What are the four main functions for the Chi-Square distribution in R and how are they used?

Authored by

stats writer

June 29, 2024

The Chi-Square distribution is a statistical tool used to analyze categorical data and test for associations between variables. In R, there are four main functions for the Chi-Square distribution: dchisq, pchisq, qchisq, and rchisq. The dchisq function calculates the probability density function, which gives the probability of obtaining a specific value or range of values from the Chi-Square distribution. The pchisq function calculates the cumulative distribution function, which gives the probability of obtaining a value less than or equal to a given value from the Chi-Square distribution. The qchisq function calculates the quantile function, which gives the value at a given probability level from the Chi-Square distribution. Lastly, the rchisq function generates random numbers from the Chi-Square distribution. These functions are used to perform various statistical tests, such as the Chi-Square goodness of fit test and the Chi-Square test for independence. They are also used to calculate confidence intervals and perform hypothesis testing in research and data analysis.

The Chi-Square Distribution in R: dchisq, pchisq, qchisq, rchisq

This tutorial explains how to work with the Chi-Square distribution in R using the following functions:

dchisq: returns the value of the Chi-Square probability density function.pchisq: returns the value of the Chi-Square cumulative density function.qchisq: returns the value of the Chi-Square quantile function.rchisq: generates a vector of Chi-Square distributed random variables.

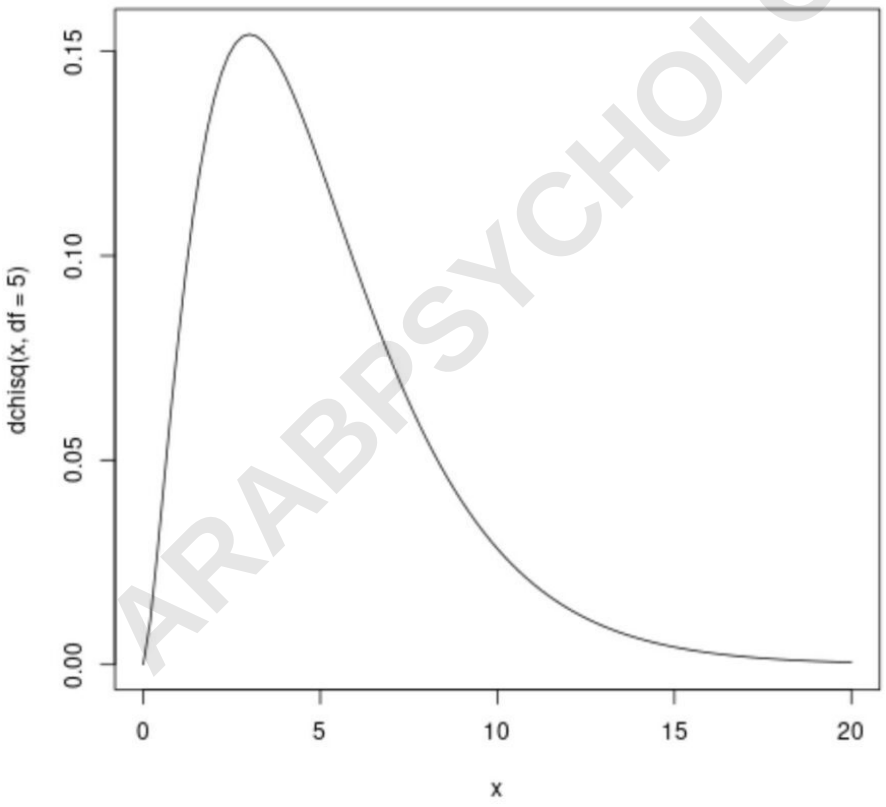
The following examples show how to use each of these functions in practice.

dchisq

We often use the dchisq() function with the curve() function to plot a Chi-Square distribution with a certain number of degrees of freedom.

For example, we can use the following code to plot a Chi-Square distribution with 5 degrees of freedom:

```
#plot Chi_Square distribution with 5 degrees of freedom
curve(dchisq(x, df=5), from=0, to=20)
```



The x-axis shows the values of a Chi-Square test statistic and the y-axis shows the corresponding value

of the probability density function.

Related:

pchisq

We often use the pchisq() function to find the that corresponds to a given Chi-Square test statistic.

For example, suppose we perform a and end up with a test statistic of X2 = 0.86404 with 2 degrees of freedom.

We can use the pchisq() function to find the p-value that corresponds to this test statistic:

```
#calculate p-value for given test statistic with 2 degrees of freedom
1-pchisq(0.86404, df=2)
```

0.6491964

The p-value turns out to be 0.6491964.

We can also confirm this is correct by using the .

qchisq

We often use the qchisq() function to find the Chi-Square critical value that corresponds to a given significance level and degrees of freedom.

For example, we can use the following code to find the Chi-Square critical value that corresponds to a significance level of .05 with 13 degrees of freedom:

qchisq(p=.95, df=13)

22.36203

The critical value turns out to be 22.36203.

We can also confirm this is correct by using the .

rchisq

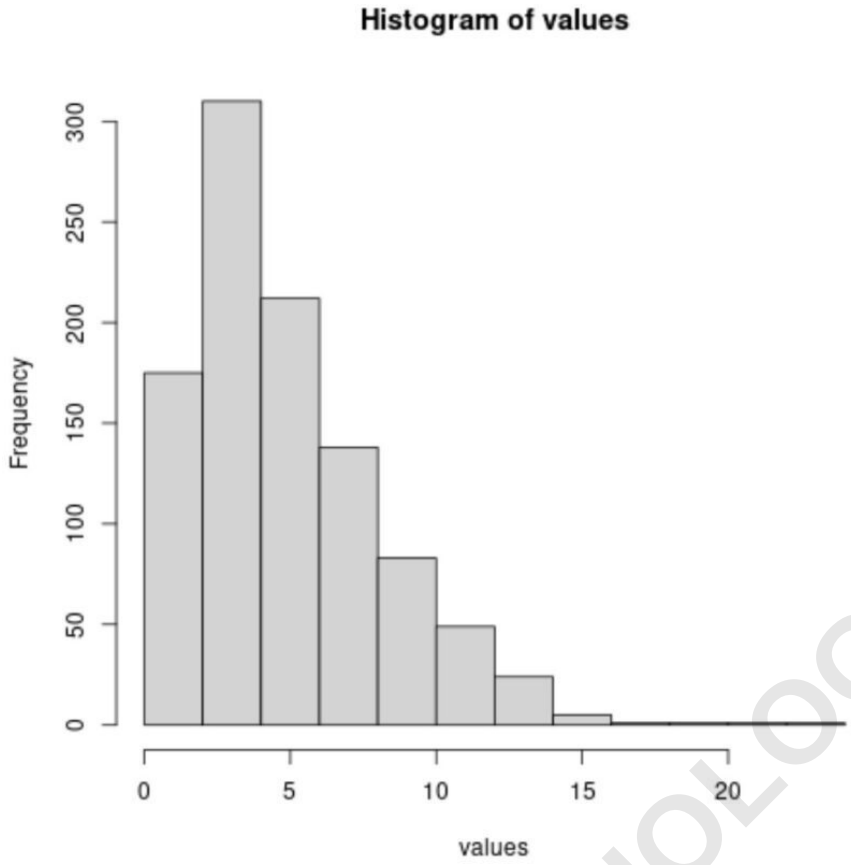
We often use the rchisq() function to generate a list of *n* random values that follow a Chi-Square distribution with a given degrees of freedom.

For example, we can use the following code to generate a list of 1,000 random values that follow a Chi-Square distribution with 5 degrees of freedom:

```
#make this example reproducible
set.seed(0)

#generate 1000 random values that follow Chi-Square dist with df=5
values <- rchisq(n=1000, df=5)

#view first five values
head(values)

8.369701 3.130487 1.985623 5.258747 10.578594 6.360859
```

We can also use the hist() function to generate a histogram to visualize this distribution of values:

```
#create histogram to visualize distribution of values
hist(values)
```



The x-axis shows the data values and the y-axis shows the frequency of those values.

Additional Resources

The following tutorials explains how to work with other distributions in R: