Conducting and Presenting Multiple Linear Regression Analysis Using R

Maria Paula Rojas Conejo 400

Fresenius University of Applied Science

Data Science and Data Analytics (WS 2025/26)

Prof. Dr. Stephan Huber

2026-12-10

Author Note

Correspondence concerning this article should be addressed to Maria Paula Rojas Conejo 400, Email: rojas_conejo.maria@stud.hs-fresenius.de

Abstract

This document outlines the requirements for successfully contributing to the course through a presentation and a project report. It introduces the process, establishes the standards, and provides tips on how students can effectively meet the expectations.

Conducting and Presenting Multiple Linear Regression Analysis Using R

1 Introduction

According to Alexopoulos (2010) when it comes to data analysis one of the main goals is to extract from raw information the accurate estimation. One of the common questions in data analysis is the statistical relationship between a response variable (Y) and explanatory variables (Xi). Alexopoulos (2010) continues to explain that an option to answer this question is to employ regression analysis in order to model its relationship.

In this project we examine the fundamental principles of multiple linear regression, a widely used statistical method as the author mention. At the same time it is intended to clarify how the method works and when it is appropriate to use.

2 What is Multiple Linear Regression?

The author Uyanık and Güler (2013) explains that regression analysis is used to determine the relationships between two or more variables that exhibit cause–effect patterns, and to make predictions based on those relationships. Similarly, Marill (2004) states that multiple linear regression is a generalization of simple linear regression in which more than one predictor variable is included. The formula is the following:

$$Y_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \cdots + b_n X_{ni} + u_i$$

Where:

- Y_i — dependent variable
- b_0 — intercept
- $b_1 \ldots b_n$ — regression coefficients
- $X_{1i} \ldots X_{ni}$ — independent variables
- u_i — disturbance error

2.1 Assupmtions

According to Ernst and Albers (2017) there are 4 assupmtions in the linear regression model, violating these assumptions can create several kinds of problems. First, the resulting estimates may be biased, meaning they do not,reflect the true population values. Second, the estimators might become inconsistent, so increasing the sample size no longer estimates will approach the true parameters. Third, may lose its efficiency, meaning it can yield less precise estimates. Fourth, hypothesis tests and confidence intervals may become unreliable. Note that these assumptions specifically apply to estimation using the Ordinary Least Squares (OLS) method (a statistical method used to estimate the parameters of a linear regression model by minimizing the sum of the squared differences between the observed and predicted values), which is the default in many statistical software programs.

1. Linearity: The model assumes a linear and additive relationship between each predictor and the mean of the dependent variable, with errors averaging zero for any combination of predictors. Problems arise if relationships are nonlinear or if measurement error is present. Multicollinearity is not an assumption, but low overlap between predictors is desirable. (Ernst & Albers, 2017)
2. Normality: The model errors are assumed to follow a normal distribution centered at zero. (Ernst & Albers, 2017)
3. Homoscedasticity: The errors are expected to have constant variance across all values of the predictors. If the spread of errors changes (heteroscedasticity), the assumption is violated. (Ernst & Albers, 2017)
4. Independence: Errors must be independent of each other, meaning their covariances are zero. This requires proper random sampling. Residual plots or autocorrelation checks help detect issues. (Ernst & Albers, 2017)

2.2 Linear Model or lm() in R

The Linear Model is is used to fit linear models, including multivariate ones. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance. The basic sintax is the following:

Basic Syntax

```
model <- lm(formula, data = dataset)
```

3 Steps to do a multiple linear regression analysis

The author Zaghi (2020) explains step by step how to conduct a multiple linear regression.

1. Define the research question: Specify outcome and predictors based on theory and prior research.

2. Prepare and explore the data: Handle missing values, recode variables, explore distributions and correlations
3. Fit the initial model with lm()
4. Perform diagnostics: Use residual plots, Q-Q plots, and VIF to check assumptions and detect influential points.
5. Refine the model if necessary: Consider transformations, adding interaction terms, or simplifying the model when justified.
6. Summarize results with broom or similar tools: Generate tidy tables of coefficients and model fit indices.
7. Report and interpret results

3.1 Preparing Data and Fitting the Model in R

R uses a concise formula syntax and the function lm() for linear models. Before fitting a model, it is good practice to inspect the data, check for missing values, and explore relationships. Below we use the built-in mtcars dataset to illustrate an MLR where miles per gallon (mpg) is predicted by weight (wt), horsepower (hp), and number of cylinders (cyl). This is a simple example similar in spirit to many introductory regression tutorials in R

```r
# Load data
data(mtcars)
# Quick overview
str(mtcars)
summary(mtcars)
# Correlation matrix for numerical variables
cor(mtcars[, c("mpg", "wt", "hp", "cyl")])
# Simple scatterplot matrix
pairs(mtcars[, c("mpg", "wt", "hp", "cyl")])
```

3.2 Fitting a multiple linear regression with lm()

```r
# Fit the model
model <- lm(mpg ~ wt + hp + cyl, data = mtcars)
# Basic summary output
summary(model)
```

Output:

- Estimated coefficients (Estimate)
- Standard errors (Std. Error)
- t-values and p-values
- Residual standard error
- R^2 and adjusted R^2
- F-statistic for the overall model

At this stage, you have a working model, but it is not yet ready to report. Next you must check whether assumptions are reasonably satisfied.

3.3 Model Diagnostics in R

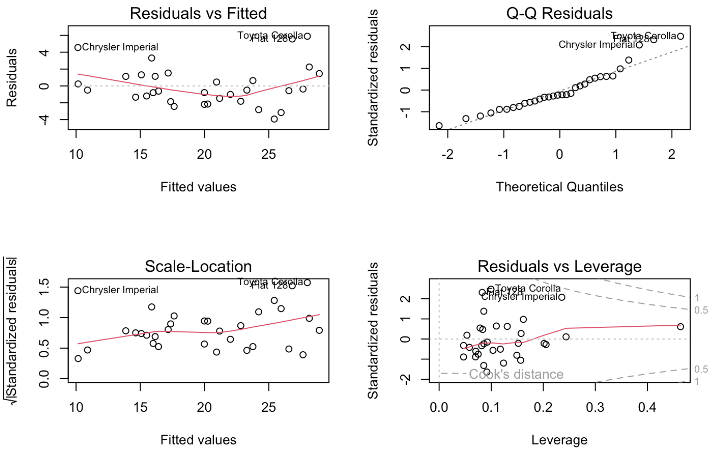
3.3.1 Residual plots and normality

```
# Basic diagnostic plots (4-panel layout)
par(mfrow = c(2, 2))
plot(model)
par(mfrow = c(1, 1))
```

Output

Figure 1

Rplot01



Key plots:

- Residuals vs Fitted – Should show a roughly random scatter (no strong curve or funnel pattern).
- Normal Q-Q – Points should follow the diagonal line if residuals are approximately normal.
- Scale-Location – Checks homoscedasticity (constant spread of residuals).
- Residuals vs Leverage / Cook's distance – Highlights influential observations.

3.3.2 Checking multicollinearity with VIF

Multicollinearity inflates standard errors and makes individual coefficients unstable even if the overall model is significant. The variance inflation factor (VIF) is a widely used measure; values above about 5–10 are often flagged as problematic.

```r
# install.packages("car")  # run once if not installed
library(car)
vif(model)
```

In this case the VIF for wt is 2.58, for hp is 3.26 and for cyl is 4.75

3.4 Presenting Multiple Regression Results

The broom package converts R's model objects (like lm) into tidy data frames, making it easy to create tables and plots of results.

```r
# install.packages("broom")  # run once
library(broom)
# Coefficient-level information
coef_table <- tidy(model, conf.int = TRUE)
coef_table
# Model-level summary (R^2, etc.)
model_fit <- glance(model)
model_fit
```

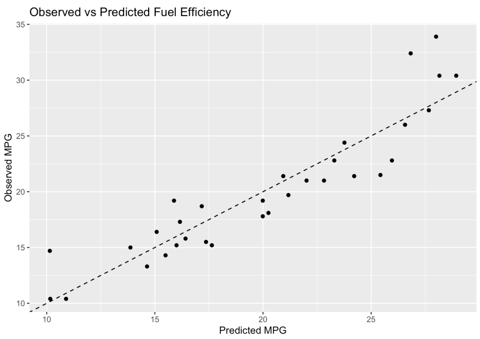
3.5 Visualizing and Communicating Predictions

```r
# Predictions for the observed data
mtcars$pred_mpg <- predict(model)
# Create a small new data set for scenario-based predictions
new_cars <- data.frame(
  wt  = c(2.5, 3.0, 3.5),  # in 1000 lb
  hp  = c(100, 150, 200),
  cyl = c(4, 6, 8)
)
predictions <- predict(model, newdata = new_cars, interval = "confidence")
cbind(new_cars, predictions)
# install.packages("ggplot2")  # run once
library(ggplot2)
ggplot(mtcars, aes(x = pred_mpg, y = mpg)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, linetype = "dashed") +
  labs(
    x = "Predicted MPG",
    y = "Observed MPG",
    title = "Observed vs Predicted Fuel Efficiency"
  )
```

Output:

Figure 2

Rplot02



4 References

Alexopoulos, E. C. (2010). Introduction to multivariate regression analysis. *Hippokratia*, *14*(Suppl 1), 23.

Ernst, A. F., & Albers, C. J. (2017). Regression assumptions in clinical psychology research practice—a systematic review of common misconceptions. *PeerJ*, *5*, e3323.

Marill, K. A. (2004). Advanced statistics: Linear regression, part II: Multiple linear regression. *Academic Emergency Medicine*, *11*(1), 94–102.

Uyanık, G. K., & Güler, N. (2013). A study on multiple linear regression analysis. *Procedia-Social and Behavioral Sciences*, *106*, 234–240.

Zaghi, M. (2020). *A step-by-step guide for running a complete multiple linear regression analysis in r*. https://medium.com/analytics-vidhya/a-step-by-step-guide-for-running-a-complete-multiple-linear-regression-analysis-in-r-c08be169fe01