How can I count the number of NA values in each column using the -R programming language?

Authored by

stats writer

June 26, 2024

Using the R programming language, one can easily count the number of NA (Not Available) values in each column of a dataset. This can be achieved by using the "na.count" function, which calculates the total number of NA values in a given column. This function can be applied to each column individually or to the entire dataset, providing a comprehensive count of all NA values present. This feature is particularly useful for data cleaning and quality control purposes, allowing users to identify and address missing data in their dataset. Overall, the "na.count" function in R simplifies the process of counting NA values and improves the efficiency of data analysis.

R: Count Number of NA Values in Each Column

You can use the following methods to count the number of NA values in each column of a data frame in R:

Method 1: Count NA Values in Each Column Using Base R

sapply(df, function(x) sum(is.na(x)))

Method 2: Count NA Values in Each Column Using dplyr

library(dplyr)

df %>% summarise(across(everything(), ~ sum(is.na(.))))

The following examples show how to use each method with the following data frame in R:

```
#create data frame
df <- data.frame(team=c('A', 'B', 'C', 'D', 'E'),
points=c(99, 90, 86, 88, NA),
assists=c(33, NA, NA, 39, 34),
rebounds=c(30, 28, 24, 24, 28))

#view data frame
df

team points assists rebounds
1 A 99 33 30
2 B 90 NA 28
3 C 86 NA 24
4 D 88 39 24
5 E NA 34 28
```

Example 1: Count NA Values in Each Column Using Base R

The following code shows how to count the number of NA values in each column using the sapply() function from base R:

```
#count NA values in each column
sapply(df, function(x) sum(is.na(x)))

team points assists rebounds
```

0 1 2 0

From the output we can see:

The team column has 0 NA values.The points column has 1 NA value.The assists column has 2 NA values.The rebounds column has 0 NA values.

Note: The sapply() function can be used to apply a function to each column in the data frame. In this example, we apply a function that counts the total number of elements equal to NA.

Example 2: Count NA Values in Each Column Using dplyr

The following code shows how to count the number of NA values in each column using the summarise() function from the package:

```
#count NA values in each column
sapply(df, function(x) sum(is.na(x)))

team points assists rebounds
0 1 2 0
```

From the output we can see:

The team column has 0 NA values.The points column has 1 NA value.The assists column has 2 NA values.The rebounds column has 0 NA values.

These results match the ones from the previous example.

Note: The dplyr method tends to be faster than the base R method when working with extremely large data frames.