# “Are all the columns equal in the given dataset?

Authored by

**stats writer**

June 24, 2024

The question "Are all the columns equal in the given dataset? pertains to the comparison of values within each column of a dataset. It seeks to determine if all the columns in the dataset contain the same number of values or if there are discrepancies in the data. This inquiry is important in data analysis as it helps to identify any potential errors or inconsistencies that may affect the accuracy and reliability of the dataset. By ensuring that all columns are equal, the dataset can be deemed to be well-structured and suitable for further analysis and interpretation. Therefore, this question serves as an important quality check in the evaluation of a dataset.

## R: Check if Multiple Columns are Equal

## You can use the following methods to check if multiple columns are equal in a data frame in R:

## Method 1: Check if All Columns Are Equal

```
library(dplyr)

#create new column that checks if all columns are equal
df <- df %>%
rowwise %>%
mutate(match = n_distinct(unlist(cur_data())) == 1) %>%
ungroup()
```

## Method 2: Check if Specific Columns Are Equal

```
library(dplyr)

#create new column that checks if columns 'A', 'C', and
```

**'D' are equal**

```
df_temp <- df %>%
select('A', 'C', 'D') %>%
rowwise %>%
mutate(match = n_distinct(unlist(cur_data())) == 1) %>%
ungroup()

#add new column to existing data frame
df$match <- df_temp$match
```

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

```
#create data frame
df = data.frame(A=c(4, 0, 3, 3, 6, 8, 7),
B=c(4, 2, 3, 5, 6, 4, 7),
C=c(4, 0, 3, 3, 5, 10, 7),
D=c(4, 0, 3, 3, 3, 8, 7))

#view data frame
df
```

**A B C D**

**1 4 4 4 4**

**2 0 2 0 0**

3 3 3 3 3

4 3 5 3 3

5 6 6 5 3

6 8 4 10 8

7 7 7 7 7

### Example 1: Check if All Columns Are Equal

We can use the following syntax to check if the value in every column in the data frame is equal for each row:

```
library(dplyr)

#create new column that checks if all columns are equal
df <- df %>%
rowwise %>%
mutate(match = n_distinct(unlist(cur_data())) == 1) %>%
ungroup()

#view updated data frame
df
```

```
# A tibble: 7 x 5
A B C D match

1 4 4 4 4 TRUE
```

2 0 2 0 0 FALSE

3 3 3 3 3 TRUE

4 3 5 3 3 FALSE

5 6 6 5 3 FALSE

6 8 4 10 8 FALSE

7 7 7 7 7 TRUE

If the value in each column is equal, then the match column returns True.

Otherwise, it returns False.

Note that you can convert True and False values to 1 and 0 by using as.numeric() as follows:

```
library(dplyr)

#create new column that checks if all columns are equal
df <- df %>%
rowwise %>%
mutate(match =
as.numeric(n_distinct(unlist(cur_data())) == 1)) %>%
ungroup()

#view updated data frame
```

```
df

# A tibble: 7 x 5
A B C D match

1 4 4 4 4 1
2 0 2 0 0 0
3 3 3 3 3 1
4 3 5 3 3 0
5 6 6 5 3 0
6 8 4 10 8 0
7 7 7 7 7 1
```

### Example 2: Check if Specific Columns Are Equal

We can use the following syntax to check if the value in columns A, C, and D in the data frame are equal for each row:

```
library(dplyr)

#create new column that checks if columns 'A', 'C', and 'D' are equal
df_temp <- df %>%
select('A', 'C', 'D') %>%
rowwise %>%
```

**mutate(match = n_distinct(unlist(cur_data())) == 1) %>%**
**ungroup()**

**#add new column to existing data frame**
**df$match <- df_temp$match**

**#view updated data frame**
**df**

**A B C D match**
**1 4 4 4 4 TRUE**
**2 0 2 0 0 TRUE**
**3 3 3 3 3 TRUE**
**4 3 5 3 3 TRUE**
**5 6 6 5 3 FALSE**
**6 8 4 10 8 FALSE**
**7 7 7 7 7 TRUE**

**Otherwise, it returns False.**