



STK 573

Metode Grafik untuk Analisis dan Penyajian Data

Pertemuan 1

Tim Dosen:

Prof. Dr. Khairil Anwar Notodiputro

Dr. Ir. Aji Hamim Wigena

Dr. Agus M Soleh

PENDAHULUAN



Pendahuluan

- Apa R?
 - R adalah implementasi sebuah lingkungan komputasi dan pemrograman bahasa statistika
- Bedanya dengan S-Plus:
 - R: free, open-source, Windows/Macintos/Unix/Linux
 - S-Plus: Commercial, Windows/Unix/Linux
- Bagaimana mendapatkannya?
 - www.r-project.org



Perintah (*Command*)

- Dua Prompt:
 - > → R siap menerima perintah
 - + → Perintah yang ditulis belum lengkap
- Tuliskan ekspresi pada prompt dan tekan ENTER umumnya akan mencetak output
 - Kecuali ekspresi penyimpanan data
- Beberapa perintah dapat *disubmit* dalam satu prompt dgn dipisahkan titik koma (;)
- Beberapa perintah dapat dikelompokkan dengan { }



Fitur Dasar R

- Kalkulator
 - R dapat digunakan sebagai kalkulator
 - Operator : + , - , * , / , ^ , %% , %/ %
- Case sensitif
- *Command Line Editing*
 - Panah atas dan bawah pada keyboard digunakan untuk memanggil perintah sebelumnya
 - Panah kiri dan kanan untuk mengedit



Ekspresi dan Assignment

- Perintah S berupa ekspresi dan assignment
 - Ekpresi : dievaluasi kemudian dicetak ke layar
> 1 - pi + exp(1.7)
[1] 3.332355
 - Assignment : dievaluasi kemudian disimpan ke objek (menggunakan '<-', '->', '=', '<<-')
- > a <- 6

Penamaan Objek

- Menggunakan huruf alfabet (A-Z, a-z), angka (0-9) dan titik (.)
- Dimulai dengan huruf alfabet atau titik
- Underscore (_) tidak diimplementasikan untuk setiap engine S → hindari (gunakan titik sebagai pemisah)
- Hindari penamaan yg sama dengan fungsi
 - Contoh : c, q, s, t, diff, length, mean, dll
- Terdapat *reserved word* yg tidak dapat digunakan → FALSE, TRUE, inf, NA, NaN, NULL, break, else, for , dll



OBJEK DATA

- Objek data dasar dalam bahasa S:
 - Vektor
 - Matriks
 - Array
 - Factor
 - List
 - Data Frame

OBJEK VEKTOR

- Vektor merupakan tipe sederhana dari objek data dalam S dalam hampir digunakan oleh semua objek data.
- Semua elemen memiliki mode yang sama.
- Untuk menciptakan vektor yang sederhana dapat menggunakan perintah concatenate (yaitu c)
 - Teladan :
 - `x <- c(1.4, 13.2, 10.9)`
 - `y <- c("sd", "smp", "sma", "pt")`



OBJEK VEKTOR

- Perintah-perintah untuk membuat vektor selain perintah `c`:
 - `scan`
 - `rep`
 - `seq`
 - `vector`
 - `as.vector`
 - operator :
 - `unlist` (lihat `list` di bawah)



AKSES TERHADAP VEKTOR

- Jika x adalah objek vektor, maka akses terhadap elemen ke- i dilakukan melalui $x[i]$
- Index i berupa bilangan integer atau nama dari index atau elemen logical dari objek



OBJEK MATRIKS

- Matriks merupakan objek data yang banyak digunakan dalam analisis statistika
- Matriks pada bahasa S umumnya dibuat melalui vektor.
- Perintah-perintah untuk membuat matriks/mengkonversi vektor menjadi matriks:
 - matrix
 - dim
 - rbind
 - cbind
 - as.matrix



AKSES TERHADAP MATRIKS

- Jika `mat` adalah objek matriks, maka akses terhadap:
 - individual element : `mat[m,n]` atau `mat[i]`
 - baris : `mat[m,]`
 - kolom : `mat[,n]`
 - `m`, `n` dan `i` adalah bilangan integer, nama baris/kolom, logical



OBJEK ARRAY

- Matriks merupakan bentuk khusus dari Array. Perbedaan diantara keduanya adalah dimensinya.
- Matriks adalah Array berdimensi dua.
- Perintah untuk membuat Array adalah:
 - array
 - dim
 - as.array



AKSES TERHADAP ARRAY

- Akses terhadap objek array, mirip dengan matriks
 - menggunakan operator []
- Output yang diperoleh bisa berupa vektor, matriks atau array dengan dimensi yang sama atau lebih rendah



OBJEK FACTOR

- Merupakan tipe khusus vektor dengan elemen data kategori
- Faktor diprint tanpa diapit oleh *quotes* ("")
→ memiliki metode print khusus
- Pada dasarnya faktor adalah vektor yang mengindikasikan taraf (level) dari peubah kategori



OBJEK FACTOR

- Perintah-perintah untuk membuat faktor:
 - factor
 - ordered
 - as.factor
 - as.ordered



OBJEK LIST

- List merupakan objek data yang elemen didalamnya boleh memiliki mode atau objek yang berbeda.
- Perintah-perintah untuk membuat list:
 - list
 - as.list
- Umumnya list digunakan untuk menyimpan hasil suatu fungsi
- Perintah unlist akan membuat list menjadi vektor.



AKSES TERHADAP LIST

- Akses terhadap list dapat dilakukan melalui beberapa cara.
 - By name : menggunakan operator \$
 - By position : menggunakan operator [[]]
 - Sebagai vektor : menggunakan operator []



OBJEK DATA FRAME

- Merupakan tipe khusus list yang digunakan untuk menyimpan data dari berbagai macam tipe dalam bentuk matriks.
- Perintah-perintah untuk membuat data frames:
 - `data.frame`
 - `read.table`
 - `as.data.frame`



AKSES TERHADAP DATA FRAME

- Akses terhadap data frame berlaku seperti matriks dan list
 - matriks : `dataolah[m,n]`
 - list : `dataolah$status` atau `dataolah[[2]]`
 - Menggunakan fungsi `attach` dan `detach` -> memperlakukan sebagai vektor



Aritmetika dan Struktur Kendali



Aritmetika

- Aritmetika berhubungan dengan:
 - Operand
 - Operator
 - Fungsi
- Operand :
 - Konstanta
contoh : 10 -1.5 1.5e10
 - Objek data
contoh : x y panjang



Aritmetika

- Operator:

Operator	Use
\$	component selection
[[]	subscripts, elements
^	exponentiation
-	unary minus
:	sequence operator
%% %/% %*%	modulus, integer divide, matrix multiply
* /	multiply, divide
+ -	add, subtract
<> <= >= == !=	comparison
!	not
& &&	and, or
~	formulas
<<- -> <- =	assignments



Aritmetika

- Fungsi-fungsi standar yang sering digunakan
 - Fungsi konversi ke Integer : round, trunc, floor, ceiling
 - Fungsi umum : abs, sign, log, log10, sqrt, exp, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh
 - Fungsi jumlah dan product dari vektor: sum, prod, cumsum, cumprod
 - Fungsi Maksimum dan Minimum : max, min, cummax, cummin
 - Fungsi range(x) menghitung $c(\min(x), \max(x))$
 - Fungsi untuk mengurutkan : sort
 - Fungsi untuk membalikan : rev



Aritmetika

- Missing value
 - Elemen dari vektor yang tidak diketahui atau tidak tersedia direpresentasikan oleh nilai khusus : NA
 - Operasi dalam NA akan menghasilkan NA
 - `x <- c(1.5, 2.3, NA)`
 - `mean(x)`
 - `[1] NA`
 - Beberapa fungsi memiliki argumen `na.rm` atau yg mirip untuk tidak menyertakan NA dalam komputasi
 - Bedakan NA dengan NaN (Not a Number)



Operasi pada Objek Data

- Operasi dalam Vektor
 - Menggunakan element by element
 - Aturan Recycle
 - Operasi dengan vektor yang memiliki panjang berbeda
 - Vektor dengan elemen sedikit akan diulang mengikuti vektor yang memiliki elemen paling banyak

Operasi pada Objek Data

- Operasi dalam Matriks
 - Transpose : menggunakan fungsi `t`
 - Perkalian matriks : `%*%`
 - Vektor akan dipertimbangkan sebagai matriks
 - $x^t x \rightarrow x \%* \% x$
 - $x x^t \rightarrow x \%o \% x$
 - Fungsi `diag` menghasilkan matriks diagonal atau mengambil diagonal matriks tergantung dari argumen



Operasi pada Objek Data

– Beberapa fungsi matriks:

- `solve` --> menghasilkan solusi sistem persamaan linier atau matriks invers
- `eigen` --> menghasilkan nilai akar ciri dan vektor ciri



Operasi pada Objek Data

- Operasi dalam Karakter
 - Karakter dan string diapit oleh quotes (" --- ")
 - Fungsi `nchar` menghitung banyaknya karakter dalam string
 - Fungsi `paste` menciptakan/menjadikan mode objek lain menjadi karakter
 - Fungsi `substr/substring` digunakan untuk mengambil bagian dalam string



Struktur Kendali

- Eksekusi bersyarat
if (kondisi)
 perintah-perintah jika kondisi T
else
 perintah-perintah jika kondisi=F
- ifelse(test, nilai jika T, nilai jika F)
- Fungsi switch beberapa diantaranya akan lebih baik dibanding menggunakan if tersarang.



Struktur Kendali

- Pengulangan
 - for (objek i_n sekuens) perintah
 - while (condition) perintah
 - repeat perintah (untuk menghentikan gunakan perintah break)

Manajemen Objek Data FRAME



Menciptakan Peubah Baru

- Lakukan seperti menciptakan vektor dengan menggunakan index atau operator seleksi
 - `dt$baru <- ekspresi`



Subsetting Data

- Hal penting untuk Subsetting data adalah membuat vektor logical seperti yg diinginkan.
- Harus dapat menterjemahkan idea rumit ke dalam vektor logic
- Fungsi yang digunakan :
==, !=, >, >=, <, <=, %in%, duplicated,
is.na, is.null, is.numeric, dll...

Ilustrasi Subsetting Data

```
> a
  gender v1 v2 v3 v4 v5 v6 v7 v8 v9 v10
1      f  8  9  5  9  9  9  5  9 11  8
2      f 15  8  9  5 10  8 10 12  9 15
3      f  8 13 12  6  7  9 14 12 12  9
4      m 14  9  8 NA 11 10  6 11 11  8
5      m  8  2  9 16  8 10  8  9  8  9
6      f 10  6 10  9 10 10  9  7  7 11
7      m  9  9  7 13  9 12 10  9 11  7
8      m  9 10  8 10  5 10  6 12  6  8
9      f 10 13  6  7 10 12 13  9  6  9
10     m 12 10  9  8 17  7  9 10  7  7
```

Original dataset, a

Misalkan diinginkan dataset baru yang memenuhi females di atas 7 pada v1 tetapi di bawah 10 pada v10, dan males jika non-missing pada v4

```
> myindex <- (a$gender=='f' & a$v1>7 & a$v10<10) | (a$gender=='m' & !is.na(a$v4))
> a.new <- a[myindex,]
> a.new
  gender v1 v2 v3 v4 v5 v6 v7 v8 v9 v10
1      f  8  9  5  9  9  9  5  9 11  8
3      f  8 13 12  6  7  9 14 12 12  9
5      m  8  2  9 16  8 10  8  9  8  9
7      m  9  9  7 13  9 12 10  9 11  7
8      m  9 10  8 10  5 10  6 12  6  8
9      f 10 13  6  7 10 12 13  9  6  9
10     m 12 10  9  8 17  7  9 10  7  7
```

Sorting Data

- Langkah 1 – Buat vektor numerik yang terurut dari data yang akan diurutkan
- Langkah 2 – Gunakan vektor ini sebagai index
- Fungsi yang berguna: `order()`, `sort()`, `which()`, `rev()`, `unique()`



Ilustrasi Sorting Data

Dataframe:

```
> a
  gender v1 v2 v3 v4 v5 v6 v7 v8 v9 v10
1      f  7  7 12 10 12  9  7  9 13  8
2      m  7 11 14  4  4 11  8  9 11  8
3      f  9  8  8  8 12  6  8 10 13  4
4      m 10  9  7  9  4 10 12  9 10  8
5      f  8  9 11  7  9 10 11  9 15 10
6      f  5 10  8  6  9  8  8 12  3  7
7      f  7 11  9  6 11 12  9 11 11 10
8      m  7 10  6  9 13  5  9 10  9 11
9      m  7 10 10  8  6  7  8  8  8 11
10     m 12  6 13  3  7 11  8 10  8  4
```

Katakan akan diurutkan berdasarkan gender kemudian v1 jika male dan v2 apabila female secara ascending?

Ilustrasi Sorting

- Langkah 1 - Buat vektor numerik yang terurut dari data yang akan diurutkan

```
newvec <- (a$gender=='m')*a$v1 + (a$gender=='f')*a$v2  
myord <- order(a$gender, newvec)
```

- Step 2 - Gunakan vektor ini sebagai index

```
aZ <- a[myord,]
```

```
> aZ  
  gender v1 v2 v3 v4 v5 v6 v7 v8 v9 v10  
1      f  7  7 12 10 12  9  7  9 13  8  
3      f  9  8  8  8 12  6  8 10 13  4  
5      f  8  9 11  7  9 10 11  9 15 10  
6      f  5 10  8  6  9  8  8 12  3  7  
7      f  7 11  9  6 11 12  9 11 11 10  
2      m  7 11 14  4  4 11  8  9 11  8  
8      m  7 10  6  9 13  5  9 10  9 11  
9      m  7 10 10  8  6  7  8  8  8 11  
4      m 10  9  7  9  4 10 12  9 10  8  
10     m 12  6 13  3  7 11  8 10  8  4
```



Selesai