How do I perform a Lack of Fit Test in R step-by-step?

Authored by

stats writer

April 26, 2024

A Lack of Fit Test in R is a statistical technique used to assess the validity of a regression model. This test helps to determine if the model adequately fits the data or if there is a lack of fit, indicating that the model may not accurately represent the relationship between the variables.

To perform a Lack of Fit Test in R, follow these steps:

Step 1: Load the necessary packages
Before starting the test, make sure to load the necessary packages in R, such as "car" and "stats" packages.

Step 2: Obtain the data
Next, obtain the data that you want to test for lack of fit. This data should have at least one independent variable and one dependent variable.

Step 3: Fit a linear regression model
Using the lm() function, fit a linear regression model with the obtained data. This model will serve as the baseline for the Lack of Fit Test.

Step 4: Generate a residual plot
Next, use the plot() function to generate a residual plot for the linear regression model. This plot will help to visually assess the fit of the model.

Step 5: Calculate the Lack of Fit Test
Using the anova() function, calculate the Lack of Fit Test. This will provide the sum of squares for the lack of fit and the residual sum of squares.

Step 6: Interpret the results
Finally, interpret the results of the Lack of Fit Test by comparing the sum of squares for the lack of fit to the residual sum of squares. A small difference between the two indicates a lack of fit, whereas a large difference suggests a good fit.

In conclusion, by following these steps, you can perform a Lack of Fit Test in R and determine if your regression model accurately represents the relationship between the variables.

Perform a Lack of Fit Test in R (Step-by-Step)

A lack of fit test is used to determine whether or not a full offers a significantly better fit to a dataset than

some reduced version of the model.

For example, suppose we would like to use *number of hours studied* to predict *exam score* for students at a certain college. We may decide to fit the following two regression models:

Full Model: Score = β0 + B1(hours) + B2(hours)2

Reduced Model: Score = β0 + B1(hours)

The following step-by-step example shows how to perform a lack of fit test in R to determine if the full model offers a significantly better fit than the reduced model.

Step 1: Create & Visualize a Dataset

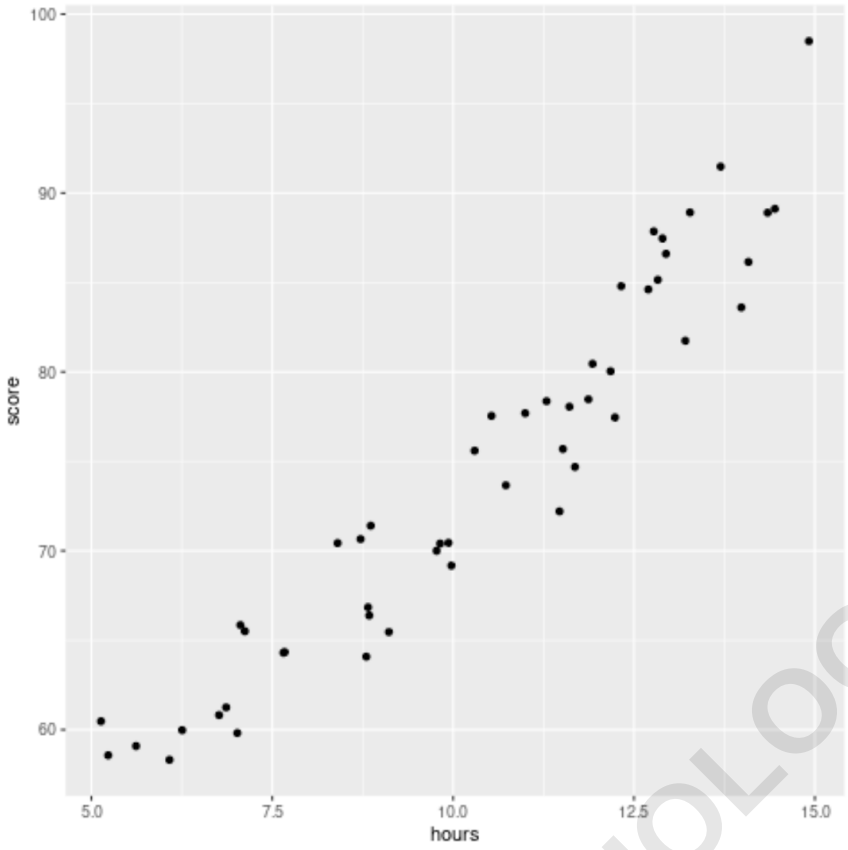
First, we'll use the following code to create a dataset that contains the number of hours studied and exam score received for 50 students:

```
#make this example reproducible
set.seed(1)

#create dataset
df <- data.frame(hours = runif(50, 5, 15), score=50)
```

```
df$score = df$score + df$hours^3/150 + df$hours*runif(50, 1, 2)

#view first six rows of data
head(df)

  hours score
1 7.655087 64.30191
2 8.721239 70.65430
3 10.728534 73.66114
4 14.082078 86.14630
5 7.016819 59.81595
6 13.983897 83.60510
```

