Package 'vascr'

February 6, 2026

Type Package

Title Process Biological Impedance Sensing Data

Version 0.1.4

Description Process complex impedance sensing datasets, including those generated by ECIS, xCELLigence and cellZscope instruments. Data can be imported to a standardised tidy format and then plotted. Support for conducting and plotting the outputs of ANOVA (with appropriate tests of statistical assumptions) and cross-correlation analysis. For data processed using this package see Hucklesby et al. (2020) <doi:10.3390/bios11050159>.

License GPL (>= 3)

URL `https://github.com/JamesHucklesby/vascr`

BugReports `https://github.com/JamesHucklesby/vascr/issues`

Depends R (>= 3.6)

Imports cli, dplyr, doFuture, foreach, furrr, ggnewscale, ggplot2, ggpubr, progressr, ggtext, glue, memoise, multcomp, patchwork, rlang, rstatix, stringr, tidyr, utils, ggrepel, nlme

Suggests bslib, DBI, DT, data.table, future, odbc, purrr, readr, readxl, shiny, shinyjs, testthat (>= 3.0.0), vdiffr, writexl, testthat, diffviewer

Config/testthat/edition 3

Encoding UTF-8

LazyData TRUE

RoxygenNote 7.3.3

NeedsCompilation no

Author James JW Hucklesby [cre, aut, cph] (ORCID: <`https://orcid.org/0000-0003-1591-6823`>),
Charles Unsworth [aut] (ORCID: <`https://orcid.org/0000-0002-9153-5232`>),
E Scott Graham [aut] (ORCID: <`https://orcid.org/0000-0002-2925-9126`>),
Catherine E angel [aut] (ORCID: <`https://orcid.org/0000-0003-1737-8539`>)

Maintainer James JW Hucklesby <`james.hucklesby@auckland.ac.nz`>

Repository CRAN

Date/Publication 2026-02-06 19:50:02 UTC

Contents

`ecis_import` *Import all ECIS values, a child of ecis_import_raw and ecis_import_model*

Description

Import all ECIS values, a child of ecis_import_raw and ecis_import_model

Usage

```
ecis_import(raw = NULL, modeled = NULL, experimentname = NULL)
```

Arguments

`raw`	A raw ABP file to import
`modeled`	A modeled APB file for import
`experimentname`	Name of the experiment to be built into the dataset

Value

A data frame containing all the data APB generated from an experiment

Examples

```
raw = system.file('extdata/instruments/ecis_TimeResample.abp', package = 'vascr')
modeled = system.file('extdata/instruments/ecis_TimeResample_RbA.csv', package = 'vascr')
experimentname = "TEST"

#Then run the import

data = ecis_import(raw ,modeled,experimentname)
#head(data)
```

`growth.df` *Vascr growth data*

Description

A dataset containing the growth curves of hCMEC/D3 cell lines seeded at various densities. The variables are as follows:

Usage

```
growth.df
```

Format

A tibble with 346370 rows and 9 variables:

Time The time at which the measurement was taken (hours)

Unit The unit the measurement was taken in

Well The well in which the measurement was taken

Value The value of the measurement taken

Frequency The frequency at which data was collected

Experiment Name of the experiment

Instrument The instrument data was collected on

SampleID The numerical ID of the sample

Sample The name of the treatment applied to the dataset

Source

Hucklesby 2020

`vascr_add_vline` *Add a vertical line to a vascr line plot*

Description

Add a vertical line to a vascr line plot

Usage

```
vascr_add_vline(plot, times.df)
```

Arguments

`plot`	The vascr plot to receive a vertical line (or lines)
`times.df`	A tibble containing "time", "color" and "label" columns to specify the addition of lines

Value

A labeled vascr plot

Examples

```
plot1_data = growth.df %>% vascr_subset(unit = "R", frequency = "4000")
plot1 = plot1_data %>% vascr_summarise("summary") %>% vascr_plot_line()

times.df = tribble(~time, ~label, ~colour, 100, "Test Point", "orange")
vascr_add_vline(plot1, times.df)

times.df = tribble(~time, ~label, 100, "ZTest Point", 150, "Test Point 2")
vascr_add_vline(plot1, times.df)
```

`vascr_apply_map` *Apply a map to a vascr dataset*

Description

Apply a map to a vascr dataset

Usage

```
vascr_apply_map(data.df, map)
```

Arguments

`data.df`	the dataset to apply to
`map`	the dataset to apply

Value

a named vascr dataset

Examples

```
lookup = system.file('extdata/instruments/eciskey.csv', package = 'vascr')
vascr_apply_map(data.df = growth.df, map = lookup)

vascr_apply_map(growth.df %>% vascr_subset(well = c("A1")), lookup)
```

`vascr_combine` *Combine ECIS data frames end to end*

Description

This function will combine ECIS data sets end to end. Preferential to use over a simple rbind command as it runs additional checks to ensure that data points are correctly generated

Usage

```
vascr_combine(..., resample = FALSE)
```

Arguments

`...`	List of data frames to be combined
`resample`	Automatically try and re sample the data set. Default is FALSE

Value

A single data frame containing all the data imported, automatically incremented by experiment

Examples

```
#Make three fake experiments worth of data
experiment1.df = vascr_subset(growth.df, experiment = "1")
experiment2.df = vascr_subset(growth.df, experiment = "2")
experiment3.df = vascr_subset(growth.df, experiment = "3")

data = vascr_combine(experiment1.df, experiment2.df, experiment3.df)
head(data)
```

`vascr_dunnett` *Run ANOVA and Dunnett's comparisons on a vascr dataset*

Description

Run ANOVA and Dunnett's comparisons on a vascr dataset

Usage

```
vascr_dunnett(data.df, unit, frequency, time, reference)
```

Arguments

`data.df`	A vascr dataset
`unit`	The unit to plot
`frequency`	The frequency to plot
`time`	The time to plot
`reference`	Reference sample to compare against. If all comparisons are needed use vascr_anova

Value

A table with the results of the Dunnett's test

Examples

```
vascr_dunnett(data.df = growth.df, unit = "R", frequency = 4000, time = 100, reference = 6)
vascr_dunnett(growth.df, "R", 4000, time = list(50, 100), 6)
```

`vascr_edit_name` *Edit a sample name in a vascr dataframe*

Description

Edit a sample name in a vascr dataframe

Usage

```
vascr_edit_name(data.df, to_remove, to_add = "")
```

Arguments

`data.df`	The data set to edit
`to_remove`	The sample to remove
`to_add`	The sample to replace with

Value

An edited vascr dataset

Examples

```
vascr_edit_name(growth.df,"HCMEC D3", "HCMEC/D3")
```

`vascr_edit_sample` *Rename a sample in a vascr dataset*

Description

Renames samples in a vascr dataset, either replacing the whole sample or parts of the string.

Usage

```
vascr_edit_sample(data.df, change_list, partial = TRUE, escape = TRUE)
```

Arguments

`data.df`	Vascr dataset to update
`change_list`	List of vectors containing pairs of search and replacement terms to replace
`partial`	TRUE or FALSE, defines if partial matches should be changed
`escape`	TRUE or FALSE, whether to escape special characters passed into the function

Value

An updated vascr data frame

Examples

```
to_rename = growth.df %>% vascr_subset(sample = c("0 cells","20,000 cells", "10,000 cells"))

to_rename$Sample %>% unique()

renamed = vascr_edit_sample(to_rename, change_list = list(c("0_cells", "Cell Free")))
print(renamed$Sample %>% unique())
```

`vascr_exclude` *Exclude samples from a vascr dataset*

Description

Exclude samples from a vascr dataset

Usage

```
vascr_exclude(data.df, well = NULL, experiment = NULL, sampleid = NULL)
```

Arguments

`data.df`	the vascr data set to exclude things from
`well`	wells to exclude
`experiment`	experiments to exclude
`sampleid`	sampleID (or vector or sampleIDs) to exclude from analysis

Value

A smaller vascr dataset

Examples

```
vascr_exclude(growth.df, c("A01", "E01"))
vascr_exclude(growth.df, sampleid = 1)
```

