How can I use SMOTE in R to handle imbalanced data? Can you provide an example?

Authored by

stats writer

June 28, 2024

SMOTE (Synthetic Minority Oversampling Technique) is a popular method used in data analysis to handle imbalanced data. It is particularly useful in scenarios where the number of observations in one class is significantly lower than the other. This technique involves creating synthetic data points for the minority class by using a combination of existing data points. In R, the SMOTE algorithm can be implemented using the "SMOTE" package. A simple example of using SMOTE in R would be to handle imbalanced data in a classification problem, where the minority class is the target variable. By applying SMOTE, the algorithm generates artificial data points for the minority class, thus balancing the dataset and improving the performance of the model.

Use SMOTE for Imbalanced Data in R (With Example)

Often when working with in machine learning, the classes in the dataset will be imbalanced.

For example:

A dataset that contains information on whether or not college players get drafted into the NBA might have 98% of players not get drafted and 2% get drafted.A dataset that contains information on whether or not patients have cancer might have 99% of patients without cancer and just 1% with cancer.A dataset that contains information on bank fraud may contain 96% of transactions that are legitimate and 4% that are fraudulent.

As a result of these imbalanced classes, the predictive model that you build is likely to perform poorly on the

minority class.

Worse still, the minority class is often the class we're most interested in predicting.

One way to address this imbalance problem is to use Synthetic Minority Oversampling Technique, often abbreviated SMOTE.

This technique involves creating a new dataset by oversampling observations from the minority class, which produces a dataset that has more balanced classes.

The easiest way to use SMOTE in R is with the SMOTE() function from the DMwR package.

This function uses the following basic syntax:

SMOTE(form, data, perc.over = 200, perc.under = 200, ...)

where:

form: A formula describing the model you'd like to fitdata: Name of the data frame perc.over: Number that

determines how many extra cases from the minority class are generatedperc.under: Number that determines how many extra cases from the majority class are generated

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

Example: How to Use SMOTE in R

Suppose we have the following dataset with 100 in R in which 90 have a class of 'Yes' and 10 have a class of 'No' for the response variable:

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

#create data frame with one response variable and two predictor variables
df <- data.frame(y=rep(as.factor(c('Yes', 'No')), times=c(90, 10)),
x1=rnorm(100),
x2=rnorm(100))

#view first six rows of data frame
head(df)
```

```
y x1 x2
1 Yes 1.2629543 0.7818592
2 Yes -0.3262334 -0.7767766
3 Yes 1.3297993 -0.6159899
4 Yes 1.2724293 0.0465803
5 Yes 0.4146414 -1.1303858
6 Yes -1.5399500 0.5767188

#view distribution of response variable
table(df$y)

No Yes
10 90
```

This is a classic example of an imbalanced dataset because the response variable that we're predicting has 90 observations that have a class of 'Yes' and just 10 observations that have a class of 'No.'

```
library(DMwR)

#use SMOTE to create new dataset that is more balanced
new_df <- SMOTE(y ~ ., df, perc.over = 2000, perc.under = 400)
```

#view distribution of response variable in new dataset
table(new_df$y)

No Yes
210 800

The resulting dataset has 210 observations with 'No' as their class and 800 observations with 'Yes' as their class.

Here's exactly how the SMOTE function produced this new dataset:

The perc.over argument specified that we wanted to add 2000/100 (i.e. 20) times the number of existing minority observations to the dataset. Since 10 observations existed in the original dataset, we added 20*10 = 200 more minority observations.The perc.under argument specified that we wanted to make the number of majority observations equal to 400/100 (i.e. 4) times the number of minority observations added to the existing minority observations. Since 200 more minority observations were added, we made the number of majority observations equal to 200 * 4 = 800 majority observations.

The end result is a dataset that still contains more majority classes, but is still more balanced than the original dataset.

You can now proceed to fit your classification algorithm of your choice to this new dataset, which should perform better on the minority class since there are more observations from the minority class in this new dataset.

Note: Feel free to play around with the perc.over and perc.under arguments in the SMOTE function to get a dataset that suits your needs.

Additional Resources

The following tutorials explain how to perform other common tasks in R: