What is the process for applying a function to each row in a matrix or data frame in R?

Authored by

stats writer

May 2, 2024

The process for applying a function to each row in a matrix or data frame in R involves using the apply() function. This function takes in three arguments: the data frame or matrix, the margin (1 for rows and 2 for columns), and the function to be applied. The apply() function then iterates through each row or column, applying the specified function to each element. This allows for efficient and streamlined manipulation of data within a matrix or data frame in R.

Apply Function to Each Row in Matrix or Data Frame in R

You can use the apply() function to apply a function to each row in a matrix or data frame in R.

This function uses the following basic syntax:

apply(X, MARGIN, FUN)

where:

X: Name of the matrix or data frame.MARGIN: Dimension to perform operation across. Use 1 for row, 2 for column.FUN: The function to apply.

The following examples show how to use this syntax in practice.

Example 1: Apply Function to Each Row in Matrix

Suppose we have the following matrix in R:

#create matrix

mat <- matrix(1:15, nrow=3)#view matrix

mat

1 4 7 10 13

2 5 8 11 14

3 6 9 12 15

We can use the apply() function to apply different functions to the rows of the matrix:

#find mean of each row

apply(mat, 1, mean)

7 8 9

#find sum of each row

apply(mat, 1, sum)

35 40 45

#find standard deviation of each row

apply(mat, 1, sd)

4.743416 4.743416 4.743416

#multiply the value in each row by 2 (using t() to

transpose the results)
t(apply(mat, 1, function(x) x * 2))

2 8 14 20 26
4 10 16 22 28
6 12 18 24 30

#normalize every row to 1 (using t() to transpose the results)
t(apply(mat, 1, function(x) x / sum(x)))

0.02857143 0.1142857 0.2 0.2857143 0.3714286
0.05000000 0.1250000 0.2 0.2750000 0.3500000
0.06666667 0.1333333 0.2 0.2666667 0.3333333

Note that if you'd like to find the mean or sum of each row, it's faster to use the built-in rowMeans() or rowSums() functions:

#find mean of each row
rowMeans(mat)

7 8 9
#find sum of each row
rowSums(mat)

35 40 45

Example 2: Apply Function to Each Row in Data Frame

Suppose we have the following matrix in R:

```
#create data frame
df <- data.frame(var1=1:3,
var2=4:6,
var3=7:9,
var4=10:12,
var5=13:15)#view data frame
df
```

```
var1 var2 var3 var4 var5
1 1 4 7 10 13
2 2 5 8 11 14
3 3 6 9 12 15
```

We can use the apply() function to apply different functions to the rows of the data frame:

```
#find mean of each row
apply(df, 1, mean)
```

7 8 9

#find sum of each row
apply(df, 1, sum)

35 40 45

#find standard deviation of each row
apply(df, 1, sd)

4.743416 4.743416 4.743416

#multiply the value in each row by 2 (using t() to transpose the results)
t(apply(df, 1, function(x) x * 2))

var1 var2 var3 var4 var5
2 8 14 20 26
4 10 16 22 28
6 12 18 24 30

#normalize every row to 1 (using t() to transpose the results)
t(apply(df, 1, function(x) x / sum(x)))

var1 var2 var3 var4 var5
0.02857143 0.1142857 0.2 0.2857143 0.3714286
0.05000000 0.1250000 0.2 0.2750000 0.3500000

0.06666667 0.1333333 0.2 0.2666667 0.3333333

Similar to matrices, if you'd like to find the mean or sum of each row, it's faster to use the built-in rowMeans() or rowSums() functions:

#find mean of each row
rowMeans(df)

7 8 9
#find sum of each row
rowSums(df)

35 40 45