How can I calculate relative frequencies using dplyr?

Authored by

stats writer

April 18, 2024

Dplyr is a useful tool for data manipulation and analysis in R. It provides a streamlined and efficient way to calculate relative frequencies within a dataset. To do so, the user can use the "group_by" function to group the data by a specific variable, and then use the "summarise" function to calculate the frequency of each group. By dividing the frequency of each group by the total number of observations, the user can obtain the relative frequency. This process can be further simplified by using the "mutate" function to add a new column with the calculated relative frequencies. Overall, dplyr offers a convenient and effective method for calculating relative frequencies in R.

Calculate Relative Frequencies Using dplyr

Often you may want to calculate the relative frequencies/proportions of values in one or more columns of a data frame in R.

Fortunately this is easy to do using functions from the dplyr package. This tutorial demonstrates how to use these functions to calculate relative frequencies on the following data frame:

```
#create data frame
df <- data.frame(team = c('A', 'A', 'A', 'B', 'B', 'B', 'B'),
position = c('G', 'F', 'F', 'G', 'G', 'G', 'F'),
points = c(12, 15, 19, 22, 32, 34, 39))

#view data frame
df

team position points
```

1 A G 12

2 A F 15

3 A F 19

4 B G 22

5 B G 32

6 B G 34

7 B F 39

Example 1: Relative Frequency of One Variable

The following code shows how to calculate the relative frequency of each team in the data frame:

```
library(dplyr)

df %>%
group_by(team) %>%
summarise(n = n()) %>%
mutate(freq = n / sum(n))

# A tibble: 2 x 3
team n freq

1 A 3 0.429
2 B 4 0.571
```

This tells us that team A accounts for 42.9% of all rows in the data frame while team B accounts for the remaining 57.1% of rows. Notice that together they add up to 100%.

The Complete Guide: How to Group & Summarize Data in R

Example 2: Relative Frequency of Multiple Variables

The following code shows how to calculate the relative frequency of positions *by* team:

```
library(dplyr)

df %>%
group_by(team, position) %>%
summarise(n = n()) %>%
mutate(freq = n / sum(n))

# A tibble: 4 x 4
# Groups: team
team position n freq

1 A F 2 0.667
2 A G 1 0.333
```

3 B F 1 0.250
4 B G 3 0.750

This tells us that:

66.7% of players on team A are in position F.33.3% of players on team A are in position G.25.0% of players on team A are in position F.75.0% of players on team B are in position G.

How to Use Mutate to Create New Variables in R

Example 3: Display Relative Frequencies as Percentages

The following code shows how to calculate the relative frequency of positions by team and how to display these relative frequencies as percentages:

```
library(dplyr)

df %>%
group_by(team, position) %>%
summarise(n = n()) %>%
mutate(freq = paste0(round(100 * n/sum(n), 0), '%'))

# A tibble: 4 x 4
```

Groups: team

team position n freq

1 A F 2 67%

2 A G 1 33%

3 B F 1 25%

4 B G 3 75%