

Rの基本的な使い方

Rの基本的な使い方

- R、Rstudioの準備
- 簡単な計算
- データの読み込み
- データの基本的な扱い方
- グラフの作成
- 追加ライブラリ・パッケージ
- その他

統計解析をRで行う際に良く使用するものを紹介

R、Rstudioの準備

Rのダウンロード(1)

- <https://www.r-project.org/>にアクセス



[\[Home\]](#)

Download

[CRAN](#)

R Project

[About R](#)

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To [download R](#), please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

- ダウンロード先を選ぶ

CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: [main page](#), [windows release](#), [windows old release](#).

If you want to host a new mirror at your institution, please have a look at the [CRAN Mirror HOWTO](#).

0-Cloud

<https://cloud.r-project.org/>

<http://cloud.r-project.org/>

Automatic redirection to servers worldwide, currently sponsored by Rstudio

Automatic redirection to servers worldwide, currently sponsored by Rstudio

Rのダウンロード(2)

• OSと種類を選ぶ

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R for Windows

Subdirectories:

[base](#)

Binaries for base distribution. This is what you want to **install R for the first time.**

[contrib](#)

Binaries of contributed CRAN packages (for R \geq 2.13.x; managed by Uwe Ligges). There is also information on [third party software](#) available for CRAN Windows services and corresponding environment and make variables.

[old contrib](#)

Binaries of contributed CRAN packages for outdated versions of R (for R $<$ 2.13.x; managed by Uwe Ligges).

[Rtools](#)

Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

• ダウンロード



CRAN
[Mirrors](#)
[What's new?](#)
[Search](#)
[CRAN Team](#)

[About R](#)

R-4.4.1 for Windows

[Download R-4.4.1 for Windows](#) (82 megabytes, 64 bit)

[README on the Windows binary distribution](#)
[New features in this version](#)

最新版でなくても問題ありません

This build requires UCRT, which is part of Windows since Windows 10 and Windows Server 2016. On older systems, UCRT has to be installed manually from [here](#).

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server.

RStudioのインストール

- <https://posit.co> にアクセス
(<https://www.rstudio.com>でも可)

①

posit PRODUCTS SOLUTIONS LEARN & SUPPORT EXPLORE MORE PRICING

DOWNLOAD RSTUDIO

| OS | Download | Size | SHA-256 |
|---------------------|--|-----------|--------------------------|
| Windows 10/11 | RSTUDIO-2024.04.2-764.EXE ± | 262.79 MB | 09E1E38A |
| macOS 12+ | RSTUDIO-2024.04.2-764.DMG ± | 664.40 MB | 060DD295 |
| Ubuntu 20/Debian 11 | RSTUDIO-2024.04.2-764-AMD64.DEB ± | 194.73 MB | 87B20155 |
| Ubuntu 22/Debian 12 | RSTUDIO-2024.04.2-764-AMD64.DEB ± | 196.64 MB | 1D0BD2F5 |
| OpenSUSE 15 | RSTUDIO-2024.04.2-764-X86_64.RPM ± | 196.89 MB | CC0E1D88 |

②

RStudio Desktop
Open Source Edition (AGPL v3)

Free

DOWNLOAD

Professional

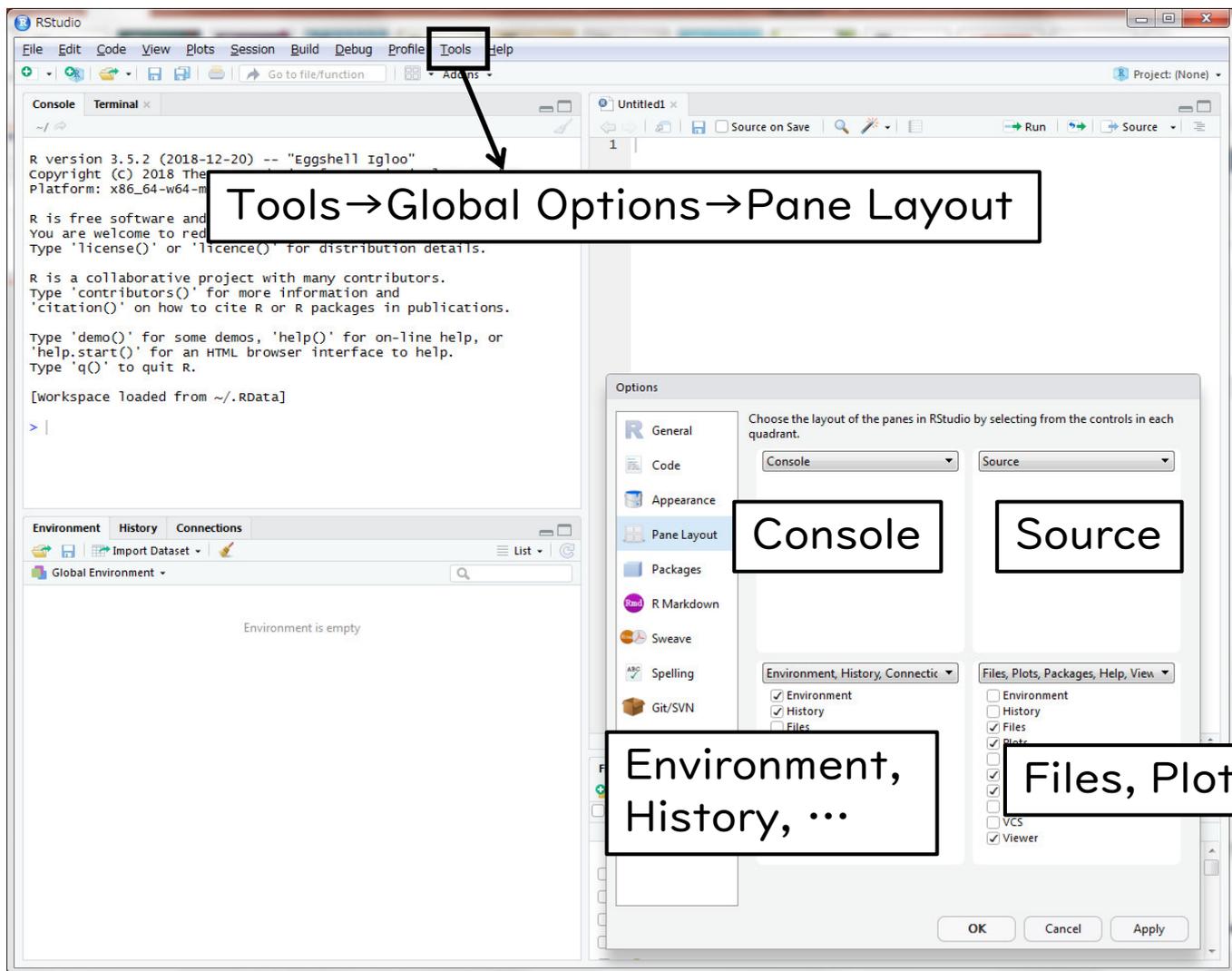
RStudio Professional
\$1,045 Per Year

BUY NOW

③

Windows 10/11 RStudio-2024.04.2-764.exe

Rstudioの画面の設定



Rstudioの画面

The screenshot displays the RStudio environment. On the left, the **Console** pane shows the R startup message and a prompt. A text box with a black border and white background is overlaid on the console, containing the text: **コンソール画面 (直接入力の場合)**. On the right, the **Source** editor shows an R script named `chemo004.R`. The script contains the following code:

```
1 rm(list=ls(all=TRUE))
2
3 # // pls regression
4 library(chemometrics)
5 library(pls)
6 #library(gclus)
7
8 data(NIR)
9
10 X0
11 Y1
12 Y2
13
14 X1
15 fo
16
17 X1 <- rbind(X1,unlist(X0[i,]))
18 }
19
20 X <- scale(X1)
21
22 <
```

A text box with a black border and white background is overlaid on the source code, containing the text: **ソースコード (まとめて実行する場合)**. An arrow points from this box to the **Source** button in the toolbar above the editor. At the bottom of the window, the **Environment** pane shows a list of objects in the global environment, including `B`, `b`, `p`, `U`, `V`, `w`, and `X`. The **Viewer** pane at the bottom right shows the **Data from cereals** documentation page.

