

# How do I create a Q-Q plot in ggplot2? Can you provide an example?

Authored by  
**stats writer**

June 26, 2024

## RECOMMENDED CITATION

stats writer (2024). *How do I create a Q-Q plot in ggplot2? Can you provide an example?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=153639>

To create a Q-Q plot in ggplot2, start by loading the ggplot2 library and inputting the desired data. Then, use the ggplot() function to create a plot, specifying the desired variables and aesthetic mappings. Next, add the geom\_qq() function to create the Q-Q plot, and customize the plot as needed with additional ggplot2 functions. An example of creating a Q-Q plot in ggplot2 can be seen in the following code:

```
ggplot(data = iris, aes(sample = Sepal.Length)) +  
geom_qq() +  
ggtitle("Q-Q Plot of Sepal Length in Iris Dataset") +  
xlab("Theoretical Quantiles") +  
ylab("Sample Quantiles") +  
theme_bw()
```

This code will create a Q-Q plot of the Sepal Length variable in the iris dataset, with a title, x and y labels, and a black and white theme.

## Create a Q-Q Plot in ggplot2 (With Example)

**A Q-Q plot, short for "quantile-quantile" plot, is used to assess whether or not a set of data potentially came from some theoretical distribution.**

**In most cases, this type of plot is used to determine whether or not a set of data follows a normal distribution.**

**If the data is normally distributed, the points in a Q-Q plot will lie on a straight diagonal line.**

**Conversely, if the points deviate significantly from the straight diagonal line, then it's less likely that the data is**

**normally distributed.**

**To create a Q-Q plot in ggplot2, you can use the `stat_qq()` and `stat_qq_line()` functions as follows:**

```
library(ggplot2)ggplot(df, aes(sample=y)) +  
stat_qq() +  
stat_qq_line()
```

**The following examples show how to use this syntax to create a Q-Q plot in two different scenarios.**

**Example 1: Q-Q Plot for Normal Data**

**The following code shows how to generate a normally distributed dataset with 200 and create a Q-Q plot for the dataset in R:**

```
library(ggplot2)
```

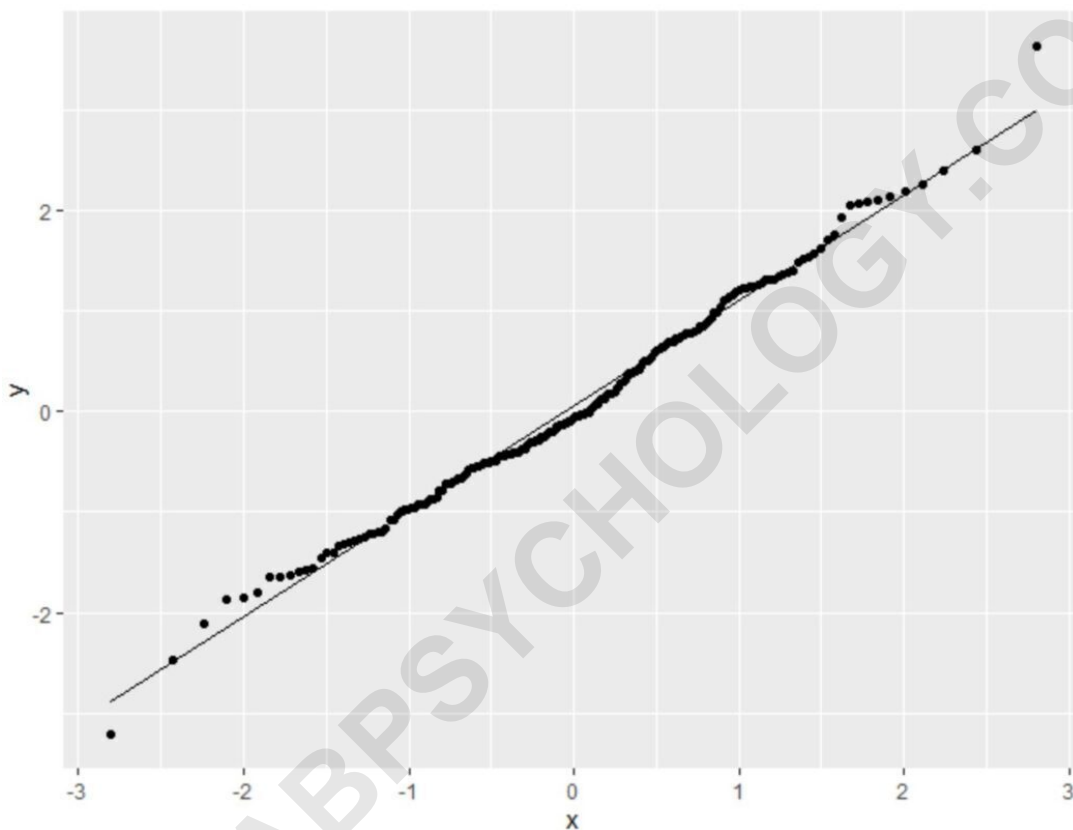
```
#make this example reproducible
```

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

```
#create some fake data that follows a normal  
distribution
```

```
df <- data.frame(y=rnorm(200))
```

```
#create Q-Q plot  
ggplot(df, aes(sample=y)) +  
stat_qq() +  
stat_qq_line()
```



We can see that the points lie mostly along the straight diagonal line with some minor deviations along each of the tails.

Based on this plot, we would assume that this set of data is normally distributed.

**Note that we could also use the color and size arguments to change the color and size of the points in the plot if we'd like to:**

```
library(ggplot2)
```

```
#make this example reproducible
```

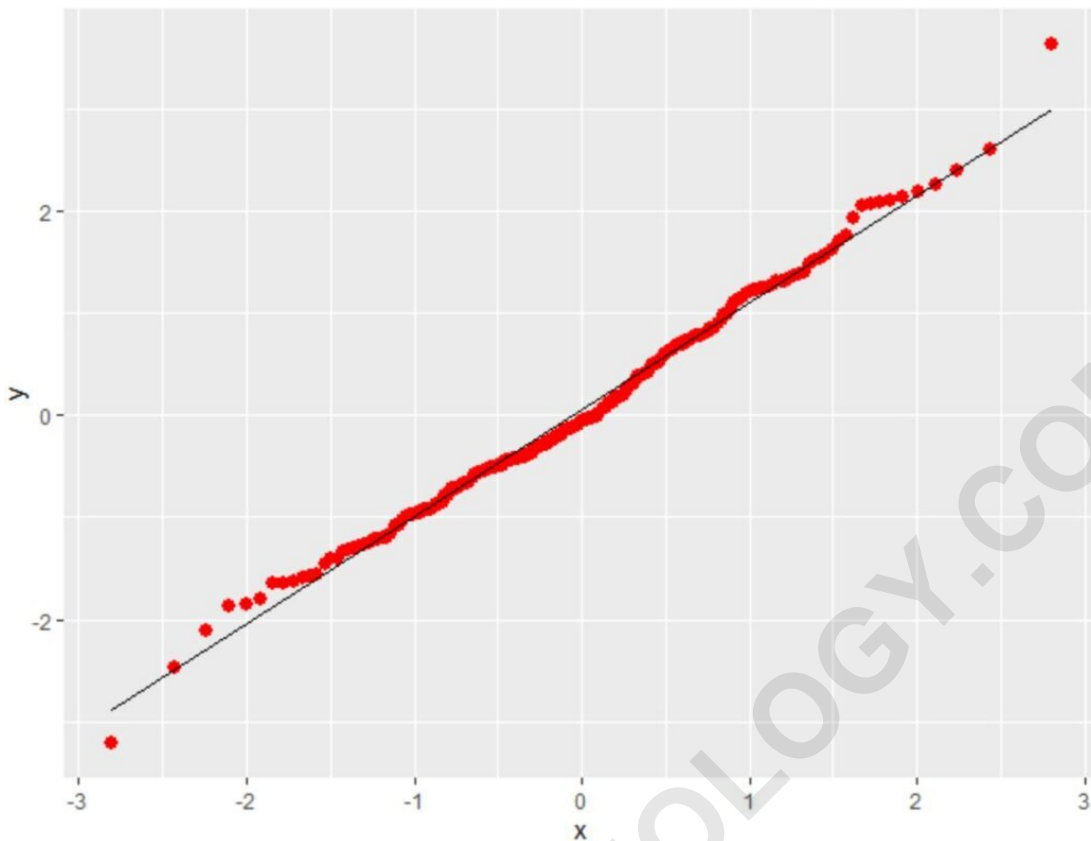
```
set.seed(1)
```

```
#create some fake data that follows a normal distribution
```

```
df <- data.frame(y=rnorm(200))
```

```
#create Q-Q plot
```

```
ggplot(df, aes(sample=y)) +  
stat_qq(size=2.5, color='red') +  
stat_qq_line()
```



### Example 2: Q-Q Plot for Non-Normal Data

**#make this example reproducible**

**set.seed(1)**

**#create some fake data that follows an exponential distribution**

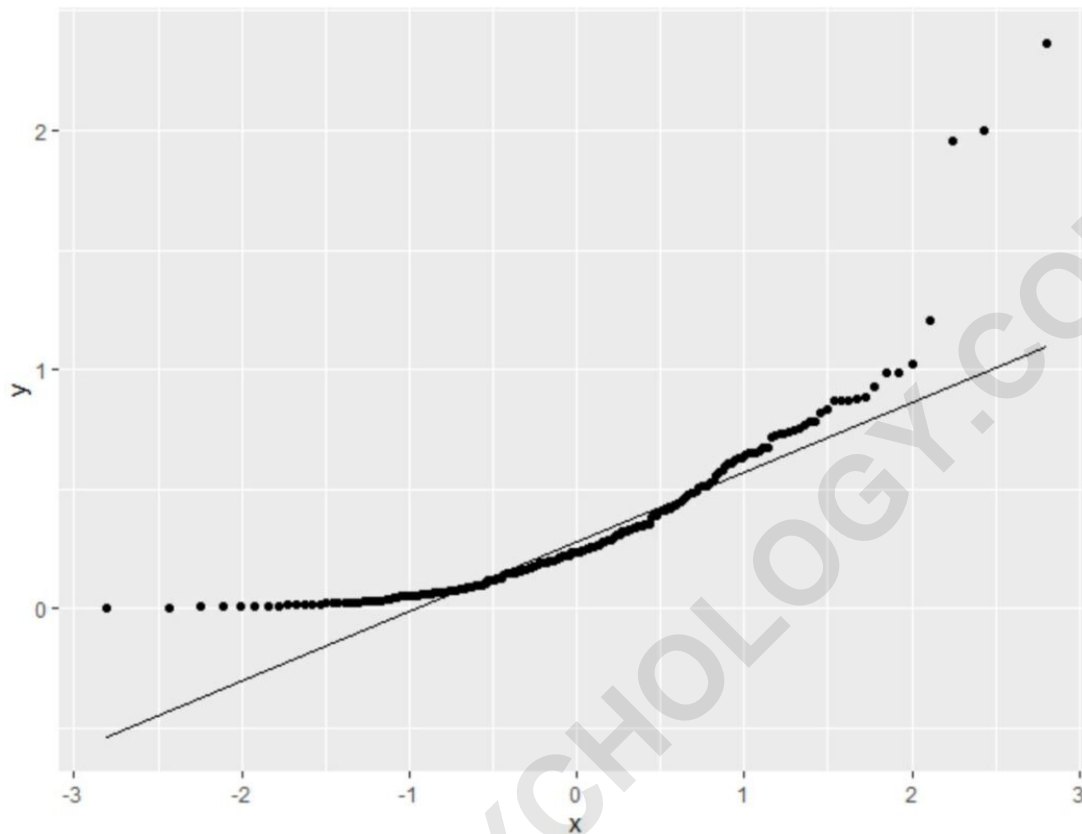
**df <- data.frame(y=rexp(200, rate=3))**

**#create Q-Q plot**

**ggplot(df, aes(sample=y)) +**

**stat\_qq() +**

## stat\_qq\_line()



**We can see that the points deviate significantly from the straight diagonal line. This is a clear indication that the set of data is not normally distributed.**

**This should make sense considering we specified that the data should follow an exponential distribution.**

**The following tutorials explain how to perform other common tasks in ggplot2:**