

Panduan Lengkap Analisis Statistika Menggunakan R Commander

Mohammad Rosidi

2020-01-07

Hidup terlalu singkat untuk sekedar dinikmati. Menulis adalah salah satu cara untuk memaknai kehidupan dan memberi inspirasi bagi sekitar.

(Mohammad Rosidi)

Contents

List of Tables	vii
List of Figures	ix
Pengantar	xi
1 Pengenalan R dan R Commander	1
1.1 Apa Itu R dan Rcmdr?	1
1.2 Memasang R dan R Commander	2
1.2.1 Memasang R pada Windows	2
1.2.2 Memasang R pada Mac OS X	4
1.2.3 Memasang R pada Linux dan Unix	6
1.2.4 Memasang R Commander	6
1.3 Menjalankan R dan R Commander	9
1.4 Tampilan R dan R Commander	9
1.4.1 Antarmuka R	9
1.4.2 Antarmuka R Commander	14
2 Manajemen Data	25
2.1 Operator Operasi Pada R	25
2.1.1 Operator Aritmatika	26
2.1.2 Operator Perbandingan	29
2.1.3 Operator Logika	31

2.2	Tipe dan Struktur Data	32
2.3	Konsep <i>Tidy Data</i>	35
2.4	Input Data pada <code>R Commander</code>	36
2.5	Membaca Data dari File Eksternal	38
2.5.1	Membaca Data dari Sumber <i>Plain Text</i> .	39
2.5.2	Membaca Data dari Sumber <i>Spreadsheet</i> dan Lainnya	42
2.6	Membaca Data dari Paket	44
2.6.1	Memuat Paket	44
2.6.2	Memuat Dataset Pada Paket	44
2.6.3	Membaca Data	45
2.7	Meyimpan dan Memuat Data	47
2.7.1	Menyimpan Dataset	47
2.7.2	Memuat Data	49
2.8	Memodifikasi Variabel pada Data	50
2.8.1	Variabel <i>Recoding</i>	50
2.8.2	Komputasi Variabel Baru	54
2.8.3	Menambahkan Variabel Nomor Observasi Pada Dataset	56
2.8.4	Standardisasi Variabel	58
2.8.5	Merubah Variabel <i>Numeric</i> Menjadi <i>Factor</i>	58
2.8.6	Melakukan <i>Binning</i> pada Variabel <i>Numeric</i>	60
2.8.7	Merubah Urutan <i>Factor Levels</i>	63
2.8.8	Melakukan <i>Drop</i> pada <i>Factor Levels</i> . . .	65
2.8.9	Merubah Nama Variabel pada Data	65
2.8.10	Menghapus Variabel	67
2.9	Memanipulasi Dataset	69

<i>Contents</i>	v
2.9.1 Melakukan Subset Dataset	70
2.9.2 Agregat Variabel pada Dataset	72
2.9.3 Melakukan <i>Drop</i> Observasi pada Dataset .	75
2.9.4 Melakukan <i>Drop</i> Observasi dengan <i>Missing Value</i>	76
2.9.5 Mengelompokkan Variabel Menjadi Variabel <i>Factor</i> dan Nilai	78
2.9.6 Menggabungkan Dua Dataset	80
2.9.7 Modifikasi Lainnya	82
2.10 Latihan	83
3 Ringkasan dan Visualisasi Data	87
4 Uji Statistik Sederhana	89
5 Linier dan <i>Generalized Linear</i> Model	91
6 Distribusi Probabilitas dan Simulasi	93



List of Tables

2.1	Operator Aritmatika R	26
2.2	Hierarki prioritas operasi operator aritmatika.	28
2.3	Operator Relasi R	29
2.4	Hierarki prioritas operasi dengan penambahan operator perbandingan.	31
2.5	Operator logika R	31
2.6	Tipe data R	33
2.7	Struktur data R	35
2.9	Penjelasan variabel dataset mtcars.	38
2.8	Sepuluh Observasi pertama dataset mtcars	40
2.10	Penjelasan terkait item pada jendela Read Text Data From File, Clipboard, or URL.	42
2.11	Penjelasan item jendela Export active data set.	49
2.12	Ketentuan recode pada submenu Recode Variables.	51
2.13	Penjelasan item jendela Recode Variables.	53
2.14	Penjelasan item convert numeric variables to factors.	60
2.15	Penjelasan item jendela convert numeric variables to factors.	62
2.16	Penjelasan item jendela reorder factor levels.	64
2.17	Penjelasan item jendela subset data set.	72
2.18	Fungsi statistika deskriptif bawaan pada R.	72

2.19 Penjelasan item jendela aggregate observations.	73
2.20 Penjelasan item jendela remove rows from active data set.	76
2.21 Penjelasan item jendela remove missing data.	77
2.22 Penjelasan item jendela stack variables.	79
2.23 Penjelasan item jendela merge data sets.	82
2.24 Hasil pemantauan parameter DO dan BOD outlet IPAL X	83
2.25 Hasil pemantauan parameter DO,BOD,dan COD 10 IPAL Kota X	84
2.26 Hasil pemantauan parameter TSS 12 IPAL Kota X	85

List of Figures

1.1	Logo R	2
1.2	Tampilan situs CRAN.	3
1.3	Tampilan halaman situs R untuk mengunduh R for windows.	3
1.4	Tampilan tahapan kunci proses instalasi R for Windows.	5
1.5	Tampilan halaman situs XQuartz.	6
1.6	Tampilan awal proses instalasi R for Mac OS X (Sumber: Fox, 2017).	7
1.7	Tampilan proses instalasi menggunakan menu Packages.	8
1.8	Tampilan proses pemnuatan paket Rcmdr.	10
1.9	Tampilan antar muka jendela R for windows.	11
1.10	Tampilan antar muka jendela R Commander.	14
2.1	Visualisasi 3 rule tidy data (Sumber: Grolemund dan Wickham, 2017).	36
2.2	Visualisasi tahapan input dataset pada R Commander.	37
2.3	Visualisasi tahapan membaca file plain text dari sistem lokal.	40
2.4	Visualisasi tahapan membaca file plain text dari internet URL.	41

2.5	Tampilan jendela konfigurasi import data berbagai format file.	43
2.6	Tampilan langkah menampilkan seluruh dataset dari paket.	45
2.7	Tampilan langkah membaca dataset pada paket. .	46
2.8	Tampilan tahapan menyimpan data ke dalam format RData.	47
2.9	Tampilan tahapan menyimpan data ke dalam format csv.	48
2.10	Tampilan tahapan memuat data dalam format RData.	50
2.11	Tampilan tahapan recoding variabel.	53
2.12	Tampilan tahapan komputasi variabel baru. . . .	57
2.13	Tampilan tahapan standardisasi variabel.	59
2.14	Tampilan tahapan merubah variabel numeric menjadi factor.	60
2.15	Tampilan tahapan binning variabel numeric. . . .	62
2.16	Tampilan tahapan merubah urutan factor level. .	64
2.17	Tampilan tahapan drop factor level.	66
2.18	Tampilan tahapan merubah nama variabel.	68
2.19	Tampilan tahapan menghapus variabel.	69
2.20	Tampilan tahapan melakukan subset pada dataset.	71
2.21	Tampilan tahapan agregasi variabel.	74
2.22	Tampilan tahapan melakukan drop observasi. .	76
2.23	Tampilan tahapan melakukan drop observasi dengan missing value.	77
2.24	Tampilan tahapan mengelompokkan variabel menjadi variabel factor dan nilai.	79
2.25	Tampilan tahapan menggabungkan dua buah dataset.	81

Pengantar



1

Pengenalan R dan R Commander

Pada Chapter 1 pembaca akan belajar mengenai perangkat lunak R, Rcmdr, dan cara instalasi keduanya. Pembaca juga akan belajar tampilan antar muka dan fitur apa saja yang disediakan pada Rcmdr. Detail terkait fitur-fitur Rcmdr yang digunakan dalam melakukan analisis statistika akan dijelaskan pada Chapter-Chapter selanjutnya.

1.1 Apa Itu R dan Rcmdr?

R Merupakan bahasa yang digunakan dalam komputasi **statistik** yang pertama kali dikembangkan oleh **Ross Ihaka** dan **Robert Gentleman** di University of Auckland New Zealand yang merupakan akronim dari nama depan kedua pembuatnya. Sebelum R dikenal ada s yang dikembangkan oleh **John Chambers** dan rekan-rekan dari **Bell Laboratories** yang memiliki fungsi yang sama untuk komputasi statistik. Hal yang membedakan antara keduanya adalah R merupakan sistem komputasi yang bersifat gratis. Logo R dapat dilihat pada Gambar 1.1.

Rcmdr atau R Commander (Fox, 2005) merupakan paket yang menyediakan tampilan antar muka (GUI) yang memungkinkan pengguna untuk berinteraksi dengan R melalui *point-and-click*. Keuntungan cara berinteraksi semacam itu membuat pengguna dapat lebih memfokuskan diri pada metode statistik yang digunakannya dibandingkan melakukan penulisan sintaks program untuk memperoleh hasil perhitungan statistik yang sama. Selain itu, sintaks yang digunakan program untuk melakukan perhitungan di-



Gambar 1.1: Logo R.

tampilkan ke dalam editor yang tersedia pada jendela `Rcmdr` secara otomatis. Hal tersebut dapat mempermudah analis dalam mempelajari kode apa saja yang perlu diketikkan untuk menghasilkan hasil perhitungan statistik yang analis perlukan.

1.2 Memasang R dan R Commander

Pada Chapter 1.2 pembaca akan belajar bagaimana cara memasang R dan R Commander. Pembaca juga akan belajar bagaimana cara menjalankan R Commander dari R.

Untuk memperoleh R pembaca dapat melakukan unduh pada situs CRAN¹. Tampilan situs untuk mengunduh R ditampilkan pada Gambar 1.2.

1.2.1 Memasang R pada Windows

R for Windows dapat diperoleh melalui tautan CRAN². Berdasarkan halaman situs diketahui bahwa saat ini versi R yang tersedia adalah versi R 3.6.2. Tampilan situs tersebut dapat dilihat pada Gambar 1.3.

Tahapan instalasi:

¹<https://cran.r-project.org/>

²<https://cran.r-project.org/>

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, Windows and Mac users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2019-12-12, Dark and Stormy Night) [R-3.6.2.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

Gambar 1.2: Tampilan situs CRAN.

R-3.6.2 for Windows (32/64 bit)

[Download R 3.6.2 for Windows](#) (83 megabytes, 32/64 bit)

[Installation and other instructions](#)
[New features in this version](#)

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server. You will need a version of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

Frequently asked questions

- [Does R run under my version of Windows?](#)
- [How do I update packages in my previous version of R?](#)
- [Should I run 32-bit or 64-bit R?](#)

Please see the [R FAQ](#) for general information about R and the [R Windows FAQ](#) for Windows-specific information.

Other builds

- Patches to this release are incorporated in the [r-patched snapshot build](#).
- A build of the development version (which will eventually become the next major release of R) is available in the [r-devel snapshot build](#).
- [Previous releases](#)

Note to webmasters: A stable link which will redirect to the current Windows binary release is [<CRAN MIRROR>/bin/windows/base/release.htm](#).

Last change: 2019-12-12

Gambar 1.3: Tampilan halaman situs R untuk mengunduh R for windows.

1. Double click R installer yang telah di unduh sehingga muncul jendela instalasi.
2. Pilih bahasa yang akan digunakan.
3. Pembaca hanya perlu menekan tombol *next* pada jedela yang muncul untuk memasang R dengan konfigurasi *default*.
4. Setelah proses instalasi selesai pembaca dapat menekan tombol *finish*.

Untuk lebih lengkapnya, pembaca dapat menyaksikan video yang dibuat oleh Xperimental Learning³.

1.2.2 Memasang R pada Mac OS X

Sebelum melakukan proses instalasi pastikan Mac OS X yang pembaca miliki *up to date*. Untuk mengetahuinya pembaca dapat menjalakan *Software Update* dari menu yang berada pada pojok kiri atas layar. Hal tersebut penting sebab R mengasumsikan bahwa sistem yang kita miliki telah *up to date*.

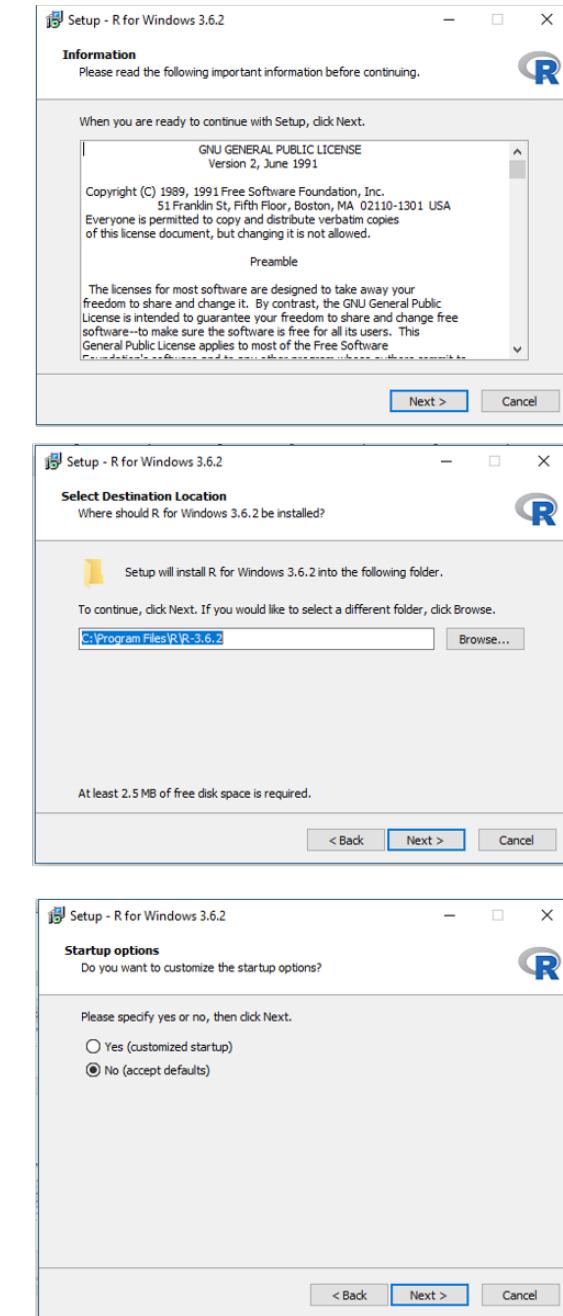
R for Mac OS X dapat diperoleh melalui tautan XQuartz⁴. Selanjutnya pembaca tinggal mengunduh file yang memiliki format file xQuartz-x.y.zz.dmg. Tampilan situs tersebut dapat dilihat pada Gambar 1.5.

Tahapan instalasi:

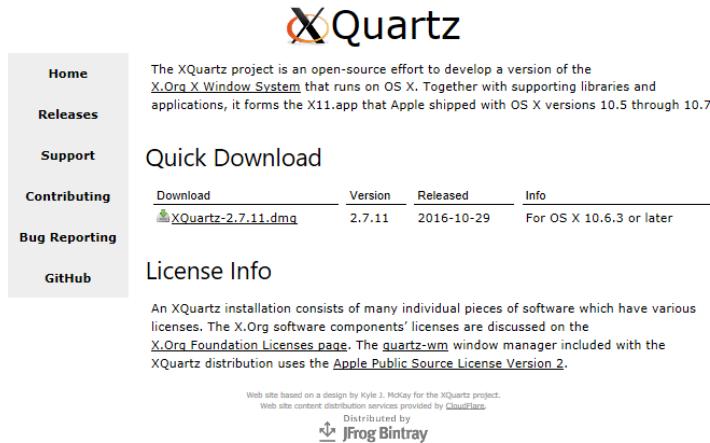
1. Mengunduh *disk image file* xQuartz-x.y.zz.dmg dimana x.y.zz merupakan versi dari xQuartz.
2. Double click pada file tersebut. Jika pembaca menemukan file xQuartz.pkg, Double click pada file tersebut. Lakukan klik pada tombol *continue* untuk konfigurasi instalasi *default*.
3. Setelah proses instalasi *log out* dari sesi komputer pembaca sekarang atau lakukan *reboot/restart* dan masuk kembali menggunakan akun Mac OS X pembaca.

³<https://www.youtube.com/watch?v=9-RrkJQQYqY>

⁴<https://www.xquartz.org/>



Gambar 1.4: Tampilan tahapan kunci proses instalasi R for Windows.



Gambar 1.5: Tampilan halaman situs XQuartz.

Alternatif metode instalasi lainnya dapat pembaca baca pada artikel yang ditulis oleh Galarnyk(2017)⁵.

1.2.3 Memasang R pada Linux dan Unix

Berdasarkan situs CRAN, R tersedia pada sejumlah distribusi linux, seperti: Debian, RedHat, SUSE, dan Ubuntu. Jika pembaca memiliki sistem linux atau unix yang tidak *compatible* berdasarkan daftar distribusi linux yang tersedia, pembaca perlu melakukan kompilasi R dari kode sumber. Prosedur untuk melakukan hal tersebut dijelaskan pada halaman R FAQ⁶.

1.2.4 Memasang R Commander

Terdapat dua buah cara untuk memasang R Commander pada R, yaitu: melalui sintaks pada R *Console* dan melalui menu Packages. Untuk melakukan instalasi menggunakan R *Console*, jalankan sintaks berikut:

⁵<https://medium.com/@GalarnykMichael/install-r-and-rstudio-on-mac-e911606ce4f4>

⁶<https://cran.r-project.org/doc/FAQ/RFAQ.html>



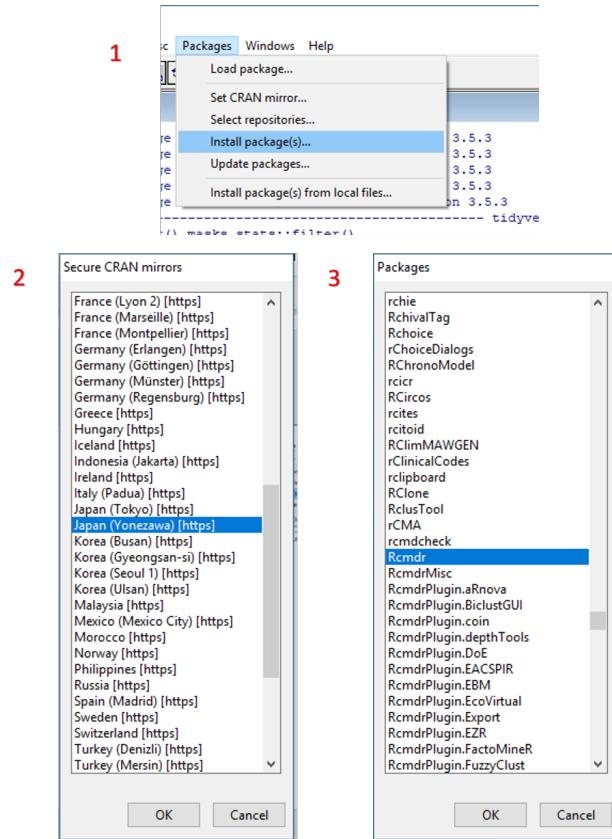
Gambar 1.6: Tampilan awal proses instalasi R for Mac OS X (Sumber: Fox, 2017).

```
install.packages("Rcmdr")
```

Program selanjutnya akan memasang R Commander dan paket-paket lain yang menjadi *dependency*-nya.

Untuk instalasi melalui menu Packages, langkah-langkah yang perlu dilakukan adalah sebagai berikut:

1. Jalankan R dengan cara *double click* pada *shortcut* R yang ada pada desktop atau melalui menu sistem operasi yang pembaca miliki.
2. Klik pada Packages/Install package(s)....



Gambar 1.7: Tampilan proses instalasi menggunakan menu Packages.

3. Pilih CRAN *mirror* yang tersedia, klik OK. Pembaca dapat pula memilih CRAN *mirror* dari Indonesia. Jika gagal pembaca dapat mencobanya dengan menggunakan CRAN *mirror* dari negara lain.
4. Pilih paket **Rcmdr**, klik OK.
5. Saat pertama kali proses instalasi akan muncul dialog yang berisi apakah pembaca setuju jika R membuat sebuah *directory* yang berisi paket **Rcmdr**.
6. R akan mengunduh paket **Rcmdr** dan *dependency*-nya.

1.3 Menjalankan R dan R Commander

Untuk menjalankan R Commander pada R terdapat dua metode, yaitu: menggunakan fungsi `library()` dan melalui menu `Packages`. Penggunaan fungsi `library()` untuk memuat `Rcmdr` ditampilkan pada sintaks berikut:

```
library(Rcmdr)
```

Untuk memuat R Commander menggunakan menu `Packages` dapat dilakukan melalui langkah-langkah berikut:

1. Jalankan R dengan cara *double click* pada *shortcut* R yang ada pada desktop atau melalui menu sistem operasi yang pembaca miliki.
2. Klik pada `Packages/Install package(s)....`
3. Setelah muncul jendela daftar paket yang telah terpasang, klik pada paket `Rcmdr`.
4. R akan memuat paket `Rcmdr` dan *dependency*-nya.

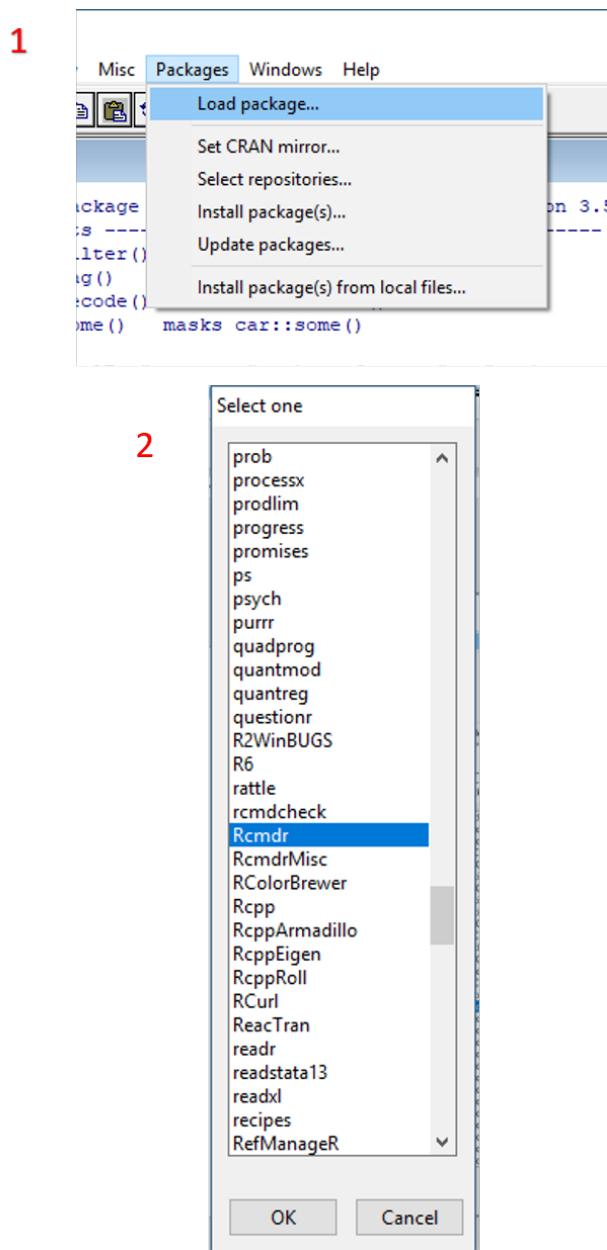
1.4 Tampilan R dan R Commander

1.4.1 Antarmuka R

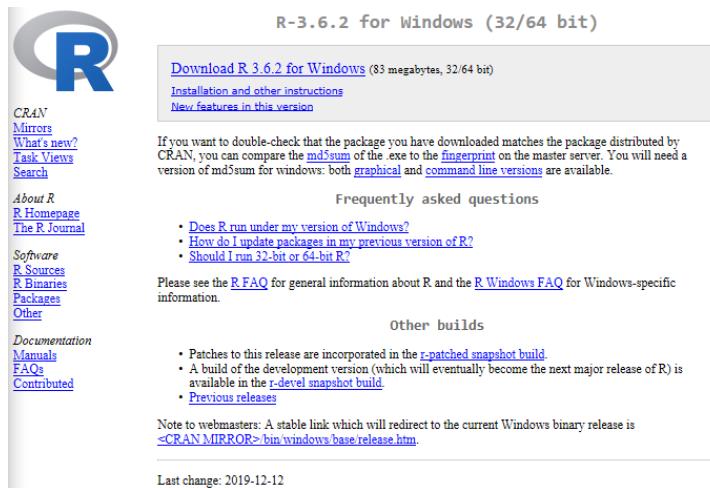
Tampilan R saat pertama kali dilankan dapat dilihat seperti pada Gambar 1.9. Pada Gambar tersebut, jendela R terbagi menjadi 4 bagian, antara lain:

Bagian 1: baris menu yang terdiri atas menu:

- **File** : menu yang berkaitan dengan cara membuat dan menyimpan *script* R, memuat dan menyimpan *history* kerja, merubah direktori kerja, mencetak dan menyimpan file, dan keluar dari R.



Gambar 1.8: Tampilan proses pemuatan paket Rcmdr.



Gambar 1.9: Tampilan antar muka jendela R for windows.

- **Edit** : menu yang berkaitan dengan sejumlah perintah untuk melakukan *editing* seperti *copy* dan *paste*, memilih atau meng*highlight* sejumlah sintaks, membersihkan *R console*, membuka *data editor*, dan perintah untuk membuka jendela pengaturan GUI R.
- **View** : menu yang memungkinkan pengguna melakukan pengaturan tampilan jendela R seperti menampilkan panel *toolbar* atau menampilkan panel status.
- **Misc** : menu yang menampilkan sejumlah submenu perintah dan pilihan untuk menghentikan proses komputasi, menampilkan dan menghapus seluruh objek yang telah dibuat, dan menampilkan paket-paket yang aktif.
- **Packages** : menu yang digunakan untuk mengatur paket-paket yang ada di R, seperti: memuat paket, memasang dan menghapus paket, memperbaharui paket, dan melakukan pengaturan repository berupa lokasi paket-paket yang akan dipasang.
- **Windows** : menu yang menampilkan fungsi pengaturan jendela-jendela yang ada pada **Bagian 4**.
- **Help** : menu bantuan R.

Secara lengkap fitur yang tersedia pada baris menu ditampilkan pada diagram pohon menu berikut:

```
File
|- Source R code...
|- New script...
|- Open script...
|- Display file(s)...
|- Load Workspace...
|- Save Workspace...
|- Load History...
|- Save History...
|- Change dir...
|- Print...
|- Save to File...
|- Exit

Edit
|- Copy
|- Paste
|- Paste commands only
|- Copy and Paste
|- Select all
|- Clear console
|- Data editor
|- GUI preferences

View
|- Toolbar
|- Statusbar

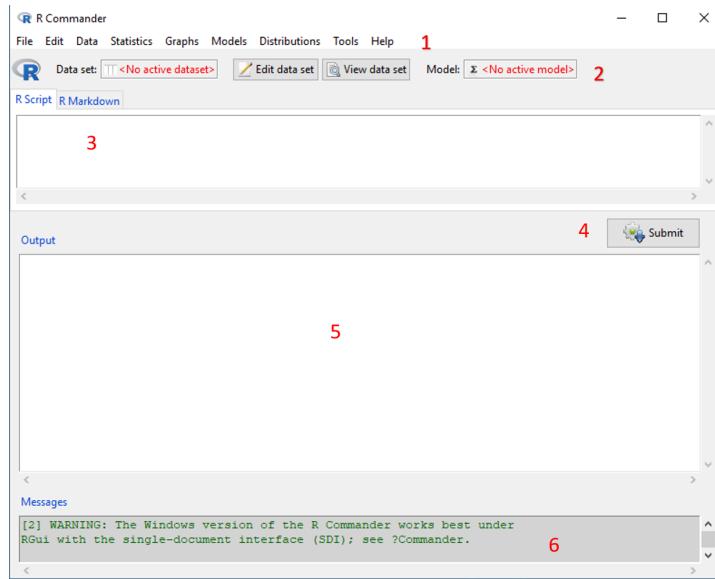
Misc
|- Stop current computation
|- Stop all computations
|- Buffered output
|- Word completion
|- Filename completion
|- List objects
|- Remove all objects
|- List search path

Packages
|- Load package...
|- Set CRAN mirror...
```

```
|-- Select repositories...
|-- Install package(s)...
|-- Update package(s)...
|-- Install package(s) from local files...
Misc
|-- Cascade
|-- Tile Horizontally
|-- Tile Vertically
|-- Arrange Icons
Help
|-- Console
|-- FAQ on R
|-- FAQ on R for Windows
|-- Manuals (in PDF)
|   |-- An Introduction to R
|   |-- R reference
|   |-- R Data Import/Export
|   |-- R Language Definition
|   |-- Writing R Extensions
|   |-- R Internals
|   |-- R Installation and Administration
|   |-- Sweave User
|-- R functions (text)...
|-- Html help
|-- Search help...
|-- search.r-project.org...
|-- Apropos...
|-- R Project home page
|-- CRAN home page
|-- About
```

Bagian 2 : panel yang berisikan *toolbar* untuk membuka *R script*, memuat dan menyimpan ruang kerja, perintah *copy and paste*, menghentikan komputasi, dan mencetak hasil perhitungan pada jendela *console* dan editor.

Bagian 3 : jendela *console*.



Gambar 1.10: Tampilan antar muka jendela R Commander.

Bagian 4 : ruang kosong lokasi jendela baru seperti *console* dan grafik dimuat.

1.4.2 Antarmuka R Commander

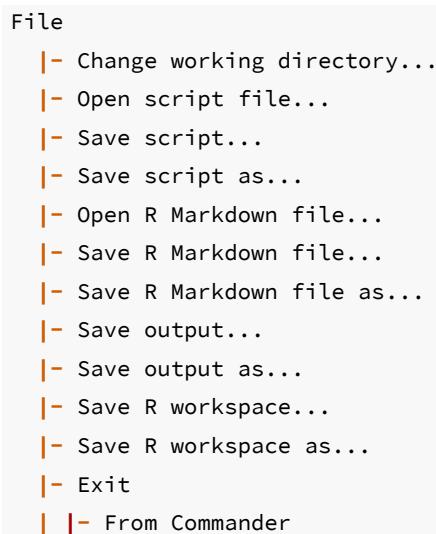
Tampilan R Commander saat pertama kali dijalankan dapat dilihat seperti pada Gambar 1.10. Pada Gambar tersebut, jendela R Commander terbagi menjadi 5 bagian, antara lain:

Bagian 1 : baris menu yang terdiri atas menu:

- **File** : menu untuk memuat dan menyimpan file *script*, menyimpan *output* dan ruang kerja R; dan keluar dari R Commander atau dari R dan R Commander.
- **Edit** : menu (*cut, copy, paste*, dll.) Untuk mengedit teks di berbagai panel dan tab. Mengklik kanan di salah satu panel atau tab ini juga memunculkan menu edit sesuai dengan tab yang digunakan.
- **Data** : menu yang terdiri atas submenu untuk membaca dan mengolah data.

- **Statistics** : menu yang terdiri atas berbagai submenu untuk melakukan berbagai analisis statistik.
- **Graphs** : menu yang terdiri atas submenu yang berisikan berbagai metode visualisasi data.
- **Models** : menu dan submenu yang digunakan untuk memperoleh ringkasan numerik, rentang keyakinan, uji hipotesis, diagnosa, dan grafik dari model statistik, dan menambahkan hasil diagnosa pada dataset seperti menambahkan residu atau error model pada dataset.
- **Distribution** : menu yang terdiri atas submenu yang digunakan untuk memperoleh probabilitas kumulatif, densitas probabilitas, kuantil, dan grafik dari distribusi statistika standar.
- **Tools** : menu yang digunakan untuk mengakses paket R (contoh: memuat dataset dari paket lainnya, memuat paket untuk menambahkan metode analisis statistik, dll), untuk memuat paket Rcmdr plug-in, mengatur sebagian besar opsi pada R Commander, dan untuk memasang *optional auxilary software*.
- **Help** : menu bantuan yang berguna untuk memperoleh informasi terkait R Commander dan paket terkait lainnya.

Secara lengkap menu dan submenu pada R Commander ditampilkan pada diagram pohon berikut:





```
| |- Subset active data set...
| |- Sort active data set...
| |- Aggregate variables in active data set...
| |- Remove row(s) from active data set...
| |- Stack variables in active data set...
| |- Remove cases with missing data...
| |- Save active data set...
| |- Export active data set...
|- Manage variables in active data set
| |- Recode variable...
| |- Compute new variable...
| |- Add observation numbers to data set
| |- Standardize variables...
| |- Convert numeric variables to factors...
| |- Bin numeric variable...
| |- Reorder factor levels...
| |- Drop unused factor levels...
| |- Define contrasts for a factor...
| |- Rename variables...
| |- Delete variables from data set ...
Statistics
|- Summaries
| |- Active data set
| |- Numerical summaries...
| |- Frequency distributions...
| |- Count missing observations
| |- Table of statistics...
| |- Correlation matrix...
| |- Correlation test...
| |- Test of normality...
| |- Transform toward normality...
|- Contingency tables
| |- Two-way table...
| |- Multi-way table...
| |- Enter and analyze two-way table...
|- Means
| |- Single-sample t-test...
```

- | | - Independent samples t-test...
- | | - Paired t-test...
- | | - One-way ANOVA...
- | | - Multi-way ANOVA...
- | - Proportions
 - | | - Single-sample proportion test...
 - | | - Two-sample proportions test...
- | - Variances
 - | | - Two-variances F-test...
 - | | - Bartlett's test...
 - | | - Levene's test...
- | - Nonparametric tests
 - | | - Two-sample Wilcoxon test...
 - | | - Single-sample Wilcoxon test...
 - | | - Paired-samples Wilcoxon test...
 - | | - Kruskal-Wallis test...
 - | | - Friedman rank-sum test...
- | - Dimensional analysis
 - | | - Scale reliability...
 - | | - Principal-components analysis...
 - | | - Factor analysis...
 - | | - Confirmatory factor analysis...
 - | | - Cluster analysis
 - | | | - k-means cluster analysis...
 - | | | - Hierarchical cluster analysis...
 - | | | - Summarize hierarchical clustering...
 - | | | - Add hierarchical clustering to data set...
- | - Fit models
 - | | - Linear regression...
 - | | - Linear model...
 - | | - Generalized linear model...
 - | | - Multinomial logit model...
 - | | - Ordinal regression model...
- Graphs
 - | - Color palette...
 - | - Index plot...
 - | - Dot plot...

- | - Histogram...
- | - Plot discrete numeric variable.....
- | - Density estimate...
- | - Stem-and-leaf display...
- | - Boxplot...
- | - Quantile-comparison plot...
- | - Symmetry boxplot...
- | - Scatterplot...
- | - Scatterplot matrix...
- | - Line graph...
- | - XY conditioning plot...
- | - Plot of means...
- | - Strip chart...
- | - Bar graph...
- | - Pie chart...
- | - 3D graph
 - | | - 3D scatterplot...
 - | | - Identify observations with mouse
 - | | - Save graph to file
- | - Save graph to file
 - | | - as bitmap...
 - | | - as PDF/Postscript/EPS...
 - | | - 3D RGL graph...

Models

- | - Select active model...
- | - Summarize model
- | - Compare model coefficients...
- | - Add observation statistics to data...
- | - Akaike Information **Criterion** (AIC)
- | - Bayesian Information **Criterion** (BIC)
- | - Stepwise model selection...
- | - Subset model selection...
- | - Confidence intervals.....
- | - Bootstrap confidence intervals...
- | - Delta method confidence interval...
- | - Hypothesis tests
 - | | - ANOVA table...

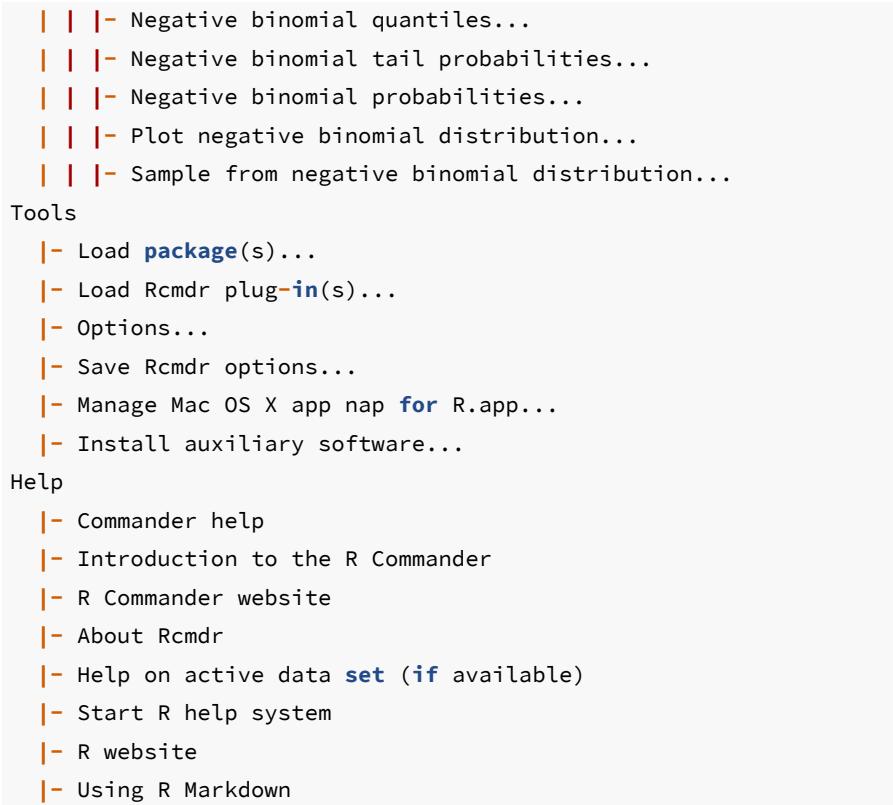
- | | - Compare two models...
- | | - Linear hypothesis...
- | - Numerical diagnostics
 - | | - Variance-inflation factors
 - | | - Breusch-Pagan test **for** heteroscedasticity...
 - | | - Durbin-Watson test **for** autocorrelation...
 - | | - RESET test **for** nonlinearity...
 - | | - Bonferroni outlier test
 - | | - Response transformation...
- | - Graphs
 - | | - Basic diagnostic plots
 - | | - Residual quantile-comparison plot...
 - | | - Component+residual plots...
 - | | - Added-variable plots...
 - | | - Influence plot...
 - | | - Effect plots...
 - | | - Predictor effect plots...

Distributions

- | - Set random number generator seed...
- | - Continuous distributions
 - | | - Normal distribution
 - | | | - Normal quantiles...
 - | | | - Normal probabilities...
 - | | | - Plot normal distribution...
 - | | | - Sample from normal distribution...
 - | | - t distribution
 - | | | - t quantiles...
 - | | | - t probabilities...
 - | | | - Plot t distribution...
 - | | | - Sample from t distribution...
 - | | - Chi-squared distribution
 - | | | - Chi-squared quantiles...
 - | | | - Chi-squared probabilities...
 - | | | - Plot chi-squared distribution...
 - | | | - Sample from chi-squared distribution...
 - | | - F distribution
 - | | | - F quantiles...

```
| | | |- F probabilities...
| | | |- Plot F distribution...
| | | |- Sample from F distribution...
| | |- Exponential distribution
| | | |- Exponential quantiles...
| | | |- Exponential probabilities...
| | | |- Plot exponential distribution...
| | | |- Sample from exponential distribution...
| | |- Uniform distribution
| | | |- Uniform quantiles...
| | | |- Uniform probabilities...
| | | |- Plot uniform distribution...
| | | |- Sample from uniform distribution...
| | |- Beta distribution
| | | |- Beta quantiles...
| | | |- Beta probabilities...
| | | |- Plot beta distribution...
| | | |- Sample from beta distribution...
| | |- Cauchy distribution
| | | |- Cauchy quantiles...
| | | |- Cauchy probabilities...
| | | |- Plot Cauchy distribution...
| | | |- Sample from Cauchy distribution...
| | |- Logistic distribution
| | | |- Logistic quantiles...
| | | |- Logistic probabilities...
| | | |- Plot logistic distribution...
| | | |- Sample from logistic distribution...
| |- Lognormal distribution
| | | |- Lognormal quantiles...
| | | |- Lognormal probabilities...
| | | |- Plot lognormal distribution...
| | | |- Sample from lognormal distribution...
| |- Gamma distribution
| | | |- Gamma quantiles...
| | | |- Gamma probabilities...
| | | |- Plot gamma distribution...
```

```
| | | |- Sample from gamma distribution...
| | | |- Weibull distribution
| | | |- Weibull quantiles...
| | | |- Weibull probabilities...
| | | |- Plot Weibull distribution...
| | | |- Sample from Weibull distribution...
| | | |- Gumbel distribution
| | | |- Gumbel quantiles...
| | | |- Gumbel probabilities...
| | | |- Plot Gumbel distribution...
| | | |- Sample from Gumbel distribution...
|- Discrete distributions
| | |- Binomial distribution
| | | |- Binomial quantiles...
| | | |- Binomial tail probabilities...
| | | |- Binomial probabilities...
| | | |- Plot binomial distribution...
| | | |- Sample from binomial distribution...
| | | |- Poisson distribution
| | | |- Poisson quantiles...
| | | |- Poisson tail probabilities...
| | | |- Poisson probabilities...
| | | |- Plot Poisson distribution...
| | | |- Sample from Poisson distribution...
| | | |- Geometric distribution
| | | |- Geometric quantiles...
| | | |- Geometric tail probabilities...
| | | |- Geometric probabilities...
| | | |- Plot geometric distribution...
| | | |- Sample from geometric distribution...
| | |- Hypergeometric distribution
| | | |- Hypergeometric quantiles...
| | | |- Hypergeometric tail probabilities...
| | | |- Hypergeometric probabilities...
| | | |- Plot hypergeometric distribution...
| | | |- Sample from hypergeometric distribution...
| | |- Negative binomial distribution
```



Item pada menu akan tidak aktif (tulisan berwarna abu-abu) apabila tidak ada sesuai dengan konteks tertentu, misal: tidak ada dataset aktif maka sebagian besar submenu **statistics** akan tidak aktif. Contoh lainnya adalah tidak adanya data kategori pada dataset aktif maka submenu tabel kontingensi tidak akan aktif.

Bagian 2: baris *toolbar* berupa tombol yang dapat digunakan untuk berinteraksi dengan objek data atau model yang ada. Tombol-tombol tersebut terdiri atas:

- **Data set** : menampilkan nama dataset yang aktif dan memilih dataset yang akan diaktifkan.
- **Edit data set** : digunakan untuk melakukan proses *editing pada dataset* seperti: merubah nilai baris dan kolom, merubah nama kolom, merubah nama baris, dan menambahkan atau menghapus observasi.

- **View data set** : melihat observasi pada dataset aktif.
- **Model** : menampilkan dan memilih model statistik yang telah dibuat.

Bagian 3: 2 buah tab lembar kerja yang terdiri atas:

- **R Script** : menampilkan *script* perintah yang digunakan untuk menghasilkan output. Pembaca dapat melakukan proses *editing* pada *script* tersebut dan menjalankannya kembali untuk menambah kompleksitas pada luaran yang dihasilkan.
- **R Markdown** : membuat dokumentasi dari analisis yang telah dilakukan.

Bagian 4: Tombol submit. Untuk menjalankan kembali *R Script* yang telah dibuat, pembaca dapat meng-*highlight script* atau sintaks yang hendak diperoleh kembali hasilnya dan tekan tombol *Submit*.

Bagian 5: *Output box*. Kotak ini berfungsi untuk menampilkan hasil perhitungan berdasarkan sintaks yang dimasukkan atau di-submit.

Bagian 6: *Messages box*. Menampilkan sejumlah pesan terkait operasi yang dilakukan. Pesan dapat berupa *error*, *warnings*, dan *note*.

2

Manajemen Data

Pada Chapter 2, penulis akan menjelaskan kepada pembaca bagaimana cara menyiapkan data sebelum dilakukan analisa pada R Commander. Adapun yang akan dijelaskan pada Chapter 2, antara lain:

- Operator operasi yang digunakan pada R,
 - Jenis dan struktur data yang ada pada R,
 - Konsep *tidy data*,
 - Input data pada R Commander,
 - Membaca data dari file eksternal,
 - Membaca data dari paket,
 - Menyimpan dan memuat data,
 - Memodifikasi Variabel pada data, dan
 - Memanipulasi Dataset.
-

2.1 Operator Operasi Pada R

Terdapat sejumlah operator operasi yang penting untuk pembaca ketahui, antara lain:

- Operator aritmatika,
- Operator perbandingan, dan
- Operator logika.

2.1.1 Operator Aritmatika

Proses perhitungan akan ditangani oleh fungsi khusus. R akan memahami urutannya secara benar. Kecuali kita secara eksplisit menetapkan yang lain. Sebagai contoh tuliskan dan jalankan sintaks berikut pada Console R (tekan enter) maupun R Commander (tekan tombol submit):

```
2+4*2
```

```
## [1] 10
```

Bandingkan dengan sintaks berikut:

```
(2+4)*2
```

```
## [1] 12
```

TIPS!: R dapat digunakan sebagai kalkulator

Berdasarkan kedua hasil tersebut dapat disimpulkan bahwa ketika kita tidak menetapkan urutan perhitungan menggunakan tanda kurung, R akan secara otomatis akan menghitung terlebih dahulu perkalian atau pembagian.

Operator aritmatika yang disediakan R disajikan pada Tabel 2.1:

Tabel 2.1: Operator Aritmatika R.

Simbol	Keterangan
+	<i>Addition</i> , untuk operasi penjumlahan
-	<i>Subtraction</i> , untuk operasi pengurangan
*	<i>Multiplication</i> , untuk operasi pembagian

Simbol	Keterangan
/	<i>Division</i> , untuk operasi pembagian
^	<i>Eksponentiation</i> , untuk operasi pemangkatan
%%	<i>Modulus</i> , Untuk mencari sisa pembagian
%/%	<i>Integer</i> , Untuk mencari bilangan bulat hasil pembagian saja dan tanpa sisa pembagian

Untuk lebih memahaminya berikut contoh sintaks penerapan operator tersebut.

```
# Addition
```

```
5+3
```

```
## [1] 8
```

```
# Subtraction
```

```
5-3
```

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

```
# Multiplication
```

```
5*3
```

```
## [1] 15
```

```
# Division
```

```
5/3
```

```
## [1] 1.667
```

```
# Eksponetiation
```

```
5^3
```

```
## [1] 125
```

```
# Modulus
5%%3
```

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

```
# Integer
5%/%3
```

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

Penggunaan operator aritmatika perlu mempertimbangkan hierarki prioritas operasinya. Pada contoh sebelumnya kita telah belajar bahwa operasi aritmatika akan dikerjakan terlebih dahulu dari yang ada di dalam tanda kurung lalu setelah itu akan diikuti oleh operasi lainnya. Secara lengkap, hierarki prioritas operasi aritmatika dirangkum pada Tabel 2.2:

Tabel 2.2: Hierarki prioritas operasi operator aritmatika.

Prioritas	Operator	Keterangan
1	+,-	unari (tanda +,-)
2	^	
3	*, /, %% , %/%	
4	+,-	binari

Berdasarkan Tabel 2.2, pembaca dapat memprediksi output dari operasi berikut:

```
-2+(3^2*2)/3
```

Operasi tersebut akan menghasilkan nilai 4 dengan urutan penggeraan sebagai berikut:

1. Pemberian tanda negatif pada angka 2

2. Operasi dalam tanda kurung dengan urutan eksponensi-asi (3^2) diikuti perkalian (9×2)
3. Operasi pembagian terhadap nilai dalam kurung dengan angka 3 ($18/3$)
4. Operasi penjumlahan ($-2+6$)

2.1.2 Operator Perbandingan

Operator relasi digunakan untuk membandingkan satu objek dengan objek lainnya. Operator yang disediakan R disajikan pada Tabel 2.3.

Tabel 2.3: Operator Relasi R.

Simbol	Keterangan	Deskripsi
<code>==</code>	sama dengan	bernilai TRUE jika kedua objek bernilai sama
<code>!=</code>	tidak sama dengan	bernilai TRUE jika kedua objek tidak bernilai sama
<code>></code>	lebih besar dari	bernilai TRUE jika nilai objek kanan lebih besar dari nilai objek kiri
<code><</code>	lebih kecil dari	bernilai TRUE jika nilai objek kanan lebih kecil dari nilai objek kiri
<code>>=</code>	lebih besar sama dengan	bernilai TRUE jika nilai objek kanan lebih besar atau sama dengan dari nilai objek kiri
<code><=</code>	lebih kecil sama dengan	bernilai TRUE jika nilai objek kanan lebih kecil atau sama dengan dari nilai objek kiri

Berikut adalah penerapan operator pada tabel tersebut:

```
x <- 34
y <- 35

# Operator >
x > y
```

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

```
# Operator <  
x < y
```

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

```
# operator ==  
x == y
```

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

```
# Operator >=  
x >= y
```

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

```
# Operator <=  
x <= y
```

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

```
# Operator !=  
x != y
```

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

Operator perbandingan memiliki hierarki prioritas yang lebih rendah dibandingkan dengan operator aritmatika. Pembaharuan Tabel 2.2 dilakukan dengan menambahkan operator perbandingan ditampilkan pada Tabel 2.4.

Tabel 2.4: Hierarki prioritas operasi dengan penambahan operator perbandingan.

Prioritas	Operator	Keterangan
1	+,-	unari (tanda +,-)
2	^	
3	*, /, %% , %/ %	
4	+,-	binari
5	<, <=, >, >=	
6	==, !=	

2.1.3 Operator Logika

Operator logika hanya berlaku pada vektor dengan tipe logical, numeric, atau complex. Semua angka bernilai 1 akan dianggap bernilai logika `TRUE`. Operator logika yang disediakan R dapat dilihat pada Tabel 2.5.

Tabel 2.5: Operator logika R.

Simbol	Keterangan
<code>&&</code>	Operator logika AND
<code> </code>	Operator logika OR
<code>!</code>	Opearator logika NOT
<code>&</code>	Operator logika AND element wise
<code> </code>	Operator logika OR element wise

Penerapannya terdapat pada sintaks berikut:

```
v <- c(TRUE, TRUE, FALSE)
t <- c(FALSE, FALSE, FALSE)

# Operator &&
print(v&&t)

## [1] FALSE
```

```
# Operator ||
print(v||t)

## [1] TRUE

# Operator !
print(!v)

## [1] FALSE FALSE  TRUE

# operator &
print(v&t)

## [1] FALSE FALSE FALSE

# Operator |
print(v|t)

## [1] TRUE  TRUE FALSE
```

operator & dan | akan mengecek logika tiap elemen pada vektor secara berpesongan (sesuai urutan dari kiri ke kanan). Operator %% dan || hanya mengecek dari kiri ke kanan pada observasi pertama. Misal saat menggunakan && jika observasi pertama `TRUE` maka observasi pertama pada vektor lainnya akan dicek, namun jika observasi pertama `FALSE` maka proses akan segera dihentikan dan menghasilkan `FALSE`.

2.2 Tipe dan Struktur Data

Data pada R dapat dikelompokan berdasarkan beberapa tipe. Tipe data pada R disajikan pada Tabel 2.6.

Tabel 2.6: Tipe data R.

Tipe Data	Contoh	Keterangan
Logical	TRUE, FALSE	Nilai Boolean
Numeric	12.3, 5, 999	Segala jenis angka
Integer	23L, 97L, 3L	Bilangan integer (bilangan bulat)
Complex	2i, 3i, 9i	Bilangan kompleks
Character	'a', "b", "123"	Karakter dan string
Factor	1, 0, "Merah"	Dapat berupa numerik atau string (namun pada proses akan terbaca sebagai angka)
Raw	Identik dengan "hello"	Segala jenis data yang disimpan sebagai raw bytes

Sintaks berikut adalah contoh dari tipe data pada R. Untuk mengetahui tipa data suatu objek kita dapat menggunakan perintah `class()`

```
# Logical
apel <- TRUE
class(apel)
```

```
## [1] "logical"
```

```
# Numeric
x <- 2.3
class(x)
```

```
## [1] "numeric"
```

```
# Integer
y <- 2L
class(y)

## [1] "integer"

# Compleks
z <- 5+2i
class(z)

## [1] "complex"

# string
w <- "saya"
class(w)

## [1] "character"

# Raw
xy <- charToRaw("hello world")
class(xy)

## [1] "raw"
```

Keenam jenis data tersebut disebut sebagai tipe data atomik. Hal ini disebabkan karena hanya dapat menangani satu tipe data saja. Misalnya hanya numeric atau hanya integer.

