

# Package: ggQQunif (via r-universe)

September 6, 2024

**Type** Package

**Title** Compare Big Datasets to the Uniform Distribution

**Version** 0.1.5

**Description** A quantile-quantile plot can be used to compare a sample of p-values to the uniform distribution. But when the dataset is big (i.e. > 1e4 p-values), plotting the quantile-quantile plot can be slow. `geom_QQ` uses all the data to calculate the quantiles, but thins it out in a way that focuses on points near zero before plotting to speed up plotting and decrease file size, when vector graphics are stored.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Imports** scales, ggplot2, dplyr

**RoxygenNote** 6.0.1

**Repository** <https://rcorty.r-universe.dev>

**RemoteUrl** <https://github.com/rcorty/ggqqunif>

**RemoteRef** HEAD

**RemoteSha** 4af43e0e78e9f6e1e7d8965144f8ef6c28ae36c6

## Contents

<code>geom_QQ_unif</code> . . . . .	2
<code>scale_x_QQ</code> . . . . .	3

<b>Index</b>	4
--------------	---

---

geom\_QQ\_unif                      *geom\_QQ\_unif*

---

## Description

geom\_QQ\_unif

## Usage

```
geom_QQ_unif(mapping = NULL, data = NULL, geom = "point",
             position = "identity", show.legend = NA, inherit.aes = TRUE,
             keep_first_n = 1024, step = 8, sorted = FALSE)
```

## Arguments

mapping	Default list of aesthetic mappings to use for plot. If not specified, must be supplied in each layer added to the plot.
data	Default dataset to use for plot. If not already a data.frame, will be converted to one by fortify. If not specified, must be supplied in each layer added to the plot.
geom	Specify which geom to plot – 'point' and 'line' are the two choices that make sense.
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification.
keep_first_n	Keep the first n p-values (closest to 0).
step	How many points to plot with each doubling, e.g. between 1/8 and 1/4, between 1/4 and 1/2, etc.
sorted	logical. Are the p-values already sorted?

## Value

a ggplot

## Examples

```
library(ggplot2)
library(dplyr)

set.seed(27599)
d <- data.frame(s = runif(n = 5e3))
```

```
d %>%  
ggplot(mapping = aes(sample = s)) +  
  geom_QQ_unif() +  
  scale_x_QQ() +  
  scale_y_QQ() +  
  theme_minimal()
```

---

scale\_x\_QQ

*scale\_x\_QQ*

---

### **Description**

Scale the axes to show negative log 10 of the p-values.

### **Usage**

scale\_x\_QQ()

scale\_y\_QQ()

# Index

`geom_QQ_unif`, 2

`scale_x_QQ`, 3

`scale_y_QQ(scale_x_QQ)`, 3