

How can I use the `pivot_wider()` function in R to create multiple columns, instead of just one, when reshaping my data from long to wide?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the `pivot_wider()` function in R to create multiple columns, instead of just one, when reshaping my data from long to wide?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151842>

The `pivot_wider()` function in R allows for efficient reshaping of data from a long to a wide format. By using this function, multiple columns can be created instead of just one, making it easier to organize and analyze data. This function is particularly useful for transforming data from a tall format, where each row represents a unique observation, to a wide format, where each column represents a variable. With `pivot_wider()`, users can specify which variables to pivot and which values to use as column names, resulting in a more organized and structured dataset. This function is a valuable tool for data manipulation and analysis in R.

R: Use `pivot_wider()` with Multiple Columns

The `pivot_wider()` function from the package in R can be used to pivot a data frame from a long format to a wide format.

If you'd like to use this function to pivot multiple columns, you can use the following syntax:

```
library(tidyr)
```

```
df_wide <- pivot_wider(df, names_from=group,  
values_from=c(values1, values2))
```

By providing multiple column names to the `values_from` argument, you can pivot multiple columns at once.

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

Example: Use `pivot_wider()` with Multiple Columns in R

Suppose we have the following data frame in R that contains information about various basketball players:

```
#create data frame
```

```
df <- data.frame(team=c('A', 'A', 'A', 'B', 'B', 'B'),  
player=c('G', 'F', 'C', 'G', 'F', 'C'),  
points=c(22, 34, 20, 15, 14, 19),  
assists=c(4, 10, 12, 9, 8, 5))
```

```
#view data frame
```

```
df
```

```
team player points assists
```

```
1 A G 22 4
```

```
2 A F 34 10
```

```
3 A C 20 12
```

```
4 B G 15 9
```

```
5 B F 14 8
```

```
6 B C 19 5
```

Now suppose we would like to pivot the values in the points and assists columns at the same time.

We can use the following syntax to do so:

```
library(tidyr)
```

```
#pivot values in points and assists columns
```

```
df_wide <- pivot_wider(df, names_from=player,  
values_from=c(points, assists))
```

```
#view wide data frame
```

```
df_wide
```

```
# A tibble: 2 x 7
```

```
team points_G points_F points_C assists_G assists_F  
assists_C
```

```
1 A 22 34 20 4 10 12
```

```
2 B 15 14 19 9 8 5
```

Notice that each value in the player column has been combined with points and assists to create a total of six new columns that showed the points and assists scored by players in each position.

The final result is a wide data frame with a total of seven columns.

Note: You can find the complete documentation for the `pivot_wider()` function .

The following tutorials explain how to use other common functions in the `tidyr` package in R:

ARABPSYCHOLOGY.COM