

How can I use the `quantile()` function in R to calculate the desired quantile of a dataset?

Authored by
stats writer

May 3, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the `quantile()` function in R to calculate the desired quantile of a dataset?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=142483>

The `quantile()` function in R is a useful tool for calculating specific quantiles of a dataset. This function takes in a numeric vector or dataset as its input and a probability value between 0 and 1 as its second argument. It then calculates the desired quantile based on the given probability, which represents the percentage of data below that quantile. The output of the `quantile()` function is the value of the quantile for the given dataset. This feature of the `quantile()` function makes it a powerful tool for analyzing datasets and understanding the distribution of data. By utilizing this function, users can easily determine the desired quantile of their dataset and make informed decisions based on the distribution of their data.

Use the quantile() Function in R

In statistics, quantiles are values that divide a ranked dataset into equal groups.

The `quantile()` function in R can be used to calculate sample quantiles of a dataset.

This function uses the following basic syntax:

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

where:

x: Name of vector
probs: Numeric vector of probabilities
na.rm: Whether to remove NA values

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

Example 1: Calculate Quantiles of a Vector

The following code shows how to calculate quantiles of a vector in R:

```
#define vector of data
```

```
data = c(1, 3, 3, 4, 5, 7, 8, 9, 12, 13, 13, 15, 18, 20, 22, 23, 24, 28)
```

```
#calculate quartiles
```

```
quantile(data, probs = seq(0, 1, 1/4))
```

```
0% 25% 50% 75% 100%
```

```
1.0 5.5 12.5 19.5 28.0
```

```
#calculate quintiles
```

```
quantile(data, probs = seq(0, 1, 1/5))
```

```
0% 20% 40% 60% 80% 100%
```

```
1.0 4.4 8.8 13.4 21.2 28.0
```

```
#calculate deciles
```

```
quantile(data, probs = seq(0, 1, 1/10))
```

```
0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
```

```
1.0 3.0 4.4 7.1 8.8 12.5 13.4 17.7 21.2 23.3 28.0
```

```
#calculate random quantiles of interest  
quantile(data, probs = c(.2, .5, .9))
```

```
20% 50% 90%  
4.4 12.5 23.3
```

Example 2: Calculate Quantiles of Columns in Data Frame

The following code shows how to calculate the quantiles of a specific column in a data frame:

```
#create data frame  
df <- data.frame(var1=c(1, 3, 3, 4, 5, 7, 7, 8, 12, 14, 18),  
var2=c(7, 7, 8, 3, 2, 6, 8, 9, 11, 11, 16),  
var3=c(3, 3, 6, 6, 8, 4, 4, 7, 10, 10, 11))
```

```
#calculate quartiles of column 'var2'  
quantile(df$var2, probs = seq(0, 1, 1/4))
```

```
0% 25% 50% 75% 100%  
2.0 6.5 8.0 10.0 16.0
```

We can also use the sapply() function to calculate the quantiles of multiple columns at once:

```
#calculate quartiles of every column
```

```
sapply(df, function(x) quantile(x, probs = seq(0, 1, 1/4)))
```

```
var1 var2 var3  
0% 1.0 2.0 3  
25% 3.5 6.5 4  
50% 7.0 8.0 6  
75% 10.0 10.0 9  
100% 18.0 16.0 11
```

Example 3: Calculate Quantiles by Group

The following code shows how to use functions from the package to calculate quantiles by a grouping variable:

```
library(dplyr)  
  
#define data frame  
df <- data.frame(team=c('A', 'A', 'A', 'A', 'B', 'B', 'B', 'B',  
'C', 'C', 'C'),  
points=c(1, 3, 3, 4, 5, 7, 7, 8, 12, 14, 18))  
  
#define quantiles of interest  
q = c(.25, .5, .75)  
  
#calculate quantiles by grouping variable
```

```
df %>%  
group_by(team) %>%  
summarize(quant25 = quantile(points, probs = q),  
quant50 = quantile(points, probs = q),  
quant75 = quantile(points, probs = q))
```

```
# A tibble: 3 x 4
```

```
team quant25 quant50 quant75
```

```
1 A 2.5 3 3.25
```

```
2 B 6.5 7 7.25
```

```
3 C 13 14 16
```

The following tutorials show how to use the `quantile()` function to calculate other common quantile values: