

# 医学統計学演習：資料 1

芳賀昭弘\*

## 1 R とは？

R は、統計解析向けのプログラミング言語です。プログラミング → コンピュータに処理させたい事柄を、コンピュータが分かる言葉で指示する行為のことで、プログラミング言語には機械にとって理解しやすい低水準言語（機械語とアセンブリ言語）と人間の言葉に近い形式で記述できる高水準言語（Fortran や C 言語など）があります。R は高水準言語に分類されます。また、高水準言語はコンパイル型とインタプリタ型<sup>\*1</sup>に分けることができるが、R はインタプリタ型の言語に分類されます。

R 以外にも、エクセル統計、SPSS など統計解析用のソフトウェアはたくさんあります。しかし、R はそれらに比して様々な点から有用であると言えます。R はフリーソフトウェアであり、また自由に利用・改変・再配布できるようになっていますが、エクセル統計、SPSS は有償であり、改変・再配布はできません。R の開発は、多くの人が無償で協力して行っているとともに、多くの研究者によって不具合や注意点の共有がなされています。様々な研究者が開発した最新の統計解析手法もパッケージという形で公開され、それをダウンロードすることで誰でも利用できます。流行りの人工知能・機械学習も R を用いて勿論行うことができるし、プログラミング言語であるため、自分なりの拡張が可能です（R 以外の統計解析向けのプログラミング言語では、python の評判が高い。python もフリーでインストールできるインタプリタ型の高水準言語（スクリプト言語）です。R か python のどちらかをマスターしておけば、どんな環境においても統計解析を簡単に行うことができます）。

R が使えるようになると、高度な実験解析が可能になります。医療の分野でも広く使われており、学部間に習得しておく、就職や大学院進学後に非常に助かることでしょう！ Windows, Mac, Linux など様々な OS 用の R 実行環境が提供されているので、自分が所有するパソコンには必ず入れておきましょう！

## 2 R を使った計算

R のコンソール画面の `>` のところに直接打ち込んでいく。

- 足し算

```
> 1 + 1
```

---

\* Electronic address: [haga@tokushima-u.ac.jp](mailto:haga@tokushima-u.ac.jp)

<sup>\*1</sup> コンパイラ言語はソースプログラムを機械語に変換するプロセスが必要ですが、インタプリタ言語は 1 ステップ（1 行）ごとに機械語に解釈しながら実行してくれる。

```
> x <- 5
> y <- 1 + 1
> x + y
```

---

<-の部分は = でもよい。

- 引き算

```
> 8 - 0.5
```

---

```
> x <- 4
> y <- 0.0001
> y - x
```

- 掛け算

```
> 2 * 5
```

---

```
> x <- 2
> y <- -3
> x * (y + 3)
```

- 割り算

```
> 2/5
```

---

```
> x <- 1/2
> y <- 1/3
> x/y/y
```

---

どのようなプログラミング言語でも、0で割るようなプログラムは書かないように常に細心の注意を払

うこと

- 平方根

```
> sqrt(2)
```

---

```
> x <- sqrt(2)
format(x, digit = 10)
```

---

format は書式を指定する。digit は、表示する桁を示している（小数点以下の桁を指定する場合は nsmall を使う）。

- 乗数

```
> 2^8
```

(2\*\*8 と書いても良い)

---

```
> r <- 2^8
> g <- 2^8
> b <- 2^8
> r * g * b
```

---

- 指数関数

```
> exp(-1)
```

---

```
> d <- 2
> a <- 0.1
> b <- 0.01
> exp(-a * d - b * d^2)
```

---

- 対数関数

```
> log(2)
```

---

```
> log 10(2)
```

---

```
> log 2(2)
```

---

- 三角関数

```
> sin(pi)
```

---

```
> cos(pi/4)
```

---

```
> asin(1)
```

---

- ベクトル演算

```
> vecex <- c(1,0,0)
> vecey <- c(0,1,0)
> vecez <- c(0,0,1)
> vecex + vecey + vecez
```

---

```
> vecx <- c(1,2,3)
> vecy <- c(3,2,1)
> vecx * vecy
```

---

```
> vecx <- c(1:3)
> vecy <- c(3:1)
> vecx %*% vecy
```

---

$c(1, \dots, n)$  と  $c(1:n)$  は同じ。:は 1 ずつ増加する数を与えてくれる便利な記号。この他に、seq(始めの

数, 終わりの数, 増分) のような等差数列を作るものや `rep(数, 繰り返し数)` といった関数がある。

- 行列演算

```
> matA <- matrix(1:9,nrow=3,ncol=3)
> matA
```

---

```
> matB <- matrix(1:9,nrow=3,ncol=3,byrow=TRUE)
> matB
```

---

```
> matA %*% matB
```

---

```
> matC <- matrix(c(5,-3,3,-1),nrow=2,ncol=2)
> det(matC)
```

---

```
> solve(matC)
```

---

```
> eigen(matC)
```

---

行列演算は、医療でも頻繁に使われます。とりわけ固有値問題は、今後の医療応用においても重要な位置付けとなるでしょう。

ここで使われた関数 (`det`, `solve`, `eigen` だけでなく `exp`, `log` など) は、特に「定義」することなく使うことができました。これは「組み込み関数」と呼ばれており、プログラム言語 (c 言語, `fortran`, `python`, `R` ...) によって組み込まれている関数の種類や宣言の仕方が異なります。使用している言語 (今の場合は `R`) に組み込まれている関数を知るにはインターネット環境があるのであれば検索が便利でしょう。もちろん、いちいち調べるのが大変なので、ある程度勉強しておいて頻出する関数の宣言の仕方や使い方は覚えておくとよいです (そのために、教科書を指定していますので、講義対象の部分は各自読

むようにしてください)。

また、組み込まれていない関数も、自分で定義することで関数として使えるようにすることもできます。この方法は次回以降に学びます。

## 3 データの読み書き

### 3.1 R でデータを作成する方法

ファイル名やデータの項目などに日本語を使いたい気持ちはよくわかります。ですが、**コンピュータ、特にプログラムを作成する際には日本語の使用はバグの温床となります**。なるべく英数字を使ってファイル名やデータを扱うように心掛けてください。

```
> EOM <- data.frame(ID = c(1:9), Before = c(0.1, 0.3, 0.3, 0.1, 0.2, 0.3, 0.6, 0.4, 0.2),  
+ Just.after = c(1.1, 0.7, 0.6, 0.8, 1.3, 0.5, 0.8, 0.7, 0.9))  
> EOM[,2]
```

---

(注意) “After” の前についている “+” は、改行すると勝手につくものなので、実際には打たない。

(試してみよう) 上の EOM[,2] において、2 を 3 に変えたらどうなる？

(試してみよう) EOM[“Before”] とするとどうなる？

(試してみよう) EOM とするとどうなる？

データを付け加えたいとき；

```
> One.hr.after <- c(0.3, 0.3, 0.4, 0.2, 0.2, 0.3, 0.6, 0.5, 0.3)  
> data.frame(EOM, One.hr.after)
```

---

```
> EOM2 <- data.frame(EOM, One.hr.after)  
> write.csv(EOM2, "EOM2.csv", quote=FALSE, row.names=FALSE)
```

(作業ディレクトリ内に EOM2.csv ファイルができていることを確認しよう)

quote=FALSE と row.names=FALSE はつけておく。前者をつけないと文字列には引用符 (“”) がついて記録される。後者をつけないと、行番号も csv に記録される。

EOM2.csv をエクセルで立ち上げてみてください。データが正しく保存されているでしょうか？

### 3.2 保存したデータの読み込み

```
> dat <- read.csv("EOM2.csv")
> str(dat)
```

## 4 演習

1. インストールした R のバージョンと PC の機種を書け。
2.  $x = 2.4$  として、 $x^2$  を求めよ。
3.  $x = -1.5$  として、 $\exp(x^2)$  を求めよ。
4.  $30^\circ$  をラジアンで表せ。
5.  $\sin(30^\circ)$  を求めよ。
6. ベクトル  $\mathbf{a}^T = (1, 2, 3, 4, 5)$ ,  $\mathbf{b}^T = (5, 4, 3, 2, 1)$  の内積  $\mathbf{a}^T \mathbf{b}$  を求めよ。
7. 次の行列  $\mathbf{A}$  の固有値と固有ベクトルを求めよ。

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 2 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \end{pmatrix}$$

8. 次のデータを R で作成し、E\_1\_WH.csv という名前で保存する方法を書け。  
(ヒント：data.frame を使う)

PatientNo.	Weight.kg	Height.cm
1	65	170
2	45	165
3	70	160
4	72	171
5	54	162

9. 上で保存した csv ファイルをエクセルで開き、隣の列にボディマス指数 (BMI) を入れて E\_1\_WH\_BMI.csv という名前で保存し、保存した csv ファイルを R で読み込む方法を書け。
10. 前問について、エクセルを使わず、R だけを使って BMI を表に追加する方法を書け。  
(ヒント：表のデータが data という変数に入れ (`data <- read.csv("E_1_WH.csv")`)), BMI を `bmi <- data[,2]/data[,3]/data[,3]*10000` とすることで追加する。data に bmi を追加するには `data <- data.frame(data, bmi)`)