

# Package: fxl (via r-universe)

October 31, 2024

**Type** Package

**Title** 'fxl' Single Case Design Charting Package

**Version** 1.7.1

**Suggests** covr, knitr, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Description** The 'fxl' Charting package is used to prepare and design single case design figures that are typically prepared in spreadsheet software. With 'fxl', there is no need to leave the R environment to prepare these works and many of the more unique conventions in single case experimental designs can be performed without the need for physically constructing features of plots (e.g., drawing annotations across plots). Support is provided for various different plotting arrangements (e.g., multiple baseline), annotations (e.g., brackets, arrows), and output formats (e.g., svg, rasters).

**License** GPL (>=3)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 4.1), rlang

**RoxygenNote** 7.2.3

**Config/testthat/edition** 3

**Repository** <https://miyamot0.r-universe.dev>

**RemoteUrl** <https://github.com/miyamot0/fxl>

**RemoteRef** HEAD

**RemoteSha** 6190fa6f2f1c7b873cc85c34e5b1f04492d2a5b0

## Contents

assert_input_type . . . . .	3
Challenge1Data . . . . .	3
Challenge2Data . . . . .	4

Challenge4Data . . . . .	5
cnvrt_coords . . . . .	5
draw_arrows . . . . .	6
draw_bar_support . . . . .	7
draw_brackets . . . . .	7
draw_cumsum_lines . . . . .	8
draw_cumsum_points . . . . .	8
draw_guide_line . . . . .	9
draw_images . . . . .	10
draw_label_facet . . . . .	10
draw_label_phase . . . . .	11
draw_legend . . . . .	11
draw_lines . . . . .	12
draw_points . . . . .	12
draw_rect . . . . .	13
draw_scr_criterion . . . . .	14
draw_scr_plines . . . . .	14
GelinoEtAl2022 . . . . .	15
Gilroyetal2015 . . . . .	16
Gilroyetal2019 . . . . .	16
Gilroyetal2019Tx . . . . .	17
Gilroyetal2021 . . . . .	18
isValidAestheticMapping . . . . .	18
isValidAXSCharacter . . . . .	19
isValidCharacterVector . . . . .	19
isValidDataFrame . . . . .	20
isValidLogicalVector . . . . .	20
isValidNumericVector . . . . .	21
KoffarnusEtAl2011 . . . . .	21
LozyEtAl2020 . . . . .	22
print.fx1 . . . . .	22
print.fx1semilog . . . . .	23
scr_anno_arrows . . . . .	23
scr_anno_brackets . . . . .	25
scr_anno_guide_line . . . . .	26
scr_anno_rect . . . . .	27
scr_bar_support . . . . .	27
scr_criterion_lines . . . . .	28
scr_cumsum_lines . . . . .	29
scr_cumsum_points . . . . .	30
scr_images . . . . .	31
scr_label_facet . . . . .	31
scr_label_phase . . . . .	32
scr_legend . . . . .	33
scr_lines . . . . .	34
scr_plines . . . . .	35
scr_plines_mbd . . . . .	36
scr_plot . . . . .	36

<i>assert_input_type</i>	3
scr_points . . . . .	38
scr_save . . . . .	39
scr_title . . . . .	40
scr_xlabel . . . . .	40
scr_xoverride . . . . .	41
scr_ylabel . . . . .	42
scr_yoverride . . . . .	43
SimulatedAcademicFluency . . . . .	44
var_map . . . . .	44

**Index** **46**

---

<code>assert_input_type</code>	<i>assert_input_type</i>
--------------------------------	--------------------------

---

**Description**

`assert_input_type`

**Usage**

`assert_input_type(object, types = character(0), tag = "")`

**Arguments**

<code>object</code>	some type of object
<code>types</code>	list of object types acceptable
<code>tag</code>	var to reference in error message

**Value**

no return value, run for side effects

---

<code>Challenge1Data</code>	<i>Twitter chart challenge data 1</i>
-----------------------------	---------------------------------------

---

**Description**

Twitter chart challenge data 1

**Usage**

`Challenge1Data`

**Format**

A data frame with 226 rows and 11 variables:

**StudyID** Extracted study ID

**FigureNum** Extracted study figure number

**PanelNum** Extracted study panel number

**CaseName** Extracted study case name

**CaseNum** Extracted study number

**X** Session

**OutcomeName** Extracted study outcome name

**Direction** Direction of trend

**Y** Outcome measure

**CondName** Extracted study condition

**CondNum** Extracted study number

---

Challenge2Data

*Twitter chart challenge data 2*

---

**Description**

Twitter chart challenge data 2

**Usage**

Challenge2Data

**Format**

A data frame with 113 rows and 5 variables:

**Participant** Participant name

**Session** Session number

**Condition** Condition name

**IWS** Incorrect word sequences

**CWS** Correct word sounds

---

Challenge4Data	<i>Twitter chart challenge data 4</i>
----------------	---------------------------------------

---

**Description**

Twitter chart challenge data 4

**Usage**

Challenge4Data

**Format**

A data frame with 189 rows and 11 variables:

**Session** Session number

**Participant** Participant name

**Phase** Phase name

**Number.Writing.Fluency** Fluency of number writing

**Dot.Number** Fluency of dot number skills

**Dot.Number.Total** Fluency of dot number skills on all sets

**Number.Total** Number writing fluency on all sets

**Number.Writing.Fluency\_Accuracy** Number writing accuracy

**Dot.Number\_Accuracy** Dot number accuracy

**Dot.Number.Total\_Accuracy** Dot number accuracy on all sets

**Number.Total\_Accuracy** Number writing accuracy on all sets

---

cnvrt_coords	<i>cnvrt_coords</i>
--------------	---------------------

---

**Description**

Pulled from the TeachingDemos package (GPLv2+ Licensed)

**Usage**

```
cnvrt_coords(x, y = NULL, input = c("usr", "plt", "fig", "dev", "tdev"))
```

**Arguments**

x                    abscissa

y                    ordinate

input                device

**Details**

Slightly hacked/trimmed

**Value**

Transformation of coordinates from local plot to figure space for phase changes

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>, Greg Snow <538280@gmail.com>

---

draw\_arrows

*draw\_arrows*

---

**Description**

drawing function

**Usage**

```
draw_arrows(core_frame, current_layer, facet_name)
```

**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

`draw_bar_support`      *draw\_bar\_support*

---

**Description**

Draw bars, but on a secondary axis

**Usage**

`draw_bar_support(core_frame, current_layer, facet_name, max_y)`

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet
<code>max_y</code>	top of y axis to match

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

`draw_brackets`      *draw\_brackets*

---

**Description**

drawing function

**Usage**

`draw_brackets(core_frame, current_layer, facet_name)`

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

`draw_cumsum_lines`      *draw\_cumsum\_lines*

---

**Description**

`draw_cumsum_lines`

**Usage**

```
draw_cumsum_lines(core_frame, current_layer, facet_name)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

`draw_cumsum_points`      *draw\_cumsum\_points*

---

**Description**

`draw_cumsum_points`

**Usage**

```
draw_cumsum_points(core_frame, current_layer, facet_name)
```



**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

draw_guide_line	<i>draw_guide_line</i>
-----------------	------------------------

---

**Description**

draw\_guide\_line

**Usage**

draw\_guide\_line(core\_frame, current\_layer, facet\_name)

**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

<code>draw_images</code>	<i>draw_images</i>
--------------------------	--------------------

---

**Description**

drawing function

**Usage**

```
draw_images(core_frame, current_layer, facet_name, zero_axis = FALSE)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet
<code>zero_axis</code>	filter out all but zeros

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

<code>draw_label_facet</code>	<i>draw_label_facet</i>
-------------------------------	-------------------------

---

**Description**

drawing function

**Usage**

```
draw_label_facet(core_frame, current_layer, facet_name)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

*draw\_label\_phase*      *draw\_label\_phase*

---

**Description**

drawing function

**Usage**

`draw_label_phase(core_frame, current_layer, facet_name)`

**Arguments**

`core_frame`      fxl object  
`current_layer`    layer to be drawn  
`facet_name`      name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

*draw\_legend*      *draw\_legend*

---

**Description**

drawing function

**Usage**

`draw_legend(core_frame)`

**Arguments**

`core_frame`      fxl object

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

draw_lines	<i>draw_lines</i>
------------	-------------------

---

**Description**

drawing function

**Usage**

