apply(), lapply(), sapply(), tapply() Function in R with Examples

Becky Lin

2023-02-13

Introduction

This tutorial aims at introducing `apply()` function collection which can be viewed as a substitute to the loop in R.

The apply() function is the basic model of the family of apply functions in R, which includes specific functions like lapply(), sapply(), tapply(), mapply(), vapply(), rapply(), bapply(), eapply(), and others. All of these functions allow us to iterate over a data structure such as a list, a matrix, an array, a DataFrame, or a selected slice of a given data structure — and perform the same operation at each element.

apply() function

apply() takes Data frame or matrix as an input and gives output in vector, list or array. Apply function in R is primarily used to avoid explicit uses of loop constructs. It is the most basic of all collections can be used over a matrice.

This function takes 3 arguments

```
apply(X, MARGIN, FUN)
```

where

- X: an array or matrix or data frame
- MARGIN: take a value or range between 1 and 2 to define where to apply the function FUN.
  - MARGIN=1: the FUN is performed on rows
  - MARGIN=2: the FUN is performed on columns
  - MARGIN = c(1,2): the FUN is performed on rows and columns.
- FUN: tells which function to apply. Built functions like mean, median, sum, min, max, and even user-defined functions can be applied.

Examples

```
data <- matrix(C<-(1:12),nrow=3, ncol=4)
data
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    4    7   10
## [2,]    2    5    8   11
## [3,]    3    6    9   12
```

```
apply(data, 2, sum)
```

```
## [1]  6 15 24 33
```

```
apply(data, 1, sum)
```

```
## [1] 22 26 30
```

```
f<-function(x){ max(x)-min(x)}
apply(data,2,f)
```

```
## [1] 2 2 2 2
```

```
apply(data, 2, range)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    4    7   10
## [2,]    3    6    9   12
```

lapply() function

lappy() returns a list of the similar length as input list object, each element of which is the result of applying FUN to the corresponding element of list. The "l" in lappy() stands for *list*. The difference between lapply() and apply() lies between the output return. The output of lapply() is a list. lapply() can be used for other objects like data frames and lists.

This function takes two arguments

```
lapply(X, FUN)
```

where

- X: a list or a vector or a data frame
- FUN: function applied to each element of X

Examples

```
movies <- c("SPYDERMAN","BATMAN","VERTIGO")
lapply(movies,tolower)
```

```
## [[1]]
## [1] "spyderman"
##
## [[2]]
## [1] "batman"
##
## [[3]]
## [1] "vertigo"
```

```
data2 = matrix(c(1:4),nrow=2,ncol=2)
lapply(data2,function(x) x+1)
```

```
## [[1]]
## [1] 2
##
## [[2]]
## [1] 3
##
## [[3]]
## [1] 4
##
## [[4]]
## [1] 5
```