How can I use bind_rows and bind_cols in dplyr?

Authored by

stats writer

May 6, 2024

Bind_rows and bind_cols are functions in the dplyr package that allow for combining and merging data frames in R. Bind_rows merges data frames by rows, combining the rows of multiple data frames into one data frame. Bind_cols merges data frames by columns, combining the columns of multiple data frames into one data frame. These functions are useful for creating new data frames from existing ones, or for adding new rows or columns to existing data frames. They can also be used for joining tables based on common variables. Overall, bind_rows and bind_cols are powerful tools for manipulating and organizing data in R.

Use bind_rows and bind_cols in dplyr (With Examples)

You can use the bind_rows() function from the package in R to bind together two data frames by their rows:

bind_rows(df1, df2, df3, ...)

Similarly, you can use the bind_cols() function from dplyr to bind together two data frames by their columns:

bind_cols(df1, df2, df3, ...)

The following examples show how to use each of these functions in practice.

Example 1: Use bind_rows()

The following code shows how to use the bind_rows() function to bind three data frames together based on

their rows:

```
library(dplyr)

#create data frames
df1 <- data.frame(team=c('A', 'A', 'B', 'B'),
points=c(12, 14, 19, 24))

df2 <- data.frame(team=c('A', 'B', 'C', 'C'),
points=c(8, 17, 22, 25))

df3 <- data.frame(team=c('A', 'B', 'C', 'C'),
assists=c(4, 9, 12, 6))

#row bind together data frames
bind_rows(df1, df2, df3)

team points assists
1 A 12 NA
2 A 14 NA
3 B 19 NA
4 B 24 NA
5 A 8 NA
6 B 17 NA
7 C 22 NA
8 C 25 NA
```

9 A NA 4
10 B NA 9
11 C NA 12
12 C NA 6

Notice that this function automatically fills in missing values with NA if the data frames do not all have the same column names.

Example 2: Use bind_cols()

The following code shows how to use the bind_cols() function to bind three data frames together based on their columns:

```
library(dplyr)

#create data frames
df1 <- data.frame(team=c('A', 'A', 'B', 'B'),
points=c(12, 14, 19, 24))

df2 <- data.frame(team=c('A', 'B', 'C', 'C'),
points=c(8, 17, 22, 25))

df3 <- data.frame(team=c('A', 'B', 'C', 'C'),
assists=c(4, 9, 12, 6))
```

#column bind together data frames
bind_cols(df1, df2, df3)

team points assists steals blocks rebounds
1 A 12 A 8 A 4
2 A 14 B 17 B 9
3 B 19 C 22 C 12
4 B 24 C 25 C 6

Notice that the original columns from each data frame appear in the final data frame in the order that we specified them in the bind_cols() function.

The following tutorials explain how to bind together data frames using the rbind() and cbind() functions from base R:

The following tutorials explain how to perform other common functions in dplyr: