

# Package ‘gghighlight’

October 15, 2022

**Type** Package

**Title** Highlight Lines and Points in 'ggplot2'

**Version** 0.4.0

**Description** Make it easier to explore data with highlights.

**License** MIT + file LICENSE

**URL** <https://yutannihilation.github.io/gghighlight/>, <https://github.com/yutannihilation/gghighlight/>

**BugReports** <https://github.com/yutannihilation/gghighlight/issues>

**Depends** ggplot2 (>= 3.3.6),  
R (>= 3.4.0)

**Imports** dplyr (>= 1.0.4),  
ggrepel,  
lifecycle,  
purrr,  
rlang,  
tibble

**Suggests** geomtextpath,  
knitr,  
rmarkdown,  
sf,  
testthat,  
vdiff (>= 0.3.0)

**VignetteBuilder** knitr

**RdMacros** lifecycle

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.1

**Config/testthat/edition** 3

## R topics documented:

gghighlight-package	2
gghighlight	2

Index	4
-------	---

---

gghighlight-package	<i>Highlight Lines and Points in 'ggplot2'</i>
---------------------	--

---

## Description

Make it easier to explore data with highlights.

---

gghighlight	<i>Highlight Data With Predicate</i>
-------------	--------------------------------------

---

## Description

### [Maturing]

`gghighlight()` highlights (almost) any geoms according to the given predicates.

## Usage

```
gghighlight(
  ...,
  n = NULL,
  max_highlight = 5L,
  unhighlighted_params = list(),
  use_group_by = NULL,
  use_direct_label = NULL,
  label_key = NULL,
  label_params = list(fill = "white"),
  keep_scales = FALSE,
  calculate_per_facet = FALSE,
  line_label_type = c("ggrepel_label", "ggrepel_text", "text_path", "label_path",
    "sec_axis"),
  unhighlighted_colour = NULL
)
```

## Arguments

<code>...</code>	Expressions to filter data, which is passed to <code>dplyr::filter()</code> .
<code>n</code>	Number of layers to clone.
<code>max_highlight</code>	Max number of series to highlight.
<code>unhighlighted_params</code>	Aesthetics (e.g. colour, fill, and size) for unhighlighted geoms. Specifying colour = NULL or fill = NULL will preserve the original colour.
<code>use_group_by</code>	If TRUE, use <code>dplyr::group_by()</code> to evaluate predicate.
<code>use_direct_label</code>	If TRUE, add labels directly on the plot instead of using a legend.
<code>label_key</code>	Column name for label aesthetics.
<code>label_params</code>	A list of parameters, which is passed to <code>ggrepel::geom_label_repel()</code> .

`keep_scales` If TRUE, keep the original data with `ggplot2::geom_blank()` so that the highlighted plot has the same scale with the data.

`calculate_per_facet` (Experimental) If TRUE, include the facet variables to calculate the grouping; in other words, highlighting is done on each facet individually.

`line_label_type` (Experimental) Method to add labels (or texts) on the highlighted lines.

"ggrepel\_label" Use `ggrepel::geom_label_repel()`.

"ggrepel\_text" Use `ggrepel::geom_text_repel()`.

"text\_path" Use `geomtextpath::geom_textline()` for lines and `geomtextpath::geom_textpath()` for paths.

"label\_path" Use `geomtextpath::geom_labelline()` for lines and `geomtextpath::geom_labelpath()` for paths.

"sec\_axis" Use secondary axis. Please refer to [Simon Jackson's blog post](#) for the trick.

`unhighlighted_colour` (Deprecated) Colour for unhighlighted geoms.

## Examples

```
d <- data.frame(
  idx = c(1, 1, 1, 2, 2, 2, 3, 3, 3),
  value = c(1, 2, 3, 10, 11, 12, 9, 10, 11),
  category = rep(c("a", "b", "c"), 3),
  stringsAsFactors = FALSE
)

# Highlight the lines whose max values are larger than 10
ggplot(d, aes(idx, value, colour = category)) +
  geom_line() + gghighlight(max(value) > 10)

# Highlight the points whose values are larger than 10
ggplot(d, aes(idx, value)) +
  geom_point() +
  gghighlight(value > 10, label_key = category)

# Specify the styles for unhighlighted layer
ggplot(d, aes(idx, value, colour = category)) +
  geom_line(linewidth = 5) +
  gghighlight(max(value) > 10,
    unhighlighted_params = list(linewidth = 1)
  )
```

# Index

`dplyr::filter()`, [2](#)  
`dplyr::group_by()`, [2](#)  
  
`geomtextpath::geom_labelline()`, [3](#)  
`geomtextpath::geom_labelpath()`, [3](#)  
`geomtextpath::geom_textline()`, [3](#)  
`geomtextpath::geom_textpath()`, [3](#)  
`gghighlight`, [2](#)  
`gghighlight`-package, [2](#)  
`ggplot2::geom_blank()`, [3](#)  
`ggrepel::geom_label_repel()`, [2](#), [3](#)  
`ggrepel::geom_text_repel()`, [3](#)