



# STK371

# Komputasi Statistik 1

*Materi 1. Pengantar R dan Objek Data*

<https://www.stat.ipb.ac.id/agusms/index.php/stk371/>

## Ruang Lingkup Materi

- Pengantar Program R
- Pemasukan Data (Tipe Dasar Objek)
- Pengelolaan Data (Modifikasi, merging, combining, subsetting, recoding, import & ekspor) secara native
- Pengelolaan Data (Modifikasi, merging, combining, subsetting, recoding, import & ekspor) menggunakan SQL

## 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](http://www.r-project.org)

## Mengapa menggunakan S

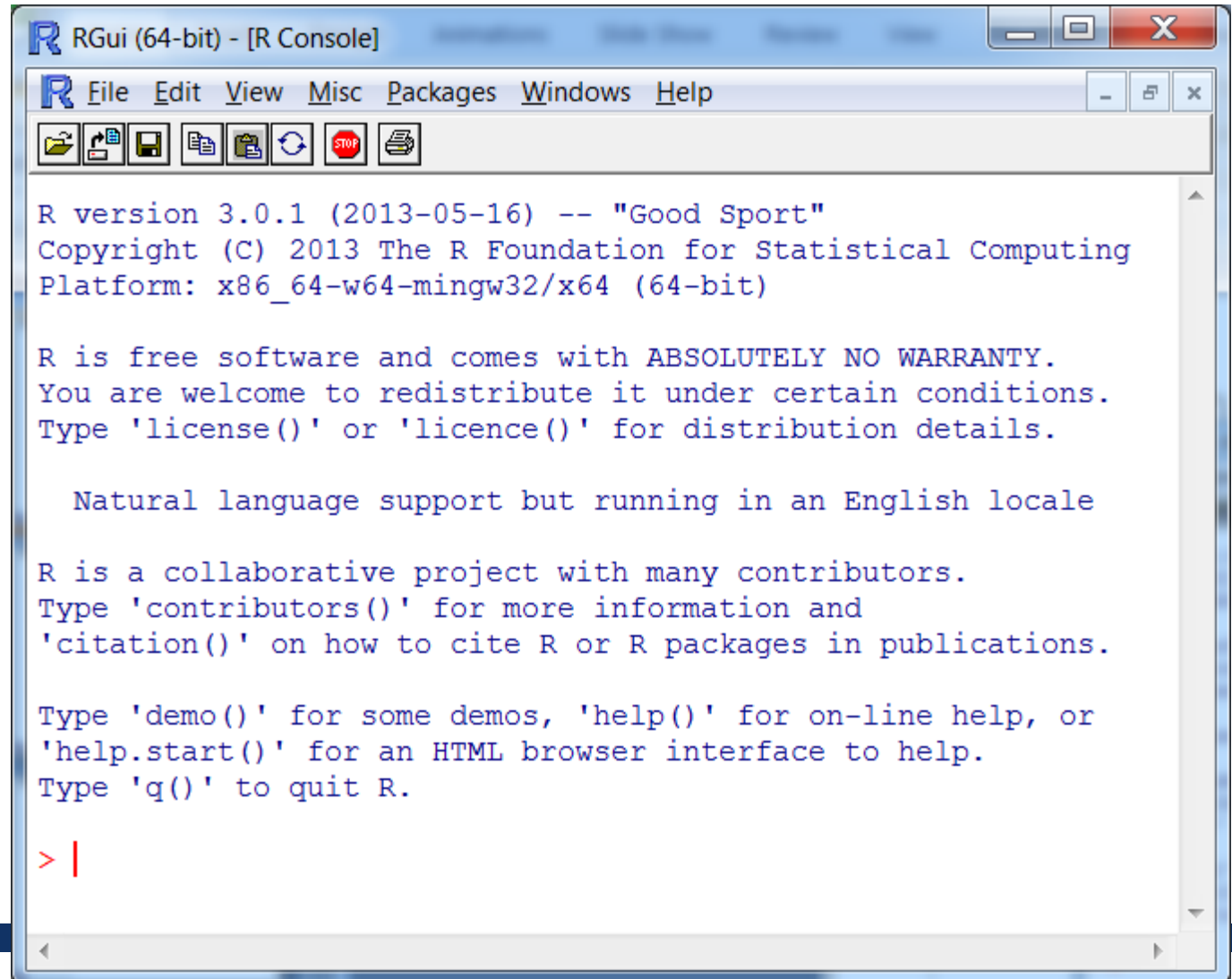
- S telah menjadi software statistika standar diantara statistisi dunia. → metode baru banyak diimplementasikan
- Telah banyak dikembangkan fungsi-fungsi statistika
- S menciptakan grafik yang lebih baik
- S mudah digunakan

## Referensi

- “An Introduction to R”; Venables, Smith dan R Development Core Team.
- Kleinman K, Horton NJ. 2010. SAS and R: Data Management, Statistical Analysis and Graphics. CRC Press, New York.
- Crawley MJ. 2005. Statistics: An Introduction using R. John Wiley & Sons, London
- Faraway JJ. 200. Linier Models with R. Chapman & Hall/CRC, Washington, DC.

# Lingkungan R

- Menjalankan R
  - Windows : klik icon R
  - Linux : tulis perintah R
- Mengakhiri R
  - Gunakan fungsi `q()`



```
R version 3.0.1 (2013-05-16) -- "Good Sport"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

## 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
  - Ekspresi : dievaluasi kemudian dicetak ke layar  

```
> 1 - pi + exp(1.7)
```

```
[1] 3.332355
```
  - Assignment : dievaluasi kemudian disimpan ke objek (menggunakan '<-', '->', '=', '<<-')

## 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, t, diff, length, mean, dll
- Terdapat *reserved word* yg tidak dapat digunakan → FALSE, TRUE, inf, NA, NaN, NULL, break, else, for , dll



# OBJEK DALAM BAHASA S



# OBJEK

- Segala sesuatu dalam S adalah objek.  
Data, fungsi, operator dll.
- Karakteristik setiap objek dilihat melalui *mode* dan *length*  
Objek data --> mode *numeric*, *complex*, *character*, dan *logical*  
Objek bahasa --> mode *function*, *expression*, *list*, dll

# OBJEK

- Objek juga memiliki *attribute*

Menjelaskan fitur dari objek

Contoh attribut: class, comment, dim, dimnames, names dll

Gunakan perintah attribute/attr untuk melihat atau merubah atribut

- Teladan :

```
# menciptakan matriks 2 x 5
```

```
x <- 1:10
```

```
attr(x,"dim") <- c(2, 5)
```

# OBJEK

- Untuk menciptakan objek sekaligus atribut gunakan structure

Teladan : `structure(1:6, dim = 2:3)`

- Mode penyimpanan untuk objek numeric (`storage.mode`)

integer, real (single), double precision

Biasanya seluruh objek numeric menggunakan mode double precision

# 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

- Teladan :

```
> x <- c(1.5, 2.3, NA, 5.4, 3.3)
```

```
> names(x) <- c('a', 'b', 'c', 'd', 'e')
```

```
> x
```

```
a      b      c      d      e
```

```
1.5  2.3  NA  5.4  3.3
```

```
> x[2]
```

```
b
```

```
2.3
```

```
> x[1:3]  
> x[c(2,4)]  
> x["d"]  
> x[!is.na(x)]  
> x[x>3]  
> x[x>3 & !is.na(x)]
```

- Bagaimana jika  $i$  negatif?

>  $x[-5]$

>  $x[-(1:3)]$

>  $x[-c(2, 4)]$

# 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
- Bagaimana jika  $m, n, i$  adalah negatif?

# 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



# AKSES TERHADAP FACTOR

- Sama seperti vektor

# OBJEK LIST

- List merupakan objek data yang elemen didalamnya boleh memiliki mode 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]]`
- Akses lain sebagai vektor :  
`attach(namadataframe)`  
`detach(namadataframe)`





# Selesai

*Next: Aritmetika dan Struktur Kendali*