

# Package ‘scplot’

June 3, 2023

**Type** Package

**Title** Plot Function for Single-Case Data Frames

**Version** 0.3.3

**Date** 2023-6-03

**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''  
Jürgen Wilbert (2023) <<https://jazznbass.github.io/scan-Book/>>.

**Depends** R (>= 3.5.0)

**Imports** ggplot2, scan (>= 0.57.0), stats, utils, mblm

**Suggests** rmarkdown, knitr

**License** GPL (>= 3)

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**NeedsCompilation** no

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

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

**Repository** CRAN

**Date/Publication** 2023-06-03 09:00:02 UTC

## R topics documented:

scplot-package . . . . .	2
add_arrow . . . . .	2
add_grid . . . . .	4
add_labels . . . . .	4
add_legend . . . . .	5
add_marks . . . . .	6

add_ridge . . . . .	7
add_statline . . . . .	8
add_text . . . . .	9
add_title . . . . .	10
as_ggplot . . . . .	11
element_point . . . . .	11
new_theme . . . . .	12
splot . . . . .	12
set_background . . . . .	13
set_base_text . . . . .	14
set_casenames . . . . .	14
set_dataline . . . . .	15
set_phasenames . . . . .	16
set_separator . . . . .	16
set_theme . . . . .	17
set_theme_element . . . . .	17
set_xaxis . . . . .	19
set_xlabel . . . . .	20

<b>Index</b>	<b>22</b>
--------------	-----------

---

splot-package	<i>Single-Case Data Plots</i>
---------------	-------------------------------

---

### 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 arrows.

**Examples**

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

---

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

---

**Description**

Add grid to an splot

**Usage**

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

**Arguments**

object	An splot object (class <code>splot</code> ) returned from the <code>splot()</code> function.
...	Line arguments (see <a href="#">element_line()</a> )

**Value**

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

**See Also**

[element\\_line\(\)](#)

**Examples**

```
data(exampleAB, package = "scan")
p1 <- splot(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 scplot object (class scplot) returned from the scplot() 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", "1". See <a href="#">element_text()</a> .
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 scplot (see [scplot\(\)](#)) 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 splot object (class splot) returned from the <code>splot()</code> function.
labels	A character vector. Replaces the automatically build labels. When a vector element is NA, the corresponding element will not be replaced (e.g. <code>c(NA, "Variable 2", NA, "Variable 4")</code> ).
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. <code>case = "all"</code> 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.
title	A list with text style parameters for the title.
text	List with text parameters ("family", "face", "colour", "size", "hjust", "vjust", "angle", "1. See <code>element_text()</code> .
background	A list with background styling arguments (fill, color, size, linetype).

### Value

An object of class `splot` (see `splot()`) with changed element legend.

---

add_marks	<i>Add marks to an splot</i>
-----------	------------------------------

---

### Description

Add marks to an splot

### 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 <code>scplot()</code> function.
case	Numerical vector with the case number or character string. <code>case = "all"</code> for all cases.
positions	Either a vector indicating the points to be highlighted or a character string with a logical expression (e.g. <code>values &lt; mean(values)</code> )
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 <code>pch</code> graphical parameter on <code>par</code> help page <a href="#">par()</a> .
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	<i>Add a ridge to an scplot</i>
-----------	---------------------------------

---

### Description

Add a ridge to an `scplot`

### Usage

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

**Arguments**

object	An splot object (class <code>splot</code> ) returned from the <code>splot()</code> 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 `splot` (see [splot\(\)](#)) with changed element ridges.

---

add_statline	<i>Add a statline to an splot</i>
--------------	-----------------------------------

---

**Description**

Add a statline to an splot

**Usage**

```
add_statline(
  object,
  stat = c("mean", "median", "min", "max", "quantile", "sd", "mad", "trend", "trendA",
    "trendA bisplit", "trendA trisplit", "trendA theil-sen", "movingMean", "moving mean",
    "moving median", "movingMedian", "loreg", "lowess", "loess"),
  phase = NULL,
  color = NULL,
  linewidth = NULL,
  linetype = NULL,
  variable = 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.
...	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 `splot` (see `splot()`) with changed element `statlines`.

---

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

---

## Description

Add test to an `splot`

## Usage

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

## Arguments

object	An <code>splot</code> object (class <code>splot</code> ) returned from the <code>splot()</code> function.
label	A Character vector with text labels.
case	Numerical vector with the case number or character string. <code>case = "all"</code> for all cases.

x	x position
y	y position
color	A character string or a number defining the color of an element.
size	Text size relative to the base text size.
angle	Angle (in [0,360])
hjust	Horizontal justification (in [0,1])
vjust	Vertical justification (in [0,1])
face	Font face ("plain", "italic", "bold", "bold.italic")

**Value**

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

---

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

---

**Description**

Add title and caption to an `scplot`

**Usage**

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

```
add_caption(object, label, ...)
```

**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()`.

**Value**

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

---

as_ggplot	<i>Creates a ggplot2 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 splot theme</i>
-----------	---------------------------------

---

**Description**

Create a new splot theme

**Usage**

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

**Arguments**

object            An splot object (class splot) returned from the splot() 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	<i>Plot single-case data</i>
-------	------------------------------

---

**Description**

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

**Usage**

```
splot(scdf)
```

**Arguments**

scdf            A single-case data-frame object (scdf).

**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

---

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

---

**Description**

Set plot and panel background of an `splot`

**Usage**

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

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

**Arguments**

<code>object</code>	An <code>splot</code> object (class <code>splot</code> ) returned from the <code>splot()</code> function.
<code>...</code>	List with rectangle parameters ( <code>"fill"</code> , <code>"colour"</code> , <code>"linewidth"</code> , <code>"linetype"</code> ). See <a href="#">element_rect()</a> .

**Value**

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

**Examples**

```
data(exampleAB, package = "scan")
p1 <- splot(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", "1". See <a href="#">element_text()</a> .

**Value**

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

---

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

---

**Description**

Set casenames of an scplot

**Usage**

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

**Arguments**

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

**Value**

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

---

set_dataline	<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", ...)
```

```
add_dataline(...)
```

## Arguments

object	An scplot object (class scplot) returned from the <code>scplot()</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 <a href="#">element_line()</a> .
point	A list with point parameters ("colour", "size", "shape"). See <a href="#">element_point()</a> .
type	Either "continuous" or "discrete"
...	As a shortcut, arguments passed here are bundled as line arguments (see <a href="#">element_line()</a> ).

## Value

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

## See Also

[element\\_line\(\)](#), [element\\_point\(\)](#)

## Examples

```
data(exampleAB_add, package = "scan")
p1 <- scplot(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 scplot() 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 <a href="#">element_text()</a> .

**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 scplot() 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 <a href="#">new_theme()</a> .
...	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 <a href="#">element_line()</a> .
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, 2
- add\_caption (add\_title), 10
- add\_dataline (set\_dataline), 15
- add\_grid, 4
- add\_labels, 4
- add\_legend, 5
- add\_marks, 6
- add\_ridge, 7
- add\_statline, 8
- add\_text, 9
- add\_theme (set\_theme), 17
- add\_title, 10
- as\_ggplot, 11
  
- element\_line(), 4, 15, 20
- element\_point, 11
- element\_point(), 15
- element\_rect(), 13
- element\_text(), 5, 6, 10, 14, 16
- extract\_theme (new\_theme), 12
  
- loess(), 9
- lowess(), 9
  
- mean(), 9
  
- new\_theme, 12
- new\_theme(), 17
  
- outlier(), 7
  
- par(), 7
  
- quantile(), 9
  
- scplot, 12
- scplot(), 3–11, 13–17, 19–21
- scplot-package, 2
  
- set\_background, 13
- set\_base\_text, 14
- set\_casenames, 14
- set\_dataline, 15
- set\_panel (set\_background), 13
- set\_phasenames, 16
- set\_separator, 16
- set\_theme, 17
- set\_theme(), 12
- set\_theme\_element, 17
- set\_xaxis, 19
- set\_xlabel, 20
- set\_yaxis (set\_xaxis), 19
- set\_ylabel (set\_xlabel), 20