Package 'TabularManifest'

March 20, 2014

Title Tabular Manifest

Description Assists the manipulation and exploration of wide datasets with tabular configuration files

Date 2014-03-20

Version 0.1-15

License LGPL

LazyData TRUE

VignetteBuilder knitr

Maintainer Will Beasley <wibeasley@hotmail.com>

URL http://melinae.com/

Depends R(>= 3.0.0),stats

Imports ggplot2,grid,mgcv,plyr,scales

Suggests datasets,devtools,knitr,RColorBrewer,testit,testthat

R topics documented:

calculate_bins 2
calculate_rounding_digits 2
construct_graph 3
create_manifest_explore_univariate 3
histogram_continuous 5
histogram_discrete 6
scatter_model_continuous_x_binary_y_logit 7
scatter_model_discrete_x_binary_y_logit 8

Index **9**

calculate_bins *Internal function for creating default bins for dataset variables.*

Description

An internal function (ie, that's not currently exposed/exported outside the package) for creating default bins for dataset variables.

Usage

```
calculate_bins(ds_observed, bin_count_suggestion = 30L)
```

Arguments

ds_observed	The data.frame with columns to calculate bins.
bin_count_suggestion	An integer or numeric value for the suggested number of bins for each variable.

Value

Returns a list, with two elements. Each element is an array with as many values as columns in ds_observed.

1. bin_width The variable name (in ds_observed).
2. bin_start The variable's class. (eg, numeric, Date, factor)

Examples

```
#tabularmanifest:::calculate_bins(ds_observed=datasets::freeny)
#tabularmanifest:::calculate_bins(ds_observed=datasets::InsectSprays)
```

calculate_rounding_digits *Internal function for calculating rounding digits for dataset variables*

Description

An internal function (ie, that's not currently exposed/exported outside the package) for creating default bins for dataset variables.

Usage

```
calculate_rounding_digits(ds_observed)
```

Arguments

ds_observed	The data.frame with columns to calculate bins.

Value

Returns a numeric vector, indicating how many rounding digits *might* be appropriate. Each element is an array with as many values as columns in ds_observed.

Examples

```
tabularmanifest:::calculate_rounding_digits(ds_observed=freeny)
tabularmanifest:::calculate_rounding_digits(ds_observed=InsectSprays)
tabularmanifest:::calculate_rounding_digits(ds_observed=beaver1)
```

---

construct_graph *Construct a graph or list of graphs*

Description

Construct a graph or list of graphs, whose characteristics are determined by a configuration file.

Usage

```
construct_graph_univariate(variable_name, ds_metadata, ds_observed)
```

Arguments

variable_name	The name of the single variable to graph.
ds_metadata	The data.frame containing the metadata. See create_manifest_explore_univariate.
ds_observed	The data.frame containing the data to be graphed.

Examples

```
#ds_observed <- beaver1
ds_observed <- InsectSprays
ds_manifest <- tabularmanifest::create_manifest_explore_univariate(ds_observed, write_to_disk=FALSE)

construct_graph_univariate(variable_name="count", ds_manifest, InsectSprays)

construct_graph_list_univariate(ds_manifest=ds_manifest, ds_observed=ds_observed)
```

---

create_manifest_explore_univariate

Create a manifest for exploratoring univariate patterns.

Description

This function creates a meta-dataset (from the data.frame passed as a parameter) and optionally saves the meta-dataset as a CSV. The meta-dataset specifies how the variables should be plotted.

Usage

```
create_manifest_explore_univariate(
    ds_observed,
    write_to_disk = TRUE,
    path_out = getwd(),
    overwrite_file = FALSE,
    default_class_graph = c(
      numeric = "histogram_continuous",
      integer = "histogram_continuous",
      factor = "histogram_discrete",
      character = "histogram_discrete",
      notMatched = "histogram_generic"
    ),
    default_format = c(
      numeric = "scales::comma",
      notMatched = "scales::comma"
    ),
    bin_count_suggestion = 30L
)
```

Arguments

`ds_observed`	The `data.frame` to create metadata for.
`write_to_disk`	Indicates if the meta-dataset should be saved as a CSV.
`path_out`	The file path to save the meta-dataset.
`overwrite_file`	Indicates if the CSV of the meta-dataset should be overwritten if a file already exists at the location.
`default_format`	A `character` array indicating which formatting function should be displayed on the axis of the univariate graph.
`default_class_graph`	A `character` array indicating which graph should be used with variables of a certain class.
`bin_count_suggestion`	An `integer` value of the number of roughly the number bins desired for a histogram.