`vascr_explode` *Separate names in a vascr data frame*

Description

Separate names in a vascr data frame

Usage

```
vascr_explode(data.df)
```

Arguments

`data.df`	the dataset to separate

Value

a separated vascr dataset, with additional columns for each variable

Examples

```
vascr_explode(growth.df)
```

`vascr_export_prism` *Export a vascr dataframe*

Description

Export a vascr dataframe

Usage

```
vascr_export_prism(
  data.df,
  filepath = tempfile("test_export", fileext = ".xlsx"),
  level = "experiments"
)
```

Arguments

`data.df`	a vascr dataset to export
`filepath`	Path to save the dataframe in
`level`	Level of replication to export, defaults to experiments

Value

A dataframe in prism format, or writes to file if filepath specified

Examples

```
filepath = tempfile("test_export", fileext = ".xlsx")
to_export = growth.df %>% vascr_subset(unit = c("R", "Rb"), frequency = c(0,4000))
vascr_export_prism(to_export, filepath)

vascr_export_prism(to_export, filepath, level = "wells")
```

`vascr_find` *Find vascr variables*

Description

These functions are utility functions that will detect if arguments are invalid, and attempt to repair them. Each type of variable has rules related to what values are possible in a valid vascr dataset.

Usage

```
vascr_find(data.df = vascr::growth.df, paramater, value = NA)
```

Arguments

`data.df`	The vascr dataset to reference, will default to the growth.df dataset if not specified
`paramater`	The parameter to search. Options are "Time", "Unit", "Well", "Frequency", "Sample", "Experiment", "SampleID" or "resampled"
`value`	the value to look up

Value

The valid vascr dataset.

Examples

```
vascr_find(growth.df, "Time")
vascr_find(growth.df, "Time", 66.97)
vascr_find(growth.df, "Time", NULL)

vascr_find(growth.df, "Unit")
vascr_find(growth.df, "Unit", "Rb")
vascr_find(growth.df, "Unit", NULL)

vascr_find(growth.df, "Well")
vascr_find(growth.df, "Well", "A1")
```

```
vascr_find(growth.df, "Sample")
vascr_find(growth.df, "Sample", "5000 cells")

vascr_find(growth.df, "Frequency")
vascr_find(growth.df, "Frequency", 4000)

vascr_find(growth.df, "Experiment")
vascr_find(growth.df, "Experiment", 1)

vascr_find(growth.df, "SampleID")
vascr_find(growth.df, "SampleID", 5)

vascr_find(growth.df, "resampled")

vascr_find(growth.df, "all")
```

`vascr_find_level` *Check the level of a vascr data frame*

Description

Check the level of a vascr data frame

Usage

```
vascr_find_level(data)
```

Arguments

`data`	The data frame to analyse

Value

The level of the dataset analysed

Examples

```
vascr_find_level(growth.df)
vascr_find_level(vascr_summarise(growth.df %>% vascr_subset(unit= "Rb"), level = "experiments"))
vascr_find_level(vascr_summarise(growth.df %>% vascr_subset(unit= "Rb"), level = "summary"))
```

`vascr_find_metadata` *Print out the characteristics of the vascr data frame*

Description

Print out the characteristics of the vascr data frame

Usage

```
vascr_find_metadata(data.df)
```

Arguments

`data.df`	the vascr data frame to interrogate

Value

prints out the parameters of the dataframe in question

Examples

```
vascr_find_metadata(growth.df)
```

`vascr_implode` *Implode individual samples from a vascr dataset*

Description

Implode individual samples from a vascr dataset

Usage

```
vascr_implode(data.df, cols = NULL)
```

Arguments

`data.df`	A vascr dataset to be imploded
`cols`	The columns to implode

Value

A vascr dataset with individual wells imploded

Examples

```
vascr_implode(growth.df)
```

`vascr_import` *Import an impedance datafile to vascr*

Description

Import an impedance datafile to vascr

Usage

```
vascr_import(
  instrument = NULL,
  raw = NULL,
  modeled = NULL,
  experiment = NULL,
  shear = FALSE
)
```

Arguments

`instrument`	Instrument to import from, either ECIS, xCELLigence or cellZscope
`raw`	Path to raw data file
`modeled`	Path to modeled data file from manufacturer's software
`experiment`	Name for the experiment being imported
`shear`	True or False, is a shear plate used, as these have a different electrode layout. Defaults to False.

