

# PEMOGRAMAN **R** DASAR

R adalah bahasa pemrograman dan perangkat lunak untuk analisis statistika dan grafik yang awalnya dibuat oleh Ross Ihaka dan Robert Gentleman dan akhirnya lanjutkan oleh R Development Core Team.

R sangat baik digunakan untuk permodelan linier dan nonlinier, uji statistik parametrik dan nonparametrik, klasifikasi, klasterisasi, dan lainnya dan juga baik dalam visualisasi data. R dan Python saat ini adalah 2 platform yang paling populer untuk Data Science. Buku ini hadir sebagai salah satu petunjuk awal dalam Pemrograman R Dasar yang sangat bermanfaat bagi pembaca.

ISBN 978-623-6636-50-3



9 786236 636503



Kamp. Srigangga, Tawugalih Praya  
Lombok Nusa Tenggara Barat  
Email. aswajahamdan@gmail.com  
WhatsApp. 08978119101

Pemrograman R Dasar

Titin Agustin Nengsih | Fadhlul Mubarak |  
Vinnny Yuliani Sundara

# PEMOGRAMAN



# DASAR



**Titin Agustin Nengsih  
Fadhlul Mubarak  
Vinnny Yuliani Sundara**



---

# PEMOGRAMAN **R** DASAR

---



---

# PEMOGRAMAN **R** DASAR

---

Penulis:

**Titin Agustin Nengsih**  
**Fadhlul Mubarak**  
**Vinny Yuliani Sundara**



# PEMOGRAMAN R DASAR

Penulis

**Titin Agustin Nengsih**  
**Fadhlul Mubarak**  
**Vinny Yuliani Sundara**

Editor

**A. A. Miftah**

Desain Cover & Penata Isi

Tim Penerbit FP. Aswaja

Cetakan Pertama: Desember 2020

Diterbitkan Oleh:



**Forum Pemuda Aswaja**

Jl. Kamp. Srigangga, Tiwugalih, Praya NTB.

Telp. : 085333011184

E-mail : [aswajahamdan@gmail.com](mailto:aswajahamdan@gmail.com)

ISBN: **978-623-6636-50-3**

Hak Cipta dilindungi undang-undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ke dalam bentuk apapun, secara elektronik maupun mekanis, termasuk fotokopi, merekam, atau dengan teknik perekaman lainnya, tanpa izin tertulis dari Penerbit. Undang-Undang Nomor 19 Tahun 2000 tentang Hak Cipta, Bab XII Ketentuan Pidana, Pasal 72, Ayat (1), (2), dan (6).

## **KATA PENGANTAR**

Alhamdulillah penulisan buku “Pemograman R Dasar” telah berhasil dirampungkan. Buku ini diharapkan dapat membantu analisis data pada pelajar, peneliti, bahkan yang tertarik dalam bidang statistika khususnya yang berminat menggunakan pemograman statistika yang *open source*.

Buku ini sangat sesuai bagi yang ingin belajar pemograman R dari dasar bahkan buku ini juga akan sangat mudah dipahami bagi yang belum pernah menggunakan pemograman statistika. Buku ini akan membantu pada pengguna yang ingin mendapatkan master pemograman R, menginstall pemograman R beserta paketnya, menganalisis menggunakan pemograman R, bahkan menampilkan menu help jika nantinya mengalami kesulitan saat menggunakan R.

**Jambi, November 2020**

**Tim Penulis**



## DAFTAR ISI

**Kata Pengantar** ..... vi

**Daftar Isi** ..... viii

### **BAB 1 PENGENALAN PEMOGRAMAN R**

A. Apa itu Pemograman R? ..... 3

B. Kelebihan Pemograman R ..... 6

C. Kekurangan R..... 12

D. Instalasi Pemograman R dan R Studio ..... 14

E. Karakteristik R ..... 28

F. Kuis..... 29

### **BAB 2 FITUR DASAR R**

A. Objek Data Dasar di R Studio ..... 31

B. Aturan Nama Penyimpanan (*Named Storage*)..... 50

C. Pembacaan Data Eksternal di R ..... 51

D. Ekstrak data di R ..... 53

E. Kuis..... 54

### **BAB 3 STATISTIKA DESKRIPTIF**

A. *Summarization* (ringkasan) Data Kontinu ..... 58

B. *Summarization* (ringkasan) untuk data kategorik ..... 76

C. Kuis..... 77

### **BAB 4 UJI T**

A. Uji T Satu Sampel (*One Sample T-Test*) ..... 79

B. Uji T Dua Sampel (*Two Sample T-Test*) ..... 84

C. Kuis..... 90



**BAB 5 FASILITAS HELF**

A. Mencari Help dari Suatu Perintah Tertentu ..... 93

B. Help Umum ..... 96

**Daftar Pustaka ..... 99**

**Biodata Penulis..... 101**

# PENGENALAN PEMOGRAMAN R

# PENGENALAN PEMOGRAMAN R

## BAB 1

# PENGENALAN PEMOGRAMAN R

### A. Apa itu Pemograman R?

Pemograman R merupakan salah satu paket analisis statistika yang bersifat *open source* atau dapat didapatkan secara gratis di situs [www.r-project.org/](http://www.r-project.org/) atau [cran.rproject.org/](http://cran.rproject.org/). Pemograman R ini termasuk ke dalam bahasa S. Terdapat dua program utama yang ditulis dengan bahasa S, yaitu S-Plus yang dikembangkan secara komersial (salah satu contohnya S+2000) dan R yang dikembangkan melalui konsep *open source*. S-Plus telah dilengkapi dengan menu yang sangat lengkap yang sering disebut sebagai *advanced Grapical User Interface* (GUI), sedangkan R lebih mengandalkan *Command Line Interface* (CLI) dari pada menu.



### **Logo R**

Paket pemrograman R sudah dilengkapi banyak kemampuan untuk menganalisis data, menampilkan grafik, hingga dapat terkoneksi dengan basis data yang terdapat pada internet karena banyak kontributor yang menyumbang paket menu *interface* untuk R. Dewasa ini R populer dipergunakan baik di bidang akademik maupun industri khususnya dibidang sains data.

Pemrograman R merupakan merupakan salah satu *software* gratis yang sangat populer di Indonesia. Kemudahan penggunaan serta banyaknya besarnya dukungan komunitas membuat R menjadi salah satu bahasa pemrograman paling populer di dunia.

Paket yang disediakan untuk analisis statistika dan analisa numerik juga sangat lengkap dan terus bertambah setiap saat. Hal ini membuat R banyak digunakan oleh para analis data.

Pemrograman R merupakan bahasa pemrograman yang digunakan dalam komputasi statistik. Pemrograman ini pertama kali dikembangkan oleh Ross Ihaka dan Robert Gentleman di *University of Auckland* New Zealand yang merupakan akronim dari nama depan kedua pembuatnya. Sebelum R dikenal ada S yang dikembangkan oleh John Chambers dan rekan-rekan dari Bell Laboratories yang memiliki fungsi yang sama untuk komputasi statistik. Hal yang membedakan antara keduanya adalah R merupakan sistem komputasi yang bersifat gratis.

Pemrograman R merupakan aplikasi sistem **statistik** yang kaya. Hal ini disebabkan banyak sekali paket yang dikembangkan oleh pengembang dan komunitas untuk keperluan analisa statistik seperti *linear regression*,

*clustering*, dan *statistical test*. Selain itu, R juga dapat ditambahkan paket - paket lain yang dapat meningkatkan fiturnya.

Software R sangat cocok untuk riset, baik statistik, ekonomi, komputasi numerik dan pemrograman komputer. Karena didukung oleh banyak tenaga ahli dibidangnya, R layak dijadikan suatu perangkat lunak acuan bagi berbagai kalangan, terlebih di kalangan akademik (dosen, mahasiswa). Diantaranya Proses analisis data statistik, dengan tampilan grafik plot yang costumized dan grafik fungsi densitas yang dapat diparalelkan dnegan histogram. Cocok untuk bidang statistika, ekonomi, dan lain lain.

## **B. Kelebihan Pemograman R**

Pemograman R memiliki beberapa kelebihan jika dibandingkan dengan pemograman statistika lainnya. Hal ini juga yang menjadi alasan banyak kalangan statistikawan

mapun non-statistikawan memilihnya sebagai paket pengolahan data statistika diantaranya:

1. Pemograman R termasuk program *open sources* (OSS-R: *Open Source Software-R*) yang multiplatform (tersedia pada sistem operasi Windows, Unix dan Linux) dan dapat diperoleh pada Situs Projek-R. Sebagai open source, skrip programnya dapat diakses, dimodifikasi dan dikembangkan sesuai keperluan dan tingkat kemampuan pengguna. R juga didukung oleh banyak ahli statistika dari berbagai universitas di seluruh dunia.
2. Penggunaan software dapat digunakan kapanpun tanpa terikat oleh masa berakhirnya lisensi atau disebut dengan *Protability*.
3. Pada pemograman R terdapat beberapa koleksi paket program analisis data, yang disebut *library* atau pustaka yang sangat luas seperti statistika deskriptif, regresi, klasifikasi, perancangan percobaan, pemodelan statistika, anova dan multivariat, atau untuk tujuan khusus seperti



Geospasial, Pengolahan Citra (*Image Processing*), bahkan untuk pengembangan *Interface* atau antarmuka grafis (GUI) R itu sendiri.

4. Pemograman R juga memiliki kemampuan pemrograman (bahasa S) dapat dikembangkan secara fleksibel untuk kepentingan khusus yang lebih lanjut.
5. Pada pemograman R terdapat variasi grafik yang sangat banyak dan berkualitas tinggi baik penampilan di layar monitor maupun dalam bentuk cetak di atas kertas. Variasi ini lebih terlihat jika pengguna menggunakan R Studio.
6. Pemograman R termasuk pemrograman yang berorientasi pada objek (*object oriented programming*). Semua hasil, baik berupa variabel, konstanta maupun fungsi, oleh R disimpan dalam bentuk objek. Keuntungannya adalah apabila apa yang telah dikerjakan R saat ini dikemudian hari diperlukan, maka R dapat mengambilnya tanpa harus melakukan perhitungan ulang dari awal. Dengan demikian proses untuk objek yang sama menjadi lebih cepat. Dalam

mencari objek yang diperlukan, mula-mula R akan melihat daftar objek yang masih ada di memori, apabila tidak ditemukan, maka R akan melanjutkan pencariannya ke hardisk yang semuanya dilakukan secara otomatis tanpa dirasakan oleh penggunanya.

7. Pemograman R juga termasuk bahasa terinterpretasi/ interpreted, bukan terkompilasi/ *compiled*. Dalam bahasa terinterpretasi setiap ekspresi/ perintah tunggal dievaluasi dan dieksekusi dengan segera. Sedangkan dalam bahasa kompilasi (C dan Fortran misalnya), maka keseluruhan program harus dikompilasi oleh sebuah kompiler yang menerjemahkan bahasa C atau Fortran tadi ke dalam bahasa mesin. Keunggulan bahasa interpretasi ini adalah fleksibilitasnya untuk dikembangkan secara bertahap, sedangkan kelemahannya adalah memerlukan lebih banyak memori. Namun dengan kemajuan perangkat keras komputer, memori dan kecepatan proses tidak lagi merupakan masalah yang serius.

8. Pemograman R berbasis S yang merupakan bahasa dasar dari paket komersial S-Plus. Kedua bahasa tersebut sangat kompatibel. Hampir semua skrip yang dihasilkan dengan program R akan dapat dijalankan pada paket S-Plus dan sebaliknya. Sebagian besar panduan atau manual pemrograman untuk S-Plus dapat dipergunakan sebagai panduan untuk R.
9. Paket statistik pemrograman R bersifat multiplatform, dengan file instalasi binary/ file tar tersedia untuk sistem operasi Windows, Mac OS, Mac OS X, Linux, Free BSD, NetBSD, iSolaris, AIX, dan HPUX.

R mempunyai karakteristik tersendiri, dimana selalu dimulai dengan prompt “>“ pada console-nya.

Selain kelebihan di atas, R mempunyai beberapa kelebihan dan fitur-fitur yang canggih dan berguna, diantaranya:

1. Efektif dalam pengelolaan data dan fasilitas penyimpanan.  
Ukuran file yang disimpan jauh lebih kecil dibanding software lainnya.
2. Lengkap dalam operator perhitungan array.
3. Lengkap dan terdiri dari koleksi tools statistik yang terintegrasi untuk analisis data, diantaranya, mulai statistik deskriptif, fungsi probabilitas, berbagai macam uji statistik, hingga time series.
4. Tampilan grafik yang menarik dan fleksibel ataupun costumized
5. Dapat dikembangkan sesuai keperluan dan kebutuhan dan sifatnya yang terbuka, setiap orang dapat menambahkan fitur-fitur tambahan dalam bentuk paket ke dalam software R

Selain kelebihan dan kelengkapan fitur-fiturnya, hal yang terpenting lainnya yakni, R bersifat multiplatform, yakni dapat diinstall dan digunakan baik pada sistem operasi Windows, UNIX/LINUX maupun pada Macintosh. Untuk

dua system operasi disebutkan terakhir diperlukan sedikit penyesuaian. Selain kelebihan disebutkan di atas, R didukung oleh komunitas yang secara aktif saling berinteraksi satu sama lain melalui Internet dan didukung oleh manual atau R help yang menyatu pada software R.

R seperti halnya piranti lunak open source lainnya memiliki basis komunitas pengguna yang sangat banyak. Daftar berbagai komunitas R sebagian dapat dilihat di Situs R-evolution. Mereka berkumpul secara rutin dalam pertemuan pengguna R (R meet up) di berbagai negara. Kegiatan tersebut saat ini telah diadakan 127 kota di 31 negara, menurut situs R user group.

### **C. Kekurangan R**

1. Pada pemrograman R interaksi utama bersifat *CLI* (*Command Line Interface*), walaupun saat ini telah dikembangkan paket yang memungkinkan kita berinteraksi dengan R menggunakan *GUI* (*Graphical*

*User Interface*) sederhana menggunakan Paket `R-Commander` yang memiliki fungsi yang terbatas. `R-Commander` sendiri merupakan *GUI* yang diciptakan dengan tujuan untuk keperluan pengajaran sehingga analisis statistik yang disediakan adalah yang klasik. Meskipun terbatas Paket ini berguna jika kita membutuhkan analisis statistik sederhana dengan cara yang sederhana.

2. Meskipun analisis statistika dalam `R` sudah cukup lengkap, namun tidak semua metode statistika telah diimplementasikan ke dalam `R`. Namun karena `R` merupakan *lingua franca* untuk keperluan komputasi statistika modern saat ini, dapat dikatakan ketersediaan fungsi tambahan dalam bentuk Paket hanya masalah waktu saja.

#### D. Instalasi Pemograman R dan R Studio

Pemograman R pada dasarnya berbasis teks atau *command line* sehingga pengguna harus mengetikkan perintah-perintah tertentu dan harus hapal perintah-perintahnya. Setidaknya jika ingin melakukan kegiatan analisa data menggunakan R kita harus selalu siap dengan perintah-perintah yang hendak digunakan sehingga buku manual menjadi sesuatu yang wajib ada saat berkeja dengan R.

Kondisi ini sering kali membingungkan bagi pengguna pemula maupun pengguna mahir yang sudah terbiasa dengan aplikasi statistik lain seperti SAS, SPSS, Minitab, dll. Alasan itulah yang menyebabkan pengembang R membuat berbagai *frontend* untuk R yang berguna untuk memudahkan dalam pengoperasian R.

R Studio merupakan salah satu bentuk *frontend* R yang cukup populer dan nyaman digunakan. Selain nyaman digunakan, R Studio memungkinkan kita melakukan

penulisan laporan menggunakan R markdown atau R Notebook serta membuat berbagai bentuk project seperti shyni, dll.

Pada R studio juga memungkinkan kita mengatur *working directory* tanpa perlu mengetikkan sintaks pada Commander, yang diperlukan hanya memilihnya di menu R Studio. Selain itu, kita juga dapat meng-import file berisikan data tanpa perlu mengetikkan pada Commander dengan cara memilih pada menu Environment.

## 1. Download dan Install R Studio

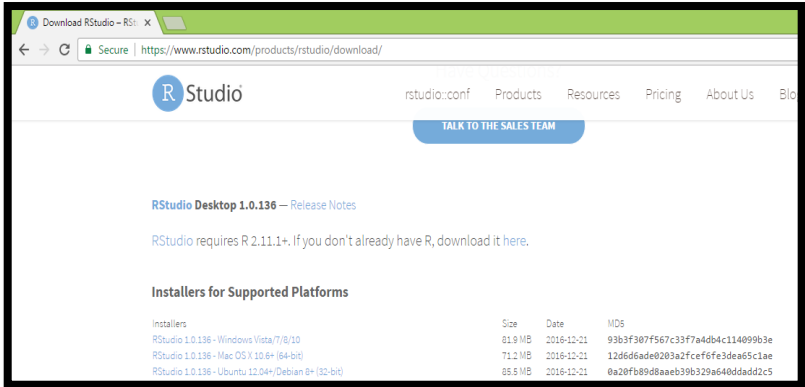
Beberapa langkah menginstall R diantaranya:

- a. Download R di link [cran.r-project.org/](https://cran.r-project.org/).

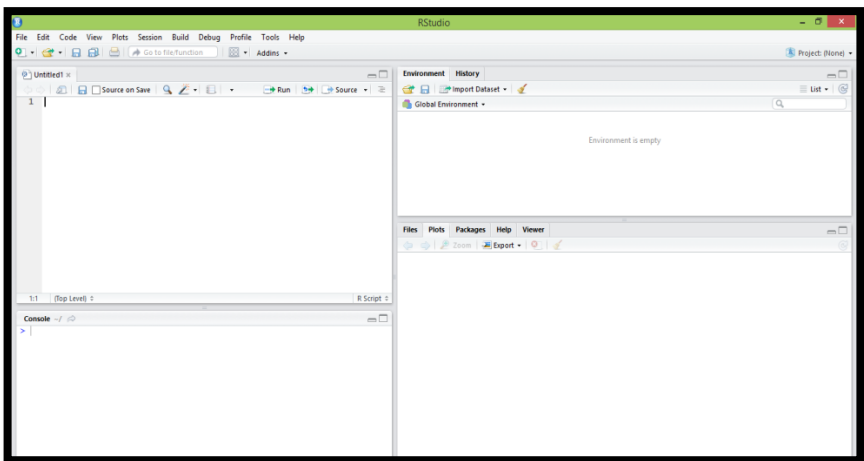




- b. Setelah R (R3.3.2.exe) dijalankan, selanjutnya download R Studio di [rstudio.com/products/rstudio/download/](https://www.rstudio.com/products/rstudio/download/)



- c. Jalankan R studio.



## Cara Memulai R

Berikut akan dipaparkan langkah-langkah untuk masuk ke dalam area kerja R. Aktifkan RStudio terlebih dahulu. Pilih File => New File => R Script, sehingga muncul tampilan yang merupakan area kerja R, di mana pada pembahasan selanjutnya, kode R akan diinput pada area tersebut. Setelah kode R diinput, selanjutnya kode R tersebut dieksekusi, sehingga muncul output berdasarkan eksekusi kode R tersebut.

Selanjutnya cara Menyimpan Data dalam Variabel (Fungsi `c`) Andaikan suatu data terdiri dari bilangan 10, 20, 40,15, 40. Misalkan data tersebut akan disimpan dalam variabel yang diberi nama `simpan`. Dalam R, fungsi `c` digunakan untuk menggabungkan satu nilai data, dengan nilai data lainnya. Perhatikan kode R berikut

```
simpan = c (10, 20, 40, 15, 40)
```

Kode R `simpan = c (10, 20, 40, 15, 40)`, dapat diartikan variabel `simpan` ditugaskan untuk menyimpan data dengan nilai 10, 20, 40, 15, 40. Data-data tersebut diapit oleh tanda buka-tutup kurung biasa, dan masing-masing nilai data dipisahkan oleh tanda koma. Perhatikan kode R `simpan = c (10, 20, 40, 15,`

40) dapat diartikan menyajikan atau menampilkan nilai data yang disimpan dalam variabel simpan.

Sekarang, kode R `simpan = c(10, 20, 40, 15, 40)` akan dieksekusi. Pilih Compile, sehingga muncul tampilan output dari hasil eksekusi kode R `simpan = c(10, 20, 40, 15, 40)`, dapat berformat HTML, PDF, dan Ms Word. Dalam percobaan kali ini, pilih HTML dan Compile.

Mengakses Nilai Data dalam Variabel secara Individual  
Misalkan variabel simpan menyimpan nilai 10, 20, 40, 15, dan 40. Andaikan hanya ingin ditampilkan nilai dari variabel simpan, pada posisi ke-3, yakni nilai 40. Perhatikan kode R berikut.

**`simpan[3]`**

Kode R di atas berarti menampilkan nilai dalam variabel simpan, pada posisi ke-3, yakni 40.

Perhatikan kode R berikut.

**`simpan[4]`**

Kode R di atas berarti menampilkan nilai dalam variabel simpan pada posisi ke-4, yakni 15.

Berikut merupakan kode R untuk menampilkan 3 buah nilai dalam variabel `simpan` pada posisi ke 1,3, dan 4.

**`simpan[c(1,3,4)]`**

Kode `simpan[c(1,3,4)]` berarti menampilkan nilai dalam variabel `simpan` pada posisi ke 1, 3, dan 4, yakni 10, 40, dan 15. Berikut merupakan kode R untuk menampilkan 3 buah nilai dalam variabel `simpan` pada indeks ke 2,3,4.

**`simpan[2:4]`**

Kode `simpan[2:4]` berarti menampilkan nilai dalam variabel `simpan` pada posisi ke-2, sampai posisi ke-4, yakni 10, 20, dan 40.

Mengubah Nilai Data dalam Variabel

Pada pemaparan sebelumnya, diketahui variabel `simpan` menyimpan nilai 10, 20, 40, 15, dan 40. Andaikan nilai dari variabel `simpan`, pada posisi ke-3, yakni nilai 40, akan diubah menjadi 100. Perhatikan kode R berikut.

**`simpan[3]=100`**

Kode R tersebut, yakni `simpan[3]=100`, dapat diartikan mengubah nilai data variabel `simpan` pada posisi ke-3 dengan nilai 100. Misalkan nilai pada posisi ke-4, yakni 15, ingin

diubah menjadi 10000. Berikut merupakan kode dalam R untuk mengubah nilai pada posisi ke-4, yakni 15 menjadi 10000.

```
simpan[4]=10000
```

Jenis Data R

Berikut merupakan berbagai jenis dari jenis data dalam R.

a. Numeric atau angka

Data yang berupa angka merupakan data numeric. Sebagai contoh 1, 100, 1500, 3000, dan seterusnya, merupakan data berjenis numerik. Untuk mengetahui jenis dari suatu data dalam R, digunakan perintah `mode`. Berikut merupakan kode R untuk mengetahui bahwa data 1000 termasuk ke dalam data berjenis numeric.

```
mode(1000)
```

b. Character atau karakter

Data yang berupa karakter diapit dengan tanda petik ganda “ ”. Sebagai contoh “Halo”, “A”, “1”, “500”, “+”, dan seterusnya merupakan data berjenis karakter.

Berikut merupakan perintah dalam R untuk

mengetahui bahwa data “HALO” termasuk ke dalam tipe data karakter.

**mode(“HALO”)**

c. Logical atau logika

Nilai dari data yang berjenis logika terdiri dari 2 nilai, yakni T atau F. T merupakan singkatan dari *True*, yang berarti benar, sementara F merupakan singkatan dari *False*, yang berarti salah. Misalkan sebuah variabel bernama *d* menyimpan sebuah data berjenis logika, yakni F.

**d=(F)**

Berikut merupakan kode dalam R untuk mengetahui bahwa nilai yang tersimpan dalam variabel *d* berjenis logika.

**mode(d)**

d. Function atau fungsi

Data yang berjenis fungsi mempunyai ciri menggunakan kata function. Berikut merupakan contoh kode program R pembuatan fungsi kuadrat.

```
kuadrat=function(x)
```

```
{
```

```
  print(x*x);
```

```
}
```

```
A=c(4)
```

```
kuadrat(A)
```

```
mode(kuadrat)
```

Perhatikan bahwa kuadrat merupakan nama variabel yang menyimpan data berjenis fungsi.

Beberapa jenis fungsi

1) Fungsi length

Fungsi length dalam R berfungsi untuk mengetahui jumlah elemen yang tersimpan atau terkandung dalam variabel. Misalkan suatu

variabel bernama A menyimpan nilai 100, 70, 80, 55, 80, 70, 80. Maka banyaknya elemen dalam variabel A adalah 7. Berikut merupakan kode R untuk menentukan banyaknya elemen yang terkandung dalam variabel A.

```
A=c(100,70,80,55,80,70,80)
```

```
length(A)
```

## 2) Fungsi sort

Fungsi sort dalam R berfungsi untuk mengurutkan data. Misalkan suatu variabel bernama A menyimpan nilai 70, 80, 50, 25, 100, 60. Berikut merupakan kode R untuk mengurutkan elemen-elemen atau nilai-nilai yang terkandung dalam variabel A.

```
A=c(70,80,50,25,100,60)
```

```
sort(A)
```



3) Fungsi diff

Misalkan diberikan data sebagai berikut.

