

# How can multiple histograms be plotted in R using examples?

Authored by  
**stats writer**

May 4, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can multiple histograms be plotted in R using examples?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=142636>

Multiple histograms can be easily and efficiently plotted in R using the "hist()" function. This function allows for the creation of histograms for multiple variables or subsets of a dataset, providing a visual representation of the distribution of each variable. The "hist()" function also allows for customization of the histograms, such as changing the number of bins, adding labels and titles, and adjusting the color and style. For example, to create histograms for three variables in a dataset named "df", the code would be:

```
hist(df$variable1)
hist(df$variable2)
hist(df$variable3)
```

This will result in three separate histograms being plotted on the same graph, allowing for easy comparison and analysis of the variables' distributions. Additionally, the "par()" function can be used to arrange the histograms in a grid or side-by-side for a more organized display. Overall, using the "hist()" function in R allows for efficient and informative visualization of multiple histograms.

## Plot Multiple Histograms in R (With Examples)

**You can use the following syntax to plot multiple histograms on the same chart in base R:**

```
hist(data1, col='red')
hist(data2, col='blue', add=TRUE)
```

**And you can use the following syntax to plot multiple histograms in ggplot2:**

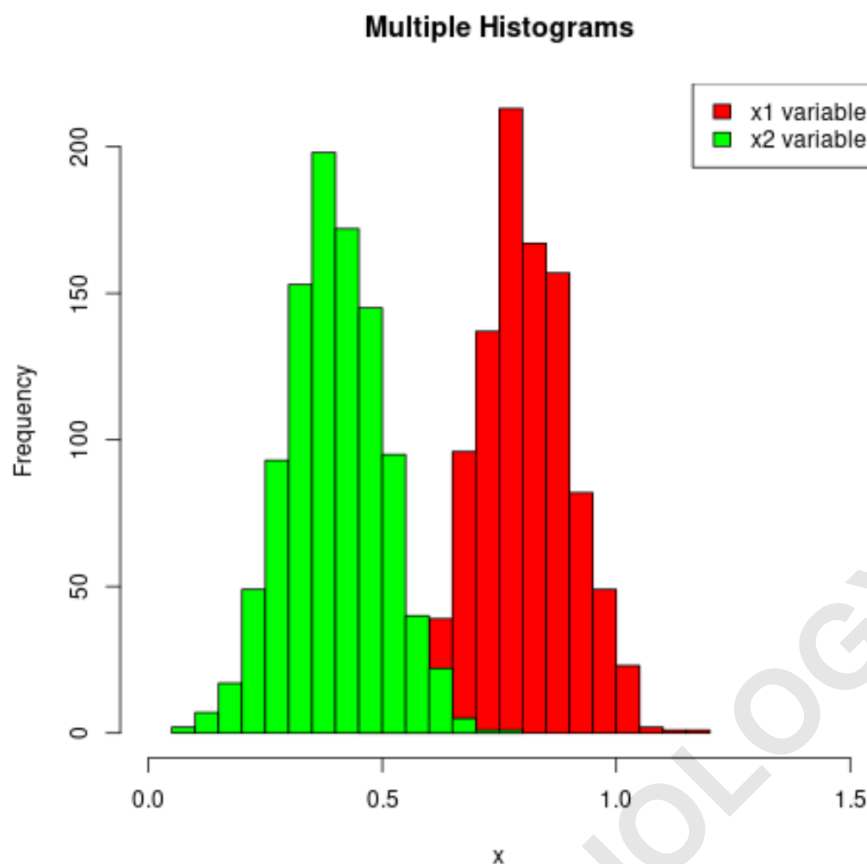
```
ggplot(df, aes(x = x_var, fill = grouping_var)) +
geom_histogram(position = 'identity', alpha = 0.4)
```

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

#### Method 1: Plot Multiple Histograms in Base R

The following code shows how to plot multiple histograms in one plot in base R:

```
#make this example reproducible  
set.seed(1)  
  
#define data  
x1 = rnorm(1000, mean=0.8, sd=0.2)  
x2 = rnorm(1000, mean=0.4, sd=0.1)  
  
#plot two histograms in same graph  
hist(x1, col='red', xlim=c(0, 1.5), main='Multiple  
Histograms', xlab='x')  
hist(x2, col='green', add=TRUE)  
  
#add legend  
legend('topright', c('x1 variable', 'x2 variable'),  
fill=c('red', 'green'))
```



### Method 2: Plot Multiple Histograms in ggplot2

The following code shows how to plot multiple histograms in one plot in R using :

```
library(ggplot2)
```

```
#make this example reproducible
```

```
set.seed(1)
```

```
#create data frame
```

```
df <- data.frame(var = c(rep('x1', 1000), rep('x2', 1000) ),
```

```
value = c(rnorm(1000, mean=0.8, sd=0.1),  
rnorm(1000, mean=0.4, sd=0.1)))
```

```
#view first six rows of data frame
```

```
head(df)
```

```
var value
```

```
1 x1 0.7373546
```

```
2 x1 0.8183643
```

```
3 x1 0.7164371
```

```
4 x1 0.9595281
```

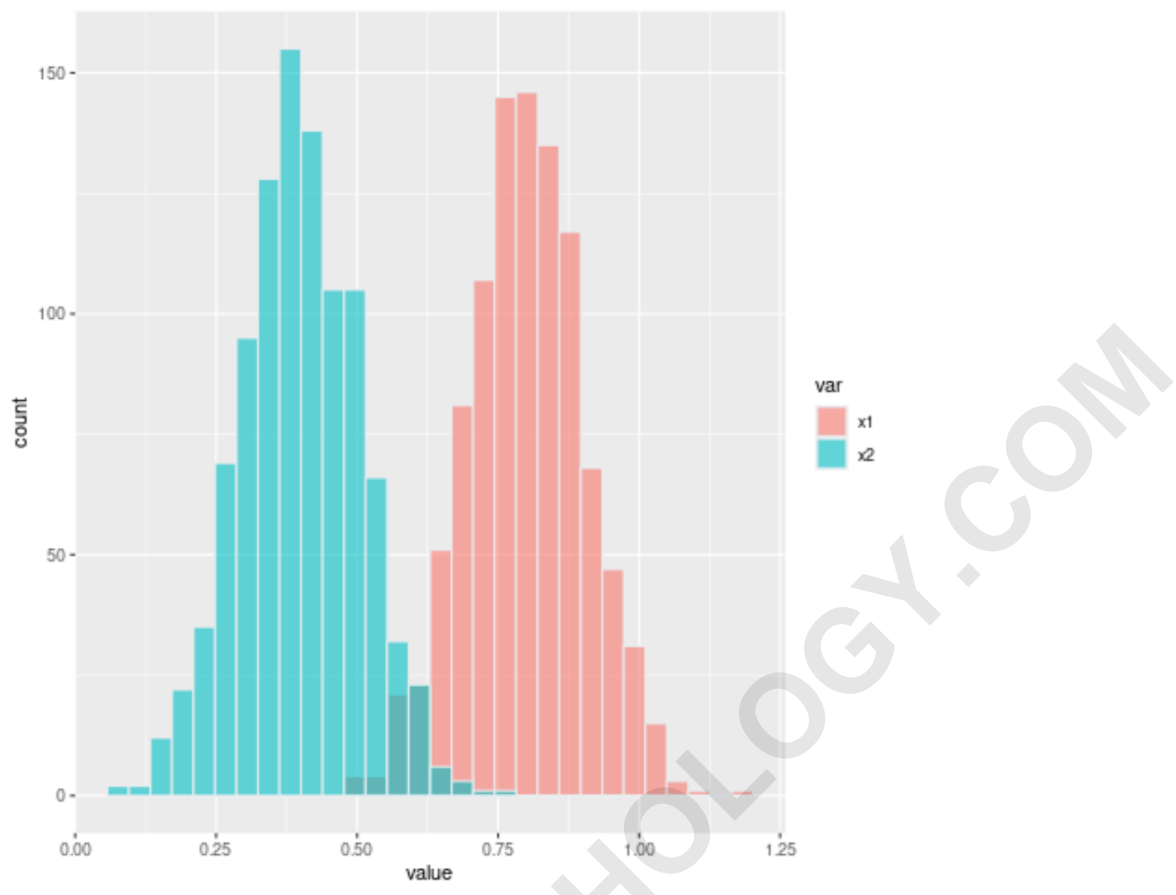
```
5 x1 0.8329508
```

```
6 x1 0.7179532
```

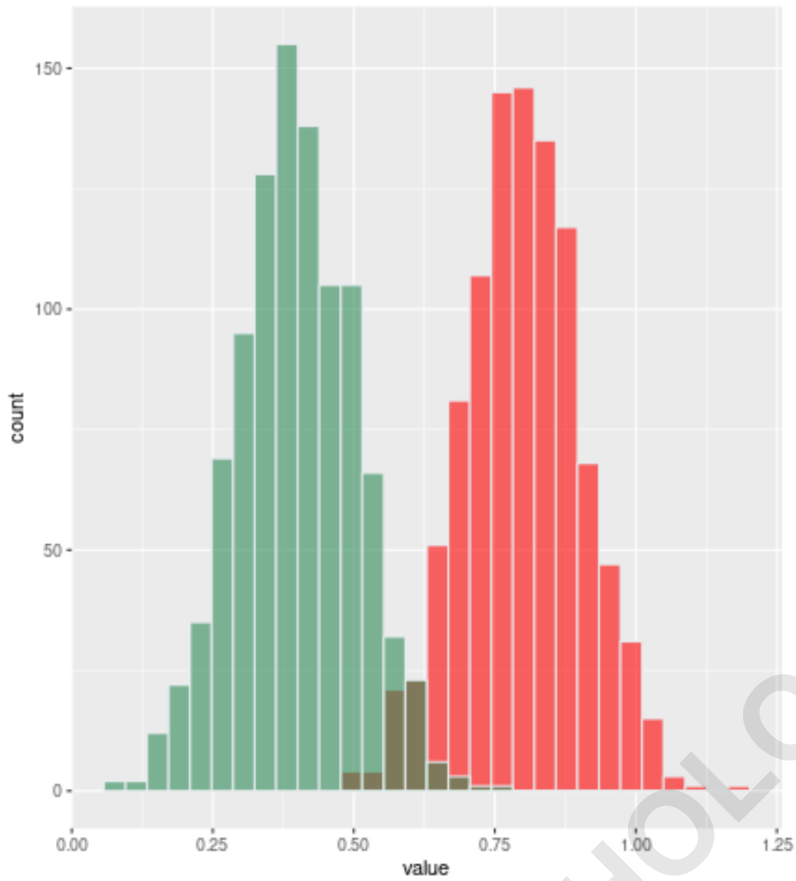
```
#plot multiple histograms
```

```
ggplot(df, aes(x=value, fill=var)) +
```

```
geom_histogram( color='#e9ecef', alpha=0.6,  
position='identity')
```



**You can quickly change the colors of the histograms by using the `scale_fill_manual()` function:**



**The following tutorials explain how to create other common charts in R:**