

How can the `lapply()` function be used with multiple arguments in R?

Authored by
stats writer

June 24, 2024

RECOMMENDED CITATION

stats writer (2024). *How can the `lapply()` function be used with multiple arguments in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151392>

The lapply() function in R is a powerful tool that allows for the application of a function to each element of a list or vector. It can also be used with multiple arguments, providing even more versatility and efficiency in data analysis. By specifying the function to be applied and the multiple arguments to be used, lapply() can perform the same operation on each element of the list or vector, resulting in a new list or vector with the same length as the original. This allows for streamlined and consistent data manipulation, making it a valuable tool for any R programmer.

R: Use lapply() Function with Multiple Arguments

The lapply() function in R can be used to apply a function to each element of a list, vector, or data frame and obtain a list as a result.

To use the lapply() function with multiple arguments, you can use the following basic syntax:

```
#define function
```

```
my_function <- function(var1,var2,var3){  
var1*var2*var3  
}
```

```
#apply function to list using multiple arguments
```

```
lapply(my_list, my_function, var2=3, var3=5)
```

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

Example: How to Use lapply() with Multiple Arguments in R

Suppose we have the following list in R:

```
#create a list
```

```
my_list <- list(A=1, B=2, C=3, D=4)
```

```
#view list
```

```
my_list
```

```
$A
```

```
1
```

```
$B
```

```
2
```

```
$C
```

```
3
```

```
$D
```

```
4
```

The following code defines a function that accepts three variables and multiplies all three variables together, then uses the lapply() function to apply this function to each value in our list:

#define function

```
my_function <- function(var1,var2,var3){  
var1*var2*var3  
}
```

#apply function to list using multiple arguments

```
lapply(my_list, my_function, var2=3, var3=5)
```

\$A

15

\$B

30

\$C

45

\$D

60

Notice that the lapply() function multiplies each value in the list by 3 and then by 5.

For example:

First value in list: $1 * 3 * 5 = 15$ Second value in list: $2 * 3$

*** 5 = 30**Third value in list: **3 * 3 * 5 = 45**Fourth value in list: **4 * 3 * 5 = 60**

Using similar syntax, you can supply as many arguments as you'd like to the lapply() function.

ARABPSYCHOLOGY.COM