Next, we'll create a scatterplot to visualize the relationship between hours and score:

```
#load ggplot2 visualization package
library(ggplot2)

#create scatterplot
ggplot(df, aes(x=hours, y=score)) +
geom_point()
```



Step 2: Fit Two Different Models to the Dataset

Next, we'll fit two different regression models to the dataset:

```
#fit full model
full <- lm(score ~ poly(hours,2), data=df)
#fit reduced model
reduced <- lm(score ~ hours, data=df)
```

Step 3: Perform a Lack of Fit Test

Next, we'll use the anova() command to perform a lack

of fit test between the two models:

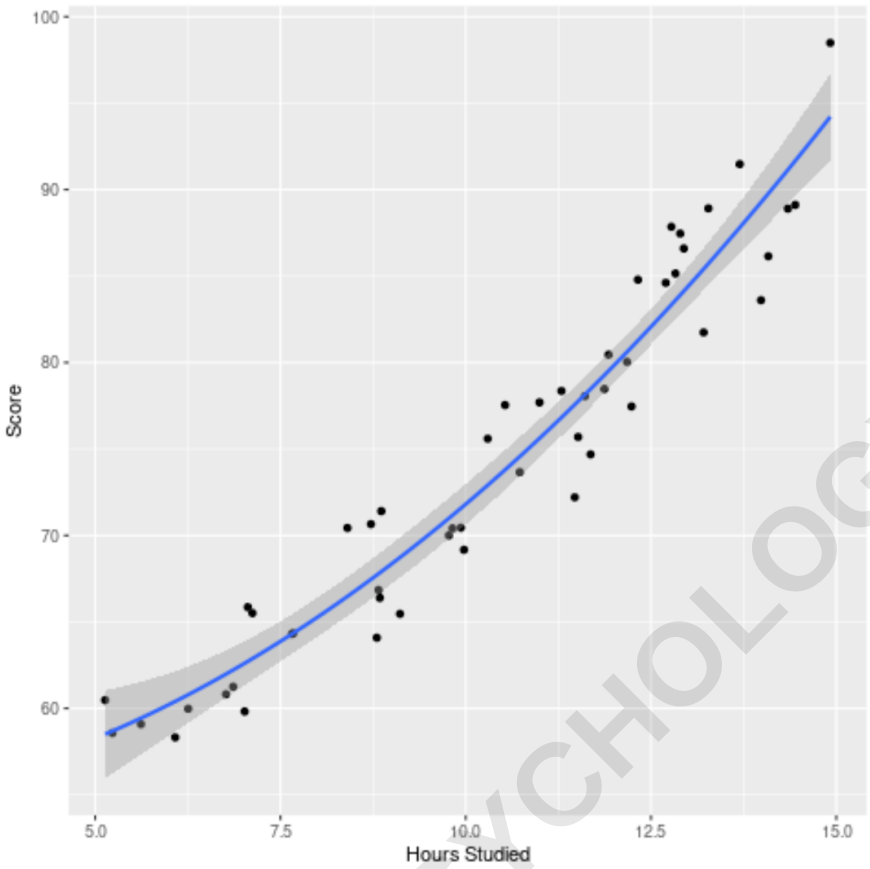
```
#lack of fit test
anova(full, reduced)

Analysis of Variance Table

Model 1: score ~ poly(hours, 2)
Model 2: score ~ hours
Res.Df RSS Df Sum of Sq F Pr(>F)
1 47 368.48
2 48 451.22 -1 -82.744 10.554 0.002144 **
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 4: Visualize the Final Model

Lastly, we can visualize the final model (the full model) relative to the original dataset:

```
ggplot(df, aes(x=hours, y=score)) +
geom_point() +
stat_smooth(method='lm', formula = y ~ poly(x,2), size = 1) +
xlab('Hours Studied') +
```

ylab('Score')



We can see that the curve of the model fits the data quite well.

How to Perform Polynomial Regression in R