

Família Apply na linguagem R

Ivanildo Batista

16 de março de 2021

Família Apply na linguagem R

As funções da família `apply()` são uma forma de gerar loop for de forma mais estilizada, pois loops for em R não são muito bons.

Principais funções da família `apply()`:

1. `apply()`
2. `lapply()`
3. `sapply()`
4. `tapply()`
5. `mapply()`
6. `vapply()`
7. `eapply()`
8. `rapply()`
9. `apply()`
10. `by()`

Usando as funções `SAPPLY()` e `APPLY(0)`

```
lista1 = list(a=(1:10),b=(45:77)) #criando uma lista  
lista1
```

```
## $a  
## [1] 1 2 3 4 5 6 7 8 9 10  
##  
## $b  
## [1] 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69  
## [26] 70 71 72 73 74 75 76 77
```

```
sapply(lista1, sum) #aplicando uma função nos elementos da lista função soma
```

```
##      a      b  
## 55 2013
```

```
sapply(lista1, mean) #função média
```

```
##      a      b  
## 5.5 61.0
```

```
x = matrix(rnorm(9),nr=3,byrow=T) #criando uma matriz  
x
```

```
##           [,1]      [,2]      [,3]  
## [1,] 0.07523009 -0.08629155 0.1181008  
## [2,] -1.07152030 -1.52314653 2.1249103  
## [3,] -0.30426313 0.02648266 -0.2508673
```