Value

Returns a `data.frame` where each row in the metadata represents a column in `ds_observed`. The metadata contains the following columns

1. `variable_name` The variable name (in `ds_observed`). `character`.
2. `remark` A blank field that allows theuser to enter notes in the CSV for later reference.
3. `class` The variable's `class` (eg, numeric, Date, factor). `character`.
4. `should_graph` A boolean value indicating if the variable should be graphed. `logical`.
5. `graph_function` The name of the function used to graph the variable. `character`.
6. `x_label_format` The name of the function used to format the *x*-axis. `character`.
7. `bin_width` The uniform width of the bins. `numeric`.
8. `bin_start` The location of the left boundary of the first bin. `numeric`.

Examples

```
create_manifest_explore_univariate(datasets::InsectSprays, write_to_disk=FALSE)

#Careful, the first column is a `ts` class.
create_manifest_explore_univariate(datasets::freeny, write_to_disk=FALSE)
```

---

histogram_continuous *Generate a Histogram for a* numeric *or* integer *variable.*

Description

Generate a histogram for a numeric or integer variable. This graph is intended to quickly provide the researcher with a quick, yet thorough representation of the continuous variable. The additional annotations may not be desired for publication-quality plots.

Usage

```
histogram_continuous(ds_observed, variable_name, bin_width = NULL,
  main_title = variable_name, x_title = paste0(variable_name,
  " (each bin is ", scales::comma(bin_width), " units wide)"),
  y_title = "Frequency", rounded_digits = 0L)
```

Arguments

ds_observed	The data.frame with the variable to graph.
variable_name	The name of the variable to graph. character.
bin_width	The width of the histogram bins. If NULL, the ggplot2 default is used. numeric.
main_title	The desired title on top of the graph. Defaults to variable_name. If no title is desired, pass a value of NULL. character.
x_title	The desired title on the *x*-axis. Defaults to the variable_name and the bin_width. If no axis title is desired, pass a value of NULL. character.
y_title	The desired title on the *y*-axis. Defaults to "Frequency". If no axis title is desired, pass a value of NULL. character.
rounded_digits	The number of decimals to show for the mean and median annotations. character.

Value

Returns a histogram as a ggplot2 object.

Examples

```
library(datasets)
#Don't run graphs on a headless machine without any the basic graphics packages installed.
if( require(grDevices) ) {
  histogram_continuous(ds_observed=beaver1, variable_name="temp", bin_width=.1)
}
```

histogram_discrete *Generate a Histogram for a* character *or* factor *variable.*

Description

Generate a histogram for a character or factor variable. This graph is intended to quickly provide the researcher with a quick, yet thorough representation of the continuous variable. The additional annotations may not be desired for publication-quality plots.

Usage

```
histogram_discrete(ds_observed, variable_name,
  levels_to_exclude = character(0), main_title = variable_name,
  x_title = NULL, y_title = "Number of Included Records",
  text_size_percentage = 6, bin_width = 1L)
```

Arguments

ds_observed	The data.frame with the variable to graph.
variable_name	The name of the variable to graph. character.
levels_to_exclude	An array of of the levels to be excluded from the histogram. Pass an empty variable (*ie*, character(0)) if all levels are desired; this is the default. character.
main_title	The desired title on top of the graph. Defaults to variable_name. If no title is desired, pass a value of NULL. character.
x_title	The desired title on the *x*-axis. Defaults to the number of included records. If no axis title is desired, pass a value of NULL. character.
y_title	The desired title on the *y*-axis. Defaults to "Frequency". If no axis title is desired, pass a value of NULL. character.
text_size_percentage	The size of the percentage values on top of the bars. character.
bin_width	(This parameter is included for compatibility with other graphing functions. It should always be 1 for discrete and boolean variables.)

Value

Returns a histogram as a ggplot2 object.

Examples

```
library(datasets)
#Don't run graphs on a headless machine without any the basic graphics packages installed.
if( require(grDevices) ) {
  histogram_discrete(ds_observed=infert, variable_name="education")
  histogram_discrete(ds_observed=infert, variable_name="age")
}
```

`scatter_model_continuous_x_binary_y_logit` *Internal function for examining a logit performance*

Description

Internal function for examining a logit performance

Usage

```
scatter_model_continuous_x_binary_y_logit(ds_plot, x_name, y_name = "y",
  yhat_name = "yhat", residual_name = "residual", alpha_point = 0.05,
  alpha_se_band = 0.15, x_label_format = scales::comma,
  color_smooth_observed = "#1b9e77", color_smooth_predicted = "#d95f02",
  color_smooth_residual = "#7570b3", vertical_limits = c(-0.05, 1.05),
  jitter_observed = ggplot2::position_jitter(w = 0, h = 0.2),
  jitter_predicted = ggplot2::position_jitter(w = 0, h = 0),
  seed_value = NA_real_)
```

Arguments

`ds_plot`	The `data.frame` of observed and predicted values to plot.
`x_name`	The name of the predictor `character`.
`y_name`	The name of the observed response `character`.
`yhat_name`	The name of the predicted response `character`.
`residual_name`	The name of the model residual. `character`.
`alpha_point`	The transparency of each plotted point. A `numeric` value from 0 to 1.
`alpha_se_band`	The transparency of the standard error bands. A `numeric` value from 0 to 1.
`x_label_format`	The name of the function used to format the *x*-axis. `character`.
`color_smooth_observed`	The plotted color of the observed values' GAM trend. `character`.
`color_smooth_predicted`	The plotted color of the predicted's GAM trend. `character`.
`color_smooth_residual`	The plotted color of the residual's GAM trend. `character`.
`vertical_limits`	The plotted limits of the response variable. A two-element `numeric` array.
`jitter_observed`	A function dictating how the observed values are jittered.
`jitter_predicted`	A function dictating how the predicted values are jittered.
`seed_value`	The value of the RNG seed, which affects jittering. No seed is set if a value of `NA` is passed. `numeric`.

`scatter_model_discrete_x_binary_y_logit` *Internal function for examining a logit performance*

Description

Internal function for examining a logit performance

Usage

```
scatter_model_discrete_x_binary_y_logit(ds_plot, x_name, y_name = "y",
  yhat_name = "yhat", residual_name = "residual", alpha_point = 0.05,
  alpha_se_band = 0.15, x_label_format = scales::comma,
  color_smooth_observed = "#1b9e77", color_smooth_predicted = "#d95f02",
  color_smooth_residual = "#7570b3", color_group_count = "tomato",
  vertical_limits = c(-0.05, 1.05),
  jitter_observed = ggplot2::position_jitter(w = 0.35, h = 0.2),
  jitter_predicted = ggplot2::position_jitter(w = 0.35, h = 0),
  seed_value = NA_real_)
```

Arguments

`ds_plot`	The `data.frame` of observed and predicted values to plot.
`x_name`	The name of the predictor `character`.
`y_name`	The name of the observed response `character`.
`yhat_name`	The name of the predicted response `character`.
`residual_name`	The name of the model residual. `character`.
`alpha_point`	The transparency of each plotted point. A `numeric` value from 0 to 1.
`alpha_se_band`	The transparency of the standard error bands. A `numeric` value from 0 to 1.
`x_label_format`	The name of the function used to format the *x*-axis. `character`.
`color_smooth_observed`	The plotted color of the observed values' GAM trend. `character`.
`color_smooth_predicted`	The plotted color of the predicted's GAM trend. `character`.
`color_smooth_residual`	The plotted color of the residual's GAM trend. `character`.
`color_group_count`	The color indicating how many cases belong to each level. `character`.
`vertical_limits`	The plotted limits of the response variable. A two-element `numeric` array.
`jitter_observed`	A function dictating how the observed values are jittered.
`jitter_predicted`	A function dictating how the predicted values are jittered.
`seed_value`	The value of the RNG seed, which affects jittering. No seed is set if a value of `NA` is passed. `numeric`.

Index

∗Topic **explore**
create_manifest_explore_univariate, 3

calculate_bins, 2
calculate_rounding_digits, 2
class, *2*, *4*
construct_graph, 3
construct_graph_list_univariate *(*construct_graph*)*, 3
construct_graph_univariate *(*construct_graph*)*, 3
create_manifest_explore_univariate, *3*, 3

histogram_continuous, 5
histogram_discrete, 6

scatter_model_continuous_x_binary_y_logit, 7
scatter_model_discrete_x_binary_y_logit, 8