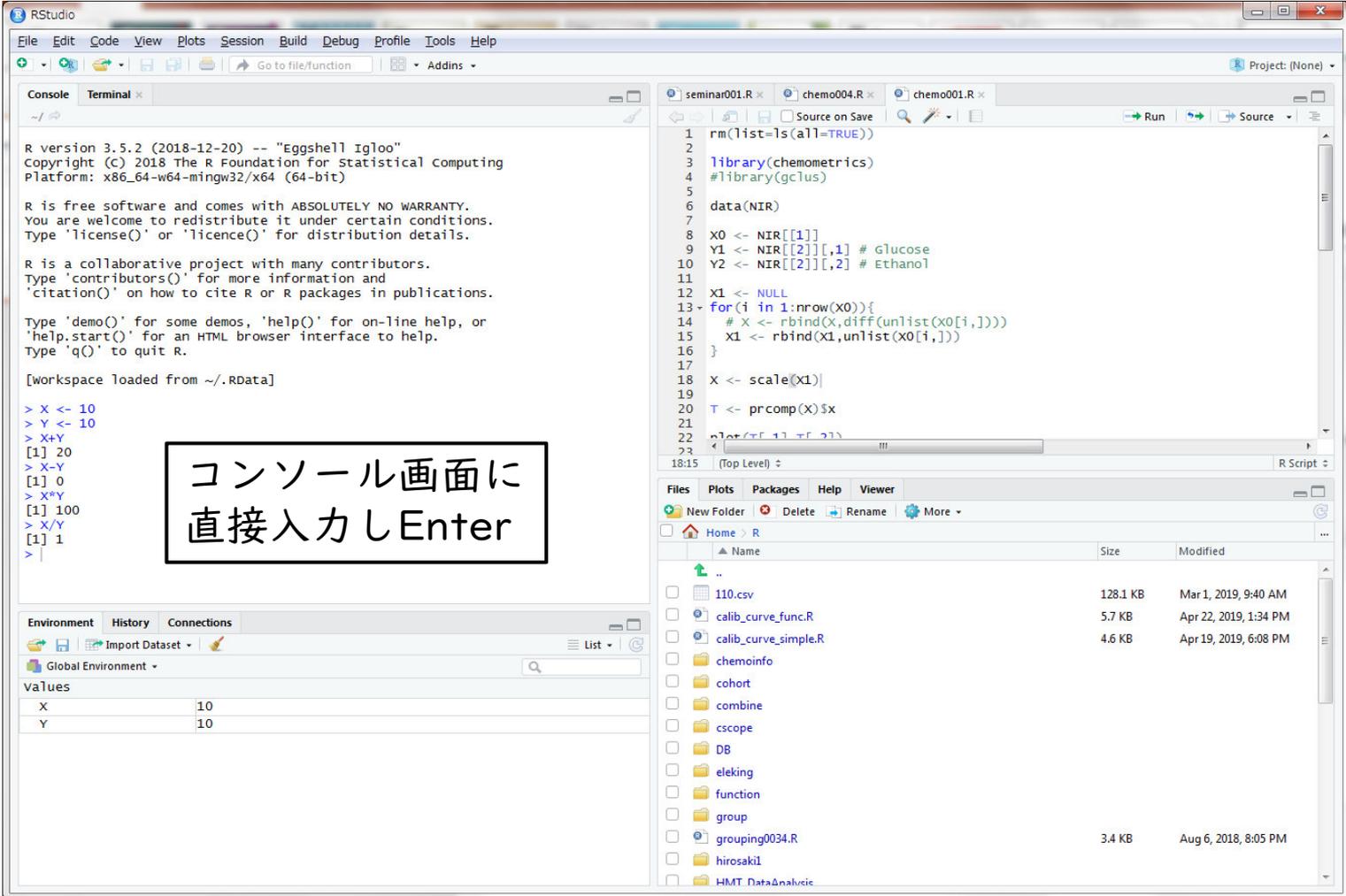
簡単な計算

データの代入と四則演算

- データの代入
 - $X \leftarrow 10$ ($X=10$ でも良いが、 $X \leftarrow 10$ と書くのが一般的)
 - $Y \leftarrow 10$
- 四則演算
 - 足し算 $X+Y$
 - 引き算 $X-Y$
 - 掛け算 $X*Y$
 - 割り算 X/Y

コンソール画面で実行して簡単に計算できる

Rstudioでの四則演算



The screenshot shows the RStudio interface. The console on the left displays the R version information and the results of several arithmetic operations performed in the terminal. A text box highlights the instruction to enter commands directly into the console. The code editor on the right shows an R script with various operations, including loading a package, creating data, and performing calculations on a matrix.

Console output:

```
R version 3.5.2 (2018-12-20) -- "Eggshell Igloo"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[workspace loaded from ~/.RData]

> X <- 10
> Y <- 10
> X+Y
[1] 20
> X-Y
[1] 0
> X*Y
[1] 100
> X/Y
[1] 1
>
```

Code editor content:

```
1 rm(list=ls(all=TRUE))
2 library(chemometrics)
3 #library(gclus)
4
5 data(NIR)
6
7 X0 <- NIR[[1]]
8 Y1 <- NIR[[2]][,1] # Glucose
9 Y2 <- NIR[[2]][,2] # Ethanol
10
11 X1 <- NULL
12 for(i in 1:nrow(X0)){
13   # X <- rbind(X,diff(unlist(X0[i,])))
14   X1 <- rbind(X1,unlist(X0[i,]))
15 }
16
17 X <- scale(X1)
18
19 T <- prcomp(X)$x
20
21 plot(T[,1],T[,2])
22
23
```

Environment window:

| Values | |
|--------|----|
| X | 10 |
| Y | 10 |

Text box: コンソール画面に直接入力しEnter

ベクトルと行列

- ベクトル

```
X <- c(1,2,3,4,5)
```

```
Y <- c(6,7,8,9,10)
```

- 行列

```
Z <- matrix(c(1,2,3,4),nrow=2, byrow=T)
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

- 行列の転置

```
t(Z)
```

$$\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

データの読み込み

Rのデータ形式

- ベクトル
 - 全ての要素が同じ型(数値または文字列)である必要がある
- 行列
 - 全ての要素が同じ型である必要がある
- リスト
 - 全ての要素が同じ型である必要はない
 - リストの各要素(ベクトル、行列)は同じ型である必要がある
- データフレーム
 - 全ての要素が同じ型である必要はない
 - 各列はリストであり、各列(リスト)は同じ型である必要がある

データの読み書き

- Rでのファイルの読み書き

- csvファイルが基本
- Excelファイルも可能だが、別途パッケージが必要

- csvファイルの読み込み [実習]

```
file <- "C:/R/mouse_data_original.csv"
```

```
X <- read.csv(file)
```

- csvファイルへの書き込み

```
write.csv(Z, file= "C:/R/matrix_z.csv")
```

csvファイルの読み込みが上手く行かない時

- ファイルの問題

- csvファイルをExcelで開き、“CSV(コンマ区切り)”の形式で保存し直す

- ファイルの読み込みプログラムの問題

- 文字コードを指定してファイルを読み込む

```
X <- read.csv(file , fileEncoding = "utf8")
```

read.csv関数でファイルが読み込めない時は、プログラムの問題だけではなく、データそのものの問題であることも多い

データの確認

```
> file <- "C:/R/mouse_data_original.csv"
> X <- read.csv(file)
> X
```

| | X | ko15 | ko16 | ko18 | ko19 | ko21 | ko22 | wt15 |
|----|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 1 | Label | KO | KO | KO | KO | KO | KO | WT |
| 2 | 200.1/2926 | 147887.53 | 451600.71 | 65290.38 | 56540.93 | 85146.33 | 162012.44 | 175177.08 |
| 3 | 205/2791 | 1778569 | 1567038 | 1482796 | 1039130 | 1223132 | 1072038 | 1950287 |
| 4 | 206/2791 | 237993.6 | 269714 | 201393.4 | 150107.3 | 176989.7 | 156797 | 276541.8 |
| 5 | 207.1/2719 | 380873 | 460629.7 | 351750.1 | 219288 | 286848.6 | 235022.6 | 417169.6 |
| 6 | 219.1/2524 | 235544.92 | 173623.38 | 82364.59 | 79480.4 | 185792.43 | 174458.77 | 244584.47 |
| 7 | 231/2516 | 117649.77 | 48960.63 | 222609.07 | 286232.15 | 435094.49 | 62168.71 | 465898.01 |
| 8 | 233/3023 | 399145.3 | 356951.3 | 410550.7 | 198416.5 | 363381.7 | 317805.8 | 397107.8 |
| 9 | 234/3024 | 76880.87 | 99526.27 | 97493.76 | 53461.71 | 88227.79 | 81072.23 | 65215.64 |
| 10 | 235.1/2695 | 171995.22 | 128945.16 | 155442.48 | 115286.25 | 37769.45 | 7663.88 | 199981.49 |
| 11 | 236.1/2524 | 252282.04 | 206031.93 | 71763.79 | 73602.47 | 186660.98 | 198804.28 | 253791.07 |

1列目は変数名(ピークの情報)、2列目からデータ

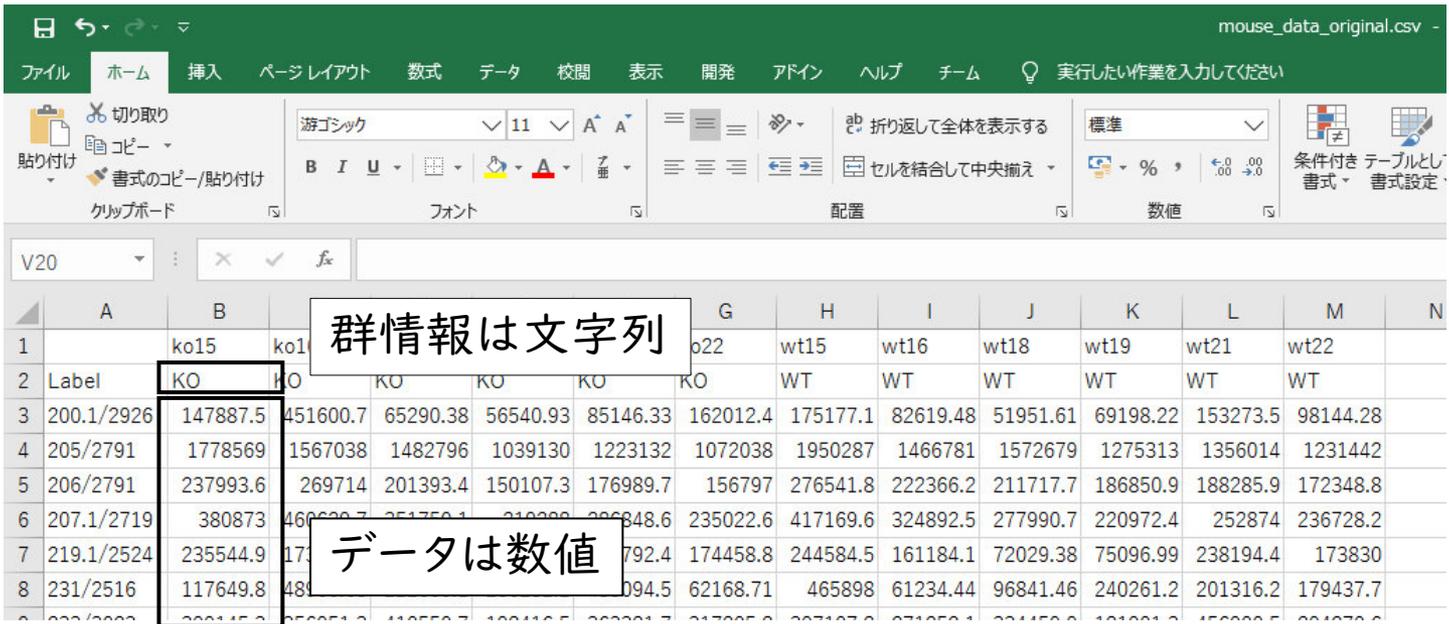
- データ部分の取得
X[,2] # 2列目のデータを取得

```
> X[,2]
```

| | | | | | | | |
|------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| [1] | "ko" | "147887.53" | "1778569" | "237993.6" | "380873" | "235544.92" | "117649.77" |
| [8] | "399145.3" | "76880.87" | "171995.22" | "252282.04" | "112440.56" | "1465988.7" | "280767.61" |
| [15] | "612169.85" | "27932.12" | "1435000.7" | "347794.8" | "78911.16" | "1420043.2" | "307708.4" |
| [22] | "229192.8" | "188817.16" | "23256.97" | "379553.6" | "1463921.1" | "7614.76" | "306981.48" |
| [29] | "666924.5" | "130223.45" | "332583.56" | "224307.75" | "72468.75" | "16540436" | "2942033" |
| [36] | "403697.8" | "2772937" | "745727.2" | "695800.8" | "414367.6" | "634718.21" | "733321.8" |
| [43] | "715122.1" | "1557473.4" | "381592.9" | "233984.2" | "306747.8" | "331377.6" | "158999.1" |
| [50] | "40056.23" | "136683.81" | "180780.82" | "4534353.6" | "2739847" | "962353.43" | "643996.2" |
| [57] | "494305.8" | "1948206" | "994373.8" | "557848.3" | "215368.1" | "125473.85" | "349067.12" |
| [64] | "312927.9" | "820897.8" | "474829.72" | "192388.21" | "293193.9" | "219733.4" | "1704155.4" |
| [71] | "1225817.5" | "388253.14" | "292071.25" | "103665.24" | "16394.42" | "107994.28" | "554864.61" |

データが文字列("で囲われている)になっており、このままでは解析出来ない

データが文字列になる理由



The screenshot shows an Excel spreadsheet with the following data:

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | |
|---|------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|------|
| 1 | | ko15 | ko16 | ko17 | ko18 | ko19 | ko20 | ko21 | ko22 | wt15 | wt16 | wt18 | wt19 | wt21 | wt22 |
| 2 | Label | KO | KO | KO | KO | KO | KO | WT | WT | WT | WT | WT | WT | | |
| 3 | 200.1/2926 | 147887.5 | 451600.7 | 65290.38 | 56540.93 | 85146.33 | 162012.4 | 175177.1 | 82619.48 | 51951.61 | 69198.22 | 153273.5 | 98144.28 | | |
| 4 | 205/2791 | 1778569 | 1567038 | 1482796 | 1039130 | 1223132 | 1072038 | 1950287 | 1466781 | 1572679 | 1275313 | 1356014 | 1231442 | | |
| 5 | 206/2791 | 237993.6 | 269714 | 201393.4 | 150107.3 | 176989.7 | 156797 | 276541.8 | 222366.2 | 211717.7 | 186850.9 | 188285.9 | 172348.8 | | |
| 6 | 207.1/2719 | 380873 | 460000.7 | 251750.1 | 210000 | 200000 | 235022.6 | 417169.6 | 324892.5 | 277990.7 | 220972.4 | 252874 | 236728.2 | | |
| 7 | 219.1/2524 | 235544.9 | 170000 | 170000 | 170000 | 170000 | 1792.4 | 174458.8 | 244584.5 | 161184.1 | 72029.38 | 75096.99 | 238194.4 | 173830 | |
| 8 | 231/2516 | 117649.8 | 480000 | 480000 | 480000 | 480000 | 094.5 | 62168.71 | 465898 | 61234.44 | 96841.46 | 240261.2 | 201316.2 | 179437.7 | |

read.csv関数でcsvファイルを読み込む際に、データの中に文字列が入っていると、全て文字列になってしまい、そのままでは主成分分析が計算できない

→ 読み込む際に1行目をスキップする、数値に書き換える、csvファイルで削除する等が必要

文字列の行をスキップして読み込む

- csvファイルの読み込み [実習]

