How to fix “there are aliased coefficients in the model” in R?

Authored by

stats writer

July 2, 2024

In the context of statistical modeling in R, it is important to consider the possibility of aliased coefficients. Aliased coefficients refer to the situation where two or more regression coefficients in a model are highly correlated and therefore cannot be uniquely estimated. This can occur due to the presence of multicollinearity, where predictor variables in the model are highly correlated with each other. When this happens, the model may produce biased or unreliable estimates of the coefficients. It is therefore necessary to carefully assess and address potential multicollinearity in order to ensure the accuracy and validity of the model's results.

Fix in R: there are aliased coefficients in the model

One error you may encounter in R is:

Error in vif.default(model) : there are aliased coefficients in the model

This error typically occurs when exists in a regression model. That is, two or more predictor variables in the model are highly (or perfectly) correlated.

When this occurs, we say that one variable is an 'alias' of another variable, which causes problems when fitting a regression model.

The following example shows how to fix this error in practice.

How to Reproduce the Error

Suppose we fit the following in R:

```
#make this example reproducible
set.seed(0)

#define data
x1 <- rnorm(100)
x2 <- rnorm(100)
x3 <- x2*3
y <- rnorm(100)

#fit regression model
model <- lm(y~x1+x2+x3)
```

We can use the vif() function from the car package to calculate the VIF values for each predictor variable in the model to determine if multicollinearity is a problem:

```
library(car)
#calculate VIF values for predictor variables
vif(model)
```

```
Error in vif.default(model) : there are aliased coefficients in the model
```

We receive an error that "there are aliased coefficients in the model."

This tells us that two or more predictor variables in the model are perfectly correlated.

How to Fix the Error

To determine which predictor variables are perfectly correlated, we can use the cor() function to create a for the variables:

```
#place variables in data frame
df <- data.frame(x1, x2, x3, y)

#create correlation matrix for data frame
cor(df)

x1 x2 x3 y
x1 1.00000000 0.126886263 0.126886263 0.065047543
x2 0.12688626 1.000000000 1.000000000 -0.009107573
x3 0.12688626 1.000000000 1.000000000 -0.009107573
y 0.06504754 -0.009107573 -0.009107573 1.000000000
```

We can see that the variables x2 and x3 have a of 1. This tells us that these two variables are causing the error because they're perfectly correlated.

To fix this error, we simply need to fit the regression

model again and leave out one of these two variables.

For simplicity, let's remove x3 and fit the regression model again:

```
library(car)

#make this example reproducible
set.seed(0)

#define data
x1 <- rnorm(100)
x2 <- rnorm(100)
x3 <- x2*3
y <- rnorm(100)

#fit regression model
model <- lm(y~x1+x2)

#calculate VIF values for predictor variables in model
vif(model)

      x1       x2
1.016364 1.016364
```

Note that we don't receive any error this time when

calculating the VIF values for the model because multicollinearity is no longer an issue.

Related:

Additional Resources

The following tutorials explain how to fix other common errors in R: