

Pengantar Praktis Pemrograman R untuk Ilmu Komputer

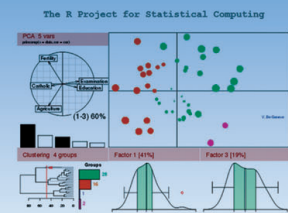
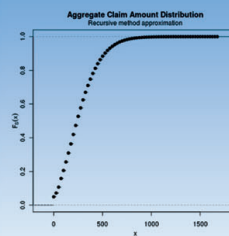
Banyak mahasiswa dan ilmuwan yang tidak memahami pentingnya penguasaan dan pengolahan data statistik terkomputerisasi untuk berbagai keperluan penelitian, dan solusi paling ampuh untuk mengatasi keterbatasan tersebut ialah dengan mempelajari bahasa R. R adalah bahasa pemrograman open source yang berhubungan dengan komputasi dan pengolahan data untuk statistika serta berhubungan dengan penampilan grafik menggunakan tools yang disediakan oleh paket-paketnya.

Buku ini sangat menarik dan menyenangkan untuk dipelajari karena menyajikan dasar pemrograman R yang terstruktur dengan memberikan contoh teori dan penerapannya yang real di dunia kerja.



Jl. Manggis IV, No. 2 RT/RW 07/04,
Tanjung Duren Selatan,
Grogol Petamburan, Jakarta Barat

Dr. Widodo Budiharto, S.Si., M.Kom. - Ro'fah Nur Rachmawati, S.Si., M.Si. Pengantar Praktis Pemrograman R untuk Ilmu Komputer



Pengantar Praktis Pemrograman R untuk Ilmu Komputer

Oleh:
Dr. Widodo Budiharto, S.Si., M.Kom
Ro'fah Nur Rachmawati, S.Si., M.Si



Pengantar Praktis Pemrograman R

untuk Ilmu Komputer

Oleh:

Dr. Widodo Budiharto, S.Si., M.Kom

Ro'fah Nur Rachmawati, S.Si., M.Si



Pengantar Praktis Pemrograman R *untuk Ilmu Komputer*

Oleh:

Dr. Widodo Budiharto, S.Si., M.Kom
Ro'fah Nur Rachmawati, S.Si., M.Si

Editor: Widodo B.
Desain: Catur S.

Cetakan 1, Februari 2013

ISBN:

Hak cipta dilindungi oleh undang-undang

Penerbit

Halaman Moeka Publishing

Jl. Manggis IV, No. 2 RT/RW 07/04
Tanjung Duren Selatan

Grogol Petamburan, Jakarta Barat

Blog: <http://halamanmoeka.blogspot.com>

Kata Pengantar

Dengan mengucapkan puji dan syukur ke hadirat Allah SWT, kami akhirnya dapat menghadirkan buku yang sangat dibutuhkan bagi mahasiswa / ilmuwan di bidang Ilmu Komputer, yaitu Pemrograman R. Banyak para mahasiswa dan ilmuwan yang tidak memahami secara dalam pentingnya penguasaan dan pengolahan data Statistik terkomputerisasi untuk berbagai keperluan penelitian, dan solusi paling ampuh untuk mengatasi keterbatasan tersebut ialah dengan mempelajari bahasa R. R adalah bahasa pemrograman open source yang berhubungan dengan komputasi dan pengolahan data untuk Statistika serta berhubungan dengan penampilan grafik menggunakan tools yang disediakan oleh paket-paketnya. Pengolahan data statistik dan komputasi data beserta simulasinya merupakan hal yang sangat penting dan harus dikuasai saat ini, mengingat kita dapat melihat trend dan mengambil keputusan dari data yang kita olah tersebut.

Buku ini sangat menarik dan menyenangkan untuk dipelajari karena menyajikan dasar pemrograman R yang terstruktur dengan memberikan contoh teori dan penerapannya yang real di dunia kerja. Buku ini juga dapat digunakan sebagai buku pegangan kuliah pada Statistika terapan dan Metode penelitian. Kritik dan saran yang membangun dari buku ini sangat penulis harapkan. Untuk Informasi kursus dan Seminar

yang berhubungan dengan R dan Teknoogi Informasi serta masukan mengenai buku ini dapat menghubungi:

Dr. Widodo Budiharto
e-Technology Center
Mutiara gading timur D3/2
Bekasi Timur
HP: 08569887384
Email: Widodo@widodo.com

Jakarta, 10 Februari 2013

Widodo Budiharto & Ro'fah

Daftar Isi

Kata Pengantar	iii
Daftar Isi	v
Bab 1 Pengantar Bahasa R.....	1
1.1 Pendahuluan	1
1.3 Memulai R	3
Bab 2 Entri Data pada R.....	8
2.1 Entri Data Menggunakan R Commander	8
2.2 Fitur Statistika R Commander	11
Bab 3 Pemrograman Dasar R.....	14
3.1 Berbagai Model Data R	14
3.1 Jenis-Jenis Data Obyek	18
Bab 4 Pemrograman Grafik R	21
4.1 Dasar Pemrograman Grafik R.....	21
4.1 Menggunakan Graph di R Commander.....	21
Bab 5 Analisa Statistik R	33
5.1 Dasar Analisa Variance pada R.....	33
Bab 6 Membuat Function	40
6.1 Loop dan Vektorisasi	40
6.2 Membuat Fungsi Sendiri	42
Bab 7 Paket Aktuar.....	46
7.1 Lose Distribution Modeling	46

7.2 Risk Theory	52
7.3 Simulasi Compound Hierarchical Model	54
7.4 Credibility Theory	55
Lampiran 1 Instalasi Program R dan Actuar	57
Daftar Pustaka	61
Tentang Penulis	62

Bab 1

Pengantar Bahasa R

Tujuan Instruksional Umum:

- Setelah membaca Bab ini, mahasiswa mampu menjelaskan peranan bahasa R dalam pengolahan data statistika.

Tujuan Instruksional Khusus:

- Mahasiswa mampu menyebutkan fungsi dan kegunaan bahasa R dalam kehidupan sehari-hari
- Mahasiswa dapat menjalankan program dasar R dari konsol dan menginstal RCommander.

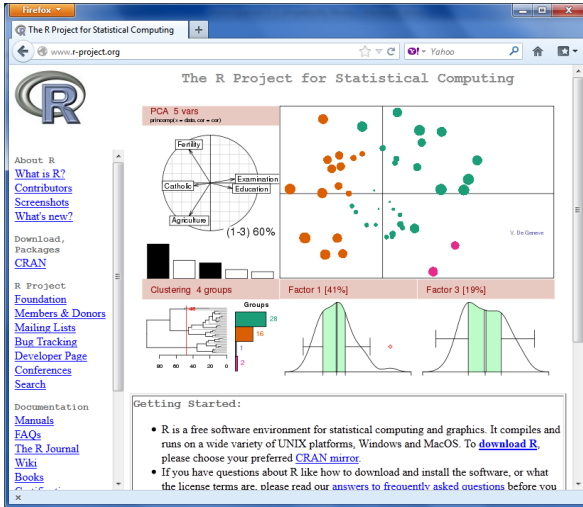
1.1 Pendahuluan

Perkembangan Teknologi Informasi berdampak positif pada berkembangnya berbagai peranti lunak pengolah data. Secara umum, ada beberapa kategori software analisis data, yang bersifat komersil seperti SPSS dan SAS, serta bersifat open source seperti R. R adalah bahasa pemrograman open source yang berhubungan dengan komputasi dan pengolahan data untuk Statistika dan yang berhubungan dengan penampilan grafik menggunakan tools yang disediakan oleh paket-paketnya yang sangat berguna di dalam penelitian dan industri. Versi awal dari R dibuat pada tahun 1992 di Universitas Auckland, New Zealand oleh Ross Ihaka dan Robert Gentleman.

Pada saat ini, source code kernel R dikembangkan terutama oleh R Core Team yang beranggotakan 17 orang

statistisi dari berbagai penjuru dunia. Selain itu, para ahli lain pengguna R di seluruh dunia juga memberikan kontribusi berupa pengembangan kode, melaporkan bug, dan membuat dokumentasi untuk R. Paket statistik R bersifat multiplatforms, dengan file instalasi binary/file tar tersedia untuk sistem operasi Windows, Mac OS, Mac OS X, Linux, Free BSD, NetBSD, iSolaris, AIX, dan HPUX. Secara umum, sintaks dari bahasa R adalah ekuivalen dengan paket statistik Splus, sehingga sebagian besar keperluan analisis statistika, dan pemrograman dengan R adalah hampir identik dengan perintah yang dikenal di Splus. R menyediakan berbagai aplikasi statistika (linear dan nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering) dan teknik penampilan simulasi grafik. Kelebihan R lainnya adalah kemudahan yang dirancang dengan baik untuk plot grafik berkualitas, termasuk simbol matematika dan rumus jika diperlukan, termasuk berbagai perbaikan di sisi tampilan aplikasi.

Pada umumnya mahasiswa membutuhkan program R berjalan terlebih dahulu, dibandingkan penguasaan ilmu Statistiknya, oleh karna itu penulis contohkan terlebih dahulu cara mengunduh dan menjalankan program R dari konsol. Gambar berikut menampilkan situs resmi R untuk tutorial dan pengunduhan di www.r-project.org dengan versi R 2.15. Paket ini diluncurkan pada tahun 2005 serta tersedia pada fasilitas network komprehensif R sejak 2006 (CRAN, <http://CRAN.R-project.org/>).



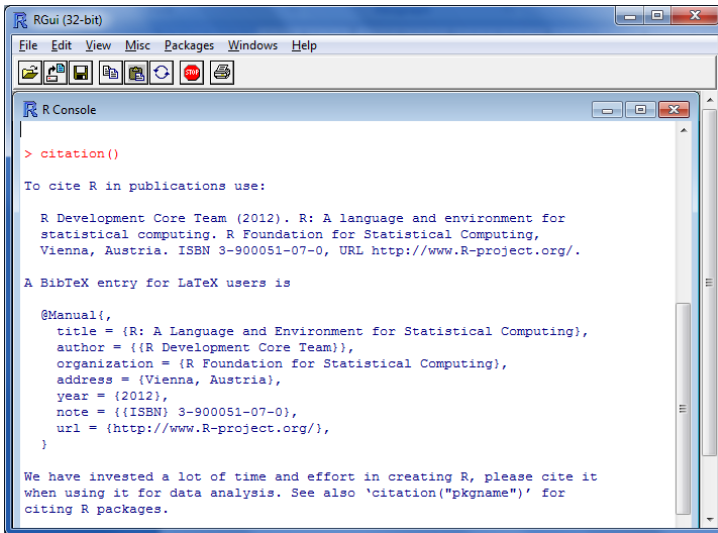
Gambar 1.1 Situs resmi dan pengunduhan R

1.2 Memulai R

Setelah proses instalasi berjalan sukses, langkah berikutnya memulai R ialah mengetikkan beberapa fungsi dasar pada konsol, berikut contoh untuk mengetahui informasi cara citasi menggunakan fungsi `citation()` yang harus diketikkan sebagai berikut :

> citation()

Hasilnya sebagai berikut:



