How can the drop function be used in R and what are some examples of its application?

Authored by

stats writer

June 23, 2024

ARABPSYCHOLOGY.COM

RECOMMENDED CITATION

stats writer (2024). *How can the drop function be used in R and what are some examples of its application?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=148566

The drop function in R is a useful tool for removing dimensions from a data structure or object. It can be applied to a variety of data types, including vectors, matrices, and data frames. By specifying the dimension or dimensions to be dropped, the drop function allows for easy manipulation of data without altering the underlying structure.

One common application of the drop function is in subsetting data frames. For example, if a data frame has both row and column names, the drop function can be used to remove one of these dimensions, resulting in a simplified subset of the original data.

Another example is in the manipulation of matrices. The drop function can be used to remove rows or columns that contain missing or irrelevant data, making it easier to perform calculations or analyses on the remaining data.

Overall, the drop function is a versatile tool for data manipulation in R, allowing for more efficient and streamlined data processing.

Use the drop Function in R (With Examples)

The drop() function in base R can be used to delete the dimensions of an array or matrix that only have one level.

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

Example 1: Use drop() to Delete Dimensions with One Level in Array

Suppose we have the following 3-dimensional array in R:

#create 3-dimensional array

ARABPSYCHOLOGY.COM

```
my_array <- c(1:10)
dim(my_array ) <- c(1,2,5)

#view array
my_array

, , 1

1 2

, , 2

3 4

, , 3

5 6

, , 4

7 8

, , 5

9 10
```

We can use the drop() function to drop the dimension

ARABPSYCHOLOGY.COM

that only has one level in the array:

```
#drop dimensions with only one level
new_array <- drop(my_array)

#view new array
new_array
```

```
1 3 5 7 9
2 4 6 8 10
```

Notice that the dimension with only one level has been dropped.

We can use the dim() function to view the new dimensions:

```
#view dimensions of new array
dim(new_array)
```

```
2 5
```

We can see that the new array only has two dimensions.

Example 2: Use drop() to Delete Dimensions with One Level in

ARABPSYCHOLOGY.COM

Matrix

Suppose we have the following matrix with seven columns and one row in R:

```
#create matrix
my_matrix <- matrix(1:7, ncol=7)

#view matrix
my_matrix

1 2 3 4 5 6 7

#view dimensions of matrix
dim(my_matrix)

1 7
```

We can use the drop() function to drop the dimension that only has one level in the matrix:

```
#drop dimensions with only one level
new_matrix <- drop(my_matrix)

#view new matrix
new_matrix
```

ARABPSYCHOLOGY.COM

1 2 3 4 5 6 7

Notice that the dimension with only one level has been dropped.

If we use the dim() function to view the dimensions, it will return NULL since the new object is no longer a matrix with two dimensions:

```
#view dimensions of new matrix
dim(new_matrix)
```

NULL

Instead, we can use length() to display the length of the vector:

```
#view length
length(new_matrix)
```

7

We can see that our vector has 7 elements in it.

ARABPSYCHOLOGY.COM