100, 50, 70, 80, 60. Misalkan dilakukan perhitungan sebagai berikut.

$$50 - 100 = -50$$

$$70 - 50 = 20$$

$$80 - 70 = 10$$

$$60 - 80 = -20$$

Sehingga hasil akhirnya adalah  $-50, 20, 10, -20$ .

$$A=c(100,50,70,80,60)$$

$$\text{diff}(A)$$

4) Fungsi sum

Fungsi sum dalam R berfungsi untuk menjumlahkan seluruh nilai data. Misalkan variabel A menyimpan nilai 50, 60, 70, 80. Maka jumlah dari seluruh nilai dalam variabel A adalah 260.

```
A=c(50,60,70,80)
```

```
sum(A)
```

5) Fungsi barplot

Fungsi barplot dalam R berfungsi untuk menyajikan data dalam bentuk diagram batang. Misalkan variabel A menyimpan data 10, 10, 10, 10, 20, 20, 30, 30, 30, 30, 30, 30. Berikut akan disajikan data pada variabel A dalam bentuk diagram batang.

```
A=c(10, 10, 10, 10, 20, 20, 30, 30, 30, 30, 30, 30)  
barplot(table(A))
```

6) Fungsi plot

Misalkan variabel bernama A menyimpan data 10,10,10,10,10,20,20,20,30,30,40.

Berikut akan digunakan fungsi table untuk mengetahui frekuensi dari masing-masing nilai data.

```
A=c(10,10,10,10,10,20,20,20,30,30,40)
```

```
table(A)
```

Diketahui nilai 10 muncul sebanyak 5, nilai 20 sebanyak 3, nilai 30 sebanyak 2, dan nilai 40 sebanyak 1. Berikut akan digunakan fungsi `plot()` untuk memplot data yang tersimpan dalam variabel A.

```
nilai=c(10,20,30,40)
```

```
frekuensi=c(5,3,2,1)
```

```
plot(nilai,frekuensi)
```

Alternatif lain untuk menyajikan data.

```
A=c(10,10,10,10,10,20,20,20,30,30,40)
```

```
plot(table(A))
```

## 2. Install packages Rstudio

*Packages* di R studio dapat diinstall untuk mendukung analisis statistika dengan langkah-langkah sebagai berikut:

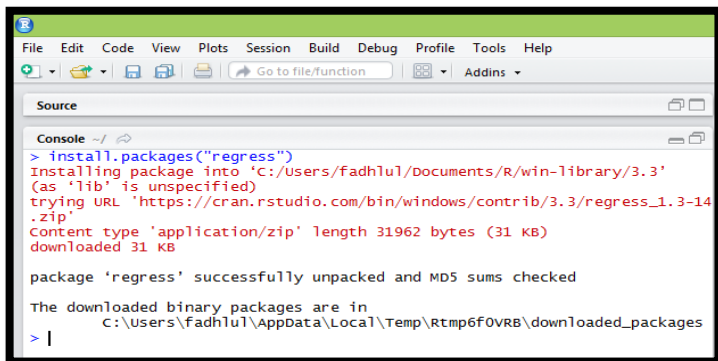
- a. Jalankan R studio.
- b. Beberapa cara install packages diantaranya:

- 1) Jika terhubung ke internet: `install.packages`  
(“namapackage”)
- 2) Jika sudah melakukan download pada *binary*  
(\**.zip*):`install.packages(“drive:/namafile.zip”,repos=`  
`NULL)`
- 3) Menggunakan menu:

Tools> Install packages(s)...>

#### Ilustrasi A.4.2

Misalkan ingin menginstall packages regresi dan komputer berhubungan dengan internet.



```
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console ~/\
> install.packages("regress")
Installing package into 'C:/Users/Fadhlul/Documents/R/win-library/3.3'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.3/regress_1.3-14
.zip'
Content type 'application/zip' length 31962 bytes (31 KB)
downloaded 31 KB

package 'regress' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\Fadhlul\AppData\Local\Temp\Rtmp6f0VRB\downloaded_packages
> |
```

## E. Karakteristik R

R memiliki karakteristik yang berbeda dengan bahasa pemrograman lain seperti C++, python, dll. R memiliki aturan/ sintaks yang berbeda dengan bahasa pemrograman yang lain yang membuatnya memiliki ciri khas tersendiri dibanding bahasa pemrograman yang lain.

Beberapa karakter pada R antara lain:

1. Bahasa R bersifat *case sensitif*. Maksudnya adalah dalam proses input R huruf besar dan kecil sangat diperhatikan.
2. Segala sesuatu yang ada pada program R akan dianggap sebagai objek. konsep objek ini sama dengan bahasa pemrograman berbasis objek yang lain seperti Java, C++, python, dll. Perbedaannya adalah bahasa R relatif lebih sederhana dibandingkan bahasa pemrograman berbasis objek yang lain.
3. *Interpreted language* atau script. Bahasa R memungkinkan pengguna untuk melakukan kerja pada R

tanpa perlu kompilasi kode program menjadi bahasa mesin.

4. Mendukung proses loop, decision making, dan menyediakan berbagai jenis operator (aritmatika, logika, dll).
5. Mendukung export dan import berbagai format file, seperti: TXT, CSV, XLS, dll.
6. Mudah ditingkatkan melalui penambahan fungsi atau *library*. Penambahan Paket dapat dilakukan secara online melalui CRAN atau melalui sumber seperti github.
7. Menyediakan berbagai fungsi untuk keperluan visualisasi data. Visualisasi data pada R dapat menggunakan Paket bawaan atau Paket lain seperti `ggplo2`, `ggvis`, dll.

## F. Kuis

Lakukan penginstallan pada packages kalibrasi (*calibrate*) dan statistika circular (*circular statistics*) !!!

# **FITUR DASAR R**

## BAB 2

# FITUR DASAR R

### A. Objek Data Dasar di R Studio

Beberapa jenis objek data yang sering digunakan di R studio diantaranya:

#### 1. Vektor (*vector*)

Vektor adalah objek data yang menyimpan satu atau lebih tipe data lain. Vektor hanya bisa menyimpan tipe data yang sama, jika memaksakan menyimpan tipe data berbeda, maka akan dikonversi ke karakter. Vektor dibuat dengan fungsi `c()`.

Vector adalah objek data paling sederhana yang ada di dalam R. Secara umum jenis vector terbagi 2, yaitu numeric dan character.



Ada banyak sekali cara untuk membuat sebuah vector di R. Di bagian ini akan dibahas beberapa cara yang banyak dan sering digunakan.

a. Fungsi `c()`

Fungsi yang paling sering digunakan untuk membuat sebuah vector adalah dengan menggunakan fungsi `c()`.

```
x <- c(2, 1, 5, 3, 1)
```

```
x
```

```
## [1] 2 1 5 3 1
```

Pada script di atas, dibuat sebuah objek `x` berupa vector numeric. Setiap elemen dipisah menggunakan tanda koma (`,`). Fungsi ini dapat digunakan untuk membuat vector numeric atau character. Indeks di R dimulai dari 1, tidak seperti kebanyakan bahasa pemrograman lain yang indeksnya dimulai dari 0. Hal ini cukup memudahkan pengguna karena umumnya manusia menghitung mulai dari 1, bukan dari 0.

ketika Anda ingin mengambil elemen ke 2 dari  
vector `x`, maka Anda dapat menjalankan perintah

```
x[2]
```

```
## [1] 1
```

Tanda titik dua/colon (:)

Untuk membuat sebuah vector numeric  
berurutan secara meningkat atau menurun. Lihat  
contoh berikut ini.

```
x <- 1:10 # 1 sampai 10
```

```
x
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

Fungsi dari operator: pada contoh di atas adalah  
membuat vector numeric dengan nilai dari 1 s/d 10.  
Tentu saja operator increment ini hanya dapat  
digunakan untuk numeric dan meningkat sebesar 1  
nilai. Operator ini dapat juga digunakan untuk  
membuat vector dengan nilai menurun.

```
# membuat vector numeric dengan nilai dari 10 s/d -  
10 secara menurun 1
```

```
x <- 10:-10 # 10 sampai -10
```

```
x
## [1] 10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -
4 -5
## [17] -6 -7 -8 -9 -10
```

b. Fungsi seq()

Fungsi ini dapat digunakan untuk membuat vector berurutan dan dengan increment tertentu.

```
x <- seq(from = 1, to = 10) # 1 sampai 10 dengan
increment 1 (default by = 1)
```

```
x
## [1] 1 2 3 4 5 6 7 8 9 10
```

Secara default increment dari fungsi seq() adalah by = 1. Jika Anda ingin nilai increment lain maka hanya perlu mengganti nilai pada argument by. Fungsi ini juga hanya dapat digunakan untuk membuat vector numeric.

```
x <- seq(from = 1, to = 20, by = 2) # 1 sampai 20
dengan increment 2
```

```
x
## [1] 1 3 5 7 9 11 13 15 17 19
```

```
x <- seq(from = 1, to = 10, length.out = 7) # 1 sampai
10, sebanyak 7 elemen, increment mengikuti
```

```
x
```

```
## [1] 1.0 2.5 4.0 5.5 7.0 8.5 10.0
```

```
x <- seq(from = 1, to = 10, along.with = 1:4) # 1  
sampai 10, sebanyak elemen dari vector lain
```

```
x
```

```
## [1] 1 4 7 10
```

Mengambil satu kolom dari dataframe atau matriks

Mengambil sebuah kolom dari sebuah dataframe akan dibahas lebih jauh di bagian dataframe Dengan menggunakan tanda dolar \$ dan diikuti dengan nama kolom yang akan diambil dari dataframe tersebut.

```
mtcars$mpg
```

```
## [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8  
19.2 17.8 16.4 17.3
```

```
## [14] 15.2 10.4 10.4 14.7 32.4 30.4 33.9 21.5 15.5  
15.2 13.3 19.2 27.3
```

```
## [27] 26.0 30.4 15.8 19.7 15.0 21.4
```

Dari data frame mtcars diambil semua nilai yang ada di kolom mpg

c. Fungsi unlist()

Fungsi ini berguna untuk menjadikan sebuah objek list menjadi sebuah vector. Pembahasan lebih lanjut akan dibahas di bagian berikutnya.

```
x <- list(mtcars$mpg, mtcars$disp)
x
## [[1]]
## [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8
19.2 17.8 16.4 17.3
## [14] 15.2 10.4 10.4 14.7 32.4 30.4 33.9 21.5 15.5
15.2 13.3 19.2 27.3
## [27] 26.0 30.4 15.8 19.7 15.0 21.4
##
## [[2]]
## [1] 160.0 160.0 108.0 258.0 360.0 225.0 360.0
146.7 140.8 167.6 167.6
## [12] 275.8 275.8 275.8 472.0 460.0 440.0 78.7
75.7 71.1 120.1 318.0
## [23] 304.0 350.0 400.0 79.0 120.3 95.1 351.0
145.0 301.0 121.0
x <- unlist(x)
x
```

```
## [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4  
22.8 19.2 17.8
```

```
## [12] 16.4 17.3 15.2 10.4 10.4 14.7 32.4 30.4  
33.9 21.5 15.5
```

```
## [23] 15.2 13.3 19.2 27.3 26.0 30.4 15.8 19.7  
15.0 21.4 160.0
```

```
## [34] 160.0 108.0 258.0 360.0 225.0 360.0 146.7  
140.8 167.6 167.6 275.8
```

```
## [45] 275.8 275.8 472.0 460.0 440.0 78.7 75.7  
71.1 120.1 318.0 304.0
```

```
## [56] 350.0 400.0 79.0 120.3 95.1 351.0 145.0  
301.0 121.0
```

Fungsi `unlist()` menggabungkan semua list menjadi sebuah vector.

Catatan penting untuk vector: walaupun ditampilkan ke samping, dimensi vector di R sebenarnya ke bawah. Bayangkan untuk sebuah vector seperti satu kolom di Ms Excel.

Semua contoh di atas untuk membuat vector adalah vector numeric. Vector numeric adalah vector yang semua elemennya bernilai dan bertipe numeric.

d. Fungsi rep()

Dengan fungsi rep() Anda dapat membuat sebuah vector dengan mengulang-ulang nilai yang diinginkan sebanyak yang dibutuhkan.

# Membuat sebuah vector numeric yang semua elemennya bernilai 3 sebanyak 10 elemen

```
x <- rep(3, 10)
```

```
x
```

```
## [1] 3 3 3 3 3 3 3 3 3 3
```

Vector character

Vector character adalah vector yang semua elemennya bertipe character.

```
y <- c("a", "A", "d", "c")
```

```
y
```

```
## [1] "a" "A" "d" "c"
```

Jika ketika membuat sebuah vector bernilai numeric namun ada satu saja elemennya bertipe character maka semua elemennya akan bertipe character.

```
c(1, 2, 3, 5, "a")
```

```
## [1] "1" "2" "3" "5" "a"
```

Di R ada 2 buah vector khusus yang bertipe character, yaitu letters dan LETTERS.

### **letters**

```
## [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l"  
"m" "n" "o" "p"
```

```
## [17] "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"
```

### **LETTERS**

```
## [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K"  
"L" "M" "N" "O" "P"
```

```
## [17] "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
```

Dua buah vector atau lebih dapat digabungkan dengan fungsi `c()`. Namun, jika salah satu vector bertipe character, maka vector hasil gabungan akan menjadi vector character. Contoh di bawah ini menggabungkan vector `x` dan `y`. Vector `x` adalah vector numeric, sedangkan `y` adalah vector character. Karena ada satu atau lebih elemen yang berupa



character, maka ketika x dan y digabungkan akan menjadi vector character.

```
c(x, y)
```

```
## [1] "3" "3" "3" "3" "3" "3" "3" "3" "3" "3" "a" "A"  
"d" "c"
```

Cara lain yang dapat digunakan untuk membuat vector character adalah menggunakan fungsi paste(), paste0() atau sprintf(). Jalankan dan perhatikan perbedaan dari contoh perintah di bawah ini.

```
paste("A", 1:5)
```

```
## [1] "A 1" "A 2" "A 3" "A 4" "A 5"
```

```
paste0("A", 1:5)
```

```
## [1] "A1" "A2" "A3" "A4" "A5"
```

```
sprintf("A%s", 1:5)
```

```
## [1] "A1" "A2" "A3" "A4" "A5"
```

## 2. Matriks (matrix)

Matriks adalah objek R dimana unsur-unsurnya diatur pada persegi panjang dua dimensi. Sebuah Matrix dibuat dengan fungsi matrix().

Matriks adalah objek di R yang memiliki 2 dimensi, baris (*row*) dan kolom (*column*), dan tipe nilainya sama. Jika ketika membuat sebuah matriks elemennya memiliki minimal 1 elemen bertipe character maka seluruh matriks tersebut akan bertipe character. Membuat matriks di R menggunakan vector yang dikonversi dimensinya.

```
x <- mtcars$mpg # sebuah vector
```

```
length(x)
```

```
## [1] 32
```

Karena vector *x* memiliki 32 elemen, maka dimensi matriks yang dapat dibuat adalah 2 angka yang hasil perkaliannya menghasilkan nilai 32. Salah satunya adalah  $8 \times 4 = 32$ .

```
m <- matrix(data = x, nrow = 8, ncol = 4)
```

```
m
```

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

```
## [1,] 21.0 22.8 14.7 19.2
```

```
## [2,] 21.0 19.2 32.4 27.3
```

```
## [3,] 22.8 17.8 30.4 26.0
```

```
## [4,] 21.4 16.4 33.9 30.4  
## [5,] 18.7 17.3 21.5 15.8  
## [6,] 18.1 15.2 15.5 19.7  
## [7,] 14.3 10.4 15.2 15.0  
## [8,] 24.4 10.4 13.3 21.4
```

Argumen `byrow = TRUE` artinya matriks akan setiap elemen `x` diisikan ke `m` memenuhi baris terlebih dahulu. Jika `byrow = FALSE` maka setiap elemen `x` diisikan ke `m` berdasarkan kolom terlebih dahulu.

```
Matrix(1:10, ncol = 5, nrow = 2, byrow = TRUE)
```

```
## [1,] [2,] [3,] [4,] [5,]  
## [1,] 1 2 3 4 5  
## [2,] 6 7 8 9 10
```

```
Matrix(1:10, ncol = 5, nrow = 2, byrow = FALSE)
```

```
## [1,] [2,] [3,] [4,] [5,]  
## [1,] 1 3 5 7 9  
## [2,] 2 4 6 8 10
```

Untuk membuat matriks dengan nilai yang sama seluruhnya, maka dapat dilakukan seperti berikut.

```
Matrix(data = 0, nrow = 5, ncol = 6)
```

```
##   [,1] [,2] [,3] [,4] [,5] [,6]  
## [1,]  0  0  0  0  0  0  
## [2,]  0  0  0  0  0  0  
## [3,]  0  0  0  0  0  0  
## [4,]  0  0  0  0  0  0  
## [5,]  0  0  0  0  0  0
```

Untuk mengakses elemen dari suatu matriks, Anda dapat menggunakan indeks dari baris atau kolomnya.

```
# Mengambil elemen matriks `m` di baris 4  
m[4, ]  
## [1] 21.4 16.4 33.9 30.4  
# Mengambil elemen matriks `m` di kolom 3  
m[, 3]  
## [1] 14.7 32.4 30.4 33.9 21.5 15.5 15.2 13.3  
# Mengambil elemen matriks `m` di baris 4 dan kolom 3  
m[4, 3]  
## [1] 33.9  
# Mengambil elemen matriks `m` di baris 4 dan 6, dan  
kolom 3  
m[c(4, 6), 3]  
## [1] 33.9 15.5
```

R menyediakan sebuah fungsi yaitu `diag()` untuk mengakses nilai-nilai pada diagonal utama sebuah matriks.

```
diag(m)
```

```
## [1] 21.0 19.2 30.4 30.4
```

Kita juga dapat mengganti nilai dari elemen suatu matriks dengan menggunakan operator assignment.

```
m[4, 3] <- 0
```

```
m[4, 3]
```

```
## [1] 0
```

m # perhatikan elemen di baris 4 kolom 3 sudah berubah jadi 0.0

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

```
## [1,] 21.0 22.8 14.7 19.2
```

```
## [2,] 21.0 19.2 32.4 27.3
```

```
## [3,] 22.8 17.8 30.4 26.0
```

```
## [4,] 21.4 16.4 0.0 30.4
```

```
## [5,] 18.7 17.3 21.5 15.8
```

```
## [6,] 18.1 15.2 15.5 19.7
```

```
## [7,] 14.3 10.4 15.2 15.0
```

```
## [8,] 24.4 10.4 13.3 21.4
```

### 3. Array

Array adalah objek R Data yang dapat menyimpan data dalam lebih dari dua dimensi. Array dibuat dengan fungsi `array()`.

### 4. Faktor (factor)

Faktor adalah objek data yang digunakan untuk mengkategorikan data. Faktor dibuat dengan fungsi `factor()`.

Factor merupakan bentuk lebih luas dari vector. Biasanya factor lebih sering digunakan untuk menyimpan data nominal atau ordinal. Misalnya vector character yang berisi "male" dan "female". Pada vector character, nilainya adalah "male" dan "female" seperti terlihat apa adanya. Namun pada factor, tampilan dari isi datanya mungkin "male" dan "female" tetapi isi dari factor adalah pengkodean numerik. Misal untuk "female" nilai sebenarnya adalah 1, sedangkan "male" berniali 2.

```
fc <- factor(c("SD", "SMA", "SMP", "SMP", "SD",  
"SMA", "SD", "SMP"))
```

```
fc
```

```
## [1] SD SMA SMP SMP SD SMA SD SMP
```

```
## Levels: SD SMA SMP
```

Nilai sebenarnya dari factor tersebut adalah

```
print.default(fc)
```

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

Factor mempunyai level, secara default levelnya adalah berdasarkan urutan alfabet. Untuk merubah level dari sebuah factor, gunakan argumen `levels =`.

```
factor(fc, levels = c("SD", "SMP", "SMA"))
```

```
## [1] SD SMA SMP SMP SD SMA SD SMP
```

```
## Levels: SD SMP SMA
```

```
factor(fc, levels = c("SD", "SMP", "SMA"), ordered =  
TRUE)
```

```
## [1] SD SMA SMP SMP SD SMA SD SMP
```

```
## Levels: SD < SMP < SMA
```

## **Daftar (list)**

Daftar adalah objek R yang mengandung angka, string, vektor dan daftar lain di dalamnya. Daftar juga dapat berisi matriks atau fungsi sebagai unsur-unsurnya. Daftar dibuat dengan fungsi `list()`.

Objek list pada dasarnya mirip seperti vector, hanya saja tipe elemennya bisa berbeda. Jika pada vector numeric semua elemennya harus berupa numerik. Pada vector character semuanya harus karakter. Pada objek list elemennya dapat berupa vector, factor, matriks, array, dataframe, bahkan list di dalam list atau objek lain seperti model prediktif yang dibuat di R.

Contoh membuat list dengan fungsi `list()`.

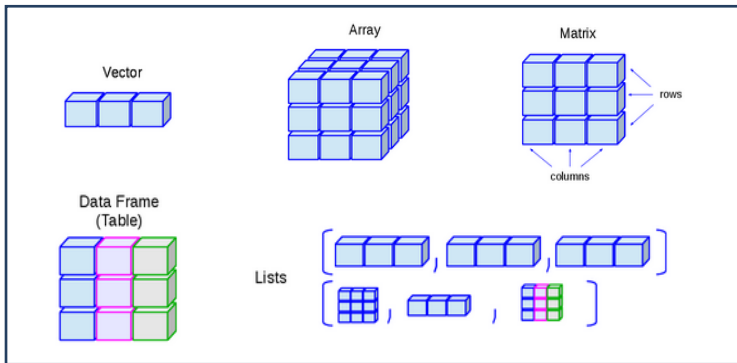
```
list(2, "A", c(4, 5, 2), iris[1:5,])
```

## **Data frame**

Sebuah frame data adalah tabel dimana setiap kolom berisi nilai-nilai dari satu variabel dan setiap baris berisi satu set nilai dari masing-masing kolom. Data frame dibuat dengan fungsi `data.frame ()`.



Dataframe layaknya sebuah tabel di Ms Excel, terdiri dari baris dan kolom dengan nama masing-masing kolom berbeda. Apa bedanya dengan matriks? Matriks hanya bisa menyimpan tipe data yang sama, numeric atau character seluruhnya. Pada dataframe, masing-masing kolom boleh memiliki tipe data yang berbeda. Dataframe seperti umumnya bentuk tabel yang sering kita gunakan.



### Ilustrasi 1 (B.1)

Misalkan ingin membuat vektor ukuran  $5 \times 1$ ; matriks ukuran  $2 \times 3$ ; array 2 dimensi dengan masing-masing dimensi terdiri dari vektor  $3 \times 1$ ; faktor yang terdiri dari 2 level yaitu M yang diulang sebanyak 3 kali

dan F yang diulang sebanyak 2 kali; serta daftar dari vektor, matriks, array, dan faktor.

```
Console ~1 ↻
> vektor<-c(1,2,3,4,5)
> matrik<-matrix(c(1:6),2,3)
> array<-array(c(1:6),dim=c(3,1,2))
> faktor<-factor(c("M","M","F","M","F"))
> daftar<-list(var1=vektor,var2=matrik,var3=array,var4=faktor)
> daftar
$var1
[1] 1 2 3 4 5


$var2
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6

$var3
, , 1
      [,1]
[1,]    1
[2,]    2
[3,]    3
, , 2
      [,1]
[1,]    4
[2,]    5
[3,]    6

$var4
[1] M M F M F
Levels: F M
```

### Ilustrasi 2 (B.2)

Misalkan ingin membuat tabel atau gabungan antara vektor ukuran 5 x 1 dan faktor yang terdiri dari 2 level yaitu M yang diulang sebanyak 3 kali dan F yang diulang sebanyak 2 kali.

```
Console ~/ 
> vektor<-c(1,2,3,4,5)
> vektor
[1] 1 2 3 4 5
>
> faktor<-factor(c("M","M","F","M","F"))
> faktor
[1] M M F M F
Levels: F M
>
> dataframe<-data.frame(var1=vektor, var2=faktor)
> dataframe
  var1 var2
1     1    M
2     2    M
3     3    F
4     4    M
5     5    F
```

## B. Aturan Nama Penyimpanan (*Named Storage*)

Pada program R studio terdapat beberapa aturan dalam penamaan objek diantaranya:

1. Tidak boleh menuliskan angka di awal nama objek.
2. Tidak menggunakan tanda operasi matematika “-, +, /, \*  
“ dalam penggabungan 2 suku kata atau lebih nama objek.
3. Tidak menggunakan spasi untuk pemisahan nama objek yang memiliki 2 suku kata atau lebih.

### Ilustrasi B.2

Misalkan ingin membuat sebuah objek “ data fadhlul mubarak ” yang memiliki nilai 98.

```
Console ~/ | ↻
> data.fadhlul.mubarak<-98
> data.fadhlulmubarak<-98
> datafadhlulmubarak<-98
> data_fadhlul_mubarak<-98
>
> 1data.fadhlul.mubarak<-98
Error: unexpected symbol in "1data.fadhlul.mubarak"
>
> data-fadhlulmubarak<-98
Error in data - fadhlulmubarak <- 98 : could not find function "-<-"
>
> data fadhlulmubarak<-98
Error: unexpected symbol in "data fadhlulmubarak"
```

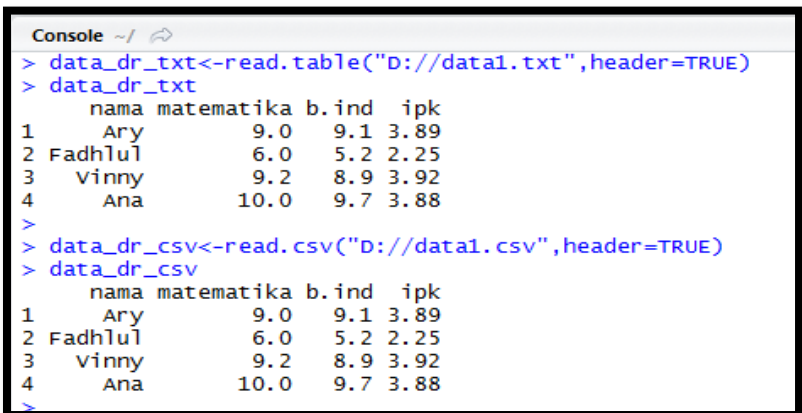
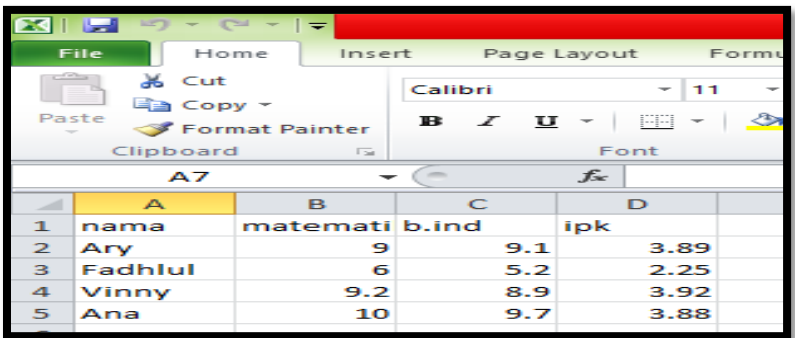
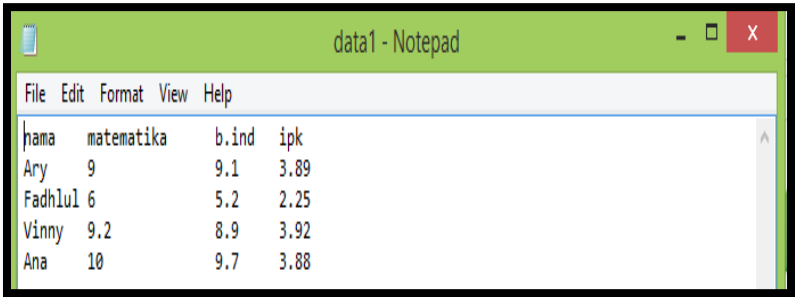
## C. Pembacaan Data Eksternal di R

Beberapa format yang sering dilakukan pembacaan data eksternal di R adalah .txt dan .csv. Data berformat.txt dapat menggunakan perintah `read.table` (“namadata”,header=TRUE) dan data berformat .csv menggunakan perintah `read.csv` (“namadata”,header=TRUE).

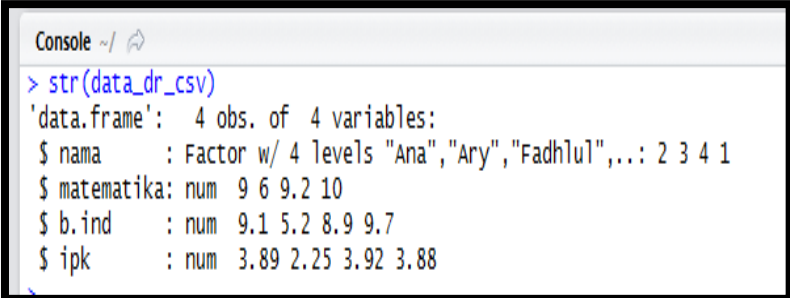
### Ilustrasi 1 (B.3)


Misalkan akan dibaca data eksternal berformat .txt dan .csv yang berada di *disk* D dengan nama “data1.txt” dan

“data1.csv” lalu diberikan nama objek baru pada R menjadi  
“data\_dr\_txt” dan “data\_dr\_csv”.



Untuk melakukan pengecekan, maka digunakan fungsi `str` (`namadata`).



```
Console ~/   
> str(data_dr_csv)  
'data.frame': 4 obs. of 4 variables:  
 $ nama      : Factor w/ 4 levels "Ana","Ary","Fadhlu1",...: 2 3 4 1  
 $ matematika: num 9 6 9.2 10  
 $ b.ind     : num 9.1 5.2 8.9 9.7  
 $ ipk      : num 3.89 2.25 3.92 3.88
```

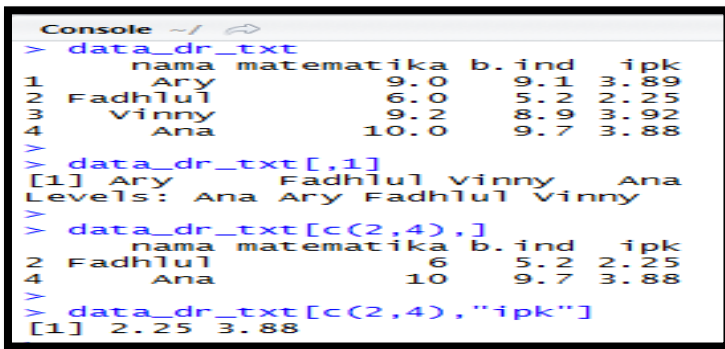
#### D. Ekstrak data di R

Operasi pengolahan data frame akan menjadi penting karena terkadang hanya beberapa data saja yang akan dilakukan analisis. Perintah pada R yang diberikan tergantung pada data yang akan diambil.

##### Ilustrasi D

Misalkan diketahui data frame dengan nama `data_dr_txt` dengan variable `nama`, `matematika`, `b.ind`, dan `ipk` yang masing-masing memiliki 4 amatan. Jika ingin mengambil semua amatan dari variable `nama` maka digunakan perintah `data_dr_txt$nama` atau `data_dr_txt[,1]`.

Jika ingin mengambil semua variable untuk amatan 2 dan 4 maka digunakan perintah `data_dr_txt[c(2,4),]`. Jika ingin mengambil amatan 2 dan 4 pada variable ipk maka digunakan perintah `data_dr_txt[c(2,4),"ipk"]`



```
Console ~/
> data_dr_txt
  nama matematika b.ind ipk
1 Ary 9.0 9.1 3.89
2 Fadhlul 6.0 5.2 2.25
3 Vinny 9.2 8.9 3.92
4 Ana 10.0 9.7 3.88
>
> data_dr_txt[,1]
[1] Ary Fadhlul Vinny Ana
Levels: Ana Ary Fadhlul Vinny
>
> data_dr_txt[c(2,4),]
  nama matematika b.ind ipk
2 Fadhlul 6 5.2 2.25
4 Ana 10 9.7 3.88
>
> data_dr_txt[c(2,4),"ipk"]
[1] 2.25 3.88
```

## E. Kuis

Gunakan data pada kuis b5.csv dan kuis b51.csv yang terdapat pada disk D untuk menentukan nilai akhir mahasiswa. Diketahui dari data kuis 51.csv terdapat beberapa variabel di antaranya jk (1= laki-laki dan 2= perempuan), status (1=belum menikah dan 2=sudah menikah), dan pekerjaan (1=dosen, 2=karyawan, dan 3=wiraswasta). Nilai UTS memiliki bobot 40%, UAS memiliki bobot 40, dan tugas memiliki bobot 20%

sehingga penjumlahan dari ketiga nilai ini disebut nilai akhir.  
Siapakah yang belum memiliki nilai tertinggi dan terendah dari  
yang memiliki pekerjaan sebagai dosen ?



# STATISTIKA DESKRIPTIF

## BAB 3

### STATISTIKA DESKRIPTIF

*Statistika (statistik) deskriptif* adalah metode-metode yang berkaitan dengan pengumpulan dan penyajian suatu gugus data sehingga memberikan informasi yang berguna. Statistik deskriptif hanya memberikan informasi mengenai data (menyajikan data). Contoh statistika deskriptif yang sering muncul adalah, tabel, diagram dan grafik. Dengan statistik deskriptif data dapat disajikan dengan lebih ringkas dan rapi serta dapat memberikan informasi inti dari kumpulan data yang ada.

Untuk tahap ini akan membahas penggunaan **R-Commander** untuk membuat statistik deskriptif dari suatu kumpulan data, khususnya pembuatan ringkasan (*summary*) data dan pembuatan tabel. Ringkasan data difokuskan pada pembuatan statistik deskriptif, yaitu ukuran-ukuran **pemusatan**,

penyebaran, kemiringan, keruncingan, dan lokasi dari data-data numerik (metrik). Sedangkan pembuatan tabel difokuskan pada data-data nonnumerik.

### A. *Summarization* (ringkasan) Data Kontinu

Beberapa analisis statistika berupa ringkasan yang sering dipakai pada data kontinu di R studio diantaranya pemusatan (*center*), penyebaran (*spread*), kecondongan (*skew*) dan keruncingan (*peaked*). Pada R Studio perintah ringkasan data kontinu digunakan fungsi `summary` (nama data kontinu).

<b>Tipe</b>	<b>Contoh</b>
Pemusatan	rataan, median, dll
Penyebaran	Varian, standar deviasi, dll
Kecondongan	skewness
Keruncingan	kurtosis

1. Rataan (mean)

$$\frac{\sum_{i=1}^n x_i}{n} \equiv \mu \equiv \bar{X}$$

Dimana  $n$  merupakan jumlah data,  $x_i$  merupakan data ke- $i$ ,  $\mu$  merupakan rata-rata populasi, dan  $\bar{x}$  merupakan rata-rata sampel.

Rataan adalah contoh tipe data atau ukuran pemusatan. Ukuran gejala pusat merupakan suatu ukuran atau nilai yang letaknya cenderung terletak dipusat data.

Berikut beberapa penjelasan mengenai ukuran gejala pusat. Smidh dan Sanders (2000:73) menyatakan sebagai berikut.

*“You know from Chapter 2 that there are several measures of central tendency. The purpose of these measures is to summarize in a single value the typical size, middle property, or central location of a set of values. The most familiar measure of central tendency is, of course, the arithmetic mean, which is*

*simply the sum of the values of a group of items divided by the number of such items. But you also saw in Chapter 2 that the median and mode are other measures of central tendency that are commonly used.” Spiegel dan Stephens (2008:62) menyatakan sebagai berikut. “An average is a value that is typical, or representative, of a set of data. Since such typical values tend to lie centrally within a set of data arranged according to magnitude, averages are also called measures of central tendency. Several types of averages can be defined, the most common being the arithmetic mean, the median, the mode, the geometric mean, and the harmonic mean. Each has advantages and disadvantages, depending on the data and the intended purpose.”*

*“Anda tahu dari Bab 2 bahwa ada beberapa ukuran tendensi sentral. Tujuan pengukuran ini adalah untuk meringkas dalam satu nilai ukuran tipikal, properti tengah, atau lokasi pusat dari sekumpulan nilai. Ukuran yang paling umum dari tendensi sentral, tentu saja, rata-rata aritmatika, yang merupakan penjumlahan dari nilai*

*sekelompok item dibagi dengan jumlah item tersebut. Tetapi Anda juga melihat di Bab 2 bahwa median dan mode adalah ukuran lain dari tendensi sentral yang umum digunakan. ” Spiegel dan Stephens (2008: 62) menyatakan sebagai berikut. “Rata-rata adalah nilai yang khas, atau mewakili, sekumpulan data. Karena nilai-nilai tipikal tersebut cenderung terletak secara terpusat dalam kumpulan data yang disusun menurut besarnya, rata-rata juga disebut ukuran tendensi sentral. Beberapa jenis rata-rata dapat didefinisikan, yang paling umum adalah mean aritmatika, median, mode, mean geometrik, dan mean harmonik. Masing-masing memiliki kelebihan dan kekurangan, tergantung datanya dan tujuan yang dimaksudkan.”*

Berdasarkan uraian di atas, nilai rata-rata dapat diartikan sebagai nilai tipikal atau representatif atau perwakilan dari suatu set data. Beberapa contoh dari ukuran gejala pusat atau rata-rata adalah rata-rata aritmatik (arithmetic mean), median, modus, rata-rata geometrik, dan rata-rata harmonik. Di antara berbagai

ukuran gejala pusat tersebut memiliki kelebihan dan kekurangan, bergantung pada data dan tujuan yang dimaksud.

Smidth dan Sanders (2000: 73) menyatakan sebagai berikut.

*“Data often have a tendency to congregate about some central value, and this central value may then be used as a summary measure to describe the general data pattern.”*

*“Data seringkali memiliki kecenderungan untuk berkumpul tentang beberapa nilai pusat, dan nilai pusat ini kemudian dapat digunakan sebagai ukuran ringkasan untuk menggambarkan pola data umum.”*

## 2. Median

Spiegel dan Stephens (2008:64) menyatakan sebagai berikut.

*“The median of a set of numbers arranged in order of magnitude (i.e., in an array) is either the middle value or the arithmetic mean of the two middle values.*

*“Geometrically the median is the value of  $X$  (abscissa) corresponding to the vertical line which divides a*

*histogram into two parts having equal areas. This value of  $X$  is sometimes denoted by  $\bar{X}$ ”.*

*“Median sekumpulan angka yang disusun dalam urutan besarnya (yaitu, dalam larik) adalah nilai tengah atau rata-rata aritmetika dari dua nilai tengah.*

*“Secara geometris median adalah nilai  $X$  (absis) yang bersesuaian dengan garis vertikal yang membagi sebuah histogram menjadi dua bagian yang memiliki luas yang sama. Nilai  $X$  ini terkadang dilambangkan dengan  $\bar{X}$ ”*

Berdasarkan uraian tersebut, median juga disebut juga dengan nilai tengah (middle value) atau rata-rata aritmatik dari dua nilai tengah. Nilai dari median membagi data menjadi dua bagian yang sama. Notasi atau simbol untuk rata-rata aritmatik sampel adalah  $\bar{X}$ , sementara notasi atau simbol median sampel adalah  $Me$ . Sebelum menghitung nilai median, terlebih dahulu data diurutkan dari yang terkecil hingga terbesar.

Berikut rumus menghitung median untuk data dengan jumlah genap dan untuk data dengan jumlah ganjil



$$\text{Me} = Q_2 = \begin{cases} x_{\frac{n+1}{2}}, & \text{jika } n \text{ ganjil} \\ \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2}, & \text{jika } n \text{ genap} \end{cases}$$

Dimana  $n$  merupakan jumlah data

Mann dan Lacke (2011: 85) menyatakan sebagai berikut.

*“The median gives the center of a histogram, with half of the data values to the left of the median and half to the right of the median. The advantage of using the median as a measure of central tendency is that it is not influenced by outliers. Consequently, the median is preferred over the mean as a measure of central tendency for data sets that contain outliers. For example, when a data set has outliers, instead of using the mean, we can use either the trimmed mean or median as a measure of central tendency.”*

*“Median memberikan pusat histogram, dengan separuh nilai data berada di kiri median dan separuh di kanan median. Keuntungan menggunakan median sebagai ukuran tendensi sentral adalah tidak dipengaruhi oleh pencilan. Akibatnya, median lebih disukai daripada mean sebagai ukuran tendensi sentral untuk kumpulan data yang berisi pencilan. Misalnya, ketika kumpulan data memiliki pencilan, daripada menggunakan mean, kita dapat menggunakan mean atau median yang dipangkas sebagai ukuran tendensi sentral.*

Berdasarkan uraian tersebut, keuntungan menggunakan median sebagai ukuran gejala pusat adalah median tidak terpengaruh oleh *outlier* (data pencilan). Oleh karena itu, median lebih disukai dibandingkan rata-rata atau mean (rata-rata aritmatik) sebagai ukuran gejala pusat, untuk data yang mengandung *outlier*.

### 3. Kuartil

Kuartil merupakan jenis-jenis dari ukuran letak. Ukuran tersebut membagi data menjadi beberapa bagian yang sama. Sebagai contoh pada ukuran kuartil terdapat tiga buah nilai. Letak dari nilai-nilai kuartil tersebut membagi data menjadi empat bagian yang sama.

Ukuran kuartil terdiri dari tiga buah nilai yang membagi data menjadi empat bagian yang sama.

Perhatikan urutan angka berikut :

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

Nilai kuartil dikelompokkan atas tiga, yakni kuartil pertama ( $Q_1$ ), kuartil kedua ( $Q_2$ ), dan kuartil ketiga ( $Q_3$ ). Angka 3, 6, dan 9 masing-masing merupakan  $Q_1$ ,  $Q_2$ , dan  $Q_3$ .

Berikut rumus untuk menghitung nilai kuartil.

$$Q_i = \begin{cases} \frac{i(n+1)}{4}, & \text{untuk } n \text{ ganjil} \\ Q_1 = x_{\frac{(n+2)}{4}} \\ Q_2 = \frac{1}{2} \left( x_{\frac{n}{2}} + x_{\frac{n}{2}+1} \right), & \text{untuk } n \text{ genap} \\ Q_3 = x_{\frac{(3n+2)}{4}} \end{cases}$$

Dimana  $n$  merupakan jumlah data.

Misalkan, disajikan data :

1, 2, 2, 3, 3, 3, 4, 5, 6, 7, 7, 8, 9, 10, 10, 11, 12, 13, 15

Diketahui banyaknya nilai data  $n = 19$ . Berikut akan dihitung nilai dari  $Q_1$ ,  $Q_2$ , dan  $Q_3$ .

$$Q_1 = \frac{1(19+1)}{4}$$

$$Q_1 = 5$$

$Q_1 = 5$  berarti nilai  $Q_1$  terletak pada data dengan urutan ke-5, yakni 3

$$Q_2 = \frac{2(19+1)}{4}$$

$$Q_2 = 10$$

$Q_2 = 10$  berarti nilai  $Q_2$  terletak pada data dengan urutan ke-10, yakni 10

$$Q_3 = \frac{3(19+1)}{4}$$

$$Q_3 = 15$$

$Q_3 = 15$  berarti nilai  $Q_3$  terletak pada data dengan urutan ke-15, yakni 7.

Ketiga nilai kuartil tersebut membagi data menjadi empat bagian yang sama. Bagian pertama adalah {1,2,2,3}, bagian kedua adalah {3,4,5,6}, bagian ketiga adalah {7,8,9,10}, dan bagian keempat adalah {11,12,13,15}. Dan kita lihat bahwa banyaknya nilai untuk masing-masing bagian adalah 4.

#### 4. Varian dan standar deviasi

$$\sum_{i=1}^n \frac{(x_i - \mu)^2}{n} \equiv \sigma^2,$$
$$\sqrt{\sum_{i=1}^n \frac{(x_i - \mu)^2}{n}} \equiv \sigma$$

Dimana  $n$  merupakan jumlah data,  $x_i$  merupakan data ke- $i$ ,  $\mu$  merupakan rata-rata populasi, dan  $\sigma$  merupakan standar deviasi populasi.

Variance (dalam hal ini variance untuk sampel) dilambangkan dengan  $\sigma^2$ . Rumus varian seperti yang sudah ditulis di atas.

Standar deviasi merupakan akar kuadrat positif variance  $\sqrt{\sigma^2} = \sigma$ . Nilai dari standar deviasi dapat diinterpretasi sebagai nilai yang menunjukkan seberapa dekat nilai-nilai data menyebar atau berkumpul di sekitar rata-ratanya. Standar deviasi merupakan salah satu dari ukuran pencaran yang paling sering digunakan. Mann dan Lacke (2011: 93) menyatakan sebagai berikut.

*“The standard deviation is the most-used measure of dispersion. The value of standard deviation tells how closely the values of a data set are clustered around the mean. In general, a lower value of the standard deviation for a data set indicates that the values of that data set are spread over a relatively smaller range around the mean. In contrast, a larger value of the standard deviation for a data set indicates that the values of that data set are spread over a relatively larger range around the mean.”*

*“Standar deviasi adalah ukuran dispersi yang paling banyak digunakan. Nilai deviasi standar menunjukkan seberapa dekat nilai-nilai kumpulan data dikelompokkan di sekitar mean. Secara umum, nilai simpangan baku yang lebih rendah untuk suatu kumpulan data menunjukkan bahwa nilai dari kumpulan data tersebut tersebar pada kisaran yang relatif lebih kecil di sekitar mean. Sebaliknya, nilai deviasi standar yang lebih besar untuk suatu kumpulan data menunjukkan bahwa nilai dari kumpulan data tersebut tersebar pada kisaran yang relatif lebih besar di sekitar mean.*

Misalkan, jika diketahui nilai varian adalah 180,7018, maka nilai standar deviasi adalah  $\sqrt{180,7018} = 13,4425$ .

Dapat dilihat bahwa semakin besar nilai standar deviasi dari suatu data, maka sebaran data cenderung jauh terhadap rata-ratanya (walaupun ada beberapa data yang dekat dengan rata-ratanya).

### 5. *Skewness* (kemiringan atau kecondongan)

Ukuran kemiringan atau *skewness* merupakan suatu nilai yang mengukur ketidaksimetrisan distribusi data. Suatu data dikatakan berdistribusi simetris sempurna bila nilai rata-rata, median, dan modus dalam data adalah sama.

Berikut adalah rumus untuk menyatakan *Skewness* (kemiringan atau kecondongan)

$$\frac{\sum_{i=1}^n (x_i - \bar{x})^3}{\sum_{i=1}^n (x_i - \bar{x})^{3/2}} \times \frac{(n-1)^{3/2}}{n-2}$$

Dimana  $n$  merupakan jumlah data,  $x_i$  merupakan data ke- $i$ ,  $\bar{x}$  merupakan rata-rata sampel, dan  $\sigma$  merupakan standar deviasi populasi.

Bila nilai kemiringan  $< 0$  atau negatif, maka kurva cenderung condong ke kiri (kurva negatif). Jika nilai kemiringan  $> 0$  atau positif, maka kurva cenderung



condong ke kanan (kurva positif). Jika nilai kemiringan mendekati 0 atau 0, maka kurva cenderung simetris. Spiegel dan Stephens (2008: 125) menyatakan sebagai berikut.

*“Skewness is the degree of asymmetry, or departure from symmetry, of a distribution. If the frequency curve (smoothed frequency polygon) of a distribution has a longer tail to the right of the central maximum than to the left, the distribution is said to be skewed to the right, or to have positive skewness. If the reverse is true, it is said to be skewed to the left, or to have negative skewness.*

*“Skewness adalah derajat asimetri, atau penyimpangan dari simetri, suatu distribusi. Jika kurva frekuensi (poligon frekuensi yang dihaluskan) dari suatu distribusi memiliki ekor yang lebih panjang di sebelah kanan maksimum pusat daripada di sebelah kiri, distribusi tersebut dikatakan miring ke kanan, atau memiliki*

*kemiringan positif. Jika kebalikannya benar, dikatakan miring ke kiri, atau memiliki kemiringan negatif”*

## 6. Kurtosis

Ukuran keruncingan atau kurtosis merupakan suatu nilai yang mengukur tingkat keruncingan atau ketinggian puncak dari distribusi data. Berikut rumus untuk menghitung kurtosis.

$$\text{Kurtosis} = \left\{ \frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum \left( \frac{x_j - \bar{x}}{\sigma} \right)^4 \right\} - \frac{3(n-1)^2}{(n-2)(n-3)}$$

Dimana  $n$  merupakan jumlah data,  $x_i$  merupakan data ke- $i$ ,  $\bar{x}$  merupakan rata-rata sampel, dan  $\sigma$  merupakan standar deviasi populasi.

Ada 3 jenis keruncingan atau ketinggian puncak dari distribusi data. Pertama, kurva normal atau mesokurtis (kurva tidak terlalu tajam dan datar). Kedua, kurva cenderung datar dan puncak tidak terlalu tinggi. Kurva ini dinamakan kurva platikurtis. Dan ketiga, puncak kurva

terlihat lancip dan tinggi. Kurva ini dinamakan kurva leptokurtis.

Spiegel dan Stephens (2008:125) menyatakan sebagai berikut.

*“Kurtosis is the degree of peakedness of a distribution, usually taken relative to a normal distribution. A distribution having a relatively high peak is called leptokurtic, while one which is flat-topped is called platykurtic. A normal distribution, which is not very peaked or very flat-topped, is called mesokurtic.”*

*“Kurtosis adalah derajat puncak dari suatu distribusi, biasanya diambil relatif terhadap distribusi normal. Distribusi yang memiliki puncak yang relatif tinggi disebut leptokurtik, sedangkan yang puncaknya rata disebut platykurtik. Distribusi normal, yang tidak terlalu memuncak atau sangat datar, disebut mesokurtik”*

## Aplikasi dalam R

Semua data yang telah dibuat, mulai dari penentuan mean, median, kuartil, varian, standard deviasi, kuartil, *skewness*, dan kurtosis disimpan terlebih dahulu dalam Microsoft Excel, dan disimpan dengan format .CSV

Berikutnya akan disimpan dalam R dengan kode  
`simpan=read.table("data.csv",header=TRUE, sep=",")`  
`#membaca data4`

Di dalam R akan tersimpan dan terbaca sesuai data yang kita buat.

Untuk mengesekusi data yang tersimpan dalam R, pilih *compile* dan pilih format output HTML.

Pada kode R tersebut, akan digunakan fungsi *describe*. Fungsi tersebut terdapat dalam package *psych*. Oleh karena itu, kode R library (*psych*) dapat diartikan mengaktifkan package *psych*. Setelah package *psych* diaktifkan, barulah fungsi *describe* dapat digunakan. Fungsi *describe* dalam hal ini digunakan untuk menentukan banyaknya data (n), rata-rata

aritmatik (mean), standar deviasi (sd), median, minimum (min), maksimum (max), range, kemiringan (skew), dan kurtosis.

## **B. *Summarization* (ringkasan) untuk data kategorik**

Pada data kategorik ringkasan yang biasa digunakan adalah tabel frekuensi. Pada R Studio perintah ringkasan data kategorik digunakan fungsi `summary` (nama data kategorik).

### **Ilustrasi A dan B**

Misalkan ingin menampilkan ringkasan dari data `c` pada disk `D` yang di dalamnya terdapat data kontinu variabel kontinu dan kategorik secara sekaligus. Pada R Studio perintah ringkasan data digunakan fungsi `summary` (nama data).