```
file <- "C:/R/mouse_data_original.csv"
```

```
X <- read.csv(file, skip=1)
```

```
X[,2]
```

```
> X <- read.csv(file, skip=1)
> X[,2]
 [1] 147887.53 1778569.00 237993.60 380873.00 235544.92 117649.77 399145.30
 [8] 76880.87 171995.22 252282.04 112440.56 1465988.70 280767.61 612169.85
[15] 27932.12 1435000.70 347794.80 78911.16 1420043.20 307708.40 229192.80
[22] 188817.16 23256.97 379553.60 1463921.10 7614.76 306981.48 666924.50
[29] 130223.45 332583.56 224307.75 72468.75 16540436.00 2942033.00 403697.80
[36] 2772937.00 745727.20 695800.80 414367.60 634718.21 733321.80 715122.10
[43] 1557473.40 381592.90 233984.20 306747.80 331377.60 158999.10 40056.23
[50] 136683.81 180780.82 4534353.60 2739847.00 962353.43 643996.20 494305.80
[57] 1948206.00 994373.80 557848.30 215368.10 125473.85 349067.12 312927.90
[64] 820897.80 474829.72 192388.21 293193.90 219733.40 1704155.40 1225817.50
[71] 388253.14 292071.25 103665.24 16394.42 107994.28 554864.61 76370.15
[78] 297015.32 648546.10 238208.46 158127.90 339832.00 221263.21 4809521.40
[85] 190747.10 1108851.28 6463382.90 325363.80 1521141.36 201588.70 94656.81
[92] 466357.20 72668.45 31913.97 13976.57 382769.56 71757.92 73353.49
```

データが数値になっており、解析を行うことが出来る

Rでのデータの取り扱い

- ファイルの読み書き

- csvファイルが基本
- Excelファイルも可能だが、別途パッケージが必要

- データの前処理

- csvファイルのセルに文字が入っていると、読み込んだ後に数値データではなく文字列として認識され、その後の解析(主成分分析等)でエラーになる
 - 出来るだけ文字を使わないようにする
 - 数字で置き換えるものは数字で置き換える
 - 解析に不要な文字列は削除する
 - 文字の入った行をスキップして読み込む

データの基本的な扱い方

データのサイズの確認

- 行数

`nrow(X)`

- 列数

`ncol(X)`

- データのサイズ

`dim(X)`

基本的なデータの扱い方(1)

- 1行1列の値を取得する

$X[1,1]$

- 1行目のデータを取得する

$X[1,]$ # ベクトル

- 2列目のデータを取得する

$X[:,2]$ # ベクトル

- 2行2列から4行4列までのデータを取得する

$X[2:4,2:4]$ # 行列

計算結果

```
> file <- "c:/R/mouse_data_original.csv"
> X <- read.csv(file, skip=1)
> nrow(X)
[1] 409
> ncol(X)
[1] 13
> dim(X)
[1] 409 13
> X[1,1]
[1] "200.1/2926"
> X[1,]
  Label      KO      KO.1      KO.2      KO.3      KO.4      KO.5      WT      WT.1      WT.2
1 200.1/2926 147887.5 451600.7 65290.38 56540.93 85146.33 162012.4 175177.1 82619.48 51951.61
  WT.3      WT.4      WT.5
1 69198.22 153273.5 98144.28
> X[,2]
 [1] 147887.53 1778569.00 237993.60 380873.00 235544.92 117649.77 399145.30
 [8] 76880.87 171995.22 252282.04 112440.56 1465988.70 280767.61 612169.85
[15] 27932.12 1435000.70 347794.80 78911.16 1420043.20 307708.40 229192.80
[22] 188817.16 23256.97 379553.60 1463921.10 7614.76 306981.48 666924.50
[29] 130223.45 332583.56 224307.75 72468.75 16540436.00 2942033.00 403697.80
[36] 2772937.00 745727.20 695800.80 414367.60 634718.21 733321.80 715122.10
[43] 1557473.40 381592.90 233984.20 306747.80 331377.60 158999.10 40056.23
[50] 136683.81 180780.82 4534353.60 2739847.00 962353.43 643996.20 494305.80
> X[2:4,2:4]
      KO      KO.1      KO.2
2 1778569.0 1567038.0 1482796.0
3 237993.6 269714.0 201393.4
4 380873.0 460629.7 351750.1
```

基本的なデータの扱い方(2)

- 1行目のデータを除いたデータを取得する
`X[-1,]` # 行列
- 1列目のデータを除いたデータを取得する
`X[, -1]` # 行列
- 最終行を除いたデータを取得する
`X[-nrow(X),]` # 行列
- 最終列を除いたデータを取得する
`X[, -ncol(X)]` # 行列

行名と列名の取得、列名の指定

- データの行名
rownames(X)
- データの列名
colnames(X)
- 列名 (Label) を指定して列 (Label) のデータを取得する
X\$Label

グラフの作成

グラフの作成

- 2列目のデータ

