データの分析と知識発見

主なデータの型と構造

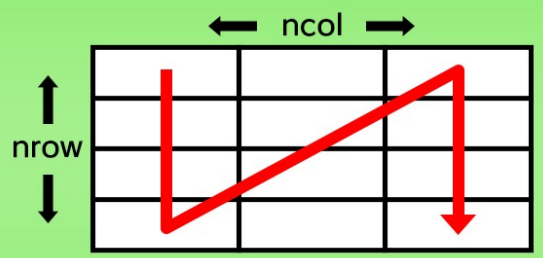
データの型	説　明	例
integer	整 数	1L
double	実数(倍精度浮動小数)	3.4、5e-10
character	文字列	“A”, “B”
logical	論理値	TRUE、FALSE

構　造	名　前	次　元	データ型の種類
vector	ベクトル	1次元	1種類
matrix	行列	2次元	1種類
data.frame	データフレーム	2次元	複 数
list	リスト	1次元	複 数

行　列

```
> A <- matrix( c(1,2,3,4,5,6) ,nrow =2, ncol=3 )
> A
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> A <- matrix( c(1,2,3,4,5,6) ,ncol=3, byrow=T)
> A
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

matrix (ベクトル,ncol = 列の数)



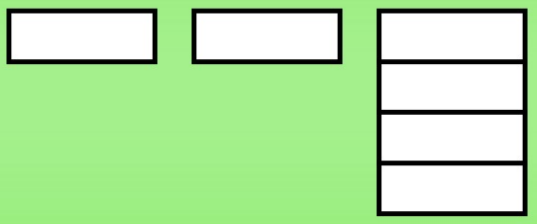
- 同じ長さ 型の異なるベクトル
- A[1,] 1行目　A[,1] 1列目
- colname() 列の名前
- rownames() 行の名前

リスト

```
> A <- matrix(c(1,2,3,4),ncol=2)
> eigen(A)
eigen() decomposition
$values
[1]  5.3722813 -0.3722813
$vectors
           [,1]       [,2]
[1,] -0.5657675 -0.9093767
[2,] -0.8245648  0.415973
```

リスト

list（要素1，　要素2，　要素3）



- 異なる要素の集まり
- t1 [[1]]　　# 1番目の要素
- t1$科目

データフレーム

```
> name <-  c(“A”,  “B”, “C”, “D”)
> h  <- c(148, 152, 154, 158)
> w <- c(52, 54, 56, 58)
> df  <- data.frame(name, h, w)
> df
   name       h       w
1       A     148      52
2       B     152      54
3       C     154      56
4       D     158      58
> names(df)
[1] “name”  “h”  “w”
```

データフレーム

data.frame

(要素1,　要素2,　要素3)

- 同じ長さ、型の異なるベクトル
- df[1 , 1:3]
- df$h

D[行番号,　列番号]

1:3 は c(1, 2, 3)

主なデータの型と構造

データの型	説　明	例
integer	整 数	1L
double	実数(倍精度浮動小数)	3.4、5e-10
character	文字列	“A”, “B”
logical	論理値	TRUE、FALSE

構　造	名　前	次　元	データ型の種類
vector	ベクトル	1次元	1種類
matrix	行列	2次元	1種類
data.frame	データフレーム	2次元	複 数
list	リスト	1次元	複 数

関数とパッケージ

- Rでは関数を使って色々な処理をする
- あらかじめ多くの関数が用意されている
- ？関数名とするとヘルプを表示する
- 同じ関数でも引数が違うと異なる振る舞いをする（多態性）
- 関数名(引数，追加の指示1，追加の指示2,....)

- Rには多くのパッケージがあり 機能を追加できる
- パッケージは関数 データ ドキュメントからなる
- 異なるパッケージで関数の名前が重複していることもある
- パッケージを指定して使う場合には パッケージ名::関数名　とする

パッケージの利用

```
> install.packages("tidyverse")
> install.packages("magrittr")
> library(tidyverse)
```

パッケージ

- 便利な機能や解析のための関数など
- インターネットに接続している環境で行うとファイルを読み込んでくれる

利用するときは library()　とする

tidyverse（複数のパッケージ）

- データフレーム拡張　tibble
- データフレーム変形　tidyr
- データフレーム変形　dplyr
- ファイル読み込み　readr
- グラフ作成　ggplot2

など

magrittr　パイプ演算など

ファイルの読み込み

```
> a01 <- read.csv("c0302.csv")
> a02 <- read_csv("c0302.csv")
```

```
name1,C,D
B1,148,52
B2,152,54
B3,154,56
B4,158,58
B5,163,60
```

read.csv：読み込まれた後の形は　data.frame

多くの場合に1列目にある名前などを行名として使うことができる

列名がない場合　header = FALSE

read_csv：読み込まれた後の形は tibble （データフレームの拡張）

行名は使わない　1行目は各列の名前

列名がない場合　col_names = FALSE

パイプ処理

```
# y <- f(x)
# z <- g(y)
# w <- h(z)
#を行う
# x <- f(x)
# h ( g( f(x) ) )
>     x %>% f() %>%
        g() %>% h()
# とすることができる
```

パイプ演算子

- base |>
- tidyr %>%

- magrittr %<>%

  # y <- x %>% f()

  を

  # x %<>% f ()

とする

データフレーム処理(1)

```
> x %>% filter(A>25) %>%
  select(c( id, B) )
```

id	A	B
1	10	a
2	20	b
3	30	c
4	20	b
5	20	c
6	30	a

id	B
3	c
6	a

```
> df5  %>% mutate( Dv = SA - mean(SA) )
```

id	SA
1	80
2	70
3	60
4	50

id	SA	Dv
1	80	15
2	70	5
3	60	-5
4	50	-15

2つのデータフレームの結合

2つのデータフレームを結合
left_join　right_join
inner_join　full_join

```
> d1 %>% left_join( d2, by="id" )
> d1 %>% right_join( d2, by="id" )
> d1 %>% inner_join( d2, by="id" )
> d1 %>% full_join( d2, by="id" )
```

id	SB
1	80
2	70
3	60
4	50
5	40

id	SC
1	75
3	65
5	55
6	45
7	35

id	SB	SC
1	80	75
2	70	NA
3	60	65
4	50	NA
5	40	55

id	SB	SC
1	80	75
3	60	65
5	40	55
6	NA	45
7	NA	35

id	SB	SC
1	80	75
3	60	65
5	40	55

id	SB	SC
1	80	75
2	70	NA
3	60	65
4	50	NA
5	40	55
6	NA	45
7	NA	35

データフレーム処理(2)

```
> df3  %>%pivot_longer(  c(SD,SE),
   names_to="subject",
   values_to="score" )
```

```
> df4 %>% pivot_wider(
 names_from = subject,  values_from=score )
```

id	SD	SE
1	80	75
2	70	65
3	60	55
4	50	45

longer →

← wider

id	subject	score
1	SD	80
1	SE	75
2	SD	70
2	SE	65
3	SD	60
3	SE	55
4	SD	50
4	SE	45

演　習

```
> df_suba <-read_csv("subA.csv")
> df_subb <-read_csv("subB.csv")
> df_subc <-read_csv("subC.csv")
> df_sub <-  df_suba %>% full_join(df_subb) %>%   full_join(df_subc)
> df_sub %>% select( c( subA,  subB  ) )
> df_long <- df_sub %>%  pivot_longer(c(subA,subB,subC) ,
      names_to =“subject”, values_to=“score” )
> df_long %>%  pivot_wider(names_from =subject, values_from=score )
> df_long %>% filter( is.na( score ) )
```