

# Data Analysis: A Simple Guide to Using R Commander

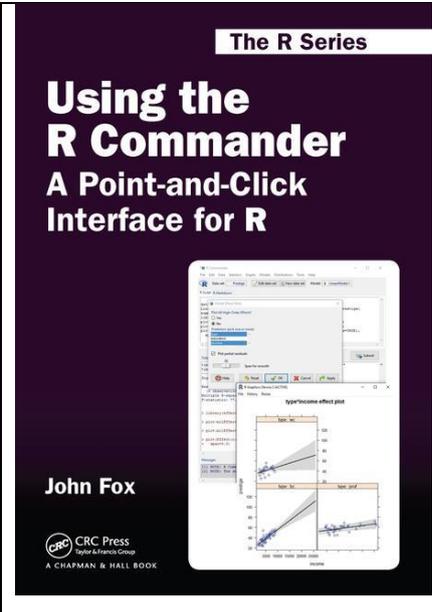
By Dr.Parinya Siriatakul (Ph.D.)

## I: Introduction

### Download R Version 4.4.1

R for Windows from <https://cran.r-project.org/bin/windows/base/>.

R for MacOS from <https://cran.r-project.org/>.

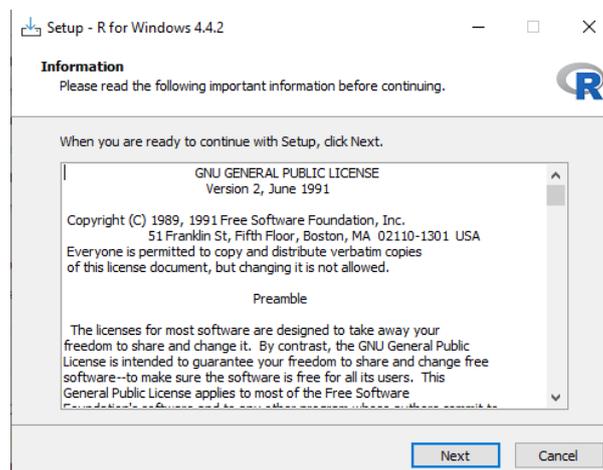
	<p><b>R Commander:</b></p> <p>is A platform-independent basic-statistics GUI (graphical user interface) for R, based on the tcltk package. R commander is free statistical software. R commander was developed as an easy to use graphical user interface (GUI) for R (freeware statistical programming language) and was developed by Prof. John Fox to allow the teaching of statistics courses and removing the hindrance of software complexity from the process of learning statistics. This means it has drop down menus that can drive the statistical analysis of data. It is considered the most viable R alternative to commercial statistical packages like SPSS. The package is highly useful to R novices, since for each analysis run it displays the underlying R code.</p>
--	--

Home page: <https://www.john-fox.ca/RCommander/installation-notes.html>

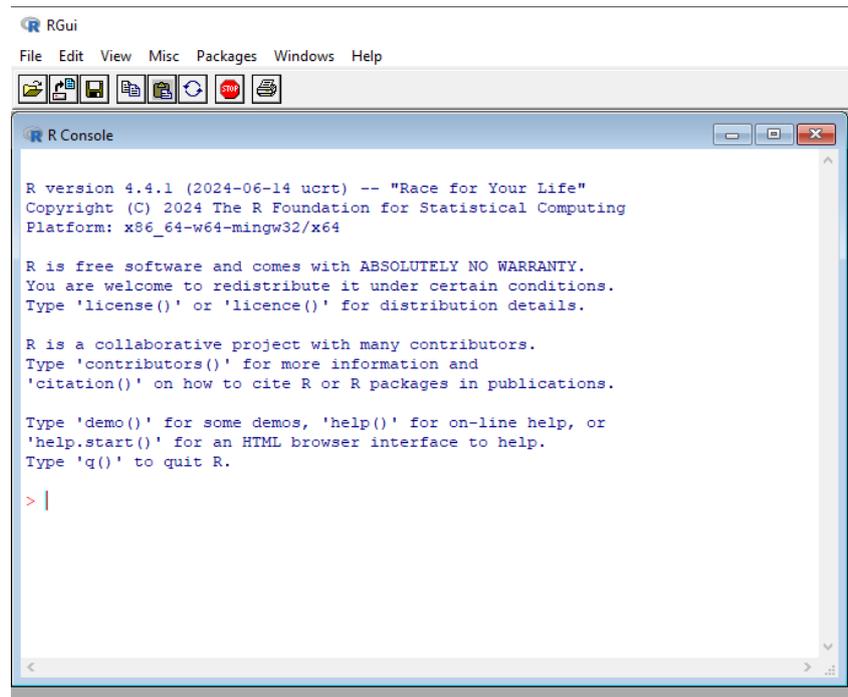
To install R Commander, follow these steps:

1.Install R: If you haven't already, make sure you have R installed on your computer.

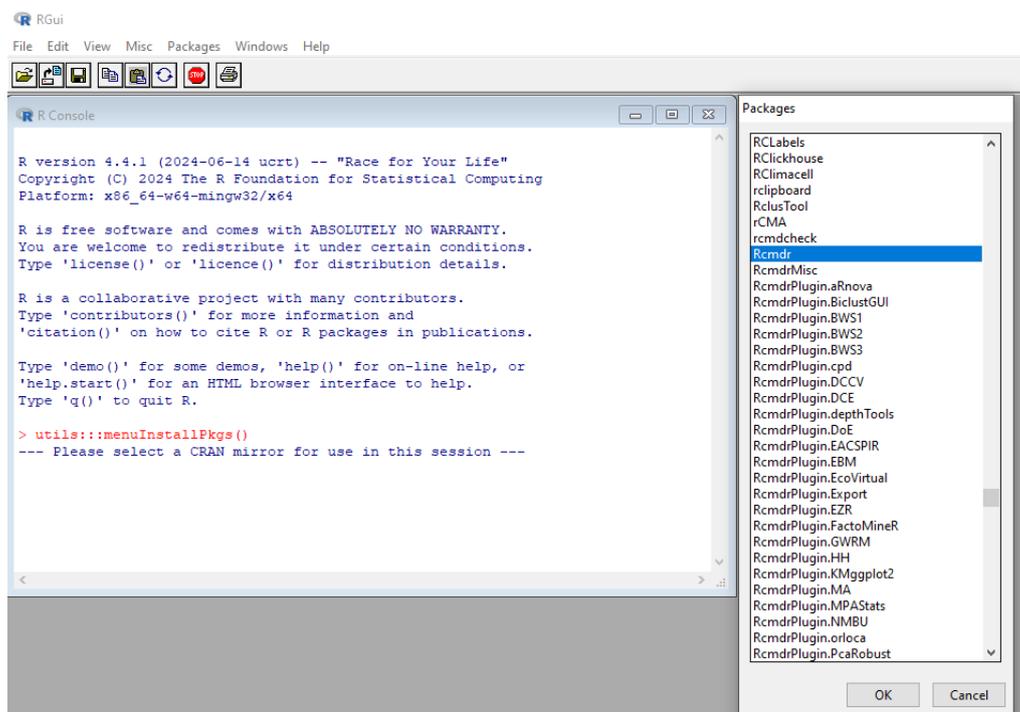
You can download it from [CRAN](https://cran.r-project.org/).



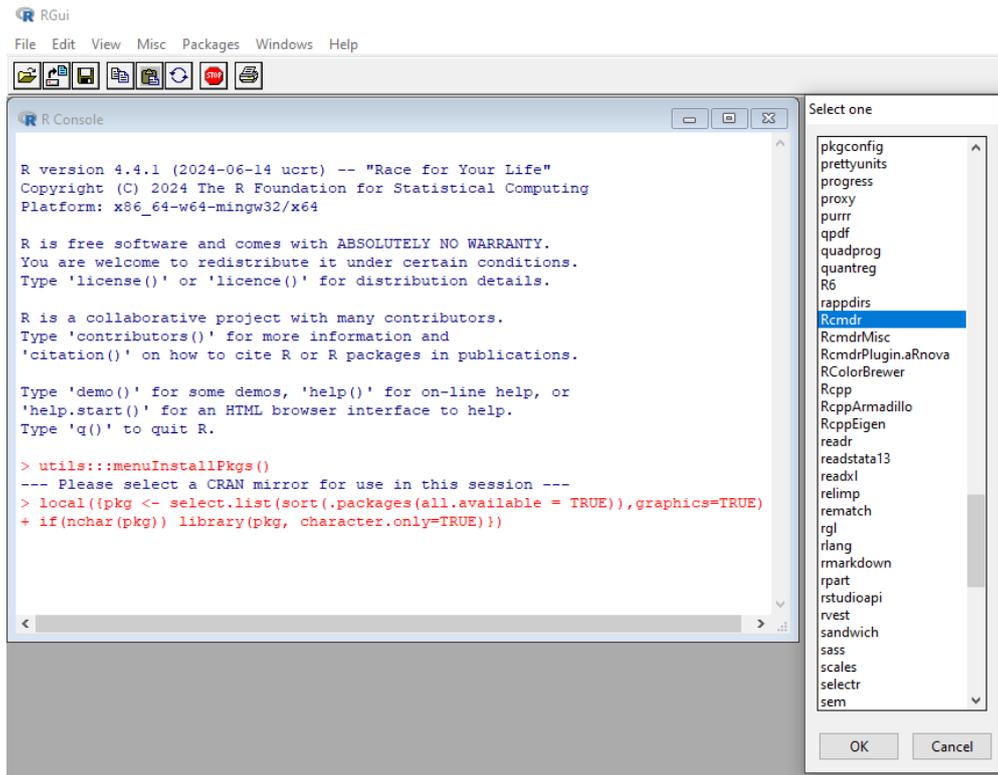
## 2. Open R or RStudio: Launch the R or RStudio application on your computer.



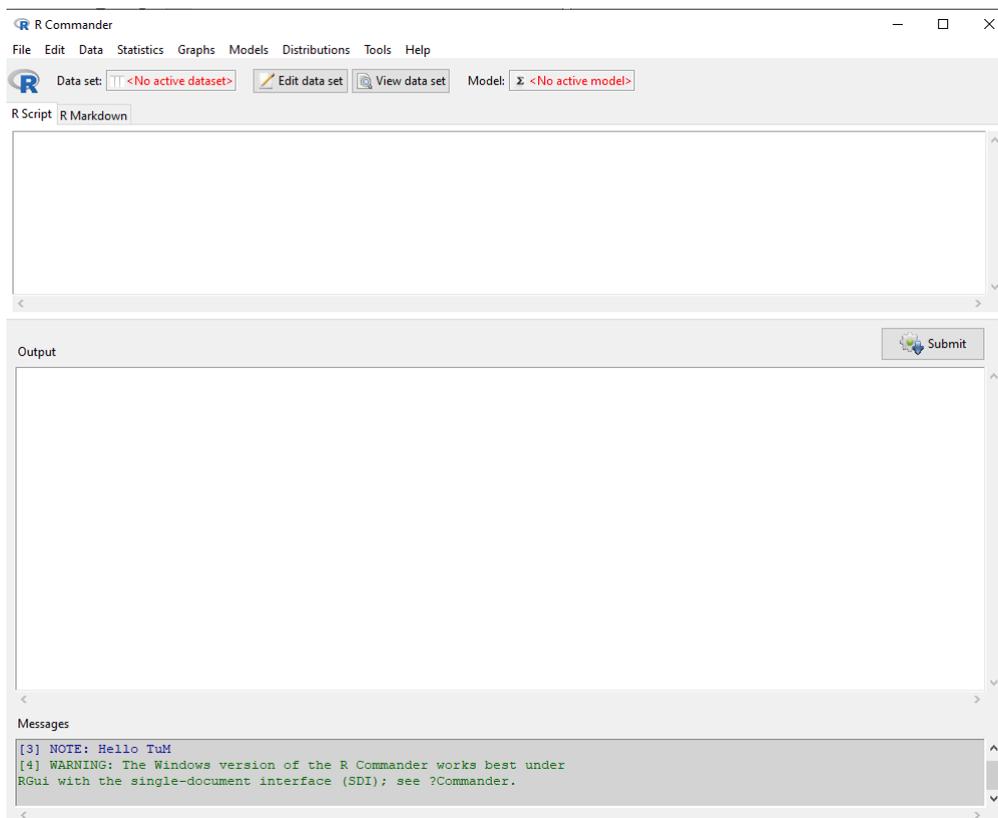
## 3. Install the R Commander Package (Rcmdr)



### 4. Load R Commander (Rcmdr)



### 5. Start Using R Commander



## II: Data Management

### 1.Importing Data into R Commander

#### CSV File:

1. Open **R Commander**.
2. In the top menu, go to **Data > Import Data > From Text File (base)**.
3. In the dialog box that appears, select the **CSV file** you want to import.
4. Make sure to choose the correct delimiter (usually a comma for CSV) and specify any other options (e.g., column names, header row).
5. Click **OK**, and the data will be imported into R Commander.

#### Excel File:

1. Go to **Data > Import Data > From Excel File**.
2. Select the **Excel file** from your computer.
3. Choose the sheet you want to import.
4. Click **OK**, and the data will appear in R Commander.

#### SPSS File:

1. Go to **Data > Import Data > From SPSS File**.
2. Browse to the SPSS file (.sav) on your computer.
3. Click **Open**, and the data will be loaded into R Commander.

### 2. Data Cleaning Techniques in R Commander

#### Recoding Variables:

1. Go to **Data > Manage Variables > Recode Variables**.
2. In the dialog box, choose the variable to recode.
3. Set the **old values** and **new values** for recoding (e.g., converting a numeric variable to a categorical one).
4. Click **OK**, and the recoded variable will be created.

#### Handling Missing Values:

1. Go to **Data > Manage Cases > Delete Cases with Missing Data** to remove rows with missing values.
  - You can choose which variables to check for missing values.

OR

2. If you prefer to **replace missing values** with a specific value (e.g., the mean or median), go to **Data > Manage Variables > Replace Missing Values**.
3. Choose the variable and the method to replace the missing data (e.g., replace with mean, median, or a constant value).
4. Click **OK**.

### Filtering and Selecting Subsets of Data:

1. To **filter** rows based on a condition, go to **Data > Select Cases**.
2. In the dialog, define your condition (e.g., `age > 30` or `income < 50000`).
3. Click **OK**, and R Commander will only show the cases that meet the condition.

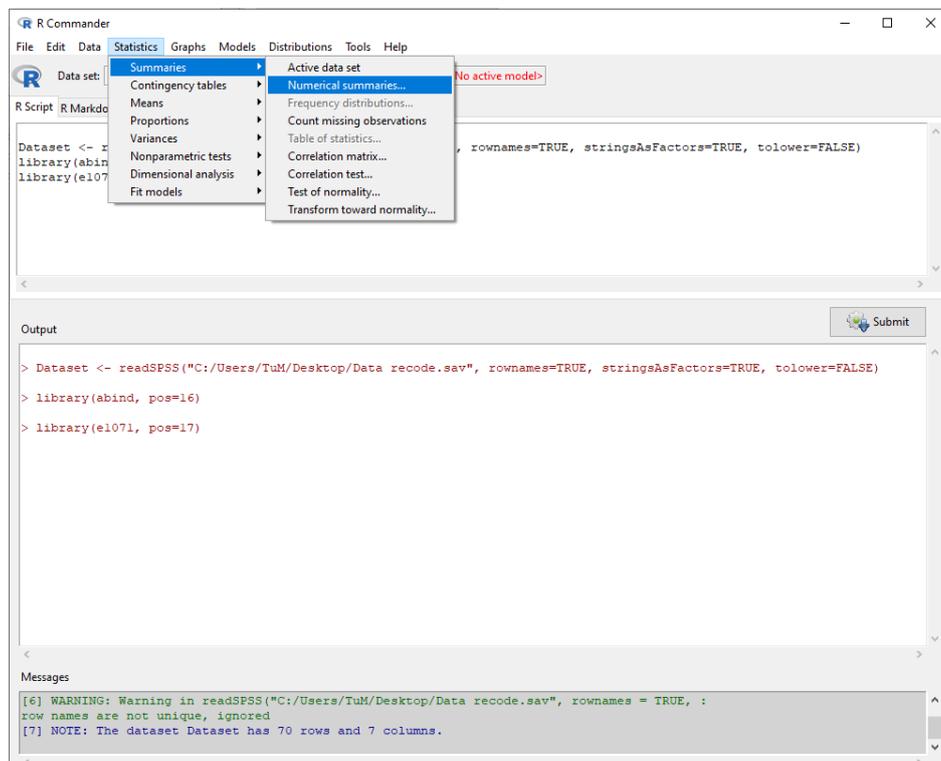
OR

4. To **select specific variables**, go to **Data > Active Data Set > Subset Active Data Set**.
5. In the dialog, you can specify which columns or rows you want to keep.

## III: Basic Descriptive Statistics

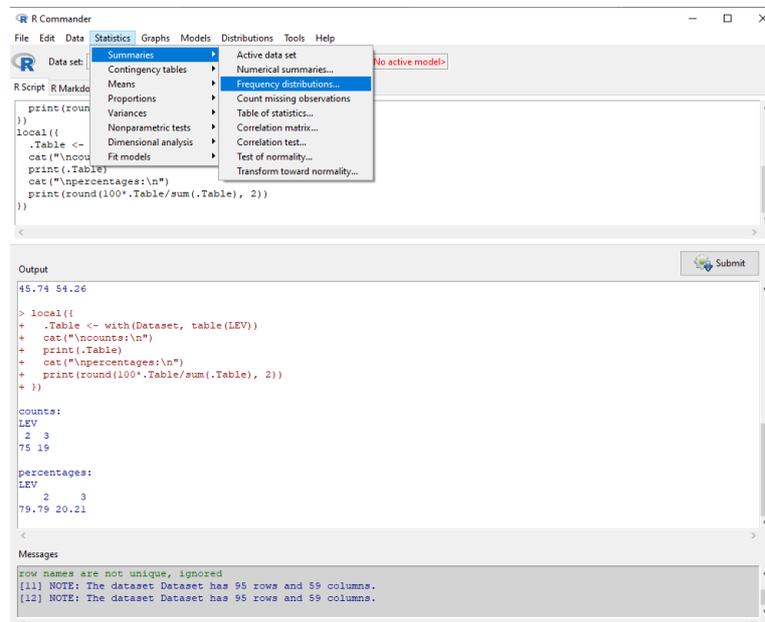
### 1. Computing Summary Statistics (Mean, Median, Standard Deviation, etc.)

1. **Open R Commander.**
2. Go to **Statistics > Summaries > Numerical Summaries**.
3. Select the variable (s) you want to analyze.
4. Choose the summary statistics you wish to compute (e.g., **mean, median, standard deviation, minimum, maximum**).
5. Click **OK**, and R Commander will display the summary statistics in the output window.
  - You can also generate **graphical summaries** like boxplots and histograms for a visual overview by going to **Graphs > Boxplot** or **Histogram**.



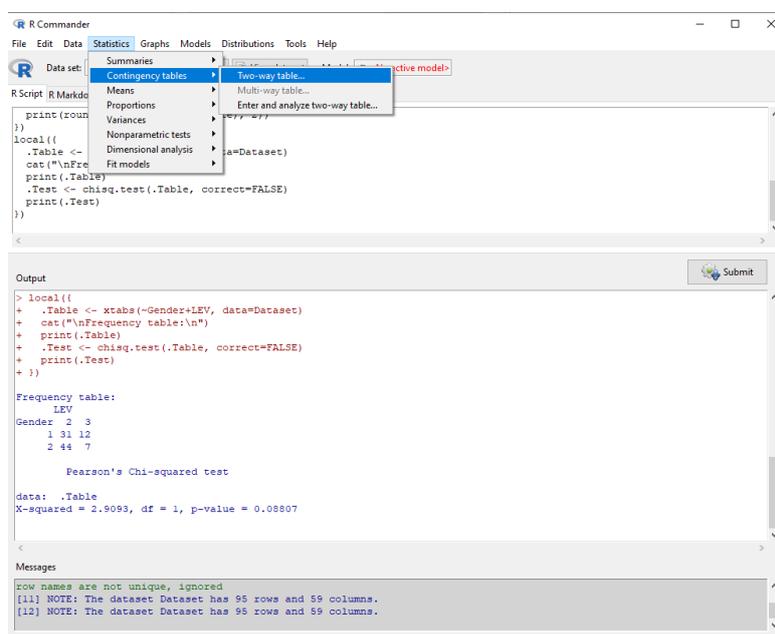
## 2. Creating Frequency Tables for Categorical Variables

1. Go to **Statistics > Summaries > Frequency Distributions**.
2. Choose the categorical variable for which you want to create a frequency table.
3. Click **OK**, and a frequency table will be displayed in the output window.
4. This table shows the count and percentage of each category within the variable.



## 3. Creating Cross-Tabulations (Contingency Tables)

1. Go to **Statistics > Contingency Tables > Two-Way Table**.
2. Select the variables for the **rows** and **columns** of the table.
3. Check the options for additional statistics such as **row percentages, column percentages, and chi-square test** if needed.
4. Click **OK**, and R Commander will display the cross-tabulation, showing the distribution of one variable across the categories of another.



**IV: Basic Statistical Test****1. Conducting Common Statistical Tests in R Commander****a. t-tests****1. Independent Samples T-test:**

- Go to **Statistics > Means > Independent samples t-test.**
- Select the dependent variable (numerical) and the grouping variable (categorical).
- Specify if the variances are assumed to be equal or unequal.
- Click **OK** to run the test.

**Output 1**

```
Welch Two Sample t-test

data: MeanAll by gender
t = -0.019255, df = 372.61, p-value = 0.9846
alternative hypothesis: true difference in means between group ชาย and group หญิง is not equal to 0
95 percent confidence interval:
-0.09027777  0.08852688
sample estimates:
mean in group ชาย mean in group หญิง
      3.530397      3.531272
```

**2. Paired Samples T-test:**

- Go to **Statistics > Means > Paired t-test.**
- Select the two paired variables.
- Click **OK** to run the test.

**Output 2**

```
Paired t-test

data: Meanbefore and Meanafter
t = -5.1434, df = 399, p-value = 0.0000004238
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
-0.1378931 -0.0616307
sample estimates:
mean difference
      -0.0997619
```

**b. ANOVA (Analysis of Variance)**

1. Go to **Statistics > Means > One-way ANOVA.**
2. Select the response variable (numerical) and the factor (categorical).
3. Click **OK.**

### Output 3

```
AnovaModel.13 <- aov(MeanAll ~ q8, data=Dataset)

summary(AnovaModel.13)
      Df Sum Sq Mean Sq F value Pr(>F)
q8      3  2.52  0.8400  4.267 0.00555 **
Residuals 396  77.96  0.1969
--
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

with(Dataset, numSummary(MeanAll, groups=q8, statistics=c("mean", "sd")))
      mean      sd data:n
-3ครั้ง      3.461186 0.3601361    109
-4ครั้ง      3.393491 0.4112771     26
ครั้งแรก      3.607089 0.4551032    204
มากกว่า5ครั้ง 3.459016 0.5428788     61
```

#### b.1 Multiple Comparison Test (Post-Hoc Test)

1. **Run ANOVA:**
  - Go to **Statistics > Means > One-way ANOVA**.
  - Select the response (numerical) variable and the factor (categorical) variable.
  - Click **OK** to execute the ANOVA.
2. **Access Multiple Comparisons:**
  - Once the ANOVA is completed and shows a significant result ( $p < 0.05$ ), proceed to post-hoc tests.
  - Go to **Statistics > Means > Multiple comparisons**.
3. **Choose the Post-Hoc Test:**
  - Select the type of multiple comparison test you wish to perform (e.g., **Tukey's HSD**).
  - Ensure the response variable and the factor are selected correctly.
4. **Set Options:**
  - You may be prompted to specify confidence levels (commonly 95%).
  - Click **OK** to run the test.

### Output 4

```
Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts

Fit: aov(formula = MeanAll ~ q8, data = Dataset)

Linear Hypotheses:
              Estimate Std. Error t value Pr(>|t|)
3-4ครั้ง - 2-3ครั้ง == 0   -0.067694  0.096843  -0.699  0.8925
ครั้งแรก - 2-3ครั้ง == 0    0.145903  0.052643   2.772  0.0279 *
มากกว่า5ครั้ง - 2-3ครั้ง == 0 -0.002169  0.070949  -0.031  1.0000
ครั้งแรก - 3-4ครั้ง == 0    0.213598  0.092398   2.312  0.0913 .
มากกว่า5ครั้ง - 3-4ครั้ง == 0  0.065525  0.103922   0.631  0.9183
มากกว่า5ครั้ง - ครั้งแรก == 0 -0.148073  0.064751  -2.287  0.0966 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)
```

```

Simultaneous Confidence Intervals

multiple Comparisons of Means: Tukey Contrasts

Fit: aov(formula = MeanAll ~ q8, data = Dataset)

Quantile = 2.5539
95% family-wise confidence level

Linear Hypotheses:

```

	Estimate	lwr	upr
3-4ครั้ง - 2-3ครั้ง == 0	-0.067694	-0.315024	0.179635
ครั้งแรก - 2-3ครั้ง == 0	0.145903	0.011456	0.280351
มากกว่า5ครั้ง - 2-3ครั้ง == 0	-0.002169	-0.183368	0.179029
ครั้งแรก - 3-4ครั้ง == 0	0.213598	-0.022380	0.449575
มากกว่า5ครั้ง - 3-4ครั้ง == 0	0.065525	-0.199884	0.330935
มากกว่า5ครั้ง - ครั้งแรก == 0	-0.148073	-0.313440	0.017295

```

      2-3ครั้ง      3-4ครั้ง      ครั้งแรก มากกว่า5ครั้ง
      "a"         "ab"         "b"         "ab"

```

```

One-way analysis of means (not assuming equal variances)

data: MeanAll and q8
F = 4.4695, num df = 3.000, denom df = 93.935, p-value = 0.005588

```

### C. Correlation

1. Go to **Statistics > Summaries > Correlation matrix**.
2. Select the variables to include in the correlation analysis.
3. Choose the correlation method (e.g., Pearson or Spearman).
4. Click **OK**.

