

Package ‘safetyCharts’

July 23, 2025

Title Charts for Monitoring Clinical Trial Safety

Version 0.3.0

Maintainer Jeremy Wildfire <jwildfire@gmail.com>

Description Contains chart code for monitoring clinical trial safety. Charts can be used as standalone output, but are also designed for use with the 'safetyGraphics' package, which makes it easy to load data and customize the charts using an interactive web-based interface created with Shiny.

URL <https://github.com/SafetyGraphics/safetyCharts>

BugReports <https://github.com/SafetyGraphics/safetyCharts/issues>

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Imports dplyr, DT, forcats, ggplot2, htmlwidgets, huxtable, jsonlite, pharmaRTF, plotly, purrr, RColorBrewer, rlang, shiny, stringr, Tendiril, Tplyr

Suggests testthat, shinytest, safetyData, safetyGraphics, yaml

Depends R (>= 4.0)

NeedsCompilation no

Author Jeremy Wildfire [aut, cre]

Repository CRAN

Date/Publication 2022-03-22 20:00:02 UTC

Contents

demogRTF_server	2
demogRTF_table	3
demogRTF_ui	3
hepExplorer	4
init_aeExplorer	5

init_aeTimelines	6
init_paneledOutlierExplorer	7
init_safetyOutlierExplorer	7
init_safetyResultsOverTime	8
init_safetyShiftPlot	8
lab_distribution_server	9
lab_distribution_ui	9
meta_aes	10
meta_dm	10
meta_ecg	11
meta_hepExplorer	12
meta_labs	13
QT_OutlierExplorer_server	13
QT_OutlierExplorer_ui	14
QT_Outlier_Explorer	14
render_widget	15
safetyOutlierExplorer_server	16
safetyOutlierExplorer_ui	16
safety_outlier_explorer	17
safety_results_over_time	18
tendril_chart	19
Index	21

demogRTF_server	<i>Demographics Table RTF - UI</i>
-----------------	------------------------------------

Description

Demographics Table RTF - UI

Usage

demogRTF_server(input, output, session, params)

Arguments

input	module input
output	module output
session	module session
params	parameters object with data and settings options.

Value

returns shiny module Server function

demogRTF_table	<i>create demographics RTF table</i>
----------------	--------------------------------------

Description

create demographics RTF table

Usage

```
demogRTF_table(data, settings)
```

Arguments

- | | |
|----------|--|
| data | demographics data frame with columns specified in settings object |
| settings | list with parameters specifying the column names for: <ul style="list-style-type: none">• sex (settings\$sex_col),• race (settings\$race_col)• age (settings\$age_Col) |

Value

rtf doc object

Examples

```
settings <- list(treatment_col = "ARM", sex_col = "SEX", race_col = "RACE", age_col = "AGE")
demogRTF_table(safetyData::sdm_dm, settings)
```

demogRTF_ui	<i>Demographics Table RTF - UI</i>
-------------	------------------------------------

Description

Demographics Table RTF - UI

Usage

```
demogRTF_ui(id)
```

Arguments

id	module id
----	-----------

Value

returns shiny module UI

hepExplorer	<i>Make standalone hepExplorer html widget</i>
-------------	--

Description

Convenience mapping of render_widget for hepExplorer.

Usage

```
hepExplorer(df = safetyData::adam_adlbc, mapping = NULL, ...)
```

Arguments

df	data frame containing lab data used to render for hepExplorer. Default is safetyData::adam_adlbc.
mapping	named list with the current data mappings. See details for default mapping.
...	additional options to be added to mapping. Will overwrite mapping.

Details

The **data** and **mapping** should match the specs described in the **hepExplorer** javascript library. Items passed in ... are added to mapping, and then the list is converted to json via `jsonlite::toJSON(mapping, auto_unbox=TRUE, null="null")`.

The default mapping shown below is designed to work with data in the CDISC ADaM format (like `safetydata::adam_adlbc`).

```
mapping <- list(
  measure_col = "PARAM",
  measure_values = list(
    ALT = "Alanine Aminotransferase (U/L)",
    AST = "Aspartate Aminotransferase (U/L)",
    TB = "Bilirubin (umol/L)",
    ALP = "Alkaline Phosphatase (U/L)"
  ),
  id_col = "USUBJID",
  value_col = "AVAL",
  normal_col_low = "A1LO",
  normal_col_high = "A1HI",
  studyday_col = "ADY",
  visit_col = "VISIT",
  visitn_col = "VISITNUM"
)
```

Parameters that are not included in the default mapping can be accessed via ...; Key options and defaults for `safetyData::adam_adlbc` shown below:

- **filters**: list of columns to be included as data filters (e.g. `filters=c("SEX","AGEGR1")`)

- `group_cols`: list of columns used to define grouping and set point color (e.g. `'filters=c("SEX","AGEGR1")'`)
- `x_options` and `y_options` - specify which labs can be used for x and y axis dropdowns. By default, all options are included on x-axis, but only Bilirubin is shown on y-axis. To allow an interactive y-axis, use `y_options="all"`.
- `baseline` - flag defining the baseline visit for each participant. `baseline` must be provided to enable the mDish view on the hep-explorer chart. Define as a list with `value_col` and `values` (e.g. `baseline=list(value_col="ABLFL", values="Y")`)
- `title` and `warningText` - Strings used to define the header text shown above the filters.

For more options see the [full specs](#) in the javascript library.

Examples

```
## Not run:
# Render widget with defaults
hepExplorer()

# Add age group to default
hepExplorer(group_cols=c("SEX", "AGEGR1"))

# Enable interactive y-axis
hepExplorer(y_options='all')

# Use custom mapping for SDTM data
hepExplorer(
  df=safetyData::sdm_lb,
  measure_col = "LBTEST",
  measure_values = list(
    ALT = "Alanine Aminotransferase",
    AST = "Aspartate Aminotransferase",
    TB = "Bilirubin",
    ALP = "Alkaline Phosphatase"
  ),
  id_col = "USUBJID",
  value_col = "LBSTRESN",
  normal_col_low = "LBORNRL0",
  normal_col_high = "LBORNRI",
  studyday_col = "LBDY",
  visit_col = "VISIT",
  visitn_col = "VISITNUM"
)

## End(Not run)
```

Description

Initialize Settings for Adverse Event Explorer widget

Usage

init_aeExplorer(data, settings)

Arguments

- data labs data structured as one record per person per visit per measurement. See details for column requirements.
- settings named list of settings

Value

returns list with data and settings

init_aeTimelines	<i>Initialize Settings for AE Timeline widget</i>
------------------	---

Description

Initialize Settings for AE Timeline widget

Usage

init_aeTimelines(data, settings)

Arguments

- data labs data structured as one record per person per visit per measurement. See details for column requirements.
- settings named list of settings

Value

returns list with data and settings

`init_paneledOutlierExplorer`*Initialize Settings for Paneled Outlier Explorer widget*

Description

Initialize Settings for Paneled Outlier Explorer widget

Usage

```
init_paneledOutlierExplorer(data, settings)
```

Arguments

data	labs data structured as one record per person per visit per measurement. See details for column requirements.
settings	named list of settings

Value

returns list with data and settings

`init_safetyOutlierExplorer`*Initialize Settings for Safety Outlier Explorer widget*

Description

Initialize Settings for Safety Outlier Explorer widget

Usage

```
init_safetyOutlierExplorer(data, settings)
```

Arguments

data	labs data structured as one record per person per visit per measurement. See details for column requirements.
settings	named list of settings

Value

returns list with data and settings

`init_safetyResultsOverTime`*Initialize Settings for Safety Results Over Time widget*

Description

Initialize Settings for Safety Results Over Time widget

Usage

```
init_safetyResultsOverTime(data, settings)
```

Arguments

<code>data</code>	labs data structured as one record per person per visit per measurement. See details for column requirements.
<code>settings</code>	named list of settings

Value

returns list with data and settings

`init_safetyShiftPlot` *Initialize Settings for Safety Shift Plot widget*

Description

Initialize Settings for Safety Shift Plot widget

Usage

```
init_safetyShiftPlot(data, settings)
```

Arguments

<code>data</code>	labs data structured as one record per person per visit per measurement. See details for column requirements.
<code>settings</code>	named list of settings

Value

returns list with data and settings

`lab_distribution_server`*lab distribution Module - Server*

Description

A simple server for a shiny module looking at lab histograms. Intended primarily for technical demos.

Usage

```
lab_distribution_server(input, output, session, params)
```

Arguments

input	module input
output	module output
session	module session
params	parameters object with data and settings options.

Value

returns shiny module Server function

`lab_distribution_ui` *Lab distribution Module - UI*

Description

A simple UI for a shiny module looking at lab histograms. Intended primarily for technical demos.

Usage

```
lab_distribution_ui(id)
```

Arguments

id	module id
----	-----------

Value

returns shiny module UI

meta_aes	<i>Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the ae domain. One record per unique data mapping</i>
----------	--

Description

Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the ae domain. One record per unique data mapping

Usage

meta_aes

Format

A data frame with X rows and 10 columns

domain Data domain

text_key Text key indicating the setting name. ' -- ' delimiter indicates a field level data mapping

col_key Key for the column mapping

field_key Key for the field mapping (if any)

type type of mapping - "field" or "column"

label Label

description Description

multiple Mapping supports multiple columns/fields

standard_adam Default values for the ADaM data standard

standard_sdtm Default values for the SDTM data standard

Source

Created for this package

meta_dm	<i>Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the dm domain. One record per unique data mapping</i>
---------	--

Description

Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the dm domain. One record per unique data mapping

Usage

```
meta_dm
```

Format

A data frame with X rows and 10 columns

domain Data domain

text_key Text key indicating the setting name. ' -- ' delimiter indicates a field level data mapping

col_key Key for the column mapping

field_key Key for the field mapping (if any)

type type of mapping - "field" or "column"

label Label

description Description

multiple Mapping supports multiple columns/fields

standard_adam Default values for the ADaM data standard

standard_sdtm Default values for the SDTM data standard

Source

Created for this package

meta_ecg	<i>Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the ecg domain. One record per unique data mapping</i>
----------	---

Description

Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the ecg domain. One record per unique data mapping

Usage

```
meta_ecg
```

Format

A data frame with 22 rows and 10 columns

domain Data domain

text_key Text key indicating the setting name. ' -- ' delimiter indicates a field level data mapping

col_key Key for the column mapping

field_key Key for the field mapping (if any)

type type of mapping - "field" or "column"

label Label

description Description

multiple Mapping supports multiple columns/fields

standard_adam Default values for the ADaM data standard

standard_sdtm Default values for the SDTM data standard

Source

Created for this package

meta_hepExplorer	<i>Metadata data frame containing information about the data mapping used to configure safetyGraphics for the hepExplorer Chart. One record per unique data mapping</i>
------------------	---

Description

Metadata data frame containing information about the data mapping used to configure safetyGraphics for the hepExplorer Chart. One record per unique data mapping

Usage

```
meta_hepExplorer
```

Format

A data frame with X rows and 10 columns

domain Data domain

text_key Text key indicating the setting name. '--' delimiter indicates a field level data mapping

col_key Key for the column mapping

field_key Key for the field mapping (if any)

type type of mapping - "field" or "column"

label Label

description Description

multiple Mapping supports multiple columns/fields

standard_adam Default values for the ADaM data standard

standard_sdtm Default values for the SDTM data standard

Source

Created for this package

meta_labs	<i>Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the labs domain. One record per unique data mapping</i>
-----------	--

Description

Metadata data frame containing information about the data mapping used to configure safetyGraphics charts for the labs domain. One record per unique data mapping

Usage

```
meta_labs
```

Format

A data frame with X rows and 10 columns

domain Data domain

text_key Text key indicating the setting name. ' -- ' delimiter indicates a field level data mapping

col_key Key for the column mapping

field_key Key for the field mapping (if any)

type type of mapping - "field" or "column"

label Label

description Description

multiple Mapping supports multiple columns/fields

standard_adam Default values for the ADaM data standard

standard_sdtm Default values for the SDTM data standard

Source

Created for this package

QT_OutlierExplorer_server	<i>QT Outlier Explorer Module - UI</i>
---------------------------	--

Description

QT Outlier Explorer Module - UI

Usage

```
QT_OutlierExplorer_server(input, output, session, params)
```

Arguments

input	module input
output	module output
session	module session
params	parameters object with data and settings options.

Value

returns shiny module Server function

QT_OutlierExplorer_ui	<i>QT Outlier Explorer Module - UI</i>
-----------------------	--

Description

QT Outlier Explorer Module - UI

Usage

QT_OutlierExplorer_ui(id)

Arguments

id	module id
----	-----------

Value

returns shiny module UI

QT_Outlier_Explorer	<i>QT Outlier Explorer</i>
---------------------	----------------------------

Description

QT Outlier Explorer

Usage

QT_Outlier_Explorer(data, settings)

Arguments

data	ECG data structured as one record per person per visit per measurement. See details for column requirements.
settings	named list of settings with the parameters specified below.

Details

The settings object provides details the columns in the data set.

- "id_col"ID column
- "value_col"Value column
- "measure_col"Measure column
- "measure_values"Measure values
- "visit_col"Visit column
- "visitn_col"Visit number column (numeric)
- "baseline_flag_col"Baseline flag column
- "baseline_flag_values"Baseline flag value

Value

returns a chart object

render_widget	<i>Render an htmlwidget using standard safetyGraphics workflow</i>
---------------	--

Description

Render an htmlwidget using standard safetyGraphics workflow

Usage

render_widget(widgetName, data, mapping)

Arguments

- | | |
|------------|---|
| widgetName | name of the widget saved in safetyCharts |
| data | named list of current data sets |
| mapping | named list with the current data mappings |

safetyOutlierExplorer_server
<i>Safety Outlier Explorer Module - UI</i>

Description

Safety Outlier Explorer Module - UI

Usage

safetyOutlierExplorer_server(input, output, session, params)

Arguments

input	module input
output	module output
session	module session
params	parameters object with data and settings options.

Value

returns shiny module Server function

safetyOutlierExplorer_ui
<i>Safety Outlier Explorer Module - UI</i>

Description

Safety Outlier Explorer Module - UI

Usage

safetyOutlierExplorer_ui(id)

Arguments

id	module id
----	-----------

Value

returns shiny module UI

`safety_outlier_explorer`*Safety Outlier Explorer*

Description

Safety Outlier Explorer

Usage

```
safety_outlier_explorer(data, settings)
```

Arguments

<code>data</code>	labs data structured as one record per person per visit per measurement. See details for column requirements.
<code>settings</code>	named list of settings with the parameters specified below.

Details

The settings object provides details the columns in the data set.

- `"id_col"` ID column
- `"value_col"` Value column
- `"measure_col"` Measure column
- `"measure_values"` Measure values
- `"studyday_col"` Study Day (numeric)

Value

returns a chart object

Examples

```
settings <- list(  
  id_col = "USUBJID",  
  measure_col = "LBTEST",  
  measure_values = c("Albumin", "Bilirubin", "Chloride"),  
  studyday_col = "VISITDY",  
  value_col = "LBORRES"  
)  
safety_outlier_explorer(safetyData::sdtm_lb, settings)
```

safety_results_over_time

Safety Results Over Time plot

Description

Safety Results Over Time plot

Usage

```
safety_results_over_time(data, settings)
```

Arguments

data	labs data structured as one record per person per visit per measurement. See details for column requirements.
settings	named list of settings with the parameters specified below.

Details

The settings object provides details the columns in the data set.

- "value_col" Value column
- "measure_col" Measure column
- "measure_values" Measure values
- "visit_col" Study Visit
- "visitn_col" Study Number
- "group_col" Grouping column
- "violins" Show Violin plots?
- "boxplots" Show Box Plots?
- "axis" set to "log" to use a log transformed axis, linear otherwise
- "drop_visit_string" Drop visits that contain this string. e.g. "unscheduled"

Value

returns a chart object

Examples

```
library(dplyr)
lb <- safetyData::sdm_lb
sub_ids <- unique(lb$USUBJID)[1:100]
lb <- lb %>% filter(USUBJID %in% sub_ids)
settings <- list(
  value_col = "LBORRES",
```

```

    measure_col = "LBTEST",
    measure_values = c("Chloride"),
    visit_col = "VISIT",
    visitn_col = "VISITNUM",
    axis = "log"
  )
safety_results_over_time(lb, settings)

# remove unscheduled visits, add violin plot and 2nd panel
settings$drop_visit_string <- "unscheduled"
settings$violins <- TRUE
settings$measure_values <- c("Albumin")
safety_results_over_time(lb, settings)

# add grouping by treatment
dm_sub <- safetyData::sdm_dm %>% select(USUBJID, ARM)
dm_lb <- dm_sub %>% left_join(lb)
settings$group_col <- "ARM"
safety_results_over_time(dm_lb, settings)

```

tendrill_chart

*Tendrill plot***Description**

Create a plot using the Tendril package

Usage

```
tendrill_chart(data, settings)
```

Arguments

data	list of data frames including dataframes named aes (adverse events) and dm (demographics)
settings	named list of domain-specific settings with the parameters specified below.

Details

The settings object provides details regarding the columns in the data sets.

- "settings\$dm\$id_col" ID column
- "settings\$dm\$treatment_col" Treatment column
- "settings\$dm\$treatment_values-group1" Name of treatment 1
- "settings\$dm\$treatment_values-group2" Name of treatment 2
- "settings\$aes\$id_col" ID column)
- "settings\$aes\$bodsys_col" Body System
- "settings\$aes\$stdy_col" Study Day

Value

returns a chart object

Index

* datasets

- meta_aes, [10](#)
- meta_dm, [10](#)
- meta_ecg, [11](#)
- meta_hepExplorer, [12](#)
- meta_labs, [13](#)

demogRTF_server, [2](#)

demogRTF_table, [3](#)

demogRTF_ui, [3](#)

hepExplorer, [4](#)

init_aeExplorer, [5](#)

init_aeTimelines, [6](#)

init_paneledOutlierExplorer, [7](#)

init_safetyOutlierExplorer, [7](#)

init_safetyResultsOverTime, [8](#)

init_safetyShiftPlot, [8](#)

lab_distribution_server, [9](#)

lab_distribution_ui, [9](#)

meta_aes, [10](#)

meta_dm, [10](#)

meta_ecg, [11](#)

meta_hepExplorer, [12](#)

meta_labs, [13](#)

QT_Outlier_Explorer, [14](#)

QT_OutlierExplorer_server, [13](#)

QT_OutlierExplorer_ui, [14](#)

render_widget, [15](#)

safety_outlier_explorer, [17](#)

safety_results_over_time, [18](#)

safetyOutlierExplorer_server, [16](#)

safetyOutlierExplorer_ui, [16](#)

tendrill_chart, [19](#)