Gambar 1.2. Menuliskan fungsi `citation()` dan hasil yang diperoleh

Contoh begitu handalnya R ialah, jika kita ingin mengetahui jumlah dari $3+300$, cukup menulis sebagai berikut:

```
> 3+300
```

```
[1] 303
```

Jika kita ingin mengetahui nilai dari $\log(2)$, cukup tulis:

```
> log(2)
```

```
[1] 0.6931472
```

Untuk memelajari variabel dasar, Anda dapat membayangkan sebuah variabel x , dengan diberi nilai 1, serta menggunakan symbol $\#$ sebagai komentar, berikut beberapa fungsi dasar yang penting:

```
> x<-1          # x sekarang bernilai 1
>y<- 1:10       # y merupakan vektor
> z<-c(1,3,5)   # z vector [ 1 3 5]

> a <- c(1,2,3,4,5)  #operasi pada sebuah vektor
> a
[1] 1 2 3 4 5

> a+1

[1] 2 3 4 5 6

> mean(a)          # mencari mean

[1] 3

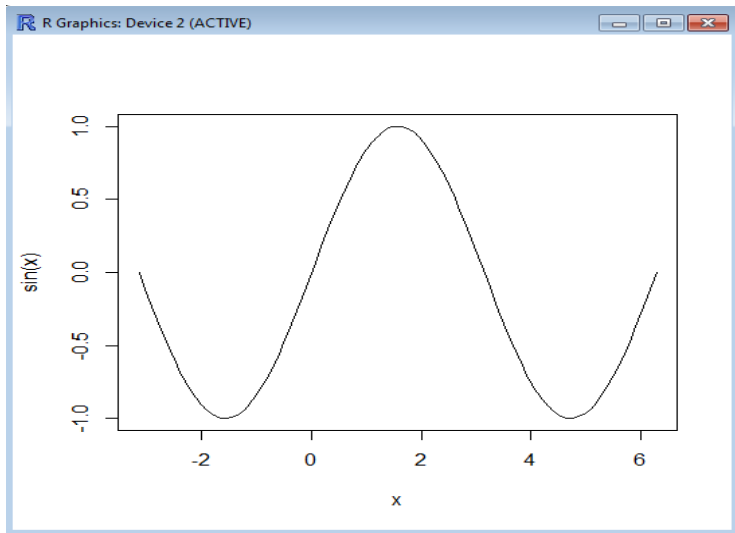
> var(a)

[1] 2.5           # mencari variance

> 4>4             # operasi logis
[1]FALSE
```

Untuk penampilan grafik, dapat dicoba kode yang simpel dan menarik menggunakan fungsi plot berikut :

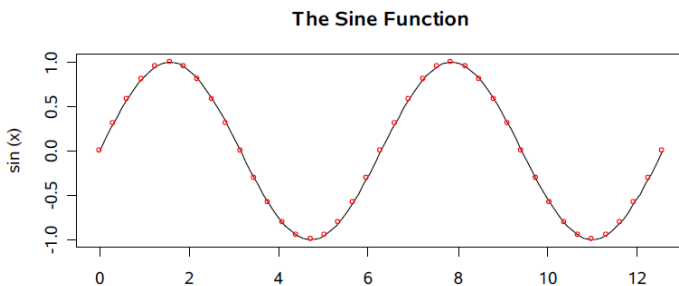
```
> plot(sin, -pi, 2*pi)
```



Gambar 1.3. Hasil penggambaran fungsi sinus

Kita juga dapat memplot sebuah grafik dengan title yang sesuai dan kustomisasi penampilan dengan points sebagai berikut:

```
plot(sin, xlim=c(0,4*pi), main="The Sine Function")  
points(x, sin(x), col="red")
```

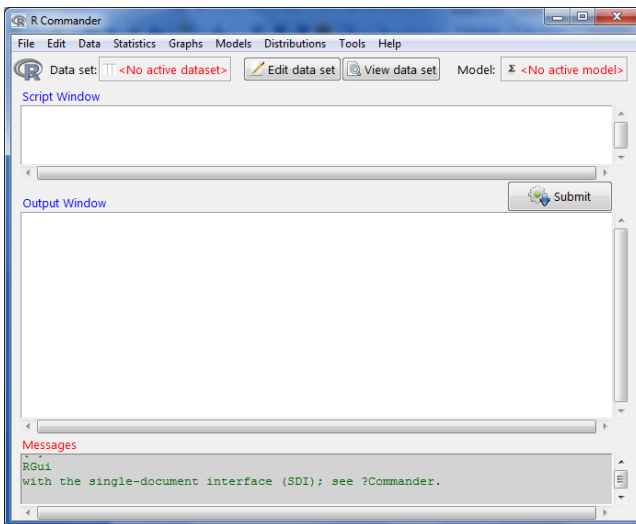


Gambar 1.4. Plot fungsi sinus dan kustomisasi penampilan

Jika sudah selesai bekerja dengan R, dapat mengetikkan q() untuk keluar dari aplikasi.

Latihan:

1. Sebutkan kemampuan metode-metode Statistika apa yang didukung oleh R
2. Buatlah program untuk menyimpan 3 buah data pada 3 buah variabel x,y,z.
3. Buatlah program untuk menampilkan grafik dari $2\cos x$
4. Jalankan program GUI sederhana bernama R Commander yang terdiri dari Library R commander. Untuk menginstal R commander, setelah mengunduh Rcmdr, maka pilih menu Packages, pilih Install package(s).



Gambar 1.5. Tampilan R Commander

Bab 2

Entri Data pada R

Tujuan Instruksional Umum

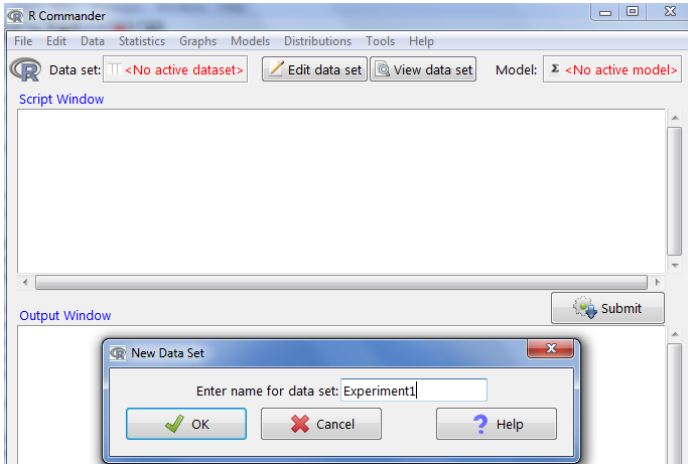
- Setelah mempelajari Bab ini, mahasiswa mampu melakukan entri data dan menggunakan fitur statistika dasar pada R.

Tujuan Instruksional Khusus

- Mahasiswa mampu mengentri data dan mengolah data menggunakan fungsi-fungsi statistik pada R menggunakan R Commander.
- Mahasiswa mampu menyebutkan dan menggunakan beberapa fungsi statistik pada R Commander

2.1 Entri Data Menggunakan R Commander

Untuk menjalankan R Commander , ketikkan perintah **library(Rcmdr)** pada jendela konsol. Jika proses berjalan sukses maka akan nampak jendela R-Commander. Pengisian data secara langsung dengan R dengan menggunakan R-commander dapat dilakukan melalui menu Data, dan pilih New dataset Setelah itu berna nama Experiment1 seperti gambar 2.1



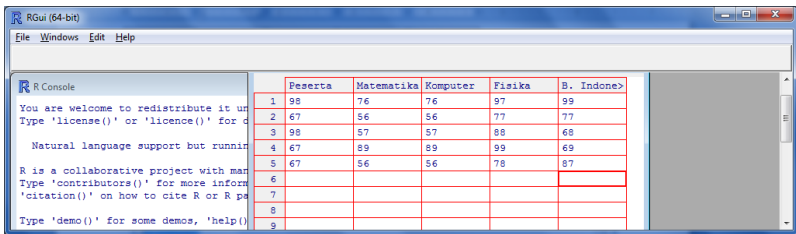
Gambar 2.1 Membuat data set baru

Kemudian klik OK, maka akan terbuka jendela Data Editor. Pengisian nama variabel dilakukan dengan cara klik pada kolom paling atas dari data editor. Sebagai contoh, masukkan data percobaan sebagai berikut:

Tabel 2.1 data percobaan

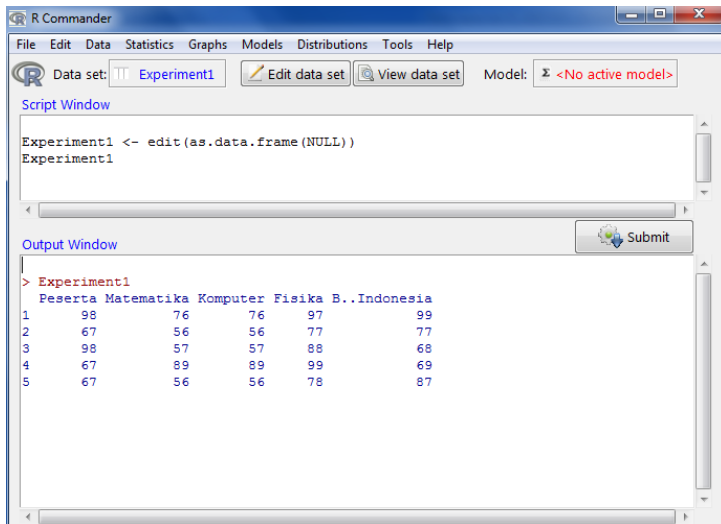
Peserta	Matematika	Komputer	Fisika	B. Indonesia
Andi	98	76	97	99
Tina	67	56	77	77
Chytia	98	57	88	68
ike	67	89	99	69
Shodiq	67	56	78	87

Berikut ini hasil dari entri data dari table di atas



Gambar 2.2 Hasil entri data

Untuk melihat hasil entri data pada output window, dapat diketikkan Experiment1, lalu diblok perintah tersebut dan klik button Submit sebagai berikut:

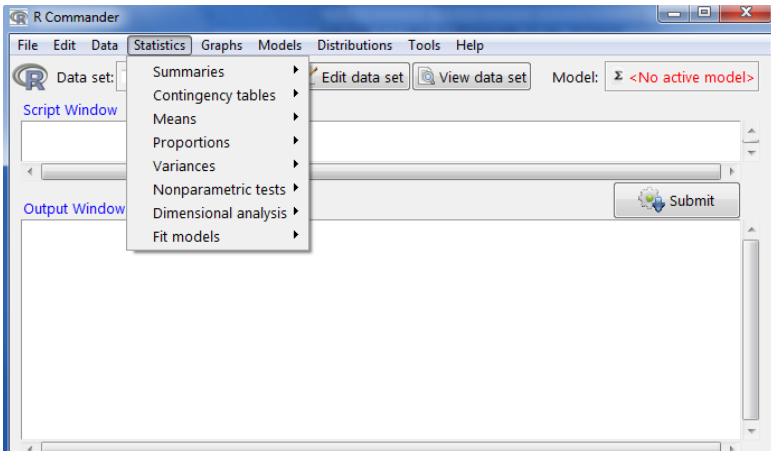


Gambar 2.3 Hasil eksekusi dari script Window

Untuk melakukan editing terhadap data Experiment1, dilakukan dengan mengklik tombol Edit data set. Setelah itu jendela Data Editor akan dibuka kembali, proses editing data dapat langsung dilakukan pada data yang ingin dirubah.

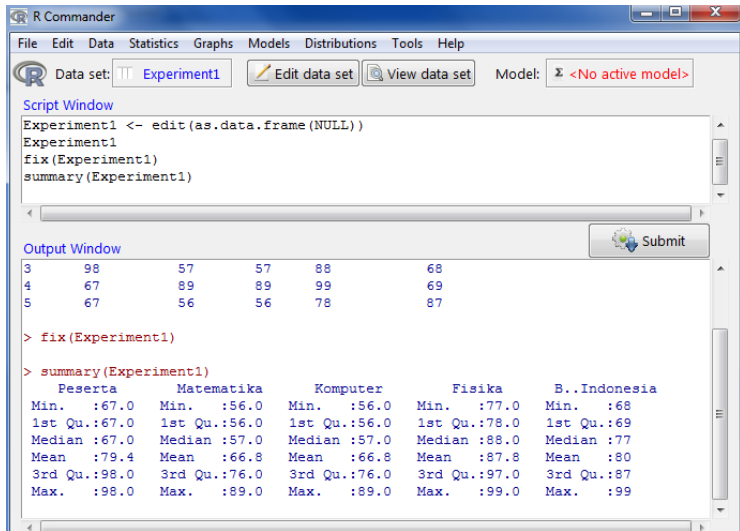
2.2 Fitur Statistika pada R Commander

Ada beberapa fitur Statistika yang memudahkan kita menganalisa data seperti Summary, means, variances dan fit models seperti gambar berikut :



Gambar 2.4 Beberapa fitur Statistika pada R Commander

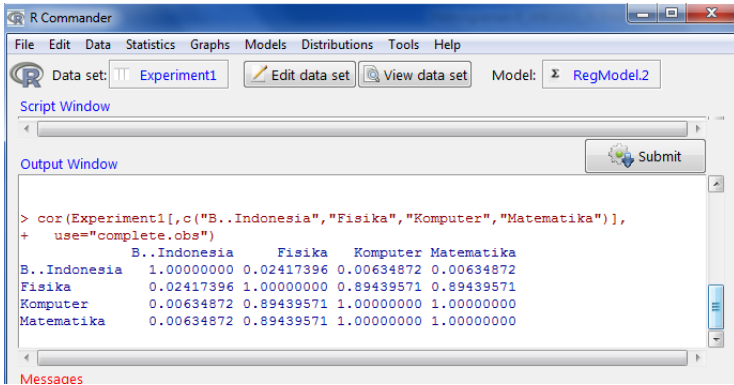
Misalnya informasi summary dari data yang kita entri menggunakan fungsi summary seperti gambar di bawah ini:



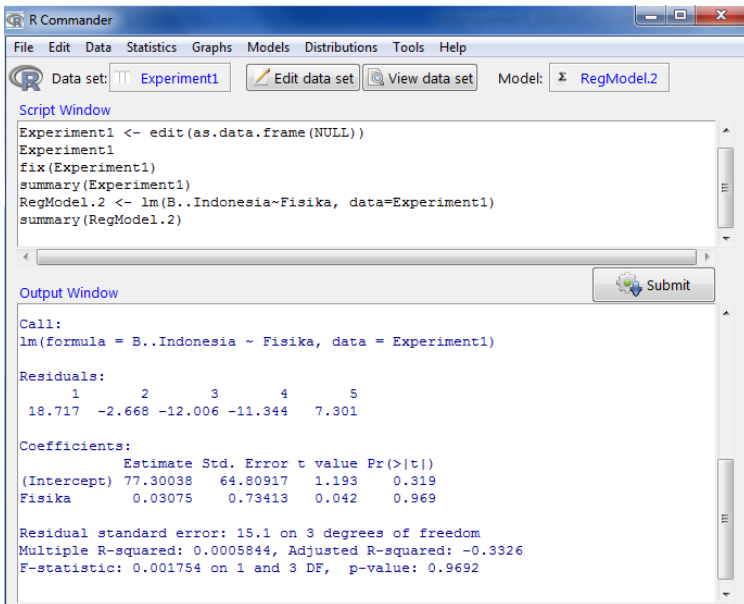
Gambar 2.4 Fungsi summary yang diterapkan pada data set

Latihan

1. Jelaskan mengenai correlation matrix dan linear regression.
2. Analisa data set percobaan anda menggunakan correlation matrix dan linear regression menggunakan R Commander seperti gambar berikut:



Gambar 2.5 Menggunakan fungsi cor untuk correlation matrix



Gambar 2.6 Analisa linear regression menggunakan R Commander

Bab 3

Pemrograman Dasar R

Tujuan Instruksional Umum

- Setelah mempelajari Bab ini, mahasiswa mampu melakukan pemrograman dasar dan penerapan fungsi statistika dasar pada R.

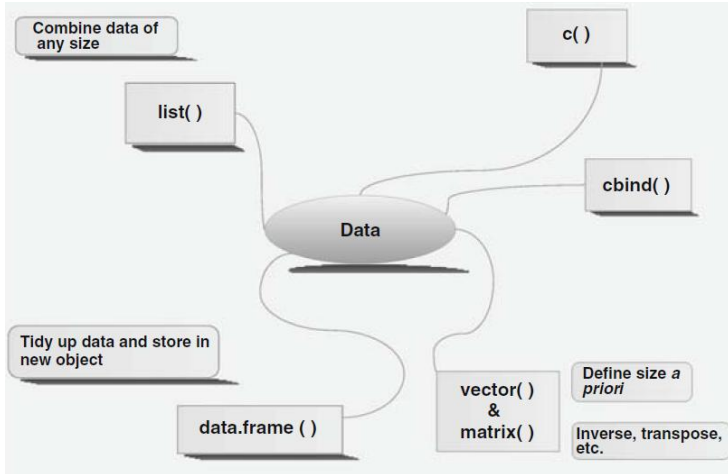
Tujuan Instruksional Khusus

- Mahasiswa mampu memrogram beberapa fungsi statistik pada R pada konsol.
- Mahasiswa mampu membuat variabel dan matriks dan menerapkannya
- Mahasiswa mampu menyebutkan dan menggunakan beberapa fungsi statistik pada R

3.1 Berbagai Model Data di R

Pada bahasa R, data dipandang sebagai suatu objek yang memiliki suatu atribut dan berbagai fungsionalitas. Sifat data ditentukan oleh type data dan mode data. Ada berbagai type data yang dikenal oleh R, antara lain vektor, matriks, list, data frame, array, factor dan fungsi built in. Berikut ini beberapa model data yang umum digunakan serta contoh penerapan fungsi built in. Untuk menyimpan

data di R ada berbagai metode seperti menggunakan fungsi `c()`, `list()`, `cbind()` dan `data.frame()` seperti gambar 3.1



Gambar 3.1 Berbagai metode menyimpan data di R

Menggabungkan Data

Untuk memasukkan beberapa data numeric ke suatu variabel bernama **Nilai**, dapat digunakan fungsi `c` (concatenate) sebagai berikut:

```
>Nilai <-c(88,66,99,56,78)
```

```
> Nilai
```

```
[1] 88 66 99 56 78
```

Untuk melihat 3 data pertama dari variabel Nilai :

```
> Nilai [1:3]
```

```
[1] 88 66 99
```

Untuk menjumlahkan nilai dari variabel Nilai,

```
> sum(Nilai)
```

```
[1] 387
```

Sedangkan untuk memasukkann data karakter, gunakan contoh berikut :

```
>VarNames <- c("Wingcrd", "Tarsus", "Head", "Wt")
```

```
> VarNames
```

```
[1] "Wingcrd" "Tarsus" "Head" "Wt"
```

Membuat Matriks

Matriks adalah data dua dimensi dimana sebagian besar fungsi-fungsi statistik dalam R dapat dianalisis dengan menggunakan bentuk matriks. Bentuk matriks ini juga banyak digunakan pada operasi fungsi-fungsi built-in untuk aljabar linear dalam R, seperti untuk penyelesaian suatu persamaan linear. Argumen yang diperlukan adalah elemen-elemen dari matriks, dan argumen optional yaitu banyaknya baris dan banyaknya kolom. Berikut contohnya:

```
>matriks.1 = matrix(c(1,2,3,4,5,6),nrow=2,ncol=3)
```

```
> matriks.1
```

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

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

Pengisian matriks secara baris perbaris dapat dilakukan dengan menggunakan argumen optional `byrow=T` . Berikut ini contoh tentang penggunaan argumen tersebut.

```
> data=c(6,2.8,7.1,5.3,7.6,9)  
> matriks.z=matrix(data,nrow=3,ncol=2,byrow=T)  
> matriks.z  
      [,1] [,2]  
[1,] 6.0 2.8  
[2,] 7.1 5.3  
[3,] 7.6 9.0
```

Data dalam bentuk matriks merupakan format yang umum di dalam statistika, pada R dapat menggunakan fungsi `matrix` dengan informasi banyaknya baris dan kolom dengan contoh sebagai berikut:

```
> DataMat<-matrix(nrow=8,ncol=4)  
> DataMat  
      [,1] [,2] [,3] [,4]  
[1,] NA NA NA NA  
[2,] NA NA NA NA  
[3,] NA NA NA NA  
[4,] NA NA NA NA  
[5,] NA NA NA NA  
[6,] NA NA NA NA  
[7,] NA NA NA NA  
[8,] NA NA NA NA
```


Pada R, dapat pula dilakukan penggabungan satu kolom atau satu baris baru kedalam matriks lain. Hal ini dapat dilakukan dengan menggunakan perintah `rbind` (untuk menambahkan ke baris) dan `cbind` (untuk menambahkan ke kolom). Perhatikan contoh-contoh berikut ini.

```
> a=matrix(c(7,4,5,8,7,8),2,3)
```

```
> a
```

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

```
> anew=cbind(a,c(1,2)) # menambahkan ke kolom ke 4 dari a
```

```
> anew
```

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

```
>anew=rbind(a,c(1,2,3)) # menambahkan ke baris ke 3 dari a
```

```
> anew
```

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

3.2 Jenis –Jenis Data Obyek

Ada berbagai jenis data obyek, diantaranya adalah data vektor yang berisi kumpulan/array bilangan dalam karakter, string atau nilai logik serta data matriks. Pada data vektor harus digunakan mode tunggal pada data, sehingga gabungan dua data atau lebih yang berbeda mode tidak dapat dilakukan kedalam satu objek vektor.

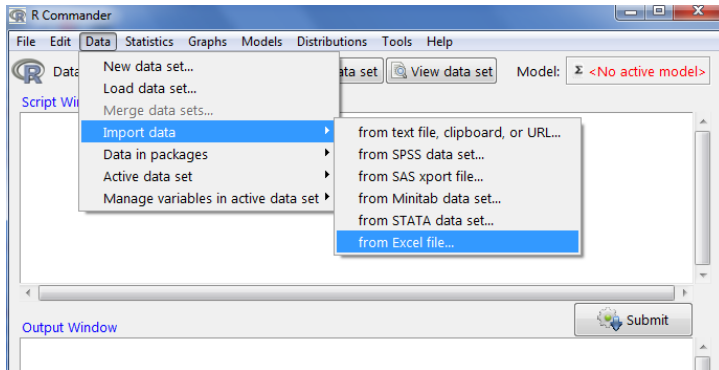
```
> A=c(1:15)
> A
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
> length(A)
[1] 15
> A[c(3,7)]      # menampilkan elemen ke 3 dan 7
[1] 3 7
> mode(A)
[1] "numeric"   # mode data numeric
```

Jenis data obyek lainnya ialah Data frame, merupakan objek yang mempunyai bentuk sama dengan matriks, terdiri atas baris dan kolom. Perbedaannya adalah data frame dapat terdiri atas mode data yang berbeda-beda untuk setiap kolomnya. Berikut contohnya:

```
matriks.1=matrix(1:9,3)
>
dataframe.1=data.frame(nomer=1:4,nama=c("Widodo","Budi","Tina","Dony"), nilai=7:10)
> matriks.1
  [,1] [,2] [,3]
[1,]  1  4  7
[2,]  2  5  8
[3,]  3  6  9
```

Latihan :

1. Buatlah program R ang menerapkan list() dan cbind()
2. Buatlah contoh data 5x5 menggunakan matriks.
3. Lakukan import data Ms. Excel menggunakan tools R Commander seperti gambar berikut:



Gambar 3.2 Import data dari Ms. Excel atau sumber database lainnya.

4. Jalankan program baca data menggunakan fungsi scan berikut:
scan("data1.txt")
Read 8 items
[1] 50 28 75 35 49 64 88 94

Bab 4

Pemrograman Grafik R

Tujuan Instruksional Umum

- Setelah mempelajari Bab ini, mahasiswa mampu melakukan pemrograman grafik dan penerapan fungsi grafik dasar pada R.

Tujuan Instruksional Khusus

- Mahasiswa mampu memrogram beberapa fungsi grafik pada R melalui konsol dan R Commander
- Mahasiswa mampu membuat visualisasi data menggunakan grafik
- Mahasiswa mampu menyebutkan dan menggunakan beberapa fungsi plot, pie dan scatterplot pada R

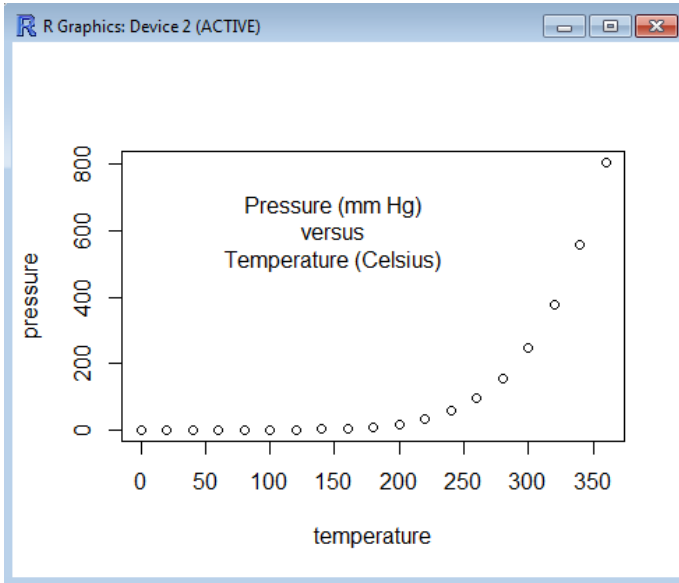
4.1 Dasar Pemrograman Grafik R

Data yang berhasil dientri atau diimport dari aplikasi lain selayaknya divisualisasikan pada grafik untuk analisa. Sebagai contoh, kita dapat menggunakan data dari R yaitu variabel `pressure`, jika kita ingin menampilkan plot dengan teks, maka gunakan fungsi `plot()` dan fungsi `text()` pada posisi piksel 150, 600 sebagai berikut:

```
>plot (pressure)

text (150,          600,          "Pressure          (mm
Hg)\nversus\nTemperature (Celsius)")
```

Hasilnya ditunjukkan pada gambar 4.1



Gambar 4.1 hasil penampilan plot *temperature*

Beberapa model grafik menggunakan `type` yang dapat digunakan pada fungsi `plot` diantaranya sebagai berikut:

```
> y <- rnorm(20) #membangkitkan nilai random
variabel
> plot(y, type="p")
> plot(y, type="l")
> plot(y, type="b")
> plot(y, type="h")
```

Untuk konfigurasi ketebalan garis dan warna dan arsiran, dapat digunakan properti seperti `lwd` , `color` dan `dashed` sebagai berikut:

```
> plot(y, type="l", lwd=3)
> plot(y, type="l", col="grey")
> plot(y, type="l", lty="dashed")
> plot(y, type="l", ylim=c(-4, 4))
```

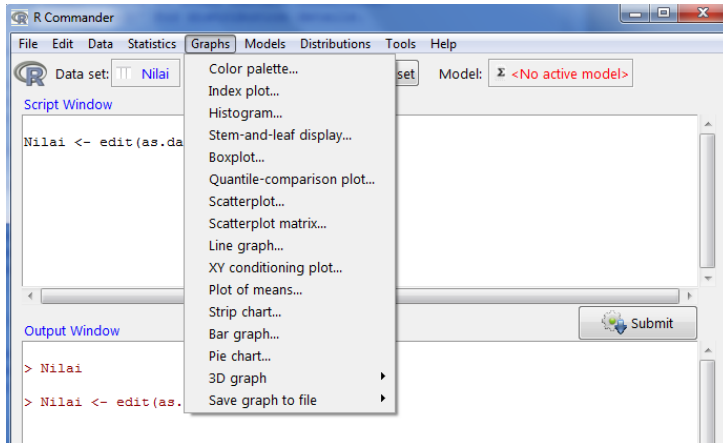
Beberapa fungsi plot lain yang sering digunakan ditunjukkan pada tabel 4.1

Tabel 4.1 beberapa fungsi dasar plot grafik

<code>pie(x)</code>	circular pie-chart
<code>boxplot(x)</code>	"box-and-whiskers" plot
<code>stripchart(x)</code>	plot of the values of <code>x</code> on a line (an alternative to <code>boxplot()</code> for small sample sizes)
<code>coplot(x~y z)</code>	bivariate plot of <code>x</code> and <code>y</code> for each value (or interval of values) of <code>z</code>
<code>interaction.plot(f1, f2, y)</code>	if <code>f1</code> and <code>f2</code> are factors, plots the means of <code>y</code> (on the <code>y</code> -axis) with respect to the values of <code>f1</code> (on the <code>x</code> -axis) and of <code>f2</code> (different curves); the option <code>fun</code> allows to choose the summary statistic of <code>y</code> (by default <code>fun=mean</code>)
<code>matplot(x,y)</code>	bivariate plot of the first column of <code>x</code> vs. the first one of <code>y</code> , the second one of <code>x</code> vs. the second one of <code>y</code> , etc.
<code>dotchart(x)</code>	if <code>x</code> is a data frame, plots a Cleveland dot plot (stacked plots line-by-line and column-by-column)
<code>fourfoldplot(x)</code>	visualizes, with quarters of circles, the association between two dichotomous variables for different populations (<code>x</code> must be an array with <code>dim=c(2, 2, k)</code> , or a matrix with <code>dim=c(2, 2)</code> if <code>k = 1</code>)
<code>assocplot(x)</code>	Cohen-Friendly graph showing the deviations from independence of rows and columns in a two dimensional contingency table
<code>mosaicplot(x)</code>	'mosaic' graph of the residuals from a log-linear regression of a contingency table
<code>pairs(x)</code>	if <code>x</code> is a matrix or a data frame, draws all possible bivariate plots between the columns of <code>x</code>
<code>plot.ts(x)</code>	if <code>x</code> is an object of class <code>"ts"</code> , plot of <code>x</code> with respect to time, <code>x</code> may be multivariate but the series must have the same frequency and dates

4.2 Menggunakan Graph di R Commander

Berbagai model plot dapat dihasilkan R, seperti Histogram, Boxplot, Scatterplot dan Pie chart sebagai contoh pada gambar 4.2 di R Commander:



Gambar 4.2 Berbagai pilihan grafik yang disediakan oleh R Commander

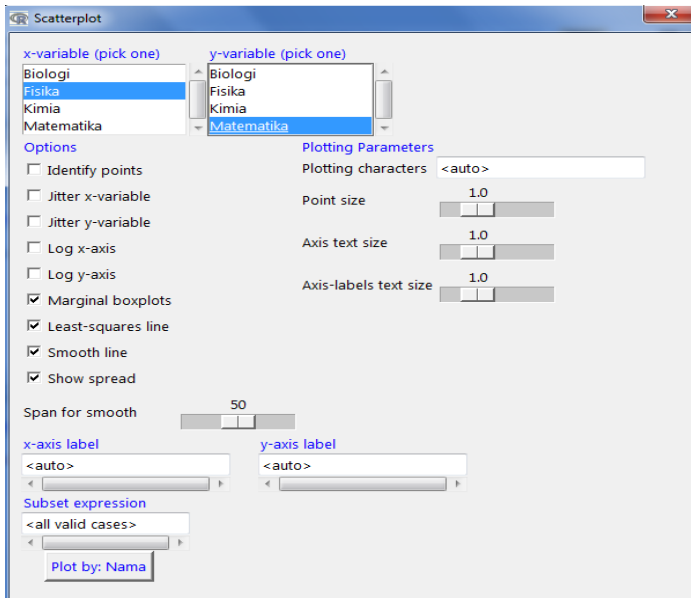
Untuk percobaan membuat grafik Scatterplot menggunakan fitur **Graphs** di R Commander, buatlah data set bernama Nilai sebagai berikut:

The image shows a screenshot of the R Data Editor window. It displays a table with 5 rows and 6 columns. The columns are labeled 'Nama', 'Matematik', 'Fisika', 'Kimia', and 'Biologi'. The data is as follows:

	Nama	Matematik	Fisika	Kimia	Biologi
1	Anton	76	87	56	87
2	Ridwan	76	67	67	66
3	Yusuf	87	65	54	89
4	Aziz	67	77	78	76
5	Santi	66	88	98	

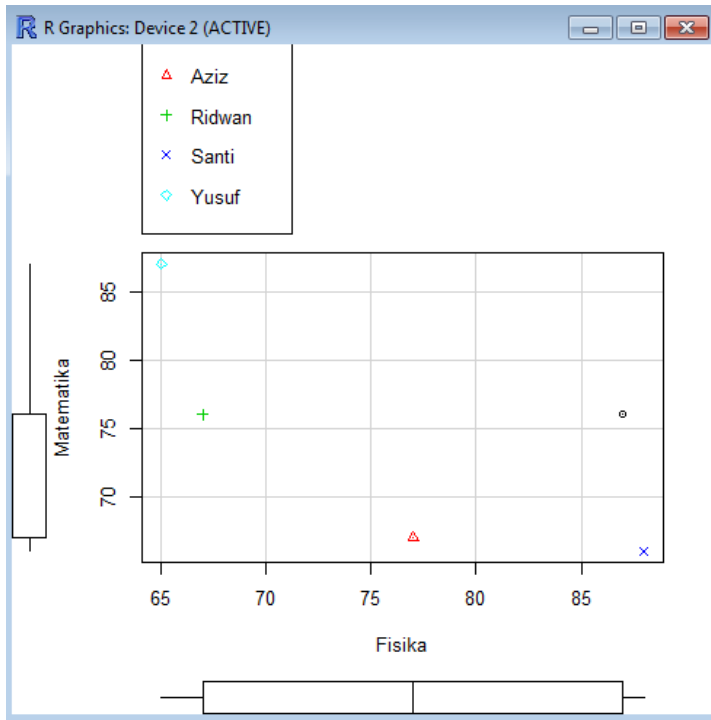
Gambar 4.3 Contoh data Nilai yang akan digunakan untuk Scaterplot().

Setelah data dientri, pilih menu Scaterplot dan atur property berikut ini, dimana kita harus menset Plot by : **Nama**.



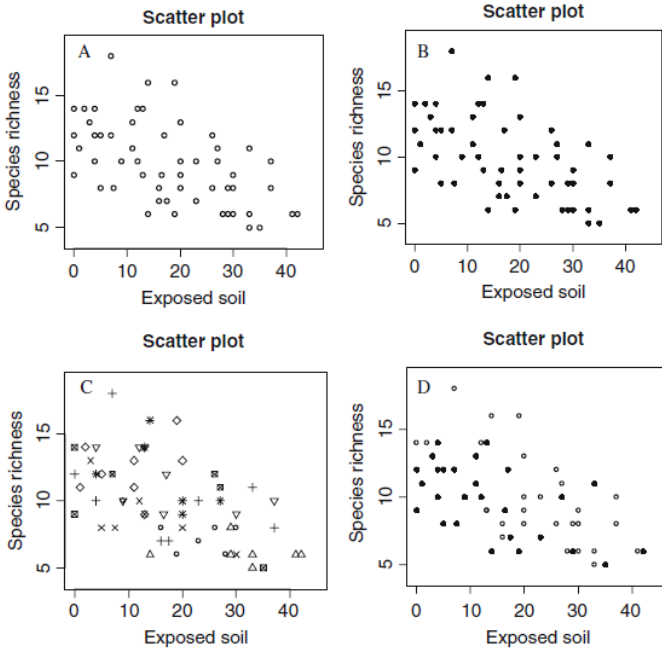
Gambar 4.4 Memilih menu Scaterplot dan mengatur properti penampilan data

Setelah klik OK, maka akan tampil grafik yang memudahkan kita menganalisa nilai siswa sebagai berikut:



Gambar 4.5 Hasil Scaterplot untuk analisa nilai siswa

Beberapa fungsi lain dari kemampuan Scaterplot ditunjukkan pada gambar 4.6 di bawah ini:



Gambar 4.6 Berbagai model Scaterplot lainnya. Untuk merubah penampilan grafik dapat dicoba fungsi lanjutan berikut misal jika ingin membangkitkan data acak dari variabel random

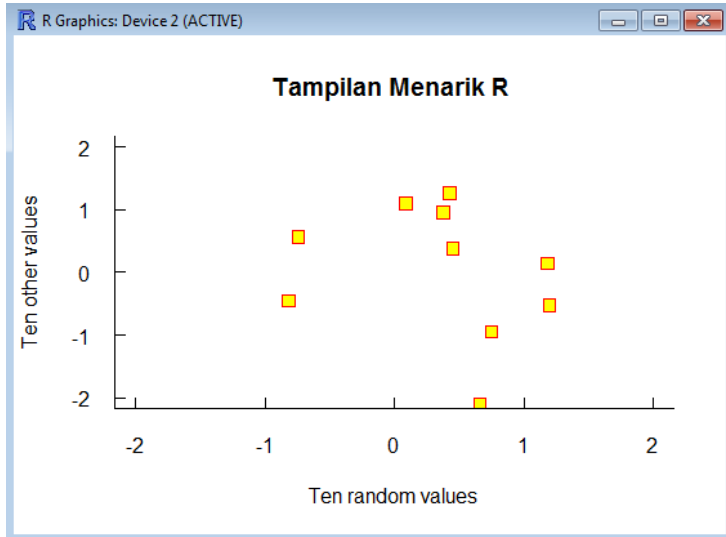
```
> x <- rnorm(10)
> y <- rnorm(10)
```

Untuk plot grafik tersebut dengan beberapa parameter seperti warna, maka ddapat digunakan perintah berikut :

```
plot(x, y, xlab="Ten random values",
     ylab="Ten other values",
```

```
xlim=c(-2, 2), ylim=c(-2, 2), pch=22,  
col="red",  
bg="yellow", bty="l", tcl=0.4,  
main="Tampilan menarik R", las=1, cex=1.5)
```

Hasilnya ditampilkan pada gambar di bawah ini:



Gambar 4.7 Hasil kustomisasi penampilan grafik

Demikianlah contoh dasar membuat grafik yang sangat mudah dan lengkap.

Latihan:

1. Berikan contoh penerapan fungsi `lines()` untuk membuat garis, `boxplot()` dan `pie()`.
2. Buatlah grafik yang menampilkan berbagai bentuk sebagai berikut:

```
par(mfrow=c(3, 2))

# Scatterplot
x <- c(0.5, 2, 4, 8, 12, 16)
y1 <- c(1, 1.3, 1.9, 3.4, 3.9, 4.8)
y2 <- c(4, .8, .5, .45, .4, .3)
par(las=1, mar=c(4, 4, 2, 4))
plot.new()
plot.window(range(x), c(0, 6))
lines(x, y1)
lines(x, y2)
points(x, y1, pch=16, cex=2)
points(x, y2, pch=21, bg="white", cex=2)
par(col="grey50", fg="grey50",
col.axis="grey50")
axis(1, at=seq(0, 16, 4))
axis(2, at=seq(0, 6, 2))
axis(4, at=seq(0, 6, 2))
box(bty="u")
mtext("Travel Time (s)", side=1, line=2,
cex=0.8)
mtext("Responses per Travel", side=2, line=2,
las=0, cex=0.8)
mtext("Responses per Second", side=4, line=2,
las=0, cex=0.8)
text(4, 5, "Bird 131")
par(mar=c(5.1, 4.1, 4.1, 2.1), col="black",
fg="black", col.axis="black")

# Histogram
# Random data
Y <- rnorm(50)
```

```
# Make sure no Y exceed [-3.5, 3.5]
Y[Y < -3.5 | Y > 3.5] <- NA
x <- seq(-3.5, 3.5, .1)
dn <- dnorm(x)
par(mar=c(4.5, 4.1, 3.1, 0))
hist(Y, breaks=seq(-3.5, 3.5), ylim=c(0,
0.5),
     col="grey80", freq=FALSE)
lines(x, dnorm(x), lwd=2)
par(mar=c(5.1, 4.1, 4.1, 2.1))

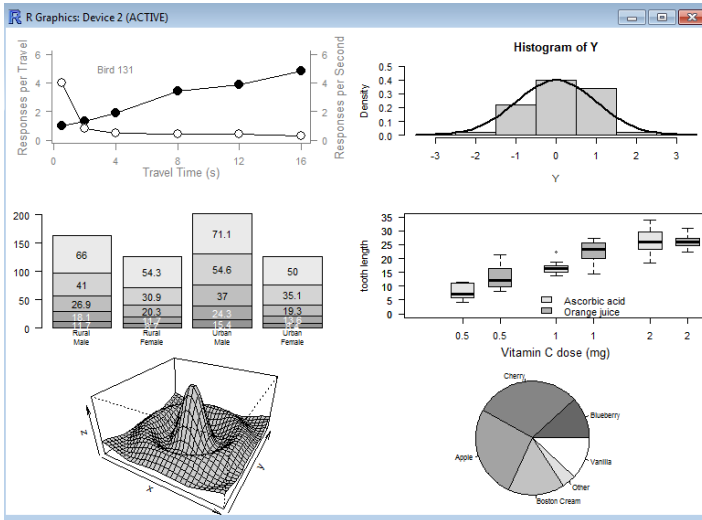
# Barplot
# Modified from example(barplot)
par(mar=c(2, 3.1, 2, 2.1))
midpts <- barplot(VADeaths, col=grey(0.5 +
1:5/12),
                  names=rep("", 4))
mtext(sub(" ", "\n", colnames(VADeaths)),
      at=midpts, side=1, line=0.5, cex=0.5)
text(rep(midpts, each=5), apply(VADeaths, 2,
cumsum) - VADeaths/2,
     VADeaths, col=rep(c("white", "black"),
times=2:3, cex=0.8))
par(mar=c(5.1, 4.1, 4.1, 2.1))

# Boxplot
# Modified example(boxplot) - itself from
suggestion by Roger Bivand
par(mar=c(3, 4.1, 2, 0))
boxplot(len ~ dose, data = ToothGrowth,
        boxwex = 0.25, at = 1:3 - 0.2,
        subset= supp == "VC",
col="grey90",
        xlab="",
        ylab="tooth length",
ylim=c(0,35))
mtext("Vitamin C dose (mg)", side=1,
line=2.5, cex=0.8)
boxplot(len ~ dose, data = ToothGrowth,
add = TRUE,
        boxwex = 0.25, at = 1:3 + 0.2,
```

```
subset= supp == "OJ",
col="grey70")
  legend(1.5, 9, c("Ascorbic acid",
"Orange juice"), bty="n",
        fill = c("grey90", "grey70"))
par(mar=c(5.1, 4.1, 4.1, 2.1))

# Persp
# Almost exactly example(persp)
  x <- seq(-10, 10, length= 30)
  y <- x
  f <- function(x,y) { r <- sqrt(x^2+y^2);
10 * sin(r)/r }
  z <- outer(x, y, f)
  z[is.na(z)] <- 1
# 0.5 to include z axis label
par(mar=c(0, 0.5, 0, 0), lwd=0.1)
  persp(x, y, z, theta = 30, phi = 30,
expand = 0.5, col = "grey80")
par(mar=c(5.1, 4.1, 4.1, 2.1), lwd=1)

# Piechart
# Example 4 from help(pie)
par(mar=c(0, 2, 1, 2), xpd=FALSE, cex=0.5)
  pie.sales <- c(0.12, 0.3, 0.26, 0.16,
0.04, 0.12)
  names(pie.sales) <- c("Blueberry",
"Cherry",
  "Apple", "Boston Cream", "Other",
"Vanilla")
  pie(pie.sales, col =
gray(seq(0.4,1.0,length=6)))
```



Gambar 4.8 Berbagai model grafik pada R

Bab 5

Analisa Statistik R

Tujuan Instruksional Umum

- Setelah memelajari Bab ini, mahasiswa mampu melakukan analisa statistik pada R.

Tujuan Instruksional Khusus

- Mahasiswa mampu memrogram beberapa fungsi statistik pada R melalui konsol
- Mahasiswa mampu menyebutkan dan menggunakan beberapa fungsi analisa statistik pada R

5.1 Dasar Analisa Variance pada R

Paket stats pada R berisi berbagai fungsi dasar analisa statistik seperti tes klasik, modle linear, distribusi, summary statistik, analisa time series dan analisa multivariate. Sebagai contoh, kita gunakan data set dari R bernama **InsectSprays** serta 6 insektisida yang ada pada R, dan digunakan untuk melihat response insects dan digunakan untuk analisa variance menggunakan fungsi aov.

```
> data(InsectSprays)
> aov.spray <- aov(sqrt(count) ~ spray, data =
InsectSprays)
> aov.spray
```


Call:

```
aov(formula = sqrt(count) ~ spray, data =  
InsectSprays)
```

Terms:

```
          spray Residuals  
Sum of Squares 88.43787 26.05798  
Deg. of Freedom    5    66
```

Residual standard error: 0.6283453
Estimated effects may be unbalanced

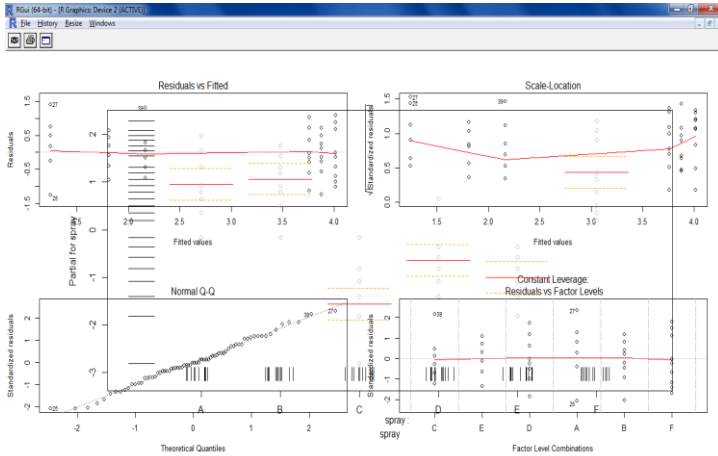
```
> summary(aov.spray)
```

```
      Df Sum Sq Mean Sq F value Pr(>F)  
spray    5  88.44  17.688   44.8 <2e-16 ***  
Residuals 66  26.06   0.395  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Lalu kita dapat membuat grafik dengan mempartisi menjadi 4 sebagai berikut:

```
> opar <- par()  
> par(mfcol = c(2, 2))  
> plot(aov.spray)  
> par(opar)  
> termplot(aov.spray, se=TRUE, partial.resid=TRUE,  
rug=TRUE)
```

Hasil program di atas berupa plot grafik ditampilkan pada gambar 5.1:



Gambar 5.1 hasil penerapan fungsi aov dengan plot.

5.3 Analisa Regresi Linear

Bentuk matematis dari model regresi linier sederhana dapat dinyatakan sebagai berikut (Draper dan Smith, 1981)

$$\hat{Y} = \beta_0 + \beta_1 X$$

atau

$$Y = \beta_0 + \beta_1 X + \epsilon$$

dengan

Y = nilai pengamatan dari peubah atau variabel tak bebas (response)

X = nilai pengamatan dari peubah atau variabel bebas (prediktor)

\hat{Y} = nilai ramalan atau prediksi dari peubah atau variabel tak bebas (respon)

ε = nilai kesalahan ramalan

β_0 = intersep atau konstanta

β_1 = *slope* atau koefisien kemiringan model regresi.

Sebagai contoh, gunakan data berikut:

```
> age=18:29
```

```
> height=c(76.1,77,78.1,78.2,78.8,79.7,79.9,81.1,81.2,81.8,82.8,83.5)
```

Lalu, gunakan fungsi `lm` untuk menghitung linear model berikut :

```
> res=lm(height~age)
```

```
> res
```

```
Call:
```

```
lm(formula = height ~ age)
```

```
Coefficients:
```

```
(Intercept)          age
    64.928         0.635
```

Lalu digunakan fungsi `abline` untuk menghasilkan garis “Line of best fit”

```
> abline(res)
```

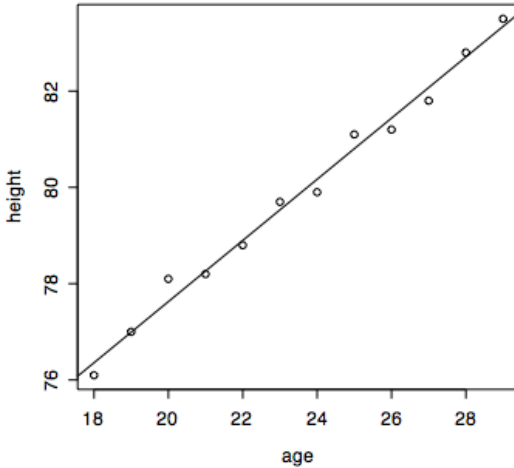


Figure 2. fungsi **abline** yang membuat "Line of Best Fit" pada scatterplot.

Latihan:

1. Jelaskan mengenai regresi linear, dan berikan contoh penerapan regresi linear pada R
2. Jelaskan dan berikan contoh Analysis of variance (ANOVA) pada R
3. Pelajari mengenai berbagai macam tabel distribusi pada R seperti tabel 5.1 di bawah:

Tabel 5.1 Tabel distribusi yang umum pada statistika

Distribusi	Nama di R	Argument tambahan
beta	beta	shape1, shape2, ncp
binomial	binom	size, prob
Cauchy	cauchy	location, scale
chi-squared	chisq	df, ncp
exponential	exp	rate
F	f	df1, df2, ncp
gamma	gamma	shape, scale
geometric	geom	prob
hypergeometric	hyper	m, n, k
log-normal	lnorm	meanlog, sdlog
logistic	logis	location, scale
negative binomial	nbinom	size, prob
normal	norm	mean, sd
Poisson	pois	lambda

signed rank	signrank	n
Student's t	t	df, ncp
uniform	unif	min, max
Weibull	weibull	shape, scale
Wilcoxon	wilcox	m, n

Bab 6

Membuat Function

Tujuan Instruksional Umum

- Setelah mempelajari Bab ini, mahasiswa mampu membuat loop dan user defined function pada R.

Tujuan Instruksional Khusus

- Mahasiswa mampu memrogram loop dan user defined function yang diinginkan pada R melalui konsol
- Mahasiswa mampu menggunakan function yang dibuat pada R

6.1 Loop dan Vektorisasi

R memiliki beberapa fitur untuk pemrograman yang mirip dengan bahasa C. Format loop dan pengecekan kondisi pada R adalah :

```
for (name in expr_1) expr_2
```

```
if (expr_1) expr_2 else expr_3
```

Misal, jika kita memiliki sebuah vektor x , dan tiap elemen x dengan nilai b , kita ingin memberikan nilai 0 ke variable y , maka programnya sebagai berikut:

```
>y <- numeric(length(x))  
for (i in 1:length(x)) if (x[i] == b) y[i] <- 0 else y[i] <- 1
```

Beberapa instruksi dapat dieksekusi berbarengan jika diletakkan di dalam kurung kurawal :

```
for (i in 1:length(x)) {  
  y[i] <- 0  
  ...  
}  
if (x[i] == b) {  
  y[i] <- 0  
  ...  
}
```

Selain itu, kita dapat mengeksekusi instruksi selagi kondisi true menggunakan while

```
while (myfun > minimum) {  
  ...  
}
```

Kita juga dapat menggunakan vektorisasi untuk loop pada pemberian nilai berulang, misalnya


```
> z <- numeric(length(x))
> for (i in 1:length(z)) z[i] <- x[i] + y[i]
```

6.2 Membuat Fungsi Sendiri

Membuat fungsi sendiri sangat dibutuhkan pada pemrograman tingkat lanjut R. Biasanya fungsi kita buat agar program utama kita mampu lakukan berbagai instruksi dengan baik pada 1 file R. Format penulisan fungsi adalah sebagai berikut:

```
name <- function(arg_1, arg_2, ...)
```

Sebagai contoh, jika kita ingin membuat fungsi bernama `myfun` yang membutuhkan 2 parameter untuk plot data, dan melakukan beberapa aksi, dapat digunakan contoh berikut yang dibuat pada script baru dan disimpan dengan nama file **buatfungsi.r**.

```
myfun <- function(S, F)
{
  data <- read.table(F)
  plot(data$V1, data$V2, type="l")
  title(S)
}
```

Lalu memanggil fungsi tersebut dapat digunakan perintah di bawah ini:

```
layout(matrix(1:3, 3, 1))
```

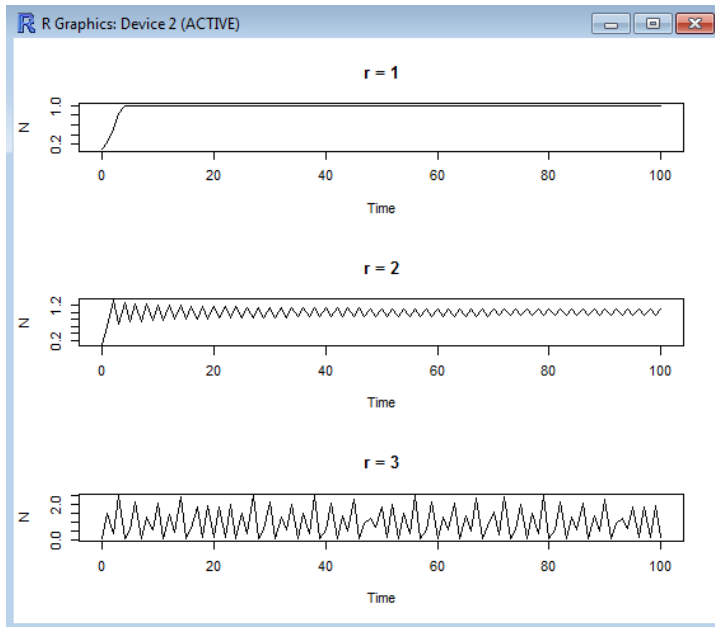
```
myfun("swallow", "Swal.dat")  
myfun("wren", "Wrenn.dat")  
myfun("dunnock", "Dunn.dat")
```

Contoh lainnya dengan parameter fungsi yang sudah kita tentukan sebagai berikut:

```
ricker <- function(nzero, r, K=1, time=100, from=0, to=time)  
{  
  N <- numeric(time+1)  
  N[1] <- nzero  
  for (i in 1:time) N[i+1] <- N[i]*exp(r*(1 - N[i]/K))  
  Time <- 0:time  
  plot(Time, N, type="l", xlim=c(from, to))  
}
```

Lalu jalankan fungsi diatas dengan memanggilnya sebagai berikut:

```
> layout(matrix(1:3, 3, 1))  
> ricker(0.1, 1); title("r = 1")  
> ricker(0.1, 2); title("r = 2")  
> ricker(0.1, 3); title("r = 3")
```



Gambar 6.1 Contoh grafik dari hasil pemanggilan fungsi

Dari gambar 6.1 diatas, terlihat bahwa dengan memiliki kemampuan pemrograman fungsi, program yang dijalankan bs lebih dinamis dan handal.

Latihan:

1. Buatlah contoh program pemanggilan fungsi penghitungan jumlah integral lipat 3.
2. Jalankan program recursive untuk puzzle berikut:

