

PRACTICAL ASSIGNMENT - 3

Aim :- Write R program to find the levels of factor of a given vector, also create a factor corresponding to Height of women dataset, which contains height and weights for a sample of a woman.

Theory :- (Brief report about the R language)

factors:

factors in R programming language are data structures that are implemented to categorize the data or represent categorical data and store it on multiple levels.

They can be stored as integers with a corresponding label to every unique integer. The R factors may look similar to character vectors, they are integers & care must be taken while using them as strings. The R factor accepts only a restricted number of distinct values. For example, a data field such as gender may contain values only from female, male or transgender. After a factor is created it only consists of levels that are by default stored alphabetically.

Attributes of factors in R language :-

x: It is the vector that needs to be converted into a factor.

levels: It is a set of distinct values which are given to the input vector x.

Labels: It is a character vector corresponding to the number

For Educational Use Only

of labels:

Exclude: This will mention all the values you want to exclude.

Ordered: This logical attribute decides whether the levels are ordered.

nmax: It will decide the upper limit for the maximum number of levels.

Creating a factor in R programming language:

The command used to create or modify a factor in R language is `factor()` with a vector as input.

The two steps to creating an R factor:

1. Creating a vector and 2. Converting the vector into a factor using function `factor()`.

Example:

```
# Creating a vector for gender
x <- c("female", "male", "male", "female")
print(x)
```

```
# Converting the vector x into a factor
# named gender
gender <- factor(x)
print(gender)
```

For Educational Use Only

Checking for a factor in R:

The function `is.factor()` is used to check whether the variable is a factor & returns "TRUE" if it is a factor.

Example:

```
gender <- factor(c("female", "male", "male", "female"))  
print(is.factor(gender))
```

Factors in Data frame:

The Data frame is similar to a 2D array with the columns containing all the values of one variable and the rows having one set of values from every column. There are four things to remember about data frames:

- column names are compulsory and cannot be empty.
- Unique names should be assigned to each row.
- The data frame's data can be only of three types - factor, numeric and character type.
- The same number of data items must be present in each column.

In R language when we create a data frame, its column is categorical data, & hence a R factor is automatically created on it.

For Educational Use Only

cut() function in R:

cut() function in R programming language is used to divide a numeric vector into different ranges. It is particularly useful when we want to convert a numeric variable into a categorical one by dividing it into intervals or bins.

Syntax:

`cut(x, breaks, labels = NULL)`

where:

x: Name of vector

breaks: Number of breaks to make or vector of break points.

labels: Labels for the resulting bins.

For Educational Use Only