

Package ‘scplot’

March 1, 2025

Type Package

Title Plot Function for Single-Case Data Frames

Version 0.5.1

Date 2025-03-01

Description Add-on for the 'scan' package that creates plots from single-case data frames ('scdf'). It includes functions for styling single-case plots, adding phase-based lines to indicate various statistical parameters, and predefined themes for presentations and publications. More information and in depth examples can be found in the online book ``Analyzing Single-Case Data with R and 'scan'` by Jürgen Wilbert (2025) <<https://jazznbass.github.io/scan-Book/>>.

Depends R (>= 4.1.0)

Imports ggplot2, scan (>= 0.61.0), stats, utils, mblm

Suggests rmarkdown, knitr

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Juergen Wilbert [aut, cre] (<<https://orcid.org/0000-0002-8392-2873>>)

Maintainer Juergen Wilbert <juergen.wilbert@uni-muenster.de>

Repository CRAN

Date/Publication 2025-03-01 13:10:03 UTC

Contents

scplot-package	2
add_arrow	3
add_grid	4
add_labels	5
add_legend	6
add_line	7

add_marks	8
add_ridge	9
add_statline	10
add_text	11
add_title	12
as_ggplot	12
element_point	13
new_theme	13
scplot.scdf	14
scplot.sc_hplm	15
scplot.sc_rand	15
scplot.sc_tauu	16
set_background	17
set_base_text	17
set_casenames	18
set_dateline	18
set_phasenames	20
set_separator	20
set_theme	21
set_theme_element	21
set_xaxis	23
set_xlabel	24
Index	26

 scplot-package

Single-Case Data Plots

Description

A collection of procedures for visualizing single-case data. It is an add-on package for the scan package.

Author(s)

Juergen Wilbert [aut, cre]

add_arrow	<i>Add arrows to an splot</i>
-----------	-------------------------------

Description

Add arrows to an splot

Usage

```
add_arrow(
  object,
  case = 1,
  x0,
  y0,
  x1,
  y1,
  color = "black",
  angle = 30,
  length = unit(5, "points"),
  type = "open",
  ends = "last",
  linewidth = 0.7
)
```

Arguments

object	An splot object (class <code>splot</code>) returned from the <code>splot()</code> function.
case	Numerical vector with the case number or character string. <code>case = "all"</code> for all cases.
x0	Origin x position of the line.
y0	Origin y position of the line.
x1	End x position of the line.
y1	End y position of the line.
color	A character string or a number defining the color of an element.
angle	Angle (in $[0,360]$)
length	Size of the arrow angels.
type	One of "open" or "closed" indicating whether the arrow head should be a closed triangle.
ends	One of "last", "first", or "both", indicating which ends of the line to draw arrow heads.
linewidth	A number with the width of the line.

Value

An object of class `splot` (see [splot\(\)](#)) with added element lines.

Examples

```
data(exampleAB, package = "scan")
p1 <- scplot(exampleAB$Anja) |>
  add_arrow(case = 1, 2, 70, 6, 55, color = "darkred")
```

add_grid	<i>Add grid to an scplot</i>
----------	------------------------------

Description

Add grid to an scplot

Usage

```
add_grid(object, ...)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
...	Line arguments (see element_line())

Value

An object of class scplot (see [scplot\(\)](#)).

See Also

[element_line\(\)](#)

Examples

```
data(exampleAB, package = "scan")
p1 <- scplot(exampleAB$Anja) |>
  set_theme("minimal") |>
  add_grid(color = "grey70")
```

add_labels	<i>Add value labels to an splot</i>
------------	-------------------------------------

Description

Add value labels to an splot

Usage

```
add_labels(  
  object,  
  nudge_y = 5,  
  nudge_x = 0,  
  round = NULL,  
  text = list(),  
  background = list(),  
  variable = ".dvar",  
  padding = NULL  
)
```

Arguments

object	An splot object (class <code>splot</code>) returned from the <code>splot()</code> function.
nudge_y	Offset on the y-axis.
nudge_x	Offset on the x-axis.
round	Number of digits of the labels.
text	List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "l... See element_text() .
background	A list with background styling arguments (fill, color, size, linetype).
variable	Name of the dataline variable to apply the style.
padding	Padding size around text.

Value

An object of class `splot` (see [splot\(\)](#)) with added/changed element labels.

add_legend	<i>Add a legend to an scplot</i>
------------	----------------------------------

Description

Add a legend to an scplot

Usage

```
add_legend(
  object,
  labels = NULL,
  section_labels = c("Lines", "Phases"),
  case = 1,
  position = "right",
  datalines = TRUE,
  statlines = TRUE,
  phases = TRUE,
  title = NULL,
  text = NULL,
  background = NULL
)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
labels	A Character vector with text labels.
section_labels	A character vector of length two. The labels for the lines section and phase section.
case	Numerical vector with the case number or character string. case = "all" for all cases.
position	The position ("none", "left", "right", "bottom", "top", or two-element numeric vector)
datalines	If TRUE, a legend for the datalines is generated.
statlines	If TRUE, a legend for the statlines is generated.
phases	If TRUE, a legend for the phases is generated. Note that you also have to set the set_panel argument (e.g., set_panel(fill = c("lightblue", "grey80"))).
title	A list with text style parameters for the title.
text	List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "l"). See element_text() .
background	A list with background styling arguments (fill, color, size, linetype).

Value

An object of class scplot (see [scplot\(\)](#)) with changed element legend.

Examples

```

data(exampleAB_add, package = "scan")
splot(exampleAB_add) |>
  set_dataline("depression") |>
  add_statline("mean") |>
  add_legend()

splot(exampleAB_add) |>
  set_dataline(label = "Psychological Wellbeing") |>
  set_dataline("depression", color = "darkblue", label = "Depression") |>
  add_statline("mean", label = "Wellbeing mean") |>
  add_statline("mean", variable = "depression", label = "Depression mean") |>
  set_phasenames(color = NA) |>
  set_panel(fill = c("lightblue", "grey80")) |>
  add_legend(
    position = "left",
    section_labels = c("Variables", "Section"),
    title = list(color = "brown", size = 10, face = 2),
    text = list(color = "darkgreen", size = 10, face = 2),
    background = list(color = "lightgrey")
  )

```

add_line

Add line to an splot

Description

Add line to an splot

Usage

```

add_line(
  object,
  case = 1,
  x0 = NULL,
  y0 = NULL,
  x1 = NULL,
  y1 = NULL,
  hline = NULL,
  vline = NULL,
  color = "black",
  linewidth = 0.7,
  linetype = "solid"
)

```

Arguments

object An splot object (class `splot`) returned from the `splot()` function.

case	Numerical vector with the case number or character string. case = "all" for all cases.
x0	Origin x position of the line.
y0	Origin y position of the line.
x1	End x position of the line.
y1	End y position of the line.
hline	y position of horizontal line.
vline	x position of vertical line.
color	A character string or a number defining the color of an element.
linewidth	A number with the width of the line.
linetype	A character string with the line type: "solid", "dashed", "dotted"

Value

An object of class `scplot` (see `scplot()`) with added element lines.

Examples

```
data(exampleAB, package = "scan")
p1 <- scplot(exampleAB$Anja) |>
  add_line(hline = 70, color = "darkred") |>
  add_line(vline = 3, color = "blue") |>
  add_line(x0 = 1, y0 = 70, x1 = 4, y1 = 80, color = "green")
```

add_marks

Add marks to an scplot

Description

Add marks to an `scplot`

Usage

```
add_marks(
  object,
  case = 1,
  positions,
  color = "red",
  size = 1,
  shape = 1,
  variable = ".dvar"
)
```


Arguments

object	An scplot object (class scplot) returned from the scplot() function.
case	Numerical vector with the case number or character string. case = "all" for all cases.
positions	Either a vector indicating the points to be highlighted or a character string with a logical expression (e.g. values < mean(values))
color	A character string or a number defining the color of an element.
size	Text size relative to the base text size.
shape	Number. See pch graphical parameter on par help page par() .
variable	Name of the dataline variable to apply the style.

Details

If positions is an object returned from an outlier analysis outlier(), the corresponding outliers are marked.

Value

An object of class scplot (see [scplot\(\)](#)) with changed element marks.

Examples

```
library(scan)
p1 <- scplot(exampleA1B1A2B2$Moritz) |> add_marks(positions = c(1,5,10,14))
p1 <- scplot(Huber2014) |> add_marks(positions = outlier(Huber2014))
```

add_ridge

Add a ridge to an scplot

Description

Add a ridge to an scplot

Usage

```
add_ridge(object, color = "grey98", variable = ".dvar")
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
color	A character string or a number defining the color of an element.
variable	Name of the dataline variable to apply the style.

Value

An object of class scplot (see [scplot\(\)](#)) with changed element ridges.

add_statline *Add a statline to an splot*

Description

Add a statline to an splot

Usage

```
add_statline(
  object,
  stat = c("mean", "median", "min", "max", "quantile", "sd", "mad", "trend", "trendA",
    "trendA theil-sen", "moving mean", "moving median", "loreg", "lowess", "loess"),
  phase = NULL,
  color = NULL,
  linewidth = NULL,
  linetype = NULL,
  variable = NULL,
  label = NULL,
  ...
)
```

Arguments

object	An splot object (class <code>splot</code>) returned from the <code>splot()</code> function.
stat	A character string for defining a statistical line or curve to be plotted.
phase	Either a numeric or a character vector specifying the reference phase (see details).
color	A character string or a number defining the color of an element.
linewidth	A number with the width of the line.
linetype	A character string with the line type: "solid", "dashed", "dotted"
variable	Name of the dataline variable to apply the style.
label	A character string which is used to set the label in a legend.
...	additional parameters passed to the statistical function.

Details

The phase argument defines the reference phase for some statistical functions ("median", "mean", "min", "max", "quantile"). The default is NULL which calculates and plots statistics for each phase separately. The arguments takes a numeric vector (phase number(s)) or a character vector (phase name(s)). When more than one phase is defines, statistics are based on the combined values of these phases. Various methods for an extrapolated *trendA* line exist: "trendA" is based on an OLS regression, "trendA theil-sen" on a nonparametric regression, and "trendA bisplit" / "trendA trisplit" are two median based approaches. Some of the functions defined in `stats` have additional arguments. The `mean()` function has a `trim` argument (e.g. `trim = 0.1`). `quantile()` has a `proportion` argument

(e.g. `prob = 0.75` for calculating the 75% quantile). `moving mean` and `moving median` have a `lag` argument (e.g. `lag = 2`). The local-regression curve function "`lowess`" (or "`loreg`") has a `proportion` argument (e.g. `f = 0.5`; see `lowess()`) and the local-regression curve function "`loess`" has a `span` argument (e.g. `span = 0.75`; see `loess()`).

Value

An object of class `scplot` (see `scplot()`) with changed element `statlines`.

add_text	<i>Add test to an scplot</i>
----------	------------------------------

Description

Add test to an `scplot`

Usage

```
add_text(
  object,
  label,
  case = 1,
  x,
  y,
  color = "black",
  size = 1,
  angle = 0,
  hjust = 0.5,
  vjust = 0.5,
  face = 1
)
```

Arguments

<code>object</code>	An <code>scplot</code> object (class <code>scplot</code>) returned from the <code>scplot()</code> function.
<code>label</code>	A Character vector with text labels.
<code>case</code>	Numerical vector with the case number or character string. <code>case = "all"</code> for all cases.
<code>x</code>	x position
<code>y</code>	y position
<code>color</code>	A character string or a number defining the color of an element.
<code>size</code>	Text size relative to the base text size.
<code>angle</code>	Angle (in <code>[0,360]</code>)
<code>hjust</code>	Horizontal justification (in <code>[0,1]</code>)
<code>vjust</code>	Vertical justification (in <code>[0,1]</code>)
<code>face</code>	Font face (" <code>plain</code> ", " <code>italic</code> ", " <code>bold</code> ", " <code>bold.italic</code> ")

Value

An object of class `scplot` (see `scplot()`) with a changed `texts` element.

<code>add_title</code>	<i>Add title and caption to an <code>scplot</code></i>
------------------------	--

Description

Add title and caption to an `scplot`

Usage

```
add_title(object, label, ...)
```

```
add_caption(object, label, header = "Note:\n", ...)
```

Arguments

`object` An `scplot` object (class `scplot`) returned from the `scplot()` function.

`label` A Character vector with text labels.

`...` List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "1". See `element_text()`.

`header` String with header above footnote/ caption

Value

An object of class `scplot` (see `scplot()`) with changed title and caption elements.

<code>as_ggplot</code>	<i>Creates a <code>ggplot2</code> object from an <code>scplot()</code> object</i>
------------------------	---

Description

Creates a `ggplot2` object from an `scplot()` object

Usage

```
as_ggplot(scplot)
```

Arguments

`scplot` An `scplot` object

Details

as_ggplot() is used when you want to return a ggplot2 object for further use with external ggplot functions.

Value

A ggplot2 plot object.

element_point	<i>Point element</i>
---------------	----------------------

Description

In conjunction with ggplot an object to represent point attributes.

Usage

```
element_point(colour = NULL, size = NULL, shape = NULL, color = NULL)
```

Arguments

size	Relative size.
shape	Point shape.
color, colour	Point colour.

Value

An object of class c("element_point", "element").

new_theme	<i>Create a new scplot theme</i>
-----------	----------------------------------

Description

Create a new scplot theme

Usage

```
new_theme()
extract_theme(object)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
--------	--

Value

An splot-theme object

An object of class `splot-theme` which can be used with the `set_theme()` function.

Examples

```
data(exampleABC, package = "scan")
my_theme <- new_theme() |>
  set_panel(color = "red") |>
  set_base_text(size = 12, color = "blue") |>
  set_dataline(color = "darkred", linewidth = 2)
p1 <- splot(exampleABC) |> set_theme(my_theme)
```

splot.scdf

Plot single-case data

Description

This function provides a plot of a single-case or multiple single-cases.

Usage

```
## S3 method for class 'scdf'
splot(object, ...)
```

Arguments

`object` A single-case data-frame object (scdf).
`...` further arguments.

Value

An object of class `splot` containing the single-case data (element `scdf`), and information about the plot style (element `theme`).

Author(s)

Juergen Wilbert

splot.sc_hplm	<i>This function generates a forest plot for the random effects of a mixed hplm model</i>
---------------	---

Description

This function generates a forest plot for the random effects of a mixed hplm model

Usage

```
## S3 method for class 'sc_hplm'
splot(object, effect = "intercept", mark = "fixed", ci = 0.95, ...)
```

Arguments

object	The return from the hplm() function.
effect	The specific effect to be plotted (default is the intercept).
mark	Set a reference line.
ci	Value between 0 and 1 for calculating the confidence interval.
...	Further arguments.

Value

A forest plot displaying Tau-U effects.

Examples

```
model <- scan::hplm(scan::Leidig2018, random.slopes = TRUE)
splot(model, effect = "level")
```

splot.sc_rand	<i>Plot Randomization Effects</i>
---------------	-----------------------------------

Description

This function generates a forest plot of Tau-U effects.

Usage

```
## S3 method for class 'sc_rand'
splot(object, type = "hist", add_density_curve = TRUE, ...)
```

Arguments

object The return from the tau_u() function.
 type Either "hist" or "xy".
 add_density_curve If TRUE, adds a density curve to the histogram.
 ... Further arguments.

Value

A forest plot displaying Tau-U effects.