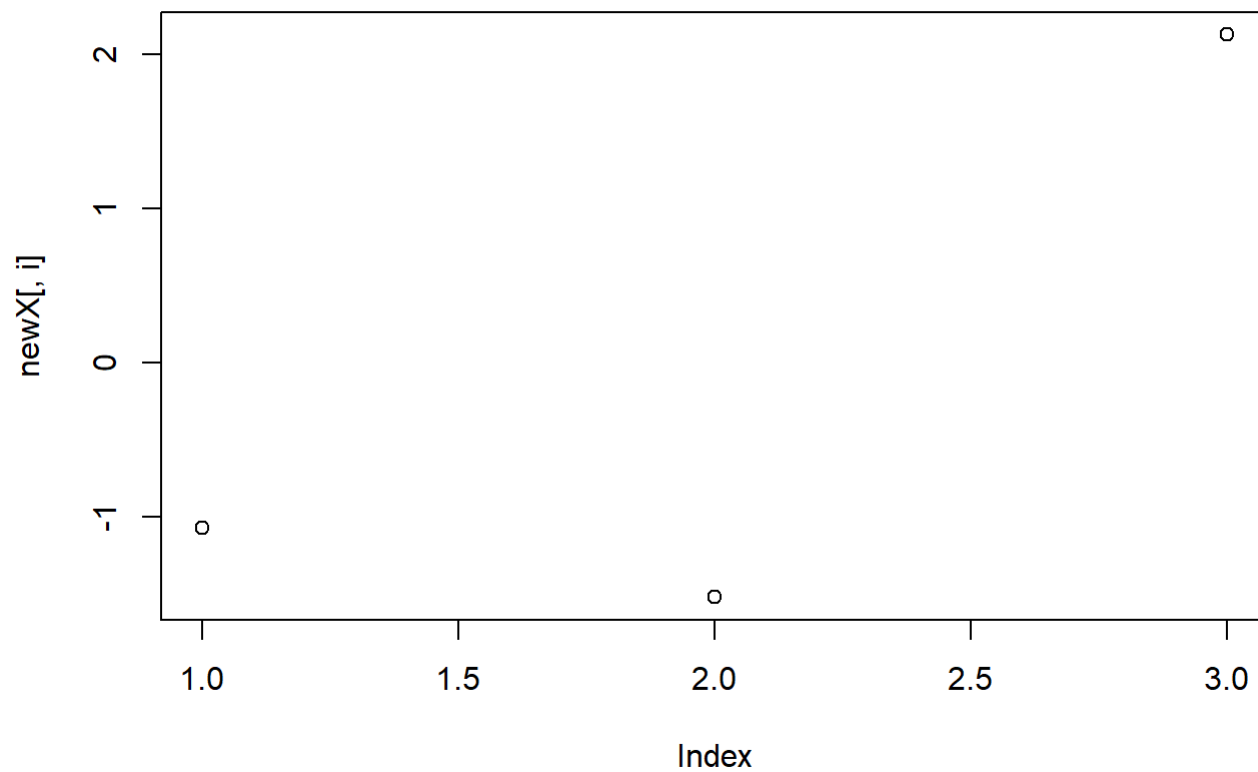
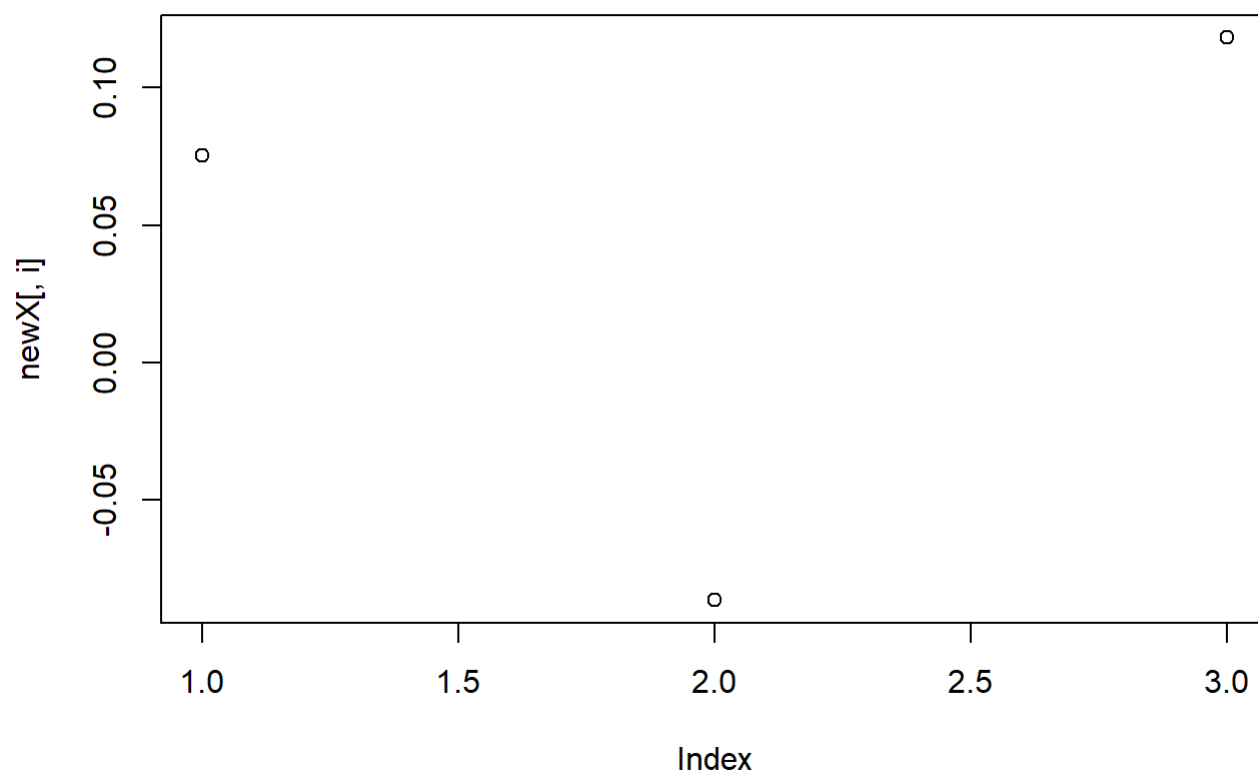
```
apply(x, 1, mean) #tirando a média das linhas
```

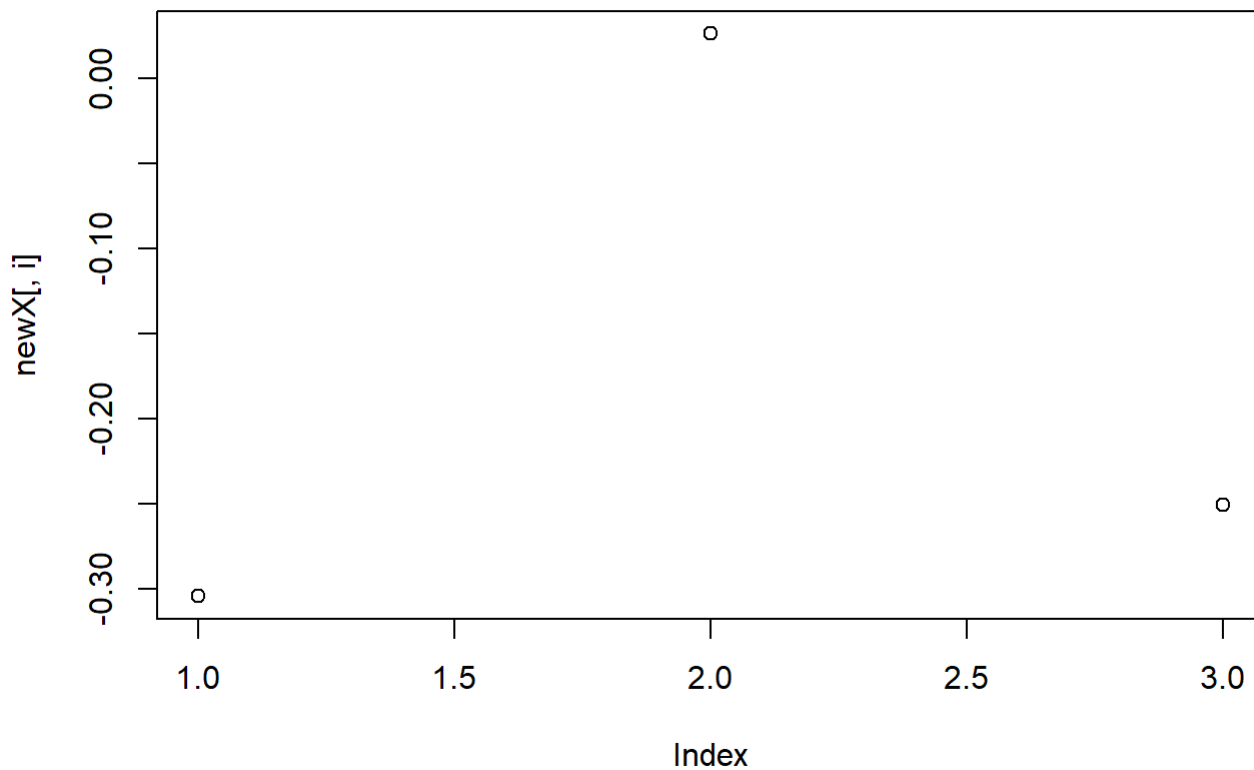
```
## [1] 0.03567978 -0.15658550 -0.17621592
```

```
apply(x, 2, mean) #tirando a média das colunas
```

```
## [1] -0.4335178 -0.5276518 0.6640479
```

```
apply(x, 1, plot) #plotando a média das linhas
```





```
## NULL
```

```
resultapply = apply(x,1,mean) #atribuindo o resultado da função a uma variável
resultapply
```

```
## [1] 0.03567978 -0.15658550 -0.17621592
```

Criando um dataframe

```
escola = data.frame(Aluno = c('Alan','Alice','Alana','Aline','Alex','Ajay'),
                    Matemática = c(90,80,85,87,56,79),
                    Geografia = c(100,78,86,90,98,67),
                    Química = c(76,56,89,90,100,87))
```

```
escola
```

```
##   Aluno Matemática Geografia Química
## 1 Alan          90         100        76
## 2 Alice          80          78        56
## 3 Alana          85          86        89
## 4 Aline          87          90        90
## 5 Alex           56          98       100
## 6 Ajay           79          67        87
```

```
escola$Geografia
```

```
## [1] 100 78 86 90 98 67
```

```
escola$Media = NA #criando nova coluna com valores NA
```

```
escola
```

```
##   Aluno Matemática Geografia Química Media
## 1  Alan          90         100       76   NA
## 2  Alice          80          78       56   NA
## 3  Alana          85          86       89   NA
## 4  Aline          87          90       90   NA
## 5  Alex           56          98      100   NA
## 6  Ajay           79          67       87   NA
```

```
escola$Media = apply(escola[,c(2,3,4)],1,mean) #substituindo valores pela média com a função apply
```

```
escola
```

```
##   Aluno Matemática Geografia Química   Media
## 1  Alan          90         100       76 88.66667
## 2  Alice          80          78       56 71.33333
## 3  Alana          85          86       89 86.66667
## 4  Aline          87          90       90 89.00000
## 5  Alex           56          98      100 84.66667
## 6  Ajay           79          67       87 77.66667
```

