

Mengenal Software Statistika “R” sebagai *Datamining Tool* di Linux

Sigit Wahyu Kartiko

gsigit[at]gmail[dot]com

Lisensi Dokumen:

Copyright © 2003-2006 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Pendahuluan

Konsep datamining baru-baru ini menjadi topik yang cukup hangat dikalangan dunia komputasi. Sebagai sebuah metode, datamining merupakan serangkaian proses penggalian informasi ke dalam database transaksi yang biasanya sudah lama dilupakan oleh pengambil keputusan. Bagaimana cara menggali informasi memerlukan berbagai aspek pengetahuan diantaranya adalah statistika.

Tool statistika yang sedang naik daun dikalangan open source salah satunya adalah R. Tidak hanya menawarkan lingkungan sistem yang lengkap, R juga bahasa yang berbasis object oriented dengan interface pemrograman C, java dan sebagainya. Terbukti melalui beberapa jurnal ilmiah bahwa R adalah tools statistika yang populer dikalangan peneliti statistika. Bagi kalangan yang bermadzab open-source tidak salahnya mengeksplorasi tools yang cukup lengkap ini dalam melakukan analisis data.

Sejarah R

Versi pertama R diluncurkan pada tahun 1992 oleh Ross Ihaka dan Robert Gentleman (1996) (singkatan R berasal dari kedua nama tersebut) yang keduanya dari the University of Auckland. Diawal pengembangannya, proyek R dibuat dengan bahasa LISP serta hanya diimplementasikan di macintosh dengan bahasa semantik Scheme. Pada saat ini source code R yang dibuat telah bersifat multiplatform, sehingga dapat dikompilasi dan dijalankan diberbagai sistem operasi berbasis *NIX dan Windows. R dikembangkan secara intensif oleh *R-core Team* yang anggotanya terdiri dari 17 orang ahli statistika (<http://www.r-project.org/contributors.html>) dan melibatkan banyak kontributor lain dari berbagai institusi diseluruh dunia. Seperti halnya linux, source code R tersedia untuk dibaca, dianalisis, identifikasi bug dan modifikasi.

Bahasa R berbasis bahasa S yang dibangun di Bell laboratories di tahun 80-an. Dengan kata lain R adalah implementasi bahasa S yang sudah dibangun oleh Rick Becker, John Chambers dan Allan Wilks, yang juga membentuk dasar dari sistem S-Plus. Dengan demikian sintak R hampir identik dengan S dengan perbedaan yang tidak terlalu banyak. Sebagai contoh mulai R versi 1.7.0 assign variabel diperbolehkan menggunakan “=” yang mana di S assign variabel menggunakan “<-”.

Mengapa menggunakan R?

Analisis data tidak dapat dilakukan tanpa melakukan teknik yang tepat. Artinya, analisis yang handal seharusnya memanfaatkan keakurasian dan kecepatan perhitungan dengan paket komputasi statistik (*statistical computing package*). Begitu banyak pilihan paket tools statistika yang kita gunakan seperti SPSS, EViews, STATA, MINITAB, SAS dan sebagainya. Masing-masing tool didesain untuk pengguna yang memiliki karakteristik berbeda-beda dan tentunya memiliki kelebihan dan kekurangan. Menurut Ihaka dan Gentleman (1996), ada beberapa alasan mengapa menggunakan R, diantaranya adalah:

1. Serba guna (*versatile*)

R adalah bahasa pemrograman, sehingga tidak ada batasan bagi pengguna untuk memakai prosedur yang hanya terdapat pada paket-paket yang standar. Bahkan pemrograman R adalah berorientasi obyek dan memiliki banyak library yang sangat bermanfaat yang dikembangkan oleh kontributor. Pengguna bebas menambah dan mengurangi library tergantung kebutuhan. R juga memiliki interface pemrograman C, python, bahkan java yang tentu saja berkat jerih payah kontributor aktif proyek R. Jadi selain bahasa R ini cukup pintar, penggunaannya pun bisa menjadi lebih pintar dan kreatif.

Beberapa analisis yang membutuhkan fungsi lanjutan memang ada yang belum tersedia dalam R. Tidak berarti R tidak menyediakan fasilitas tersebut, namun lebih karena faktor waktu. Jadi hanya menunggu waktu saja package lanjutan tersebut tersedia.

2. Interaktif (*interactive*)

Pada saat ini analisis data membutuhkan pengoperasian yang interaktif. Apalagi jika data yang dianalisis adalah data yang bergerak. R dilengkapi dengan konektivitas ke database server, olap, maupun format data web service seperti XML, spreadsheet dan sebagainya. Sehingga apabila data set berubah hasil analisis pun dapat segera ikut berubah (*real time*).

3. Berbasis S yaitu turunan dari tool statistik komersial S-Plus.

R hampir seluruhnya kompatibel dengan S-Plus. Artinya sebagian besar kode program yang dibuat oleh S dapat dijalankan di S-plus kecuali fungsi-fungsi yang sifatnya *add-on packages* atau tambahan yang dibuat oleh kontributor proyek R.

4. Populer.

Secara umum SAS adalah software statistika komersial yang populer, namun demikian R atau S adalah bahasa yang paling populer digunakan oleh peneliti di bidang statistika. Beberapa tulisan berupa jurnal statistika mengkonfirmasi kebenaran hal ini. R juga populer untuk aplikasi kuantitatif dibidang keuangan.

Secara fair Ihaka dan Gentlemen menyebutkan kekurangan dari R adalah tidak mudah untuk dipelajari. Beberapa requirement awal diperlukan sebelum memperoleh manfaat dari R diperoleh seperti pemahaman tentang dasar-dasar pemrograman. Namun menurut pendapat penulis, user linux seharusnya sudah memiliki pemahaman tentang dasar-dasar pemrograman sehingga akan lebih mudah dan produktif dalam mengeksplorasi tools statistika yang hebat ini.

Instalasi R

Bagi pengguna linux instalasi R cukup mudah dilakukan. Penulis akan memberikan contoh bagaimana menginstall di linux ubuntu. Bagi pengguna selain ubuntu dapat menggunakan *source code* yang langsung dapat didownload dari website R.

Instalasi R melalui “apt” di ubuntu

Dengan menggunakan apt-get kita dapat secara mudah mendapatkan paket R ini melalui *repository* ubuntu. Ketikkan perintah berikut pada console (huruf R menggunakan kapital):

```
$ sudo apt-get install r-base-dev r-base-core
```

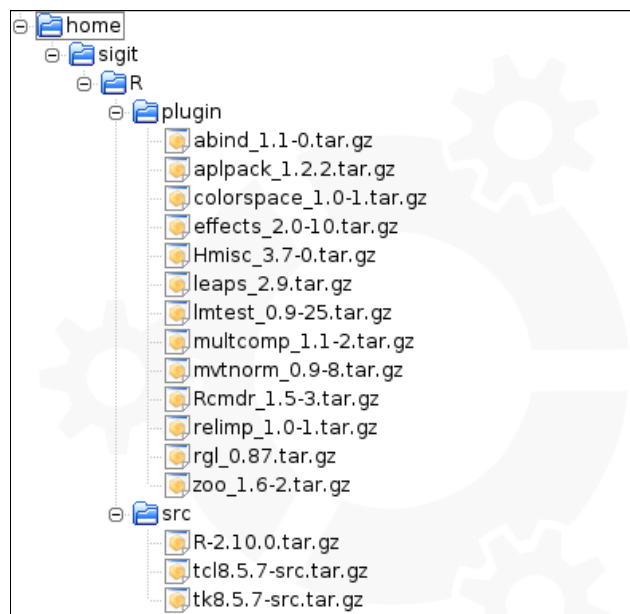
Jika muncul prompt password, ketikkan password user ubuntu anda lalu tungguhlah beberapa saat sampai proses download dan instalasi selesai.