Examples

```
## Not run:
res <- scan::rand_test(scan::exampleAB$Anja, limit = 1)
scplot(res, type = "hist")

scplot(res, type = "xy")

## End(Not run)
```

scplot.sc_tauu

Plot Tau-U Effects

Description

This function generates a forest plot of Tau-U effects.

Usage

```
## S3 method for class 'sc_tauu'
scplot(object, effect = 1, ...)
```

Arguments

object The return from the tau_u() function.
 effect The specific effect to be plotted (default is "A vs. B - Trend A").
 ... Further arguments.

Value

A forest plot displaying Tau-U effects.

Examples

```
res <- scan::tau_u(scan::Leidig2018)
scplot(res, effect = 3)
```

set_background	<i>Set plot and panel background of an scplot</i>
----------------	---

Description

Set plot and panel background of an scplot

Usage

```
set_background(object, ...)
```

```
set_panel(object, ...)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
...	List with rectangle parameters ("fill", "colour", "linewidth", "linetype"). See element_rect() .

Value

An object of class scplot (see [scplot\(\)](#)).

Examples

```
data(exampleAB, package = "scan")
p1 <- scplot(exampleAB) |>
  set_background(fill = "lightblue", colour = "darkblue", linewidth = 1.5) |>
  set_panel(fill = "deepskyblue", color = "darkblue", linewidth = 0.3)
```

set_base_text	<i>Set base text parameters of an scplot</i>
---------------	--

Description

Set base text parameters of an scplot

Usage

```
set_base_text(object, ...)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
...	List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "l..."). See element_text() .

Value

An object of class `scplot` (see `scplot()`).

<code>set_casenames</code>	<i>Set casenames of an scplot</i>
----------------------------	-----------------------------------

Description

Set casenames of an `scplot`

Usage

```
set_casenames(object, labels = NULL, position = NULL, background = list(), ...)
```

Arguments

<code>object</code>	An <code>scplot</code> object (class <code>scplot</code>) returned from the <code>scplot()</code> function.
<code>labels</code>	A Character vector with text labels.
<code>position</code>	Either <code>"topleft"</code> , <code>"bottomleft"</code> , <code>"topright"</code> , <code>"bottomright"</code> , <code>"strip-right"</code> , <code>"strip-top"</code> , or a numerical vector of length 2 with the x and y position (e.g. <code>c(19, 20)</code>).
<code>background</code>	A list with background styling arguments (<code>fill</code> , <code>color</code> , <code>size</code> , <code>linetype</code>).
<code>...</code>	List with text parameters (<code>"family"</code> , <code>"face"</code> , <code>"colour"</code> , <code>"size"</code> , <code>"hjust"</code> , <code>"vjust"</code> , <code>"angle"</code> , "1". See <code>element_text()</code> .

Value

An object of class `scplot` (see `scplot()`) with a changed `casenames` element.

<code>set_dataline</code>	<i>Set data lines of an scplot</i>
---------------------------	------------------------------------

Description

Either set aesthetics of the default data line or add another data line.

Usage

```
set_dataline(  
  object,  
  variable = NULL,  
  line,  
  point,  
  type = "continuous",  
  label = NULL,  
  ...  
)  
  
add_dataline(...)
```

Arguments

object	An splot object (class <code>splot</code>) returned from the <code>splot()</code> function.
variable	String. The name of a new variable for adding a new line. If left empty, the aesthetics of the default data line are changed.
line	List with line parameters ("colour", "linewidth", "linetype", "lineend", "arrow"). See element_line() .
point	A list with point parameters ("colour", "size", "shape"). See element_point() .
type	Either "continuous" or "discrete"
label	A character string which is used to set the label in a legend.
...	As a shortcut, arguments passed here are bundled as line arguments (see element_line()).

Value

An object of class `splot` (see [splot\(\)](#)) with a changed `datalines` element.