Value

A vascr dataset for subsequent analysis

Examples

```
# ECIS
raw = system.file('extdata/instruments/ecis_TimeResample.abp', package = 'vascr')
modeled = system.file('extdata/instruments/ecis_TimeResample_RbA.csv', package = 'vascr')
vascr_import("ECIS", raw, modeled, "ECIS_Data")

# xCELLigence
raw = system.file('extdata/instruments/xcell.plt', package = 'vascr')
# No modeling for this system
vascr_import("xCELLigence", raw, experiment = "xCELLigence")

# cellZscope
model = system.file("extdata/instruments/zscopemodel.txt", package = "vascr")
raw = system.file("extdata/instruments/zscoperaw.txt", package = "vascr")
vascr_import("cellzscope", raw, model, "cellZscope")
```

```
#' # SciоSpec
raw = system.file("extdata/instruments/ScioSpec", package = "vascr")
vascr_import("sciospec", raw, model, "ScioSpec")
```

---

vascr_load *Load a vascr dataset*

Description

Load a vascr dataset

Usage

```
vascr_load(path)
```

Arguments

`path`	the path to a .vascr file containing the saved dataset

Value

A vascr dataset

Examples

```
path = system.file("extdata/test.vascr", package = "vascr")
vascr_load(path)
```

---

vascr_normalise *Normalize ECIS data to a single time point*

Description

This function normalises each unique experiment/well combination to it's value at the specified time. Contains options to do this either by division or subtraction. Can be run twice on the same dataset if both operations are desired.

Usage

```
vascr_normalise(data.df, normtime, divide = FALSE)
```

Arguments

`data.df`	Standard vascr data frame
`normtime`	Time to normalize the data to
`divide`	If set to true, data will be normalized via a division. If set to false (default) data will be normalized by subtraction. Default is subtraction

Value

A standard ECIS dataset with each value normalized to the selected point.

Examples

```
data = vascr_normalise(growth.df, normtime = 100)
head(data)
```

`vascr_plot_anova` *Make a display with all the ANOVA analysis pre-conducted*

Description

Make a display with all the ANOVA analysis pre-conducted

Usage

```
vascr_plot_anova(
  data.df,
  unit,
  frequency,
  time,
  reference = NULL,
  separate = "x",
  rotate = 45
)
```

Arguments

`data.df`	vascr dataset to plot
`unit`	unit to plot
`frequency`	frequency to plot
`time`	timepoint to plot at
`reference`	Sample to reference post-hoc analysis to
`separate`	Value to use when separating comparasons in the output (default x)
`rotate`	degrees of rotation used for labeling the X axis

Value

A matrix of different ANOVA tests

Examples

```
# Run, comparing only to a reference
vascr_plot_anova(data.df = small_growth.df, unit = "R", frequency = 4000, time = 100,
          reference = "5,000_cells + HCMEC D3_line")
```

`vascr_plot_grid` *Plot out each replicate well in a grid, with QC overlays*

Description

Plot out each replicate well in a grid, with QC overlays

Usage

```
vascr_plot_grid(data.df, threshold = 0.2)
```

Arguments

`data.df`	a vascr formatted data frame of single values
`threshold`	threshold at which a data point is determined to be an outlier

Value

A plot to be used for QC

Examples

```
grid.df = growth.df %>% vascr_subset(unit = "R", frequency = "4000", experiment  = 1)
vascr_plot_grid(grid.df)
```

`vascr_plot_line` *Plot a line graph with vascr*

Description

Plot a line graph with vascr

Usage

```
vascr_plot_line(
  data.df,
  errorbars = Inf,
  alpha = 0.3,
  text_labels = TRUE,
  facet_expt = TRUE,
  show_linetypes = TRUE,
  explanatory = FALSE
)
```

Arguments

`data.df`	The vascr data set to plot
`errorbars`	Type of error bars, Inf for ribbons, 0 for no errors and any integer to plot every nth line
`alpha`	Transparency of the error ribbon plotted
`text_labels`	Show or hide well labels
`facet_expt`	Facet out different experiments, defaults to TRUE
`show_linetypes`	Include the line type for each experiment in the key. Default TRUE
`explanatory`	Whether to include commentary about unit meaning on y axis (default TRUE)

Value

a ggplot containing the data

Examples

```
vascr_plot_line(data.df = growth.df %>% vascr_subset(unit = "Rb"))
vascr_plot_line(growth.df %>% vascr_subset(unit = "Rb") %>% vascr_summarise(level = "experiments"))
data.df = growth.df %>% vascr_subset(unit = "Rb") %>% vascr_summarise(level = "summary")
vascr_plot_line(data.df, text_labels = FALSE)

growth.df %>% vascr_subset(unit = "R", frequency = 4000)  %>%
vascr_plot_line(facet = FALSE)

growth.df %>% vascr_subset(unit = "R", frequency = 4000)  %>%
vascr_plot_line(explanatory = TRUE)
```

`vascr_plot_line_dunnett` *Create a line plot with Dunnett's statistics*

Description

Create a line plot with Dunnett's statistics

Usage

```
vascr_plot_line_dunnett(
  data.df,
  unit = "R",
  frequency = 4000,
  time = 100,
  reference = "0_cells + HCMEC D3_Line",
  normtime = NULL
)
```

Arguments

`data.df`	A vascr dataset
`unit`	Unit to calculate
`frequency`	Frequency to calculate from
`time`	Time to calculate
`reference`	Sample to reference testing against
`normtime`	Time to normalise the line plot to, note this does not affect underlying statistical test

Value

A line plot, annotated with the P-values determined by Dunnett's test

Examples

```
vascr_plot_line_dunnett(small_growth.df, unit = "R", frequency = 4000, time = 25,
    reference = "0_cells + HCMEC D3_Line")
vascr_plot_line_dunnett(small_growth.df, unit = "R", frequency = 4000, time = list(25,100),
    reference = "0_cells + HCMEC D3_Line")
vascr_plot_line_dunnett(small_growth.df, unit = "R", frequency = 4000, time = 180,
    reference = "20,000_cells + HCMEC D3_Line")
```

`vascr_plot_resample` *Plot the data re sampling process*

Description

Plot the data re sampling process

Usage

```
vascr_plot_resample(
  data.df,
  unit = "R",
  frequency = "4000",
  well = "A01",
  newn = 20,
  plot = TRUE,
  rug = TRUE,
  points = FALSE
)
```

Arguments

`data.df`	Dataset to analyse
`unit`	Unit to use, defaults to R
`frequency`	Frequency to use, defaults to 4000
`well`	Well to use, defaults to A01 (or first well in plate)
`newn`	New number of timepoints to compare to current
`plot`	Return a ggplot or the underlying data. Defaults to TRUE, returning the plot.
`rug`	Show rug lot, defaults true
`points`	Show points, defaults to false

Value

A plot showing how well the resampled data conforms to the actual data set

Examples

```
vascr_plot_resample(growth.df)
vascr_plot_resample(growth.df, plot = FALSE)
```

`vascr_resample_time` *Resample a vascr dataset*

Description

Impedance sensing data is often not collected simultaneously, which creates issues summarising and plotting the data. This function interpolates these data to allow these downstream functions to happen.

Usage

```
vascr_resample_time(
  data.df,
  npoints = vascr_find_count_timepoints(data.df),
  t_start = min(data.df$Time),
  t_end = max(data.df$Time),
  rate = NULL,
  force_timepoint = NULL,
  include_disc = TRUE
)
```

Arguments

`data.df`	The vascr dataset to resample
`npoints`	Manually specificity the number of points to resample at, default is the same frequency as the input dataset
`t_start`	Time to start at
`t_end`	Time to end at
`rate`	Time between timepoints
`force_timepoint`	Force a specific timepoint to be part of the resample
`include_disc`	Add an additional data point either side of a discrepancy. Defaults TRUE

Value

An interpolated vascr dataset

Examples

```
# Automatically re sample, mimicking the input data as closely as possible
vascr_resample_time(growth.df)

# Fully controlled resample with advanced options
vascr_resample_time(growth.df, t_start = 5, t_end = 20, rate = 5, force = c(1,2,3))
```

`vascr_samples` *List out the samples currently in a vascr data set*

Description

List out the samples currently in a vascr data set

Usage

```
vascr_samples(data.df)
```

Arguments

`data.df`	The vascr data set to analyse

Value

A printout of the samples, Sample ID's and experiments where they occur

Examples

```
vascr_samples(growth.df)
```

`vascr_save` *Save a vascr dataset*

Description

Save a vascr dataset

Usage

```
vascr_save(data.df, path)
```

Arguments

`data.df`	The vascr dataset to save
`path`	The path to save the file to

Value

A .vascr file containing a vascr dataset

Examples

```
path = tempfile()
vascr_save(growth.df, path = path)
```

`vascr_shiny` *Launch the vascr UI*

Description

Launch the vascr UI

Usage

```
vascr_shiny(data.df)
```

Arguments

`data.df`	Data to preload into shiny app

Value

A shinyApp to work with vascr data

Examples

```
if(interactive()){
  vascr_shiny()
}
```

`vascr_subset` *Subset a vascr data set based on a number of factors*

Description

Subset a vascr data set based on a number of factors

Usage

```
vascr_subset(
  data.df,
  time = NULL,
  unit = NULL,
  well = NULL,
  frequency = NULL,
  experiment = NULL,
  instrument = NULL,
  sampleid = NULL,
  sample = NULL,
  subsample = NULL,
  remove_na_value = TRUE,
  remove_excluded = TRUE
)
```

Arguments

`data.df`	vascr data set to subset
`time`	Specified times. Individual values in a list will be subset out. If vectors are present in the list, values between the two most extreme values will be returned.
`unit`	Units to subset. These are checked for integrity against possible units and the dataset itself
`well`	Wells to select
`frequency`	Frequencies to include in the data set.
`experiment`	Experiments to include in the data set. Can be addressed either by name, or by the numerical order that they were loaded into vascr_combine in
`instrument`	Which instruments to include values from
`sampleid`	List of ID's to be used. Sample names will be re-ordered accordingly for display.
`sample`	Sample to subset
`subsample`	Frequency values should be sub-sampled to
`remove_na_value`	Should NA values be removed (default true)
`remove_excluded`	Should excluded values be removed (default true)

Value

The subset dataset, based on the values selected

Examples

```
vascr_subset(growth.df)
vascr_subset(growth.df, time = 40)
vascr_subset(growth.df, time = NULL)

vascr_subset(growth.df, unit = "Rb")
vascr_subset(growth.df, unit = "R")
vascr_subset(growth.df, well = "A1")

vascr_subset(growth.df, time = c(5,20))
```

`vascr_summarise` *Summarise a vascr data set down to a particular level*

Description

Summarise a vascr data set down to a particular level

Usage

```
vascr_summarise(data.df, level = "wells")
```

Arguments

`data.df`	Data set to summarize
`level`	Level to summarise to, either "median_deviation", "ANOVA", "summary", "experiment" or "wells"

Value

The summarized data set

Examples

```
rbgrowth.df = vascr_subset(growth.df, unit = "Rb")

vascr_summarise(rbgrowth.df, level = "summary")
vascr_summarise(rbgrowth.df, level = "experiment")
vascr_summarise(rbgrowth.df, level = "wells")
```

`vascr_zero_time` *Set a particular time point as 0 in a vascr dataset*

Description

Allows the user to change the time designated as zero to allow clearer plotting of treatments

Usage

```
vascr_zero_time(data.df, time = 0)
```

Arguments

`data.df`	A vascr dataset
`time`	The time to be set to 0

Value

An adjusted vascr dataset

Examples

```
vascr_zero_time(growth.df, 50)
```

`%>%` *re-export magrittr pipe operator*

Description

re-export magrittr pipe operator

re-export tribble

re-export glue

Value

Various outputs, please see documentation for source package

description Used to easily make pipe operator available

A tibble, made available for easy data entry manually

A string with glue substitution completed

Index