```
draw_lines(core_frame, current_layer, facet_name)
```

**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

draw_points	<i>draw_points</i>
-------------	--------------------

---

**Description**

drawing function

**Usage**

```
draw_points(core_frame, current_layer, facet_name, zero_axis = FALSE)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet
<code>zero_axis</code>	filter out all but zeros

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

<code>draw_rect</code>	<i>draw_rect</i>
------------------------	------------------

---

**Description**

drawing function

**Usage**

```
draw_rect(core_frame, current_layer, facet_name, zero_axis = FALSE)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>current_layer</code>	layer to be drawn
<code>facet_name</code>	name of facet
<code>zero_axis</code>	filter out all but zeros

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

draw\_scr\_criterion     *draw\_scr\_criterion*

---

**Description**

drawing function

**Usage**

```
draw_scr_criterion(core_frame, current_layer, facet_name)
```

**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

draw\_scr\_plines     *draw\_scr\_plines*

---

**Description**

drawing function

**Usage**

```
draw_scr_plines(core_frame, current_layer, facet_name)
```

**Arguments**

core_frame	fxl object
current_layer	layer to be drawn
facet_name	name of facet

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

GelinoEtAl2022

*Plotting data from Koffarnus et al. (2011)*

---

**Description**

Treatment data from Koffarnus et al. (2011)

**Usage**

GelinoEtAl2022

**Format**

A data frame with 9 rows and 1 variables:

**Condition** Baseline vs. policy condition

**Time** Time of study

**SC** Slope change dummy code

**yhat** Predicted value from model

**Count1** Count for site 1

**Count2** Count for site 2

**Count3** Count for site 3

**Count4** Count for site 4

**Facet** Facet/subplot number

**Source**

<doi:https://doi.org/10.1002/jaba.967>

---

Gilroyetal2015

*Plotting data from Gilroy et al. (2015)*

---

### Description

This is data either extracted or included (with permission) to assist with illustrating and replicating the functionality of the package. Specifically, this data is used to illustrate conventions regarding a multiple probe design and how that may be constructed in this package.

### Usage

Gilroyetal2015

### Format

A data frame with 40 rows and 6 variables:

**Participant** Participant name

**Session** Session number

**Condition** Condition name

**Responding** Responding rates

**PhaseNum** Phase number

**LineOff** Offset for phase line

### Source

<doi:https://doi.org/10.1016/j.rasd.2015.04.004>

---

Gilroyetal2019

*Plotting data from Gilroy et al. (2019) - FA*

---

### Description

Data from this study (provided with permission) is used to illustrate how findings from a functional analysis can be drawn using the package. Generally, its the same type of functionality required in an alternating treatment design approach.

### Usage

Gilroyetal2019



**Format**

A data frame with 15 rows and 9 variables:

**Session** Session number

**SIB** Rates of self-injury

**AGG** Rates of aggression

**DIS** Rates of disruptive behavior

**Prompt** Rates of prompting

**Comply** Rates of compliance

**SR** Duration of reinforcement

**CTB** Rates of combined target behavior

**Condition** Functional analysis condition

**Source**

<doi:https://doi.org/10.1080/17518423.2019.1646342>

---

Gilroyetal2019Tx

*Plotting data from Gilroy et al. (2019) - Treatment*

---

**Description**

This is data either extracted or included (with permission) to assist with illustrating and replicating the functionality of the package. Specifically, this data is used to illustrate conventions regarding combined design elements that include multiple baseline and reversal design elements.

**Usage**

Gilroyetal2019Tx

**Format**

A data frame with 86 rows and 8 variables:

**Participant** Participant name

**Session** Session number

**Condition** Functional analysis condition

**CTB** Rates of combined target behavior

**FCR** Rates for communication response for function 1

**FCR2** Rates for communication response for function 2

**PhaseNum** Sequenced phase number

**LineOff** Offset of phase line

**Source**

<doi:https://doi.org/10.1080/17518423.2019.1646342>

---

 Gilroyetal2021

*Plotting data from Gilroy et al. (2015) - Treatment*


---

**Description**

Treatment data from Gilroy et al. (2021)

**Usage**

Gilroyetal2021

**Format**

A data frame with 69 rows and 7 variables:

**Participant** Participant name

**Session** Session number

**Condition** Functional analysis condition

**Responding** Rates of responding

**Reinforcers** Reinforcer deliveries

**PhaseNum** Sequenced phase number

**LineOff** Offset of phase line

**Source**

<doi:https://doi.org/10.1002/jaba.826>

---

 isValidAestheticMapping

*isValidAestheticMapping*


---

**Description**

isValidAestheticMapping

**Usage**

isValidAestheticMapping(object = NULL, name = NULL)

**Arguments**

object            dataframe (hopefully)

name             name for object

**Value**

no return value, run for side effects

---

`isValidAXSCharacter`     *isValidAXSCharacter*

---

**Description**

`isValidAXSCharacter`

**Usage**

`isValidAXSCharacter(object = NULL, name = NULL)`

**Arguments**

<code>object</code>	some type of object
<code>name</code>	parameter name

**Value**

no return value, run for side effects

---

`isValidCharacterVector`  
*isValidCharacterVector*

---

**Description**

`isValidCharacterVector`

**Usage**

`isValidCharacterVector(object = NULL, length = -1, name = NULL)`

**Arguments**

<code>object</code>	some type of object
<code>length</code>	expected length
<code>name</code>	parameter name

**Value**

no return value, run for side effects

---

`isValidDataFrame`      *isValidDataFrame*

---

**Description**

`isValidDataFrame`

**Usage**

```
isValidDataFrame(object = NULL, name = NULL)
```

**Arguments**

<code>object</code>	dataframe (hopefully)
<code>name</code>	name for object

**Value**

no return value, run for side effects

---

`isValidLogicalVector`      *isValidLogicalVector*

---

**Description**

`isValidLogicalVector`

**Usage**

```
isValidLogicalVector(object = NULL, length = -1, name = NULL)
```

**Arguments**

<code>object</code>	some type of object
<code>length</code>	expected length
<code>name</code>	parameter name

**Value**

no return value, run for side effects

---

isValidNumericVector    *isValidNumericVector*

---

**Description**

isValidNumericVector

**Usage**

```
isValidNumericVector(object = NULL, length = -1, name = NULL)
```

**Arguments**

object	some type of object
length	expected length
name	parameter name

**Value**

no return value, run for side effects

---

KoffarnusEtA12011    *Plotting data from Koffarnus et al. (2011)*

---

**Description**

Treatment data from Koffarnus et al. (2011)

**Usage**

```
KoffarnusEtA12011
```

**Format**

A data frame with 14979 rows and 3 variables:

**X** Session/day number

**ID** Participant ID on the Y axis

**Code** Status for treatment

**Source**

<doi:https://doi.org/10.1093/alcalc/agr057>

---

 LozyEtAl2020

 (TODO) Plotting data from Lozy et al. (2020)
 

---

**Description**

Treatment data from Lozy et al. (2020)

**Usage**

LozyEtAl2020

**Format**

A data frame with 91 rows and 5 variables:

**Session** Session number

**Participant** Participant name

**KM** Kinesthetic movement choices

**TD** Traditional drill choices

**Phase** Sequenced phase number

**Source**

<doi:https://doi.org/10.1002/jaba.677>

---

 print.fxl

 print.fxl
 

---

**Description**

Override the final call to print the fxl object. catches the obj and prints out layers in the sequence laid out by the user

**Usage**

```
## S3 method for class 'fxl'
print(x, ...)
```

**Arguments**

x	fxl object
...	inherits from generic

**Value**

no return, executed for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

*print.fxlsemitlog*      *print.fxlsemitlog*

---

**Description**

Override the final call to print the fxl object. catches the obj and prints out layers in the sequence laid out by the user

**Usage**

```
## S3 method for class 'fxlsemitlog'  
print(x, ...)
```

**Arguments**

x                    fxlsemitlog object  
...                   inherits from generic

**Value**

no return, executed for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

*scr\_anno\_arrows*      *scr\_anno\_arrows*

---

**Description**

Add a layer with arrows to direct attention on the plot

**Usage**

```
scr_anno_arrows(  
  core_frame,  
  arrows = NULL,  
  facet = NULL,  
  color = "black",  
  length = 0.25,  
  angle = 30,  
  code = 2,  
  lwd = 1,  
  lty = 1  
)
```

**Arguments**

core_frame	fxl class
arrows	list of keyed entries to be drawn on respective facets
facet	the facet which will be drawn upon
color	from base
length	from base
angle	from base
code	from base
lwd	from base
lty	from base

**Details**

Generally useful for avoiding a legend

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>



---

scr\_anno\_brackets      *scr\_anno\_brackets*

---

### Description

Add a layer with brackets on plot

### Usage

```
scr_anno_brackets(  
  core_frame,  
  brackets = NULL,  
  facet = NULL,  
  color = "black",  
  length = 0.25,  
  angle = 30,  
  code = 2,  
  lwd = 1,  
  lty = 1  
)
```

### Arguments

core_frame	fxl class
brackets	list of keyed entries to be drawn on respective facets
facet	the facet which will be drawn upon
color	from base
length	from base
angle	from base
code	from base
lwd	from base
lty	from base

### Value

a layer to the core plotting object

### Author(s)

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_anno\_guide\_line    *scr\_anno\_guide\_line*

---

### Description

This is an annotation illustrating an aim/reduction line

### Usage

```
scr_anno_guide_line(  
  core_frame,  
  coords,  
  facet = NA,  
  color = "black",  
  lty = 1,  
  lwd = 1  
)
```

### Arguments

core_frame	fxl object
coords	start and finish coords for aim line
facet	panel to draw upon
color	from base
lty	line type
lwd	line width

### Value

a layer to the core plotting object

### Author(s)

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_anno\_rect                    *scr\_anno\_rect*

---

**Description**

scr\_anno\_rect

**Usage**

```
scr_anno_rect(core_frame, rects = NULL, color = "black", fill = "black")
```

**Arguments**

core_frame	fxl object
rects	list of keyed entries to be drawn on respective facets
color	from base
fill	from base

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_bar\_support                    *scr\_bar\_support*

---

**Description**

Adds a supplemental bar to the figure, if relevant to the data

**Usage**

```
scr_bar_support(  
  core_frame,  
  color = rgb(0.8, 0.8, 0.8, alpha = 0.25),  
  alpha = 1,  
  guide_line = NULL,  
  guide_line_type = 1,  
  guide_line_size = 1,  
  guide_line_color = "black",  
  mapping = NULL,  
  label = "",  
  styler = NA,  
  width = 0.8  
)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>color</code>	from base
<code>alpha</code>	from base
<code>guide_line</code>	(optional) aim line for bars
<code>guide_line_type</code>	(optional) aim line type for bars
<code>guide_line_size</code>	(optional) aim line size for bars
<code>guide_line_color</code>	(optional) aim line color for bars
<code>mapping</code>	(optional) if overriding draw (i.e., different response)
<code>label</code>	description for bar
<code>styler</code>	a lambda function that returns dynamic styling parameters
<code>width</code>	width of bar

**Value**

a layer to the core plotting object

---

`scr_criterion_lines`    *scr\_criterion\_lines*

---

**Description**

`scr_criterion_lines`

**Usage**

```
scr_criterion_lines(
  core_frame,
  lty = 1,
  color = "black",
  size = 1,
  lines = NULL
)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>lty</code>	from base
<code>color</code>	from base
<code>size</code>	from base
<code>lines</code>	lines to draw

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

*scr\_cumsum\_lines*      *scr\_cumsum*

---

**Description**

Draw lines, but as a cumulative and rolling sum

**Usage**

```
scr_cumsum_lines(core_frame, lty = 1, color = "black", size = 1, mapping)
```

**Arguments**

<code>core_frame</code>	fxl object
<code>lty</code>	from base
<code>color</code>	from base
<code>size</code>	from base
<code>mapping</code>	from base

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_cumsum\_points      *scr\_cumsum\_points*

---

### Description

scr\_cumsum\_points

### Usage

```
scr_cumsum_points(  
  core_frame,  
  pch = 21,  
  color = "black",  
  fill = "black",  
  cex = 1,  
  mapping  
)
```

### Arguments

core_frame	fxl object
pch	from base
color	from base
fill	from base
cex	from base
mapping	(optional) if overriding draw (i.e., different response)

### Value

a layer to the core plotting object

### Author(s)

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_images	<i>scr_images</i>
------------	-------------------

---

**Description**

scr\_images

**Usage**