```

    area <- function(f, a, b, eps = 1.0e-06,
lim = 10) {
    fun1 <- function(f, a, b, fa, fb, a0,
eps, lim, fun) {
    ## function 'fun1' is only visible
inside 'area'
    d <- (a + b)/2
    h <- (b - a)/4
    fd <- f(d)
    a1 <- h * (fa + fd)
    a2 <- h * (fd + fb)
    if(abs(a0 - a1 - a2) < eps || lim ==
0)
        return(a1 + a2)
    else {
        return(fun(f, a, d, fa, fd, a1,
eps, lim - 1, fun) +
            fun(f, d, b, fd, fb, a2,
eps, lim - 1, fun))
    }
    }
    fa <- f(a)
    fb <- f(b)
    a0 <- ((fa + fb) * (b - a))/2
    fun1(f, a, b, fa, fb, a0, eps, lim,
fun1)
}

```

Bab 7

Paket Aktuar

Paket Actuar

Paket aktuar merupakan paket favorit untuk analisa asuransi/aktuaria. Anda harus mengunduh paket aktuar tersendiri lalu disertakan pada program R seperti pada petunjuk bagian ***lampiran 1. Instalasi Program R dan Actuar***. Secara umum paket aktuar dapat dipisahkan menjadi 4 kategori penerapan utama:

7.1 Loss Distribution Modeling

Peran utama aktuari ialah memodelkan jumlah klaim untuk pembuatan *rate*, atau evaluasi resiko lainnya. Paket aktuar berisi berbagai fungsi untuk *loss distribution modeling dan risk theory*. Program R juga menyertakan fungsi-fungsi probabilitas (*probability laws*) untuk menghitung *probability density function (pdf)* dan *cumulative distribution function (cdf)*.

Family	Distribution	Root
Transformed beta	Transformed beta	trbeta
	Burr	burr
	Loglogistic	llogis
	Paralogistic	paralogis
	Generalized Pareto	genpareto
	Pareto	pareto
	Inverse Burr	invburr
	Inverse Pareto	invpareto
	Inverse paralogistic	invparalogis
Transformed gamma	Transformed gamma	trgamma
	Inverse transformed gamma	invtrgamma
	Inverse gamma	invgamma
	Inverse Weibull	invweibull
	Inverse exponential	invexp
Other	Loggamma	lgamma
	Single parameter Pareto	pareto1
	Generalized beta	genbeta

Tabel 7.1. Probability laws yang didukung oleh actuar

Actuar juga menyediakan fasilitas grouped data, dengan contoh sebagai berikut:

Group	Frequency (Line 1)	Frequency (Line 2)
(0, 25]	30	26
(25, 50]	31	33
(50, 100]	57	31
(100, 150]	42	19
(150, 250]	65	16
(250, 500]	84	11

Tabel 7.2. Contoh data percobaan

Menggunakan R, dapat deprogram sebagai berikut:

```
> x <- grouped.data(Group = c(0, 25, 50, 100,
+ 150, 250), Line.1 = c(30, 31, 57, 42, 65, 84),
+ Line.2 = c(26,
+ 33, 31, 19, 16, 11))
```

Maka objek x akan disimpan secara internal sebagai kelas
list

```
> class(x)
[1] "grouped.data" "data.frame"
```

Dengan method print, objek ini dapat ditampilkan dengan
cara

```
> x
  Group Line.1 Line.2
1 (0, 25] 30 26
2 (25, 50] 31 33
3 (50, 100] 57 31
4 (100, 150] 42 19
5 (150, 250] 65 16
6 (250, 500] 84 11
```

Kemudian, paket mendukung method ekstrasi dan
pergantian untuk “grouped.data” menggunakan operator [dan [<-.

- (i) Ekstraksi vector dari batasan group

```
> x[, 1]
[1] 0 25 50 100 150 250 500
```

- (ii) Ekstraksi vector atau matrik dari frekwensi group

```
> x[, -1]
Line.1 Line.2
1 30 26
2 31 33
3 57 31
4 42 19
5 65 16
6 84 11
```

- (iii) Ekstraksi sebuah subset dari keseluruhan objek

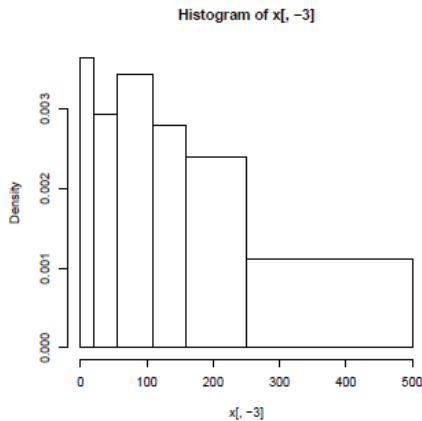
```
> x[1:3, ]
Group Line.1 Line.2
1 (0, 25] 30 26
2 (25, 50] 31 33
3 (50, 100] 57 31
```

Untuk operasi pergantian (*replacement 1 atau lebih group*), berikut contohnya:

```
> x[1, 2] <- 22
> x
Group Line.1 Line.2
1 (0, 25] 22 26
2 (25, 50] 31 33
3 (50, 100] 57 31
4 (100, 150] 42 19
5 (150, 250] 65 16
```


6 (250, 500] 84 11

Serta dapat menampilkan histogram dengan sangat mudah sebagai berikut:



Gambar 7.1. Histogram menggunakan R

Paket R berisi fungsi `summary` untuk objek data group menggunakan fungsi `mean`

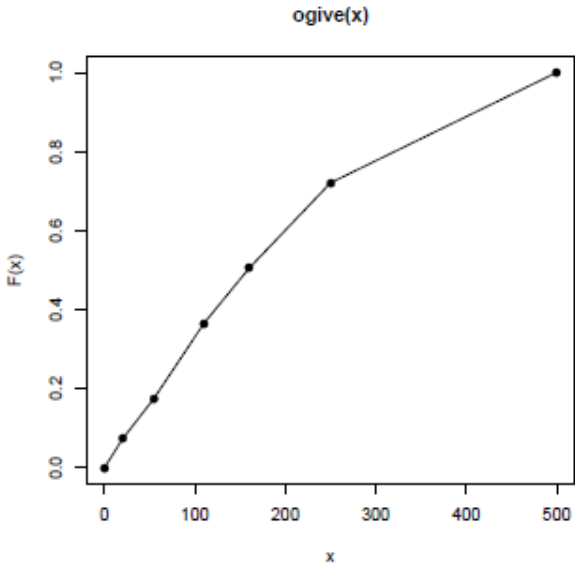
```
> mean(x)
Line.1 Line.2
188.0 108.2
```

R menyediakan juga fungsi dataset yang mudah, misalnya klaim dental individu dan klaim dental tergroup

```
> data("dental")
> dental
```

```
[1] 141 16 46 40 351 259 317 1511 107 567
```

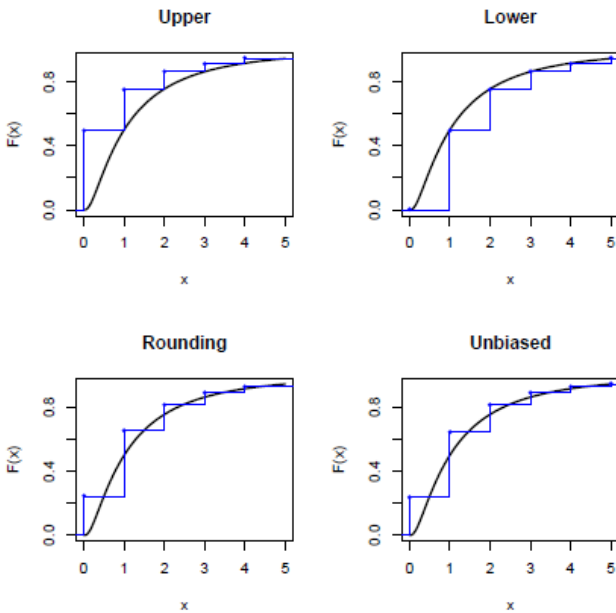
```
> data("gdental")
> gdental
  Cj      nj
1 (0, 25]   30
2 ( 25, 50] 31
3 ( 50, 100] 57
4 (100, 150] 42
5 (150, 250] 65
6 (250, 500] 84
7 (500, 1000] 45
8 (1000, 1500] 10
9 (1500, 2500] 11
10 (2500, 4000] 3
```



Gambar 7.2. Plot data x

7.2 Risk Theory

Risk theory mengacu pada teknik pemodelan dan pengukuran resiko yang diasosiasikan dengan portofolio dari kontrak-kontrak asuransi. Beberapa teknik numeric untuk menghitung distribusi agregat klaim membutuhkan distribusi aritmatika klaim yang diskrit yaitu distribusi yang didefinisikan pada $0; h; 2h; \dots$ untuk beberapa step (atau span, atau lag) h . Saat ini proses pendiskritan (*discretization*) memiliki 4 metode antara lain upper discretization, lower discretization, rounding dan unbiased.



Gambar 7.3. Perbandingan 4 metode diskretisasi

Contoh penerapan pada R:

```
> fx <- discretize(pgamma(x, 2, 1), method = "upper",  
+ from = 0, to = 17, step = 0.5)  
> fx <- discretize(pgamma(x, 2, 1), method = "unbiased",  
+ lev = levgamma(x, 2, 1), from = 0, to = 17, step = 0.5)
```

Fungsi `aggregateDist` dapat menghitung atau aproksimasi `aggregate claim amount random variable S`.

```
> fx <- discretize(pgamma(x, 2, 1), from = 0, to = 22,  
+ step = 0.5, method = "unbiased", lev = levgamma(x, 2,  
1))  
R> Fs <- aggregateDist("recursive", model.freq =  
"poisson",  
+ model.sev = fx, lambda = 10, x.scale = 0.5)  
> summary(Fs)
```

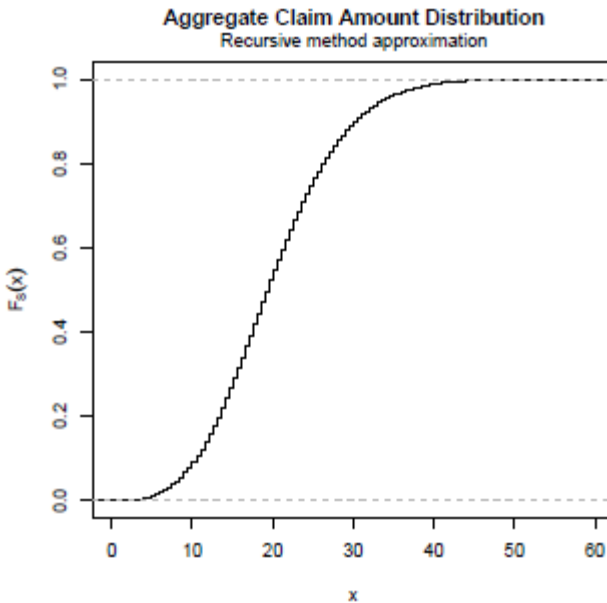
Aggregate Claim Amount Empirical CDF:
Min. 1st Qu. Median Mean 3rd Qu. Max.
0.0 14.5 19.5 20.0 25.0 71.0

Objek `Fs` berisi `cdf empiris`

```
> knots(Fs)  
[1] 0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5  
[13] 6.0 6.5 7.0 7.5 8.0 8.5 9.0 9.5 10.0 10.5 11.0 11.5  
[25] 12.0 12.5 13.0 13.5 14.0 14.5 15.0 15.5 16.0 16.5 17.0 17.5  
[37] 18.0 18.5 19.0 19.5 20.0 20.5 21.0 21.5 22.0 22.5 23.0 23.5  
[49] 24.0 24.5 25.0 25.5 26.0 26.5 27.0 27.5 28.0 28.5 29.0 29.5  
[61] 30.0 30.5 31.0 31.5 32.0 32.5 33.0 33.5 34.0 34.5 35.0 35.5  
[73] 36.0 36.5 37.0 37.5 38.0 38.5 39.0 39.5 40.0 40.5 41.0 41.5  
[85] 42.0 42.5 43.0 43.5 44.0 44.5 45.0 45.5 46.0 46.5 47.0 47.5  
[97] 48.0 48.5 49.0 49.5 50.0 50.5 51.0 51.5 52.0 52.5 53.0 53.5  
[109] 54.0 54.5 55.0 55.5 56.0 56.5 57.0 57.5 58.0 58.5 59.0 59.5  
[121] 60.0 60.5 61.0 61.5 62.0 62.5 63.0 63.5 64.0 64.5 65.0 65.5  
[133] 66.0 66.5 67.0 67.5 68.0 68.5 69.0 69.5 70.0 70.5 71.0
```

Untuk menampilkan grafik dari plot data ini menggunakan fungsi `plot`:

```
> plot(Fs, do.points = FALSE, verticals = TRUE, xlim = c(0, 60))
```



Gambar 7.4 Plot aggregate claim amount distribution

7.3 Simulasi Compound Hierarchical Model

Function simul mensimulasikan portofolio data dimana komoonen frekwensi dan severity memiliki sebuah struktur hirarkis, berikut contohnya:

```
> nodes <- list(cohort = 2, entity = c(4, 3), year = c(4,  
+ 4, 4, 4, 5, 5, 5))  
Journal of Statistical Software 27  
> mf <- expression(cohort = rexp(2), entity = rgamma(cohort,  
+ 1), year = rpois(weights * entity))
```

```

> ms <- expression(cohort = rnorm(2, sqrt(0.1)), entity =
  rnorm(cohort,
+ 1), year = rlnorm(entity, 1))
> wjt <- runif(31, 0.5, 2.5)
> pf <- simul(nodes = nodes, model.freq = mf, model.sev = ms,+
  weights = wjt)

```

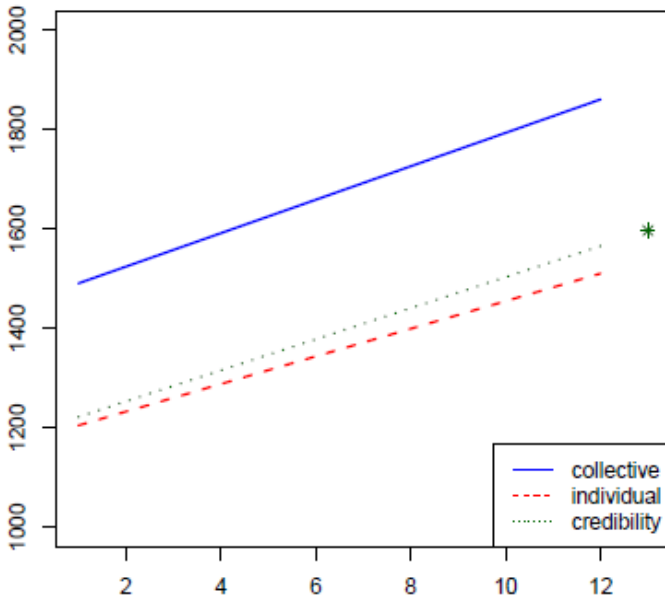
7.4 Credibility Theory

Credibility model ialah tool aktuaria untuk mendistribusikan premium diantara group heterogen, dengan kata lain metode prediksi. Salah satu fungsi yang digunakan ialah `cbind` sebagai berikut:

```

> X <- cbind(cohort = c(1, 2, 1, 2, 2), hachemeister)
> fit <- cm(~cohort + cohort:state, data = X, ratios =
  ratio.1:ratio.12,
+ weights = weight.1:weight.12, method = "iterative")
> fit
Call:
cm(formula = ~cohort + cohort:state, data = X, ratios =
  ratio.1:ratio.12,
  weights = weight.1:weight.12, method = "iterative")
Structure Parameters Estimators
Collective premium: 1746
Between cohort variance: 88981
Within cohort/Between state variance: 10952
Within state variance: 139120026
The function returns a _tted model object

```



Gambar 7.5 Hasil garis regresi dari collective, individual dan credibility.

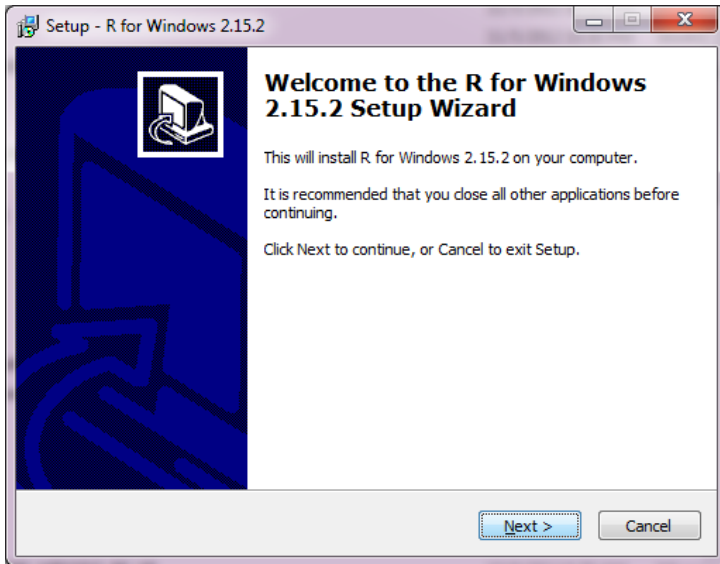
Lampiran 1

Instalasi Program R dan Actuar

Instalasi R

Paket terbaru R yang dapat diinstal yaitu R 2.15.2 for Windows dan actuar 1.1.5.

Berikut tampilan proses instalasi R



Gambar 1. Tampilan instalasi program R

R 2.15.2 for Windows

=====

This distribution contains a binary distribution of R-2.15.2 to run on Windows XP and later (including 64-bit versions of Windows) on ix86 and x86_64 chips. It is designed to be as close as possible to the implementation on Unix, but see the list of differences below. The port was by Guido Masarotto, Brian Ripley and Duncan Murdoch.

Installation

=====

The distribution is distributed as an installer R-2.15.2-win.exe. Just run this for a Windows-style installer. It contains all the R components, and you can select what you want installed.

For more details, including command-line options for the installer and how to uninstall, see the rw-FAQ. If you are reading this as part of an installed version of R, that is both a file rw-FAQ in this directory and a file doc/html/rw-FAQ.html.

Usage

=====

There are two versions of the R executable in R-2.15.2\bin\i386 (32-bit) and R-2.15.2\bin\x64 (64-bit). By default only the first is

installed on 32-bit versions of Windows, and both on 64-bit OSes.

GUI:

Rgui.exe runs as a standard Windows GUI executable and provides an R console in its own window. It takes the standard R command-line arguments; these are most easily set by making a shortcut to `...\R-2.15.2\bin\i386\Rgui.exe` (or `x64\Rgui.exe`) and adding the arguments in the Target field. Set the 'Start in' field of the shortcut to the directory you want to use as the working directory (where workspaces and files are saved to and loaded from). The installer creates a shortcut on the desktop for you (by default).

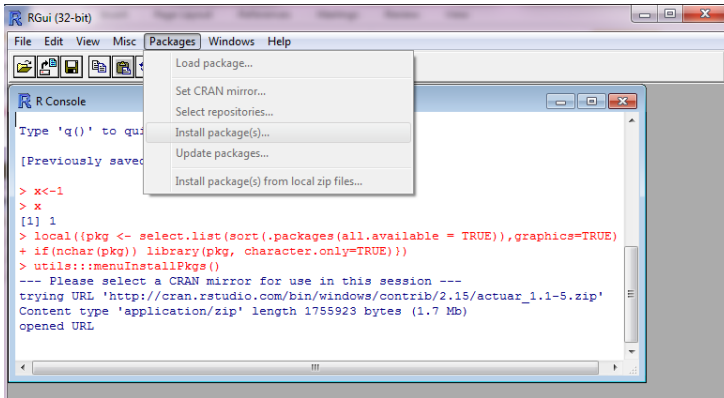
You have a choice of interface: SDI (separate windows) or MDI (multiple child windows in one large window). MDI is the default: select SDI by adding `--sdi` in the Target field: this can also be selected from the installer.

Command-line editing is available: see Help | Console for details.

The menus provide shortcuts to some R commands such as `help`, `help.start`, `apropos`, `ls`, `load`, `save.image`, `search`.

Instalasi Aktuar

Anda harus meload paket Aktuar dari Program yang terkoneksi internet. Berikut tampilan instalasi paket actuar:



Gambar 2. Tampilan instalasi paket actuar

Daftar Pustaka

1. **Alain F. Zur et.al, A Beginner's Guide to R**, Springer, 2009.
2. Nicholas Walliman, **Research Method Basics**, Roudedge Publisher, 2011
1. Emanuel Paradis, **R for the Beginner**, Institut des Sciences de l' _ Evolution, Paris, 2005
2. Suhartono, **Analisis Data Statistik dengan R**, Lab Statistika ITS, 2010
3. Annette J. Dobson , **An Introduction to Generalized Linear Models**, Chapman and Hall, London., 1990
4. Peter McCullagh and John A. Nelder, **Generalized Linear Models. Second edition**, Chapman and Hall, London, 1989
5. John A. Rice (1995), **Mathematical Statistics and Data Analysis**. Second edition. Duxbury Press, Belmont, CA, 1995.
6. <http://cran.r-project.org/>
7. www.Widodo.com
8. <http://cran.r-project.org/doc/manuals/r-release/R-intro.html>

Tentang Penulis



Dr. Widodo Budiharto adalah Akademisi dan Peneliti Senior di School of Computer Science, BINUS University- Jakarta. Bidang riset beliau adalah Robot Vision, Kecerdasan Buatan dan Applied Science. Beliau akan terus memdedikasina dirinya

mengembangkan teknologi dan sains di Indonesia melalui buku-bukunya. Beliau dapat dihubungi di Widodo@widodo.com.



Ro'fah Nur Rachmawati, S.Si., M.Si adalah Akademisi di School of Computer Science, BINUS University- Jakarta. Bidang Riset beliau adalah Proses Stokastik dan pemodelan Matematika. Beliau dapat dihubungi di rrachmawati@binus.edu

Buku ini diterbitkan oleh:

**Halaman Moeka Publishing:
Penerbit dan Jasa Penerbitan Buku**

<http://halamanmoeka.blogspot.com>

Menyediakan:

Jasa Penyuntingan | Layout | Desain Cover | Ilustrasi
Penerbitan Buku Indie | Cetak Buku | Cetak Ebook

Email: *terbit.buku@gmail.com*
