How can we extract p-values from the lm() function in R?

Authored by

stats writer

June 25, 2024

The "lm()" function in R is commonly used for linear regression analysis. It allows users to fit a linear model to their data and obtain important statistical information, such as the coefficients and their corresponding p-values. To extract the p-values from the "lm()" function, one can use the "summary()" function, which provides a summary of the linear model including the p-values for each variable. Alternatively, the "coef()" function can be used to specifically extract the p-values for the coefficients. Overall, the "lm()" function in R provides a convenient way to obtain p-values for linear regression models, allowing for a thorough analysis of the relationship between variables.

Extract P-Values from lm() Function in R

You can use the following methods to extract p-values from the in R:

Method 1: Extract Overall P-Value of Regression Model

```
#define function to extract overall p-value of model
overall_p <- function(my_model) {
f <- summary(my_model)$fstatistic
p <- pf(f,f,f,lower.tail=F)
attributes(p) <- NULL
return(p)
}

#extract overall p-value of model
overall_p(model)
```

Method 2: Extract Individual P-Values for Regression Coefficients

summary(model)$coefficients

The following example shows how to use these methods in practice.

Example: Extract P-Values from lm() in R

Suppose we fit the following model in R:

```
#create data frame
df <- data.frame(rating=c(67, 75, 79, 85, 90, 96, 97),
points=c(8, 12, 16, 15, 22, 28, 24),
assists=c(4, 6, 6, 5, 3, 8, 7),
rebounds=c(1, 4, 3, 3, 2, 6, 7))

#fit multiple linear regression model
model <- lm(rating ~ points + assists + rebounds, data=df)
```

We can use the summary() function to view the entire summary of the regression model:

```
#view model summary
summary(model)
```

Call:

lm(formula = rating ~ points + assists + rebounds, data = df)

Residuals:
1 2 3 4 5 6 7
-1.5902 -1.7181 0.2413 4.8597 -1.0201 -0.6082 -0.1644

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 66.4355 6.6932 9.926 0.00218 **
points 1.2152 0.2788 4.359 0.02232 *
assists -2.5968 1.6263 -1.597 0.20860
rebounds 2.8202 1.6118 1.750 0.17847

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

Residual standard error: 3.193 on 3 degrees of freedom
Multiple R-squared: 0.9589, Adjusted R-squared: 0.9179
F-statistic: 23.35 on 3 and 3 DF, p-value: 0.01396

At the very bottom of the output we can see that the overall p-value for the regression model is 0.01396.

If we would like to only extract this p-value from the model, we can define a custom function to do so:

```
#define function to extract overall p-value of model
overall_p <- function(my_model) {
f <- summary(my_model)$fstatistic
p <- pf(f,f,f,lower.tail=F)
attributes(p) <- NULL
return(p)
}

#extract overall p-value of model
overall_p(model)
```

0.01395572

Notice that the function returns the same p-value as the model output from above.

To extract the p-values for the individual regression coefficients in the model, we can use the following syntax:

```
#extract p-values for individual regression coefficients in model
summary(model)$coefficients
```

(Intercept) points assists rebounds

0.002175313 0.022315418 0.208600183 0.178471275