See Also

[element_line\(\)](#), [element_point\(\)](#)

Examples

```
data(exampleAB_add, package = "scan")  
splot(exampleAB_add) |>  
  set_dataline("depression", color = "darkblue")
```

set_phasenames	<i>Set phasenames of an scplot</i>
----------------	------------------------------------

Description

Set phasenames of an scplot

Usage

```
set_phasenames(object, labels = NULL, position = NULL, ...)
```

Arguments

object	An scplot object (class scplot) returned from the <code>scplot()</code> function.
labels	A Character vector with text labels.
position	Character string either 'left', 'center', or 'none'.
...	List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "1", ...). See element_text() .

Value

An object of class scplot (see [scplot\(\)](#)) with a changed phasenames element.

set_separator	<i>Set separator line in an scplot</i>
---------------	--

Description

Set separator line in an scplot

Usage

```
set_separator(object, ...)
```

Arguments

object	An scplot object (class scplot) returned from the <code>scplot()</code> function.
...	List with line parameters ("colour", "linewidth", "linetype").

Value

An object of class scplot (see [scplot\(\)](#)).

set_theme	<i>Add a theme of to an scplot</i>
-----------	------------------------------------

Description

Possible themes are: 'basic', 'grid', 'default', 'small', 'tiny', 'big', 'minimal', 'dark', 'sienna', 'phas

Usage

```
set_theme(object, theme, ...)
```

```
add_theme(...)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
theme	A character string with a predefined graphical theme or a theme object created with new_theme() .
...	Further character strings or scplot-theme objects that are "added" on top.

Value

An object of class scplot (see [scplot\(\)](#)) with a changed theme element.

set_theme_element	<i>Set a theme element</i>
-------------------	----------------------------

Description

Set a theme element

Usage

```
set_theme_element(object, ...)
```

Arguments

object	An scplot object (class scplot) returned from the scplot() function.
...	various style parameter

Details

Usually, you don't need this function. Possible theme elements are: "text", "plot.background", "panel.background", "panel.spacing.y", "dataline", "datapoint", "statline", "axis.expand.x", "axis.expand.y", "axis.line.x", "axis.line.y", "axis.ticks.length", "axis.ticks", "axis.title.y", "axis.title.x", "axis.text.x", "axis.text.y", "plot.title", "plot.caption", "plot.margin", "casenames", "casenames.strip", "casenames.background", "casenames.position", "phasenames", "phasenames.position.x", "separators", "separators.extent", "label.text", "label.background", "label.padding", "grid", "legend.position", "legend.background", "legend.text", "legend.title", "legend.margin".

The elements are of the following classes:

- text = c("element_text", "element"),
- plot.background = c("element_rect", "element"),
- panel.spacing.y = c("simpleUnit", "unit", "unit_v2"),
- dataline = "list",
- datapoint = "list",
- statline = c("element_line", "element"),
- axis.expand.x = "numeric",
- axis.expand.y = "numeric",
- axis.line.x = c("element_line", "element"),
- axis.line.y = c("element_line", "element"),
- axis.ticks.length = c("simpleUnit", "unit", "unit_v2"),
- axis.ticks = c("element_line", "element"),
- axis.title.y = c("element_text", "element"),
- axis.title.x = c("element_text", "element"),
- axis.text.x = c("element_text", "element"),
- axis.text.y = c("element_text", "element"),
- plot.title = c("element_text", "element"),
- plot.caption = c("element_text", "element"),
- plot.margin = c("margin", "simpleUnit", "unit", "unit_v2"),
- casenames = c("element_text", "element"),
- casenames.strip = c("element_rect", "element"),
- casenames.background = c("element_rect", "element"),
- casenames.position = "character",
- phasenames = c("element_text", "element"),
- phasenames.position.x = "character",
- separators = c("element_line", "element"),
- separators.extent = "character",
- label.text = c("element_text", "element"),
- label.background = c("element_rect", "element"),

- label.padding = "numeric", grid = c("element_line", "element"),
- legend.position = "character",
- legend.background = c("element_rect", "element"),
- legend.text = c("element_text", "element"),
- legend.title = c("element_text", "element"),
- legend.margin = c("margin", "simpleUnit", "unit", "unit_v2")

Value

An object of class `scplot` (see `scplot()`) with a changed theme element.

Examples