```
scr_images(core_frame, image, cex = 1, mapping)
```

**Arguments**

core_frame	fxl object
image	should be RGML image
cex	from base
mapping	(optional) if overriding draw (i.e., different response)

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_label_facet	<i>scr_label_facet</i>
-----------------	------------------------

---

**Description**

scr\_label\_facet

**Usage**

```
scr_label_facet(  
  core_frame,  
  color = "black",  
  cex = 1,  
  adj = 0.5,  
  face = 1,  
  x = NULL,  
  y = NULL,  
  labels = NULL  
)
```

**Arguments**

core_frame	fxl object
color	from base
cex	from base
adj	from base
face	like 'font' from base
x	global x position for labels
y	global y position for labels
labels	as stated

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_label_phase	<i>scr_label_phase</i>
-----------------	------------------------

---

**Description**

labels to be drawn on plots (typically for phases/conditions, but not necessarily)

**Usage**

```
scr_label_phase(  
  core_frame,  
  color = "black",  
  cex = 1,  
  adj = 0.5,  
  face = 1,  
  x = NULL,  
  y = NULL,  
  facet = NULL,  
  labels = NULL  
)
```



**Arguments**

core_frame	fxl object
color	from base
cex	from base
adj	from base
face	like 'font' from base
x	location
y	location
facet	facet of interest
labels	as stated

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_legend

*scrlegend*

---

**Description**

Information for drawing legend onto plots

**Usage**

```
scr_legend(  
  core_frame,  
  panel = NA,  
  legend,  
  bg = NULL,  
  col = NULL,  
  pt_bg = NULL,  
  lty,  
  pch,  
  box_lty = 0,  
  adj = c(0, 0.5),  
  bty = "n",  
  cex = 1,  
  horiz = FALSE,  
  position = "topright",  
  pt_cex = 1,  
  text_col = "black",  
  border = "black"  
)
```

**Arguments**

core_frame	fxl object
panel	facet to be drawn on
legend	from base
bg	from base
col	from base
pt_bg	color, for point
lty	from base
pch	from base
box_lty	from base
adj	alignment
bty	from base
cex	from base
horiz	from base
position	from base
pt_cex	from base
text_col	from base
border	border status (from base)

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_lines

*scr\_lines*

---

**Description**

scr\_lines

**Usage**

```
scr_lines(core_frame, lty = 1, color = "black", size = 1, mapping)
```

**Arguments**

core_frame	fxl object
lty	from base
color	from base
size	from base
mapping	from base

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_plines	<i>scr_plines</i>
------------	-------------------

---

**Description**

scr\_plines

**Usage**

```
scr_plines(core_frame, lines = NULL, lwd = 1, lty = 1, col = "black")
```

**Arguments**

core_frame	fxl object
lines	phase lines to be drawn
lwd	from base
lty	from base
col	from base

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_plines_mbd	<i>scr_plines_mbd</i>
----------------	-----------------------

---

**Description**

scr\_plines\_mbd

**Usage**

```
scr_plines_mbd(core_frame, lty = 1, lines = NULL)
```

**Arguments**

core_frame	fxl object
lty	phase lines types
lines	phase lines to be drawn

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_plot	<i>scrplot</i>
----------	----------------

---

**Description**

Core object for establishing fxl object and layers

**Usage**

```
scr_plot(  
  data,  
  aesthetics = NULL,  
  mai = c(0.375, 0.375, 0.25, 0.25),  
  omi = c(0.25, 0.25, 0.25, 0.25),  
  xaxs = "i",  
  yaxs = "i",  
  ncol = 1,  
  family = "sans",  
  bty = "l",  
  layout = NA,
```