```
Console ~/ | ↵
> data<-read.csv("D://c.csv",header=TRUE)
> data
  nama      jk      status pekerjaan uts uas
1 fadhul laki-laki belum menikah dosen 95 30
2 mubarak laki-laki belum menikah karyawan 88 85
3 hijrah laki-laki sudah menikah karyawan 87 86
4 ary laki-laki sudah menikah karyawan 98 100
5 yunus laki-laki sudah menikah karyawan 90 98
6 ana perempuan sudah menikah karyawan 100 80
7 pipit perempuan sudah menikah wiraswasta 100 98
8 sabil laki-laki sudah menikah wiraswasta 89 76
9 ray laki-laki sudah menikah wiraswasta 85 80
10 razi laki-laki belum menikah wiraswasta 100 85
11 roza perempuan sudah menikah wiraswasta 99 99
12 vinny perempuan belum menikah dosen 99 100
13 permata perempuan sudah menikah wiraswasta 85 95
14 dara perempuan belum menikah dosen 100 85
15 fakhrur laki-laki sudah menikah karyawan 95 90
16 tamtama laki-laki sudah menikah wiraswasta 86 98
17 farid laki-laki sudah menikah wiraswasta 100 89
18 raffan laki-laki belum menikah dosen 85 85
19 adi laki-laki sudah menikah karyawan 89 95
20 rafiqa perempuan belum menikah dosen 100 100
>
> summary(data)
      nama      jk      status      pekerjaan      uts      uas
adi   : 1 laki-laki:13 belum menikah: 7 dosen      : 5 Min.   : 85.00 Min.   : 30.0
ana   : 1 perempuan: 7 sudah menikah:13 karyawan   : 7 1st Qu.: 87.75 1st Qu.: 85.0
ary   : 1 wiraswasta:8 Median   : 95.00 Median   : 89.5
dara  : 1 Mean     : 93.50 Mean     : 87.7
fadhlul: 1 3rd Qu.:100.00 3rd Qu.: 98.0
fakhrur: 1 Max.    :100.00 Max.    :100.0
(other):14
>
```

### C. Kuis

Berdasarkan data c pada disk D tampilkan ringkasan (min, kuartil 1, median, mean, kuartil 3, maksimal, varian, standar deviasi) untuk nilai uts dan uas pada masing-masing jenis kelamin, status, dan pekerjaan.

# UJI T

## BAB 4

### UJI T

#### A. Uji T Satu Sampel (*One Sample T-Test*)

Uji T satu sampel merupakan prosedur uji t yang digunakan membandingkan rata-rata suatu sampel dengan suatu nilai konstanta tertentu. Nilai konstanta tersebut pada umumnya merupakan sebuah nilai parameter untuk mengukur nilai populasi. Uji T biasanya digunakan jika jumlah sampel data dibawah 30. Uji t satu sampel memiliki syarat, yaitu : data yang digunakan data kuantitatif (skala : interval atau rasio), dan memenuhi asumsi distribusi normal. Pada RStudio perintah uji t satu sampel adalah `t.test(y,mu=..)`.



## 1. Hipotesis

Ada beberapa bentuk uji t satu sampel, yaitu :

### a. Uji Sisi Kiri

$H_0 : \mu \geq \mu_0$  rata-rata sampel paling rendah  
sebesar  $\mu_0$

$H_1 : \mu < \mu_0$  rata-rata sampel kurang dari  $\mu_0$

### b. Uji Sisi Kanan

$H_0 : \mu \leq \mu_0$  rata-rata sampel paling tinggi sebesar  $\mu_0$

$H_1 : \mu > \mu_0$  rata-rata sampel lebih dari  $\mu_0$

### c. Uji Dua Sisi

$H_0 : \mu = \mu_0$  rata-rata sampel mencapai  $\mu_0$

$H_1 : \mu \neq \mu_0$  rata-rata sampel tidak mencapai  $\mu_0$

## 2. Statistik Uji

$$t = \frac{\bar{x} - \mu_0}{s / \sqrt{n}}$$

Dimana  $\bar{x}$  merupakan rata-rata sampel,  $\mu_0$  merupakan rata-rata parameter,  $s$  merupakan standar deviasi,  $n$  merupakan jumlah sampel.

### 3. Kriteria Uji

Dalam menguji hipotesis, kriteria penolakan atau penerimaan  $H_0$  didasarkan pada nilai *p-value* atau nilai *t*-hitung. Nilai *p-value* di bandingkan dengan nilai  $\alpha$  ( $\alpha$ ), sedangkan nilai *t*-hitung di bandingkan dengan nilai *t*-tabel. Kriteria uji tersebut adalah sebagai berikut :

#### a. Uji Sisi Kiri

Jika nilai *p-value* (*sig.*)  $< \alpha$ , maka  $H_0$  di tolak

Jika nilai *p-value* (*sig.*)  $> \alpha$ , maka  $H_0$  di terima

atau

Jika nilai *t*-hitung  $< t$ -tabel ( $-t_\alpha$ ), maka  $H_0$  di tolak

Jika nilai *t*-hitung  $> t$ -tabel ( $-t_\alpha$ ), maka  $H_0$  di terima

#### b. Uji Sisi Kanan

Jika nilai *p-value* (*sig.*)  $< \alpha$ , maka  $H_0$  di tolak

Jika nilai *p-value* (*sig.*)  $> \alpha$ , maka  $H_0$  di terima

atau

Jika nilai t-hitung  $>$  t-tabel ( $t_{\alpha}$ ), maka  $H_0$  di tolak

Jika nilai t-hitung  $<$  t-tabel ( $t_{\alpha}$ ), maka  $H_0$  di terima

c. Uji Dua Sisi

Jika nilai *p-value* (*sig.*)  $<$   $\alpha/2$ , maka  $H_0$  di tolak

Jika nilai *p-value* (*sig.*)  $>$   $\alpha/2$ , maka  $H_0$  di terima

atau Jika nilai t-hitung  $>$  t-tabel ( $-t_{\alpha/2}$ ), maka  $H_0$  di

tolak Jika nilai t-hitung  $<$  t-tabel ( $-t_{\alpha/2}$ ), maka  $H_0$  di

terima

### **Ilustrasi A**

Misalkan akan dibuktikan bahwa dugaan selama ini rata-rata nilai uas mahasiswa pada data c di disk D sebesar 88.

```
Console ~/1 ↵
> data<-read.csv("D://c.csv",header=TRUE)
> data
  nama      jk      status pekerjaan uts uas
1 fadhlul laki-laki belum menikah dosen 95 30
2 mubarak laki-laki belum menikah karyawan 88 85
3 hijrah laki-laki sudah menikah karyawan 87 86
4 ary laki-laki sudah menikah karyawan 98 100
5 yunus laki-laki sudah menikah karyawan 90 98
6 ana perempuan sudah menikah karyawan 100 80
7 pipit perempuan sudah menikah wiraswasta 100 98
8 sabil laki-laki sudah menikah wiraswasta 89 76
9 ray laki-laki sudah menikah wiraswasta 85 80
10 razi laki-laki belum menikah wiraswasta 100 85
11 roza perempuan sudah menikah wiraswasta 99 99
12 vinny perempuan belum menikah dosen 99 100
13 permata perempuan sudah menikah wiraswasta 85 95
14 dara perempuan belum menikah dosen 100 85
15 fahrur laki-laki sudah menikah karyawan 95 90
16 tamtama laki-laki sudah menikah wiraswasta 86 98
17 farid laki-laki sudah menikah wiraswasta 100 89
18 rafhan laki-laki belum menikah dosen 85 85
19 adi laki-laki sudah menikah karyawan 89 95
20 rafiqa perempuan belum menikah dosen 100 100
> t.test(data$uas,mu=80)

      One Sample t-test

data:  data$uas
t = 2.2059, df = 19, p-value = 0.0399
alternative hypothesis: true mean is not equal to 80
95 percent confidence interval:
 80.39396 95.00604
sample estimates:
mean of x
 87.7
```

Terlihat p-value yang diperoleh sebesar 0,0399. Hal ini mengindikasikan bahwa anggapan selama ini yang menyatakan bahwa rata-rata nilai uas 80 ditolak. Hal ini juga mengindikasikan bahwa rata-rata nilai uas lebih besar daripada anggapan selama ini.

## B. Uji T Dua Sampel (*Two Sample T-Test*)

Uji t dua sample merupakan prosedur uji t yang digunakan membandingkan parameter dua populasi berbeda, apakah sama atau tidak. Berdasarkan hubungan antar populasinya, uji t di bedakan menjadi dua, yaitu uji t dua sampel berpasangan dan uji t dua sampel bebas.

### 1. Uji t dua sampel berpasangan (*Dependent sample t-test/ Paired sample t-test*)

Tujuannya adalah membandingkan rata-rata dua grup yang saling berpasangan, yaitu sampel dengan subjek yang sama namun terdapat dua perlakuan/kejadian berbeda, misalnya pengukuran sampel terhadap *treatment* tertentu sebelum dan sesudahnya. Perintah R Studio terkait uji t 2 sampel berpasangan adalah `t.test(sampel1,sampel2,alternative="..", conf.level=...,paired=TRUE)`.

#### a. Hipotesis

$H_0 : \mu_B = 0$  Tidak ada perbedaan antara sampel 1 dan 2

$H_1 : \mu_1 \neq \mu_2$  terdapat perbedaan antara sampel 1 dan 2

### b. Statistik Uji

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2} - 2r \left( \frac{s_1}{\sqrt{n_1}} \right) \left( \frac{s_2}{\sqrt{n_2}} \right)}}$$

Dimana  $\bar{x}_1$  merupakan rata-rata sampel 1,  $\bar{x}_2$  merupakan rata-rata sampel 2,  $s_1^2$  merupakan varians sampel 1,  $s_2^2$  merupakan varians sampel 2,  $s_1$  merupakan simpangan baku sampel 1,  $s_2$  merupakan simpangan baku sampel 2, dan  $r$  merupakan korelasi antara dua sampel.

### c. Kriteria Uji

Dalam menguji hipotesis, kriteria penolakan atau penerimaan  $H_0$  didasarkan pada nilai  $p$ -value atau nilai  $t$ -hitung.

Jika nilai  $p$ -value ( $sig.$ )  $< \alpha$ , maka  $H_0$  di tolak

Jika nilai  $p$ -value ( $sig.$ )  $> \alpha$ , maka  $H_0$  di terima

atau

Jika nilai  $t$ -hitung  $>$   $t$ -tabel, maka  $H_0$  di tolak

Jika nilai  $t$ -hitung  $<$   $t$ -tabel, maka  $H_0$  di terima

## 2. Uji $t$ dua sampel saling bebas (*Independent sample $t$ -test*)

Tujuannya adalah membandingkan rata-rata dua grup yang tidak saling berpasangan, yaitu dua subjek yang berbeda. Pada RStudio perintah uji  $t$  2 sampel saling bebas adalah `t.test(sampel1,sampel2)`.

### a. Hipotesis

$H_0 : \mu_1 = \mu_2$  rata-rata sampel 1 sama dengan rata-rata sampel 2

$H_1 : \mu_1 \neq \mu_2$  rata-rata sampel 1 tidak sama dengan rata-rata sampel 2

**b. Statistik Uji**

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2} \left(\frac{1}{n_1}\right) \left(\frac{1}{n_2}\right)}}$$

Dimana  $\bar{x}_1$  merupakan rata-rata sampel 1,  $\bar{x}_2$  merupakan rata-rata sampel 2,  $s_1^2$  merupakan varians sampel 1,  $s_2^2$  merupakan varians sampel 2,  $s_1$  merupakan simpangan baku sampel 1, dan  $s_2$  merupakan simpangan baku sampel 2.

**c. Kriteria Uji**

Dalam menguji hipotesis, kriteria penolakan atau penerimaan  $H_0$  didasarkan pada nilai *p-value* atau nilai *t*-hitung. Jika nilai *p-value* (*sig.*)  $< \alpha$ , maka  $H_0$  di tolak

Jika nilai *p-value* (*sig.*)  $> \alpha$ , maka  $H_0$  di terima.

Atau jika nilai *t*-hitung  $> t$ -tabel, maka  $H_0$  di tolak. Jika nilai *t*-hitung  $< t$ -tabel, maka  $H_0$  di terima




## **Ilustrasi B**

Seorang Guru matematika di salah satu SMA Negeri di Bogor ingin menguji apakah terdapat perbedaan nilai ujian matematika antara Kelas A dan B. Guru tersebut mengambil sampel secara acak sebanyak 15 orang siswa dari masing-masing kelas. Data tersimpan dengan nama `datad2.csv` pada disk D. Ujilah, apakah kelas A dan B memiliki rata-rata nilai Ujian matematika yang sama?

Hipotesis dari kasus diatas adalah

$H_0 : \mu_1 = \mu_2$  rata-rata nilai ujian Kelas A sama dengan  
Kelas B

$H_1 : \mu_1 \neq \mu_2$  rata-rata nilai ujian Kelas A tidak sama dengan  
Kelas B

```
Console ~/ 
> datad2<-read.csv("D://datad2.csv",header=TRUE)
> datad2
  No.  A  B
1    1  80 68
2    2  90 57
3    3  70 46
4    4  86 57
5    5  82 44
6    6  88 51
7    7  91 62
8    8  93 54
9    9  80 49
10   10 79 48
11   11 89 64
12   12 88 48
13   13 85 58
14   14 79 55
15   15 91 52
>
> t.test(datad2$A,datad2$B)

      Welch Two Sample t-test

data:  datad2$A and datad2$B
t = 12.671, df = 27.743, p-value = 4.695e-13
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 25.59535 35.47132
sample estimates:
mean of x mean of y
 84.73333  54.20000
```

Berdasarkan output R di atas, kita dapat nilai t-hitung sebesar 12.6713, dan t-tabel sebesar 1.20014, maka  $H_0$  di tolak. Hal tersebut mengindikasikan bahwa rata-rata nilai ujian Kelas A tidak sama dengan Kelas B, atau dengan kata lain terdapat perbedaan rata-rata nilai ujian untuk Kelas A dan B.

### C. Kuis

Dari hasil penelitian sebelumnya diketahui bahwa rata-rata tinggi badan mahasiswa adalah 170 cm dan berat badan 70 kg. Seorang peneliti ingin melakukan penelitian mengenai tinggi badan dan berat badan yang di ambil dari 20 mahasiswa secara acak. Data tersimpan dengan nama kuis d1.csv pada disk D. Berdasarkan 20 data mahasiswa di atas, ujilah :

1. Apakah benar jika ada seorang dosen yang mengatakan bahwa rata-rata tinggi mahasiswa paling rendah adalah 170 cm?
2. Apakah benar jika ada mahasiswa yang mengatakan bahwa rata-rata berat badan mahasiswa paling tinggi 70 kg?
3. Apakah rata-rata tinggi badan dan berat badan pada penelitian saat ini masih sama dengan penelitian sebelumnya?



# FASILITAS HELP

## BAB 5

### FASILITAS HELF

#### A. Mencari Help dari Suatu Perintah Tertentu

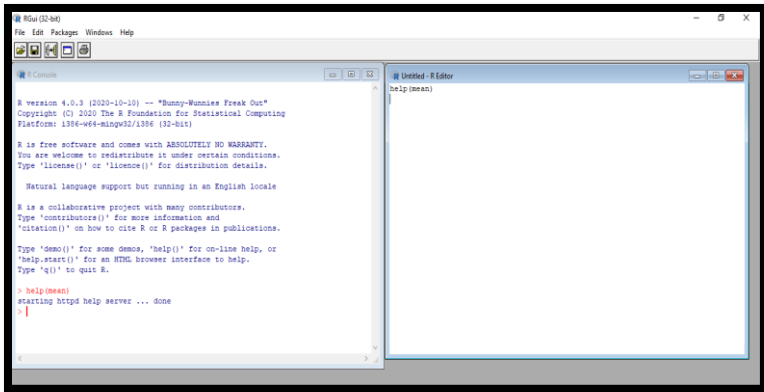
Untuk memperoleh bantuan terkait suatu perintah tertentu kita dapat menggunakan fungsi `help()`. Secara umum format yang digunakan adalah sebagai berikut:

`Help (nama_perintah)`

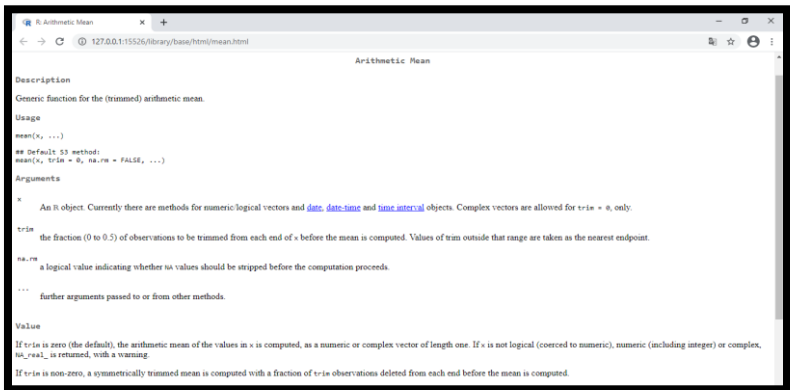
Atau dapat juga menggunakan tanda tanya (?) pada awal `nama_perintah` seperti berikut:

`?nama_perintah`

Misalkan kita kebingungan terkait bagaimana cara menuliskan perintah untuk menghitung rata-rata suatu vektor. Kita dapat mengetikkan perintah berikut untuk mengakses fasilitas *help*.



```
R version 4.0.3 (2020-10-10) -- "Bunny-Whisker Freak Out"  
Copyright (C) 2020 The R Foundation for Statistical Computing  
Platform: i386-w64-mingw32/x86_64  
  
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.  
  
Some demos: 'demo()', 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> help(mean)  
starting httpd help server ... done  
> |
```



```
Arithmetic Mean  
Description  
Generic function for the (trimmed) arithmetic mean.  
Usage  
mean(x, ...)  
## Default S3 method:  
mean(x, trim = 0, na.rm = FALSE, ...)  
Arguments  
x  
An R object. Currently there are methods for numeric logical vectors and date, date-time and time interval objects. Complex vectors are allowed for trim = 0, only.  
trim  
the fraction (0 to 0.5) of observations to be trimmed from each end of x before the mean is computed. Values of trim outside that range are taken as the nearest endpoint.  
na.rm  
a logical value indicating whether NA values should be stripped before the computation proceeds.  
...  
further arguments passed to or from other methods.  
Value  
If trim is zero (the default), the arithmetic mean of the values in x is computed, as a numeric or complex vector of length one. If x is not logical (coerced to numeric), numeric (including integer) or complex, is.na(x) is returned, with a warning.  
If trim is non-zero, a symmetrically trimmed mean is computed with a fraction of trim observations deleted from each end before the mean is computed.
```

Keterangan pada jendela pada gambar adalah sebagai berikut:

1. Pada bagian jendela kiri atas jendela *help*, diberikan keterangan nama dari perintah yang sedang ditampilkan.

2. Selanjutnya, pada bagian atas dokumen, ditampilkan informasi terkait nama perintah, dan nama Paket yang memuat perintah tersebut. Pada gambar di atas informasi terkait perintah dan nama Paket ditunjukkan pada teks `mean {base}` yang menunjukkan perintah `mean()` pada Paket (Paket) *base* (Paket bawaan R).
3. Setiap jendela *help* dari suatu perintah tertentu selanjutnya akan memuat bagian-bagian berikut:
  - a. *Title*
  - b. *Description*: deskripsi singkat tentang perintah.
  - c. *Usage*: menampilkan sintaks perintah untuk penggunaan perintah tersebut.
  - d. *Arguments*: keterangan mengenai *argument/ input* yang diperlukan pada perintah tersebut.
  - e. *Details*: keterangan lebih lengkap tentang perintah tersebut.
  - f. *Value*: keterangan tentang *output* suatu perintah dapat diperoleh pada bagian ini.



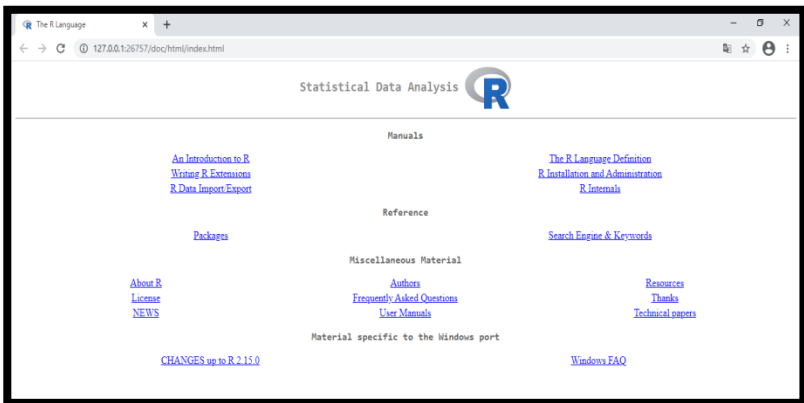
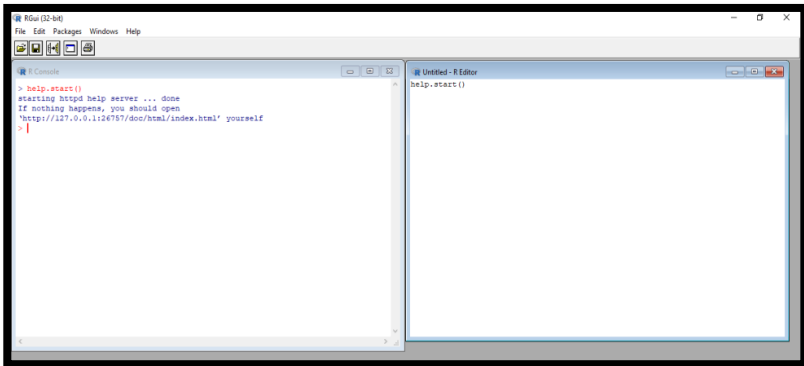
- g. *Author(s)*: memberikan keterangan tentang *Author* dari perintah tersebut.
- h. *References*: seringkali referensi yang dapat digunakan untuk memperoleh keterangan lebih lanjut terhadap suatu perintah ditampilkan pada bagian ini.
- i. *See also*: bagian ini berisikan daftar perintah/fungsi yang berhubungan erat dengan perintah tersebut.
- j. *Example*: berisikan contoh-contoh penggunaan perintah tersebut.

## B. Help Umum

Kita juga dapat membaca beberapa dokumen manual yang ada pada R. Untuk melakukannya jalankan perintah berikut:

```
help.start()
```

Output yang dihasilkan berupa link pada sejumlah dokumen yang dapat kita klik. Tampilan halaman yang dihasilkan disajikan pada gambar



# DAFTAR PUSTAKA

## DAFTAR PUSTAKA

Bloomfield, V.A. 2014. Using R for Numerical Analysis in Science and Engineering. CRC Press.

Primartha, R. 2018. Belajar Machine Learning Teori dan Praktik. Penerbit Informatika.

Rosadi, D. 2016. Analisis Statistika dengan R. Gadjah Mada University Press.

Rosidi, M. 2019. Metode Numerik Menggunakan R untuk Teknik Lingkungan. Bookdown.

Tim Pusat R. 2018. R: A Language and Environment for Statistical Computing. R Manuals.

Venables, W.N. Smith D.M. dan R Core Team. 2018. An Introduction to R. R Manuals.



## BIODATA PENULIS



**Titin Agustin Nengsih**, lahir pada 16 Agustus 1982 di Kerinci, Provinsi Jambi. Latar belakang pendidikan yang ditempuh adalah Sarjana(S1)di bidang Statistika yang lulus pada tahun 2000 dari Institut Pertanian Bogor (IPB). Pendidikan Magister (S2) juga di bidang yang sama dan ditempuh di kampus yang sama, Institut Pertanian Bogor (IPB) dengan beasiswa Kementerian Agama, lulus pada tahun 2010. Setelah itu melanjutkan pendidikan doktoral (S3) di Kota Strasbourg Perancis pada tahun 2020 dengan beasiswa *Mora Scholarship* Kementerian Agama. Mengawali karier di Universitas Islam Negeri Sulthan Thaha Saifuddin Jambi sebagai dosen PNS pada tahun 2006 pada Fakultas Syariah dan tahun 2016 di Fakultas Ekonomi dan Bisnis Islam. Pada tahun 2015-2016 menjabat sebagai sekretaris Prodi Ekonomi Islam dan aktif di pusat kajian sosial keagamaan, Koordinator bidang ekonomi di BI dan organisasi wanita. Saat ini menjabat sebagai Wakil Dekan bidang Administrasi Umum, Perencanaan dan keuangan yang sebelumnya menjadi ketua Pusat Kajian Ketrampilan Akademik Mahasiswa dan Karier Alumni. Salah satu pengalaman dalam bidang penelitian adalah di analisis Regresi *Partial Least Square* dan Data Hilang.

## BIODATA PENULIS



**Fadhlul Mubarak**, lahir di Ujung Gading (Sumatera Barat) pada April 1992. Pendidikan Sarjana (S1) lulus pada tahun 2011 di Universitas Hasanuddin pada bidang Statistika yang disponsori Beasiswa Yayasan Belanda. Selanjutnya lulus pada tahun 2017 pada pendidikan Magister (S2) pada bidang Statistika Terapan Institut Pertanian Bogor yang disponsori Beasiswa Unggulan. Pernah juga mengikuti kuliah Administrasi Bisnis Universitas Anadolu pada tahun 2018. Saat ini sedang menyelesaikan pendidikan Doktor (Ph.D) di Universitas Tekni Eskisehir (Turki) yang disponsori Beasiswa Pemerintah Turki. Pada tahun 2016-2017 pernah bekerjasama dengan Badan Kebijakan Fiskal untuk melakukan pemodelan. Peminatan penelitian untuk saat ini adalah pada analisis spasial runtun waktu.

## BIODATA PENULIS



**Vinny Yuliani Sundara**, lahir di Jambi, pada tanggal 18 Juli 1992. Pada tahun 2010 penulis terdaftar sebagai mahasiswa jurusan matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Penelusuran Kemampuan Akademik dan Bakat (PKAB), dan menyelesaikan studi program sarjana (S1) pada tahun 2014. Pada tahun yang sama penulis melanjutkan studi program master (S2) pada program studi Statistika di Institut Pertanian Bogor dengan beasiswa Lembaga Pengelola Dana Pendidikan (LPDP). Saat ini ia menjadi dosen Statistika di UIN Sulthan Thaha Saifuddin Jambi.