

How can I use the `pivot_longer()` function to transform all columns in a dataset?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the `pivot_longer()` function to transform all columns in a dataset?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151844>

The pivot_longer() function is a useful tool for transforming data in a dataset. It allows you to convert multiple columns into rows, making it easier to analyze and visualize the data. This function is particularly useful when dealing with datasets that have a wide format, where each column represents a different variable. By using the pivot_longer() function, you can convert these columns into rows, making it easier to compare and manipulate the data. This function can be applied to all columns in a dataset, allowing for a comprehensive transformation of the data.

R: Use pivot_longer() on All Columns

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

If you'd like to use this function to pivot all of the columns in the data frame into a long format, you can use the following syntax:

```
library(tidyr)
```

```
df_long <- pivot_longer(df, cols = everything())
```

Note that the cols argument specifies which columns to pivot and everything() specifies that we want to pivot every column.

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

Example: Use pivot_longer() on All Columns in R

Suppose we have the following data frame in R that shows the number of points scored by various basketball players during three different games:

```
#create data frame
```

```
df <- data.frame(game1=c(20, 30, 33, 19, 22, 24),  
game2=c(12, 15, 19, 19, 20, 14),  
game3=c(22, 29, 18, 12, 10, 11))
```

```
#view data frame
```

```
df
```

```
game1 game2 game3
```

```
1 20 12 22
```

```
2 30 15 29
```

```
3 33 19 18
```

```
4 19 19 12
```

```
5 22 20 10
```

```
6 24 14 11
```

The data frame is currently in a wide format.

However, suppose we'd like to pivot the data frame to a

long format by pivoting all three columns.

We can use the following syntax to do so:

```
library(tidyr)
```

```
#pivot all columns into long data frame
```

```
df_long <- pivot_longer(df, cols = everything())
```

```
#view long data frame
```

```
df_long
```

```
# A tibble: 18 x 2
```

```
name value
```

```
1 game1 20
```

```
2 game2 12
```

```
3 game3 22
```

```
4 game1 30
```

```
5 game2 15
```

```
6 game3 29
```

```
7 game1 33
```

```
8 game2 19
```

```
9 game3 18
```

```
10 game1 19
```

```
11 game2 19
```

12 game3 12
13 game1 22
14 game2 20
15 game3 10
16 game1 24
17 game2 14
18 game3 11

Notice that the column names game1, game2 and game3 are now used as values in a new column called "name" and the values from these original columns are placed into one new column called "value."

The final result is a long data frame.

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