```

    layout_h = NA,
    layout_v = NA,
    semi_color_major_y = "blue",
    semi_color_midpoint_y = "blue",
    semi_color_minor_y = "lightgray",
    semi_color_major_x = "lightgray",
    semilog = FALSE
  )

```

### Arguments

data	submitted data (not opinionated on naming)
aesthetics	references for data in frame
mai	margins in inches
omi	outer margins in inches
xaxs	x axis formatting, relative to hanging space
yaxs	y axis formatting, relative to hanging space
ncol	TODO
family	font family
bty	TODO
layout	TODO
layout_h	TODO
layout_v	TODO
semi_color_major_y	TODO
semi_color_midpoint_y	TODO
semi_color_minor_y	TODO
semi_color_major_x	TODO
semilog	determine if this is a semilog type of plot

### Value

class of 'fxl' that contains necessary plotting elements

### Author(s)

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_points	<i>scr_points</i>
------------	-------------------

---

**Description**

scr\_points

**Usage**

```
scr_points(  
  core_frame,  
  pch = 21,  
  color = "black",  
  fill = "black",  
  cex = 1,  
  styler = NA,  
  data = NA,  
  mapping  
)
```

**Arguments**

core_frame	fxl object
pch	from base
color	from base
fill	from base
cex	from base
styler	a lambda function that returns dynamic styling parameters
data	(optional) if overriding data
mapping	(optional) if overriding draw (i.e., different response)

**Value**

a layer to the core plotting object

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

`scr_save``scrsave`

---

### Description

Function for outputting fxl object at preset size (certain journal are opinionated on size, format, and density)

### Usage

```
scr_save(  
  core_frame,  
  units = "in",  
  name = "test.tiff",  
  format = "tiff",  
  width = 8,  
  height = 4,  
  res = 600  
)
```

### Arguments

<code>core_frame</code>	fxl object
<code>units</code>	from base
<code>name</code>	from base
<code>format</code>	type of image to save in
<code>width</code>	from base
<code>height</code>	from base
<code>res</code>	from base

### Value

no return, executed for side effects

### Author(s)

Shawn Gilroy <sgilroy1@lsu.edu>

scr\_title                    *scrtitle*

---

**Description**

Override the title

**Usage**

```
scr_title(core_frame, var, color = "black", cex = 1, adj = 0.5, face = 1)
```

**Arguments**

core_frame	fxl object
var	string
color	from base
cex	from base
adj	from base
face	like 'font' from base

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_xlabel                    *xlabel*

---

**Description**

Override the x axis label

**Usage**

```
scr_xlabel(  
  core_frame,  
  var,  
  color = "black",  
  cex = 1,  
  adj = 0.5,  
  face = 1,  
  line = 0  
)
```



**Arguments**

core_frame	fxl object
var	string
color	from base
cex	from base
adj	from base
face	like 'font' from base
line	line width

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_xoverride	<i>xoverride</i>
---------------	------------------

---

**Description**

Override the x axis limits

**Usage**

```
scr_xoverride(  
  core_frame,  
  var,  
  xdelta = 1,  
  xticks = NULL,  
  xdraws = NULL,  
  xrotation = NULL,  
  xticksceex = 1,  
  xlabeloffset = NULL,  
  xtickslabs = NULL,  
  xticksadj = 1  
)
```

**Arguments**

core_frame	fxl object
var	string for title
xdelta	skips between ticks (can override)
xticks	specify ticks, vector or named list
xdraws	which x axes to draw
xrotation	degree to rotate positioned labels
xticksceX	expansion factor for labels
xlabeloffset	offset to push labels downward
xtickslabs	labels for x axis
xticksadj	alignment for custom labels

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr\_ylabel                      *ylabel*

---

**Description**

Override the y axis label

**Usage**

```
scr_ylabel(  
  core_frame,  
  var,  
  color = "black",  
  cex = 1,  
  adj = 0.5,  
  face = 1,  
  line = 0  
)
```

**Arguments**

core_frame	fxl object
var	string
color	from base
cex	from base
adj	from base
face	like 'font' from base
line	line width

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

scr_yoverride	<i>yoverride</i>
---------------	------------------

---

**Description**

Override the y axis (or axes) limits

**Usage**

```
scr_yoverride(
  core_frame,
  var,
  ydelta = 1,
  yticks = NULL,
  ydraws = NULL,
  ytickslabs = NULL
)
```

**Arguments**

core_frame	fxl object
var	from base
ydelta	skips between ticks (can override)
yticks	tick values (numerical)
ydraws	specify axes manual
ytickslabs	tick labels

**Value**

nothing, run for side effects

**Author(s)**

Shawn Gilroy <sgilroy1@lsu.edu>

---

SimulatedAcademicFluency

*Plotting data for Hypothetical Academic MTSS*

---

**Description**

Plotting data for Hypothetical Academic MTSS

**Usage**

SimulatedAcademicFluency

**Format**

A data frame with 168 rows and 7 variables:

**Rates** Rates of change

**Times** Multiplier for model level

**index** Individual id

**starts** Modeled baseline start

**jitter** Jitter offset

**pred** Prediction from model

**err** Residual error

---

var\_map

*var\_map*

---

**Description**

This helper function maps out relationships to be parsed later on

**Usage**

var\_map(...)

**Arguments**

... map expressed relationships out

*var\_map*

45

**Value**

list of exprs to map variables to plotting methods

# Index

## \* datasets

- Challenge1Data, [3](#)
  - Challenge2Data, [4](#)
  - Challenge4Data, [5](#)
  - GelinoEtAl2022, [15](#)
  - Gilroyetal2015, [16](#)
  - Gilroyetal2019, [16](#)
  - Gilroyetal2019Tx, [17](#)
  - Gilroyetal2021, [18](#)
  - KoffarnusEtAl2011, [21](#)
  - LozyEtAl2020, [22](#)
  - SimulatedAcademicFluency, [44](#)
- [assert\\_input\\_type, 3](#)
- [Challenge1Data, 3](#)
- [Challenge2Data, 4](#)
- [Challenge4Data, 5](#)
- [cnvrt\\_coords, 5](#)
- [draw\\_arrows, 6](#)
- [draw\\_bar\\_support, 7](#)
- [draw\\_brackets, 7](#)
- [draw\\_cumsum\\_lines, 8](#)
- [draw\\_cumsum\\_points, 8](#)
- [draw\\_guide\\_line, 9](#)
- [draw\\_images, 10](#)
- [draw\\_label\\_facet, 10](#)
- [draw\\_label\\_phase, 11](#)
- [draw\\_legend, 11](#)
- [draw\\_lines, 12](#)
- [draw\\_points, 12](#)
- [draw\\_rect, 13](#)
- [draw\\_scr\\_criterion, 14](#)
- [draw\\_scr\\_plines, 14](#)
- [GelinoEtAl2022, 15](#)
- [Gilroyetal2015, 16](#)
- [Gilroyetal2019, 16](#)
- [Gilroyetal2019Tx, 17](#)
- [Gilroyetal2021, 18](#)
- [isValidAestheticMapping, 18](#)
- [isValidAXSCharacter, 19](#)
- [isValidCharacterVector, 19](#)
- [isValidDataFrame, 20](#)
- [isValidLogicalVector, 20](#)
- [isValidNumericVector, 21](#)
- [KoffarnusEtAl2011, 21](#)
- [LozyEtAl2020, 22](#)
- [print.fx1, 22](#)
- [print.fx1semilog, 23](#)
- [scr\\_anno\\_arrows, 23](#)
- [scr\\_anno\\_brackets, 25](#)
- [scr\\_anno\\_guide\\_line, 26](#)
- [scr\\_anno\\_rect, 27](#)
- [scr\\_bar\\_support, 27](#)
- [scr\\_criterion\\_lines, 28](#)
- [scr\\_cumsum\\_lines, 29](#)
- [scr\\_cumsum\\_points, 30](#)
- [scr\\_images, 31](#)
- [scr\\_label\\_facet, 31](#)
- [scr\\_label\\_phase, 32](#)
- [scr\\_legend, 33](#)
- [scr\\_lines, 34](#)
- [scr\\_plines, 35](#)
- [scr\\_plines\\_mbd, 36](#)
- [scr\\_plot, 36](#)
- [scr\\_points, 38](#)
- [scr\\_save, 39](#)
- [scr\\_title, 40](#)
- [scr\\_xlabel, 40](#)
- [scr\\_xoverride, 41](#)
- [scr\\_ylabel, 42](#)
- [scr\\_yoverride, 43](#)
- [SimulatedAcademicFluency, 44](#)
- [var\\_map, 44](#)