```
data(exampleABC, package = "scan")
p1 <- scplot(exampleABC) |>
  set_theme_element(
    axis.ticks.length = unit(0, "points"),
    axis.line.y = element_line(color = "darkred", linewidth = 2),
    panel.background = element_rect(color = "darkblue", linewidth = 1),
    panel.spacing.y = unit(0, "points"),
    phasenames = element_text(color = "#00000000")
  )
```

set_xaxis

Set axis parameters of an scplot

Description

Set axis parameters of an `scplot`

Usage

```
set_xaxis(
  object,
  limits = NULL,
  increment = NULL,
  increment_from = NULL,
  line = NULL,
  expand = NULL,
  ...
)
```

```
set_yaxis(
  object,
  limits = NULL,
  increment = NULL,
  increment_from = NULL,
```

```

    line = NULL,
    expand = NULL,
    ...
)

```

Arguments

object	An splot object (class splot) returned from the splot() function.
limits	Lower and upper limits of the axis (e.g., limits = c(0, 20) sets the axis to a scale from 0 to 20). With multiple single-cases you can use limits = c(0, NA) to scale the axis from 0 to the maximum of each case. limits is not set by default, which makes splot set a proper scale based on the given data.
increment	An integer. Increment of the x-axis. 1 :each mt value will be printed, 2 : every other value, 3 : every third values etc.
increment_from	Number from which increment starts to count. Usually set to 0 if you want marks like 1,5,10,15,...
line	List with line parameters ("colour", "linewidth", "linetype", "lineend", "arrow"). See element_line() .
expand	Vector with two values.
...	Further styling arguments: color, size, face, family, hjust, vjust, lineheight, angle, linetype, lineend, arrow, fill, margin.

Value

An object of class splot (see [splot\(\)](#)) with changed xaxis and yaxis elements.

set_xlabel	<i>Set label for axis</i>
------------	---------------------------

Description

Set label for axis

Usage

```
set_xlabel(object, label = NULL, ...)
```

```
set_ylabel(object, label = NULL, ...)
```

Arguments

object	An splot object (class splot) returned from the splot() function.
label	A Character vector with text labels.
...	Further styling arguments: color, size, face, family, hjust, vjust, lineheight, angle, linetype, lineend, arrow, fill, margin.

Value

An object of class `scplot` (see [scplot\(\)](#)) with a changed `xlabel` or `ylabel` element.

Index

- * **package**
 - scplot-package, 2

- add_arrow, 3
- add_caption (add_title), 12
- add_dataline (set_dataline), 18
- add_grid, 4
- add_labels, 5
- add_legend, 6
- add_line, 7
- add_marks, 8
- add_ridge, 9
- add_statline, 10
- add_text, 11
- add_theme (set_theme), 21
- add_title, 12
- as_ggplot, 12

- element_line(), 4, 19, 24
- element_point, 13
- element_point(), 19
- element_rect(), 17
- element_text(), 5, 6, 12, 17, 18, 20
- extract_theme (new_theme), 13

- loess(), 11
- lowess(), 11

- mean(), 10

- new_theme, 13
- new_theme(), 21

- par(), 9

- quantile(), 10

- scplot (scplot.scdf), 14
- scplot(), 3–6, 8, 9, 11, 12, 17–21, 23–25
- scplot-package, 2
- scplot.sc_hplm, 15

- scplot.sc_rand, 15
- scplot.sc_tauu, 16
- scplot.scdf, 14
- set_background, 17
- set_base_text, 17
- set_casenames, 18
- set_dataline, 18
- set_panel (set_background), 17
- set_phasenames, 20
- set_separator, 20
- set_theme, 21
- set_theme(), 14
- set_theme_element, 21
- set_xaxis, 23
- set_xlabel, 24
- set_yaxis (set_xaxis), 23
- set_ylabel (set_xlabel), 24