How can we use a subset of a data frame in the lm() function in R?

Authored by

stats writer

June 24, 2024

The lm() function in R is used for linear regression modeling and requires a data frame as an input for the dependent and independent variables. However, it is possible to use a subset of a data frame in the lm() function by specifying the desired columns or rows within the function. This allows for more efficient and targeted analysis, as only a portion of the data frame will be used in the regression model. Additionally, subsets can be created based on specific conditions or criteria, providing further flexibility in the analysis. Overall, utilizing subsets in the lm() function allows for more precise and customized regression modeling in R.

Use Subset of Data Frame with lm() Function in R

You can use the subset argument to only use a subset of a data frame when using the function to fit a regression model in R:

```
fit <- lm(points ~ fouls + minutes, data=df, subset=(minutes>10))
```

This particular example fits a regression model using points as the response variable and fouls and minutes as the predictor variables.

The subset argument specifies that only the rows in the data frame where the minutes variable is greater than 10 should be used when fitting the regression model.

The following example shows how to use this syntax in practice.

Example: How to Use Subset of Data Frame with lm() in R

Suppose we have the following data frame in R that contains information about the minutes played, total fouls, and total points scored by 10 basketball players:

```
#create data frame
df <- data.frame(minutes=c(5, 10, 13, 14, 20, 22, 26, 34, 38, 40),
fouls=c(5, 5, 3, 4, 2, 1, 3, 2, 1, 1),
points=c(6, 8, 8, 7, 14, 10, 22, 24, 28, 30))

#view data frame
df

minutes fouls points
1 5 5 6
2 10 5 8
3 13 3 8
4 14 4 7
5 20 2 14
6 22 1 10
7 26 3 22
8 34 2 24
9 38 1 28
```

10 40 1 30

Suppose we would like to fit the following multiple linear regression model:

points = β0 + β1(minutes) + β2(fouls)

However, suppose we only want to use the rows in the data frame where the minutes variable is greater than 10.

We can use the lm() function with the subset argument to fit this regression model:

```
#fit multiple linear regression model (only for rows where minutes>10)
fit <- lm(points ~ fouls + minutes, data=df, subset=(minutes>10))

#view model summary
summary(fit)
```

Call:
lm(formula = points ~ fouls + minutes, data = df, subset = (minutes >

10))

Residuals:
3 4 5 6 7 8 9 10
1.2824 -2.5882 2.2000 -1.9118 2.3588 -1.7176 0.1824 0.1941

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) -11.8353 4.9696 -2.382 0.063046 .
fouls 1.8765 1.0791 1.739 0.142536
minutes 0.9941 0.1159 8.575 0.000356 ***

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

Residual standard error: 2.255 on 5 degrees of freedom
Multiple R-squared: 0.9574, Adjusted R-squared: 0.9404
F-statistic: 56.19 on 2 and 5 DF, p-value: 0.0003744

We can use the nobs() function to see how many observations from the data frame were actually used to fit the regression model:

#view number of observations used to fit model
nobs(fit)

8

We can see that 8 rows from the data frame were used to fit the model.

If we view the original data frame we can see that exactly 8 rows had a value greater than 10 for the minutes variable, which means only those rows were used when fitting the regression model.

For example, we could use the following syntax to fit a regression model using only the rows in the data frame where minutes is greater than 10 and fouls is less than 4:

```
#fit multiple linear regression model (only where minutes>10 & fouls<4)
fit <- lm(points ~ fouls + minutes, data=df, subset=(minutes>10 & fouls<4))

#view number of observations used to fit model
nobs(fit)
```

7

From the output we can see that 7 rows from the data frame were used to fit this particular model.