

How to Calculate Residual Sum of Squares in R

Authored by
stats writer

December 13, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Calculate Residual Sum of Squares in R*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=107379>

Understanding the Concept of Residuals in Regression

In the field of statistics and machine learning, the evaluation of a regression model hinges fundamentally on the concept of the **residual**. A residual serves as a measure of the error in prediction for a specific data point. It is defined as the vertical distance between the actual observed value (the dependent variable) and the value predicted by the fitted regression line or surface.

Quantifying this error is crucial for understanding model performance. The fundamental calculation for a residual is expressed simply as:

Residual = Observed value - Predicted value

If a residual is positive, the model underpredicted the observed value; if it is negative, the model overpredicted. The primary objective of fitting a model using the Ordinary Least Squares (OLS) method is to minimize the magnitude of these residuals collectively, thereby ensuring that the model provides the optimal fit to the underlying data distribution.

The Significance of the Residual Sum of Squares (RSS)

While individual residuals provide localized error information, the **Residual Sum of Squares (RSS)** is an aggregate metric that summarizes the total discrepancy between the observed data and the model's predictions across the entire dataset. It is arguably the most fundamental metric for determining a regression model's goodness-of-fit.

The mechanism of squaring each residual before summation serves two critical statistical purposes. Firstly, it prevents positive and negative errors from canceling each other out, which would artificially deflate the error metric. Secondly, squaring the errors imposes a heavier penalty on larger prediction errors, aligning the calculation with the goal of finding the line that minimizes these substantial deviations.

RSS represents the amount of variance in the dependent variable that remains **unexplained** by the independent variables included in the model. Consequently, when comparing two competing regression models fit to the same data, the model exhibiting the **lower RSS value** is preferred, as it signifies a smaller total prediction error and thus a tighter fit to the observations.

The Mathematical Formula for RSS

The formal definition of the Residual Sum of Squares is derived directly from the residuals of all N data points in the sample. If e_i represents the i^{th} residual, the RSS is the sum of the squares of these residuals.

The formula is written concisely as:

$$\text{Residual sum of squares} = \sum (e_i)^2$$

The key components of this formula are:

Σ : This Greek symbol signifies summation, indicating that we must aggregate the squared errors across all i observations.

e_i : Represents the i^{th} residual, calculated as the difference between the observed Y_i and the predicted \hat{Y}_i value.

It is important to remember that RSS is an absolute measure, meaning its magnitude depends heavily on the scale of the response variable. While it is excellent for internal comparison (comparing models on the same data), it cannot be used to compare the performance of models where the dependent variables have different units or orders of magnitude.

Calculating RSS in R: Essential Methods

The R programming language, being a powerhouse for statistical analysis, makes calculating the residual sum of squares exceptionally simple for any model fitted using the `lm()` function. Analysts typically rely on one of two highly efficient methods to extract this statistic directly from the model object.

Before proceeding with the calculation, we first must fit our desired regression model. Assuming we have a dataframe `df` with a response variable `y` and predictors `x1`, `x2`, etc., the process begins:

Build the regression model using the `lm()` function

```
model <- lm(y ~ x1 + x2 + ..., data = df)
```

Once the model is successfully created, the RSS can be extracted using either a specialized function that retrieves the deviance of a linear model (Method 1) or by manually calculating the sum of the squared errors (Method 2). Both methods are computationally fast and return identical results.

The two canonical ways to calculate RSS in R are demonstrated below:

Calculate residual sum of squares (Method 1: Using `deviance()`)

```
deviance(model)
```

Calculate residual sum of squares (Method 2: Summing squared residuals)

```
sum(resid(model)^2)
```

These two simple commands offer analysts powerful, immediate insight into the fitting efficiency of their statistical model.

Practical Example: Modeling Fuel Efficiency with mtcars

To demonstrate the application of these methods, we will utilize R's built-in **mtcars** dataset. We aim to construct a multiple linear regression model predicting miles per gallon (**mpg**) based on two predictors: vehicle weight (**wt**) and engine horsepower (**hp**).

First, we load and inspect the dataset to confirm the variables we will be using in our model specification:

```
# View the first six rows of the mtcars dataset
```

```
head(mtcars)
```

```
mpg cyl disp hp drat wt  qsec vs am gear carb
Mazda RX4 21.0 6 160 110 3.90 2.620 16.46 0 1 4 4
Mazda RX4 Wag 21.0 6 160 110 3.90 2.875 17.02 0 1 4 4
Datsun 710 22.8 4 108 93 3.85 2.320 18.61 1 1 4 1
Hornet 4 Drive 21.4 6 258 110 3.08 3.215 19.44 1 0 3 1
Hornet Sportabout 18.7 8 360 175 3.15 3.440 17.02 0 0 3 2
Valiant 18.1 6 225 105 2.76 3.460 20.22 1 0 3 1
```

The following code block executes the model fitting and then calculates the Residual Sum of Squares using both established methodologies.

Method 1: Using the deviance() Function in R

The `deviance()` function is a generic function in R. When applied to an object of class `lm` (a linear model), it automatically returns the Residual Sum of Squares. This method is the most commonly used due to its brevity and clarity.

We first define our model (`model`) where `mpg` is predicted by `wt` and `hp`, and then pass this object to `deviance()`:

```
# Build the multiple linear regression model
```

```
model <- lm(mpg ~ wt + hp, data = mtcars)
```

```
# Calculate residual sum of squares (method 1)
```

```
deviance(model)
```

195.0478

The resulting RSS value, 195.0478, represents the total measure of prediction error accumulated across all 32 cars in the dataset for this specific model configuration.

Method 2: Summing Squared Residuals Manually

The second method involves explicitly extracting the individual residuals and performing the mathematical summation step-by-step. The `resid()` function extracts the vector of errors (`$e_i`), which we then square (2) and sum (`sum()`).

While more verbose, this method directly reflects the mathematical definition of the Residual Sum of Squares, providing verification and transparency regarding the calculation.

```
# Calculate residual sum of squares (method 2)
```

```
sum(resid(model)^2)
```

195.0478

Confirming the previous result, this method also yields an RSS of 195.0478, demonstrating that both approaches are reliable means of calculating this critical metric in R.

Model Comparison Using RSS and R-squared

One of the most powerful uses of RSS is in model selection. We can compare two competing multiple linear regression models that aim to predict `mpg` but use different predictor variables. Let Model 1 remain `mpg ~ wt + hp`, and define Model 2 as `mpg ~ wt + disp` (using displacement instead of horsepower).

We calculate the RSS for both models to see which one leaves less unexplained variance in the response variable:

```
# Build two different models
```

```
model1 <- lm(mpg ~ wt + hp, data = mtcars)
```

```
model2 <- lm(mpg ~ wt + disp, data = mtcars)
```

```
# Calculate residual sum of squares for both models
```

```
deviance(model1)
```

195.0478

```
deviance(model2)
```

```
246.6825
```

The comparison reveals that Model 1 (RSS = 195.05) is superior to Model 2 (RSS = 246.68). Since the total prediction error for Model 1 is lower, it provides a better overall fit to the relationship between the predictors and miles per gallon.

We can validate this finding by calculating the standardized metric of fit, **R-squared**, which measures the proportion of the variance in the dependent variable that is predictable from the independent variables. Higher R-squared corresponds directly to lower RSS.

```
# Calculate R-squared for both models
```

```
summary(model1)$r.squared
```

```
0.8267855
```

```
summary(model2)$r.squared
```

```
0.7809306
```

The R-squared for Model 1 is 0.8268, which is higher than Model 2's 0.7809. This confirms that Model 1 is able to explain significantly more of the variance in fuel efficiency than Model 2, thereby validating the conclusion drawn from the comparison of the Residual Sum of Squares values.

Conclusion: Utilizing RSS for Robust Model Validation

The Residual Sum of Squares is more than just an intermediate statistic; it is a critical diagnostic tool essential for robust regression analysis. It provides a straightforward, powerful method for gauging model fit and, crucially, for making informed choices between competing statistical hypotheses.

By mastering the simple functions provided by the base R programming language--specifically `deviance()` and `sum(resid()^2)`--analysts can quickly obtain this metric and ensure they select the model that minimizes prediction errors, ultimately leading to more reliable and accurate forecasting.