

R の練習 (r1) 2012-11-02

久保拓弥 kubo@ees.hokudai.ac.jp

この授業の web page: <http://goo.gl/wijx2>

統計ソフトウェア R は研究にたいへん役にたつ free software (無料で入手でき, しかも内部を自由に調べられる) です. 今回は R のデータ操作・作図の基本わざを説明します.

R を使ったデータ解析の基本的な流れは次のようになります:

1. データを読みこむ (データフレーム data.frame を作る)
2. 読みこんだデータをいろいろ整理する (データフレームの操作)
3. データをさまざまな方法で図示する
4. 統計モデリングの設計・あてはめを行う
5. あてはめの結果やモデルの予測を図示する
6. 解析結果をさまざまな方法で出力し, 保存する

今日は時間も限られているので, データの読みこみ, 基本的なデータフレーム操作, 簡単な図示について説明します. 上述の授業 web site のあちこちを見て, さらに発展したわざも勉強してください.

1 R でデータフレームの操作

1.1 データを読みこんで data.frame を作り, それを表示する

```
> d <- read.csv("data.csv")
```

```
> d
```

```
  treatment size seed
1   control 21.3    9
2     trtX 24.2   19
3   control 12.0    1
4     trtX 16.1    4
5   control 21.8   13
6     trtX 20.2    6
7   control 22.7    8
8     trtX 23.8    8
9   control 19.5    7
10    trtX 26.4   22
11  control 20.1    3
12    trtX 27.3   31
```

```
13 control 22.5 14
14 trtX 21.8 19
15 control 18.6 4
16 trtX 25.3 26
17 control 23.5 11
18 trtX 19.7 6
19 control 27.9 22
20 trtX 22.0 17
```

> head(d) # 最初の 6 行が表示される

```
  treatment size seed
1 control 21.3 9
2 trtX 24.2 19
3 control 12.0 1
4 trtX 16.1 4
5 control 21.8 13
6 trtX 20.2 6
```

> head(d, 3) # 最初の 3 行が表示される

```
  treatment size seed
1 control 21.3 9
2 trtX 24.2 19
3 control 12.0 1
```

> tail(d, 3) # 最後の 3 行が表示される

```
  treatment size seed
18 trtX 19.7 6
19 control 27.9 22
20 trtX 22.0 17
```

> edit(d) # d を編集する

1.2 data.frame から行と列をとりだす

> d[1:3,] # 1 行めから 3 行めをとりだす

```
  treatment size seed
1 control 21.3 9
2 trtX 24.2 19
3 control 12.0 1
```

> d[c(1, 3, 5),] # 1, 3, 5 行めをとりだす

```
  treatment size seed
```

```
1 control 21.3 9
3 control 12.0 1
5 control 21.8 13
```

```
> d[, 1] # 1 列めをとりだす
```

```
[1] control trtX control trtX control trtX ... 略
Levels: control trtX
```

```
> d[4:6, 2:3] # 4-6 行めの 2-3 列めをとりだす
```

```
size seed
4 16.1 4
5 21.8 13
6 20.2 6
```

```
# 列の選びかたに 3 とおりある (どれも重要)
```

```
> d[, 3] # 3 列めをとりだす
```

```
[1] 9 19 1 4 13 6 8 8 7 22 3 31 14 19 4 26 11 6 22 17
```

```
> d$seed # 上とおなじことをやっている
```

```
[1] 9 19 1 4 13 6 8 8 7 22 3 31 14 19 4 26 11 6 22 17
```

```
> d[, "seed"] # これも同じ
```

```
[1] 9 19 1 4 13 6 8 8 7 22 3 31 14 19 4 26 11 6 22 17
```

1.3 data.frame から条件つきデータとりだし

treatment が trtX のデータ

```
> d[d$treatment == "trtX",]
```

```
  treatment size seed
2      trtX 24.2  19
4      trtX 16.1   4
6      trtX 20.2   6
8      trtX 23.8   8
10     trtX 26.4  22
12     trtX 27.3  31
14     trtX 21.8  19
16     trtX 25.3  26
18     trtX 19.7   6
20     trtX 22.0  17
```

size が 25.0 より大きいデータ

```
> d[d$size > 25.0,]
```

```

treatment size seed
10      trtX 26.4  22
12      trtX 27.3  31
16      trtX 25.3  26
19  control 27.9  22

```

seed が 6 以下であるデータ

```

> d[d$seed <= 6,]
3      control 12.0    1
4       trtX 16.1    4
11     control 20.1    3
15     control 18.6    4
...

```

seed が 6 以下, かつ 2 より大

```

> d[d$seed <= 6 & d$seed > 2,]
...

```

seed が 6 より大, または 2 以下

```

> d[d$seed > 6 | d$seed <= 2,]
...

```

1.4 data.frame 内での並びかえ

```

> d <- d[order(d$size),] # d$size の小さい順に並べかえる
> d <- d[rev(order(d$size)),] # d$size の大きい順に並べかえる

```

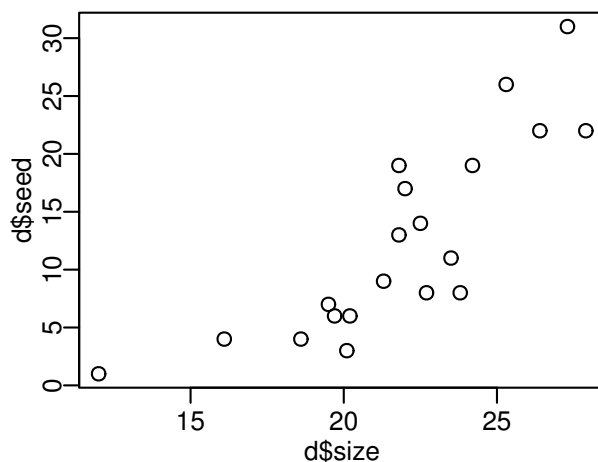
2 R で作図

R 作図の基本 (plot() 関数を使う場合)

- いっぺんに図を作ろうとするのではなく, 必要な要素を足していく
- plot() で「わく」を描く
- points(), lines(), legend() で必要なものを追加していく
- par(new = TRUE) による方法は使わないほうがよい (わくを何重にも描くことになったりするから)

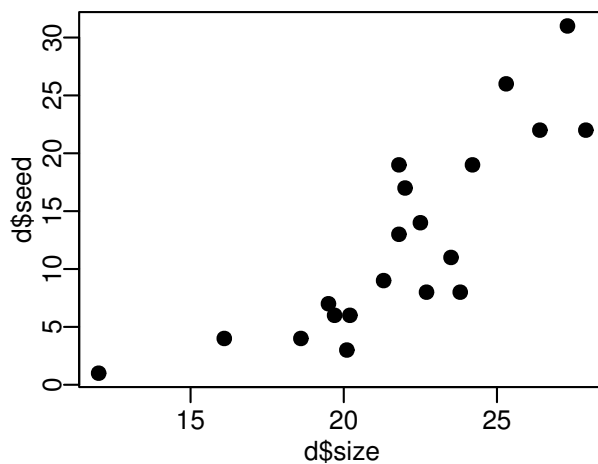
2.1 data.frame のデータを表示する

```
> d <- read.csv("r1.csv")  
> plot(d$size, d$seed)
```



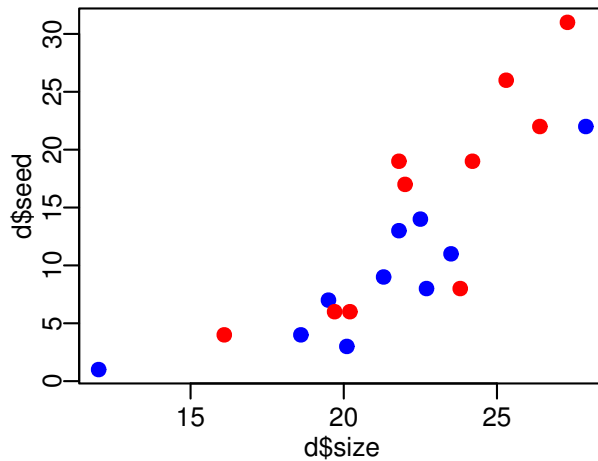
pch 引数で点の種類を変える

```
> plot(d$size, d$seed, pch = 19)
```



col 引数で点の色を変える

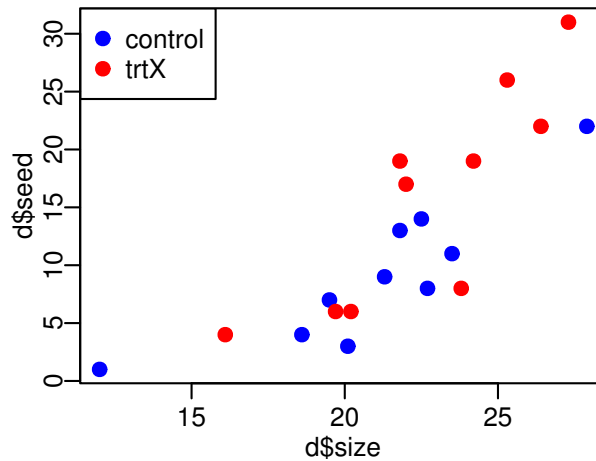
```
> plot(d$size, d$seed, pch = 19, col = c("blue", "red")[d$treatment])
```



legend() 関数で凡例を追加

```
# 上の図に legend を追加
```

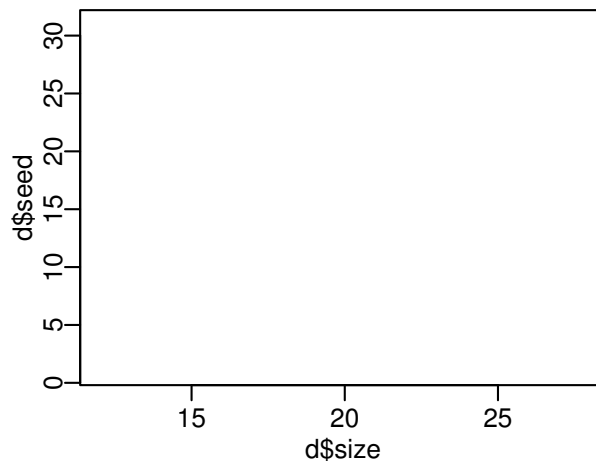
```
> legend("topleft", legend = levels(d$treatment), pch = 19, col = c("blue", "red"))
```



2.2 図を順にかさねていくわざ

最初にわくだけ描く

```
> plot(d$size, d$seed, type = "n") # わくだけ描く
```

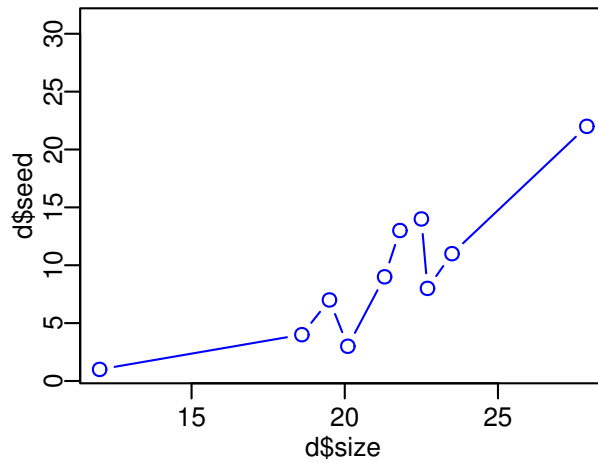


処理が control である線だけ描く

```
> dC <- d[d$treatment == "control",] # treatment が control のデータだけ
```

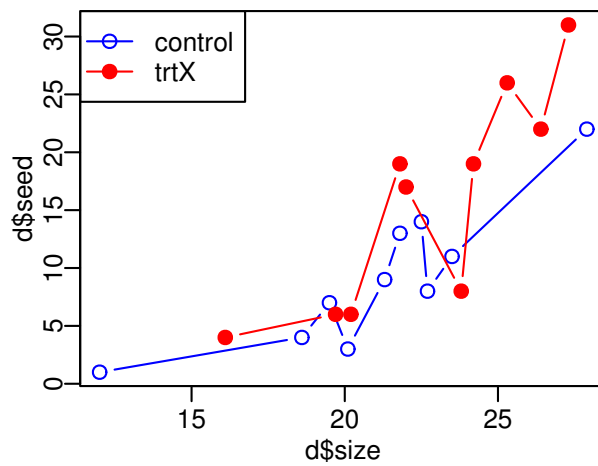
```
> dC <- dC[order(dC$size),] # size 順にならびかえる
```

```
> lines(dC$size, dC$seed, pch = 21, col = "blue") # 線を追加
```



次に処理が trtX である線を描き，凡例を追加する

```
> dX <- d[d$treatment == "trtX",] # treatment が trtX のデータだけ
> dX <- dX[order(dX$size),] # size 順にならびかえる
> lines(dX$size, dX$seed, pch = 21, col = "red") # 線を追加
> legend("topleft", legend = levels(d$treatment),
       pch = c(21, 19), col = c("blue", "red"), lwd = 1)
```



3 その他あれこれ

- pdf(), jpg(), png() といった device 指定でいろいろな形式で図を出力できる
- R 作図に慣れてきたら, library(lattice) や library(ggplot2) で, より「全体像のみやすい」図を作ろう

– library(lattice) を使った条件ごとプロットの例:

```
> d <- d[order(d$size),] # size 順にデータを並びかえ  
> print(xyplot(seed ~ size | treatment, data = d, type = "b"))
```

