

Package ‘dfmeta’

March 27, 2018

Type Package

Title Meta-Analysis of Phase I Dose-Finding Early Clinical Trials

Version 1.0.0

Date 2018-03-25

Author Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno U

Maintainer Artemis Toumazi <artemis.toumazi@gmail.com>

Description Meta-analysis approaches for Phase I dose finding early phases clinical trials in order to better suit requirements in terms of maximum tolerated dose (MTD) and maximal dose regimen (MDR). This package has currently three different approaches: (a) an approach proposed by Zohar et al, 2011, <doi:10.1002/sim.4121> (denoted as ZKO), (b) the Variance Weighted pooling analysis (called VarWT) and (c) the Random Effects Model Based (REMB) algorithm, where user can input his/her own model based approach or use the existing random effect logistic regression model (named as glimem) through the ‘dfmeta’ package.

License GPL (>= 3) | file LICENSE

Depends R (>= 3.0.2), lme4, stats4, plyr, data.table

Imports ggplot2 (>= 2.0.0), methods, graphics, grDevices, stats, utils

LazyData true

URL <http://github.com/artemis-toumazi/dfmeta>

BugReports <http://github.com/artemis-toumazi/dfmeta/issues>

RoxygenNote 6.0.1

SystemRequirements C++11

NeedsCompilation no

Repository CRAN

Date/Publication 2018-03-27 16:01:30 UTC

R topics documented:

dfmeta-package	2
doseRecords	3
findFirstLast	4
glimem	5
MA_estimates	7
plot,ZKO,missing-method	9
REMB-class	9
show-methods	10
Toxdata	10
VarWT	11
ZKO	13
ZKO-class	15
Index	16

dfmeta-package	<i>Meta-Analysis of Phase I Dose-Finding Early Phases Clinical Trials.</i>
----------------	--

Description

Meta-analysis approaches for Phase I dose finding early phases clinical trials in order to better suit requirements in terms of maximum tolerated dose (MTD) and maximal dose regimen (MDR). This package has currently three different approaches: (a) an approach proposed by Zohar et al, 2011 (denoted as ZKO), (b) the Variance Weighted pooling analysis (called VarWT) and, (c) the Random Effects Model Based (REMB) algorithm, where user can put his/her own model based approach or use the existing random effect logistic regression model (named as glimem) through 'dfmeta' package.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

Maintainer: Artemis Toumazi <artemis.toumazi@gmail.com>

References

Zohar, S. , Katsahian, S. and O'Quigley, J. (2011), An approach to meta analysis of dose finding studies. *Statist. Med.*, <doi:10.1002/sim.4121>.

Description

This function make a random effects analysis that can be used after in the random effects model based algorithm.

Usage

```
doseRecords(data)
```

Arguments

`data` a data frame containing information for each patient at each simulation and trial.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

References

Zohar, S. , Katsahian, S. and O'Quigley, J. (2011), An approach to meta analysis of dose finding studies. *Statist. Med.*, <doi:10.1002/sim.4121>.

See Also

[MA_estimates](#), [glimem](#)

Examples

```
#####
### Give a toxicity data for each simulation and each trial. ###
#####

### Using a toxicity data for each simulation and each trial including in the dfmeta package.
data("Toxdata")
dim(Toxdata)
colnames(Toxdata)

## NOTE: The toxicity dataset must have the same structure as below. ##
## Check it before running the VarWT function! ##
str(Toxdata)

analyf0 <- Toxdata[order(Toxdata$simulation), ]

#####
```

```
## Taking the records from random effect analysis ##
#####
rec <- doseRecords(analyf0)
```

findFirstLast *The first or the last occurrence of a unique value.*

Description

This function can be used to find the first or the last occurrence of each unique value in a column.

Usage

```
findFirstLast(myDF, findFirst = TRUE)
```

Arguments

myDF a data.frame or a matrix.
findFirst a logical variable indicating the first or the last occurrence.

Details

By default, it finds the first occurrence (i.e. findFirst = TRUE). If instead, we want to find the last, we must set findFirst = FALSE.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

See Also

[VarWT](#)

Examples

```
#####
### Give a toxicity data for each simulation and each trial. ###
#####

### Using a toxicity data for each simulation and each trial including in the dfmeta package.

data("Toxdata")
dim(Toxdata)
colnames(Toxdata)

## NOTE: The toxicity dataset must have the same structure as below. ##
## Check it before running the VarWT function! ##
str(Toxdata)
```

```

analyf0 <- Toxdata[order(Toxdata$simulation), ] # sort the imported data by simulation.

## Using the function findFirstLast to find when starts the first trial. ##
first.trial = findFirstLast(analyf0)$Trial

```

glimem

*The logistic regression mixed effect model.***Description**

This function corresponds to the generalized logistic regression mixed effect model. This model is a model-based method for combining toxicities across the trials and cycles. We model:

$$\log\left(\frac{R_{ij}(x)}{1 - R_{ij}(x)}\right) = \mu_{ij}(x) + Z_i,$$

where Z_i 's are assumed to be i.i.d $\mathcal{N}(0, \sigma^2(x))$. Notice that $\mu_{i1}(x), \mu_{i2}(x), \dots, \mu_{iP}(x) = \bar{\mu}_i$ represents the mean toxicities on the logit scale across the cycles.

Usage

```

glimem(simData, sim0, sim1, family = binomial, link = "logit", nAGQ,
       control = glmerControl(optimizer = "bobyqa"))

```

Arguments

simData	a data frame containing the variables named in the formula (i.e. a toxicity data for each patient at each simulation and trial.
sim0	the simulation starting point; defaults to the minimum number of the simulation in the given data frame.
sim1	the simulation ending point; defaults to the maximum number of the simulation in the given data frame.
family	a distribution family for the response variable; defaults to binomial distribution. See glm and glmer for details.
link	a specification for the model link function. This can be a name/expression, a literal character string or a length-one character vector; defaults to logit link function. See glmer for details.
nAGQ	an integer scalar - the number of points per axis for evaluating the adaptive Gauss-Hermite approximation to the log-likelihood; defaults to 1, corresponding to the Laplace approximation. Values greater than 1 produce greater accuracy in the evaluation of the log-likelihood at the expense of speed.
control	a list (of correct class, resulting from lmerControl() or glmerControl() respectively) containing control parameters, including the nonlinear optimizer to be used and parameters to be passed through to the nonlinear optimizer, see the <code>*lmerControl*</code> documentation for details.

Value

A list is returned, consisting of the generalized logistic regression mixed effect model's results. The output generated by this function contains the following components:

m	a summary of the generalized logistic regression model.
coeff	the random and the fixed effects coefficients for each explanatory variable for each level of each grouping factor.
simData	a data frame including the predicted values and the residuals for the selected simulation.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

See Also

[MA_estimates](#)

Examples

```
## Not run:
#####
### Give a toxicity data for each simulation and each trial. ###
#####

### Using a toxicity data for each simulation and each trial including in the dfmeta package.
data("Toxdata")
dim(Toxdata)
colnames(Toxdata)

## NOTE: The toxicity dataset must have the same structure as below. ##
## Check it before running the VarWT function! ##
str(Toxdata)

analyf0 <- Toxdata[order(Toxdata$simulation), ]

## Using the random effect analysis's function including in the dfmeta package ##
simData <- doseRecords(analyf0)
NewData <- simData$doseRecords

#####
## In the following example the function glimem is using to find the predicted random ##
##### effect center mu only for the simulation 10 #####
#####

mod <- glimem(NewData, 10, 10, family = binomial, link = "logit", nAGQ = 1,
              control = glmerControl(optimizer = "bobyqa"))
mod

## End(Not run)
```

MA_estimates	<i>Executing the random effect model-based algorithm (REMB).</i>
--------------	--

Description

This function computes the meta-analysis estimates using one of the random effect model-based algorithms.

Usage

```
MA_estimates(data, sim0, sim1, family = binomial, link = "logit", nAGQ = 1,
  control = glmerControl(optimizer = "bobyqa"))
```

Arguments

data	a data frame containing the variables named in the formula (i.e. a toxicity data for each patient at each simulation and trial).
sim0	the simulation starting point; defaults to the minimum number of the simulation in the given data frame.
sim1	the simulation ending point; defaults to the maximum number of the simulation in the given data frame.
family	a distribution family for the response variable; defaults to binomial distribution. See glm and glmer for details.
link	a specification for the model link function. This can be a name/expression, a literal character string or a length-one character vector; defaults to logit link function. See glmer for details.
nAGQ	an integer scalar - the number of points per axis for evaluating the adaptive Gauss-Hermite approximation to the log-likelihood; defaults to 1, corresponding to the Laplace approximation. Values greater than 1 produce greater accuracy in the evaluation of the log-likelihood at the expense of speed.
control	a list (of correct class, resulting from lmerControl() or glmerControl() respectively) containing control parameters, including the nonlinear optimizer to be used and parameters to be passed through to the nonlinear optimizer, see the <code>*lmerControl*</code> documentation for details.

Value

An object of class "REMB" is returned, consisting of prediction of the probability of toxicity and estimations. Objects generated by MA_estimates contain at least the following components:

dataTox	A data frame containing information for each patient at each simulation and trial.
sim0	The simulation starting point; must be numeric.
sim1	The simulation ending point; must be numeric.

family	A distribution family for the response variable; defaults to binomial distribution. See glm and glmer for details.
link	A specification for the model link function. This can be a name/expression, a literal character string or a length-one character vector; defaults to logit link function. See glmer for details.
estimates	A data frame including the predicted toxicity probabilities (column named as "rndpt"), at each simulation, trial and dose, using either the predicted random effect center "mu" (column named as "PredMu"), if it's available, or the value ratio (i.e. # of toxicities / # of subjects) if the corresponding "mu" is not available. Note that, "PredMu" value is equal to "NA" when there is no prediction for it.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

See Also

[glimem](#)

Examples

```
## Not run:
#####
### Give a toxicity data for each simulation and each trial. ###
##### using "Toxdata" including in package #####
#####

data("Toxdata")
dim(Toxdata)
colnames(Toxdata)

## NOTE: The toxicity dataset must have the same structure as below. ##
## Check it before running the VarWT function! ##

str(Toxdata)

## Running the MA_estimates function! ##

test <- MA_estimates(Toxdata, 10, 10, family = binomial, link = "logit", nAGQ = 1,
  control = glmerControl(optimizer = "bobyqa"))

### You can see the "show" S4 method to present the REMB algorithm's results
test

#####
## Taking the predicted probability of toxicity ##
#####

test@estimates
```



```
## End(Not run)
```

```
plot,ZKO,missing-method
```

The graphical representation of dose-toxicity curve.

Description

A plotting function representing graphically the dose-toxicity relationship including the target probability of toxicity. In addition you can see as circles the proportional sizes of the number of patients evaluated for toxicity at each dose level.

Usage

```
## S4 method for signature 'ZKO,missing'  
plot(x, y = NA, ...)
```

Arguments

x a "ZKO" object.
y the "y" argument is not used in the plot-method for "ZKO" object.
... other arguments to the `plot.default` function can be passed here.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anaund N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

References

Zohar, S. , Katsahian, S. and O'Quigley, J. (2011), An approach to meta analysis of dose finding studies. *Statist. Med.*, <doi:10.1002/sim.4121>.

```
REMB-class
```

An S4 class to represent the results using the random effect model-based (REMB) algorithm.

Description

An S4 class to represent the results using the random effect model-based (REMB) algorithm.

Slots

- `dataTox` A data frame containing information for each patient at each simulation and trial.
- `sim0` The simulation starting point; must be numeric.
- `sim1` The simulation ending point; must be numeric.
- `family` A distribution family for the response variable; defaults to binomial distribution. See [glm](#) and [glmer](#) for details.
- `link` A specification for the model link function. This can be a name/expression, a literal character string or a length-one character vector; defaults to logit link function. See [glmer](#) for details.
- `estimates` A data frame including the predicted toxicity probabilities (column named as "rndpt") using either the predicted random effect center "mu" (column named as "PredMu"), if it's available, or the value ratio (i.e. # of toxicities / # of subjects) if the corresponding "mu" is not available. Note that, "PredMu" value is equal to "NA" when there is no prediction for it.

`show-methods` *S4 Methods for Function show*

Description

S4 Methods for function show.

Methods

- `signature(object = "ZKO")` S4 method to store and present the ZKO algorithm's results.
- `signature(object = "REMB")` S4 method to store and present the REMB algorithm's results.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

Toxdata *A toxicity database.*

Description

A dataframe consisting 60000 observations of 6 different variables.

Usage

```
data("Toxdata")
```

Format

A data frame with 60000 observations on the following 6 variables.

`simulation` a numeric vector including the simulation.

`dose` a numeric vector of the dose levels.

`Trial` a numeric vector which gives the number of the trial.

`n` a numeric vector of the number of allocation.

`x` a numeric vector giving the number of toxicities.

`trueptoxic` a numeric vector consisting the true probability of toxicity.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

References

Zohar, S., Katsahian, S. and O'Quigley, J. (2011), An approach to meta-analysis of dose-finding studies. *Statist. Med.*

Examples

```
data("Toxdata")
## maybe str(Toxdata): plot(Toxdata) ...
```

 VarWT

The variance weighted (VarWT) pooling approach.

Description

An alternative method to combine toxicities across the trials. Accordingly, we estimate the pooled toxicity probability at dose level x using the

$$R_c(x) = \sum_{i=1}^R \frac{\sigma_i^2(x)}{\sigma^2(x)} \bar{R}_i(x),$$

where

$$\sigma_i^2(x) = N_i(x) \bar{R}_i(x) (1 - \bar{R}_i(x))$$

and

$$\sigma^2(x) = \sum_{i=1}^R \sigma_i^2(x), x \in D.$$

Then, to find the pooled estimates of MTD you can use either (a) the non-parametric pooled MTD estimator can be defined by the dose $x_0 \in D$ such that:

$$|R_c(\hat{x}_0) - \theta| \leq |R_c(x) - \theta|, x \neq \hat{x}_0$$

OR (b) the model-based algorithm.

Usage

```
VarWT(dataTox, saveName)
```

Arguments

dataTox	a data frame containing information for each patient at each simulation and trial.
saveName	the name of the .txt output which will be created and saved in the current working directory; must be in characters; defaults to NULL. If saveName = NULL, means that nothing will be saved in the working directory.

Value

A list is returned, combining the toxicities across the trials. The output, generated by the function VarWT, contains the following components:

dataTox	A data frame containing the estimated pooled toxicity probabilities at each dose level for each simulation.
---------	---

If saveName is not equal to "NULL" then a .txt file is saved in the working directory storing the estimated pooled toxicity probabilities at each dose level for each simulation.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

See Also

[ZKO](#)

Examples

```
## Not run:
## Using the toxicity dataset for each simulation and each trial, which is
## existing in the dfmeta package.

data("Toxdata")
dim(Toxdata)
colnames(Toxdata)

## NOTE: The toxicity dataset must have the same structure as below. ##
## Check it before running the VarWT function! ##

str(Toxdata)

#####
## Running the VarWT function! ##
#####

WT <- VarWT(Toxdata)
```

```
##### If we want to save the result in a text file #####
##### we need to give a name in the argument "saveName" #####

WT <- VarWT(Toxdata, tempdir())

## End(Not run)
```

ZKO

Meta-analysis approach based on Zohar et al, 2011.

Description

This function utilizes the meta-analysis approach proposed by Zohar et al, 2011. It utilizes the power model

$$\Psi(x, \alpha) = (\alpha(x))^a$$

in order to estimate a by running simulated CRMs using

$$R_C(x) = \sum_{i=1}^r \frac{N_{i.}}{N_{..}} \bar{R}_i(x).$$

Then the estimated dose-toxicity curve is used to estimate the maximum tolerated dose (MTD).

Usage

```
ZKO(dataTox, doses, target = 0.3, dr, nbSimu)
```

Arguments

dataTox	a data frame containing information for each patient at each simulation and trial. The data frame must have the following structure with the column's names be set as: 1) "doses", representing the drug's dose levels, 2) "proba", representing the true probabilities of toxicity, 3) "nbPatients", representing the number of patients at each dose level and 4) "nbTox", representing the number of DLTs obtained at each dose level. The data frames with different structure will not be accepted in the function.
doses	a vector containing the drug's dose levels.
target	the target of the probability of toxicity; defaults to 0.3.
dr	the initial guess of dose-toxicity relationship (usually is selected based on information from experts).
nbSimu	the number of simulated CRM studies.

Value

An object of class "ZKO" is returned, consisting the estimated probability of toxicity. Objects generated by ZKO function contain at least the following components:

dataTox	A data frame containing information for each patient at each simulation and trial. The data frame must be have the following structure with the column's names be set as: 1) "doses", 2) "proba", 3) "nbPatients" and 4) "nbTox". Note that, any data frames with different structure will not be accepted in the function.
target	The target of probability of toxicity.
doses	A vector with the doses panel.
ti	A vector including the total number of toxicities at each dose level.
ni	A vector consisting the total number of patients at each dose level.
realite	A vector with the observed toxicity probabilities at each dose level.
relfreq	A vector with the mean relative frequencies of allocation at each dose level.
estimAlpha	The estimated value for variable alpha.
estimProb	The estimated probability of toxicity.

Author(s)

Artemis Toumazi <artemis.toumazi@gmail.com>, Sarah Zohar <sarah.zohar@inserm.fr>, Anand N. Vidyashankar <avidyash@gmu.edu>, Jie Xu <jxu13@gmu.edu> and Moreno Ursino <moreno.ursino@inserm.fr>

References

Zohar, S., Katsahian, S. and O'Quigley, J. (2011), An approach to meta analysis of dose finding studies. *Statist. Med.*, <doi:10.1002/sim.4121>.

See Also

[VarWT](#)

Examples

```
## Not run:

#####
#### A simple example using a simple data ####
#####

vanGroeningen <- rbind(c(25,0/6, 6), c(35, 0/5,5), c(40, 4/11, 11), c(45, 3/5, 5))
Hoff <- rbind(c(30, 3/9, 9), c(35, 2/3, 3), c(40, 2/3, 3))
Zhu <- rbind(c(30, 0/12, 12), c(35, 2/3, 3))
dataTox <- rbind(vanGroeningen, Hoff, Zhu)
dataTox <- data.frame(dataTox)
colnames(dataTox) <- c("doses", "proba", "nbPatients")
nbTox <- dataTox$proba*dataTox$nbPatients
dataTox <- data.frame(dataTox, nbTox)
doses <- c(25, 30, 35, 40, 45)
```

```

nbSimu <- 10
zko <- ZKO(dataTox, doses, target = 0.3, dr = c(0.05, 0.15, 0.2, 0.3, 0.35), nbSimu)

#####
##### A second example using the data "Toxdata" #####
##### included in the "dfmeta" R package #####
#####

data(Toxdata)
pos <- which(Toxdata$simulation == 0)
dataTox2 <- Toxdata[pos, ]
dataTox2 <- dataTox2[, c(2,6,4,5)]

### Remember that the structure of the database must be in the following structure ###

colnames(dataTox2) <- c("doses", "proba", "nbPatients", "nbTox")
doses <- 0:5
nbSimu <- 10
zko2 <- ZKO(dataTox2, doses, target = 0.3, dr = c(0.05, 0.15, 0.2, 0.3, 0.35, 0.45),
nbSimu)

## End(Not run)

```

ZKO-class

An S4 class to represent the results using the ZKO algorithm.

Description

An S4 class to represent the results using the ZKO algorithm.

Slots

`dataTox` A data frame containing information for each patient at each simulation and trial.

`target` The target of probability of toxicity.

`doses` A vector with the doses panel.

`ti` A vector including the total number of toxicities at each dose level.

`ni` A vector consisting the total number of patients at each dose level.

`realite` A vector with the observed toxicity probabilities at each dose level.

`relfreq` A vector with the mean relative frequencies of allocation at each dose level.

`estimAlpha` The estimated value for variable alpha.

`estimProb` The estimated probability of toxicity.

Index

*Topic **datasets**

Toxdata, [10](#)

*Topic **methods**

show-methods, [10](#)

*Topic **package**

dfmeta-package, [2](#)

dfmeta (dfmeta-package), [2](#)

dfmeta-package, [2](#)

doseRecords, [3](#)

findFirstLast, [4](#)

glimem, [3](#), [5](#), [8](#)

glm, [5](#), [7](#), [8](#), [10](#)

glmer, [5](#), [7](#), [8](#), [10](#)

glmerControl, [5](#), [7](#)

lmerControl, [5](#), [7](#)

MA_estimates, [3](#), [6](#), [7](#)

plot, ZKO, missing-method, [9](#)

plot.default, [9](#)

REMB-class, [9](#)

show, REMB-method (show-methods), [10](#)

show, ZKO-method (show-methods), [10](#)

show-methods, [10](#)

Toxdata, [10](#)

VarWT, [4](#), [11](#), [14](#)

ZKO, [12](#), [13](#)

ZKO-class, [15](#)