Perform OLS Regression in R (With Example)

Authored by

Mohammed looti

October 28, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Perform OLS Regression in R (With Example)*. PSYCHOLOGICAL STATISTICS. Retrieved from https://statistics.arabpsychology.com/?p=4666

Ordinary least squares (OLS) regression is a fundamental statistical technique used to estimate the relationship between two or more variables. This method determines the line of best fit that minimizes the sum of the squared differences between the observed data points and the regression line. It is a powerful tool for understanding how changes in one or more predictor variables affect a response variable.

The OLS method enables us to derive the simple linear regression equation, which is expressed as:

? = b0 + b1x

where the components are defined as follows:

?: The estimated value of the response variable based on the model.
b0: The intercept of the regression line, representing the expected value of Y when X is zero.
b1: The slope of the regression line, indicating the change in Y for a one-unit change in X.

By establishing this equation, we gain insights into the quantitative relationship between the predictor and response variables. Furthermore, the model can be used for practical prediction--estimating the value of the response variable given a specific input value for the predictor variable. The following detailed, step-by-step example demonstrates how to execute and interpret OLS regression using the R programming language.

Step 1: Preparing the Dataset in R

To illustrate the OLS process, we will begin by constructing a synthetic dataset. This dataset will track the performance of 15 hypothetical students and includes two primary variables that we aim to relate statistically:

Total hours studied (our predictor variable, X)
Exam score (our response variable, Y)

Our objective is to perform OLS regression, treating the hours studied as the independent variable that predicts the resulting exam score. This analysis will help us quantify the impact of study time on academic performance. The code below shows how to define this data structure, known as a data frame, within the R environment:

```
#create dataset
df <- data.frame(hours=c(1, 2, 4, 5, 5, 6, 6, 7, 8, 10, 11, 11, 12, 12, 14),
score=c(64, 66, 76, 73, 74, 81, 83, 82, 80, 88, 84, 82, 91, 93, 89))

#view first six rows of dataset
```

head(df)

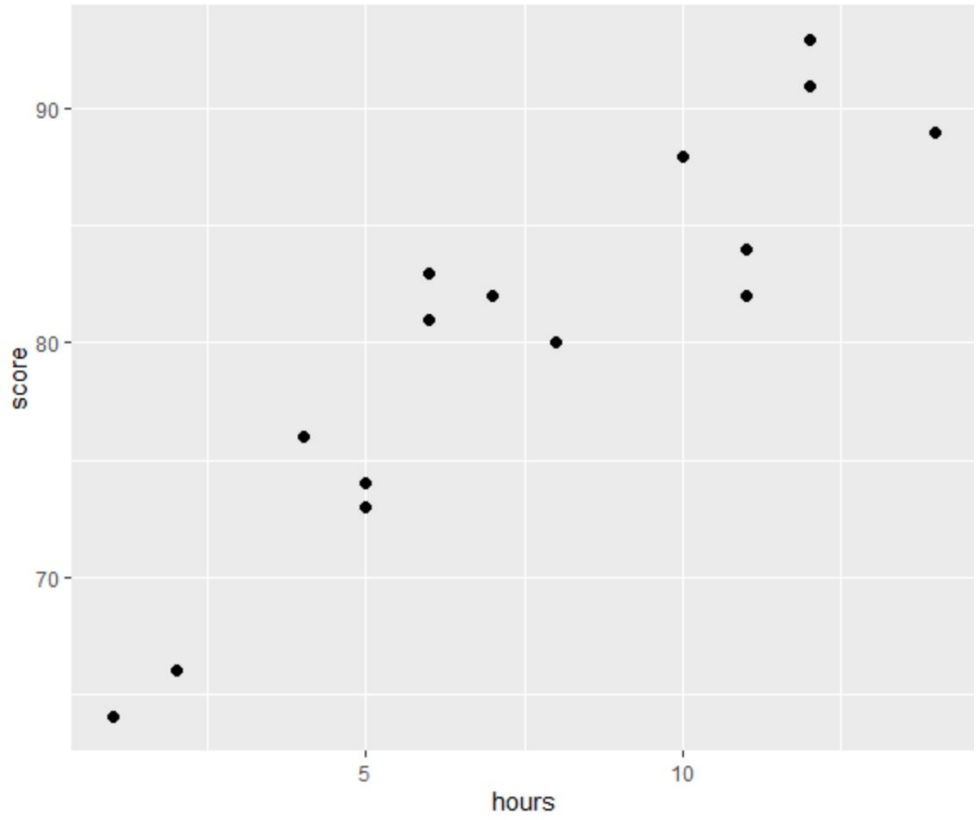
hours score
1 1 64
2 2 66
3 4 76
4 5 73
5 5 74
6 6 81

Step 2: Exploring the Data Through Visualization

Prior to fitting any statistical model, it is essential to visualize the data. Visualization allows us to confirm whether a linear relationship is plausible and helps us identify potential issues like non-linearity or extreme outliers. We begin by generating a scatter plot to visually assess the correlation between hours studied and exam score:

library(ggplot2)

#create scatter plot
ggplot(df, aes(x=hours, y=score)) +
geom_point(size=2)



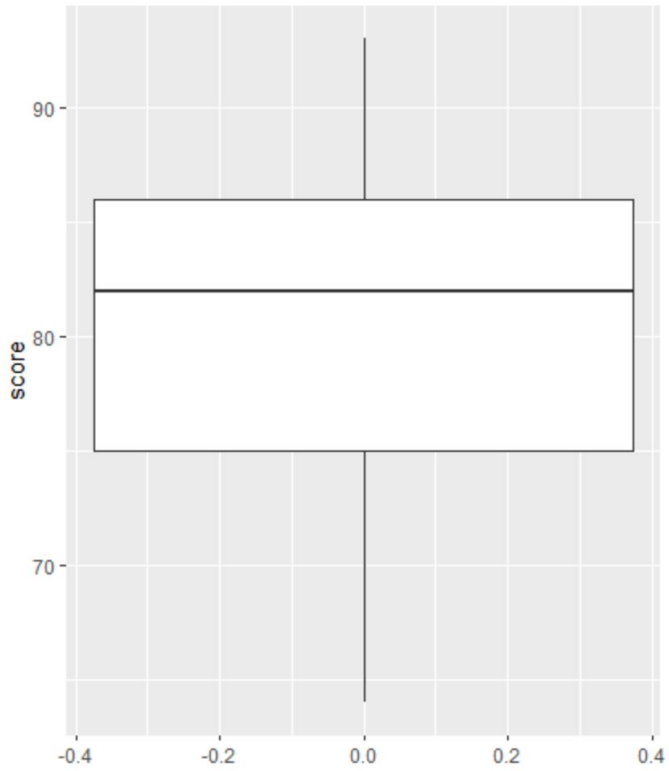
The resulting plot clearly suggests that the relationship is approximately linear. As the total number of hours studied increases, there is a distinct, positive trend showing a corresponding increase in the exam score. This initial visual confirmation supports the choice of a linear regression model.

Next, it is good practice to check the distribution of the response variable (score) and specifically screen for outliers, which can disproportionately influence the regression coefficients. We utilize a boxplot for this purpose.

Note on Outlier Detection in R: In standard boxplots generated by R, an observation is conventionally flagged as an outlier if it falls 1.5 times the interquartile range (IQR) above the third quartile (Q3) or 1.5 times the IQR below the first quartile (Q1). These points are typically marked by a small isolated circle on the plot.

library(ggplot2)

#create scatter plot
ggplot(df, aes(y=score)) +
geom_boxplot()



The boxplot confirms that no isolated circles are present, indicating that there are no significant outliers in the exam score distribution that would require special handling or transformation before running the OLS model.

Step 3: Executing OLS Regression and Interpreting Results

With the data prepared and inspected, we proceed to fit the simple linear regression model. In R, this is achieved using the built-in **lm()** function, where we define the relationship: score is modeled as a function of hours. The syntax for the model fitting and subsequent summary generation is shown below:

#fit simple linear regression model
model <- lm(score~hours, data=df)

#view model summary
summary(model)

Call:
lm(formula = score ~ hours)

Residuals:
Min 1Q Median 3Q Max

-5.140 -3.219 -1.193 2.816 5.772

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 65.334 2.106 31.023 1.41e-13 ***
hours 1.982 0.248 7.995 2.25e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.641 on 13 degrees of freedom
Multiple R-squared: 0.831, Adjusted R-squared: 0.818
F-statistic: 63.91 on 1 and 13 DF, p-value: 2.253e-06

Based on the calculated coefficients from the model summary, the finalized fitted regression equation is:

Score = 65.334 + 1.982 × (hours)

The coefficient for hours (**1.982**) is the slope, meaning that for every additional hour a student studies, the average expected exam score increases by **1.982** points. The intercept value (**65.334**) provides the baseline expectation: it represents the average expected exam score for a student who studies zero hours.

This equation can now be used for prediction. For instance, if we want to estimate the score for a student who studies for 10 hours, we substitute this value into the equation:

Score = 65.334 + 1.982 × (10) = 85.154. Therefore, the expected exam score is approximately **85.15**.

A thorough interpretation of the remaining key metrics in the model summary is essential for judging the model's quality and significance:

Pr(>|t|) (p-value): This value assesses the statistical significance of each coefficient. The p-value for *hours* (2.25e-06) is far below the conventional threshold of 0.05, leading us to conclude that there is a highly statistically significant association between the number of hours studied and the resulting exam score.
Multiple R-squared: This metric, often denoted as R^2, quantifies the proportion of the variance in the response variable that is predictable from the predictor variable(s). A larger R-squared indicates a better fit. In this case, **83.1%** of the variation observed in the exam scores can be successfully explained by the variation in the hours studied.
Residual standard error: This value represents the average distance that the observed data

points fall from the regression line. It serves as a measure of the model's accuracy. A lower value signifies a tighter fit. Here, the average observed exam score deviates by **3.641** points from the score predicted by the regression line.

F-statistic & p-value: These values assess the overall significance of the entire regression model. The F-statistic (**63.91**) and its corresponding p-value (**2.253e-06**) test the null hypothesis that all regression coefficients are zero. Since the p-value is extremely small (less than 0.05), we reject the null hypothesis, confirming that our model is statistically significant and that *hours* is a useful predictor for explaining the variation in *score*.

Step 4: Creating Residual Plots for Assumption Checks

The reliability of the OLS model hinges on meeting several key statistical assumptions, primarily concerning the residuals (the differences between observed and predicted values). We must ensure assumptions of Homoscedasticity and Normality are met to trust our coefficient estimates and p-values.

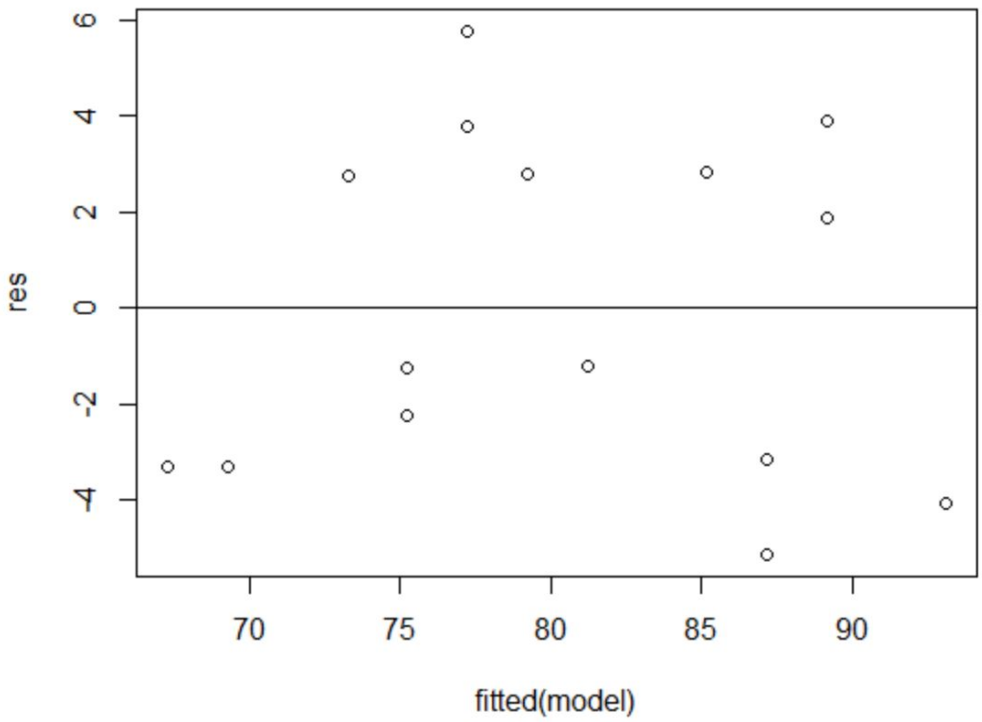
The assumption of **homoscedasticity** requires that the variance of the residuals remains constant across all levels of the predictor variable. Violation of this assumption (heteroscedasticity) can lead to inefficient and unreliable standard errors.

To check for homoscedasticity, we generate a **residuals vs. fitted plot**. The x-axis shows the fitted (predicted) values, and the y-axis shows the corresponding residuals. For the assumption to hold, the residuals must exhibit a random and even distribution around the value zero, showing no discernible patterns (like a cone or curve):

```
#define residuals
res <- resid(model)

#produce residual vs. fitted plot
plot(fitted(model), res)

#add a horizontal line at 0
abline(0,0)
```



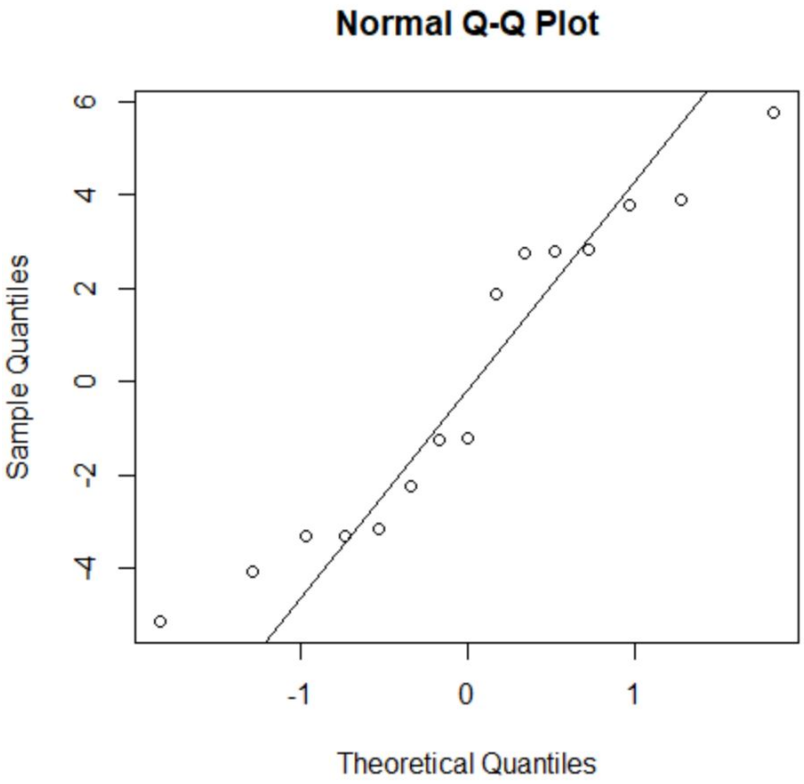
The plot confirms that the residuals are scattered randomly around the horizontal line at zero, without any obvious systematic pattern or fanning effect. We can therefore conclude that the assumption of homoscedasticity is met for this model.

The second crucial assumption, **normality**, dictates that the model's residuals should be approximately normally distributed. This is necessary for the validity of hypothesis tests (like the t-tests for coefficients).

We verify this assumption using a **Q-Q plot (Quantile-Quantile plot)**. If the data follows a normal distribution, the plotted points should closely adhere to a theoretical straight line drawn at a 45-degree angle:

```
#create Q-Q plot for residuals
qqnorm(res)

#add a straight diagonal line to the plot
qqline(res)
```



While the points exhibit slight deviations at the extremes, the majority of the data falls closely along the reference line. These minor deviations are typical for real-world data, and they are generally not severe enough to invalidate the normality assumption in this context.

Since both the normality and homoscedasticity assumptions for the residuals are reasonably satisfied, we have successfully verified the underlying conditions required for the OLS regression model. This means that the statistical output and conclusions derived from our model are robust and reliable.

A Note on Violations: If serious violations of these assumptions were detected, statistical methods such as transforming our data (e.g., using log transformations) or employing alternative regression models would be necessary to achieve a reliable fit.

Additional Resources for R Statistics

For those interested in expanding their proficiency in R, the following tutorials cover other common statistical and data manipulation tasks: