

Package: Certara.VPCResults (via r-universe)

October 12, 2026

Title Generate Visual Predictive Checks (VPC) Using 'shiny'

Version 3.0.2

Description Utilize the 'shiny' interface to parameterize a Visual Predictive Check (VPC), including selecting from different binning or binless methods and performing stratification, censoring, and prediction correction. Generate the underlying 'tidyvp' and 'ggplot2' code directly from the user interface and download R or Rmd scripts to reproduce the VPCs in R.

Depends R (>= 4.0)

License LGPL-3

URL <https://certara.github.io/R-VPCResults/>

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, xpose, ggnewscale, testthat (>= 3.0.0)

Imports shinyAce, shinymeta, bslib (>= 0.7.0), colourpicker, data.table, dplyr, DT, grDevices, ggplot2, plotly, shiny (>= 1.7.0), shinycssloaders, shinyjs, shinyWidgets, shinyjqui, sortable, tidyvp, htmltools, rlang, scales

Config/testthat/edition 3

NeedsCompilation no

Author James Craig [aut, cre], Mike Talley [aut], Certara USA, Inc [cph, fnd]

Maintainer James Craig <james.craig@certara.com>

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libssl-dev zlib1g-dev

Repository <https://certara-jcraig.r-universe.dev>

Date/Publication 2024-12-02 15:30:02 UTC

RemoteUrl <https://github.com/cran/Certara.VPCResults>

RemoteRef HEAD

RemoteSha 9c1ac1585be6e6219d2d1d142cbc064bfa4cc3b8

Contents

theme_certara	2
vpcResultsUI	3
write_code	4
Index	5

theme_certara	<i>A ggplot2 theme for Certara.</i>
---------------	-------------------------------------

Description

A ggplot2 theme for Certara.

Usage

```
theme_certara(  
  base_size = 11,  
  base_family = "",  
  base_line_size = base_size/22,  
  base_rect_size = base_size/22,  
  grid = c("none", "horizontal", "both")  
)
```

Arguments

- base_size base font size, given in pts.
- base_family base font family
- base_line_size base size for line elements
- base_rect_size base size for rect elements
- grid Which grid lines should appear? Horizontal only, both horizontal and vertical, or none (default). [continuous_scale\(\)](#).

Details

There are 3 variants of the theme: no grid `theme_certara()`, full grid `theme_certara(grid = "both")`, and horizontal grid lines only `theme_certara(grid = "horizontal")`.

Value

An object of class `theme()`.

vpcResultsUI*Shiny GUI to parameterize Visual Predictive Check (VPC)*

Description

Use tidyvpc package to parameterize VPC from Shiny GUI and customize plot using ggplot2. Code generation functionality allows you to reproduce tidyvpc object and ggplot2 plot in your local R session via R script saved to your working directory.

Usage

```
vpcResultsUI(  
  observed,  
  simulated,  
  ObsName = NULL,  
  vpc.type = c("continuous", "categorical"),  
  tagged = NULL,  
  settings = NULL,  
  ...  
)
```

Arguments

observed	Observed input data.
simulated	Simulated input data.
ObsName	Optional character value specifying the name of the observed dependent variable (e.g., ObsName).
vpc.type	Character value specifying type of VPC.
tagged	A list of tagged VPC's or model diagnostics created from previous Certara.VPC or Certara.ModelResults session.
settings	A list of plot customization settings used in previous Certara.VPC or Certara.ModelResults session.
...	Arguments for initiation from Pirana.

Value

If interactive(), returns a list of tagged diagnostics from the Shiny application, otherwise returns TRUE.

Examples

```
if (interactive()) {  
  vpcResultsUI(tidyvpc::obs_data, tidyvpc::sim_data)  
}
```

write_code	<i>Write code to R script from tagged diagnostics</i>
------------	---

Description

Use this function to write code to R script from diagnostics tagged in Certara's VPC Results Shiny Application.

Usage

```
write_code(tagged, file)
```

Arguments

tagged	List of tagged objects from returned from <code>VPCResultsUI()</code> .
file	Character specifying path of output file. If missing, it will be saved as <code>code.R</code> in working directory.

Value

Returns NULL after writing to file.

Examples

```
if (interactive()) {  
  tagged_diagnostics <-  
    vpcResultsUI(tidyvpc::obs_data, tidyvpc::sim_data)  
  
  write_code(tagged_diagnostics, "tagged_vpc.R")  
}
```

Index

`continuous_scale`, [2](#)

`theme`, [2](#)

`theme_certara`, [2](#)

`vpcResultsUI`, [3](#)

`write_code`, [4](#)