

Package ‘ggstudent’

May 8, 2026

Type Package

Title Continuous Confidence Interval Plots using t-Distribution

Version 0.1.2

License GPL (>= 2)

Description Provides an extension to 'ggplot2' (Wickham, 2016, <[doi:10.1007/978-3-319-24277-4](https://doi.org/10.1007/978-3-319-24277-4)>) for creating two types of continuous confidence interval plots (Violin CI and Gradient CI plots), typically for the sample mean. These plots contain multiple user-defined confidence areas with varying colours, defined by the underlying t-distribution used to compute standard confidence intervals for the mean of the normal distribution when the variance is unknown. Two types of plots are available, a gradient plot with rectangular areas, and a violin plot where the shape (horizontal width) is defined by the probability density function of the t-distribution. These visualizations are studied in (Helske, Helske, Cooper, Ynnerman, and Besancon, 2021) <[doi:10.1109/TVCG.2021.3073466](https://doi.org/10.1109/TVCG.2021.3073466)>.

Encoding UTF-8

BugReports <https://github.com/helske/ggstudent/issues>

URL <https://github.com/helske/ggstudent>

Depends R (>= 3.1.0)

Imports dplyr, ggplot2, stats

Suggests scales

RoxygenNote 7.3.2

NeedsCompilation no

Author Jouni Helske [aut, cre] (ORCID:
<<https://orcid.org/0000-0001-7130-793X>>)

Maintainer Jouni Helske <jouni.helske@iki.fi>

Repository CRAN

Date/Publication 2025-04-09 07:40:14 UTC

Contents

geom_student 2

Index 4

geom_student	<i>Student CI plot</i>
--------------	------------------------

Description

A Student CI plot (or Violin CI plot) is a mirrored density plot similar to violin plot but instead of kernel density estimate it is based on the density of the t-distribution. It can be thought of as a continuous "confidence interval density" (hence the name), which could reduce the dichotomous interpretations due to a fixed confidence level. `geom_student` can also be used to draw Gradient CI plots (using argument `type`), which replaces the violin shaped density with a rectangle.

Usage

```
geom_student(
  mapping = NULL,
  data = NULL,
  position = "identity",
  width = 0.25,
  type = "density",
  scale = TRUE,
  draw_lines = NULL,
  draw_mean = TRUE,
  show.legend = NA,
  inherit.aes = TRUE,
  ...
)
```

Arguments

<code>mapping</code>	Set of aesthetic mappings. See <code>[ggplot2::layer()]</code> for details.
<code>data</code>	The data to be displayed in this layer. See <code>[ggplot2::layer()]</code> for details.
<code>position</code>	A position adjustment to use on the data for this layer. See <code>[ggplot2::layer()]</code> for details.
<code>width</code>	Scaling parameter for the width of the violin/rectangle.
<code>type</code>	Type of the plot. The default is "density" which draws violin style density plot, whereas "box" draws a rectangle shaped gradient plot.
<code>scale</code>	If "TRUE" (default), violins/rectangles are scaled according to the maximum width of the groups ($\max(dt(\theta, df) / se)$).
<code>draw_lines</code>	If not NULL (default), draw horizontal lines at the given quantiles of the density estimate.

draw_mean	If TRUE (default), draw horizontal line at mean.
show.legend	logical. Should this layer be included in the legends? See [ggplot2::layer()] for details.
inherit.aes	If 'FALSE', overrides the default aesthetics. See [ggplot2::layer()] for details.
...	Other arguments passed to [ggplot2::layer()], such as fixed aesthetics.

Value

A ggplot object.

References

Helske, J., Helske, S., Cooper, M., Ynnerman, A., & Besancon, L. (2021). Can visualization alleviate dichotomous thinking? Effects of visual representations on the cliff effect. *IEEE Transactions on Visualization and Computer Graphics*, 27(8), 3397-3409 doi: 10.1109/TVCG.2021.3073466

Examples

```
library("dplyr")
library("ggplot2")
library("scales")

ci_levels <- c(0.999, 0.95, 0.9, 0.8, 0.5)
n <- length(ci_levels)
ci_levels <- factor(ci_levels, levels = ci_levels)
PlantGrowth %>% dplyr::group_by(group) %>%
  dplyr::summarise(
    mean = mean(weight),
    df = dplyr::n() - 1,
    se = sd(weight)/sqrt(df + 1)) %>%
  dplyr::full_join(
    data.frame(group =
      rep(levels(PlantGrowth$group), each = n),
      level = ci_levels), by = "group") -> d

p <- ggplot(data = d, aes(group)) +
  geom_student(aes(mean = mean, se = se, df = df,
    level = level, fill = level), draw_lines = c(0.95, 0.5))
p
g <- scales::seq_gradient_pal("#e5f5f9", "#2ca25f")
p + scale_fill_manual(values=g(seq(0,1,length = n))) + theme_bw()

p2 <- ggplot(data = d, aes(group)) +
  geom_student(aes(mean = mean, se = se, df = df,
    level = level, fill = level), type = "box", draw_lines = c(0.95, 0.5))
p2
```

Index

geom_student, [2](#)