

# Rの基本～ベクトル～

Rによるデータサイエンス(2)

# ベクトル

- 複数の値を1次元(1列)に収納できる型
- 作り方1：要素を列挙して作る

```
> a <- c(1,2,3)
> a
[1] 1 2 3
```

- 作り方2：範囲を指定して作る

```
> b <- 1:5
> b
[1] 1 2 3 4 5
```

# ベクトル

- 作り方3：範囲と分割数を指定して作成

```
> c <- seq(-2, 2, length=11)
> c
[1] -2.0 -1.6 -1.2 -0.8 -0.4  0.0  0.4  0.8  1.2  1.6  2.0
```

- 作り方4：範囲と間隔を指定して作成

```
> d <- seq(-pi, pi, by = 0.5)
> d
[1] -3.1415927 -2.6415927 -2.1415927 -1.6415927 -1.1415927 -
[10]  1.3584073  1.8584073  2.3584073  2.8584073
```

# ベクトルの利用

- 複数の点に対する関数の値を計算

```
> x <- seq(-pi, pi, by = 0.01)
> y <- sin(x)
> y
 [1] -1.224647e-16 -9.999833e-03 -1.999867e-02 -2.999550e-02
 [8] -6.994285e-02 -7.991469e-02 -8.987855e-02 -9.983342e-02
[15] -1.395431e-01 -1.494381e-01 -1.593182e-01 -1.691823e-01
```

- 複数の点をプロット

```
> plot(x,y)
> plot(x,y,type="l")
```

線でプロット



# ベクトルの統計量の計算

- 平均

```
> mean(y)  
[1] 1.725511e-06
```

- 標準偏差

```
> sd(y)  
[1] 0.7072861
```

- 2つのベクトルの相関

```
> cor(x,y)  
[1] 0.7784288
```

# ベクトルの操作

- ソート

```
> a = c(5, 3, 1, 2, 4)
> a
[1] 5 3 1 2 4
> b <- sort(a)
> b
[1] 1 2 3 4 5
```

- ベクトルを逆に並べる

```
> c <- rev(b)
> c
[1] 5 4 3 2 1
```

# ベクトルの要素の抽出

- 位置を指定して抽出

```
> b  
[1] 1 2 3 4 5  
> b[2]  
[1] 2  
> b[3:5]  
[1] 3 4 5
```

- 条件を指定して抽出

```
> b  
[1] 1 2 3 4 5  
> b[b < 4]  
[1] 1 2 3  
> b[1 < b & b < 4]  
[1] 2 3
```