How can categorical variables be created in R? Can you provide some examples?

Authored by

stats writer

May 1, 2024

Categorical variables are a type of data that represent categories or groups rather than numerical values. In R, categorical variables can be created by using the "factor" function. This function allows for the conversion of a vector of strings or numbers into a factor, which can then be assigned different levels or categories. For example, if we have a vector of genders (male, female) and we want to create a categorical variable, we can use the "factor" function to convert it into a factor with two levels: male and female. Other examples of categorical variables in R could be education level (high school, college, graduate), income level (low, medium, high), or marital status (single, married, divorced). These variables can be useful for data analysis and visualizations, as they allow for the comparison and grouping of data based on different categories.

Create Categorical Variables in R (With Examples)

You can use the following syntax to create a in R:

```
#create categorical variable from scratch
cat_variable <- factor(c('A', 'B', 'C', 'D'))

#create categorical variable (with two possible values) from existing variable
cat_variable <- as.factor(ifelse(existing_variable < 4, 1, 0))

#create categorical variable (with multiple possible values) from existing variable
cat_variable <- as.factor(ifelse(existing_variable < 3, 'A',
ifelse(existing_variable < 4, 'B',
ifelse(existing_variable < 5, 'C',
ifelse(existing_variable < 6, 'D',0)))))
```

The following examples show how to use this syntax in practice.

Example 1: Create a Categorical Variable from Scratch

The following code shows how to create a categorical variable from scratch:

```
#create data frame
df <- data.frame(var1=c(1, 3, 3, 4, 5),
var2=c(7, 7, 8, 3, 2),
var3=c(3, 3, 6, 10, 12),
var4=c(14, 16, 22, 19, 18))

#view data frame
df
var1 var2 var3 var4
1 1 7 3 14
2 3 7 3 16
3 3 8 6 22
4 4 3 10 19
5 5 2 12 18

#add categorical variable named 'type' to data frame
df$type <- factor(c('A', 'B', 'B', 'C', 'D'))
```

#view updated data frame
df

var1 var2 var3 var4 type
1 1 7 3 14 A
2 3 7 3 16 B
3 3 8 6 22 B
4 4 3 10 19 C
5 5 2 12 18 D

Example 2: Create a Categorical Variable (with Two Values) from Existing Variable

The following code shows how to create a categorical variable from an existing variable in a data frame:

```
#create data frame
df <- data.frame(var1=c(1, 3, 3, 4, 5),
var2=c(7, 7, 8, 3, 2),
var3=c(3, 3, 6, 10, 12),
var4=c(14, 16, 22, 19, 18))

#view data frame
df
```

var1 var2 var3 var4
1 1 7 3 14

2 3 7 3 16

3 3 8 6 22

4 4 3 10 19

5 5 2 12 18

#add categorical variable named 'type' using values from 'var4' column

df$type <- as.factor(ifelse(df$var1 < 4, 1, 0))

#view updated data frame

df

var1 var2 var3 var4 type

1 1 7 3 14 1

2 3 7 3 16 1

3 3 8 6 22 1

4 4 3 10 19 0

5 5 2 12 18 0

Using the ifelse() statement, we created a new categorical variable called "type" that takes the following values:

1 if the value in the 'var1' column is less than 4.0 if the value in the 'var1' column is not less than 4.

Example 3: Create a Categorical Variable (with Multiple Values) from Existing Variable

The following code shows how to create a categorical variable (with multiple values) from an existing variable in a data frame:

```
#create data frame
df <- data.frame(var1=c(1, 3, 3, 4, 5),
var2=c(7, 7, 8, 3, 2),
var3=c(3, 3, 6, 10, 12),
var4=c(14, 16, 22, 19, 18))

#view data frame
df
var1 var2 var3 var4
1 1 7 3 14
2 3 7 3 16
3 3 8 6 22
4 4 3 10 19
5 5 2 12 18

#add categorical variable named 'type' using values from 'var4' column
df$type <- as.factor(ifelse(df$var1 < 3, 'A',
ifelse(df$var1 < 4, 'B',
```

```
ifelse(df$var1 < 5, 'C',
ifelse(df$var1 < 6, 'D', 'E')))))

#view updated data frame
df

var1 var2 var3 var4 type
1 1 7 3 14 A
2 3 7 3 16 B
3 3 8 6 22 B
4 4 3 10 19 C
5 5 2 12 18 D
```

Using the ifelse() statement, we created a new categorical variable called "type" that takes the following values:

'A' if the value in the 'var1' column is less than 3.Else,
'B' if the value in the 'var1' column is less than 4.Else,
'C' if the value in the 'var1' column is less than 5.Else,
'D' if the value in the 'var1' column is less than 6.Else,
'E'.