How can I use dplyr to find and identify duplicate elements in a dataset?

Authored by

stats writer

June 27, 2024

Dplyr is a powerful tool in the R programming language that allows for efficient data manipulation and analysis. One of its useful functions is identifying and handling duplicate elements within a dataset. By using the "dplyr" package, users can easily find and identify duplicate elements in their dataset by using the "group_by" and "count" functions. These functions group the dataset by specific variables and then count the number of occurrences of each unique combination, making it easy to spot duplicates. This allows for quick and accurate identification of duplicate data points, which can then be further manipulated or removed as needed. Overall, using dplyr for duplicate detection provides a convenient and efficient method for maintaining clean and accurate data in any analysis.

Find Duplicate Elements Using dplyr

You can use the following methods to find duplicate elements in a data frame using dplyr:

Method 1: Display All Duplicate Rows

library(dplyr)

#display all duplicate rows
df %>%
group_by_all() %>%
filter(n()>1) %>%
ungroup()

Method 2: Display Duplicate Count for All Duplicated Rows

library(dplyr)#display duplicate count for all duplicated

rows

df %>%

add_count(col1, col2, col3) %>%

filter(n>1) %>%

distinct()

This tutorial explains how to use each method in practice with the following data frame:

#create data frame

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

position=c('G', 'G', 'F', 'F', 'G', 'G', 'F', 'F'),

points=c(10, 10, 8, 14, 15, 15, 17, 17))

#view data frame

df

team position points

1 A G 10

2 A G 10

3 A F 8

4 A F 14

5 B G 15

6 B G 15

7 B F 17

8 B F 17

Example 1: Display All Duplicate Rows

The following code shows how to display all duplicate rows in the data frame:

```
library(dplyr)

#display all duplicate rows in data frame
df %>%
group_by_all() %>%
filter(n()>1) %>%
ungroup()

# A tibble: 6 x 3
team position points

1 A G 10
2 A G 10
3 B G 15
4 B G 15
5 B F 17
6 B F 17
```

The result is a data frame that contains 6 rows, each of which is a duplicated row.

Note: If you only want to know which rows have duplicate values across specific columns, you could use something like group_by(team) instead to find rows that have duplicate values in the team column only.

Example 2: Display Duplicate Count for All Duplicated Rows

The following code shows how to display the duplicate count for all of the duplicated rows in the data frame:

```
library(dplyr)

#display duplicate count for each row
df %>%
add_count(team, position, points) %>%
filter(n>1) %>%
distinct()
```

```
team position points n
1 A G 10 2
2 B G 15 2
3 B F 17 2
```

The n column displays the total number of duplicates for each row.

For example:

The row with values A, G, and 10 occurs 2 times in the data frame.The row with values B, G, and 15 occurs 2 times in the data frame.The row with values B, F, and 17 occurs 2 times in the data frame.

Note: If you only want to know which rows have duplicate values across specific columns, then only include those specific columns within the add_count() function.