How can the predict() function be used with the lm() function in R?

Authored by

stats writer

June 24, 2024

The predict() function in R is used to make predictions based on a statistical model. When used in conjunction with the lm() function, which is used to fit linear regression models, the predict() function can be used to make predictions on new data points that were not used to create the original model. This allows for testing the accuracy of the model and making predictions for future observations. The predict() function takes in the linear regression model as an argument, along with the new data points, and outputs a vector of predicted values. This allows for efficient and accurate predictions to be made using the lm() function in R.

Use the predict() Function with lm() in R

The lm() function in R can be used to fit linear regression models.

Once we've fit a model, we can then use the predict() function to predict the response value of a new .

This function uses the following syntax:

predict(object, newdata, type="response")

where:

object: The name of the model fit using the glm() functionnewdata: The name of the new data frame to make predictions fortype: The type of prediction to make.

The following example shows how to use the lm() function to fit a linear regression model in R and then

how to use the predict() function to predict the response value of a new observation the model hasn't seen before.

Example: Using the predict() Function with lm() in R

Suppose we have the following data frame in R that contains information about various 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 using minutes played and total fouls to predict the number of points scored by each player:

points = β0 + β1(minutes) + β2(fouls)

We can use the lm() function to fit this model:

```
#fit multiple linear regression model
fit <- lm(points ~ minutes + fouls, data=df)

#view summary of model
summary(fit)
```

```
Call:
lm(formula = points ~ minutes + fouls, data = df)

Residuals:
Min 1Q Median 3Q Max
-3.5241 -1.4782 0.5918 1.6073 2.0889

Coefficients:
```

Estimate Std. Error t value Pr(>|t|)
(Intercept) -11.8949 4.5375 -2.621 0.0343 *
minutes 0.9774 0.1086 9.000 4.26e-05 *
fouls 2.1838 0.8398 2.600 0.0354 *

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

Residual standard error: 2.148 on 7 degrees of freedom
Multiple R-squared: 0.959, Adjusted R-squared: 0.9473
F-statistic: 81.93 on 2 and 7 DF, p-value: 1.392e-05

Using the coefficients from the model output, we can write the fitted regression equation:

points = -11.8949 + 0.9774(minutes) + 2.1838(fouls)

We can then use the predict() function to predict the number of points that a player will score who plays for 15 minutes and has 3 total fouls:

```
#define new observation
newdata = data.frame(minutes=15, fouls=3)

#use model to predict points value
predict(fit, newdata)
```

1
9.317731

The model predicts that this player will score 9.317731 points.

Note that we can also make several predictions at once if we have a data frame that has multiple new observations.

For example, the following code shows how to use the fitted regression model to predict the points values for three players:

```
#define new data frame of three cars
newdata = data.frame(minutes=c(15, 20, 25),
fouls=c(3, 2, 1))

#view data frame
newdata
```

```
  minutes fouls
1      15     3
2      20     2
3      25     1
```

#use model to predict points for all three players
predict(model, newdata)

1 2 3
9.317731 12.021032 14.724334

Here's how to interpret the output:

The predicted points for the player with 15 minutes and 3 fouls is 9.32.The predicted points for the player with 20 minutes and 2 fouls is 12.02.The predicted points for the player with 25 minutes and 1 foul is 14.72.

Notes on Using predict()

The names of the columns in the new data frame should exactly match the names of the columns in the data frame that were used to build the model.

Notice that in our previous example, the data frame we used to build the model contained the following column names for our predictor variables:

minutesfouls

Thus, when we created the new data frame called

newdata we made sure to also name the columns:

minutesfouls

If the names of the columns do not match, you'll receive the following :

Error in eval(predvars, data, env)

Keep this in mind when using the predict() function.