### Output 5

```

Pearson's product-moment correlation

data: MeanA and MeanAll
F = 34.403, df = 398, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8381141 0.8878202
sample estimates:
      cor
0.8650752

```

```
> cor(Dataset[,c("MeanA", "MeanAll", "MeanB", "MeanC", "MeanE")], use="complete")
      MeanA  MeanAll  MeanB  MeanC  MeanE
MeanA  1.0000000  0.8650752  0.5971402  0.6892052  0.6206054
MeanAll 0.8650752  1.0000000  0.7862835  0.8493444  0.8313798
MeanB  0.5971402  0.7862835  1.0000000  0.5112419  0.4807644
MeanC  0.6892052  0.8493444  0.5112419  1.0000000  0.6553221
MeanE  0.6206054  0.8313798  0.4807644  0.6553221  1.0000000
```

## V: Regression Analysis

### Running Simple and Multiple Linear Regression in R Commander

#### 1. Load Your Data:

- Go to **Data > Import data** to load your dataset into R Commander if you haven't done so already.

#### 2. Run Simple Linear Regression:

- Navigate to **Statistics > Fit models > Linear regression**.
- Select your dependent (response) variable and the independent (predictor) variable.
- Click **OK** to run the model.

#### 3. Run Multiple Linear Regression:

- Follow the same steps as above but select multiple predictor variables for the model.
- Ensure that your independent variables are correctly chosen and entered.

### Output 6

```
lm(formula = MeanSocialSkills ~ Age + Delayed + GPA + MeanAttachment +
    MeanEmpathy + SignLanguage, data = Dataset)

Residuals:
    Min       1Q   Median       3Q      Max
-0.95807 -0.15574 -0.00306  0.15799  0.94231

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.06782    0.27517   7.515 2.22e-12 ***
Age          -0.00579    0.01176  -0.492  0.6230
Delayed      -0.32091    0.05013  -6.402 1.18e-09 ***
GPA          -0.01673    0.04572  -0.366  0.7149
MeanAttachment 0.22266    0.04382   5.081 8.96e-07 ***
MeanEmpathy  0.19944    0.04470   4.462 1.39e-05 ***
SignLanguage  0.02450    0.01168   2.097  0.0373 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2825 on 189 degrees of freedom
(4 observations deleted due to missingness)
Multiple R-squared:  0.4599, Adjusted R-squared:  0.4428
F-statistic: 26.83 on 6 and 189 DF, p-value: < 2.2e-16
```

**Diagnostics:**

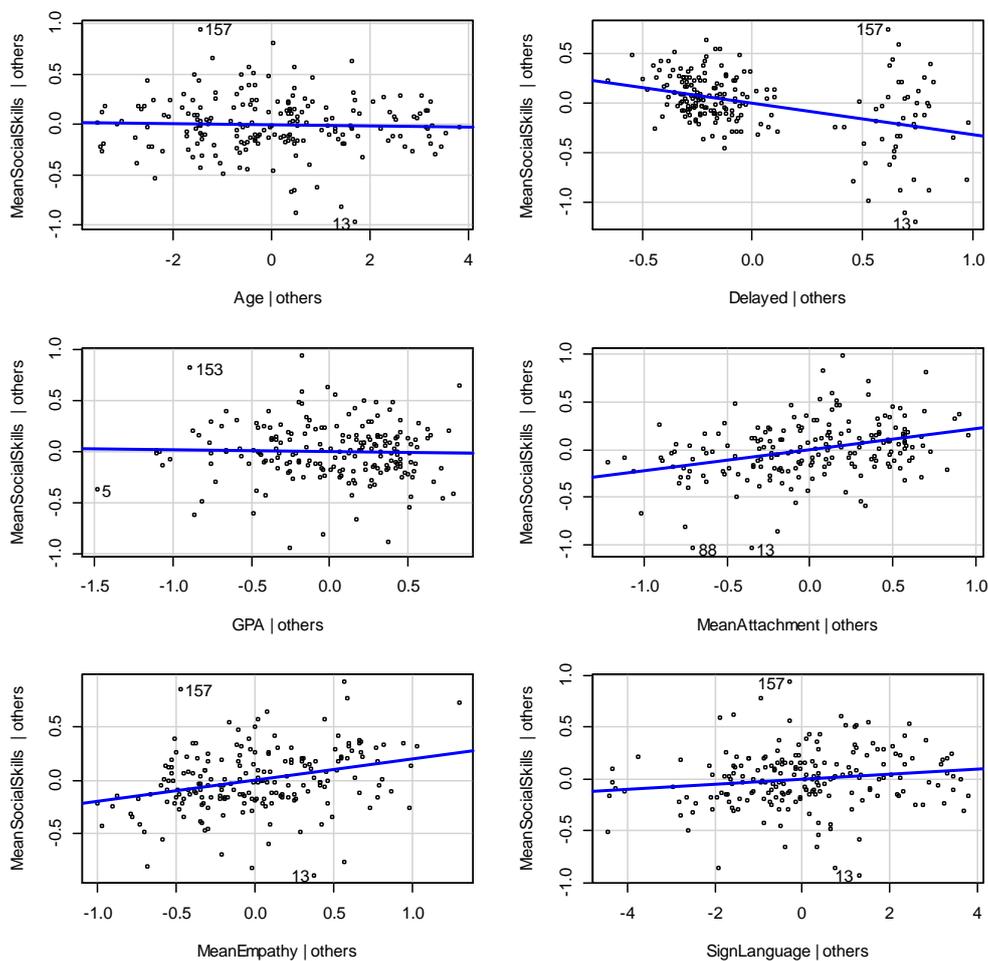
- **R-squared:**
  - This metric shows the proportion of the variance in the dependent variable explained by the predictors. A higher R-squared indicates a better fit.
- **Residual Standard Error:**
  - Reflects the average distance between the actual data points and the fitted line.
- **p-values for Coefficients:**
  - Check if coefficients are significantly different from zero (common threshold:  $p < 0.05$ ).
- **Residuals:**
  - Examine the residuals to check for patterns; ideally, they should be randomly scattered with no obvious trend.
- **F-statistic:**
  - Indicates whether the overall model is significant.

**Multiple Regression Assumptions**

**1.Linearity**

**Output 7**

Added-Variable Plots



## 2.Independence of Errors (No Autocorrelation)

### Output 8

```
Durbin-Watson test

lata: MeanSocialSkills ~ Age + Delayed + GPA + MeanAttachment + MeanEmpathy + SignLanguage
DW = 2.0845, p-value = 0.7045
alternative hypothesis: true autocorrelation is greater than 0
```

## 3.Homoscedasticity (Constant Variance of Errors)

### Output 9

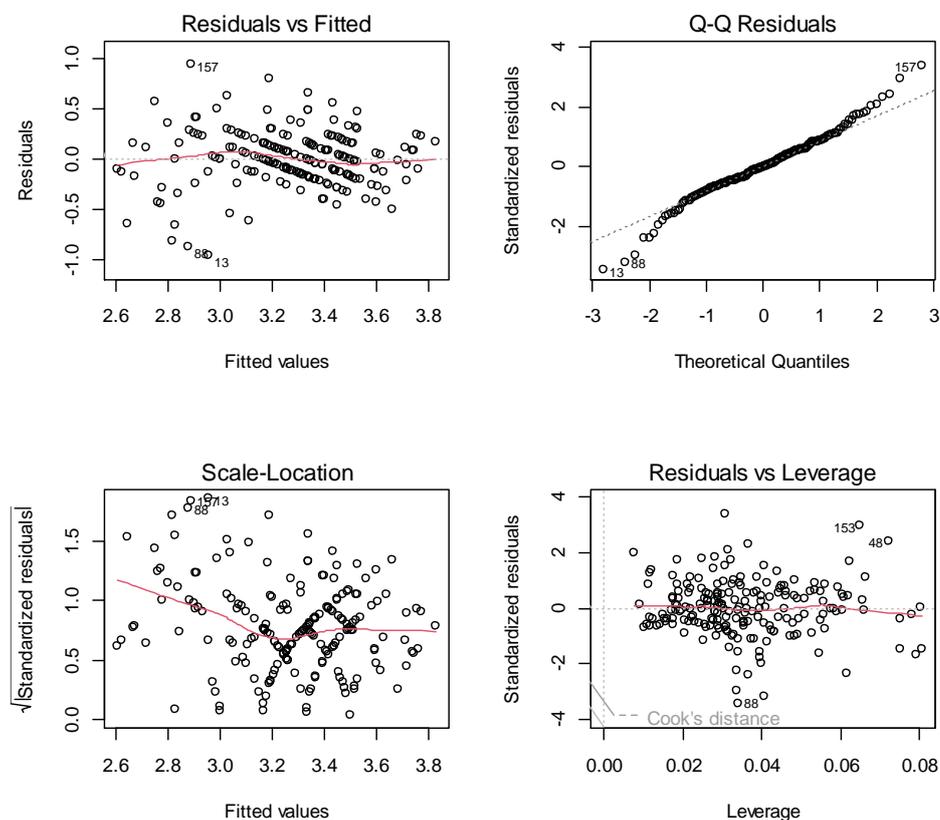
```
Breusch-Pagan test

lata: MeanSocialSkills ~ Age + Delayed + GPA + MeanAttachment + MeanEmpathy + SignLanguage
BP = 28.4, df = 1, p-value = 0.00000009864
```

## 4.Normality of Errors

### Output 10

eanSocialSkills ~ Age + Delayed + GPA + MeanAttachment + MeanEmpathy



## 5.No Multicollinearity

### Output 11

```
> vIF(RegModel.1)
      Age      Delayed      GPA MeanAttachment      MeanEmpathy      SignLanguage
1.019975  1.124471  1.086001  1.209832  1.232797  1.144125

> round(cov2cor(vcov(RegModel.1)), 3) # Correlations of parameter estimates
      (Intercept)      Age Delayed      GPA MeanAttachment      MeanEmpathy      SignLanguage
(Intercept)      1.000 -0.529 -0.259 -0.541      -0.342      -0.422      -0.203
Age              -0.529  1.000  0.056 -0.065      0.057      0.061      0.066
Delayed          -0.259  0.056  1.000  0.011      0.133      0.110      0.257
GPA              -0.541 -0.065  0.011  1.000      -0.011      0.138      -0.214
MeanAttachment   -0.342  0.057  0.133 -0.011      1.000      -0.366      -0.041
MeanEmpathy      -0.422  0.061  0.110  0.138      -0.366      1.000      0.098
SignLanguage     -0.203  0.066  0.257 -0.214      -0.041      0.098      1.000
```

## 6.No Significant Outliers or Influential Points

### Output 12

```
> outlierTest(RegModel.1)
No Studentized residuals with Bonferroni p < 0.05
Largest |rstudent|:
      rstudent unadjusted p-value Bonferroni p
13 -3.554843      0.0004784      0.093767
```

## Standardizing Variables for Beta Coefficients in R Commander:

R Commander itself doesn't directly output Beta coefficients from a regression, but you can calculate them by standardizing the variables first or using specific packages.

### Method 1: Manually Standardizing Variables:

1. **Standardize Your Variables:**
  - Go to **Data > Manage variables in active data set > Standardize variables.**
  - Select the independent and dependent variables you want to standardize (e.g., creating  $z_x$  and  $z_y$  for predictors and response).
2. **Run Regression with Standardized Variables:**
  - Go to **Statistics > Fit models > Linear regression.**
  - Select the standardized dependent variable ( $z_y$ ) as the response and standardized predictors (e.g.,  $z_{x1}$ ,  $z_{x2}$ , etc.) as the independent variables.
  - Click **OK** to run the regression. The coefficients obtained will now be Beta coefficients.

**Output 13**

```

Call:
lm(formula = Z.MeanSocialSkills ~ Z.Age + Z.Delayed + Z.GPA +
    Z.MeanAttachment + Z.MeanEmpathy + Z.SignLanguage, data = Dataset)

Residuals:
    Min       1Q   Median       3Q      Max
-2.54763 -0.41414 -0.00814  0.42012  2.50574

Coefficients:
              Estimate Std. Error t value    Pr(>|t|)
(Intercept)  -0.01436    0.05367   -0.268    0.7893
Z.Age         -0.02669    0.05421   -0.492    0.6230
Z.Delayed     -0.36536    0.05707  -6.402 0.00000000118 ***
Z.GPA         -0.02051    0.05607   -0.366    0.7149
Z.MeanAttachment  0.30044    0.05912   5.081 0.000000089583 ***
Z.MeanEmpathy  0.26669    0.05977   4.462 0.00001392260 ***
Z.SignLanguage  0.12054    0.05748   2.097    0.0373 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.7513 on 189 degrees of freedom
(4 observations deleted due to missingness)
Multiple R-squared:  0.4599, Adjusted R-squared:  0.4428
F-statistic: 26.83 on 6 and 189 DF,  p-value: < 2.2e-16

```

**Method 2: Using the lm.beta Package in R:**

1. **Run the Regression in R:**
  - Run your regression model in R Commander or through R code, such as

```
lm_model <- lm(Y ~ X1 + X2, data = dataset).
```
2. **Calculate Beta Coefficients:**
  - Use the `lm.beta` package:
    - Open the **R Script** window in R Commander and type:

```
library(lm.beta)
model <- lm(Y ~ X1 + X2, data = dataset)
beta_model <- lm.beta(model)
summary(beta_model)
```
    - Execute the code to display the Beta coefficients in the **Console**.

**Output 14**

```

Call:
lm(formula = MeanSocialSkills ~ Age + Delayed + GPA + MeanAttachment +
    MeanEmpathy + SignLanguage, data = Dataset)

Residuals:
    Min       1Q   Median       3Q      Max
-0.95807 -0.15574 -0.00306  0.15799  0.94231

Coefficients:
              Estimate Standardized Std. Error t value Pr(>|t|)
(Intercept)   2.06782             NA    0.27517   7.515 2.22e-12 ***
Age           -0.00579           -0.02658    0.01176  -0.492  0.6230
Delayed       -0.32091           -0.36291    0.05013  -6.402 1.18e-09 ***
GPA           -0.01673           -0.02038    0.04572  -0.366  0.7149
MeanAttachment 0.22266            0.29878    0.04382   5.081 8.96e-07 ***
MeanEmpathy   0.19944            0.26483    0.04470   4.462 1.39e-05 ***
SignLanguage  0.02450            0.11991    0.01168   2.097  0.0373 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2825 on 189 degrees of freedom
(4 observations deleted due to missingness)
Multiple R-squared:  0.4599,    Adjusted R-squared:  0.4428
F-statistic: 26.83 on 6 and 189 DF,  p-value: < 2.2e-16

```

**Resources****Books:**

- **"R for Data Science" by Hadley Wickham and Garrett Grolemund** – While this book focuses on R in general, it can provide a great foundation before diving into GUI tools like R Commander.
- **"An Introduction to R" by W. N. Venables and D. M. Smith** – This foundational text includes basics that complement the use of R Commander for statistical analysis.
- **"Using R for Introductory Statistics" by John Verzani** – This book has sections related to using R Commander, particularly for basic statistical analysis.

**Official Resources:**

- **R Commander CRAN Page** – Rcmdr package on CRAN. This page includes installation details, dependencies, and links to further documentation.
- **John Fox's R Commander Manual** – The creator of R Commander, John Fox, has documentation that can be helpful for learning how to use it effectively. Check out his website or related manuals.

**Online Tutorial:**

- **Quick-R by DataCamp** – This site often provides guides for using R and can include tips on working with R Commander for beginners