

# Package: pplot (via r-universe)

May 17, 2026

**Type** Package

**Title** Chronological and Ordered p-Plots for Empirical Data

**Version** 0.9

**Date** 2025-04-04

**Author** Roland Pfister [aut, cre], Christian Frings [aut]

**Maintainer** Roland Pfister <mail@roland-pfister.net>

**Suggests** MBESS

**Description** Generates chronological and ordered p-plots for data vectors or vectors of p-values. The p-plot visualizes the evolution of the p-value of a significance test across the sampled data. It allows for assessing the consistency of the observed effects, for detecting the presence of potential moderator variables, and for estimating the influence of outlier values on the observed results. For non-significant findings, it can diagnose patterns indicative of underpowered study designs. The p-plot can thus either back the binary accept-vs-reject decision of common null-hypothesis significance tests, or it can qualify this decision and stimulate additional empirical work to arrive at more robust and replicable statistical inferences.

**License** GPL

**NeedsCompilation** no

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

**Date/Publication** 2025-04-07 16:30:06 UTC

**RemoteUrl** <https://github.com/cran/pplot>

**RemoteRef** HEAD

**RemoteSha** 6240ce429778e121a840a10561578b1072b72a7c

## Contents

pplot . . . . .	2
pplot.pvals . . . . .	3
pplot.t.test . . . . .	4

---

pplot

*Generating p-plots*

---

## Description

Generates chronological and ordered p-plots for data vectors or vectors of p-values. The p-plot visualizes the evolution of the p-value of a significance test across the sampled data. It allows for assessing the consistency of the observed effects, for detecting the presence of potential moderator variables, and for estimating the influence of outlier values on the observed results. For non-significant findings, it can diagnose patterns indicative of underpowered study designs. The p-plot can thus either back the binary accept-vs-reject decision of common null-hypothesis significance tests, or it can qualify this decision and stimulate additional empirical work to arrive at more robust and replicable statistical inferences.

## Details

Package: pplot  
Type: Package  
Version: 0.9  
Date: 2025-04-04  
License: GPL-3

This package contains the following functions:

[pplot.pvals](#): Generates p-plot from a series of p-values.

[pplot.t.test](#): Generates chronological or ordered p-plot from a series of data points.

Version history:

**v0.9 | 2025-04-03** | Package release.

## Author(s)

Roland Pfister <mail(at)roland-pfister.net>, Christian Frings;

---

pplot.pvals	<i>Generate p-plot from a series of p-values</i>
-------------	--

---

### Description

Takes a series of p-values and plots them against the corresponding number of observations included in a test. Adds reference lines at the chosen alpha level (horizontal line) and at the latest point in time the p-plot crosses the significance threshold (if any).

### Usage

```
pplot.pvals(xs,  
            ps,  
            xtitle = "Observation",  
            alpha = 0.05,  
            n.sig = TRUE,  
            ylim.p = c(0,1),  
            pcol.alpha = 125)
```

### Arguments

xs	Vector of observations that the p values are computed on.
ps	vector of p-values.
xtitle	Axis title for the x-axis of the p-plot.
alpha	Significance threshold to be used for determining oversampling.
n.sig	Logical: Show vertical line at last crossing of the significance threshold?
ylim.p	Plot limits on the y-axis.
pcol.alpha	Transparency of the data points; set to 255 for solid fill.

### Details

pplot.pvals generates a p-plot from two vectors: xs containing the sample sizes underlying the test results, and ps as the resulting p-values.

### Value

pplot.pvals() returns a single integer with the number of oversampled datasets (-1 for a non-significant test at the full sample size).

### Author(s)

Roland Pfister

### See Also

[pplot.t.test](#);

**Examples**

```

# Show p-plot for significant test (the simulation
# has a power of > 88% to return a significant
# effect).
testdata <- rnorm(64, mean = 0.4, sd = 1)
ps <- matrix(data = NA, nrow = 60, ncol = 1)
for (itest in c(2:64)) {
  ps[itest] <- t.test(testdata[1:itest])$p.value
}
xs <- c(1:64)
pplot.pvals(xs,ps)

# Show p-plot for non-significant test (simulation
# does not include a true effect).
testdata <- rnorm(64, mean = 0.0, sd = 1)
ps <- matrix(data = NA, nrow = 60, ncol = 1)
for (itest in c(2:64)) {
  ps[itest] <- t.test(testdata[1:itest])$p.value
}
xs <- c(1:64)
pplot.pvals(xs,ps)

```

---

pplot.t.test

*Generate p-plot from a series of data points*


---

**Description**

Takes a series of data points and computes either a chronological or ordered p-plot for testing the mean of the data points against 0 with a one-sample t-test.

**Usage**

```

pplot.t.test(data,
              sort = FALSE,
              startindex = 2,
              produce.plot = TRUE,
              xtitle = "Observation",
              plottype = 3,
              alpha = 0.05,
              alternative = "two.sided",
              n.sig = TRUE,
              n.sig.adjust = -1,
              ylim.p = c(0,1),
              ylim.es = -1,
              conf.level = 0.95,
              pcol.alpha = 125)

```

**Arguments**

data	Vector of data points.
sort	Logical. TRUE generates an ordered p-plot, FALSE generates a chronological p-plot.
startindex	Number of observations to begin the p-plot with (minimum 2).
produce.plot	Logical whether to produce plot or not. Setting to FALSE can be used to extract series of p-values as well as effect size estimates from the data.
xtitle	Axis title for the x-axis of the p-plot.
plottype	1 = only p-plot, 2 = only effect size plot, 3 = both plots.
alpha	Significance threshold to be used for determining oversampling.
alternative	Argument passed on to the t.test() function (see the corresponding documentation).
n.sig	Logical: Show vertical line at last crossing of the significance threshold?
n.sig.adjust	-1 = auto, 0 = left-adjusted, 0.5 = centered, 1 = right.
ylim.p	Plot limits on the y-axis of the p-plot. Vector of min, max.
ylim.es	Plot limits on the y-axis of the effect size plot. Either a vector of min and max, or only one value to be used as minimum with maximum being determined as a function of the data.
conf.level	Confidence level for effect size CIs.
pcol.alpha	Transparency of the data points; set to 255 for solid fill.

**Details**

pplot.t.test generates a chronological or ordered p-plot from a vector of data points. It returns a vector of p-values, or a data frame containing p-values and effect size estimates.

**Value**

pplot.t.test() returns either a vector of p-values, a data frame containing effect size estimates, or both (depending on the plot type argument).

**Author(s)**

Roland Pfister

**See Also**

[pplot.pvals](#); [ci.sm](#);

**Examples**

```
# Show p-plot for significant test (the simulation
# has a power of > 88% to return a significant
# effect).
testdata <- rnorm(64, mean = 0.4, sd = 1)
pplot.t.test(testdata)

# Show p-plot for non-significant test (simulation
# does not include a true effect).
testdata <- rnorm(64, mean = 0.0, sd = 1)
pplot.t.test(testdata)
```

# Index

- \* **package**

- pplot, 2

- \* **utilities**

- pplot.pvals, 3

- pplot.t.test, 4

ci.sm, 5

pplot, 2

pplot-package (pplot), 2

pplot.pvals, 2, 3, 5

pplot.t.test, 2, 3, 4