

How can I use R to calculate leverage statistics?

Authored by
stats writer

April 23, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use R to calculate leverage statistics?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=138391>

R is a statistical programming language that can be used to calculate and analyze various financial metrics. One such metric is leverage, which measures the amount of debt a company has in relation to its equity. By using R, you can input financial data and use various statistical functions to calculate leverage statistics such as debt-to-equity ratio and interest coverage ratio. This can provide valuable insights into a company's financial health and risk level. Additionally, R offers the ability to graphically display and interpret the results, making it a powerful tool for financial analysis.

Calculate Leverage Statistics in R

In statistics, an is considered an outlier if it has a value for the response variable that is much larger than the rest of the observations in the dataset.

Similarly, an observation is considered to have high leverage if it has a value (or values) for the predictor variables that are much more extreme compared to the rest of the observations in the dataset.

One of the first steps in any type of analysis is to take a closer look at the observations that have high leverage since they could have a large impact on the results of a given model.

This tutorial shows a step-by-step example of how to calculate and visualize the leverage for each observation in a model in R.

Step 1: Build a Regression Model

First, we'll build a using the built-in mtcars dataset in R:

```
#load the dataset
```

```
data(mtcars)
```

```
#fit a regression model
```

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

```
#view model summary
```

```
summary(model)
```

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 30.735904 1.331566 23.083 < 2e-16 ***

disp -0.030346 0.007405 -4.098 0.000306 ***

hp -0.024840 0.013385 -1.856 0.073679 .

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

Residual standard error: 3.127 on 29 degrees of freedom

Multiple R-squared: 0.7482, Adjusted R-squared: 0.7309

F-statistic: 43.09 on 2 and 29 DF, p-value: 2.062e-09

Step 2: Calculate the Leverage for each Observation

Next, we'll use the `hatvalues()` function to calculate the leverage for each observation in the model:

```
#calculate leverage for each observation in the model  
hats <- as.data.frame(hatvalues(model))
```

```
#display leverage stats for each observation  
hats
```

```
hatvalues(model)  
Mazda RX4 0.04235795  
Mazda RX4 Wag 0.04235795  
Datsun 710 0.06287776  
Hornet 4 Drive 0.07614472  
Hornet Sportabout 0.08097817  
Valiant 0.05945972  
Duster 360 0.09828955  
Merc 240D 0.08816960  
Merc 230 0.05102253  
Merc 280 0.03990060  
Merc 280C 0.03990060  
Merc 450SE 0.03890159  
Merc 450SL 0.03890159
```

Merc 450SLC 0.03890159
Cadillac Fleetwood 0.19443875
Lincoln Continental 0.16042361
Chrysler Imperial 0.12447530
Fiat 128 0.08346304
Honda Civic 0.09493784
Toyota Corolla 0.08732818
Toyota Corona 0.05697867
Dodge Challenger 0.06954069
AMC Javelin 0.05767659
Camaro Z28 0.10011654
Pontiac Firebird 0.12979822
Fiat X1-9 0.08334018
Porsche 914-2 0.05785170
Lotus Europa 0.08193899
Ford Pantera L 0.13831817
Ferrari Dino 0.12608583
Maserati Bora 0.49663919
Volvo 142E 0.05848459

Typically we take a closer look at observations that have a leverage value greater than 2.

An easy way to do this is to sort the observations based

on their leverage value, descending:

```
#sort observations by leverage, descending  
hats), ]
```

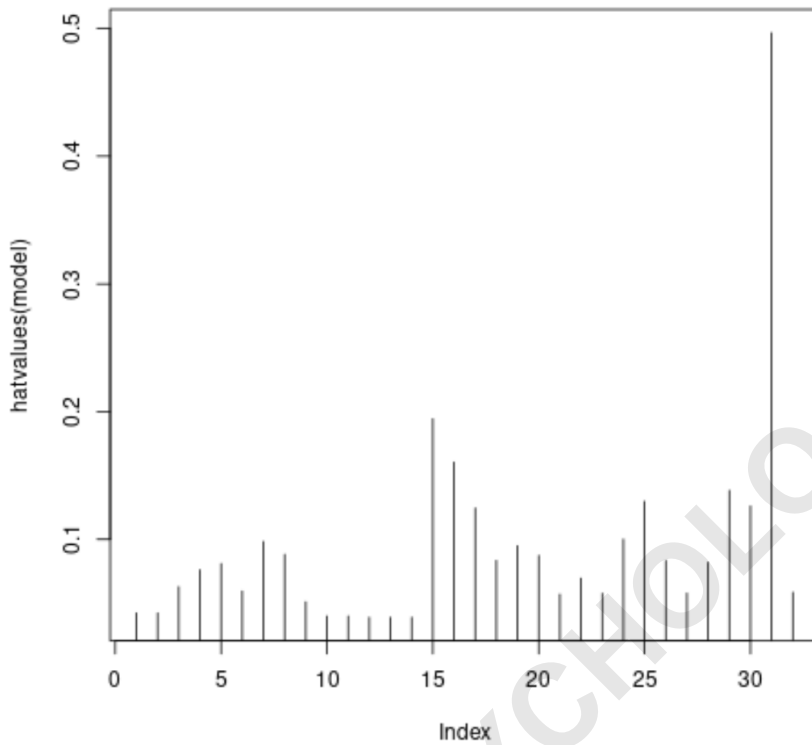
```
0.49663919  0.19443875  0.16042361  0.13831817  
0.12979822  0.12608583  
0.12447530  0.10011654  0.09828955  0.09493784  
0.08816960  0.08732818  
0.08346304  0.08334018  0.08193899  0.08097817  
0.07614472  0.06954069  
0.06287776  0.05945972  0.05848459  0.05785170  
0.05767659  0.05697867  
0.05102253  0.04235795  0.04235795  0.03990060  
0.03990060  0.03890159  
0.03890159  0.03890159
```

We can see that the largest leverage value is 0.4966. Since this isn't greater than 2, we know that none of the observations in our dataset have high leverage.

Step 3: Visualize the Leverage for each Observation

Lastly, we can create a quick plot to visualize the leverage for each observation:

```
#plot leverage values for each observation  
plot(hatvalues(model), type = 'h')
```



The x-axis displays the index of each observation in the dataset and the y-value displays the corresponding leverage statistic for each observation.