Mengkompilasi R dari source

Apabila anda ingin menggunakan versi terbaru bahkan versi *heavy development* R anda dapat mengkompilasi langsung kode sumbernya. Penulis akan mencontohkan source R dan beberapa source aplikasi lain yang penting terkait dengan R (**gambar 1**). Anda harus melengkapi kompiler anda dengan beberapa program tambahan sebagai contoh kompiler anda harus sudah tersedia '**gfortran**' (kompiler fortran), jadi tidak hanya '**gcc**' saja.

1. Membuat direktori kerja

Langkah pertama yang anda persiapkan adalah menentukan direktori tempat kita bekerja. Sebagai contoh disini penulis menggunakan direktori /home/sigit/R sebagai direktori kerja. Susunan direktorinya dan filenya adalah sebagai berikut:



Gambar 1: Contoh direktori kerja instalasi R

2. Menginstal tcl-tk

Tcl-tk digunakan untuk memanfaatkan fasilitas plugin GUI untuk **R** yaitu **Rcmdr**. Cara instalasinya adalah ketikkan perintah ini pada prompt lalu enter di setiap akhir baris sebagai berikut.

```
$ cd /home/sigit/R/src/  
$ tar -xvzf tcl8.5.7-src.tar.gz  
$ cd tcl8.5.7/unix/  
$ sudo ./configure && sudo make && sudo make install  
$ tar -xvzf tk8.5.7-src.tar.gz  
$ cd tk8.5.7/unix/  
$ sudo ./configure && sudo make && sudo make install
```

3. Compile dan Install R

Setelah itu tiba waktunya kita menginstal R. Perintah-perintah berikut ini silakan diketik lalu enter disetiap akhir baris sebagai berikut:

a. Menuju direktori kerja:

```
$ cd /home/sigit/R/src/
```

b. Ekstrak source R (versi 2.10):

```
$ tar xvzf R-2.10.0.tar.gz  
$ cd R-2.10.0
```

c. Atur konfigurasi kompilasi, dalam hal ini termasuk instalasi syarat harus terpenuhi adalah instalasi tcl dan tk terlebih dahulu¹. Selain itu tetapkan letak program binari R hasil kompilasi. Dalam tutorial ini dicontohkan di “/home/sigit/R/R-2.10.0/program”.

```
$ sudo ./configure --with-tcltk --with-tcl-config=/home/sigit/R/src/tcl8.5.7/unix/tclConfig.sh  
--with-tk-config=/home/sigit/R/src/tk8.5.7/unix/tkConfig.sh --prefix=/home/sigit/R/R-  
2.10.0/program --enable-R-shlib
```

d. Lakukan kompilasi dan mengizinkan direktori “library” sebagai tempat menaruh file-file plugin R.

```
$ sudo make  
$ sudo make install  
$ sudo chmod 775 -R /home/sigit/R/R-2.10.0/program/lib/R/library/
```

e. Menambahkan variabel environment path

Agar perintah R dapat dipanggil oleh console maka perlu menambahkan variable environment PATH dalam difile .bashrc yang terletak di direktori /home/sigit/. Editlah file /home/sigit/.bashrc menggunakan editor favorit anda dan tambahkan baris berikut:

```
export R_HOME=/home/sigit/R/R-2.10.0/program  
export PATH=$PATH:$R_HOME/bin
```

Setelah itu simpan dan buka console/shell baru. Cobalah tes dengan mengetikkan R di console baru tersebut. Jika tampil halaman awal (lihat sub bab “Mencoba R”) maka R berhasil dikenal

1 Cara instalasi Tcl/Tk dari source dapat dilihat melalui website www.tcl.tk/doc/howto/compile.html

sebagai aplikasi yang dapat dipanggil di *environment shell*.

4. Menginstal plugin R utama

Untuk menggunakan R dengan interface GUI maka dapat ditambahkan file-file plugin. Ketikkan perintah dibawah ini:

```
$ cd /home/sigit/R/src/plugin/  
$ R CMD INSTALL -l /home/sigit/R/R-2.10.0/program/lib/R/library/ \  
colorspace_1.0-1.tar.gz zoo_1.6-2.tar.gz mvtnorm_0.9-8.tar.gz rgl_0.87.tar.gz \  
relimp_1.0-1.tar.gz multcomp_1.1-2.tar.gz lmtest_0.9-25.tar.gz leaps_2.9.tar.gz \  
Hmisc_3.7-0.tar.gz effects_2.0-10.tar.gz aplpack_1.2.2.tar.gz abind_1.1-0.tar.gz \  
Rcmdr_1.5-3.tar.gz
```

Mencoba R

Apabila proses instalasi telah selesai maka sekarang saatnya mencoba R. Ketikkan perintah berikut:

```
$ R
```

Jika muncul tampilan dibawah ini berarti R berhasil terinstall dengan baik (versi R tergantung versi ubuntu anda).

```
R version 2.10.0 (2009-10-26)  
Copyright (C) 2009 The R Foundation for Statistical Computing  
ISBN 3-900051-07-0  
  
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.  
Natural language support but running in an English locale  
  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
[Previously saved workspace restored]  
  
>
```

Mencoba demo dalam *package/plugin* bawaan

Hasil instalasi tersebut mengandung package atau plugin yang sudah dapat di coba.. Beberapa package menyediakan demonstrasi berupa file dan perintah yang dapat langsung di eksekusi. Mari kita mencoba demo yang sudah tersedia. Ketikkan saja perintah-perintah di bawah ini.

```
> demo()
```

Hasilnya adalah sebagai berikut.

```

Demos in package 'base':
is.things      Explore some properties of R objects and
                is.FOO() functions. Not for newbies!
recursion     Using recursion for adaptive integration
scoping       An illustration of lexical scoping.

Demos in package 'graphics':
Hershey       Tables of the characters in the Hershey vector
                fonts
Japanese      Tables of the Japanese characters in the
                Hershey vector fonts
graphics      A show of some of R's graphics capabilities
image         The image-like graphics builtins of R
persp         Extended persp() examples
plotmath      Examples of the use of mathematics annotation

Demos in package 'stats':
glm.vr        Some glm() examples from V&R with several
                predictors
lm.glm        Some linear and generalized linear modelling
                examples from 'An Introduction to Statistical
                Modelling' by Annette Dobson
nlm           Nonlinear least-squares using nlm()
smooth        'Visualize' steps in Tukey's smoothers

Use 'demo(package = .packages(all.available = TRUE))'
to list the demos in all *available* packages.
    
```

Kita dapat memilih melalui fungsi **demo()** dengan cara mengetikkan perintah format "demo(nama_demo)". Sebagai contoh ketikkan perintah berikut:

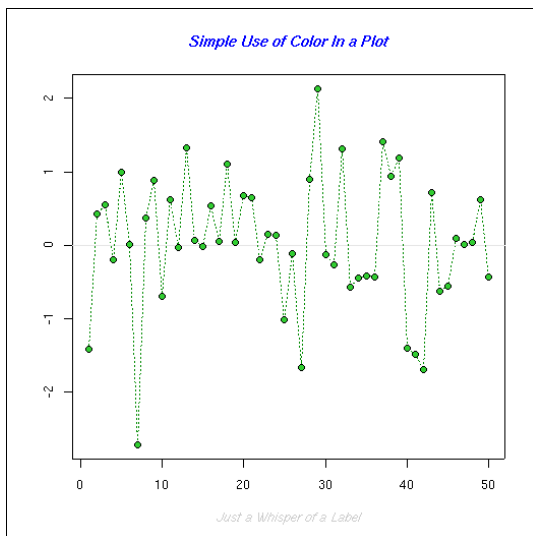
```
> demo(graphics)
```

Hasilnya adalah sebagai berikut:

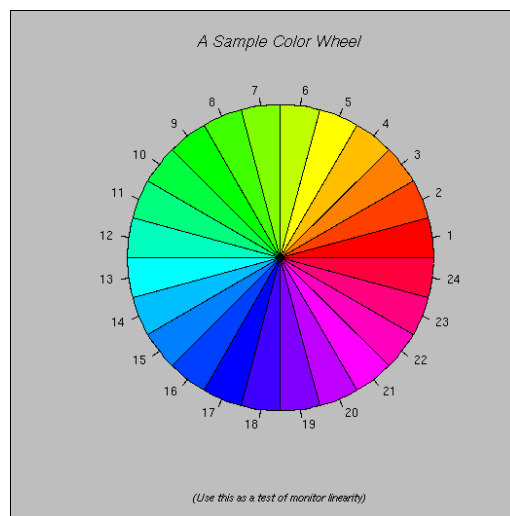
```

demo(graphics)
---- ~~~~~
Type <Return> to start :
    
```

Selanjutnya anda tekan enter untuk menampilkan demo dalam bentuk grafis.



Gambar 2: contoh grafik scatter



Gambar 3: contoh spektrum warna

R dan matematika

R dapat digunakan sebagai kalkulator. Tidak hanya sesederhana itu, R juga dapat menyelesaikan problematika matematika seperti persamaan kuadrat, matriks, kalkulus, trigonometri dan sebagainya. Sangat membantu bagi anda yang tidak ingin bersusah payah menggunakan cara-cara manual dalam menyelesaikan problematika matematika.

R sebagai kalkulator lanjutan (Rositter, 2009)

Ketikkan perintah-perintah berikut dan tekan enter

```
> pi*7  
[1] 021.99115
```

Pi atau π adalah konstanta lingkaran.

```
> 1 / 2^2 + 2 * pi  
[1] 6.533185
```

Dibawah ini perhitungan logaritma dalam satu baris masing-masing dipisahkan dengan tanda semicolon (;)

```
> log(10); log10(10); log2(10)  
[1] 2.3026  
[1] 1  
[1] 3.3219
```

Untuk membulatkan gunakan fungsi round().

```
> round(log(10))  
[1] 2
```

Untuk mencari akar persamaan kuadrat gunakan fungsi sqrt().

```
> sqrt(8)  
[1] 2.828427
```

Kita juga dapat menyelesaikan problem trigonometri. Untuk mencari $\sin(30^\circ)$, maka untuk mengubah menjadi radian harus dikalikan dengan $2\pi/360$ atau $\pi/180$.

```
> sin(30 * pi/180)  
[1] 0.5
```

Persamaan Kuadrat

Menyelesaikan persamaan kuadrat sederhana untuk mencari titik potong pada sumbu x ($y=0$) memang mudah dilakukan secara manual. Akan tetapi jika persamaan kuadratnya pangkat 3,4,5 dan seterusnya akan sangat rumit menyelesaikannya. Dengan R kita dengan sangat mudah menyelesaikannya caranya adalah dengan menggunakan fungsi "*polyroot()*".

Contoh:

1. Persamaan kuadrat $x^2 + 2x - 8 = 0$. Carilah titik potong sumbu x!

$$(x+4)(x-2) = 0$$

jadi $x = 4$ atau $x = 2$

2. Persamaan kuadrat $3 - 5/2x - 3x^2 = 0$

Dengan R kita dapat menyelesaikan dengan cara sbb.

- buka R lalu di dalam console ketikkan baris perintah berikut dan diakhiri dengan tekan enter

```
> polyroot ( c(3,-5/2,-3))
```

- hasilnya adalah:

```
[1] 0.6666667+0i -1.5000000+0i
```

Penjelasan:

- $c(3,-5/2,-3)$ artinya c dalam R adalah *vector* yaitu kumpulan nilai-nilai yang disimpan dalam satu daftar atau hampir mirip seperti array/list
- aturan memasukkan nilai dalam vector adalah dari koefisien melekat pada x yang pangkatnya paling kecil yaitu 0 sampai terbesar. dalam contoh no 2 berarti urutannya adalah
 $3x^0 - 5/2x^1 - 3x^2$
- sehingga jika x dihilangkan maka berturut-turut sesuai dengan pangkat yang ada pada x adalah 3, $-5/2$ dan -3
- jika pangkatnya lompat berarti koefisien yang melekat pada x nilainya 0
- hasil perhitungannya ada 2 nilai yaitu 0.667 (atau $2/3$) dan -1.5
- i adalah integer jadi ditambah dengan 0 integer.

Operasi Matriks

Matriks berguna sekali dalam menyelesaikan suatu persamaan. Dalam matematika dasar dikenal cara menyelesaikan suatu persamaan dengan cara substitusi dan pengurangan. Mungkin kalau persamaannya hanya terdiri dari 2 persamaan, penyelesaiannya tidak terlalu masalah. Bagaimana jika persamaannya ada 3 atau lebih. Menjadi tidak praktis jika menggunakan cara substitusi dan pengurangan.

Contoh :

1. Diketahui 2 persamaan sbb:

a) $x + 4y = 9$...persamaan 1

b) $2x + y = 4$...persamaan 2

Maka dengan cara pengurangan dapat diselesaikan dengan langkah-langkah sbb:

langkah 1: kalikan persamaan 1 dengan angka 2 sehingga menghasilkan persamaan baru sbb.

$$\begin{array}{r} x + 4y = 9 \quad | \times 2 \rightarrow 2x + 8y = 18 \\ 2x + y = 4 \quad | \times 1 \rightarrow \underline{2x + y = 4} \\ \hline 7y = 14 \end{array}$$

$$y = 2$$

langkah 2: substitusikan y pada persamaan pertama

$$x + 8 = 9$$

$$x = 9 - 8$$

$$x = 1$$

dengan demikian $x = 1$ dan $y = 2$;

2. Diketahui 3 persamaan sbb:

a) $7x - y - z = 0$

b) $10x - 2y + z = 8$

c) $6x + 3y - 2z = 7$

Jika penyelesaiannya menggunakan cara pada contoh no 1 maka menjadi tidak praktis. Dengan matriks kita dapat menyelesaikannya dengan lebih praktis. Caranya adalah dengan mengubah persamaan di atas menjadi matriks.

$$\begin{bmatrix} 7 & -1 & -1 \\ 10 & -2 & 1 \\ 6 & 3 & -2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 8 \\ 7 \end{bmatrix}$$

disederhanakan menjadi

$$A \cdot x = d$$

Bentuk umum persamaan matriks diatas adalah

$$A \cdot x = d$$

sehingga untuk mencari x maka :

$$x = A^{-1} \cdot d$$

x disini adalah nilai x, y, z yang akan kita selesaikan. A^{-1} adalah invers matriks A yang tidak lain adalah persamaan pada sisi kiri tanda = (sama dengan) dan d adalah persamaan disisi kanan tanda sama dengan.

Dengan R kita dapat menyelesaikan dengan cara sbb.

- buka R lalu di dalam console ketikkan baris-baris perintah berikut dan selalu diakhiri dengan tekan enter

```
> A <- matrix(c(7,-1,-1,10,-2,1,6,3,-2),3,3,TRUE)
> d<-matrix(c(0,8,7),3,1,TRUE)
> solve(A) %*% d
```

- hasilnya adalah:

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

- Jadi $x = 1$, $y = 3$ dan $z = 4$

Kalkulus Differensial

Untuk mengetahui perubahan posisi x menjadi x' suatu fungsi f(x) maka diperlukan suatu turunan fungsi f'(x). Perubahan ini disebut sebagai 'differens' dari sebuah variabel. Misalnya kita mempunyai fungsi $y=f(x)$. Maka dx dan dy adalah bentuk differens dari x dan y dimana dy/dx adalah turunan dari f(x).

1. Contoh 1:

Suatu persamaan : $y=x^2$

maka dapat diselesaikan sbb:

$$\frac{dy}{dx} = y' = 4x^{(4-1)} = 4x^3$$

2. Contoh 2 :

$$y = (3x^2 - 13)^3$$

Bagaimana cara menyelesaikan turunan fungsi tersebut? Betapa rumitnya jika menggunakan perhitungan manual.

Dengan R kita dapat menyelesaikan dengan cara sbb.

- buka R lalu di dalam console ketikkan baris perintah berikut dan diakhiri dengan tekan enter

```
> D(expression((3*x^2-13)^3),"x")
```

- hasilnya adalah:

```
3 * (3 * (2 * x) * (3 * x^2 - 13)^2)
```

atau bentuk sederhananya adalah:

$18x(x^2-13)^2$

R dan Statistika Deskriptif

Dalam paket standar R terdapat contoh *dataframe* atau tabel data yang sudah tersedia. Sebagai contoh telah tersedia *dataframe* “cars” yang terdiri dari dua kolom “speed” dan “distance”. Ketikkan perintah “cars” yang merupakan nama *dataframe* sebagai berikut:

```
> cars
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
7    10   18
8    10   26
9    10   34
10   11   17
11   11   28
12   12   14
.....
```

Dataframe cars adalah data observasi atas uji coba kecepatan kendaraan dan jarak yang berhasil ditempuh. Baris 1 menunjukkan percobaan ke-1 yaitu mobil dengan speed 4 mencapai jarak 2. Percobaan ke-2 mobil dengan speed 4 mencapai jarak 10 dst.

Kita dapat menyajikan ikhtisar data tersebut dengan fungsi *summary()*.

```
> summary(cars)
  speed          dist
Min.   : 4.0    Min.   : 2.00
1st Qu.:12.0    1st Qu.: 26.00
Median :15.0    Median : 36.00
Mean   :15.4    Mean   : 42.98
3rd Qu.:19.0    3rd Qu.: 56.00
Max.   :25.0    Max.   :120.00
```

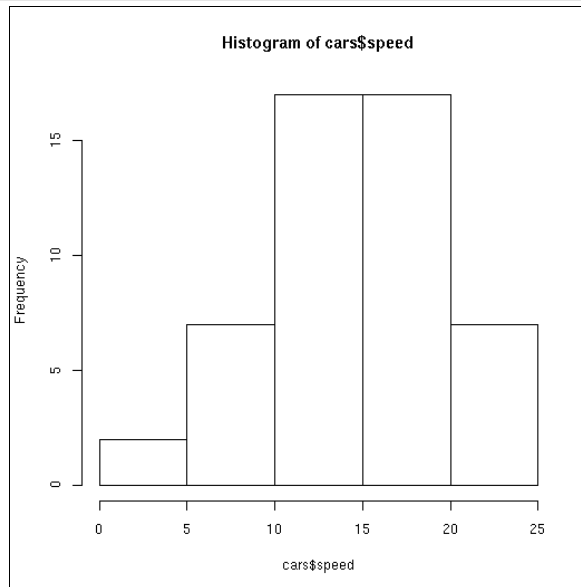
Min adalah nilai minimal, 1st Qu adalah kuartil pertama, Median adalah nilai tengah, Mean adalah nilai rata-rata, 3rd Qu adalah kuartil ketiga dan Max adalah nilai maksimum.

Untuk menampilkan grafik histogram atas frekuensi data maka kita dapat menggunakan fungsi *hist()*.

Menampilkan grafik Histogram

Berikut ini adalah cara menampilkan grafik histogram kolom speed. Ketikkan perintah menampilkan grafik tersebut dengan fungsi `hist()`. Perhatikan argumen didalam tanda kurung `cars$speed`. Maksud dari argumen ini adalah kolom `speed` pada dataframe `cars`.

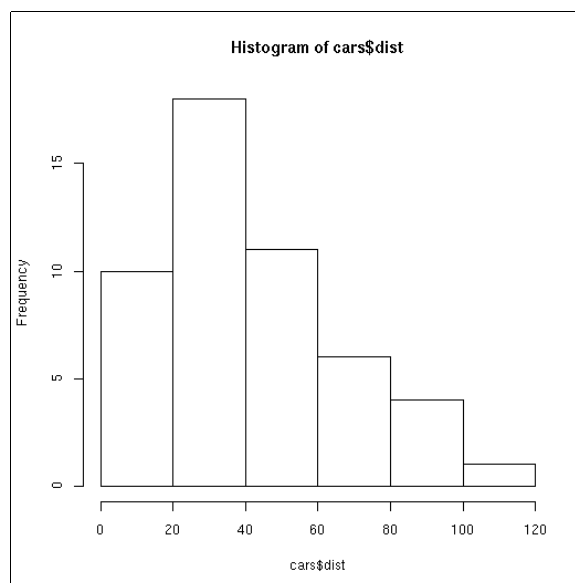
```
> hist(cars$speed)
```



Gambar 4: Histogram kolom "speed" pada dataframe "cars"

Sedangkan untuk menampilkan grafik histogram kolom `dist` pada dataframe `cars` adalah sebagai berikut

```
> hist(cars$dist)
```



Gambar 5: Histogram kolom "dist" pada dataframe "cars"

Gambar dapat disimpan menggunakan fungsi `jpeg()` atau `png()`. Fungsi-fungsi tersebut dipanggil sebelum fungsi `hist()` dan ditutup dengan fungsi `dev.off()`. Untuk lebih jelasnya kita akan menyimpan grafik tersebut berupa file bernama `hist_speed.jpg` dengan ukuran 400 pixel x 400 pixel. Maka baris perintahnya adalah sbb:

```
> jpeg("hist_speed.jpg",h=400,w=400)
> hist(cars$speed)
> dev.off()
```

File “`hist_speed.jpg`” ditaruh sejajar dengan `workspace` R kita yaitu tempat dimana kita memanggil R pada terminal.

Jika ingin menutup R pada `workspace` terminal ketikkan perintah `q()` atau `quit()`. Selanjutnya akan ditanyakan apakah anda menyimpan perintah-perintah yang sebelumnya telah diketikkan pada `workspace`. Apabila anda memilih “y” yang berarti simpan maka semua perintah tersebut berada dalam file “.Rhistory”.

Penutup

Materi pada tulisan ini merupakan pengenalan singkat salah satu tools statistika yang cukup *powerfull* dalam proses *datamining* yaitu R. Selain mengetahui instalasi R juga dapat menerapkan beberapa fungsi matematika, statistika dan *plotting* yang membantu dalam memahami materi selanjutnya yaitu pemrograman R untuk *datamining*. Anda dapat mengeksplorasi lebih lanjut cara penggunaan R melalui dokumentasi yang rapi dan lengkap dari para kontributor aktif di <http://www.R-project.org>.

Referensi

1. Ihaka, R. & Gentleman, R. (1996). *"R: A Language for Data Analysis and Graphics"*. *Journal of Computational and Graphical Statistics* 5 (3): 299–314. www.jstor.org
2. Rossiter, D. G. (2009). *Introduction to the R Project for Statistical Computing for use at ITC*. Accessed:03-03- 2010. [http:// www.itc.nl/ personal/ rossiter](http://www.itc.nl/personal/rossiter)
3. Williams, Graham J. *Rattle: A Data Mining GUI for R*. Accessed:06-04-2012. http://journal.r-project.org/archive/2009-2/RJournal_2009-2_Williams.pdf
4. <http://www.R-project.org>

Biografi Penulis



Sigit Wahyu Kartiko. Menyelesaikan DIV di Sekolah Tinggi Akuntansi Negara dan meraih gelar Magister Ekonomi di Fakultas Ekonomi Universitas Indonesia. Memiliki kompetensi pada bidang *programming* (java, scala, R, C, C++), *database*, *accounting* (privat, publik), *public sector economics*.