

How can I use the `replace()` function in R?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the `replace()` function in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165029>

The `replace()` function in R is a useful tool that allows you to replace specific values in a vector, matrix, or data frame with new values. This function takes in three parameters: the original data object, the value to be replaced, and the new value. It can be used in various scenarios, such as correcting data errors, updating outdated values, or converting data types. The `replace()` function is simple to use and can greatly improve the efficiency and accuracy of data manipulation in R.

Use the `replace()` Function in R

The `replace()` function in R can be used to replace specific elements in a vector with new values.

This function uses the following syntax:

```
replace(x, list, values)
```

where:

x: Name of vector
list: Elements to replace
values: Replacement values

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

Example 1: Replace One Value in Vector

The following code shows how to replace the element in position 2 of a vector with a new value of 50:

```
#define vector of values
```

```
data <- c(3, 6, 8, 12, 14, 15, 16, 19, 22)
```

```
#define new vector with a different value in position 2
```

```
data_new <- replace(data, 2, 50)
```

```
#view new vector
```

```
data_new
```

```
3 50 8 12 14 15 16 19 22
```

Notice that the element in position 2 has changed, but every other value in the original vector remained the same.

Example 2: Replace Multiple Values in Vector

The following code shows how to replace the values of multiple elements in a vector with new values:

```
#define vector of values
```

```
data <- c(2, 4, 6, 8, 10, 12, 14, 16)
```

```
#define new vector with different values in position 1, 2,  
and 8
```

```
data_new <- replace(data, c(1, 2, 8), c(50, 100, 200))
```

```
#view new vector
```

data_new

50 100 6 8 10 12 14 200

Notice that the elements in position 1, 2, and 8 all changed.

Example 3: Replace Values in Data Frame

The following code shows how to replace the values in a certain column of a data frame that meet a specific condition:

```
#define data frame
```

```
df <- data.frame(x=c(1, 2, 4, 4, 5, 7),  
y=c(6, 6, 8, 8, 10, 11))
```

```
#view data frame
```

```
df
```

```
x y
```

```
1 1 6
```

```
2 2 6
```

```
3 4 8
```

```
4 4 8
```

```
5 5 10
```

6 7 11

#replace values in column 'x' greater than 4 with a new value of 50

```
df$x <- replace(df$x, df$x > 4, 50)
```

#view updated data frame

df

x y

1 1 6

2 2 6

3 4 8

4 4 8

5 50 10

6 50 11

All other values in the data frame remained the same.

Additional Resources