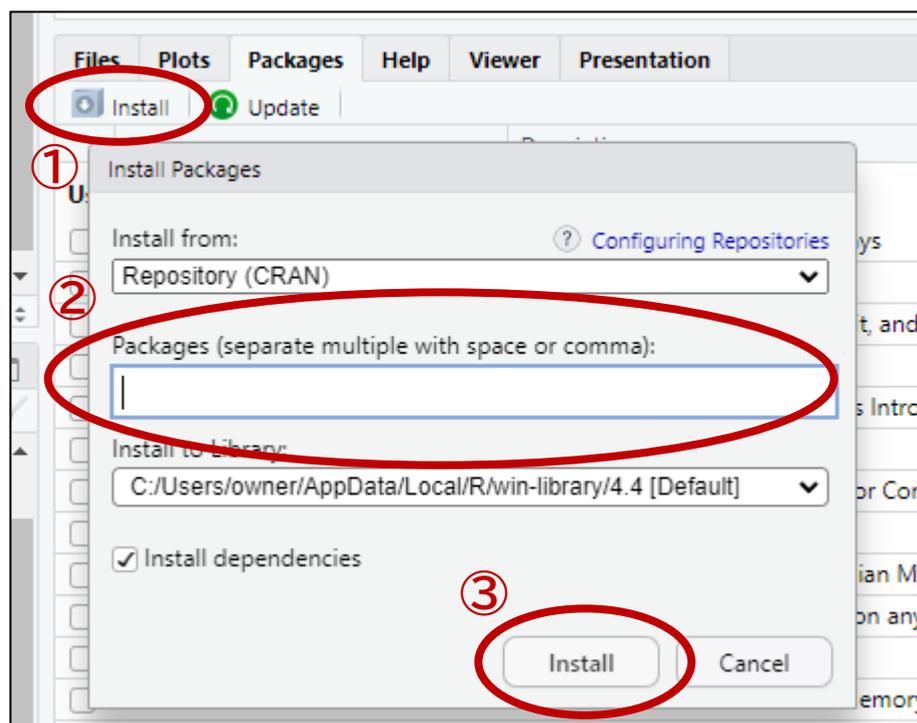
- **パッケージ** (package) = 便利な関数の寄せ集め。
- Rを使うときは、baseパッケージだけでなく、外部から便利なパッケージをインストールして使うと非常に便利。
- パッケージの例：
 - ggplot2: データを柔軟かつ綺麗に可視化するためのパッケージ
 - dplyr: データに様々な操作を加えるためのパッケージ
 - tidyr: データをtidy dataに変形するためのパッケージ
 - readr: 様々なフォーマットのデータを読み書きするためのパッケージ
などなど

※ここに挙げたパッケージは、いずれもtidyverseという包括的なパッケージに含まれているので、tidyverseをインストールすれば自動的にこれらもインストールされる。
- パッケージをインストールし、「library(パッケージ名)」を実行することで、R上でそのパッケージの関数が使えるようになる。
※毎回library()を実行しないとパッケージ内の関数には使えないので注意！

外部パッケージのインストール方法 (1)

■ CRAN (Comprehensive R Archive Network)。

- 世界中の研究者や開発者が作ったRパッケージを集めて公開している公式のリポジトリ。
- CRANに登録されているパッケージは、RStudioから簡単にインストールできる。



- ① RStudioの右下にあるPackagesタブにあるInstallボタンをクリック。
- ② パッケージ名を入力
- ③ Installをクリック

もしくは、R上で以下のコードを実行する。
`install.packages("pkgname")`
pkgnameの部分をパッケージ名に変える。

外部パッケージのインストール方法 (2)

■ GitHub

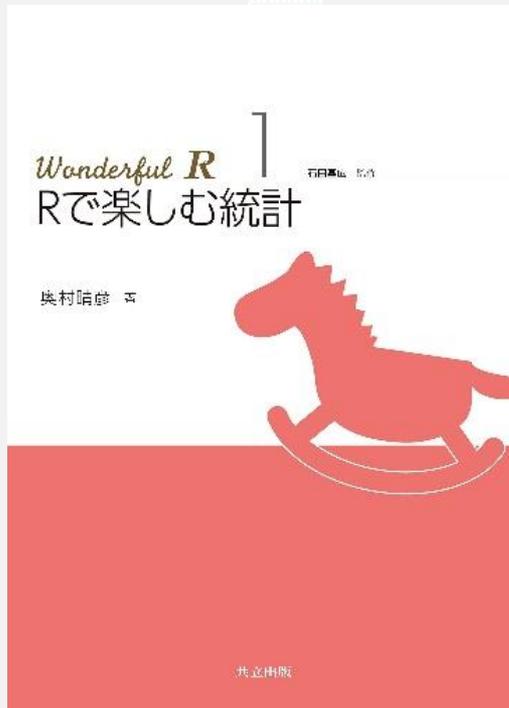
- ソフトウェアのソースコードを共有・管理するためのWebサービス。
- CRANに登録されていないパッケージや、CRAN登録前の開発版のパッケージなども公開されている。
- 武藤もいくつか自作パッケージをGitHubで公開している。
→ <https://github.com/mutopsy/>

- ① CRANからdevtoolsパッケージをインストール。
(ソフトウェア開発支援用パッケージ)
- ② R上で以下のようなコードを実行する。
`devtools::install_github("mutopsy/pwranova")`
pwranovaパッケージがGitHubからインストールされる。

自分で勉強したい人向けの書籍



Rについて1から丁寧に
勉強したい人向け



Rと統計学をまとめて
勉強したい人向け



Rの書籍を1冊読んだ後でさらに
ステップアップしたい人向け