

Modul 1

PENGENALAN R

Pendahuluan

R adalah suatu bahasa komputer dan merupakan lingkungan pemrograman interaktif untuk analisis data dan grafik. Bahasa R adalah bahasa tingkat tinggi (*very high level language*) untuk komputasi. Bahasa R memungkinkan kita untuk menghitung, melihat data dan program secara interaktif dengan umpan balik yang cepat sehingga memungkinkan kita untuk belajar dan memahami tentang data.

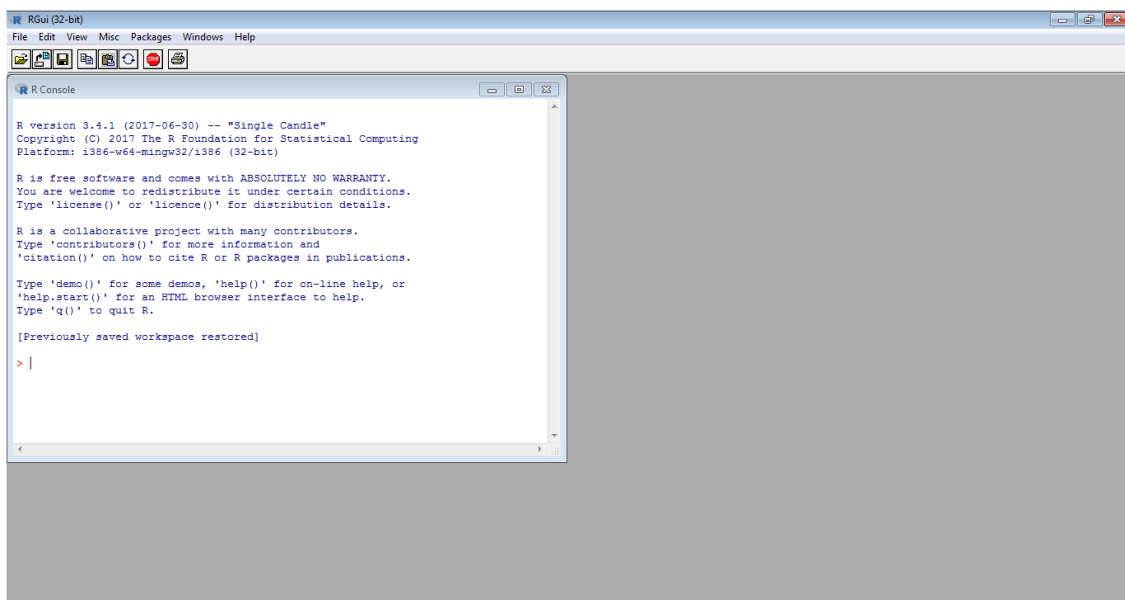
Tujuan utama dari lingkungan R adalah untuk memungkinkan dan mendorong terciptanya analisis data yang baik. Untuk mencapai tujuan ini, R:

1. memberikan fasilitas-fasilitas umum dan mudah digunakan untuk organisasi, penyimpanan dan pemanggilan data.
2. memberikan teknik-teknik komputasi dan metoda-metoda numerik.
3. memungkinkan membuat fungsi-fungsi sesuai dengan keinginan pemakai.
4. memberikan cara interaktif, informatif dan fleksibel untuk memandang data.

R dapat digunakan pada berbagai bidang seperti analisis keuangan, penelitian statistika, manajemen, akademis, matematika, grafik dan analisis data.

Menjalankan R

Untuk menjalankan R klik pada ikon R dan akan muncul tampilan sebagai berikut:



Prompt pada R adalah `>`. Untuk keluar dari R ketikkan

```
>q()
```

atau dengan memilih Exit pada menu File.

Ekspresi dan Data

R dapat diumpamakan sebagai kalkulator yang sangat ampuh. Misalkan untuk mencari hasil dari ekspresi berikut

```
>3*(11.5+2.3)
```

R akan memberikan hasil

```
[1] 41.4
```

Untuk perintah penugasan (*assignment*)

```
>g<-4913
```

R akan membentuk objek dengan nama `g`. Untuk melihat isi objek itu

```
>g
```

dan R akan menampilkan

```
[1] 4913
```

yaitu berarti objek `g` berupa vektor dengan panjang 1 dengan isi 4913. R telah membentuk suatu variabel `g` dengan nilai 4913 di direktorinya. Untuk melihat isi direktori dapat dilakukan dengan

```
>ls()
```

Selama objek-objek tersebut tidak dihapus, objek-objek tersebut akan selalu tersimpan di dalam direktori R.

Untuk menghapus suatu objek `x` di direktori digunakan perintah

```
>rm(x)
```

R dapat bekerja pada seluruh objek secara sekaligus. Operasi dapat dilakukan terhadap objek tanpa merubah isi objek tersebut. Misalnya

```
>g*1000
```

```
[1] 4913000
```

Fungsi

R merupakan bahasa fungsi. Setiap perintah diinterpretasikan sebagai evaluasi fungsi. Komputasi numerik, grafik atau lainnya pada R dilakukan oleh fungsi dengan cara memanggil fungsi tersebut dengan memberikan nama fungsi diikuti/tanpa diikuti oleh argumen-argumen di dalam tanda kurung (). R menyediakan banyak fungsi-fungsi yang siap untuk digunakan. Pemakai juga dimungkinkan untuk mendefinisikan fungsi-fungsi baru yang tidak disediakan R sesuai dengan keinginan pemakai.

Salah satu fungsi yang digunakan untuk membentuk vektor adalah fungsi `c`. Fungsi `c` digunakan untuk mengumpulkan (*to collect*) argumen –argumen membentuk suatu vektor. Misalnya

```
>x<-c(1,3,7,8)
```

membentuk objek yang mempunyai nama `x` berbentuk vektor dengan panjang 4 dengan unsur-unsur 1, 3, 7 dan 8. Sehingga bila kita lihat isi `x`

```
>x
[1] 1 3 7 8
```

Berbagai operasi dapat dilakukan terhadap objek `x` tersebut. Misalnya

```
>x*2
[1] 2 6 14 16
>sqrt(x)
[1] 1.000000 1.732051 2.645751 2.828427
```

Dua contoh perintah di atas memanipulasi setiap unsur pada vektor `x`. Dua contoh berikut

```
>sum(x)
[1] 19
>mean(x)
[1] 4.75
```

memberikan hasil satu nilai. Sedangkan contoh berikut ini

```
>range(x)
[1] 1 8
```

memberikan hasil dua nilai.

Perintah penugasan berikut

```
>y<-1:6
```

adalah cara lain untuk menyatakan perintah

```
>y<-c(1,2,3,4,5,6)
```

Fungsi-fungsi lain pada R diantaranya adalah `sin(x)`, `log(x)`, `sort(x)`, `unique(x)`, `stem(x)` dan `median(x)`.

Operator

R mempunyai operator aritmetika yang sama dengan bahasa pemrograman umum lainnya, yaitu:

```
+      -      *      /      ^
```

Operator tersebut bekerja terhadap objek seperti contoh berikut:

```
>2^c(3,5,10)
```

```
[1] 8 32 1024
```

```
>x<-1:5
```

```
>abs(x-mean(x))
```

```
[1] 2 1 0 1 2
```

R memiliki operator relasi sebagai berikut

```
>      >=     <=     <      ==     !=
```

masing-masing menunjukkan $>$, \geq , \leq , $<$, $=$ dan \neq . Contoh pemakaian operator adalah sebagai berikut

```
>x<-c(3,5,1,6,8)
```

```
>x>5
```

```
[1] F F F T T
```

menunjukkan nilai logika hasil perbandingan masing-masing elemen vektor x dengan 5.

Operator logika and, or dan not pada R masing-masing digunakan simbol

```
&      |      !
```

Ekstraksi Data

Data dapat diekstrak dari suatu objek menggunakan ekspresi dalam tanda kurung [] yang mengikuti objek tersebut. Misalnya

```
>nilai<-c(45,65,76,34,67,80,97,57)

>nilai[1:3]

[1] 46 65 76

>nilai[nilai>65]

[1] 76 67 80 97

>nilai[-1]

[1] 65 76 34 67 80 97 57
```

Grafik

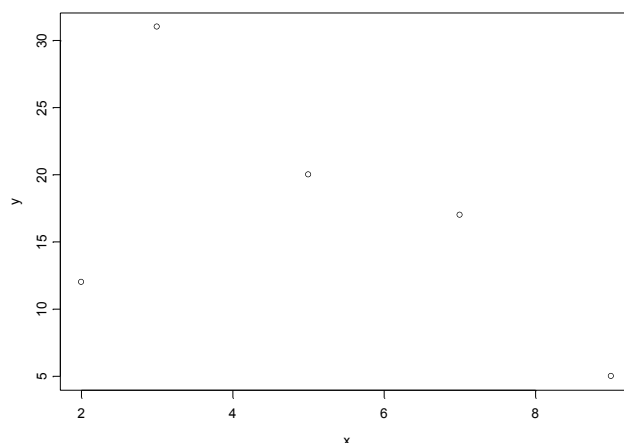
R dapat menampilkan grafik. Penyajian grafik sangat penting dalam memahami data. Misalkan kita melakukan perintah plot sebagai berikut

```
>x<-c(2,5,3,7,9)

>y<-c(12,20,31,17,5)

>plot(x,y)
```

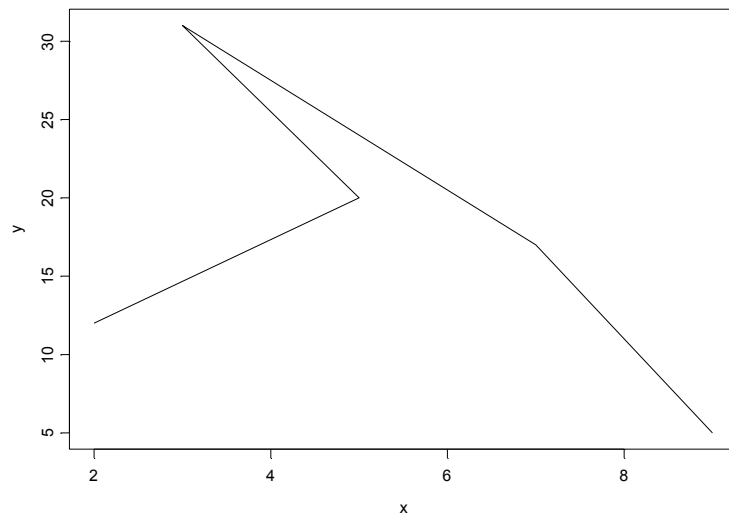
Perintah-perintah di atas akan menghasilkan plot sebagai berikut:



Bila kita ingin menghubungkan titik-titik tersebut dengan garis, dapat dilakukan dengan perintah

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

dan akan menghasilkan plot sebagai berikut



Untuk menggambarkan suatu fungsi, pertama bangkitkan suatu vektor yang berisi nilai-nilai dimana nilai fungsi tersebut akan diplot. Misalkan kita ingin menggambarkan grafik fungsi $\sin(x)$ pada -4π sampai 4π . Maka

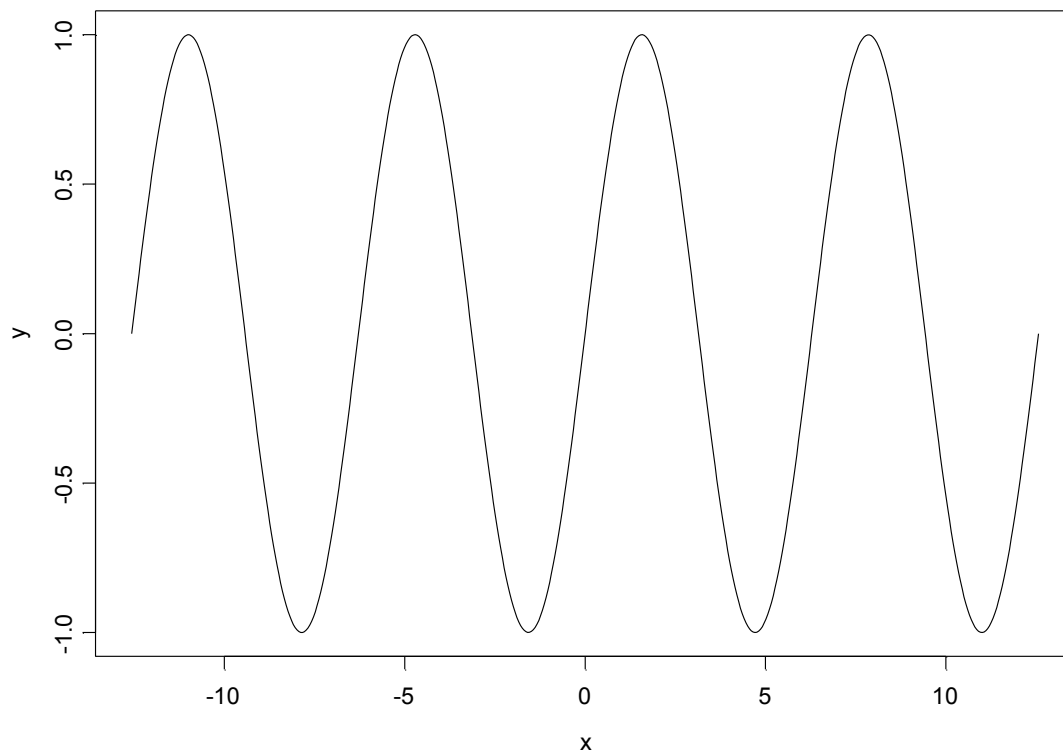
```
>x<-seq(-4*pi,4*pi,length=500)
```

akan menghasilkan suatu vektor x dengan panjang (banyak elemen) 500 dengan nilai dari -4π sampai 4π . Kemudian dihitung nilai $\sin(x)$ dan disimpan di variabel y

```
>y<-sin(x)
```

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

akan menghasilkan plot sebagai berikut:



Penulisan Fungsi

Selain kita dapat menggunakan fungsi-fungsi yang telah disediakan pada R, kita dapat juga membuat fungsi-fungsi yang sesuai dengan tujuan kita. Misalnya kita ingin membuat fungsi akar pangkat tiga dari suatu bilangan. Ini dapat dilakukan dengan cara sebagai berikut

```
>akar_pangkat_tiga<-function(x) x^(1/3)
```

Suatu fungsi dengan nama `akar_pangkat_tiga` telah dibuat dan siap untuk digunakan. Misalnya

```
>akar_pangkat_tiga(8)
```

```
[1] 2
```

```
>akar_pangkat_tiga(c(1,8,27))
```

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

Fungsi dapat pula dibuat untuk menghasilkan nilai logika benar (T) atau salah (F). Misalnya

```
>is.min<-function(x) x==min(x)
```

```
>y<-c(2,4,6,7,2,2,6)
>is.min(y)
[1] T F F F T T F
```

Untuk melihat isi suatu fungsi dapat dilakukan dengan mengetikkan nama fungsi tersebut

```
>akar
function(x)
x^(1/3)
```

Untuk mengedit fungsi dapat dilakukan dengan fungsi `ed`. Misalkan kita ingin mengedit fungsi akar

```
>akar<-ed(akar)
```

Perintah ini akan menampilkan layar untuk mengedit isi fungsi akar.

Latihan:

1. Lakukanlah perhitungan berikut pada R
 - a. $2 + 3 \times (5 - 7^4)$
 - b. $\sqrt{10}$
 - c. $\frac{1}{2\pi} e^{-\frac{1}{2}(3^2)}$
 - d. $\frac{7-5}{(8-3)^2}$
2. Lakukanlah perintah berikut
 - a. Simpanlah vektor dengan elemen-elemen 3, -1, 7, 10, 8, 5, 4, 0, 6, -3 di objek x.
 - b. Hitung $2 \times x$ dan simpan di y
 - c. hitung $x+y$
 - d. buat vektor z dengan elemen bilangan bulat 1 sampai 10
 - e. hitung x/z
 - f. hitung rata-rata x dengan menggunakan `mean(x)`
 - g. hitung jumlah elemen x dengan perintah `sum(x)`
 - h. hitung median dengan perintah `median(x)`
 - i. hitung x^2
 - j. hitung $\log(z)$
3. Lakukanlah perintah berikut untuk mengenal tipe objek
 - a. `nama="Budi"`
 - b. `mode(nama)`
 - c. `nilai=85`
 - d. `mode(nilai)`
 - e. `lulus=nilai>56`

- f. `mode(lulus)`
4. Dari objek `x`, lakukan
 - a. tampilkan elemen ke-1 sampai ke 8 dari vektor `x`, dengan perintah `x[1:8]`
 - b. tampilkan elemen vektor `x` kecuali elemen ke-2
 - c. simpan elemen vektor `x` yang lebih besar nol di objek `w`
 - d. tampilkan elemen `x` yang kurang dari nol atau yang lebih dari 5
5. Lakukanlah perintah berikut
 - a. tentukan panjang dari `x`, gunakan `length(x)`
 - b. tentukan panjang dari `z`
 - c. plotlah `x` dan `z`. lakukan `plot(x,z)` dan `plot(z,x)`. lihat perbedaannya
 - d. ulangi perintah c dengan menambahkan `type="l"`. `plot(x,z,type="l")`
 - e. buatlah gambar persegi dengan panjang satu satuan dengan perintah `plot`
6. Kurva normal adalah grafik dari fungsi kepekatan peluang normal

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$
 - a. buatlah vektor `x` yang berisi sekuens bilangan dari -10 sampai 10 dengan panjang 40
 - b. buatlah vektor `y` yang berisi nilai fungsi `f(x)`. ambil $\mu = 4$ dan $\sigma = 2$
 - c. buatlah plot kurva normal tersebut
7. a. Buatlah fungsi sederhana dalam satu baris untuk menghitung nilai `f(x)` pada 6.c dengan memberi nama fungsi tersebut `normal`.
 - b. ulangi menghitung vektor `y` dengan menggunakan fungsi yang dibuat pada a.

Modul 2

EKSPRESI DAN DATA

Pendahuluan

R adalah suatu bahasa komputer dan merupakan lingkungan pemrograman interaktif untuk analisis data dan grafik. Ketika kita mengetikkan suatu ekspresi, R akan menginterpretasikan dan mengevaluasinya. Ekspresi dapat berupa ekspresi aljabar biasa maupun ekspresi fungsi seperti beberapa contoh berikut:

```
>mean(nilai)

>median(abs(nilai-median(nilai)))

>range(rnorm(1000))

>pnorm(3.7079)

>sqrt(sum((nilai-mean(nilai))^2)/(length(nilai)-1))
```

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

Vektor

Vektor adalah variabel dengan satu atau lebih unsur dengan tipe sejenis. Skalar adalah vektor yang berukuran satu. Untuk membentuk vektor dapat dilakukan dengan tiga cara:

(a) Sekuens

```
y=seq(0,10,by=0.5)
z=seq(0,2*pi,length=100)
```

(b) Fungsi c

```
x=c(2,4,6,2,7,8,10,6)
y=c(3,2,4,7,NA,4,6)
```

(c) Fungsi scan

```
x=scan()
```

Matriks

Objek berupa matriks dapat dibuat dengan perintah

```
A=matrix(c(2,3,1,5,3,8),nrow=2, byrow=T)
```

Array

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

```
x=array(1:24,dim=c(2,3,4))
```

Data frame

Data frame sering digunakan dalam analisis data dengan R. Data frame adalah generalisasi dari matriks dengan kolom yang berbeda mode.

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

Ekspresi dan Operator

Operator aritmetika untuk penjumlahan, pengurangan, perkalian, pembagian dan pangkat adalah

+ - * / ^

Operator perbandingan yang membandingkan dua nilai dan menghasilkan nilai benar/true(T) atau salah/false(F) adalah

< > <= >= == !=

yang berarti <, >, ≤, ≥, =, dan ≠.

Operator logika untuk operasi dan (*and*), atau (*or*) dan tidak (*not*) adalah

& | !

Operator ":" membuat barisan (sekuens) antara dua bilangan dengan step ±1 seperti

```
>8:17
```

```
[1] 8 9 10 11 12 13 14 15 16 17
```

```
>10.5:6.5
```

```
[1] 10.5 9.5 8.5 7.5 6.5
```

Operator numerik berikut digunakan untuk operasi bilangan bulat, yaitu:

```
%/%      %%
```

Operator `%/%` adalah pembagian bulat, yaitu menghasilkan bilangan bulat terbesar yang lebih kecil atau sama dengan pembagian dua bilangan. Sedangkan operator `%%` memberikan sisa pembagian atau operator modulo.

Pernyataan Penugasan (*Assignment Statement*)

Penugasan dibentuk oleh operator “`<-`”. Misalnya

```
>x<-9:23
```

memberikan nama `x` ke suatu objek yang berisi bilangan bulat 9 sampai 23. Penugasan dapat dilakukan secara ganda seperti

```
>y<-x<-0
```

yang memberikan nilai 0 pada objek `x` dan `y`. Tidak seperti ekspresi yang lain, suatu perintah penugasan tidak menampilkan suatu hasil di layar monitor.

Subskrip dan Indeks

Data dapat diekstraksi dari suatu objek dengan cara:

```
x[subskrip]
```

Nilai dari subskrip dapat berupa:

1. Bilangan bulat positif: yang menyatakan elemen dari objek `x`. Misalnya `x[1]` menyatakan nilai pertama dari `x`, atau `x[1:5]` untuk nilai pertama sampai kelima dari `x`.
2. Nilai logika (T atau F): elemen `x` untuk nilai logika benar (T) yang akan dipilih. Misalnya `x[x>0]` atau `nama[tinggi>170 & umur<30]`
3. Bilangan bulat negatif: semua elemen kecuali elemen yang diberikan pada subskrip akan dipilih. Misalnya `x[-2]` memberikan semua elemen `x` kecuali elemen yang kedua.

Fungsi (*Function*)

Semua perintah di R dilakukan menggunakan fungsi, seperti:

<code>mean(x)</code>	menghitung rata-rata
<code>plot(x, y)</code>	plot x vs y
<code>c(1, -2.1, 3.2, 5.78)</code>	membentuk vektor
<code>runif(5)</code>	membangkitkan bilangan acak antara 0 dan 1

Fungsi dipanggil atau dijalankan dengan mengetikkan nama fungsi diikuti dengan argumen-argumen di dalam tanda kurung:

```
>nama.fungsi(arg1, arg2, ...)
```

Misalkan `mean(x)` memanggil fungsi `mean` untuk menghitung rata-rata dari elemen dari `x`. Argumen bisa tidak ada atau bersifat opsional.

Ada lebih dari 800 fungsi dan package yang telah disediakan pada R yang dapat digunakan untuk analisis data dan grafik. Bila kita memerlukan fungsi untuk tujuan tertentu yang tidak terdapat pada R, kita dapat membuat atau mendefinisikan fungsi baru.

Manipulasi Data

Fungsi `c` menggabungkan semua argumen membentuk suatu vektor. Contoh

```
>bil.prima<-c(1:3, 5, 7, 11)
>bil.prima
[1] 1 2 3 5 7 11
>2^c(1, 3, 5)
[1] 2 8 32
>score<-c(1, NA, 27.5)
>score
[1] 1 NA 27.5
```

NA berarti “Not Available” atau “missing”. Seperti pembagian dengan nol.

Fungsi `rep` mengulang argumen pertama sebanyak argumen kedua. Contoh

```
>rep(1:3, 2)
```

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

Fungsi `seq` adalah bentuk umum dari operator ":" . Ekspresi `seq(m,n)` ekuivalen dengan `m:n`. Ekspresi `seq(x)` ekuivalen dengan `1:x`.

Fungsi ini dapat juga digunakan dengan argumen `by`. Misalnya

```
>seq(-1,1,by=.1)
```

```
[1] -1.0 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1
[11] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9
[21] 1.0
```

```
>seq(-pi,pi, length=10)
```

```
[1] -3.1415927 -2.4434610 -1.7453293 -1.0471976 -0.3490659
[6] 0.3490659 1.0471976 1.7453293 2.4434610 3.1415927
```

Fungsi `sort` mengurutkan data pada argumennya. Misalkan

```
>x<-c(3,2,6,4,5,2,7,8,5,7)
```

```
>sort(x)
```

```
[1] 2 2 3 4 5 5 6 7 7 8
```

R mempunyai fungsi-fungsi matematika seperti:

`sqrt`, `abs`, `sin`, `cos`, `tan`, `asin`, `acos`, `atan`, `sinh`, `cosh`, `tanh`, `asinh`, `acosh`, `atanh`, `exp`, `log`, `log10`, `gamma`, dan `lgamma` dll

Selain itu terdapat fungsi-fungsi dasar, yaitu: `ceiling`, `floor`, `trunc`, `round`, dan `signif`.

Import Data dari Excel

- Perintah `scan()`
- import dari excel
- dll

Pembangkitan Bilangan Acak

Bilangan acak dengan sebaran (distribusi) tertentu dapat dibangkitkan menggunakan fungsi R. Misalkan kita ingin membangkitkan bilangan acak yang berdistribusi seragam antara 0 dan 1, yaitu dengan perintah

```
>runif(100)
```

Untuk membangkitkan bilangan acak yang terdistribusi normal baku digunakan

```
>rnorm(100)
```

Perintah

```
>rt(75,2)
```

adalah untuk membangkitkan 75 bilangan acak yang terdistribusi menurut sebaran-t dengan derajat bebas 2.

Kode	Sebaran
beta	Beta
cauchy	Cauchy
chisq	Chi-square
exp	exponential
f	F
gamma	Gamma
lnorm	Log-normal
logis	logistic
norm	normal
stab	stable
t	Student's t
unif	uniform

Untuk membangkitkan bilangan acak dengan sebaran tertentu dilakukan dengan mengetikkan nama fungsi yang diawali “r”+kode sebaran yang sesuai. Selain itu kita dapat juga mendapatkan fungsi untuk menghitung peluang, kuantil dan nilai fungsi kepekatan peluang dengan mengetikkan “p”, “q” dan “d” + kode sebaran. Misalnya

```
>pnorm(2.5)
```

```
[1] 0.9937903
```

```
>qt(0.95,7)
```

```
[1] 1.89508
```

Latihan:

1. Buatlah vektor:

(a) (1, 2, 3, ..., 19, 20)

(b) (20,19, ..., 2, 1)

(c) (1, 2, 3, ..., 19, 20, 19, 18, ..., 2, 1)

(d) (4, 6, 3) dan masukkan kedalam objek dengan nama tmp.

Untuk (e), (f) dan (g) gunakan fungsi rep.

- (e) (4, 6, 3, 4, 6, 3, ..., 4, 6, 3) dimana terdapat 10 kali muncul 4.
- (f) (4, 6, 3, 4, 6, 3, ..., 4, 6, 3, 4) dimana terdapat 11 kali muncul 4. 10 kali muncul 6, dan 10 kali muncul 3.
- (g) (4, 4, ..., 4, 6, 6, ..., 6, 3, 3, ..., 3) dimana 10 kali muncul 4, 20 kali muncul 6, dan 30 kali muncul 3.
2. Buatlah vektor dari nilai-nilai $e^x \cos(x)$ untuk $x = 3, 3.1, 3.2, \dots, 6$.
3. Buatlah vektor-vektor:

(a) $(0.1^3 \cdot 0.2^1, 0.1^6 \cdot 0.2^4, \dots, 0.1^{36} \cdot 0.2^{34})$ (b) $\left(2, \frac{2^2}{2}, \frac{2^3}{3}, \dots, \frac{2^{25}}{25}\right)$

4. Hitunglah $\sum_{i=10}^{100} (i^3 + 4i^2)$

5. Misalkan

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 5 & 2 & 6 \\ -2 & -1 & -3 \end{bmatrix}$$

- (a) Dapatkan invers dari matriks A dengan fungsi `solve`.
- (b) Periksa bahwa $A^3 = \mathbf{0}$ dimana $\mathbf{0}$ adalah matriks nol berukuran 3×3 .
- (c) Gantilah elemen-elemen kolom ketiga dengan jumlah elemen-elemen kolom kedua dan ketiga
6. Buatlah matriks

$$A = \begin{bmatrix} 2 & 1 \\ -3 & 4 \end{bmatrix} \text{ dan } B = \begin{bmatrix} 3 & 2 \\ 5 & -1 \end{bmatrix}$$

Lakukan $A * A$ pada prompt `>`. Jelaskan operasi apa yang terjadi!

7. Buatlah vektor $x=(0,1,2,3,4)$. Lakukan $x \% \circ \% x$ pada prompt `>`. Jelaskan operasi apa yang terjadi!
8. Lihatlah help untuk fungsi `outer`. Buatlah matriks berikut menggunakan fungsi tersebut.

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 6 & 7 \\ 4 & 5 & 6 & 7 & 8 \end{pmatrix}$$

9. Buatlah matriks B terdiri dari 15 baris:

$$B = \begin{bmatrix} 10 & -10 & 10 \\ 10 & -10 & 10 \\ \vdots & \vdots & \vdots \\ 10 & -10 & 10 \end{bmatrix}$$

Hitunglah $B^T B$. (a) dengan menggunakan perkalian matriks. (b) dengan menggunakan fungsi `crossprod`.

10. Selesaikanlah sistem persamaan linier berikut:

$$x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 = 7$$

$$2x_1 + x_2 + 2x_3 + 3x_4 + 4x_5 = -1$$

$$3x_1 + 2x_2 + x_3 + 2x_4 + 3x_5 = -3$$

$$4x_1 + 3x_2 + 2x_3 + x_4 + 2x_5 = 5$$

$$5x_1 + 4x_2 + 3x_3 + 2x_4 + x_5 = 17$$

dengan menuliskan dalam bentuk persamaan matriks $Ax = g$

11. Bacalah data berikut: 75 56 60 83 75 67 90 56 64 66

(a) melalui keyboard dengan menggunakan fungsi `scan()`

(b) membaca dari file ASCII.

12. Bacalah data berikut dengan menggunakan `read.table`

No.	Nama	Matematika	Fisika
1	Amir	78	80
2	Budi	67	75
3	Rahmi	80	85
4	Rini	60	60
5	Rina	50	62
7	Ihsan	55	63
8	Hani	70	72
9	Khairul	83	95
10	Imam	74	82

Buatlah diagram pencar dengan sumbu datar nilai matematika dan sumbu tegak nilai fisika.

Modul 3

STATISTIKA DESKRIPTIF

Pendahuluan

Statistika deskriptif adalah metode-metode yang berkaitan dengan pengumpulan dan penyajian data sehingga memberikan informasi yang berguna. Kombinasi komputasi yang bersifat interaktif dan penampilan grafik pada R memberikan fasilitas yang baik untuk memahami data. Untuk mempelajari lebih lanjut berbagai fasilitas R dapat dimanfaatkan link Rseek.org.

Menghitung Statistik

Untuk menghitung beberapa statistik digunakan beberapa fungsi yang telah tersedia dalam R. Beberapa fungsi dasar dapat digunakan seperti

mean	rata-rata
median	median
var	ragam(variansi)
range	nilai minimum dan maksimum
quantile	persentil, desil, kuartil
summary	min, Q1, median, mean, Q3, max

Fungsi summary akan memberikan nilai minimum, kuartil 1, median, rata-rata, kuartil 3 dan nilai maksimum.

```
> x<-c(2,2,0,1,3,3,2,2,2,2)
> summary(x)

Min. 1st Qu. Median Mean 3rd Qu. Max.
0.0   2.0     2.0   1.9   2.0     3.0
```

Menampilkan Grafik

Penyajian data dalam bentuk diagram atau grafik digunakan untuk membuat data lebih dapat dipahami karena penyajian secara visual lebih mudah dilihat.

Diagram dahan daun

Diagram dahan daun (stem-and-leaf display) dari data dapat dilakukan dengan

```
>stem(x)
```

Misalkan

```
>nilai.matematika<-c(67,74,58,90,82,66,86,45,95,74)
```

```
>nilai.fisika<-c(72,65,66,86,72,56,79,80,89,56)
```

```
>stem(nilai.matematika)
```

N = 10 Median = 74

Quartiles = 66, 86

Decimal point is 1 place to the right of the colon

```
4 : 5
5 : 8
6 : 67
7 : 44
8 : 26
9 : 05
```

>

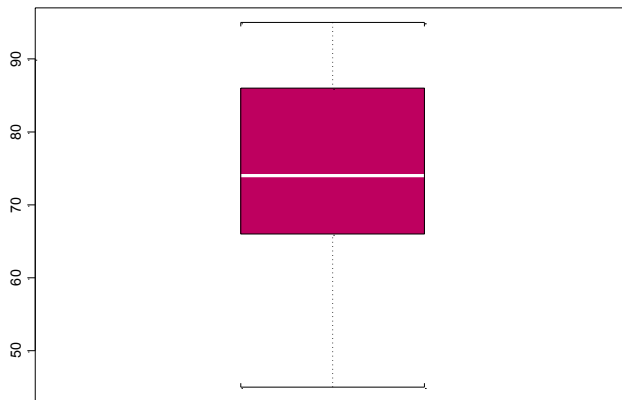
Diagram kotak garis

Untuk membandingkan beberapa kelompok data sering digunakan diagram kotak garis (boxplot). Untuk membuat diagram kotak garis suatu kelompok data digunakan

```
>boxplot(x)
```

Misalkan

```
>boxplot(nilai.matematika, ylab="Nilai")
```

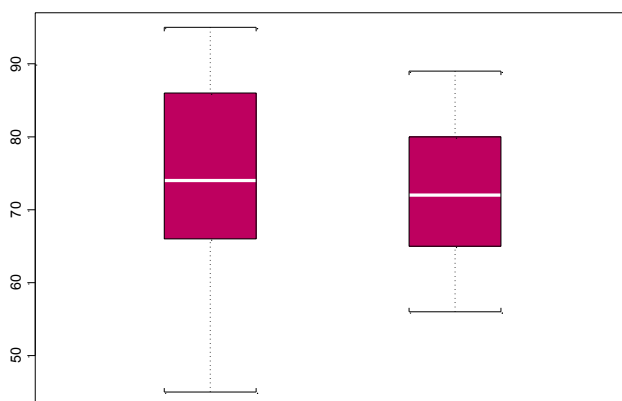


Sedangkan bila kita ingin membandingkan beberapa kelompok data, misalkan x , y dan z, digunakan

```
>boxplot(x,y,z)
```

Dari data yang di atas kita dapat melakukan pembandingan nilai matematika dan fisika dengan perintah

```
> boxplot(nilai.matematika,nilai.fisika,
xlab="Pelajaran", ylab="Nilai", main="Nilai Matematika dan
Fisika")
```



Histogram

Untuk membuat histogram dari data digunakan

```
>hist(x, xlab="label x",ylab="Frekuensi")
```

Fungsi ini mempunyai argumen untuk mengatur bentuk tampilan histogram.

Diagram pencar

Untuk membuat diagram pencar (*scatter plot*) dari data berpasangan (x,y) digunakan fungsi

```
>plot(x,y)
```

Fungsi plot ini mempunyai argumen opsional, yaitu type yang mengatur bentuk plot. Misalkan

```
>plot(x,y,type="l",xlab="label sumbu-x", ylab="label sb-y", main="judul")
```

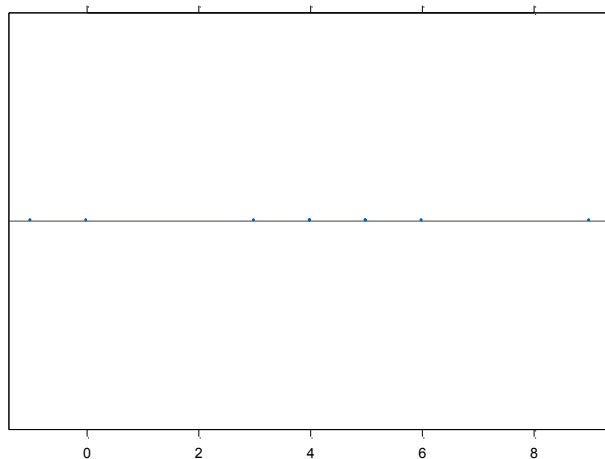
menghasil plot dengan type garis, yaitu setiap titik dihubungkan dengan garis sesuai urutan data.

Dotplot

Fungsi stripchart digunakan untuk menampilkan data pada garis bilangan. Misalnya

```
>x<-c(3,-1,4,5,4,9,6,5,4,0)
```

```
>stripchart(x,method="stack")
```

*Deret waktu*

Untuk memplot data yang berupa deret waktu digunakan plot.

```
>plot(x)
```

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

```
>plot(x,type="l",main="Judul",xlab="t",ylab="y")
```

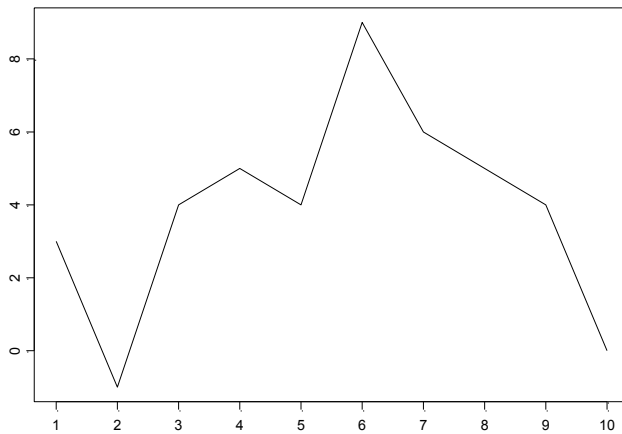
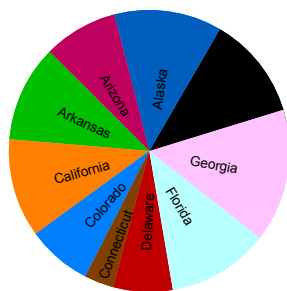


Diagram Lingkaran

Untuk membuat diagram lingkaran digunakan perintah pie.

```
>pie(murder, names=state)
```

menampilkan diagram lingkaran berdasarkan data pada objek murder dan label setiap potongan dengan nama-nama pada objek state.



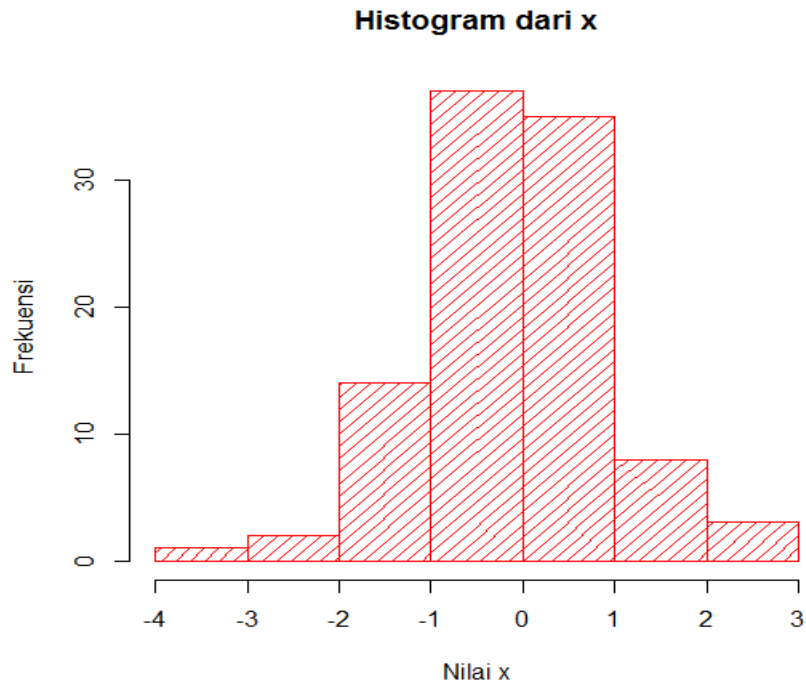
Histogram

Untuk membuat histogram digunakan fungsi hist.

Misalkan untuk data yang telah disimpan pada objek berbentuk vektor x

```
>x<-rnorm(100)
```

```
> hist(x, xlab="Nilai x", ylab="Frekuensi", main="Histogram dari x", density=15,col="red")
```

**Latihan:**

1. Perhatikan data berikut:

	Bukan Perokok	Perokok Sedang	Perokok Berat
Hipertensi	21	36	30
Tidak Hipertensi	48	26	19

- Buatlah matriks berukuran 2 x 3 menggunakan data tersebut. Simpan pada objek dengan nama dataset1.
- Beri nama baris menggunakan fungsi row.names.
- Beri nama kolom menggunakan fungsi colnames.
- Buatlah diagram batang dengan perintah berikut


```
> barplot(dataset1, main="Distribusi Pengidap Hipertensi menurut Kebiasaan Merokok",
        xlab="Kebiasaan Merokok", ylab="Jumlah", col=c("red", "blue"),
        legend=rownames(dataset1), beside=TRUE)
> abline(h=0)
```

2. Perhatikan data berikut:

Jenis Film	Komedi	Aksi	Roman	Drama	Fiksi
Jumlah Penggemar	4	5	6	1	4

- Buatlah vektor dengan nama jml.penggemar yang berisi jumlah penggemar.
- Buatlah vektordengan nama jenis.film yang berisi jenis film.
- Buatlah diagram lingkaran dengan perintah berikut:


```
> pie(jml.penggemar, labels=jenis.film, main="Diagram Lingkaran Penggemar Jenis Film")
```
- Buatlah diagram lingkaran dengan menampilkan jumlah penggemar dan persentasenya.

3. Lakukan perintah scan untuk membaca data yang telah disediakan berupa ASCII file dengan nama dataiq.txt. Data tersebut berupa IQ anak dan mental ibunya (ND=non-depressed, D=depressed).

- (a) Buatlah 2 boxplot berdampingan IQ anak untuk ibu ND dan D.
 (b) Buatlah histogram untuk IQ anak dengan ibu ND dan D. Sebelah atas untuk ND, sebelah bawah untuk D. Gunakan perintah `par(mfrow=c(2,1))`

4. Berikut ini adalah data panjang danau terbesar di dunia.

Nama danau	Benua	Panjang (dalam mil)
Caspian Sea	Asia-Eropa	760
Tanganyika	Afrika	420
Baykal	Asia	395
Balkhash	Asia	376
Malawi	Afrika	360
Superior	Amerika Utara	350
Michigan	Amerika Utara	307
Great Slave	Amerika Utara	298
Aral Sea	Asia	280
Winnipeg	Amerika Utara	266
Victoria	Afrika	250
Erie	Amerika Utara	241
Huron	Amerika Utara	206
Ontario	Amerika Utara	193
Great Bear	Amerika Utara	192

- (a) Buatlah data frame untuk data di atas.
 (b) Buatlah dotplot untuk panjang danau tersebut.
 (c) Buatlah plot dengan fungsi `dotchart`.

5. Perhatikan data berikut.

Nilai Tes, x	Nilai Akhir, y	Nilai Tes, x	Nilai Akhir, y
50	53	90	54
35	41	80	91
35	61	60	48
40	56	60	71
55	68	60	71
65	36	40	47
35	11	55	53
60	70	50	68
90	79	65	57
35	59	50	79

- (a) Buatlah diagram pencar dengan menggunakan fungsi `plot`. Sumbu datar untuk nilai tes dan sumbu tegak untuk nilai akhir.
 (b) Tambahkan pada gambar tersebut garis regresinya.
 (c) Buatlah plot sisaan (residuals) vs nilai dugaan.

- (d) Buatlah histogram sisaan.
- (e) Buatlah normal probability plot untuk sisaan.

Modul 4

STATISTIKA INFERENSIA

Pendahuluan

Statistika inferensia adalah metode-metode yang berhubungan dengan analisis sebagian data untuk kemudian sampai pada peramalan atau penarikan kesimpulan mengenai keseluruhan data (populasi). Statistika inferensia terbagi dua yaitu pendugaan parameter dan pengujian hipotesis.

Pengujian Hipotesis Rata-rata

Perintah pada R untuk menguji hipotesis mengenai satu atau dua populasi adalah

```
t.test(x, y=NULL, alternative="two.sided", mu=0, paired=F,  
       var.equal=F, conf.level=.95)
```

a. Satu Populasi

Misalkan akan diuji hipotesis mengenai nilai tengah satu populasi sebagai berikut:

$$H_0 : \mu = 80$$

$$H_1 : \mu \neq 80$$

Nilai-p adalah peluang bahwa nilai mutlak dari T lebih besar dari t-hitung. Kesimpulan: Tolak H_0 bila nilai-p lebih kecil dari taraf nyata yang ditetapkan. Misalkan data telah disimpan di objek x. Pengujian hipotesis dengan taraf nyata 0,05 dapat dilakukan dengan perintah

```
>t.test(x, mu=80)
```

Untuk hipotesis alternatif

$$H_1 : \mu < 80$$

digunakan perintah

```
> t.test(x, alternative="less", mu=80)
```

Sedangkan bila hipotesis alternatif

$$H_1 : \mu > 80$$

digunakan perintah

```
> t.test(x, alternative="greater", mu=80)
```

b. Dua Populasi

Untuk menguji nilai tengah dua populasi dengan hipotesis nol

$$H_0 : \mu_1 = \mu_2$$

dan hipotesis alternatif

$$H_1 : \mu_1 \neq \mu_2$$

digunakan perintah

```
>t.test(x,y)
```

Untuk hipotesis alternatif

$$H_1 : \mu_1 < \mu_2$$

digunakan perintah

```
>t.test(x,y,alternative="less")
```

Sedangkan untuk hipotesis alternatif

$$H_1 : \mu_1 > \mu_2$$

digunakan perintah

```
>t.test(x,y,alternative="greater")
```

Bila ragam kedua populasi diasumsikan sama maka perintah yang digunakan adalah

```
>t.test(x,y,alternative="greater", var.equal=T)
```

c. Data Berpasangan

Untuk pengujian data berpasangan dengan hipotesis

$$H_0 : \mu_D = 10$$

melawan hipotesis tandingan

$$H_1 : \mu_D \neq 10$$

digunakan perintah

```
>t.test(x,y,mu=10,paired=T)
```

Untuk

hipotesis

alternatif

$$H_0 : \mu_D < 10$$

digunakan perintah

```
>t.test(x,y, alternative="less", mu=10,paired=T)
```

Untuk hipotesis alternatif
 $H_0 : \mu_D > 10$

digunakan perintah

```
>t.test(x,y, alternative="greater", mu=10, paired=T)
```

Regresi Linier Sederhana

Analisis regresi linier sederhana meliputi pendugaan parameter regresi pada model yang melibatkan satu peubah bebas X dan satu peubah tak bebas Y berikut:

$$Y_i = \alpha + \beta X_i + \varepsilon_i$$

Setelah Nilai-nilai pengamatan X dan Y disimpan pada 2 objek yang berbentuk vektor dengan panjang yang sama, maka penghitungan dugaan parameter regresi tersebut dilakukan dengan perintah:

```
>lsfit(x,y)
```

Selain menghasilkan output koefisien regresi a dan b, perintah ini juga menghasilkan output sisaan (residual).

Selain dengan perintah lsfit, analisis regresi dapat juga dilakukan dengan perintah berikut

```
>lm(y~x)
```

Bila output ini disimpan pada objek dengan nama fit1, maka dilakukan perintah berikut

```
>fit1<-lm(y~x)
```

Kemudian untuk menampilkan informasi tentang penyebaran sisaan, koefisien regresi, standar error dan uji signifikan dari peubah tak bebas dalam model digunakan perintah

```
>summary(fit1)
```

Untuk menampilkan garis regresi terlebih dahulu di plot data dengan perintah

```
>plot(x,y)
```

dan kemudian

```
>abline(fit1)
```

Untuk melihat kecocokkan garis regresi terhadap data diperlukan untuk memplot nilai dugaan terhadap sisaan dengan perintah

```
>plot(fitted(fit1), residuals(fit1))
```

Apabila model yang digunakan adalah:

$$Y_i = \beta X_i + \varepsilon_i$$

maka perintah yang digunakan untuk mendapatkan nilai dugaan parameter adalah:

```
>lsfit(x,y,intercept=F)
```

Korelasi Linier

Untuk mengukur keeratan hubungan linier antara dua peubah X dan Y digunakan koefisien korelasi Pearson

$$r = \frac{n \sum_{i=1}^n x_i y_i - (\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{\sqrt{[n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2][n \sum_{i=1}^n y_i^2 - (\sum_{i=1}^n y_i)^2]}}$$

Pada R koefisien korelasi antara peubah X dan Y dapat diperoleh dengan mengetikkan

```
>cor(X,Y)
```

Untuk menguji hipotesis mengenai korelasi

$$H_0 : \rho = 0$$

$$H_1 : \rho > 0$$

digunakan perintah

```
>cor.test(x,y, alt="g")
```

Latihan:

1. Perhatikan data berikut:

	Masa Putar (menit)						
Perusahaan1	102	86	98	109	92		
Perusahaan2	81	165	97	134	92	87	114

- (a) Ujilah apakah ragam kedua populasi tersebut sama atau berbeda.
 (b) Lakukan uji hipotesis apakah nilai tengah masa putar film perusahaan 2 lebih 10 menit dari nilai tengah masa putar film perusahaan 1 dengan menggunakan fungsi t.test. Gunakan taraf nyata 0.1.
2. Data berikut adalah konsumsi BBM (km/ltr) menggunakan ban radial dan ban biasa yang dicobakan pada 9 mobil.

	1	2	3	4	5	6	7	8	9
Radial	14.2	14.7	16.6	17.0	16.7	14.5	15.7	16.0	17.4
Biasa	14.1	14.9	16.2	16.9	16.8	14.4	15.7	15.8	16.9

Pada taraf nyata 0.025, dapatkah disimpulkan bahwa mobil dengan ban radial lebih hemat bahan bakar daripada mobil dengan ban biasa?

3. Dengan menggunakan data no.4 pada Latihan modul 3, lakukan
- (a) analisis regresi dengan menggunakan fungsi `lsfit`.
 - (b) analisis regresi dengan menggunakan fungsi `lm`.
 - (c) ulangi (b) dengan menyimpan hasilnya dengan nama `fit1`. Kemudian tampilkan hasil analisis menggunakan fungsi `summary`.
 - (d) Sediakan tempat grafik 2 baris dan 2 kolom dengan perintah `par(mfrow=c(2,2))`. Lakukan perintah `plot(fit1)`.
 - (e) analisis korelasi untuk mendapatkan nilai koefisien korelasi dan ujilah apakah koefisien korelasi tersebut signifikan? Gunakan taraf nyata 0.05. Gunakan fungsi `cor` dan `cor.test`.

4. Perhatikan data berikut:

Panjang bayi (cm), y	Umur (hari), x1	Bobot lahir (kg), x2
57.5	78	2.75
52.8	69	2.15
61.3	77	4.41
67.0	88	5.52
53.5	67	3.21
62.7	80	4.32
56.2	74	2.31
68.5	94	4.30
69.2	102	3.71

- (a) Simpanlah data tersebut dalam bentuk data frame.
 - (b) Lakukan analisis regresi linier berganda dengan menggunakan perintah `lm`.
 - (c) Buatlah plot untuk semua pasangan variabel tersebut menggunakan fungsi `pairs`.
5. **Uji Kebebasan.** Uji kebebasan digunakan untuk menguji hipotesis kebebasan antara dua variabel. Uji kebebasan dalam R menggunakan fungsi `chisq.test` (lihat help untuk fungsi ini). Gunakan data no.1 pada Latihan Modul 3 untuk menguji apakah terdapat kaitan antara kebiasaan merokok dengan penyakit hipertensi.

Modul 5

PEMROGRAMAN DALAM R

Pendahuluan

R menyediakan fasilitas untuk membuat fungsi yang didefinisikan oleh pengguna (*user-defined function*). Dengan fasilitas ini memungkinkan pengguna untuk membuat program analisis yang lebih fleksibel dengan menggunakan fungsi-fungsi *built-in* di dalam R.

Fungsi built-in adalah fungsi yang telah tersedia dalam R seperti

```
mean(x)    # rata-rata aritmetika
```

```
plot(x, y) # plot x dan y
```

```
c(2, -3, 5, 5, 4.47, 6.02)    # menggabung data menjadi vektor
```

```
runif(5)    # 5 bilangan acak antara 0 dan 1
```

Secara umum, fungsi dipanggil/dijalankan dengan mengetikkan nama fungsi diikuti oleh argumen dalam tanda kurung

```
fun.name(arg1, arg2, ...)
```

Perintah di atas memanggil fungsi dengan nama `fun.name`, dan `arg1`, `arg2`, ... adalah argumen-argumen yang diperlukan. Misalnya `mean(x)` memanggil fungsi `mean` untuk menghitung rata-rata aritmetik dari unsur-unsur `x`. Argumen ada yang bersifat opsional atau tidak harus diberikan terutama pada fungsi-fungsi rumit yang dibuat oleh pengguna tingkat lanjut.

Penulisan Fungsi

Secara umum, struktur penulisan fungsi di dalam R adalah sebagai berikut.

```
nama_fungsi=function(arg1, arg2, ...)  
{  
..... isi dari fungsi  
}
```

Penulisan dapat dilakukan dalam 2 cara, yaitu melalui R-console dan R-editor. Penulisan melalui R-console dilakukan dengan menuliskan fungsi langsung pada prompt >. Misalkan akan dibuat fungsi untuk menghitung rata-rata harmonik dengan rumus

$$HM = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}$$

Anggaplah fungsi ini tidak ada tersedia dalam R. Maka fungsi tersebut dapat kita tulis sebagai berikut.

```
> harmonic.mean<-function(x)
+ {
+   n<-length(x)      # menghitung banyak pengamatan
+   jumlah<-sum(1/x)
+   rata.harmonik<-n/jumlah
+ }
```

Pada contoh di atas, nama fungsi adalah harmonic.mean dan argumennya adalah x yang berbentuk vektor.

```
> x=c(2,1,3,4,5)
> harmonic.mean(x)
[1] 2.189781
```

Sedangkan melalui R-editor, pendefinisian fungsi dilakukan dengan langkah-langkah sebagai berikut.

- Memunculkan R-editor dengan klik File > New Script
- Mengetikkan fungsi pada R-editor.
- Simpan fungsi tersebut.
- Membuat source R code dengan klik File > Source R code dan memilih file yang telah disimpan.

Tipe Data dan Operator

- Tipe Data: numerik, karakter/string, logika
- Operator Aritmetika
- Operator Logika

Aliran Pengendalian (Control Flow)

1. Statemen **if**

```
if (kondisi) {ekspresi}
```
2. Statemen **if-else**

```
if (kondisi) {ekspresi1} else {ekspresi2}
```
3. Statemen **ifelse**

```
ifelse(kondisi, expr1, expr2)
```
4. Statemen **for**

```
for (name in expr1) {expr2}
```
5. Statemen **break** dan **next**

`break` : stop dan keluar loop yg sedang dieksekusi

`next` : stop iterasi yang sedang berjalan dan langsung mulai iterasi selanjutnya
6. Statemen **return** dan **stop**

`return(expr)` : stop fungsi yang sedang diakses atau dievaluasi dan munculkan output nilai dari `expr`.

`stop(message)` : digunakan untuk memberikan tanda adanya kesalahan dengan menghentikan evaluasi dari fungsi yang sedang diakses dan menampilkan message.
7. Statemen **repeat**

```
repeat {expr}
```
8. Statemen **while**

```
while (condition) {expr}
```

Argumen dari suatu fungsi

1. Optional argument
2. Required argument

Pengaturan tampilan output

1. Fungsi tab dan newline
 - `\t` : untuk tab
 - `\n` : untuk mengganti baris
2. Perintah `print`
3. Perintah `format`
4. Perintah `cat`
5. Perintah `paste`

Dalam membuat fungsi, sebaiknya dihindari penggunaan *looping*. Operasi vektor dalam R dilakukan secara *elementwise*. Dengan demikian operasi dilakukan secara lebih cepat dan efisien. Hal ini dapat dilakukan dengan memanfaatkan fungsi-fungsi yang

tersedia dalam R (*built-in function*) untuk operasi vektor maupun matriks seperti `sum`, `mean`, `prod`, `apply`, dll. Sebagai ilustrasi,

- Untuk menjumlahkan dua buah vektor x dan y yang memiliki panjang n , lebih baik menggunakan $z=x+y$ daripada `for (i in 1:n) z[i]<-x[i]+y[i]`.
- Untuk mengalikan elemen-elemen dari suatu vektor (x_1, x_2, \dots, x_n) , dapat digunakan fungsi `prod`. Misalkan fungsi `rata.geom` untuk menghitung rata-rata geometrik dapat dituliskan sebagai berikut.

```
rata.geom<-function(x) prod(x)^(1/length(x))
```

List

Output suatu fungsi dapat berupa *list*. Untuk membuat *list* digunakan perintah `list`. List merupakan kumpulan dari berbagai objek. Contoh:

```
> x.lis<- list(a=1:10,b=letters[1:3],c=matrix(1:10,ncol=2))
```

Objek dengan nama `x.lis` berupa list dengan isi `a` berupa vektor numerik, `b` berupa vektor karakter dan `c` berupa matriks. Untuk mengambil nilai masing-masing adalah sebagai berikut.

```
> x.lis$a
[1] 1 2 3 4 5 6 7 8 9 10
> x.lis$b
[1] "a" "b" "c"
> x.lis$c
      [,1] [,2]
[1,]    1    6
[2,]    2    7
[3,]    3    8
[4,]    4    9
[5,]    5   10
```

Dalam contoh fungsi berikut outputnya berupa list.

```
> rumusABC<-function(A,B,C){
  D<-B^2-4*A*C
  x1<-(-B-sqrt(D))/(2*A)
  x2<-(-B+sqrt(D))/(2*A)
  akar<-list(akar1=x1,akar2=x2)
  return(akar)
}
> rumusABC(3,5,1)
$akar1
[1] -1.434259
```

```

$akar2
[1] -0.2324081

> hasil<-rumusABC(3,5,1)
$akar1
[1] -1.434259

$akar2
[1] -0.2324081

> hasil$akar1
[1] -1.434259
> hasil$akar2
[1] -0.2324081
>

```

Latihan.

1. Tuliskanlah fungsi `geo.mean` untuk menghitung rata-rata geometrik dari `xVec` yang merupakan vektor (x_1, x_2, \dots, x_n) .

Rumus rata-rata geometrik:

$$GM = \sqrt[n]{x_1 x_2 \dots x_n}$$

Hitunglah rata-rata geometrik dari 2, 3, 1, 4, 5.

2. Tuliskan fungsi `tmpFn1` dan `tmpFn2` sedemikian sehingga jika `xVec` adalah vektor (x_1, x_2, \dots, x_n) , maka `tmpFn1(xVec)` akan menghasilkan output vektor $(x_1, x_2^2, \dots, x_n^n)$ dan `tmpFn2(xVec)` akan menghasilkan vektor $(x_1, \frac{x_2^2}{2}, \dots, \frac{x_n^n}{n})$.
3. Tuliskan fungsi `tmpFn3` dengan 2 argumen `x` dan `n` dimana `x` adalah suatu bilangan tunggal dan `n` adalah bilangan bulat positif. Fungsi tersebut menghasilkan nilai

$$1 + \frac{x}{1} + \frac{x^2}{2} + \dots + \frac{x^n}{n}$$

4. Tuliskan fungsi `tmpFn4(xVec)` sedemikian sehingga jika `xVec` adalah vektor $\mathbf{x} = (x_1, x_2, \dots, x_n)$ maka `tmpFn4(xVec)` menghasilkan vektor rata-rata bergerak:

$$\frac{x_1 + x_2 + x_3}{3}, \frac{x_2 + x_3 + x_4}{3}, \dots, \frac{x_{n-2} + x_{n-1} + x_n}{3}$$

Ujilah fungsi tersebut, misalnya dengan menjalankan `tmpFn4(c(1:5, 6:1))`.

5. Perhatikan fungsi berikut

$$f(x) = \begin{cases} x^2 + 2x + 3, & x < 0 \\ x + 3, & 0 \leq x < 2 \\ x^2 + 4x - 7, & 2 \leq x \end{cases}$$

Tuliskan fungsi `tmpFn5` dengan argumen `xVec`. Fungsi tersebut menghasilkan vektor nilai-nilai fungsi $f(x)$ yang dihitung dari nilai-nilai `xVec`. Plotlah fungsi $f(x)$ untuk $-3 < x < 3$.

6. Buatlah fungsi untuk mendapatkan persamaan regresi linier sederhana $\hat{y}_i = a + bx_i$ dengan menggunakan metode kuadrat terkecil. Fungsi tersebut membutuhkan 2 argumen yaitu vektor $\mathbf{x} = (x_1, x_2, \dots, x_n)$ yang merupakan nilai pengamatan peubah bebas dan vektor $\mathbf{y} = (y_1, y_2, \dots, y_n)$ sebagai nilai pengamatan peubah tak bebas. Rumus untuk koefisien regresi tersebut adalah:

$$b = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad a = \bar{y} - b\bar{x}$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i}{n} \quad \bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

7. Lanjutkan fungsi pada soal no. 6 untuk menghasilkan tabel analisis ragam sebagai berikut.

Sumber Keragaman	db	JK	KT	F-hitung	P-value
Regresi	1	<i>JKR</i>	<i>KTR</i>	<i>f</i>	<i>P</i>
Galat	$n - 2$	<i>JKG</i>	<i>KTG</i>		
Total	$n - 1$	<i>JKT</i>			

dengan

$$JKT = \sum_{i=1}^n (y_i - \bar{y})^2 \quad KTR = \frac{JKR}{1} = JKR \quad f = \frac{KTR}{KTG}$$

$$JKR = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 \quad KTG = JKG / (n - 2) \quad \hat{y}_i = a + bx_i$$

$$JKG = JKT - JKR \quad P = Prob(F > f) \text{ dimana } F \text{ terdistribusi menurut sebaran } F \text{ dengan derajat bebas } db_1 = 1 \text{ dan } db_2 = n - 1.$$

8. **Autokorelasi.** Diberikan suatu vektor (x_1, x_2, \dots, x_n) , autokorelasi sampel dengan lag k didefinisikan sebagai

$$r_k = \frac{\sum_{i=k+1}^n (x_i - \bar{x})(x_{i-k} - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Jadi

$$r_1 = \frac{\sum_{i=2}^n (x_i - \bar{x})(x_{i-1} - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{(x_2 - \bar{x})(x_1 - \bar{x}) + \dots + (x_n - \bar{x})(x_{n-1} - \bar{x})}{(x_1 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}$$

Buatlah fungsi `autocor(xVec)` yang mengambil argumen `xVec` yang berupa vektor dan menghasilkan sebuah *list* yang berisi dua nilai, yaitu: r_1 dan r_2 . Dapatkan nilai r_1 dan r_2 untuk vektor (2, 5, 8, ..., 53, 56).

9. **Simulasi Selang Kepercayaan.** Selang kepercayaan $(1-\alpha) \times 100\%$ bagi μ mempunyai tingkat kepercayaan $1-\alpha$. Artinya bila dilakukan pengambilan sampel berukuran n dari populasi normal dengan nilai tengah μ dan ragam σ^2 sebanyak 1000 kali dan dihitung selang kepercayaannya, maka hanya kira-kira $1000 \times (1-\alpha)$ kali dari selang kepercayaan-selang kepercayaan tersebut yang tidak memuat μ . Buatlah suatu fungsi untuk melakukan simulasi yang menunjukkan hal tersebut. Yang menjadi argumen fungsi tersebut adalah ukuran sampel n dan banyaknya pengulangan M . Ambil nilai $\alpha=0,05$. Sampel diambil dari populasi normal dengan nilai tengah 50 dan ragam 16. Selang kepercayaan $(1-\alpha) \times 100\%$ bagi μ adalah

$$\left(\bar{x} - t_{\alpha} \frac{s}{\sqrt{n}}, \bar{x} + t_{\alpha} \frac{s}{\sqrt{n}} \right)$$

dengan derajat bebas sebaran- t adalah $n-1$.

Tampilan dari output adalah seperti berikut:

Ukuran sampel: 10

Banyak sampel: 12

Rata2	Batas bawah	Batas atas	Memuat nilai tengah?
49.22511	46.33073	52.1195	TRUE
50.28744	46.06664	54.50825	TRUE
49.55385	44.93077	54.17692	TRUE
51.60982	48.13788	55.08176	TRUE
52.59998	50.55548	54.64449	FALSE
50.12085	47.32521	52.9165	TRUE
48.94468	46.88493	51.00443	TRUE
49.42196	46.61917	52.22475	TRUE
50.92691	48.23954	53.61428	TRUE
52.32674	50.55472	54.09876	FALSE
47.88183	45.4682	50.29546	TRUE
50.38743	47.43977	53.33508	TRUE

Persentase selang yang tidak memuat nilai tengah: 16.67%