Selain menggunakan fungsi `class()`, kita dapat pula menggunakan fungsi `is.numeric()`, `is.character()`, `is.logical()`, dan sebagainya berdasarkan jenis data apa yang ingin kita cek. Berbeda dengan fungsi `class()`, output yang dihasilkan pada fungsi seperti `is.numeric()` adalah nilai Boolean sehingga fungsi ini hanya digunakan untuk mengecek apakah jenis data pada objek sama seperti yang kita pikirkan. Sebagai contoh disajikan pada sintaks berikut:

Struktur data diklasifikasikan berdasarkan dimensi data dan tipe data di dalamnya (homogen atau heterogen). Klasifikasi jenis data disajikan pada Tabel 2.7.

Tabel 2.7: Struktur data R.

Dimensi	Homogen	Heterogen
1d	Atomik vektor	List
2d	Matriks	Dataframe
nd	Array	

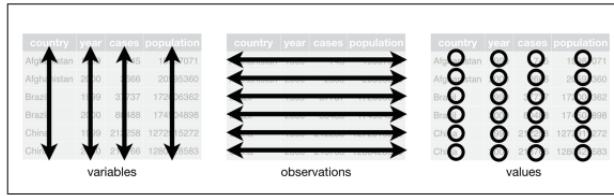
Berdasarkan Tabel tersebut dapat kita lihat bahwa objek terbagi atas dua buah struktur data yaitu homogen dan heterogen. Objek dengan struktur data homogen hanya dapat menyimpan satu tipe atau jenis data saja (numerik saja atau factor saja), sedangkan objek dengan struktur data heterogen akan dapat menyimpan berbagai jenis data.

2.3 Konsep *Tidy Data*

Sebelum memulai analisa terhadap data yang kita miliki, umumnya kita akan merapikan data yang akan kita gunakan. Tujuannya adalah agar data yang akan digunakan sudah siap untuk dilakukan analisa dengan software tertentu seperti R atau R Commander, dimana pada dataset perlu jelas antara variabel dan nilai (*value*), serta untuk mempermudah dalam memperoleh informasi pada data. Sebelum kita melakukan analisa di dataset tersebut, kita harus tahu terlebih dahulu apa saja syarat suatu dataset dikatakan rapi (*tidy*). Berikut adalah syaratnya:

- Setiap variabel harus memiliki kolomnya sendiri
- Setiap observasi harus memiliki barisnya sendiri
- Setiap nilai berada pada sel tersendiri

Ketiga syarat tersebut saling berhubungan sehingga jika salah satu syarat tersebut tidak terpenuhi, maka dataset belum bisa dikatakan *tidy*. Ketiga syarat tersebut dapat divisualisasikan melalui Gambar 2.1.



Gambar 2.1: Visualisasi 3 rule tidy data (Sumber: Grolemund dan Wickham, 2017).

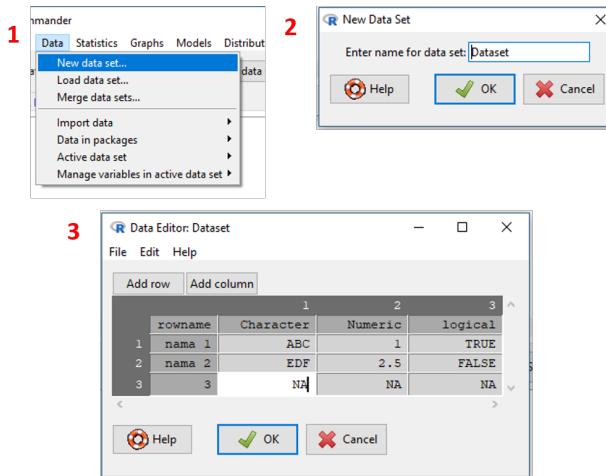
2.4 Input Data pada R Commander

Input data dapat dilakukan secara langsung pada **R Commander**. Input data secara langsung umumnya dilakukan jika jumlah data yang kita miliki relatif kecil. Untuk melakukanya jalakan tahapan berikut:

1. Pada menu, klik **Data/New data set....** Klik **ok**,
2. Pada jedela yang muncul, ketikkan nama dataset yang kita inginkan. Klik **ok**
3. Pada jendela **Data Editor:Nama_Dataset**, ketikkan data yang kita miliki.
4. Untuk menambah baris klik tombol **Add row**, sedangkan untuk menambah kolom klik tombol **Add column**.
5. Untuk mengubah nama kolom, klik pada bagian nama kolom dan ketikkan nama kolom yang diinginkan.
6. Secara *default rowname* akan dinamai urutan observasi. Namun, kita dapat memberikan nama pada masing-masing kolom dengan cara meng-klik nama baris pada tiap barisnya.
7. Untuk mengecek dataset yang telah kita buat, klik **toolbar View data set**.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.2.

Terdapat beberapa hal yang perlu diperhatikan pada proses input data, antara lain:



Gambar 2.2: Visualisasi tahapan input dataset pada R Commander.

- Dalam pemberian *rownames*. Jika *rownames* mengandung spasi, input *rownames* disertai tanda *quote* (contoh:"Nama Baris"). Jika tidak ingin menggunakan spasi pada *rownames*, gunakan tanda titik atau koma sebagai pemisah kata (contoh:Nama.Baris atau Nama_Baris).
- Pastikan *rownames* bersifat unik (tidak ada duplikasi nama).
- Pembaca dapat menggunakan tombol pada *keyboard* untuk bernavigasi pada *data editor* atau gunakan klik kiri pada sel yang ingin dituju.
- Jenis data pada kolom akan secara otomatis ditentukan oleh seluruh data yang ada pada kolomnya. Jika data berupa angka, program secara otomatis mengkonversinya menjadi data *numeric*. Jika data berupa campuran angka atau karakter, secara otomatis program mengkonversinya menjadi *character*.
- Pembaca dapat memperluas area sel dengan cara menggeser sisi sel atau dengan cara memperluas melalui menggeser ujung jendela editor.
- Dataset yang telah dibuat dapat diedit kembali dengan cara klik *toolbar Edit data set*.

2.5 Membaca Data dari File Eksternal

Pada Chapter 2.5, pembaca akan mempelajari bagaimana cara melakukan import data dari berbagai sumber seperti *plain text*, *spreadsheets*, SPSS, STATA, SAS, dan Minitab. Sebelum melakukan hal tersebut terdapat beberapa hal yang perlu pembaca perhatikan, antara lain:

- Pastikan data berada dalam format *tidy data* (lihat Chapter 2.3), dan
- Pastikan *missing value* berada pada notasi yang konsisten.

Data yang berasal dari berbagai sumber akan memberikan format notasi *missing value* yang berbeda-beda, misalnya data yang berasal dari database akan memberikan notasi `NULL` terhadap *missing value*, sedangkan data yang berasal dari *spreadsheet* akan memberikan notasi *missing value* berdasarkan operasi yang dilakukan pada datanya (contoh: `#VALUE!` untuk hasil operasi 2 buah objek berbeda tipe datanya). `RCommander` tidak dapat menangani kondisi di mana pada satu kolom data terdapat lebih dari 1 notasi *missing value*. Untuk mengatasi hal tersebut, pembaca perlu menyerasikan notasi *missing value* pada data (contoh: mengubahnya menjadi notasi `NA` atau dikosongkan jika data bersumber dari *spreadsheet*).

Pada Chapter 2.5, pembaca akan diberikan contoh bagaimana melakukan import data yang disajikan pada Tabel 2.8.

Adapun penjelasan terkait Tabel 2.8 ditampilkan pada Tabel 2.9

Tabel 2.9: Penjelasan variabel dataset mtcars.

Variabel	Keterangan
<code>mpg</code>	Mil/(US) galon
<code>cyl</code>	Jumlah silinder
<code>disp</code>	<i>Displacement</i> (cu.in)

Variabel	Keterangan
hp	<i>Gross horsepower</i>
drat	Rasio gandar belakang
wt	Berat (1000 lb)
qsec	Watu tempuh 1/4 mil
vs	Mesin (0= <i>V-shape</i> , 1= <i>straight</i>)
am	Transmisi (0=otomatis, 1=manual)
gear	Jumlah <i>gear</i> depan
carb	Jumlah karburator

2.5.1 Membaca Data dari Sumber *Plain Text*

Terdapat 3 buah metode untuk membaca data dari *plain text*. Metode tersebut dibagi berdasarkan lokasi file *plain text* tersebut berada.

Membaca file yang berada pada sistem lokal

1. Pada menu `Data`, klik `Data/Import data/from text file,clipboard,or URL....`
2. Pada jendela yang muncul, isikan spesifikasi file (lihat Tabel 2.10) dan nama objek dataset yang diinginkan. Pada bagian `Location of Data File` pilih `Local file system`. Klik `OK`.
3. Pada jendela `Windows Explorer` yang muncul, pilih file yang hendak dibaca. Klik `Open`.
4. Untuk melihat dataset yang berhasil dibuat, klik pada `toolbar View data set`.

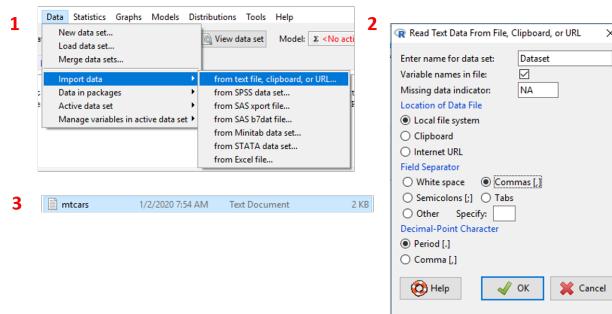
Visualisasi tahapan tersebut ditampilkan pada Gambar 2.3.

Membaca file yang berada pada *clipboard*

1. *Highlight* tabel dataset yang pembaca miliki dan *copy* dataset tersebut. Dataset tersebut selanjutnya akan tersimpan pada *clipboard*.
2. Pada menu `Data`, klik `Data/Import data/from text file,clipboard,or URL....`

Tabel 2.8: Sepuluh Observasi pertama dataset mtcars

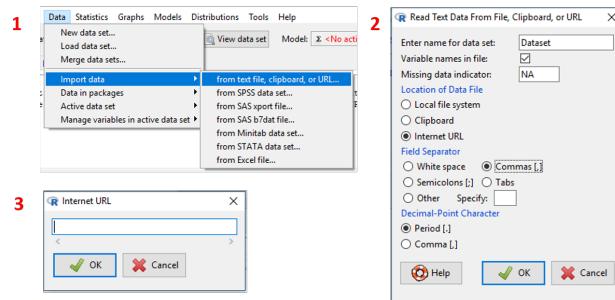
mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4

**Gambar 2.3:** Visualisasi tahapan membaca file plain text dari sistem lokal.

3. Pada jendela yang muncul, isikan spesifikasi file (lihat Tabel 2.10) dan nama objek dataset yang diinginkan. Pada bagian Location of Data File pilih clipboard . Klik ok.
4. Dataset akan secara otomatis dibuat oleh program dengan mengambil data yang tersimpan pada clipboard.
5. Untuk melihat dataset yang berhasil dibuat, klik pada toolbar View data set.

Membaca file yang berada pada URL

1. Copy halaman URL lokasi dataset berada.



Gambar 2.4: Visualisasi tahapan membaca file plain text dari internet URL.

2. Pada menu Data, klik Data/Import data/from text file,clipboard,or URL....
3. Pada jendela yang muncul, isikan spesifikasi file (lihat Tabel 2.10) dan nama objek dataset yang diinginkan. Pada bagian Location of Data File pilih Internet URL. Klik OK.
4. Pada jendela yang muncul tempelkan (*pasting*) halaman URL yang telah di *copy* sebelumnya.
5. Untuk melihat dataset yang berhasil dibuat, klik pada toolbar View data set.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.4.

Sintaks yang muncul pada R Console saat proses telah dilakukan adalah sebagai berikut:

```
# sistem lokal
Dataset <- read.table("D:/mtcars.txt", header=TRUE,
                      sep="", na.strings="NA", dec=".",
                      strip.white=TRUE)

# clipboard
Dataset <- read.table("clipboard", header=TRUE,
                      sep="", na.strings="NA",
                      dec=". ", strip.white=TRUE)

# URL
Dataset <- read.table("www.abcd.com/mtcars.txt",
```

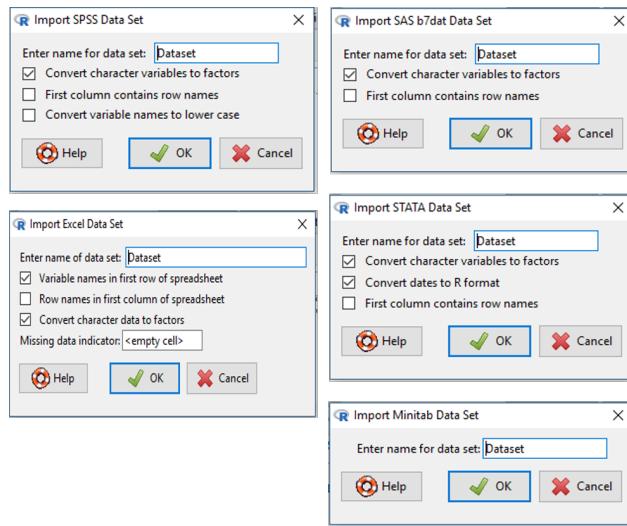
```
header=TRUE, sep="", na.strings="NA",
dec=".," , strip.white=TRUE)
```

Tabel 2.10: Penjelasan terkait item pada jendela Read Text Data From File, Clipboard, or URL.

No. Item	Jenis Input	Keterangan
1. Enter name for data set	<i>text</i> <i>input</i>	Input nama dataset yang diinginkan sebagai output
2. Variable name in file	<i>Check box</i>	Jika di centang, program membaca baris pertama tabel sebagai nama kolom
3. Missing value indikator	<i>text</i> <i>input</i>	Karakter yang mengidikasikan missing value dalam file (misal: <i>White space</i> , NA, NaN, dll)
4. Location of Data File	<i>radio</i> <i>button</i>	Lokasi file yang akan dibaca berada
5. Field Separator	<i>radio</i> <i>button</i>	Pemisah antar kolom data yang digunakan
6. Decimal-Point Character	<i>radio</i> <i>button</i>	Karakter yang digunakan sebagai penunjuk <i>decimal-point</i>

2.5.2 Membaca Data dari Sumber *Spreadsheet* dan Lainnya

Format data lain yang dapat dibaca oleh R Commander adalah `xlsx` (Excel), `.dta` (STATA), `.sav` (SPSS), `.sas7bdat` dan `.xport` (SAS), serta `.mtp` (minitab). Cara membaca data dengan format tersebut cukup sederhana dilakukan pada R Commander. Pembaca hanya perlu menuju menu `Data/Import data` dan memilih sumber data yang ingin dibaca. Pada jendela yang muncul (kecuali format `.xport`) pembaca diminta untuk melakukan sejumlah konfigurasi seperti apakah *rownames* terletak pada kolom pertama, apakah perlu mengubah jenis data karakter menjadi faktor, dll. Pada kon-



Gambar 2.5: Tampilan jendela konfigurasi import data berbagai format file.

disi dimana pembaca diminta untuk mengkonversi karakter menjadi faktor, penulis menyarankan untuk tidak melakukannya saat awal membaca data sebab akan menyulitkan pada saat melakukan analisis data selanjutnya. Konversi karakter menjadi faktor dilakukan pada sejumlah variabel yang memang ingin diubah menjadi faktor (bisa numerik atau karakter). Tampilan jendela konfigurasi awal saat membaca data ditampilkan pada Gambar 2.5.

Contoh sintaks yang akan muncul saat proses tersebut selesai adalah sebagai berikut:

```
# .xport
Datasets <- read.xport("D:/mtcars.xport")

# .sasb7dat
Dataset <- readSAS("D:/mtcars.sas7bdat",
                     stringsAsFactors=TRUE, rownames=FALSE)

# .dta
Dataset <- readStata("D:/mtcars.dta", convert.dates=TRUE,
                      stringsAsFactors=TRUE, rownames=FALSE)
```

2.6 Membaca Data dari Paket

2.6.1 Memuat Paket

Jika pembaca ingin mengakses dataset dari paket yang pembaca inginkan, pembaca dapat menjalankan perintah berikut pada R Console:

```
library(nama_paket)
```

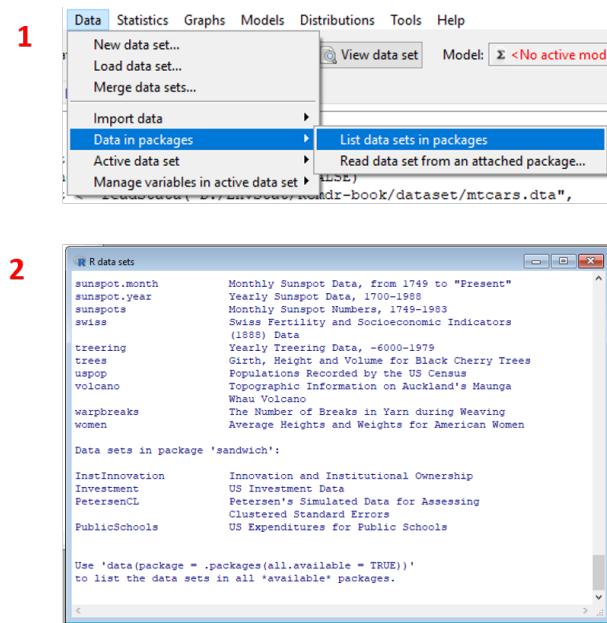
Jika paket tersebut belum terpasang, jalankan perintah berikut:

```
install.packages("nama_paket")
```

Jika pembaca kesulitan menggunakan cara tersebut, pembaca dapat menggunakan metode yang sama dengan cara memasang paket `Rcmdr` (lihat Chapter 1.2.4) dan memuat paket `Rcmdr` (lihat Chapter 1.3).

2.6.2 Memuat Dataset Pada Paket

Untuk mengecek dataset apa saja yang tersedia paket yang telah aktif, lakukan langkah-langkah berikut:



Gambar 2.6: Tampilan langkah menampilkan seluruh dataset dari paket.

1. Pada menu Data, klik Data/Data in packages/List data set in packages.
2. Jendela R data sets yang memberikan daftar seluruh dataset yang tersedia pada paket yang telah dimuat akan muncul.

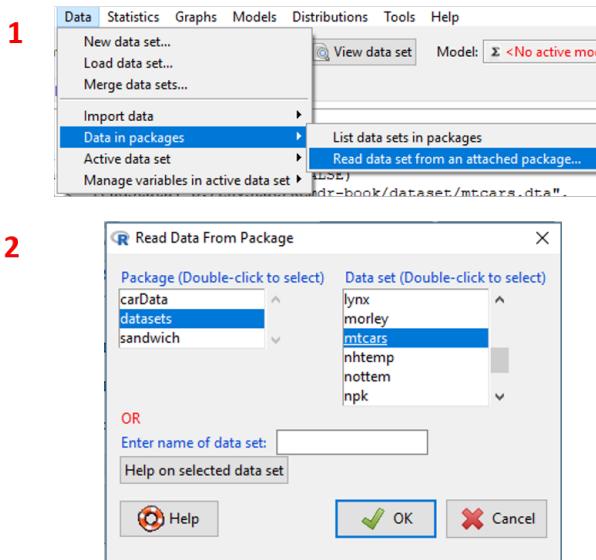
Pada proses tersebut, R Script akan memunculkan sebuah sintaks berikut:

```
data()
```

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.6.

2.6.3 Membaca Data

Untuk membaca dataset dari paket, jalankan langkah-langkah berikut:



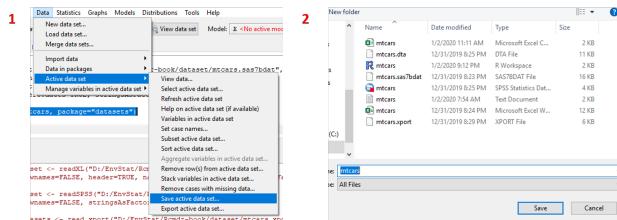
Gambar 2.7: Tampilan langkah membaca dataset pada paket.

1. Pada menu Data, klik Data/Data in packages/Read data set from an attached package....
2. Pada kotak package, *double click* paket yang pembaca ingin lihat datasetnya. Daftar dataset selanjutnya akan muncul pada kotak Data set.
3. Pilih dataset yang pembaca ingin baca. Pembaca dapat merubah nama dataset yang akan dibaca melalui kotak Enter name of data set. Klik OK.
4. Untuk melihat dataset yang telah dimuat, klik toolbar View data set.

Pada contoh berikut, penulis mencoba memuat dataset `mtcars` dari paket `datasets`. Sintaks yang muncul pada R Script dan kotak Output adalah sebagai berikut:

```
data(mtcars, package="datasets")
```

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.7.



Gambar 2.8: Tampilan tahapan menyimpan data ke dalam format RData.

2.7 Meyimpan dan Memuat Data

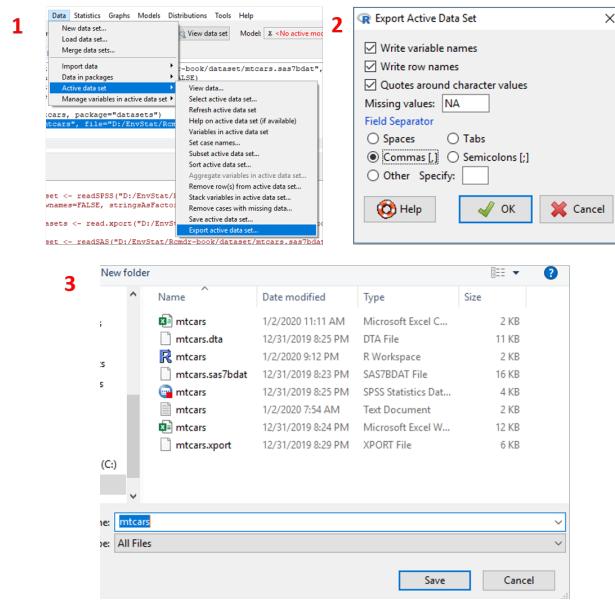
2.7.1 Menyimpan Dataset

Data pada **R Commander** dapat disimpan ke dalam format **.RData**. Penyimpanan dalam format tersebut akan mempermudah pembaca untuk memperoleh data tersebut saat akan dibaca kembali dan pembaca tidak perlu mengulangi kembali proses membaca data pada bagian sebelumnya. Untuk dapat menyimpan data menggunakan format **.RData** yang telah berhasil dibaca pada **R Commander**, pembaca dapat melakukan langkah-langkah berikut:

1. Pada menu **Data**, klik **Data/Active data set/Save active data set....**
2. Pada jendela **windows Explorer** yang muncul, navigasikan ke lokasi atau folder di mana data tersebut akan disimpan. Beri nama data tersebut sesuai dengan nama yang diinginkan. Klik **Save**.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.8.

Pada **R Script** dan kotak **Output** akan muncul sintaks berikut yang menandakan bahwa data yang ada pada **R Commander** telah disimpan pada folder yang telah penulis inginkan.



Gambar 2.9: Tampilan tahapan menyimpan data ke dalam format csv.

```
save("mtcars", file="D:/EnvStat/Rcmdr-book/dataset/mtcars.RData")
```

Selain menyimpan data ke dalam format .RData, R Commander juga dapat menyimpan data ke dalam format .csv. Untuk melakukannya jalankan langkah berikut:

1. Pada menu Data, klik Data/Active data set/Export active data set....
2. Pada jendela yang muncul, spesifikasikan format data yang akan disimpan (lihat Tabel 2.11).
3. Pada jendela Windows Explorer yang muncul, navigasikan ke lokasi atau folder di mana data tersebut akan disimpan. Beri nama data tersebut sesuai dengan nama yang diinginkan. Klik Save.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.9.

Pada R Script dan kotak output akan muncul sintaks berikut yang

menandakan bahwa data yang ada pada R Commander telah disimpan pada folder yang telah penulis inginkan.

```
write.table(mtcars, "D:/mtcars2.csv", sep=",",
            col.names=TRUE, row.names=TRUE,
            quote=TRUE, na="NA")
```

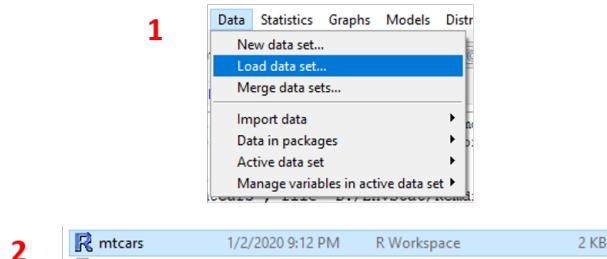
Tabel 2.11: Penjelasan item jendela Export active data set.

No. Item	Jenis Im- put	Keterangan
1. Write variable names	check box	pilihan apakah nama variabel akan disertakan ke dalam file csv
2. Write row names	check box	pilihan apakah nama baris disertakan ke dalam file csv
3. Quotes around character values	check box	pilihan apakah tipe data karakter diberi tanda petik
4. Missing values	text input	simbol atau karakter <i>missing value</i> yang digunakan pada file csv
5. Field Separator	radio button	pemisah antar kolom data yang digunakan

2.7.2 Memuat Data

Data yang telah disimpan ke dalam format .RData dapat langsung dimuat ke dalam R Commander tanpa perlu menspesifikasikan kembali format data yang hendak dibaca. Untuk melakukannya jalankan langkah berikut:

1. Pada menu Data, klik Data/Load data set....
2. Pada jendela Windows Explorer yang muncul, navigasikan ke lokasi atau folder di mana data tersebut berada. Pilih data yang akan dimuat. Klik Open.



Gambar 2.10: Tampilan tahapan memuat data dalam format RData.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.10.

2.8 Memodifikasi Variabel pada Data

Menu `Data/Manage variables in active data set` merupakan menu yang dibuat untuk memodifikasi variabel dan membuat variabel baru pada dataset. Pada Chapter 2.8, penulis akan menjelaskan cara menggunakan submenu `Recode variables` untuk merubah level pada tipe data `factor` dan membuat variabel baru dari variabel dengan tipe data `numeric`, penulis juga akan menjelaskan cara menggunakan submenu `Reorder Factor Levels` untuk mengubah urutan `factor` yang semula berdasarkan urutan alfabet menjadi sesuai dengan yang kita inginkan, serta informasi lain terkait penggunaan submenu pada menu `Data/Manage variables in active data set`.

2.8.1 Variabel *Recoding*

Terdapat dua kegunaan submenu `Recode Variables`, yaitu: membuat `factor` baru dengan cara mentransformasi variabel `numeric` menjadi `factor`, serta merubah urutan level suatu `factor`.

Ketentuan `recode` pada `Recode Variables` secara umum memiliki for-

mula `*nilai-lama=nilai-baru*`, dimana `nilai-baru` (nilai awal variabel tercatat) dispesifikasi sesuai dengan Tabel 2.12. Berikut adalah sejumlah informasi yang perlu pembaca perhatikan untuk memahami penggunaan submenu `Recode Variables`:

- Jika nilai variabel lama yang tercatat sama sekali ketentuan `recode` yang kita tetapkan, nilai tersebut hanya dibawa menuju `Recode Variables`. Sebagai contoh, jika aturan "`sangat setuju"="setuju`" ditetapkan, tapi nilai lama untuk faktor "setuju" tidak dicatat, maka kedua nilai lama (faktor lama) tersebut digabung menjadi satu sehingga ("`sangat setuju`", "`setuju`")=`"setuju"`.
- Jika nilai variabel lama yang dicatat memenuhi lebih dari satu ketetapan `recode`, maka ketentuan pertama yang berlaku diterapkan. Sebagai contoh, jika terdapat variabel `pendapatan` dengan ketentuan `recode lo:2.500.000="rendah"` dan `2.500.000:7.500.000="sedang"`, maka sebuah observasi dengan variabel `pendapatan` bernilai `2.500.000` akan dikategorikan sebagai "`rendah`".
- Seperti yang telah dijabarkan pada poin sebelumnya, karakter spesial `lo` digunakan untuk menyatakan nilai variabel `numeric` terkecil, sedangkan `hi` digunakan untuk variabel `numeric` dengan nilai tertinggi.
- Pada setiap ketentuan `recode` pastikan diakhiri dengan nilai `else` yang menunjukkan nilai lain diluar ketentuan `recode` sebelumnya (termasuk *missing value*).

Tabel 2.12: Ketentuan recode pada submenu Recode Variables.

No.	Nilai Lama	Contoh Ketentuan <i>Recode</i>
1.	nilai individual(<i>a</i>)	<code>99=NA; NA="missing"; "sangat setuju"="setuju"</code>
2.	sebuah set nilai (<i>a,b,...,k</i>)	<code>1,3,5,...,k="ganjil"; "sangat setuju","cukup setuju"="setuju"</code>
3.	rentang numerik (<i>a:b</i>)	<code>1901:2000="abad 21"; lo:2.000.000="rendah"; 10.000.000:hi="tinggi"</code>

No.	Nilai Lama	Contoh Ketentuan <i>Recode</i>
4.	lainnya (<i>else</i>)	<i>else</i> ="lainnya"

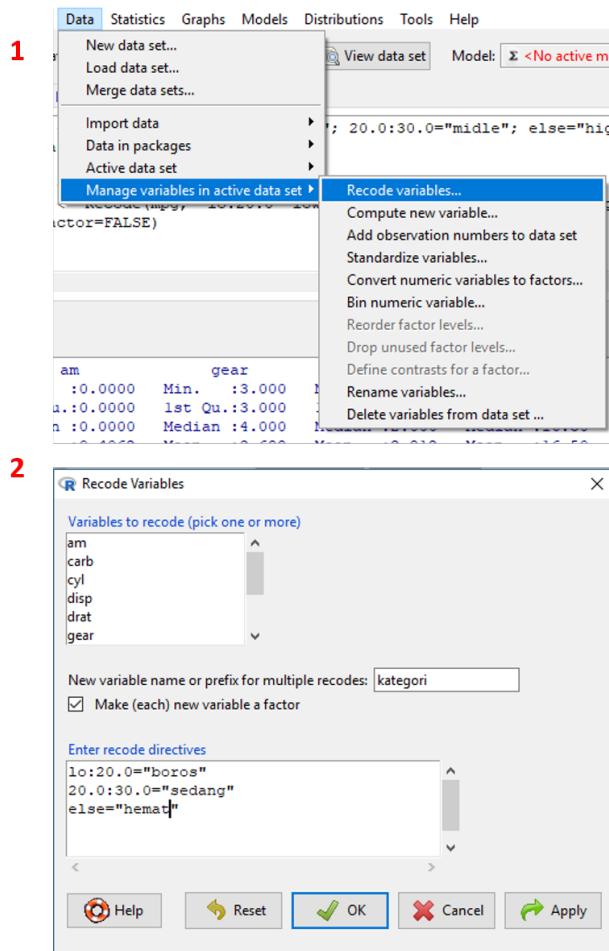
Pada dataset `mtcars` (lihat Tabel 2.8), misalkan kita ingin menambahkan variabel baru berupa `factor` yang menyatakan klasifikasi kendaraan berdasarkan tingkat penggunaan bahan bakar per mil (`mpg`). Untuk kendaraan dengan ketentuan `mpg lo:20="boros"`, `20:30="sedang"`, dan `else="hemat"`. Berikut adalah langkah-langkah untuk melakukannya:

1. Pada menu `Data`, klik `Data/Manage variables in active data set/Recode variables...`
2. Pada jendela yang muncul pilih variabel yang ingin dilakukan *recoding*, tentukan ketentuan *recodingnya*, dan nama variabel baru yang dihasilkan. Penjelasan terkait jendela tersebut ditampilkan pada Tabel 2.13.
3. Spesifikasikan apakah tipe data variabel tersebut adalah `factor` atau bukan dengan mencentang *checkbox*. Klik `ok`.
4. Untuk melihat variabel baru yang telah terbentuk, klik `View data set`.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.11.

Proses tersebut akan memunculkan sintaks pada `R Script` sebagai berikut:

```
mtcars <- within(mtcars, {kategori <- Recode(mpg,
  `lo:20.0="boros"; 20.0:30.0="sedang"; else="hemat"; `,
  as.factor=TRUE)
})
```



Gambar 2.11: Tampilan tahapan recoding variabel.

Tabel 2.13: Penjelasan item jendela Recode Variables.

No.	Jenis Input	Keterangan	
1.	Variables to recode	<i>select box</i>	daftar nama variabel yang akan di <i>recode</i>
2.	New variable name or..	<i>text input</i>	nama baru untuk variabel yang dibuat

No.	Item	Jenis Input	Keterangan
3.	Make (each) variable..	<i>check box</i>	pilihan apakah variabel baru adalah faktor
4.	Enter recode directives	<i>text input</i>	kotak untuk memasukkan ketentuan <i>recode</i>

2.8.2 Komputasi Variabel Baru

Pada saat analisis data, variabel yang kita miliki terkadang tidak cukup untuk menjelaskan suatu fenomena. Namun, kita dapat membentuk sebuah variabel baru yang dapat membantu menjelaskan fenomena tersebut. Pembentukan variabel baru dapat berupa transformasi sebuah variabel atau pembentukan variabel berdasarkan beberapa formulasi beberapa variabel lain.

Dalam melakukan transformasi variabel, kita dapat memanfaatkan sejumlah operator operasi yang telah penulis jelaskan pada Chapter 2.1.1. Selain operator tersebut, terdapat sejumlah fungsi operasi aritmatika yang tersedia pada R. Fungsi-fungsi tersebut, antara lain:

1. Logaritma dan eksponensial

Untuk contoh fungsi logaritmik dan eksponensial jalankan sintaks berikut:

```
log2(8) # logaritma basis 2 untuk 8
```

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

```
log10(8) # logaritma basis 10 untuk 8
```

```
## [1] 0.9031
```

```
exp(8) # eksponensial 8
```

```
## [1] 2981
```

2. Fungsi trigonometri

fungsi trigonometri yang ditampilkan seperti sin,cos, tan, dll.

```
cos(x) # cos x
sin(x) # Sin x
tan(x) # Tan x
acos(x) # arc-cos x
asin(x) # arc-sin x
atan(x) #arc-tan x
```

**PENTING!!!: x dalam fungsi trigonometri memiliki satuan radian

Berikut adalah salah satu contoh penggunaannya:

```
cos(pi)
```

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

3. Fungsi matematik lainnya

Fungsi lainnya yang dapat digunakan adalah fungsi absolut, akar kuadrat, dll. Berikut adalah contoh sintaks penggunaan fungsi absolut dan akar kuadrat.

```
abs(-2) # nilai absolut -2
```

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

```
sqrt(4) # akar kuadrat 4  
## [1] 2
```

Untuk memahami permasalahan terkait komputasi variabel baru, kita akan membuat variabel baru pada dataset `mtcars` yang telah dijelaskan pada Tabel 2.8. Variabel baru yang akan kita buat adalah variabel rasio antara jarak tempuh per satuan bahan bakar (`mpg`) terhadap berat kendaraan (`wt`) dan kita namai variabel baru tersebut `rwt`. Variabel baru ini dapat menjadi alternatif lain dalam menjelaskan efisiensi suatu mobil yang ditandai dengan rasio antara jarak tempuh terhadap bobot dan konsumsi bahan bakarnya. Berikut adalah tahapan untuk melakukannya:

1. Pada menu `Data`, klik `Data/Manage variables in active data set/Compute new variable`.
2. Pada jendela yang muncul, ketikkan formula pembentuk variabel baru pada kotak `Expression to compute`.
3. Untuk memasukkan nama variabel ke dalam formula, pembaca dapat mengetikkan nama variabel secara manual atau melakukan *double click* nama variabel yang tersedia pada kotak `Current variables`.
4. Ketikkan nama variabel baru pada kotak `New variable name`. Klik `OK`.
5. Untuk mengecek variabel yang telah terbentuk, klik `View data set`.

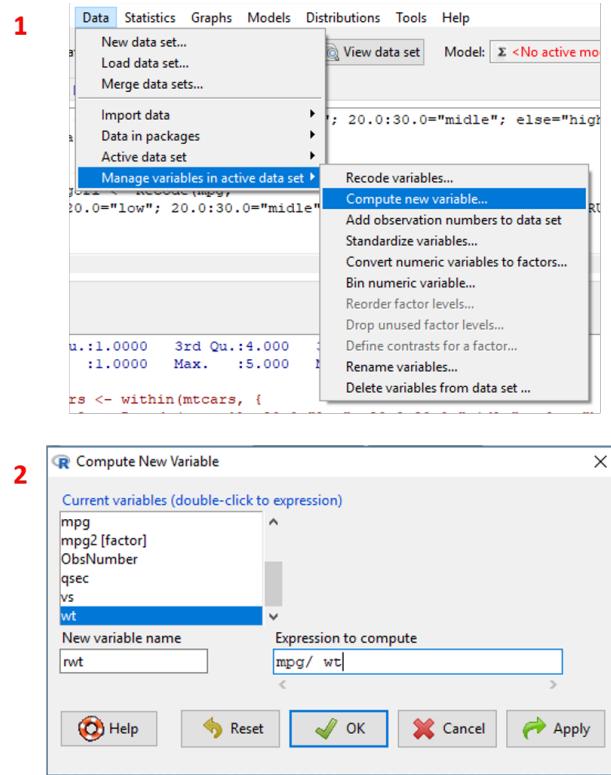
Visualisasi tahapan tersebut ditampilkan pada Gambar 2.12.

Berdasarkan tahapan tersebut, sintaks yang terbentuk pada `R Script` adalah sebagai berikut:

```
mtcars$rwt <- with(mtcars, mpg / wt)
```

2.8.3 Menambahkan Variabel Nomor Observasi Pada Dataset

Untuk menambahkan variabel nomor observasi pada dataset jalankan langkah-langkah berikut:



Gambar 2.12: Tampilan tahapan komputasi variabel baru.

1. Pada menu Data, klik Data/Manage variables in active data set/Add observation numbers to data set.
2. Variabel ObsNumber berupa nomor observasi akan secara otomatis ditambahkan pada akhir kolom dataset.
3. Untuk mengecek variabel baru tersebut, klik toolbar View data set.

Pada akhir tahapan, sintaks berikut tercetak pada R Script:

```
dataset$ObsNumber <- 1:"number of observation"
```

2.8.4 Standardisasi Variabel

Standardisasi variabel bertujuan untuk mentransformasi variabel sehingga variabel tersebut memiliki nilai rata-rata 0 dan simpangan baku 1. Tahapan melakukan standardisasi variabel pada R Commander adalah sebagai berikut:

1. Pada menu **Data**, klik **Data/Manage variables in active data set/Standardize variables....**
2. Pada jendela yang muncul, pilih variabel yang akan di standardisasi. Pembaca dapat memilih lebih dari satu variabel dengan cara menekan tombol **ctrl+klik** (pada Windows) saat memilih variabel. Klik **ok**.
3. Untuk mengecek variabel yang telah distandardisasi, klik **toolbar View data set**.

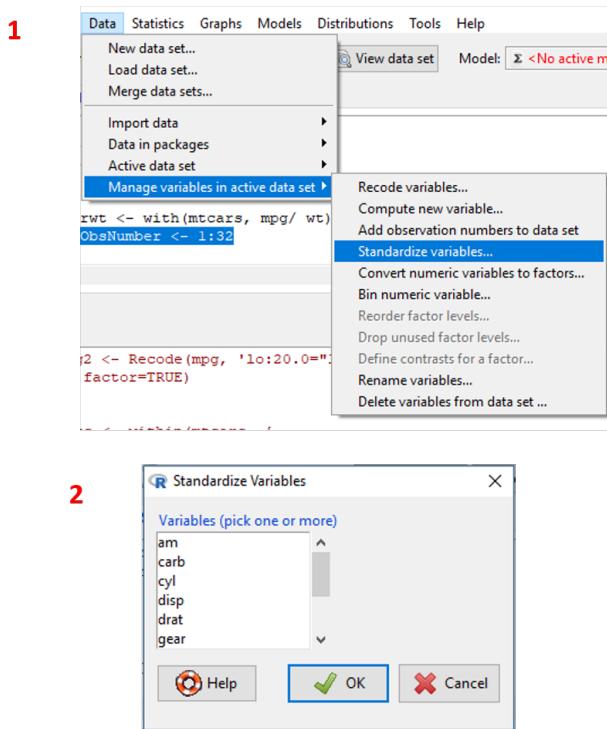
Visualisasi tahapan tersebut ditampilkan pada Gambar 2.13.

Sintaks yang digunakan dalam tahapan tersebut adalah sebagai berikut:

```
dataset <- local({  
  .Z <- scale(dataset[,c("namavariabel")])  
  within(dataset, {  
    Z.namavariabel <- .Z[,nomor_kolom_variabel]  
  })  
})
```

2.8.5 Merubah Variabel *Numeric* Menjadi *Factor*

Pada Chapter 2.8.1, kita telah belajar bagaimana cara melakukan *recoding* variabel dan membentuk *factor*. Pada Chapter 2.8.5, kita akan membahas cara merubah variabel *numeric* menjadi *factor* tanpa perlu melakukan proses *recoding*. Pada Chapter ini, kita hanya perlu mesuplai variabel *numeric* yang akan diubah menjadi *factor*. Kita juga dapat menambahkan label pada *factor* tersebut. Untuk melakukannya jalankan langkah-langkah berikut:

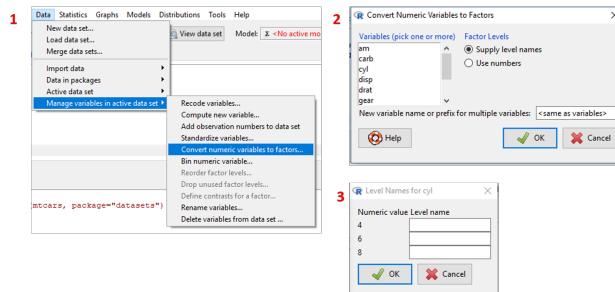


Gambar 2.13: Tampilan tahapan standardisasi variabel.

1. Pada menu Data, klik Data/Manage variables in active data set/Convert numeric variables to factors.
2. Spesifikasikan variabel yang akan diubah menjadi *factor* (lihat Tabel 2.14). Klik OK
3. Untuk mengecek apakah variabel telah terkonversi menjadi *factor*, klik Statistics/Summaries/Active data set. Variabel yang telah dikonversi menjadi *factor* akan memberikan ringkasan data berupa tabel kontingensi (tidak menampilkan mean, min, max, dll).

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.14.

Sintaks yang dihasilkan pada proses tersebut adalah sebagai berikut:



Gambar 2.14: Tampilan tahapan merubah variabel numeric menjadi factor.

```
# use numbers
dataset <- within(dataset, {
  namavariabel <- as.factor(namavariabel)
})

# supply level names
dataset<- within(dataset, {
  namavariabel <- factor(namavariabel,
                         labels=c('label1','label2',...))
})
```

Tabel 2.14: Penjelasan item convert numeric variables to factors.

No. Item	Jenis In-put	Keterangan
1. Variables	select box	daftar nama variabel yang akan diubah menjadi <i>factor</i>
2. Factor levels	radio button	pilihan apakah perlu menambahkan label pada <i>factor</i> atau tidak
3. New variable name or...	text input	jika tidak diisi maka variabel lama akan diganti variabel baru (tidak ada variabel baru ditambahkan)

2.8.6 Melakukan *Binning* pada Variabel Numeric

Binning variabel numeric merupakan cara untuk mengelompokkan nilai variabel *numeric* ke dalam suatu kelas berdasarkan rentang tertentu. Penetapan rentang pada proses *binning* di R Commander terbagi atas 3 metode, yaitu:

- **Equal-count** : membagi data ke dalam kelas berdasarkan jumlah kelas yang sama.
- **Natural break** : membagi data ke dalam kelas berdasarkan jarak terdekat (biasanya menggunakan jarak Euclidian) pada pusat masing-masing kelas. Algoritma pengelompokan yang biasa digunakan adalah algoritma *k-means* (baca K-Means¹).

Tahapan untuk melakukan proses *binning* variabel numeric antara lain:

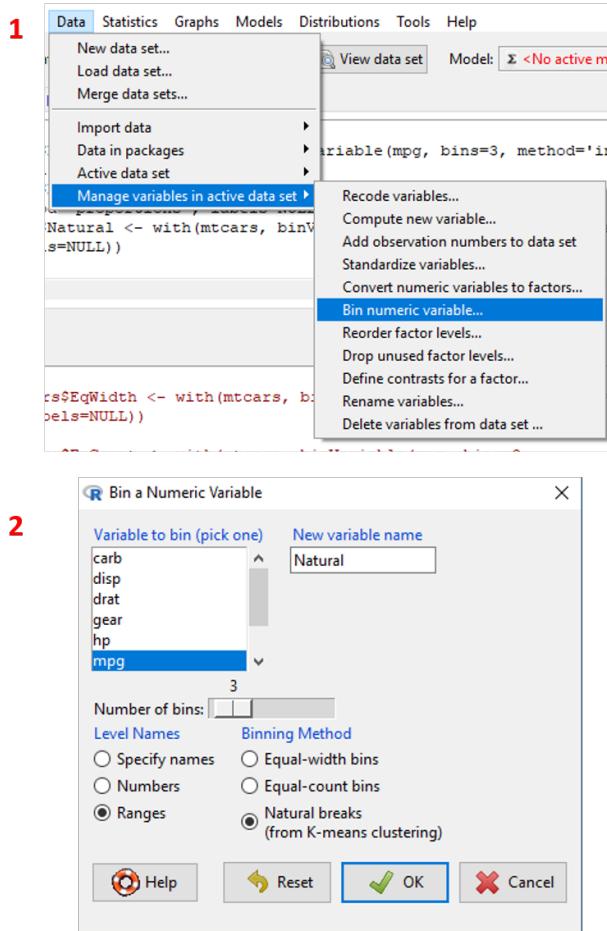
1. Pada menu Data, klik `Data/Manage variables in active data set/Bin numeric variable....`
2. Pada jendela yang muncul spesifikasikan variabel numeric yang akan di *binning* dan metode pengelompokan yang digunakan (lihat Tabel 2.15). Klik **ok**.
3. Untuk mengecek variabel hasil *binning*, klik *toolbar view data set*.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.15.

Sintaks yang digunakan pada masing-masing metode *binning*, antara lain:

```
# Equal-width
dataset$namavariabel <- with(dataset, binVariable(variabel,
                                                 bins="jumlah_bin", method='intervals',
                                                 labels=c("label1",...)))
dataset$namavariabel <- with(dataset, binVariable(variabel,
                                                 bins="jumlah_bin", method='proportions',
                                                 labels=c("label1",...)))
dataset$namavariabel <- with(dataset, binVariable(variabel,
                                                 bins="jumlah_bin", method='natural',
                                                 labels=c("label1",...)))
```

¹<https://id.wikipedia.org/wiki/K-means>



Gambar 2.15: Tampilan tahapan binning variabel numeric.

Tabel 2.15: Penjelasan item jendela convert numeric variables to factors.

No. Item	Jenis In-put	Keterangan
1. Variables to bin	select box	daftar nama variabel <i>numeric</i> yang akan dilakukan <i>binning</i>

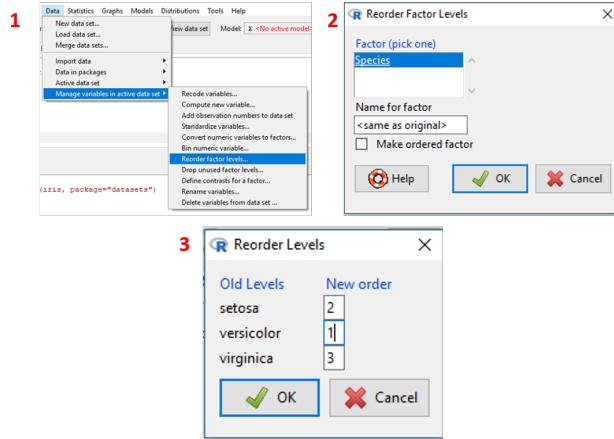
No. Item	Jenis In-put	Keterangan
2. New variable name	<i>text input</i>	Input nama variabel baru
3. Number of bins	<i>slider</i>	spesifikasi jumlah bin atau kelas yang digunakan
4. Level Names	<i>radio button</i>	spesifikasi metode penamaan bin
5. Binning Method	<i>radio button</i>	spesifikasi metode <i>binning</i>

2.8.7 Merubah Urutan *Factor Levels*

Secara umum saat kita merubah sebuah variabel *character* atau *string* menjadi *factor* level *factor* hasil konversi tersebut diurutkan berdasarkan abjad. Sebuah variabel yang terdiri dari nilai “setuju”, “netral”, dan “tidak setuju”, jika diubah menjadi *factor* akan memiliki urutan level “netral”, “setuju”, dan “tidak setuju”. Urutan tersebut tidak benar dan perlu dirubah. Untuk melakukannya pada R Commander jalankan langkah-langkah berikut:

1. Pada menu Data, klik Data/Manage variables in active data set/Reorder factor levels....
2. Pada jendela yang muncul spesifikasikan variable *factor* yang akan dirubah (lihat Tabel 2.16). Klik OK.
3. Pada jendela yang muncul, rubah urutan *factor* lama. Klik OK.
4. Untuk mengecek *factor level*, jalankan sintaks berikut:

```
# ubah nama dataset dan variabel
levels(dataset$namavariabel)
```



Gambar 2.16: Tampilan tahapan merubah urutan factor level.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.16.

Sintaks dari tahapan merubah urutan *factor* secara umum ditzampilkan sebagai berikut:

```
dataset$namavariabel <- with(dataset, factor(namavariabel,
                                             levels=c('level1',
                                                      'level2',
                                                      ....),
                                             ordered=TRUE))
```

Tabel 2.16: Penjelasan item jendela reorder factor levels.

No. Item	Jenis In-put	Keterangan
1. Factor	select box	daftar variabel <i>factor</i> pada dataset
2. Name for factor	text input	nama variabel <i>factor</i> yang baru (jika ingin membuat variabel baru)

No. Item	Jenis In-put	Keterangan
3. Make ordered factor	check box	spesifikasi apakah <i>factor</i> akan diurutkan atau tidak

2.8.8 Melakukan *Drop* pada *Factor Levels*

Saat melakukan subset pada dataset yang akan dijelaskan pada Chapter 2.9 sering kali tidak semua *factor level* ada pada dataset tersebut (sejumlah *factor level* memiliki observasi nol) yang berpengaruh pada analisis data yang kita lakukan. Untuk mengatasinya, kita dapat melakukan *drop* pada *factor level* tersebut. Untuk melakukannya, jalankan langkah-langkah berikut:

1. Pada menu Data, klik `Data/Manage variables in active data set/Drop unused factor levels....`
2. Pada jendela yang muncul spesifikasikan variable *factor* yang akan didrop *factor levelnya*. Klik **ok**.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.17.

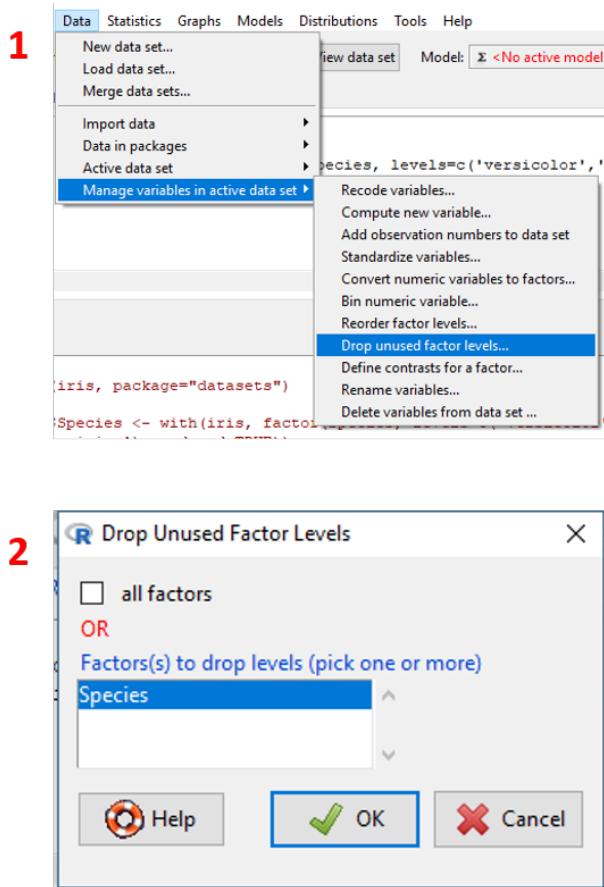
Sintaks untuk melakukan *drop factor levels* secara umum adalah sebagai berikut:

```
dataset <- within(dataset, {
  namavariabel <- droplevels(namavariabel)
})
```

2.8.9 Merubah Nama Variabel pada Data

Untuk merubah nama variabel pada `R Commander` dapat dilakukan dengan dua cara, antara lain:

Cara 1



Gambar 2.17: Tampilan tahapan drop factor level.

1. Klik toolbar Edit data set.
2. Pada jendela dataset yang muncul , double click nama variabel yang ingin dirubah dan ubah nama variabel tersebut. Klik ok

Cara 2

1. Pada menu Data, klik Data/Manage variables in active data set/Rename variables....
2. Pada jendela yang muncul pilih variabel yang ingin dirubah namanya. Klik ok.

3. Pada jendela yang muncul, isikan nama variabel baru dan Klik **ok** jika telah selesai.
4. Untuk mengecek apakah proses telah berhasil, klik *toolbar View data set*.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.18.

Sintaks untuk merubah nama variabel secara umum adalah sebagai berikut:

```
names(dataset)[c(nomorvariabel)] <- c("namavariabelbaru")
```

2.8.10 Menghapus Variabel

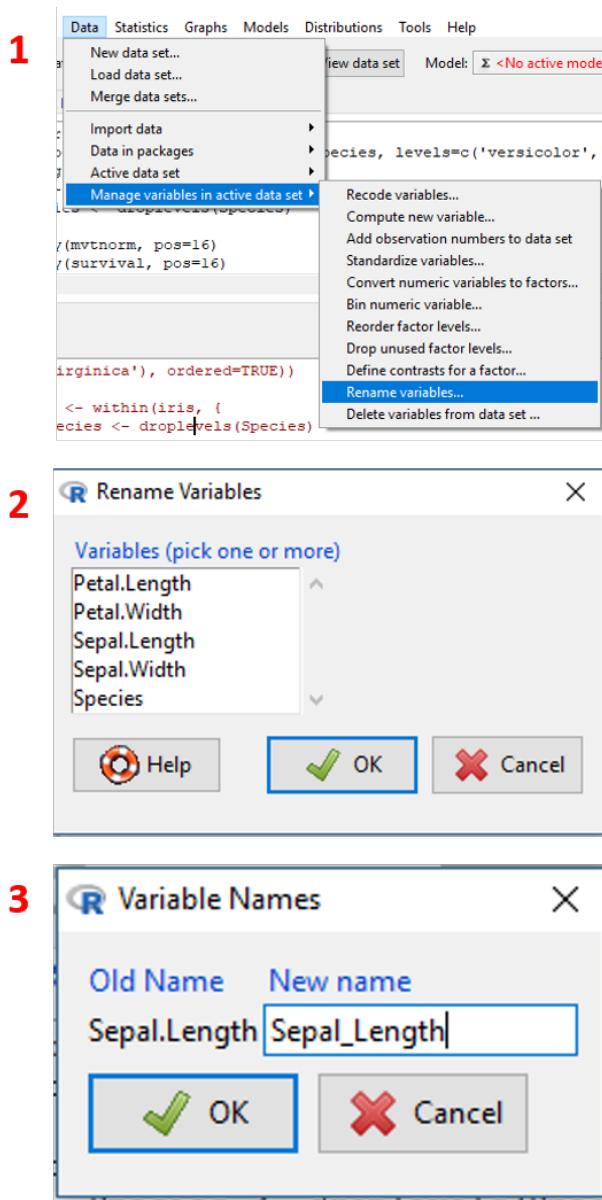
Untuk menghapus variabel pada dataset, jalankan langkah-langkah berikut:

1. Pada menu **Data**, klik **Data/Manage variables in active data set/Delete variables from data set....**
2. Pada jendela yang muncul pilih variabel yang ingin dihapus. Klik **ok**.
3. Untuk mengecek apakah proses telah berhasil, klik *toolbar View data set*.

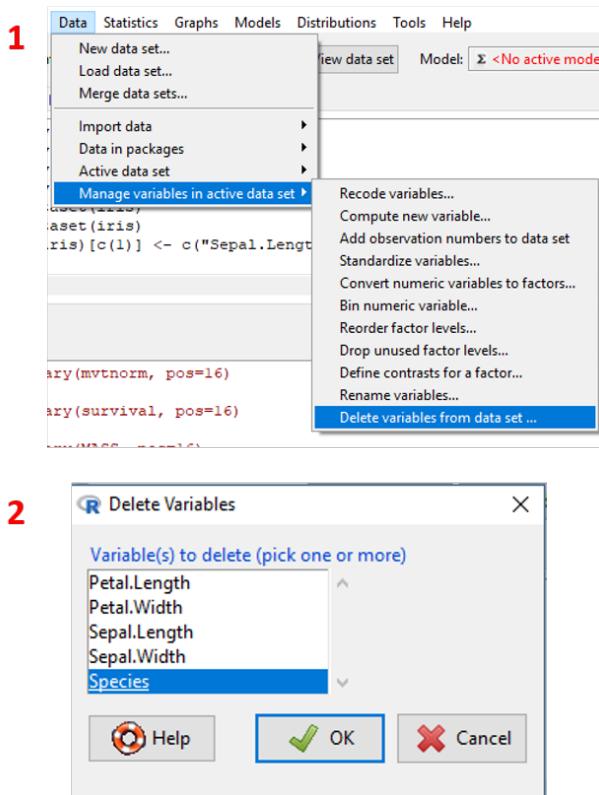
Visualisasi tahapan tersebut ditampilkan pada Gambar 2.19.

Sintaks untuk menghapus variabel secara umum adalah sebagai berikut:

```
dataset <- within(dataset, {  
  namavariabel1 <- NULL  
  namavariabel2 <- NULL  
})
```



Gambar 2.18: Tampilan tahapan merubah nama variabel.



Gambar 2.19: Tampilan tahapan menghapus variabel.

2.9 Memanipulasi Dataset

Pada Chapter 2.9, kita akan belajar bagaimana memanipulasi dataset. Adapun yang menjadi topik bahasan dalam Chapter 2.9, antara lain:

- Melakukan subset dataset,
- Agregasi variabel pada dataset,
- Melakukan *drop* observasi pada dataset,
- Melakukan *drop* observasi dengan *missing value*,
- Mengelompokkan variabel menjadi variabel *factor* dan nilai,
- Menggabungkan dua dataset, dan,

- Modifikasi lainnya.

2.9.1 Melakukan Subset Dataset

Pada analisis data, kita sering kali tidak membutuhkan seluruh observasi dari data yang kita miliki. Kita akan melakukan filter untuk memperoleh data sesuai dengan keperluan kita. Selain itu, melihat subset data berdasarkan kriteria tertentu membantu kita untuk melakukan analisis eksploratif terhadap data yang kita miliki.

Untuk melakukan subset data, kita memerlukan sebuah ekspresi atau formula yang dapat mengecek satu persatu data yang memenuhi formula subset yang telah dibentuk. Sebagai contoh, kita ingin memperoleh dataset tanpa nilai *missing value* pada variabel `namavariabel`. Berikut adalah contoh sintaks formula atau ekspresi yang digunakan:

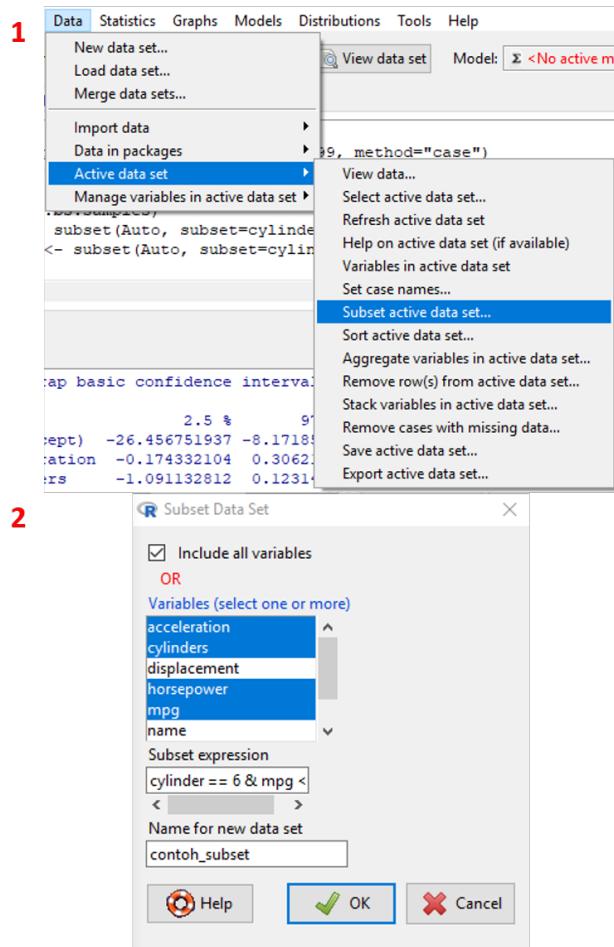
```
!is.na(namavariabel)
```

Untuk membuat formula atau ekspresi tersebut, kita dapat menggunakan kembali operator operasi yang telah dijelaskan pada Chapter 2.1. Langkah-langkah untuk melakukan subset pada dataset adalah sebagai berikut:

1. Pada menu `Data`, klik `Data/Active data set/Subset active data set....`
2. Pada jendela yang muncul spesifikasikan variabel yang akan dipilih pada dataset baru dan formula subset yang digunakan (lihat Tabel 2.17). Klik `ok`.
3. Untuk mengecek dataset, klik `toolbar view data set`.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.20.

Secara umum sintaks untuk melakukan proses subset adalah sebagai berikut:



Gambar 2.20: Tampilan tahapan melakukan subset pada dataset.

```
contoh_subset <- subset(dataset,subset=formula_subset)
```

Tabel 2.17: Penjelasan item jendela subset data set.

No. Item		Jenis In-put	Keterangan
1.	Include all variables	check box	opsi apakah akan menyertakan seluruh variabel pada hasil subset
2.	Variables	select box	daftar nama variabel yang dapat dipilih untuk ditampilkan pada hasil subset
3.	Subset expression	text input	formula atau ekspresi subset yang digunakan
4.	Name for new data set	text input	opsi untuk memberikan nama pada dataset baru atau tidak (dataset lama akan dihapus)

2.9.2 Agregat Variabel pada Dataset

Submenu `aggregate variables in active dataset` memberikan ringkasan nilai satu atau beberapa variabel berdasarkan level variabel *factor* dan menghasilkan dataset baru dengan satu observasi untuk tiap level *factor*. Proses agregasi mengaplikasikan beberapa fungsi seperti `mean()`, `sum()`, atau fungsi lainnya untuk menghasilkan sebuah nilai untuk setiap observasi pada tiap variabel dan level *factor*.

Fungsi statistika deskriptif bawaan yang dapat digunakan pada R Commander ditampilkan pada Tabel 2.18.

Tabel 2.18: Fungsi statistika deskriptif bawaan pada R.

Fungsi	Keterangan
<code>mean</code>	rata-rata
<code>median</code>	median
<code>sum</code>	jumlah seluruh observasi dalam sebuah variabel
<code>quantile</code>	kuantil data
<code>min</code>	nilai observasi minimum

Fungsi	Keterangan
<code>max</code>	nilai observasi maksimum
<code>IQR</code>	rentang antar kuartil
<code>mad</code>	simpangan absolut median
<code>sd</code>	simpangan baku
<code>var</code>	varians

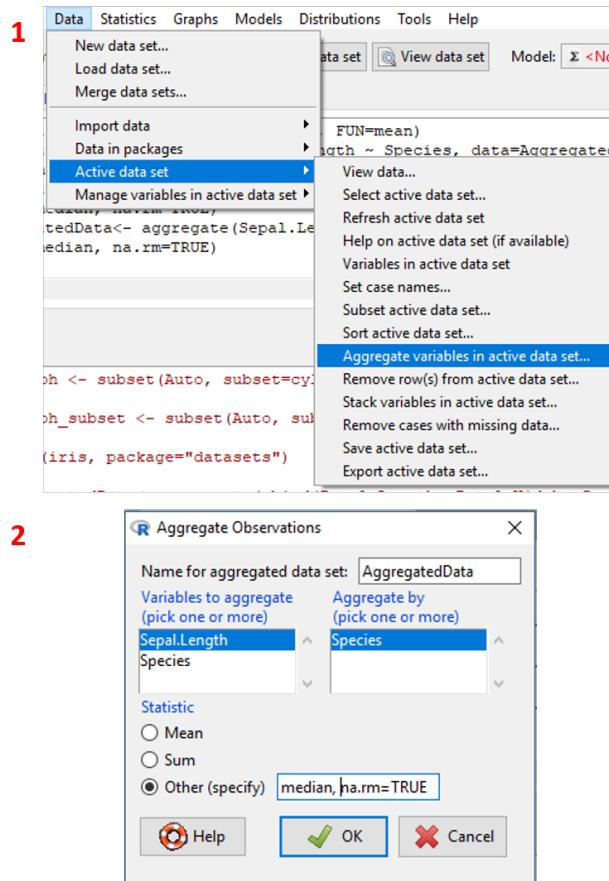
Tahapan agregasi variabel dilakukan melalui langkah-langkah berikut:

1. Pada menu `Data`, klik `Data/Active data set/Aggregate variables in active data set....`
2. Spesifikan variabel yang akan dilakukan agregasi, *factor level* yang digunakan, dan fungsi agregat yang digunakan (lihat Tabel 2.19). Klik `ok`.
3. Untuk mengecek dataset, klik `toolbar view data set`.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.21.

Tabel 2.19: Penjelasan item jendela aggregate observations.

No. Item	Jenis In-put	Keterangan
1.	Name <i>text</i> for <i>input</i>	input nama dataset baru yang dihasilkan aggregated..
2.	Variables <i>select</i> to <i>box</i>	daftar nama variabel yang dapat dipilih untuk diagregasi aggregate
3.	Aggregated <i>select</i> by <i>box</i>	daftar nama variabel factor yang digunakan untuk agregasi
4.	Statistic <i>text</i> <i>imput</i>	fungsi statistik yang digunakan untuk aggregasi



Gambar 2.21: Tampilan tahapan agregasi variabel.

Sintaks untuk melakukan proses agragasi secara umum dituliskan sebagai berikut:

```
datasetbaru<- aggregate(dataframe ~ var_factor,
                         data=dataset, FUN=statistic, ...)
```

Argumen ... pada sintaks tersebut merupakan argumen tambahan pada fungsi statistik yang digunakan. Secara umum argumen tambahan yang digunakan fungsi yang ditampilkan pada Tabel 2.18 adalah na.rm=TRUE, yaitu: jika data mengandung *missing value*, maka observasi yang mengandung *missing value* tersebut akan di

drop. Selain argumen tersebut, fungsi `quantile` memerlukan argumen spesifikasi `prob` atau spesifikasi kuantil yang akan ditampilkan (misal: `prob=0.5` untuk kuantil ke-50 atau median). Untuk menambahkan argumen tersebut, pembaca dapat menspesifikasikannya pada bagian `Statistic` pada jendela `aggregate observations` yaitu pada pilihan `Other (specify)` seperti yang ditunjukkan pada Gambar 2.21.

2.9.3 Melakukan *Drop* Observasi pada Dataset

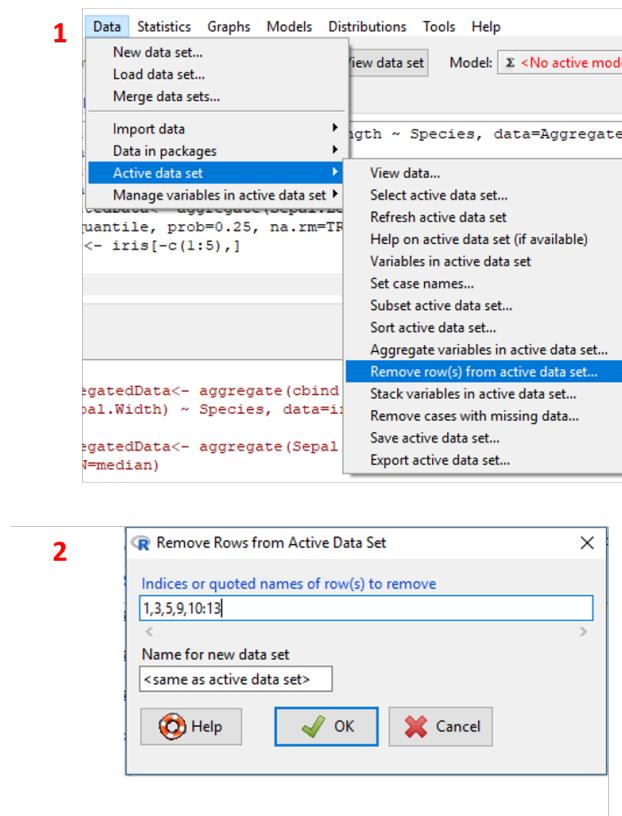
Kita dapat melakukan drop terhadap sejumlah baris observasi berdasarkan indeks (nomor baris observasinya) atau nama baris observasinya. Seleksi baris yang akan *didrop* dilakukan menggunakan metode yang ditampilkan pada Tabel 2.12, yaitu: baris individual (contoh: `1`), set baris (contoh: `1,2,3...,k`), atau rentang baris (contoh: `1:5`) untuk seleksi menggunakan indeks. Langkah-langkah untuk melakukan seleksi baris yang akan *didrop* adalah sebagai berikut:

1. Pada menu `Data`, klik `Data/Active data set/Remove row(s) from active data set....`
2. Spesifikan index baris yang akan di drop dan nama output dataset baru (lihat Tabel 2.20). Klik `ok`.
3. Untuk mengecek dataset, klik `toolbar view data set`.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.22.

Sintaks yang digunakan untuk melakukan *drop* baris observasi secara umum adalah sebagai berikut:

```
datasetbaru <- dataset[-c(indeks),]
```



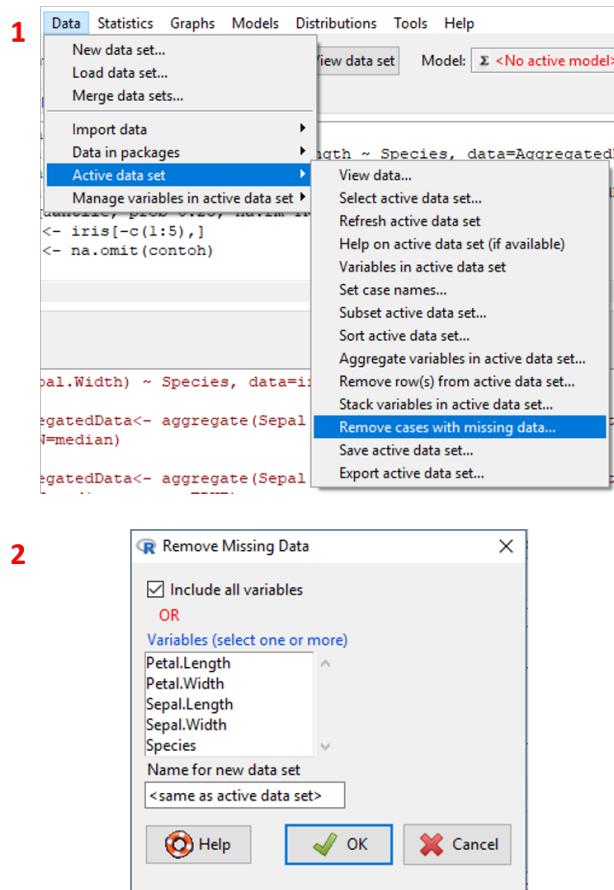
Gambar 2.22: Tampilan tahapan melakukan drop observasi.

Tabel 2.20: Penjelasan item jendela remove rows from active data set.

No. Item	Jenis In-put	Keterangan
1. Indices or quoted...	<i>text</i> or <i>input</i>	indeks atau nama baris yang akan di <i>drop</i> atau dihapus
2. Name for new data set	<i>text</i> or <i>input</i>	input nama dataset baru yang dihasilkan

2.9.4 Melakukan *Drop Observasi* dengan *Missing Value*

Selain menggunakan submenu *subset active data set*, *drop missing value* dapat pula dilakukan dengan menggunakan submenu *remove cases with missing data*. Perbedaan antara metode pertama dan kedua adalah pada metode kedua *drop missing value* dilakukan pada seluruh variabel dalam dataset. Tahapan untuk melakukannya adalah sebagai berikut:



Gambar 2.23: Tampilan tahapan melakukan drop observasi dengan missing value.

Tabel 2.21: Penjelasan item jendela remove missing data.

No. Item	Jenis In-put	Keterangan
1. Include all variables	check box	pilihan apakah akan menyertakan seluruh variabel pada dataset baru atau tidak
2. Variables	select box	daftar variabel yang dapat dipilih untuk disertakan dalam dataset baru

No. Item	Jenis In-put	Keterangan
3. Name for new data set	<i>text</i> <i>input</i>	input nama dataset baru

2.9.5 Mengelompokkan Variabel Menjadi Variabel *Factor* dan Nilai

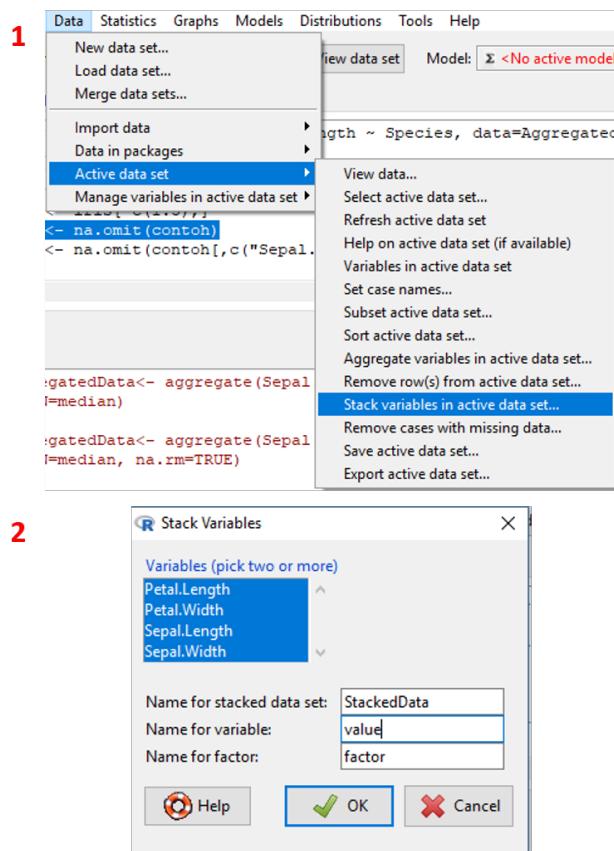
Submenu `stack variables in active data set` membuat sebuah dataset baru dimana dua atau lebih variabel ditumpuk menjadi variabel satu variabel dengan nilai variabel sebelumnya ditampilkan pada variabel nilai. Jika terdapat n observasi dalam dataset dan k variabel, maka dataset baru akan terdiri dari 2 variabel (*factor* dan *numeric*) dan $n \times k$ observasi. Tahapan untuk melakukannya adalah sebagai berikut:

1. Pada menu **Data**, klik `Data/Active data set/Stack variables in active data set....`
2. Spesifikan variabel yang akan dipilih untuk dataset baru dan nama output dataset (lihat Tabel 2.22). Klik **ok**.
3. Untuk mengecek dataset, klik `toolbar view data set`.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.24.

Terdapat dua buah sintaks dalam melakukan proses pengelompokan data. Sintaks pertama melakukan pengelompokan data, sedangkan sintaks kedua merubah nama kolom dataset baru. Kedua sintaks tersebut adalah sebagai berikut:

```
datasetbaru <- stack(dataset[, c("variabel1", "variabel2", ..)])
names(datasetbaru) <- c("value", "factor")
```



Gambar 2.24: Tampilan tahapan mengelompokkan variabel menjadi variabel factor dan nilai.

Tabel 2.22: Penjelasan item jendela stack variables.

No. Item	Jenis In-put	Keterangan
1.	Variables select box	daftar variabel yang dapat dipilih untuk dikelompokkan
2.	Name for staked data..	text input nama dataset baru hasil pengelompokan

No.	Item	Jenis In- put	Keterangan
3.	Name for variable	<i>text</i> <i>input</i>	nama variabel nilai hasil pengelompokan
4.	Name for factor	<i>text</i> <i>input</i>	nama variabel <i>factor</i> hasil pengelompokan nama variabel pada dataset sebelumnya

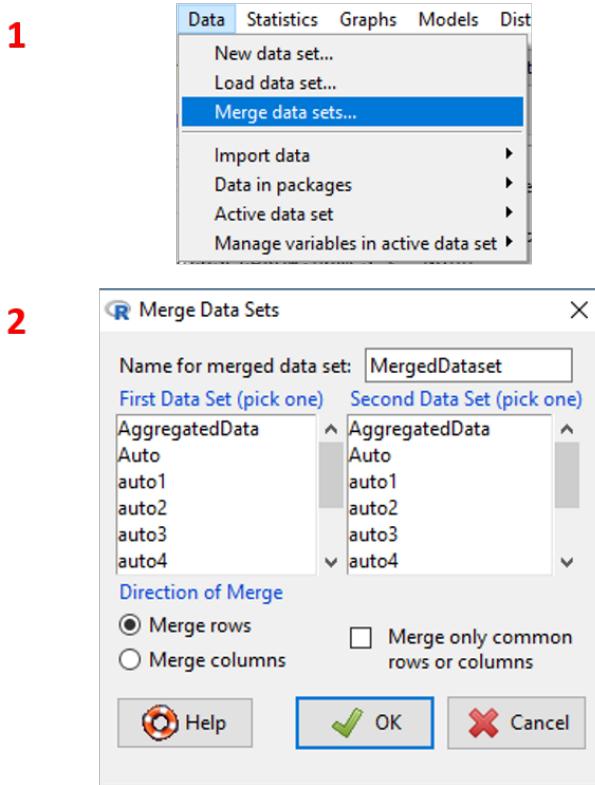
2.9.6 Menggabungkan Dua Dataset

Untuk menggabungkan dua buah dataset, kedua dataset perlu memiliki elemen kunci yang sama. Secara *default* R Commander mengambil *rownames* sebagai elemen kunci, sehingga elemen kunci yang dimiliki oleh masing-masing dataframe haruslah konsisten satu sama lain. Penggabungan dataset dapat dilakukan melalui penggabungan kolom dan baris. Penggabungan juga dapat dilakukan melalui elemen unik dari baris maupun kolom. Maksudnya adalah R Commander hanya menggabungkan dataset yang memiliki elemen kunci sama (*rownames* sama) pada kedua dataset. Jika terdapat observasi pada dataset 1 dan tidak memiliki *rownames* atau elemen kunci sama pada dataset 2, maka observasi tersebut akan dihapus dari proses penggabungan data. Tahapan penggabungan dua buah dataset adalah sebagai berikut:

1. Pada menu **Data**, klik **Data/Merge data set....**
2. Pada jendela yang muncul, klik dataset yang akan digabungkan dan cara penggabungannya (lihat Tabel 2.23). Klik **OK**.
3. Untuk mengecek dataset, klik **toolbar view data set**.

Visualisasi tahapan tersebut ditampilkan pada Gambar 2.25.

Sintaks yang digunakan untuk menggabungkan dua buah dataset secara umum adalah sebagai berikut:



Gambar 2.25: Tampilan tahapan menggabungkan dua buah dataset.

```
# penggabungan baris tanpa mempertimbangkan
# elemen unik
datasetbaru <- mergeRows(data1, data2, common.only=FALSE)

# penggabungan kolom dengan mempertimbangkan
# elemen unik
datasetbaru <- merge(data1, data2, all=FALSE, by="row.names")
rownames(datasetbaru) <- datasetbaru$Row.names
datasetbaru$Row.names <- NULL
```

Tabel 2.23: Penjelasan item jendela merge data sets.

No. Item	Jenis In-put	Keterangan
1. Name for merge data set	<i>text</i> <i>input</i>	input nama dataset baru
2. First Data set	<i>select</i> <i>box</i>	daftar dataset pertama yang akan digabung
3. Second Data set	<i>select</i> <i>box</i>	daftar dataset kedua yang akan digabung
4. Direction or Merge	<i>radio</i> <i>but-ton</i>	pilihan cara penggabungan
5. Merge only common...	<i>check</i> <i>box</i>	opsi apakah hanya observasi dengan elemen unik yang ada pada dua buah dataset yang akan digabungkan

2.9.7 Modifikasi Lainnya

Daftar submenu yang digunakan untuk melakukan modifikasi dataset lainnya pada R Commander, antara lain:

- **Select active data set** : submenu ini memiliki fungsi sama dengan *toolbar Data set*, yaitu: memilih dataset yang diaktifkan. Jika pada jendela yang muncul terdapat beberapa dataset yang dapat diaktifkan, pembaca dapat memilih satu dataset dari daftar tersebut untuk diaktifkan.
- **Refresh active data set** : submenu yang berfungsi untuk melakukan *reset* informasi yang dipertahankan terkait dataset yang aktif, seperti: nama variabel pada dataset, jenis data pada dataset tersebut, dll. Pembaca mungkin perlu melakukan *refresh* terhadap dataset yang aktif jika pembaca melakukan sebuah perubahan pada dataset tersebut diluar menu R Commander.

Tabel 2.24: Hasil pemantauan parameter DO dan BOD outlet IPAL X

Bulan	DO	BOD
1	1.09	10.09
2	5.23	8.13
3	6.95	11.13
4	4.35	9.27
5	6.96	12.01
6	-0.96	10.26
7	3.47	11.84
8	3.98	11.40
9	5.68	9.20
10	4.72	9.55
11	4.13	8.04
12	5.93	11.05

- `Help on active data set` : menampilkan dokumentasi terkait dataset yang aktif (biasanya dataset yang ada dalam sebuah paket).
- `Variables in active data set` : menampilkan daftar variabel yang ada dalam data set.
- `Set case name` : memilih sebuah variabel pada dataset aktif dan selanjutnya menjadikannya variabel tersebut sebagai *rownames* observasi dalam dataset tersebut.

2.10 Latihan

1. Sebuah IPAL memiliki laporan hasil pemantauan parameter *dissolved oxygen* (DO) dan *biochemical oxygen demand* (BOD) pada outlet IPAL yang ditampilkan pada Tabel 2.24. Inputkan data tersebut ke dalam `R Commander`!
2. Data pengukuran inlet dan outlet konsentrasi BOD,

Tabel 2.25: Hasil pemantauan parameter DO,BOD,dan COD 10 IPAL Kota X

	DOinlet	DOoutlet	BODinlet	BODoutlet	CODinlet	CODoutlet
IPAL 1	6.27	0.87	34.35	9.08	59.30	11.62
IPAL 2	5.10	2.12	37.19	9.11	53.20	25.19
IPAL 3	4.26	2.76	28.87	10.24	49.30	27.90
IPAL 4	4.29	3.02	37.38	4.75	51.94	17.89
IPAL 5	5.51	1.66	24.23	13.54	54.57	28.06
IPAL 6	4.19	0.32	33.26	8.63	45.55	23.26
IPAL 7	6.18	2.82	31.13	11.14	50.75	23.36
IPAL 8	6.00	1.51	32.37	13.51	54.50	8.00
IPAL 9	4.03	0.87	38.60	6.98	49.33	18.27
IPAL 10	4.92	2.61	33.78	8.76	58.32	28.17

COD, dan DO dari 10 IPAL yang ada di Kota X ditampilkan pada Tabel 2.25. Hitung efisiensi pengolahan masing-masing IPAL tersebut untuk ketiga parameter tersebut dan amati perubahan rasio BOD/COD pada inlet dan outlet masing-masing IPAL tersebut!

3. Lakukan *data merging* pada elemen unik antara Tabel 2.25 dan Tabel 2.26!

Tabel 2.26: Hasil pemantauan parameter TSS 12 IPAL Kota X

	TSSinlet	TSSoutlet
IPAL 1	20.69	8.01
IPAL 2	20.23	20.62
IPAL 3	19.21	14.88
IPAL 4	18.66	12.91
IPAL 5	20.31	6.45
IPAL 6	18.54	9.33
IPAL 7	21.25	5.27
IPAL 8	19.56	17.32
IPAL 9	19.09	9.68
IPAL 10	21.04	9.41
IPAL 11	20.43	15.62
IPAL 12	20.19	8.42



3

Ringkasan dan Visualisasi Data



4

Uji Statistik Sederhana



5

Linier dan Generalized Linear Model



6

Distribusi Probabilitas dan Simulasi



Referensi

1. Fox, J. 2005. **The R Commander: A Basic-Statistics Graphical User Interface to R.** Journal of Statistical Software.Vol:14(9), p:1-42.
2. Fox, J. 2017. **Using the R Commander: A Point-and-Click Interface for R.** CRC Press.
3. Fox, J. Valat, M.B. 2018. **Getting Started With the R Commander.**<https://socialsciences.mcmaster.ca/jfox/Misc/Rcmdr/Getting-Started-with-the-Rcmdr.pdf>.
4. Gio, P.U. Irawan, D.E. 2016. **Belajar Statistika dengan R (disertai beberapa contoh perhitungan manual).** USU Press : Medan.
5. Nguyen-Feng, V. Stellmack, M.A. 2016. **A Guide to Data Analysis in R Commander.** University of Minnesota.
6. Primartha, R. 2018. **Belajar Machine Learning Teori dan Praktik.** Penerbit Informatika : Bandung.
7. Quick-R. **Data Input.** <https://www.statmethods.net/input/index.html>
8. Quick-R. **Data Management.** <https://www.statmethods.net/management/index.html>
9. Rosadi,D. 2011. **Analisis Ekonometrika dan Runtun Waktu Terapan dengan R.** Penerbit Andi: Yogyakarta.
10. Rosadi,D. 2016. **Analisis Statistika dengan R.** Gadjah Mada University Press: Yogyakarta.
11. Rosidi, M. 2019. **Metode Numerik Menggunakan R Untuk Teknik Lingkungan.** https://bookdown.org/moh_rosidi2610/Metode_Numerik/.
12. STHDA. **Importing Data Into R .** <http://www.sthda.com/english/wiki/importing-data-into-r>

13. STHDA. Exporting Data From R. <http://www.sthda.com/english/wiki/exporting-data-from-r>
14. STDHA. Getting Help With Functions In R Programming. <http://www.sthda.com/english/wiki/getting-help-with-functions-in-r-programming> .
15. Venables, W.N. Smith D.M. and R Core Team. 2018. **An Introduction to R.** R Manuals.
16. Wickham, H. Grolemund G. 2016. **R For Data Science: Import, Tidy, Transform, Visualize, And Model Data.** O'Reilly Media, Inc.
17. Widodo, B. Rachmawati, R.N. 2013. **Pengantar Praktis Pemrograman R untuk Ilmu Komputer.** Halaman Moeka Publishing : Jakarta.