

MODUL 4

MANAJEMEN DATA MENGGUNAKAN COMMENT LINE



CAPAIAN PEMBELAJARAN

1. Praktikan dapat melakukan manajemen data menggunakan Comment line



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. Komputer
2. Software R



DASAR TEORI

R merupakan bahasa pemrograman berorientasi objek. Data dalam R dikenal melalui nama yang merupakan objek. R adalah bahasa yang case sensitive yang berarti penggunaan huruf besar dan huruf kecil sebagai nama objek adalah berbeda. Jadi ABC berbeda dengan abc. Tipe objek dapat berupa skalar, vektor, matriks, dan array. Dalam bentuk struktur data yang lebih kompleks, objek dapat berupa data frame, list, atau fungsi.

A. VEKTOR

Vektor merupakan suatu array atau himpunan bilangan, character atau string, logical value, dan merupakan objek paling dasar yang dikenal dalam R. Pada data vektor harus digunakan mode tunggal pada data, sehingga gabungan dua data atau lebih yang berbeda mode tidak dapat dilakukan kedalam satu objek vektor. Jika ini dilakukan, maka R akan mengubah data ke mode yang lebih umum. Untuk membentuk vektor dapat dilakukan dengan tiga cara:

1. Sekuens

```
> seq (a1,a2, ...)
```
2. Replica

```
> rep(x, n)
```

3. Fungsi c

> c(a1,a2, ...)

4. Fungsi scan

> scan(a1,a2, ...)

Fungsi c efektif untuk data yang jumlahnya sedikit. Untuk data dalam jumlah yang besar, fungsi lain yang dapat digunakan adalah scan. Fungsi scan ini digunakan untuk data dalam jumlah yang besar dan tidak memiliki kepala (*header*) dan data tersimpan dalam suatu berkas teks.

B. MATRIKS

Matriks atau data array dua dimensi adalah salah satu tipe data yang banyak digunakan dalam pemrograman statistik. Proses entry data matriks dilakukan dengan menggunakan fungsi matrix. Argumen yang diperlukan adalah elemen dari matriks, dan argumen optional yaitu banyaknya baris nrow dan banyaknya kolom ncolom. Berikut skrip untuk memasukkan data matrik dengan ukuran banyaknya baris ada n dan banyaknya kolom ada m:

```
> matrix(c(a1,a2,a3,...),nrow= n ,ncol= m)
```

Dimensi, length dan mode dari suatu matriks dapat dilihat dengan menggunakan perintah dim, length, dan mode .

Pada R, dapat pula dilakukan penggabungan satu kolom atau satu baris baru ke dalam matriks lain. Hal ini dapat dilakukan dengan menggunakan perintah **rbind** (untuk menambahkan ke baris) dan **cbind** (untuk menambahkan ke kolom).

Ada beberapa operator yang biasa digunakan untuk operasi matriks dan vektor. Ringkasan dari operator-operator ini dapat dilihat pada tabel berikut :

No	Nama Fungsi/ Operasi	Fungsi R
1	pembentukan matriks	matrix(data, nbaris, nkolom);
2	pembentukan barisan	seq(awal, akhir,kenaikan); seq(awal,akhir,length=n)
3	barisan berpola	rep((el.),replik) atau rep((elemen),each=n)
4	transpose matriks	t(x)
5	determinan matriks	det(x)
6	matriks diagonal	diag(data)

7	diagonal matriks	diag(matriks)
8	terak (<i>trace</i>) matriks	tr(matriks)
9	perkalian matriks	x%*%y
10	inverse matriks	solve(x)
11	jumlah berturutan elemen- elemen yang sudah dikenakan fungsi <i>f</i>	sum(f(x))
12	hasil kali berturu- tan elemen-elemen yang sudah dikenakan fungsi <i>f</i>	prod(f(x))
13	menunjuk elemen ke <i>i</i> vektor v atau elemen ke <i>i, j</i> matriks x	v[i], x[i,j]

C. ARRAY

Array adalah objek bertipe numerik dengan atribut dimensi. Perbedaan antara matriks dan array adalah bahwa array mempunyai dimensi lebih dari dua.

```
>array(c(a1,a2,...),dim=c(n1,n2,n3))
```

D. FRAME

Data frame merupakan objek yang mempunyai bentuk sama dengan matriks, yaitu terdiri atas baris dan kolom. Perbedaannya adalah data frame dapat terdiri atas mode data yang berbeda untuk setiap kolomnya. Misalkan saja, kolom pertama adalah numeric, kolom kedua adalah string/character, dan kolom ketiga adalah logical. Objek data frame dapat dibuat dengan menggunakan perintah.

```
>data.frame(c(a1,a2,a2,..), c(b1,b2,b3,...))
```

E. LIST

List (daftar) dalam R adalah objek yang terdiri dari koleksi terurut suatu objek yang disebut sebagai komponen. Suatu komponen dalam list tidak harus sejenis atau mode yang sama. Sebagai contoh, suatu list dapat terdiri dari vektor numerik, nilai logika, matriks, vektor kompleks, array karakter, fungsi, dan lain lain.

Penulisan berikut adalah contoh membuat suatu list: List (daftar) baru dapat dibentuk dari objek yang sudah ada dengan menggunakan fungsi `list()`. Cara penulisannya adalah sebagai berikut:

```
> list(name_1=object_1, ..., name_m=object_m)
```

Perintah tersebut akan membentuk list dengan m komponen dan menggunakan `object_1`, ..., `object_m` serta memberikan nama masing-masing dengan argument `names`. Jika nama dihilangkan, komponennya hanya berupa bilangan



PRAKTIK

A. VEKTOR

1. Memasukkan data dengan perintah `seq`

```
> seq(0,3,by=0.5) #Memasukkan data dari 0 sampai 3 dengan penambahan 0.5
```

```
[1] 0.0 0.5 1.0 1.5 2.0 2.5 3.0
```

```
> seq(0,2,length=5) #Memasukkan data dari 0 sampai 3 dengan panjang 5
```

```
[1] 0.0 0.5 1.0 1.5 2.0
```

2. Memasukkan data dengan perintah `rep`

```
> rep(c("A","B","C"),5) # Memasukkan data ("A","B","C"), diulang sebanyak 5 kali
```

```
[1] "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C"
```

```
> rep(1:3,each=5) # Memasukkan data 1 sampai 3, masing-masing sebanyak 5 kali
```

```
[1] 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3
```

```
> rep(1:3,5) # Memasukkan data 1 sampai 3, diulang sebanyak 5 kali
```

```
[1] 1 2 3 1 2 3 1 2 3 1 2 3 1 2 3
```

```
> rep(1:3,each=2,5) # Memasukkan data 1 sampai 3, masing-masing 2 kali, diulang  
                    sebanyak 5 kali
```

```
[1] 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
```

3. Memasukkan data dengan menggunakan fungsi `c`.

```
>x <- c(2,5,6,10,9) # memasukkan data dan simpan pada variabel x
```

```
>x # melihat nilai-nilai variabel x
```

```
[1] 2 5 6 10 9
```

```
>x[1] # mengakses elemen ke-1 dari x
```

```
[1] 2
```

```
>x[3] # mengakses elemen ke-3 dari x
```

```
[1] 6
```

```
>x[2:4] # mengakses elemen ke-2 s.d ke 4 dari x
```

```
[1] 5 6 10
```

```
>x[-1] # mengakses x, tetapi tanpa elemen ke-1
```

```
[1] 5 6 10 9
```

```
>x[-c(2,4)] # mengakses x, tetapi tanpa elemen ke 2 dan 4
```

```
[1] 2 6 9
```

```
>x[-(1:3)] # mengakses x, tetapi tanpa elemen ke 1 s.d. 3
```

```
[1] 10 9
```

```
>length(x) # menghitung banyaknya data
```

```
[1] 5
```

4. Memasukkan data dengan fungsi scan.

Data dalam jumlah yang besar dan tidak memiliki kepala (*header*) dan data tersimpan dalam suatu berkas teks. Sebagai contoh berikut ini adalah data lebar sayap serangga rumah tersimpan dalam berkas `serangga.txt` dengan bentuk seperti berikut.

```
3.6 3.7 3.8 3.8 3.9 3.9 4.0 4.0 4.0 4.0
4.0 4.1 4.3 4.5 4.5 4.6 4.3 4.4 4.5 4.6
```

Memasukkan data serangga dapat dilakukan dengan perintah berikut.

```
>serangga <- scan("serangga.txt") # memasukkan data Read 20 items
```

```
># data serangga.txt ini tersimpan pada direktori kerja Anda
```

```
># jika tidak demikian Anda harus mengisi lokasi file
```

5. Memasukkan data dengan fungsi `read.table`

Untuk data yang berisi kepala, kita dapat menggunakan fungsi `read.table`. Berikut adalah data waktu tunggu antar erupsi dan lama erupsi simpan data tersebut sebagai `erupsi.txt`.

erupsi	waktu
3	79
1	54
3	74
2	62
4	85

Untuk memasukkan data dengan `read.table`:

```
>erupsi <- read.table("erupsi.txt",header=T)
># header=T berarti mengikuti kepala, jika tidak ingin
#gunakan header=F
>erupsi
      erupsi waktu
        3     79
        1     54
        3     74
        2     62
        4     85
```

B. MATRIKS

1. `> x<-matrix(c(4,5,3,6),4,1)`

```
> x
      [,1]
[1,]    4
[2,]    5
[3,]    3
[4,]    6
```

2. `> 2*x`

```
      [,1]
[1,]     8
[2,]    10
[3,]     6
```

```
[4,]      12
```

C. ARRAY

1. `> x <- array(1:20, dim=c(4,5))` #Perintah tersebut akan membentuk array berukuran 4 x 5

```
> x
```

```
      [,1] [,2] [,3] [,4] [,5]
```

```
 [1,] 1  5  9 13 17
```

```
 [2,] 2  6 10 14 18
```

```
 [3,] 3  7 11 15 19
```

```
 [4,] 4  8 12 16 20
```

D. FRAME

1. `> x = data.frame(nama=c("A", "B", "C", "D"), berat=c(45, 65, 70, 57))`
`> x`

```
      nama berat
1     A     45
2     B     65
3     C     70
4     D     57
```

2. `> nama <- c("Ana", "Banu", "Cici", "Dido", "Erik")`
`> tahun <- c(1992, 1995, 1993, 1999, 1994)`
`> lahir <- data.frame(nama, tahun)`
`> lahir`

```
      nama tahun
1     Ana  1992
2    Banu  1995
3     Cici  1993
4     Dido  1999
5     Erik  1994
```

E. LIST

```
1. > Lst <- list(name="Fred", wife="Mary", no.children=3, child.ages = c(4,7,9))
> Lst
$name
[1] "Fred"
$wife
[1] "Mary"
$no.children
[1] 3
$child.ages
[1] 4 7 9
```



LATIHAN

1. Berikut ini data penjualan pada bulan April di toko Murah

Jenis barang	Jumlah penjualan
Alat tulis	48
Kosmetik	53
Sabun mandi	48
Mi instan	52
Minyak goreng	46
Gula	50
Minuman	61
Obat	46
Snack	70

- Entry data tersebut menggunakan Frame
- simpan data dengan nama Jual
- Tampilkan data

2. Diketahui matrik $A = \begin{bmatrix} 2 & 1 \\ 4 & 2 \\ 1 & 5 \end{bmatrix}$ dan $B = \begin{bmatrix} 1 & 3 \\ 1 & 2 \\ 1 & 4 \end{bmatrix}$ tentukan :

- $2A$
- $A + B$
- Transpose B



TUGAS

Perhatikan data Pre tes dan post tes berikutberikut

PRE	POST
40	48
35	53
30	48
60	52
45	46
50	50
45	61
46	46
67	70

- Entry data tersebut dengan menggunakan MSEXcel simpan dengan nama NILAI
- Dengan menggunakan fungsi read.table baca data NILAI.xls
- Tampilkan data set tersebut.



REFERENSI

PUSTAKA :

- [1] John Verzani, "Using R for Introductory Statistics," Second Edition, CUNY/College of Staten Island New York, USA, 2014.
- [2] Emmanuel Paradis, "R for Beginners",
- [3] Suhartono,"Analisis Data Statistik dengan R", Graha Ilmu, Yogyakarta, 2009
- [4] W. John Braun and Duncan J.Murdoch, "A First Course in Statistical Programming with R", Second Edition
- [5] Tony Fischetti "Data Analysis with R" Packt Publishing Ltd., Birmingham, 2015