```
plot(X[,2])
```

- 赤色を指定する

```
plot(X[,2], col="red")
```

- 線を指定する

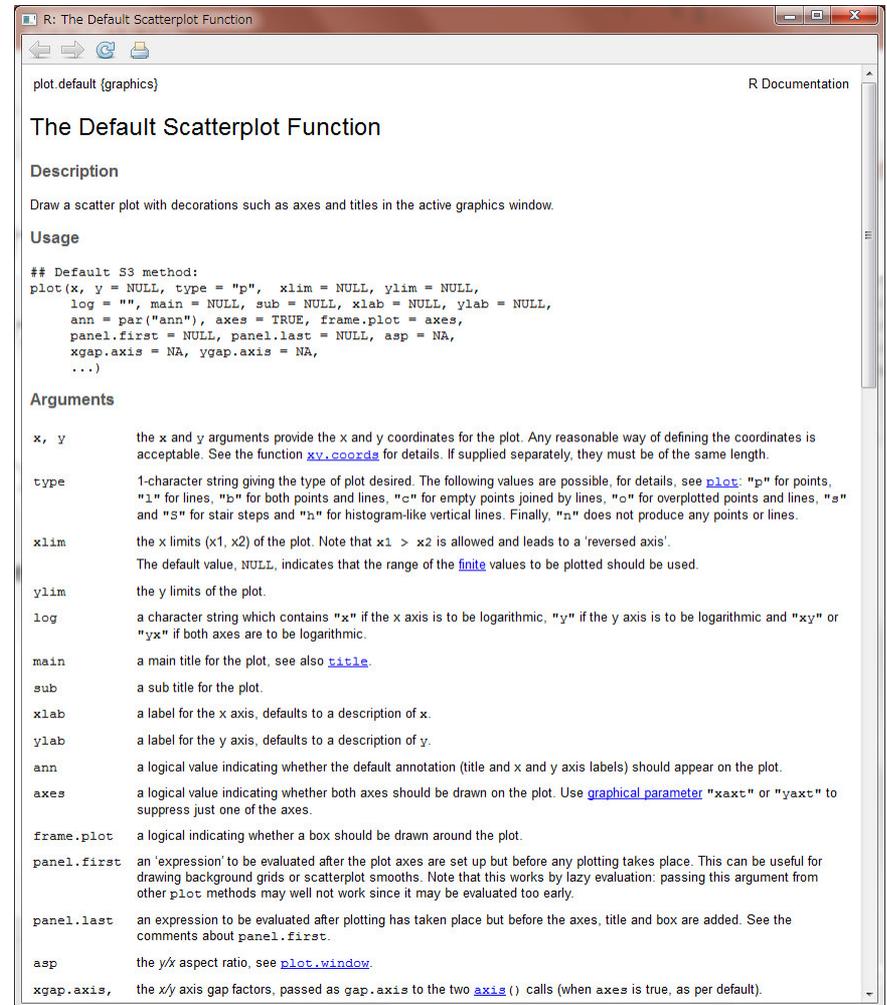
```
plot(X[,2], type="l")
```

- 形を●に指定する

```
plot(X[,2], pch=16)
```

- 大きくする

```
plot(X[,2], cex=2)
```



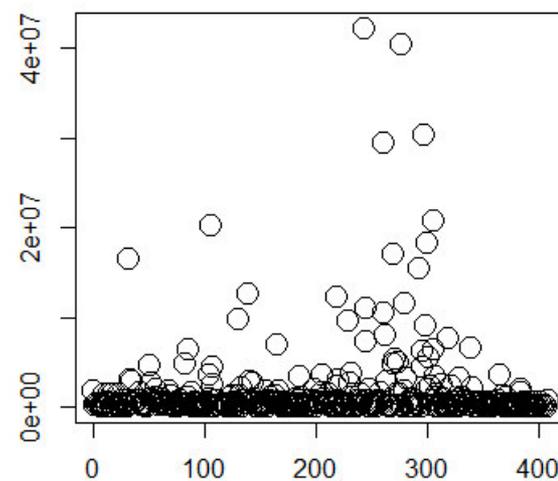
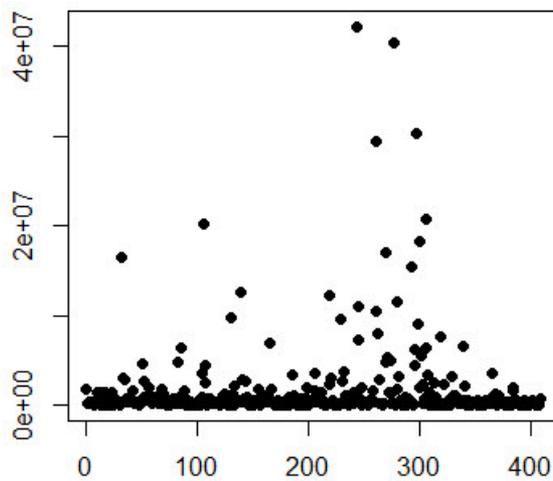
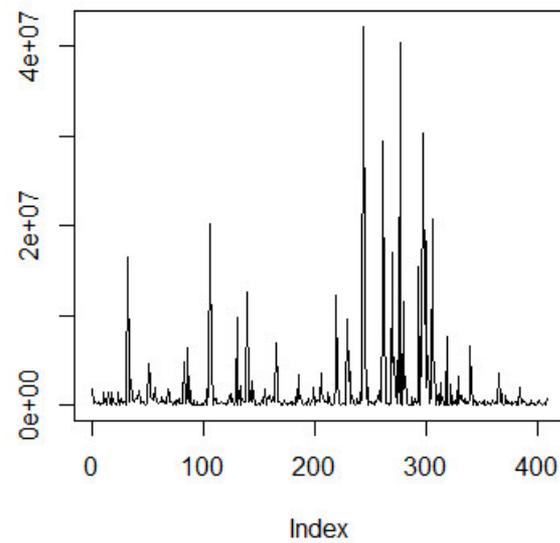
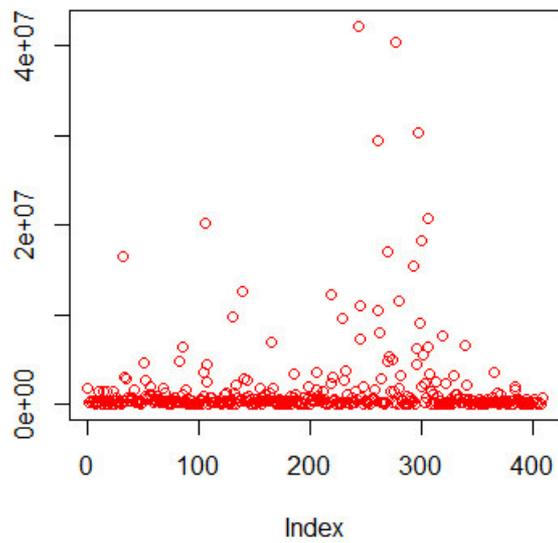
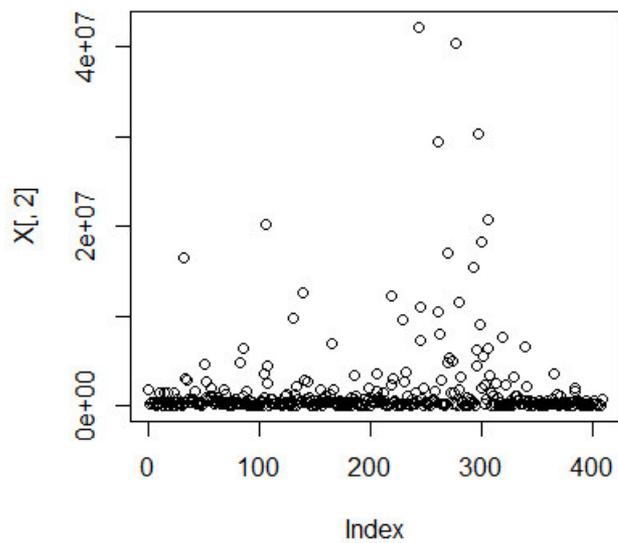
The screenshot shows the R documentation for the `plot.default` function. The window title is "R: The Default Scatterplot Function". The content includes the following sections:

- Description:** Draw a scatter plot with decorations such as axes and titles in the active graphics window.
- Usage:**

```
## Default S3 method:  
plot(x, y = NULL, type = "p", xlim = NULL, ylim = NULL,  
      log = "", main = NULL, sub = NULL, xlab = NULL, ylab = NULL,  
      ann = par("ann"), axes = TRUE, frame.plot = axes,  
      panel.first = NULL, panel.last = NULL, asp = NA,  
      xgap.axis = NA, ygap.axis = NA,  
      ...)
```
- Arguments:**
 - `x, y`: the `x` and `y` arguments provide the `x` and `y` coordinates for the plot. Any reasonable way of defining the coordinates is acceptable. See the function [xy.coords](#) for details. If supplied separately, they must be of the same length.
 - `type`: 1-character string giving the type of plot desired. The following values are possible, for details, see [plot](#): "p" for points, "l" for lines, "b" for both points and lines, "c" for empty points joined by lines, "o" for overplotted points and lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
 - `xlim`: the `x` limits (`x1`, `x2`) of the plot. Note that `x1 > x2` is allowed and leads to a 'reversed axis'. The default value, `NULL`, indicates that the range of the [finite](#) values to be plotted should be used.
 - `ylim`: the `y` limits of the plot.
 - `log`: a character string which contains "x" if the `x` axis is to be logarithmic, "y" if the `y` axis is to be logarithmic and "xy" or "yx" if both axes are to be logarithmic.
 - `main`: a main title for the plot, see also [title](#).
 - `sub`: a sub title for the plot.
 - `xlab`: a label for the `x` axis, defaults to a description of `x`.
 - `ylab`: a label for the `y` axis, defaults to a description of `y`.
 - `ann`: a logical value indicating whether the default annotation (title and `x` and `y` axis labels) should appear on the plot.
 - `axes`: a logical value indicating whether both axes should be drawn on the plot. Use [graphical parameter](#) "xaxt" or "yaxt" to suppress just one of the axes.
 - `frame.plot`: a logical indicating whether a box should be drawn around the plot.
 - `panel.first`: an 'expression' to be evaluated after the plot axes are set up but before any plotting takes place. This can be useful for drawing background grids or scatterplot smooths. Note that this works by lazy evaluation: passing this argument from other `plot` methods may well not work since it may be evaluated too early.
 - `panel.last`: an expression to be evaluated after plotting has taken place but before the axes, title and box are added. See the comments about `panel.first`.
 - `asp`: the `y/x` aspect ratio, see [plot.window](#).
 - `xgap.axis`, `ygap.axis`: the `x/y` axis gap factors, passed as `gap.axis` to the two [axis\(\)](#) calls (when `axes` is true, as per default).

詳しくは、`plot.default`関数のヘルプを参照

各グラフ



追加ライブラリ・パッケージ

追加パッケージの利用

- パッケージを探す

- Rをインストール直後の状態では、解析できる手法は限られているので、追加のパッケージをインストールして利用する
- パッケージはRの公式サイトCRAN(The Comprehensive R Archive Network)で公開されている
- 例えば、"chemometrics R package"でGoogle検索すると、CRANのchemometricsパッケージがヒットする

```
chemometrics: Multivariate Statistical Analysis in Chemometrics
```

```
R companion to the book "Introduction to Multivariate Statistical Analysis in Chemometrics" written by K. Varmuza and P. Filzmoser (2009).
```

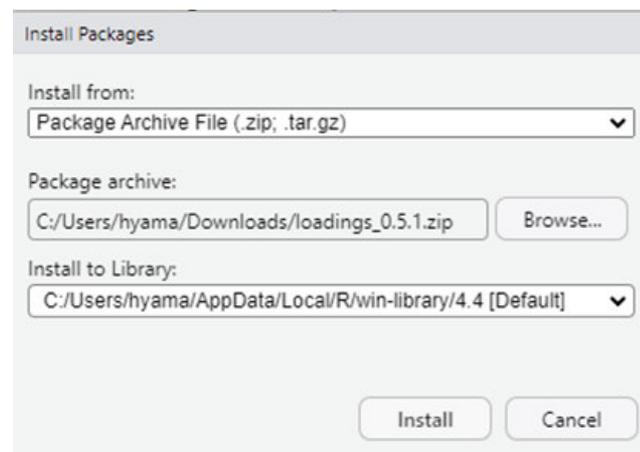
```
(https://cran.r-project.org/web/packages/chemometrics/index.html)
```

- 主成分分析とPLS

- 主成分分析prcompは、Rの基本パッケージstatsに含まれているので、追加のパッケージのインストールは不要
- Partial least squares (PLS)はstatsに含まれないので、loadingsもしくはchemometricsパッケージをインストールして使用

追加パッケージのインストール

- コマンドでインストールする場合(要ネット環境)
`install.packages("loadings")`
- ローカルでインストールする場合
 - <https://cran.r-project.org/web/packages/loadings/index.html>
 - の下の方にあるWindows binariesにあるzipファイルをダウンロードし、
Rstudioのメニューから
Tools→Install packages…
を選択し、
保存したzipファイルを選択して
Installボタンを押す



loadingsの依存パッケージであるgeigenも同様にインストールする
<https://cran.r-project.org/web/packages/geigen/index.html>

Rパッケージの調査(cranlogsパッケージの利用)

- cranlogsパッケージの読み込み

```
# install.packages("cranlogs")  
library(cranlogs)
```

- パッケージのダウンロード数の確認

```
dl_top <- cran_top_downloads(when = "last-month",  
                             count = 10)
```

- 機械学習パッケージの比較

```
dl_caret <- cran_downloads(packages = "caret",  
                           when = "last-month")  
dl_mlr <- cran_downloads(packages = "mlr",  
                         when = "last-month")
```

その他

Rstudio(またはR)の機能：コマンドの履歴

The screenshot shows the RStudio interface with the following components:

- Console:** Displays the R command prompt and the execution of several commands. The commands are: `rm(list=ls(all=TRUE))`, `library(chemometrics)`, `data(NIR)`, `X0 <- NIR[[1]]`, `Y1 <- NIR[[2]][,1] # Glucose`, and `Y2 <- NIR[[2]][,2] # Ethano1`. A warning message is visible: "要求されたパッケージ rpart をロード中です" and "パッケージ 'chemometrics' はバージョン 3.5.3 の R の下で進られました".
- Code Editor:** Shows the source code for the script, which matches the commands in the console.
- Environment/History/Connections:** The History tab is active, showing a list of commands that have been executed, including the same commands as in the console.

Two callout boxes provide additional information:

- The first callout box, with an arrow pointing to the console, states: "キーボードの”上矢印↑”で1つ前のコマンド、“下矢印↓”で1つ後のコマンドが表示される".
- The second callout box, with an arrow pointing to the History tab, states: "Historyタブには、コマンドの履歴が表示される".

?関数名 (?prcomp) でヘルプの表示

The screenshot shows the RStudio interface with the following elements:

- Console:** Shows the command `?prcomp` and the help text for the `prcomp` function.
- Environment:** Shows the loaded packages: `rm(list=ls(all=TRUE))`, `library(chemometrics)`, `data(NIR)`, `X0 <- NIR[[1]]`, `Y1 <- NIR[[2]][,1] # Glucose`, `Y2 <- NIR[[2]][,2] # Ethanol`, `?matrix`, `?library`, `?prcomp`, and `?prcomp`.
- Files/Plots/Package Viewer:** Shows the file `combine001.R` and the package `R: Principal Components Analysis`.
- Help Window:** A new window titled `R: Principal Components Analysis` is open, displaying the help text for the `prcomp` function. The text includes the function signature `prcomp {stats}`, the title `Principal Components Analysis`, the description `Performs a principal components analysis on the given data matrix and returns the results as an object of class prcomp.`, and the usage `prcomp(x, ...)`.

A box highlights the title bar of the help window, with an arrow pointing to it and the text `新しいウィンドウで表示` (Display in a new window).

Windows 10でRstudioを使用する際の注意

- インストール先の「ライブラリのパス」または「Rのプログラムのファイルパス」に日本語が入っていると、エラーが起こる
- パッケージのインストールがうまくいかない場合は、Rstudioを管理者権限で起動(右クリック→” 管理者として実行”)してパッケージをインストールするとうまくいく
- One Driveを使用している(パッケージやファイルのパスにOne Driveが入っていると、エラーが起こる

上記のような問題が全ての場合に起こるわけではなく、ケースバイケース