Usando as funções TAPPLY() e SQLDF()

```
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```
require(sqldf)

escola2 = data.frame(Aluno = c('Alan','Alice','Alana','Alan','Alice','Alana'),
                     Semestre = c(1,1,1,2,2,2),
                     Matemática = c(90,80,85,87,56,79),
                     Geografia = c(100,78,86,90,98,67),
                     Química = c(76,56,89,90,100,87))

escola2
```

```
##   Aluno Semestre Matemática Geografia Química
## 1  Alan         1         90       100       76
## 2 Alice         1         80        78       56
## 3 Alana         1         85        86       89
## 4  Alan         2         87        90       90
## 5 Alice         2         56        98      100
## 6 Alana         2         79        67       87
```

```
sqldf("select aluno, sum(Matemática), sum(Geografia), sum(Química) from escola2
      group by aluno")
```

```
##   Aluno sum(Matemática) sum(Geografia) sum(Química)
## 1  Alan             177             190             166
## 2 Alana             164             153             176
## 3 Alice             136             176             156
```

```
tapply((escola2$Matemática),escola2$Aluno,sum)
```

```
##   Alan Alana Alice
##   177   164   136
```

Usando a função BY()

```
by(escola2[,c(2,3,4)], escola2$Semestre,colSums)
```

```
## escola2$Semestre: 1
##   Semestre Matemática  Geografia
##           3         255         264
## -----
## escola2$Semestre: 2
##   Semestre Matemática  Geografia
##           6         222         255
```

Usando a função LAPPLY()

```
#geram os mesmos resultados, mas de formas diferentes
lapply(lista1, sum)
```

```
## $a
## [1] 55
##
## $b
## [1] 2013
```

```
sapply(lista1, sum)
```

```
##      a      b
##    55 2013
```

Usando a função VAPPLY()

```
vapply(lista1, fivenum, c(Min.=0, "1st Qu." = 0, Median = 0, "3rd Qu"=0,Max=0))
```

```
##           a  b
## Min.      1.0 45
## 1st Qu.    3.0 53
## Median     5.5 61
## 3rd Qu     8.0 69
## Max      10.0 77
```

Usando a função REPLICATE()

```
replicate(7, runif(10))
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 0.5749761 0.2394426 0.492168450 0.7477393 0.53168580 0.37121206 0.1009692
## [2,] 0.6934932 0.9978868 0.613453142 0.7122944 0.04564391 0.72610567 0.5136278
## [3,] 0.2659396 0.4881051 0.546283292 0.7716817 0.92860953 0.68357087 0.6516230
## [4,] 0.9743998 0.8640832 0.685452727 0.5330366 0.60890800 0.27107948 0.3840857
## [5,] 0.2390672 0.2023549 0.557534721 0.4846555 0.16010096 0.96173024 0.5575167
## [6,] 0.5044847 0.2540577 0.369291903 0.7962191 0.84067644 0.82655264 0.9868646
## [7,] 0.7925869 0.1561709 0.480274981 0.9064354 0.73264987 0.91229680 0.2752528
## [8,] 0.3511884 0.5218825 0.131210626 0.5701949 0.52154995 0.08048398 0.9066362
## [9,] 0.7371229 0.3803081 0.002937342 0.5334225 0.87510420 0.44463348 0.1853891
## [10,] 0.8976507 0.8682901 0.389746829 0.1079901 0.48620920 0.04120728 0.9533933
```

Usando as funções RAPPLY() e MAPPLY()

```
mapply(rep, 1:4,4:1)
```

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

```
lista2 = list(a=c(1:5),b=c(6:10))
lista2
```

```
## $a
## [1] 1 2 3 4 5
##
## $b
## [1] 6 7 8 9 10
```

```
rapply(lista2, sum)
```

```
## a b
## 15 40
```

```
rapply(lista2, sum, how='list') #resultado anterior, mas em forma de lista
```

```
## $a
## [1] 15
##
## $b
## [1] 40
```