

Foundations of R Software

Lecture 48

Introduction to Statistical Functions

:::

Introduction, Frequencies and Partition Values

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Descriptive statistics:

First hand tools which gives first hand information.

- **Central tendency of data**
- **Variation in data**
- **Structure and shape of data tendency**
- **Relationship study**

Graphical as well as analytical tools are used.

Absolute and relative frequencies:

Suppose there are 10 persons coded into two categories as male (M) and female (F).

M, F, M, F, M, M, M, F, M, M.

Use a_1 and a_2 to refer to male and female categories.

There are 7 male and 3 female persons,
denoted as $n_1 = 7$ and $n_2 = 3$

The number of observations in a particular category is called the absolute frequency.

Absolute and relative frequencies:

The relative frequencies of a_1 and a_2 are

$$f_1 = \frac{n_1}{n_1 + n_2} = \frac{7}{10} = 0.7 = 70\%$$

$$f_2 = \frac{n_2}{n_1 + n_2} = \frac{3}{10} = 0.3 = 30\%$$

This gives us information about the proportions of male and female persons.

Absolute and relative frequencies:

`table(variable)` creates the absolute frequency of the `variable` of the data file.

Enter data as `x`

`table(x)` # absolute frequencies

`table(x)/length(x)` # relative frequencies

Absolute and relative frequencies:

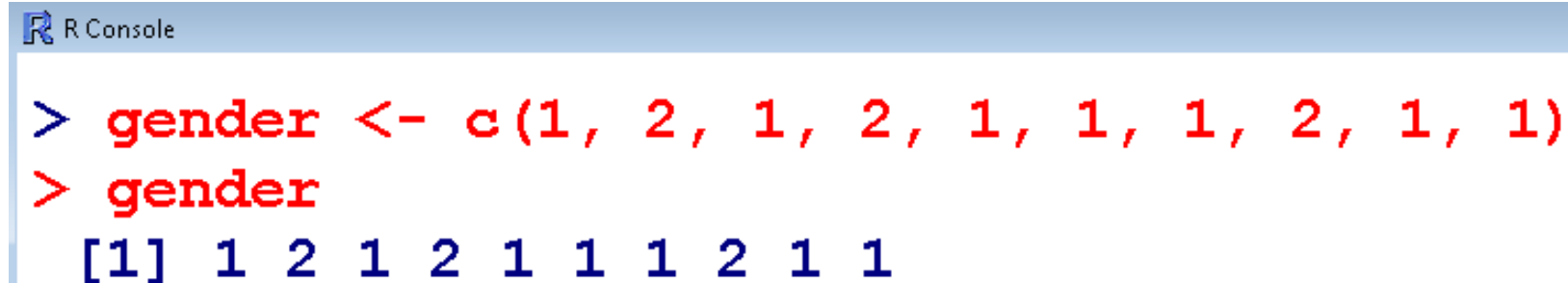
Example: Code the 10 persons by using, say 1 for male (M) and 2 for female (F).

M, F, M, F, M, M, M, F, M, M
1, 2, 1, 2, 1, 1, 1, 2, 1, 1

```
> gender <- c(1, 2, 1, 2, 1, 1, 1, 2, 1, 1)
```

```
> gender
```

```
[1] 1 2 1 2 1 1 1 2 1 1
```

A screenshot of an R console window with a light blue header bar containing the R logo and the text "R Console". The console shows the same R code and output as the previous block, with the code in red and the output in blue.

```
> gender <- c(1, 2, 1, 2, 1, 1, 1, 2, 1, 1)
> gender
[1] 1 2 1 2 1 1 1 2 1 1
```

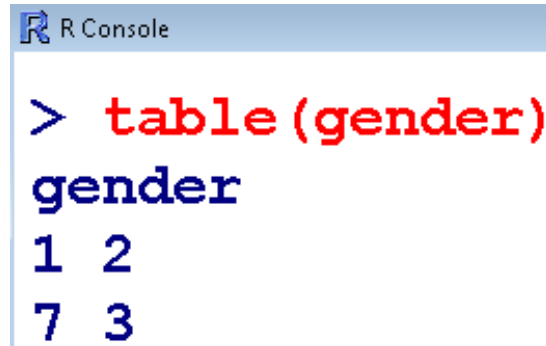
Absolute and relative frequencies:

```
> table(gender) # Absolute frequencies
```

```
gender
```

```
1 2
```

```
7 3
```



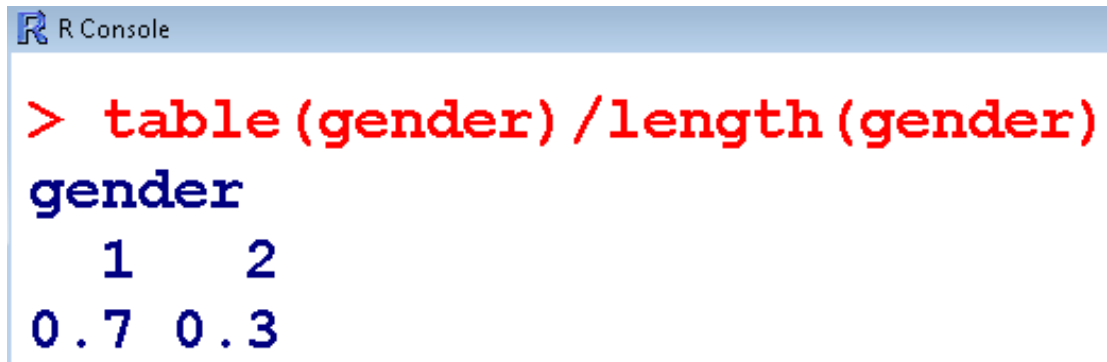
```
R Console  
  
> table(gender)  
gender  
1 2  
7 3
```

```
> table(gender)/length(gender) #Relative freq.
```

```
gender
```

```
1 2
```

```
0.7 0.3
```



```
R Console  
  
> table(gender)/length(gender)  
gender  
1 2  
0.7 0.3
```

Absolute and relative frequencies:

Example:

Consider the following data on pizza home delivery.

- There are three branches (East - coded as 1, West - coded as 2, Central - coded as 3) of the restaurant.

The 100 values from code Directions are as follows:

```
direction =  
c(1,1,2,1,2,3,2,2,3,3,3,1,2,3,2,2,3,1,1,3,3,1,2  
,1,3,3,3,2,2,2,2,1,2,2,1,1,1,3,2,2,1,2,3,2,2,1,  
2,3,3,2,1,2,2,3,1,1,2,1,2,3,2,3,2,2,3,1,2,3,3,3  
,2,1,1,1,2,1,1,2,1,2,3,3,1,2,3,3,2,1,2,3,2,1,3,  
2,2,2,2,3,2,2)
```

Absolute and relative frequencies:

Example:

```
> table(direction) # Absolute frequencies
```

```
direction
```

```
  1   2   3  
28 43 29
```

```
> table(direction)/length(direction) # Relative  
                                         frequencies
```

```
direction
```

```
    1    2    3  
0.28 0.43 0.29
```

Absolute and relative frequencies:

Example:

```
R Console
> direction = c(1,1,2,1,2,3,2,2,3,3,3,1,2,3,2,2,3,1,1,3,3,1,2,1,3,$
> direction
[1] 1 1 2 1 2 3 2 2 3 3 3 1 2 3 2 2 3 1 1 3 3 1 2 1 3 3
[32] 1 2 2 1 1 1 3 2 2 1 2 3 2 2 1 2 3 3 2 1 2 2 3 1 1 2 1 2 3 2 3
[63] 2 2 3 1 2 3 3 3 2 1 1 1 2 1 1 2 1 2 3 3 1 2 3 3 2 1 2 3 2 1 3
[94] 2 2 2 2 3 2 2
> table(direction)
direction
 1  2  3
28 43 29
> table(direction)/length(direction)
direction
 1    2    3
0.28 0.43 0.29
>
```

Partition values:

Such values divides the total frequency given data into required number of partitions.

Quartile: Divides the data into 4 equal parts.

Decile: Divides the data into 10 equal parts.

Percentile: Divides the data into 100 equal parts.

Partition values:

quantile function computes quantiles corresponding to the given probabilities.

The smallest observation corresponds to a probability of 0 and the largest to a probability of 1.

```
quantile(x, ...)
```

```
quantile(x, probs = seq(0, 1, 0.25), ...)
```

Arguments

x numeric vector whose sample quantiles are wanted,

probs numeric vector of probabilities with values in [0, 1].

Partition values:

Example: Marks of 15 students are

```
> marks = c(68, 82, 63, 86, 34, 96, 41, 89, 29,  
51, 75, 77, 56, 59, 42)
```

```
> quantile(marks)
```

0%	25%	50%	75%	100%
29.0	46.5	63.0	79.5	96.0

Default values

```
> quantile(marks, probs=c(0,0.25,0.5,0.75,1))
```

0%	25%	50%	75%	100%
29.0	46.5	63.0	79.5	96.0

Partition values:

Example: Marks of 15 students are

Default values

```
> quantile(marks, probs=c(0,0.25,0.5,0.75,1))
```

0%	25%	50%	75%	100%
29.0	46.5	63.0	79.5	96.0

Defining probabilities

```
> quantile(marks, probs=c(0,0.20,0.4,0.6,0.8,1))
```

0%	20%	40%	60%	80%	100%
29.0	41.8	57.8	70.8	82.8	96.0

Partition values:

Example: Marks of 15 students are

```
R Console
> marks = c(68, 82, 63, 86, 34, 96, 41, 89, 29, 51, 75, 77, 56, 59, 42)
> marks
[1] 68 82 63 86 34 96 41 89 29 51 75 77 56 59 42
>
> quantile(marks)
 0%  25%  50%  75% 100%
29.0 46.5 63.0 79.5 96.0
>
> quantile(marks, probs=c(0,0.25,0.5,0.75,1))
 0%  25%  50%  75% 100%
29.0 46.5 63.0 79.5 96.0
>
> quantile(marks, probs=c(0,0.20,0.4,0.6,0.8,1))
 0%  20%  40%  60%  80% 100%
29.0 41.8 57.8 70.8 82.8 96.0
> |
```