

# How to Calculate Odds Ratios from a Logistic Regression Model in R

Authored by  
**stats writer**

November 19, 2025

## RECOMMENDED CITATION

stats writer (2025). *How to Calculate Odds Ratios from a Logistic Regression Model in R*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=97148>

The Odds ratios (OR) in logistic regression serve as a critical measure of association, providing a clear comparison between the odds of an event occurring in one defined group versus another reference group. This metric is foundational for interpreting predictive models where the dependent variable is binary. The core mechanism for calculating this value involves taking the exponentiated coefficients derived from the statistical model. Once obtained, the resulting odds ratio allows analysts to robustly interpret the magnitude and direction of the association between the independent variables and the likelihood of the specific outcome.

## Introduction: The Necessity of Odds Ratios for Interpretation

Logistic regression is the primary analytical tool used when fitting a regression model in R where the response variable is binary, meaning it can take on only two values (e.g., success/failure, 0/1). While the model estimates the probability of the outcome, the raw coefficients generated by R are expressed on the logit scale, which represents the log of the odds. These log-odds coefficients are mathematically complex and difficult to translate into practical, real-world implications for non-statisticians.

The coefficients in a fitted logistic regression model quantify the average change in the log-odds of the response variable associated with a one-unit increase in the respective predictor variable. For instance, a coefficient of 0.5 means that a one-unit increase in X leads to an increase of 0.5 in the log-odds of Y occurring. This logarithmic scale is necessary for modeling but hinders straightforward interpretation regarding risk or likelihood.

Consequently, researchers and analysts are often more interested in calculating the odds ratio for the predictor variables instead. The odds ratio is obtained by exponentiating the log-odds coefficient, transforming the result back to a ratio scale that is intuitive: a value of 2 means the odds double, while a value of 0.5 means the odds are halved. This transformation provides the clearest path to communicating the predictive strength of the model variables.

## Base Syntax for Extracting Point Estimates in R

The calculation of the odds ratio for every predictor variable in an R logistic regression model is remarkably simple, provided the model object has been correctly fitted using the `glm()` function with `family='binomial'`. The core requirement is to extract the vector of coefficients from the model object and then apply the exponentiation function.

The R function `coef(model)` efficiently retrieves all estimated coefficients, including the intercept. By wrapping this function call within the base R exponential function, `exp()`, we instantaneously convert all log-odds estimates into odds ratios. This allows for rapid assessment of the directional impact and magnitude of each predictor.

To quickly calculate the odds ratios (point estimates) for each predictor variable in the model, the following syntax is utilized:

```
exp(coef(model))
```

## Calculating Confidence Intervals for Reliable Odds Ratios

While the point estimate (the calculated odds ratio) is crucial, it is incomplete without a measure of statistical uncertainty. Therefore, calculating a 95% confidence interval (CI) for each odds ratio is standard practice. The confidence interval provides a range within which the true population odds ratio is expected to lie 95% of the time, helping to determine if the effect is statistically significant (i.e., if the interval excludes 1).

To obtain the confidence intervals, R's `confint(model)` function is applied to the model object. Critically, because the log-odds coefficients are used to calculate these limits, the resulting interval must also be exponentiated to correspond correctly to the odds ratio scale. This is achieved by applying `exp()` around the entire structure.

The syntax below achieves this by combining the coefficient estimates and their confidence limits using `cbind()` before exponentiation, resulting in a matrix that displays the odds ratio alongside its lower (2.5%) and upper (97.5%) bounds:

```
exp(cbind(Odds_Ratio = coef(model), confint(model)))
```

This comprehensive output is invaluable. If the 95% confidence interval for an odds ratio spans the value 1 (e.g., from 0.8 to 1.5), it signifies that the predictor is not statistically significant at the 0.05 level, as we cannot rule out the possibility of no association between the predictor and the odds of the outcome.

## Example Application: Leveraging the ISLR Default Dataset

To illustrate these techniques, we will apply the syntax to a real-world dataset. The standard **Default** dataset, readily available within the popular ISLR package, is an excellent choice for demonstrating credit risk modeling using logistic regression. This dataset comprises 10,000 observations detailing individual financial and demographic information.

Our first step involves loading the required package and examining the structure of the data using the `head()` function. This ensures that the variables are correctly identified and ready for model fitting. This dataset allows us to predict the probability of an individual defaulting on their credit card debt based on their financial and student status.

The following code snippet loads the necessary library and presents a summary of the first few entries in the **Default** dataset:

### library(ISLR)

```
#view first five rows of Default dataset  
head(Default)
```

```
default student balance income  
1 No No 729.5265 44361.625  
2 No Yes 817.1804 12106.135  
3 No No 1073.5492 31767.139  
4 No No 529.2506 35704.494  
5 No No 785.6559 38463.496  
6 No Yes 919.5885 7491.559
```

The dataset includes the following critical variables used in our predictive model:

**default:** The binary outcome variable (Yes/No) we aim to predict.

**student:** A categorical predictor indicating whether the individual is a student.

**balance:** The continuous measure of the average credit card balance.

**income:** The continuous measure of the individual's income.

## Fitting the Logistic Model and Viewing Log-Odds Coefficients

We proceed by constructing a logistic regression model that predicts the probability of defaulting based on the predictors `student`, `balance`, and `income`. The Generalized Linear Model (`glm`) function is employed, and the crucial step is specifying `family='binomial'` to correctly fit the link function appropriate for binary data.

Prior to viewing the summary, we adjust R's display options to disable scientific notation (`options(scipen=999)`). This improves the readability of the coefficient estimates, especially those that are very close to zero, ensuring that the raw numerical values are clearly displayed for interpretation before conversion to odds ratios. The output of the model summary provides the raw estimates in the log-odds scale.

The R commands to fit the model and view its summary, along with the output, are as follows:

```
#fit logistic regression model
```

```
model <- glm(default~student+balance+income, family='binomial', data=Default)
```

```
#disable scientific notation for model summary
```

```

options(scipen=999)

#view model summary
summary(model)

Call:
glm(formula = default ~ student + balance + income, family = "binomial",
data = train)

Deviance Residuals:
Min 1Q Median 3Q Max
-2.5586 -0.1353 -0.0519 -0.0177 3.7973

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -11.478101194 0.623409555 -18.412 <0.0000000000000002 ***
studentYes -0.493292438 0.285735949 -1.726 0.0843 .
balance 0.005988059 0.000293765 20.384 <0.0000000000000002 ***
income 0.000007857 0.000009965 0.788 0.4304
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2021.1 on 6963 degrees of freedom
Residual deviance: 1065.4 on 6960 degrees of freedom
AIC: 1073.4

Number of Fisher Scoring iterations: 8

```

From the output, we observe that for **balance**, the coefficient estimate is **0.005988**. In log-odds terms, this means a one-unit increase in balance is associated with an average increase of 0.005988 in the log of the odds of defaulting. While mathematically precise, this value requires conversion to an odds ratio for meaningful practical interpretation.

## Final Calculation of Odds Ratios and Confidence Bounds

To finalize our analysis, we now calculate the odds ratio point estimates and their corresponding confidence intervals. We use the two primary syntaxes introduced earlier to first find the raw odds ratios and then to construct the robust table including the confidence boundaries.

First, calculating the odds ratio for each predictor variable using exponentiation of the coefficients:

```
#calculate odds ratio for each predictor variable
exp(coef(model))
```

```
(Intercept) studentYes balance income
0.00001903854 0.52373166965 1.00575299051 1.00000303345
```

Next, calculating the odds ratio alongside the 95% confidence interval for each predictor variable:

```
#calculate odds ratio and 95% confidence interval for each predictor variable
exp(cbind(Odds_Ratio = coef(model), confint(model)))
```

```
Odds_Ratio 2.5 % 97.5 %
(Intercept) 0.00001903854 0.000007074481 0.0000487808
studentYes 0.52373166965 0.329882707270 0.8334223982
balance 1.00575299051 1.005308940686 1.0062238757
income 1.00000303345 0.999986952969 1.0000191246
```

## Interpreting the Calculated Odds Ratios

The odds ratio for each coefficient represents the average multiplicative increase or decrease in the odds of an individual defaulting, assuming all other predictor variables in the model are held constant (*ceteris paribus*). This careful interpretation is essential to avoid misleading conclusions about the relationship between variables.

Focusing on **balance**, the calculated odds ratio is approximately **1.0057**. Since this value is slightly greater than 1, it indicates a positive association with the odds of default. Specifically, for each additional dollar increase in the balance carried by an individual, the odds that the individual defaults on their loan increase by a factor of 1.0057. The accompanying confidence interval (1.0053 to 1.0062) does not include 1, confirming that this positive effect is statistically significant.

For the categorical variable **studentYes**, the odds ratio is **0.5237**, which is less than 1. This suggests that, relative to non-students (the reference category), students have lower odds of defaulting when controlling for balance and income. The odds of default for a student are about 52.37% of the odds for a non-student with the same balance and income. The interval (0.3299 to 0.8334) is entirely below 1, confirming statistical significance.

Finally, the predictor **income** yields an odds ratio of **1.000003**. This value is extremely close to 1, indicating a negligible effect. Crucially, its 95% confidence interval spans 1 (0.999987 to 1.000019). Because the interval includes 1, we conclude that income is not a statistically significant predictor of default odds in this model structure, suggesting no discernible relationship after

accounting for student status and balance. This disciplined method of analyzing both the point estimate and the confidence interval is crucial for robust reporting.

[How to Interpret Pr\(>|z|\) in Logistic Regression Output in R](#)

ARABPSYCHOLOGY.COM