

Package ‘OWEA’

November 3, 2018

Type Package

Title Optimal Weight Exchange Algorithm for Optimal Designs for Three Models

Version 0.1.1

Maintainer Shuai Hao <shuaih0303@gmail.com>

Description An implementation of optimal weight exchange algorithm Yang(2013) <doi:10.1080/01621459.2013.806268> for three models. They are Crossover model with subject dropout, crossover model with proportional first order residual effects and interference model. You can use it to find either A-opt or D-opt approximate designs. Exact designs can be automatically rounded from approximate designs and relative efficiency is provided as well.

License GPL-3

Encoding UTF-8

LazyData true

Suggests knitr, rmarkdown

Imports gtools (>= 3.8.1), MASS, shiny (>= 1.1.0)

RoxygenNote 6.1.0

NeedsCompilation no

Author Shuai Hao [aut, cre],
Min Yang [aut],
Wei Zheng [aut]

Repository CRAN

Date/Publication 2018-11-03 15:30:03 UTC

R topics documented:

design	2
design_app	3
eff	4
effLB	5
infor_design	5
OWEA	6
summary.dropout	7

verify_equivalence
 result of general equivalence theorem, the last entry is the value of directional derivative

time computing time for approximate design

See Also

[eff](#), [effLB](#), [summary](#)

Examples

```
# NOTE: max_iter is usually set to 40.
# Here max_iter = 5 is for demenstration only.
# crossover dropout model
## D-optimal

example1 <- design('dropout',10,0,3,3,drop=c(0,0,0.5), max_iter = 5)
summary(example1)
eff(example1) # efficiency from rounding
effLB(example1) # obtain lower bound of efficiency

## A-optimal
design('dropout',10,1,3,3,drop=c(0,0,0.5), max_iter = 5)

# proportional model
## D-optimal
design('proportional',10,0,3,3, sigma = diag(1,3),tau = matrix(sqrt(1+3),
  nrow=3, ncol=1),lambda = 0.2, max_iter = 5)

## A-optimal
design('proportional',10,1,3,3, sigma = diag(1,3), tau = matrix(sqrt(1+3),
  nrow=3, ncol=1),lambda = 0.2, max_iter = 5)

# interference model
## D-optimal
design('interference',10,0,3,3, sigma = diag(1,3), max_iter = 5)

## A-optimal
design('interference',10,1,3,3, sigma = diag(1,3), max_iter = 5)
```

design_app

Shiny App for design function

Description

A function to launch graphical interface to design function.

Usage

```
design_app()
```

Examples

```
## Not run:
design_app() # launching the app.
## End(Not run)
```

eff	<i>Efficiency generic function</i>
-----	------------------------------------

Description

A generic function that returns the efficiency for either exact designs to approximate designs or exact design to a given design

Usage

```
eff(exact_design, ex = NULL)

## Default S3 method:
eff(exact_design, ex = NULL)

## S3 method for class 'dropout'
eff(exact_design, ex = NULL)

## S3 method for class 'proportional'
eff(exact_design, ex = NULL)

## S3 method for class 'interference'
eff(exact_design, ex = NULL)
```

Arguments

`exact_design` A S3 object returned by design function.
`ex` Matrix. Design to be compared to. Default is NULL.

Value

Numeric. Relative Efficiency.

If `ex` is given return relative efficiency by

$$\Phi_{example} / \Phi_{exact_design};$$

If `ex` is missing

return relative efficiency by

$$\Phi_{approx_design} / \Phi_{exact_design}.$$

See Also

see examples in [design](#).

effLB

Lower Bound Efficiency for Crossover-Dropout Model

Description

The function take S3 object of class 'dropout' as input and return its lower bound of efficiency of exact design.

Usage

```
effLB(exact_design)
```

Arguments

`exact_design` A object of class returned by design function.

Value

A list of relevant numerics.

`optimal` Optimal Criterion

`lower.bound` Lower Bound of the exact design

`optimal.value` The value of objective function at optimal approximate design

See Also

see examples in [design](#).

infor_design

Generic function for information matrix.

Description

Returns a information matrix for a given design

Usage

```
infor_design(design, t, ...)  
  
## Default S3 method:  
infor_design(design, t)  
  
## S3 method for class 'dropout'  
infor_design(design, t, ...)  
  
## S3 method for class 'interference'  
infor_design(design, t, ...)  
  
## S3 method for class 'proportional'  
infor_design(design, t, ...)
```

Arguments

design	Matrix. A design, each row is a design point with weight or repetition on the last entry.
t	Numeric. Number of levels of treatments.
...	Other control parameter to be passed to methods

Value

An information matrix.

OWEA

OWEA: A package for optimal designs by implementing optimal weight exchange algorithm.

Description

The OWEA package provides realizations for three models: crossover with subject dropout, crossover with proportional first order residual, and interference model

Key functions

[design](#), [design_app](#), [eff](#), [effLB](#), [summary](#)

summary.dropout	<i>Summary method for S3 object</i>
-----------------	-------------------------------------

Description

Return summary info for S3 object return by design function.

Usage

```
## S3 method for class 'dropout'  
summary(object, ...)  
  
## S3 method for class 'proportional'  
summary(object, ...)  
  
## S3 method for class 'interference'  
summary(object, ...)
```

Arguments

object	A S3 object of class 'dropout', 'proportional', or 'interference'.
...	other control parameters, but usually not necessary.

Value

A list of key info.

exact_design	exact design and its repetitions
approximate_design	approximate design and its weights
computing_time	computing time for approximate design

See Also

see examples in [design](#).

Index

design, [2](#), [5–7](#)
design_app, [3](#), [6](#)

eff, [3](#), [4](#), [6](#)
effLB, [3](#), [5](#), [6](#)

infor_design, [5](#)

OWEA, [6](#)
OWEA-package (OWEA), [6](#)

summary, [3](#), [6](#)
summary.dropout, [7](#)
summary.interference (summary.dropout),
[7](#)
summary.proportional (summary.dropout),
[7](#)