

How can I remove rows with NA values using the dplyr package in R?

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I remove rows with NA values using the dplyr package in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165699>

The dplyr package in R provides efficient and user-friendly functions for data manipulation tasks. One common task is removing rows with missing or NA values from a dataset. This can be achieved by using the dplyr function "filter()" which allows for the selection of specific rows based on certain conditions. By specifying the condition to remove rows with NA values using the "is.na()" function, the filter() function will only keep rows that do not have NA values. This provides a simple and effective way to clean up datasets and ensure accurate analysis.

Remove Rows with NA Values Using dplyr

You can use the following methods from the package to remove rows with NA values:

Method 1: Remove Rows with NA Values in Any Column

```
library(dplyr)
```

```
#remove rows with NA value in any column
```

```
df %>%
```

```
na.omit()
```

Method 2: Remove Rows with NA Values in Certain Columns

```
library(dplyr)
```

```
#remove rows with NA value in 'col1' or 'col2'
```

```
df %>%
```

```
filter_at(vars(col1, col2), all_vars(!is.na(.)))
```

Method 3: Remove Rows with NA Values in One Specific Column

```
library(dplyr)
```

```
#remove rows with NA value in 'col1'
```

```
df %>%
```

```
filter(!is.na(col1))
```

The following examples show how to use these methods in practice with the following data frame:

```
#create data frame with some missing values
```

```
df <- data.frame(team=c('A', 'A', 'B', 'B', 'C'),
```

```
points=c(99, 90, 86, 88, NA),
```

```
assists=c(33, NA, 31, 39, 34),
```

```
rebounds=c(NA, 28, 24, 24, 28))
```

```
#view data frame
```

```
df
```

```
team points assists rebounds
```

```
1 A 99 33 NA
```

```
2 A 90 NA 28
```

```
3 B 86 31 24
```

4 B 88 39 24

5 C NA 34 28

Method 1: Remove Rows with NA Values in Any Column

The following code shows how to remove rows with NA values in any column of the data frame:

```
library(dplyr)
```

```
#remove rows with NA value in any column
```

```
df %>%
```

```
na.omit()
```

```
team points assists rebounds
```

3 B 86 31 24

4 B 88 39 24

The only two rows that are left are the ones without any NA values in any column.

Method 2: Remove Rows with NA Values in Certain Columns

The following code shows how to remove rows with NA values in any column of the data frame:

```
library(dplyr)
```

```
#remove rows with NA value in 'points' or 'assists' columns
```

```
df %>%
```

```
filter_at(vars(points, assists), all_vars(!is.na(.)))
```

```
team points assists rebounds
```

```
1 A 99 33 NA
```

```
2 B 86 31 24
```

```
3 B 88 39 24
```

The only rows left are the ones without any NA values in the 'points' or 'assists' columns.

Method 3: Remove Rows with NA Values in One Specific Column

The following code shows how to remove rows with NA values in one specific column of the data frame:

```
library(dplyr)
```

```
#remove rows with NA value in 'points' column
```

```
df %>%
```

```
filter(!is.na(points))
```

```
team points assists rebounds
```

```
1 A 99 33 NA
```

2 A 90 NA 28

3 B 86 31 24

4 B 88 39 24

The only rows left are the ones without any NA values in the 'points' column.

Additional Resources

The following tutorials explain how to perform other common operations using dplyr: