

Package ‘mlpwr’

October 14, 2022

Title A Power Analysis Toolbox to Find Cost-Efficient Study Designs

Version 1.0.0

Description We implement a surrogate modeling algorithm to guide simulation-based sample size planning. The method is described in detail in a recent preprint (Zimmer & Debelak (2022) <[doi:10.31234/osf.io/tnhb2](https://doi.org/10.31234/osf.io/tnhb2)>).

It supports multiple study design parameters and optimization with respect to a cost function. It can find optimal designs that correspond to a desired statistical power or that fulfill a cost constraint.

License GPL (>= 3)

URL <https://github.com/flxzimmer/mlpwr>

BugReports <https://github.com/flxzimmer/mlpwr/issues>

Imports utils, stats, DiceKriging, digest, ggplot2, randtoolbox, rlist, WeightSVM, rgenoud

Suggests knitr, lme4, lmerTest, mirt, pwr, rmarkdown, simr, sn, tidyr

Encoding UTF-8

RoxygenNote 7.2.1

VignetteBuilder knitr

NeedsCompilation no

Author Felix Zimmer [aut, cre] (<<https://orcid.org/0000-0002-8127-0007>>),
Rudolf Debelak [aut] (<<https://orcid.org/0000-0001-8900-2106>>)

Maintainer Felix Zimmer <felix.zimmer@mail.de>

Repository CRAN

Date/Publication 2022-10-14 13:30:02 UTC

R topics documented:

example.simfun	2
find.design	2
plot.designresult	4
print.summary.designresult	5
summary.designresult	6

Index

7

example.simfun	<i>Example simulation functions</i>
----------------	-------------------------------------

Description

These simulation functions can be loaded as examples or templates. They correspond to the functions in the 'simulation_functions' vignette. The vignette contains the code and a description of the simfuns.

Usage

```
example.simfun(name)
```

Arguments

name	Name of the simulation function as a character. Currently available are: 'ttest', 'anova', 'glm2', 'irt1', 'irt2', 'l
------	---

Value

The returned object is the simulation function.

Examples

```
simfun = example.simfun('ttest')
simfun(400)
```

find.design	<i>Find optimal study designs</i>
-------------	-----------------------------------

Description

Perform a surrogate modeling approach to search for optimal study design parameters. For further guidance on how to use the package and the find.design function specifically, see the [Readme.md file](#).

Usage

```
find.design(
  simfun,
  boundaries,
  power = NULL,
  evaluations = 4000,
  ci = NULL,
  ci_perc = 0.95,
```

```

time = NULL,
costfun = NULL,
cost = NULL,
surrogate = NULL,
n.startsets = 4,
init.perc = 0.2,
setsize = 100,
continue = NULL,
dat = NULL,
silent = FALSE,
autosave_dir = NULL,
control = list()
)

```

Arguments

simfun	function to generate hypothesis test results with. Takes design parameters as input and outputs a logical (result of the hypothesis test). The function can take the designs through one argument as a vector or through multiple arguments. For example, function(x) where x is later used with x=c(n,k) for two design parameters n and k is valid. Also valid is a definition using function(n,k).
boundaries	list containing lower and upper bounds of the design space. The list should consist of named vectors, each containing the upper and lower bound for the respective design parameter dimensions. For one design parameter dimension, can also be a vector containing the upper and lower bounds.
power	numeric; desired statistical power
evaluations	integer; number of dgf evaluations to be performed before termination
ci	numeric; desired width of the confidence interval at the predicted value on termination.
ci_perc	numeric; specifying the desired confidence interval, e.g. 95% or 99%.
time	integer; seconds until termination
costfun	function that takes a vector of design parameters as input and outputs a cost, e.g. monetary costs. Necessary for simfuns with multiple input dimensions.
cost	numeric; cost threshold. Design parameter set with highest power is searched among sets that fulfill this cost threshold.
surrogate	character; which surrogate model should be used. The default is 'logreg' for one design parameter and 'gpr' for multiple design parameters. The current options are: 'gpr', 'svr', 'logreg', 'reg' for one-dimensional designs and 'gpr' and 'svr' for multi-dimensional designs.
n.startsets	integer; number of startsets used per dimension of simfun
init.perc	numeric; percentage of evaluations used for the initialization phase
setsize	The number of draws from the simfun in each iteration
continue	Object of class designresult as created by the find.design function. Will be used to continue the search, using all collected simulation results so far.
dat	list of data from a previous design result.

silent logical; suppresses output during the search.
autosave_dir character; file location for saving the dat object after each update.
control list specifying arguments passed to the surrogate models. For example, `list(covtype='gauss')` can be used with the `gpr` surrogate to use a different covariance structure than the default.

Value

function returns an object of class `designresult`

Examples

```

#Load a simulation function
simfun = example.simfun('ttest')
# Perform the search
ds = find.design(simfun = simfun, boundaries = c(100,300), power = .95)
# Output the results
summary(ds)
# Plot results
plot(ds)

# Two-dimensional simulation function:
simfun = example.simfun('anova')
# Perform the search
res = find.design(simfun = simfun,
  costfun = function(n,n.groups) 5*n+20*n.groups,
  boundaries = list(n = c(10, 150), n.groups = c(5, 30)),
  power = .95)
# Output the results
summary(ds)
# Plot results
plot(ds)

```

`plot.designresult` *Plot search result*

Description

Plot a one- or two-dimensional graph of the result.

Usage

```

## S3 method for class 'designresult'
plot(
  x,
  design = NULL,
  adderrorbars = NULL,

```

```

    addribbon = NULL,
    trim = TRUE,
    type = "heat",
    ...
)

```

Arguments

x	Object of class designresult as created by the find.design function.
design	Specify a design as a list. Can be used to make a 1D plot for a two-dimensional simfun. Set NA for the dimension that should be plotted and set a value for all others. For example: design=list(n=NA,k=9)
adderrorbars	logical. Plots errorbars in the 1D plot if TRUE. Default is FALSE (also if specified as NULL).
addribbon	logical. Adds ribbon in the 1D plot if TRUE. Default is TRUE (also if specified as NULL).
trim	logical. Option to trim the plotting area for the 2D line plot. The trimmed area is the area where the line is plotted. Default is TRUE.
type	character indicating the type of the 2D plot. Can be 'heat'(default) or 'line'.
...	additional arguments to be passed.

Value

A ggplot object

Examples

```

#Load a simulation function
simfun = example.simfun('ttest')
# Perform the search
ds = find.design(simfun = simfun, boundaries = c(100,300), power = .95)
# Plot results
plot(ds)

```

```
print.summary.designresult
```

Print Summary of the search result

Description

Print Summary of the search result

Usage

```

## S3 method for class 'summary.designresult'
print(x, ...)

```

Arguments

x Object of class designresult as created by the find.design function
... additional arguments to be passed

Value

An object of class summary.designresult

Examples

```
#Load a simulation function
simfun = example.simfun('ttest')
# Perform the search
ds = find.design(simfun = simfun, boundaries = c(100,300), power = .95)
# Output the results
summary(ds)
```

summary.designresult *Summary of the search result*

Description

Produce summary statistics of the results and the algorithm run.

Usage

```
## S3 method for class 'designresult'
summary(object, ...)
```

Arguments

object Object of class designresult as created by the find.design function
... additional arguments to be passed

Value

An object of class summary.designresult

Examples

```
#Load a simulation function
simfun = example.simfun('ttest')
# Perform the search
ds = find.design(simfun = simfun, boundaries = c(100,300), power = .95)
# Output the results
summary(ds)
```

Index

`example.simfun`, [2](#)

`find.design`, [2](#)

`plot.designresult`, [4](#)

`print.summary.designresult`, [5](#)

`summary.designresult`, [6](#)