

Package ‘idem’

June 30, 2017

Title Inference in Randomized Controlled Trials with Death and Missingness

Version 2.2

Author Chenguang Wang [aut, cre],
Andrew Leroux [aut, cre],
Elizabeth Colantuoni [aut],
Daniel O Scharfstein [aut],
Trustees of Columbia University [cph] (tools/make_cpp.R, R/stanmodels.R)

Maintainer Chenguang Wang <cwang68@jhmi.edu>

Description In randomized studies involving severely ill patients, functional outcomes are often unobserved due to missed clinic visits, premature withdrawal or death. It is well known that if these unobserved functional outcomes are not handled properly, biased treatment comparisons can be produced. In this package, we implement a procedure for comparing treatments that is based on the composite endpoint of both the functional outcome and survival. The procedure was proposed in Wang et al. (2016) <doi:10.1111/biom.12594>. It considers missing data imputation with a sensitivity analysis strategy to handle the unobserved functional outcomes not due to death.

Depends R (>= 3.3.0), Rcpp (>= 0.12.5), rstan (>= 2.14)

License GPL (>= 3)

LinkingTo StanHeaders (>= 2.11.0), rstan (>= 2.14), BH (>= 1.60.0),
Rcpp (>= 0.11.0), RcppEigen

RcppModules stan_fit4idem_mod

Imports sqldf (>= 0.4), survival (>= 2.38), parallel (>= 3.2), methods

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

Suggests knitr, shiny, rmarkdown, pander, DT, shinythemes

VignetteBuilder knitr

NeedsCompilation yes

Repository CRAN
Date/Publication 2017-06-30 13:30:33 UTC

R topics documented:

| | |
|----------------------------|-----------|
| idem-package | 2 |
| abc | 3 |
| idem-parameters | 4 |
| imBs | 5 |
| imChkPars | 6 |
| imEstimate | 7 |
| imFitModel | 8 |
| imImpAll | 9 |
| imImpSingle | 10 |
| imMisTable | 11 |
| imNeedImp | 12 |
| imPlotCompleters | 12 |
| imPlotComposite | 13 |
| imPlotContour | 14 |
| imPlotImputed | 15 |
| imPlotMisPattern | 16 |
| imPlotSurv | 17 |
| imShiny | 17 |
| imTest | 18 |
| Index | 20 |

| | |
|--------------|---|
| idem-package | <i>Inference in Randomized Clinical Trials with Death and Missingness</i> |
|--------------|---|

Description

This package contains the functions for drawing inference in randomized clinical trials with death and intermittent missingness.

Notation

Consider a two-arm randomized study. Let Y_k denote outcome measured at time t_k and Z denote a functional endpoint that is a function of Y . Let L denote the survival time. Let X denote the baseline covariates and T denote the treatment assignment.

Ranking

If two subjects were both alive at the end of the study, they are ranked based on functional outcome Z . If at least one subject was dead at the end of the study, they are ranked based on survival time L .

Treatment effect, θ is defined as the probability that the outcome for a random individual randomized to treatment $T = 0$ is less than the outcome of a random individual randomized to treatment $T = 1$ minus the probability that the outcome for a random individual randomized to treatment $T = 0$ is greater than the outcome of a random individual randomized to treatment $T = 1$.

Missingness

In order to estimate θ in the presence of missing data, we need to impute Z for subjects alive at the end of the study with Y_k missing for some k .

The benchmark assumption we consider for the imputation is the complete case missing value (CCMV) restrictions. We then consider exponential tilting models for introducing sensitivity parameters for evaluating the robustness of the findings with regards to different missing data mechanism assumptions. The models are as follows:

$$f(Y_{mis}^{(s)}|Y_{obs}^{(s)}, Y_0, X, T, S = s) \propto \exp(\beta_T Z) f(Y_{mis}^{(s)}|Y_{obs}^{(s)}, Y_0, X, T, S = 1)$$

where S denotes the missingness patterns, $S = 1$ denotes the completers and β_T denotes the sensitivity parameter for arm T .

Graphical user interface (GUI)

This package provides a web-based GUI. See [imShiny](#) for details.

References

Wang C, Scharfstein DO, Colantuoni E, Girard T, Yan Y (2016). Inference in Randomized Trials with Death and Missingness.

abc

Example dataset

Description

The Awakening and Breathing Controlled (ABC) trial randomized critically ill patients receiving mechanical ventilation 1:1 within each study site to management with a paired sedation plus ventilator weaning protocol involving daily interruption of sedative through spontaneous awakening trials (SATs) and spontaneous breathing trials (SBTs) or sedation per usual care (UC) and SBTs.

The example dataset is from a single site substudy in ABC. The researchers assessed differences in cognitive, psychological and functional outcomes at 3 and 12 months after randomization. , respectively).

Format

A dataframe with 5 variables:

AGE Age

TRT Treatment assignment. 0: UC + SBT, 1: SAT + SBT

SURV Survival days

Y2 Cognitive score at 12 months

Y1 Cognitive score at 3 months

References

T. D. Girard, J. P. Kress, B. D. Fuchs, J. W. W. Thomason, W. D. Schweickert, B. T. Pun, D. B. Taichman, J. G. Dunn, A. S. Pohlman, P. A. Kinniry, J. C. Jackson, A. E. Canonico, R. W. Light, A. K. Shintani, J. L. Thompson, S. M. Gordon, J. B. Hall, R. S. Dittus, G. R. Bernard, and E. W. Ely. Efficacy and safety of a paired sedation and ventilator weaning protocol for mechanically ventilated patients in intensive care (awakening and breathing controlled trial): a randomised controlled trial. *Lancet*, 371:126-134, 2008.

idem-parameters

List of parameters for idem analysis

Description

The parameters used by most of the functions in *idem* are organized as a list. These parameters include variable names in the analysis dataset, endpoint specification, duration of the study, etc..

Arguments

| | |
|-----------|--|
| trt | Variable name for the Control (0) and Intervention (1) treatment assignments in the dataset. |
| surv | Variable name for the survival (time to event) variable in the dataset. |
| outcome | Chronologically ordered vector of variable names for clinical outcomes in the dataset excluding baseline. |
| y0 | Variable name of the baseline clinical outcome. |
| cov | Vector of variable names for the covariates used in the imputation procedure for missing clinical outcomes. |
| endfml | R expression indicating the user-specified final outcome of interest. This is the function for Z of one or more of Y_k 's. |
| duration | Length of the study. This is the time at which subjects' are assumed to be censored. |
| bounds | Numeric vector of lower and upper bounds for subjects' imputed clinical outcomes. |
| trt.label | label of the treatment arms |
| unitTime | Unit of time measurement for survival and function outcome time points |

Examples

```
## for example abc dataset

lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"),
               y0=NULL, endp=c("Y2"),
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2",
               duration=365, bounds=c(0,100));
```

imBs

Bootstrap analysis

Description

Bootstrap analysis

Usage

```
imBs(impr.rst, n.boot = 100, n.cores = 1, update.progress = NULL,
     quantiles = 0.5)
```

Arguments

| | |
|------------------------------|---|
| <code>impr.rst</code> | A class IDEM.IMP list containing complete data with relevant missing values imputed. See imImpAll . |
| <code>n.boot</code> | Number of bootstrap samples |
| <code>n.cores</code> | Number of cores for parallel computation |
| <code>update.progress</code> | Parameter reserved for run idem in GUI mode |
| <code>quantiles</code> | Quantiles of the composite endpoint to be reported |

Value

A class IDEM.BOOT list with length `n.boot+1`. Each item in the list is a class {IDEM.RST} list (see [imEstimate](#)). The first item corresponds to the estimation result on the original dataset.

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
```

```

rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
                  normal=TRUE, chains = 4, iter = 2000, warmup = 1000);
rst.boot <- imBs(rst.imp, n.boot = 10, n.cores = 5, quantiles = c(0.25,0.5,0.75));
## End(Not run)

```

imChkPars

Check parameter specification

Description

Check if the idem-parameters are correctly specified and consistent with the data

Usage

```
imChkPars(data.all, lst.var, html = FALSE)
```

Arguments

| | |
|----------|--|
| data.all | Original dataset |
| lst.var | see idem-parameters |
| html | logic indicator for the format of the error messages |

Value

NULL if the specification is correct.

Text messages if html=TRUE.

Examples

```

err.lst.var <- list(trt="TRT", outcome=c("Y1","Y2"),
                  y0=NULL, endfml="Y3", bounds=c(10,20),
                  duration=365);

imChkPars(abc, err.lst.var);

```

| | |
|------------|------------------------------------|
| imEstimate | <i>Treatment effect estimation</i> |
|------------|------------------------------------|

Description

Estimate treatment effect and median of the composite endpoint from using imputed data

Usage

```
imEstimate(imp.rst, quantiles = 0.5, ...)
```

Arguments

| | |
|------------------|---|
| imp.rst | A class IDEM.IMP list containing complete data with relevant missing values imputed. See imImpAll . |
| quantiles | Quantiles of the composite endpoint to be reported |
| ... | Options for ranking subjects using the composite endpoint |
| cut.z | Clinically meaningful difference in the functional outcome |
| cut.surv | Clinically meaningful difference in survival time |

Value

A class IDEM.RST list contains

list.var List of parameters

theta A dataset with columns Delta0, Delta1, $\hat{\theta}$

quantiles A dataset with columns Delta, Trt, Quantiles

survivor A dataset for survivors with columns Delta0, Delta1, Mean0, Mean1, Diff

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFit(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, lst.var, deltas=c(-0.25,0,0.25),
                  normal=TRUE, iter=300, n.imp=2, thin=10, p.scale=10);
rst.est <- imEstimate(rst.imp, quantiles=c(0.25,0.5,0.75));
## End(Not run)
```

imFitModel

*Model fitting***Description**

Fit linear imputation models to the observed data from complete survivors for each treatment arm at each time point

Usage

```
imFitModel(data.all = NULL, lst.var = NULL)
```

Arguments

| | |
|----------|-------------------------------------|
| data.all | Original dataset |
| lst.var | see idem-parameters |

Value

A class IDEM.FIT list of modeling fitting results with the following items

lst.var List of parameters

rst.mdl A list of modeling fitting results for each model with

- lm** results from function `lm`
- formula** model formula
- coef** model coefficients
- res** residuals
- h** bandwidth of residuals for kernel density estimation

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));

rst.fit <- imFitModel(abc, lst.var);
```


imImpAll

*Impute missing data***Description**

Impute missing data for all the subjects or a small sample of the subjects

Usage

```
imImpAll(data.all, fit.rst, normal = TRUE, n.imp = 5, endponly = TRUE,
         deltas = 0, update.progress = NULL, imputeNone = FALSE, ...)
```

Arguments

| | |
|-----------------|--|
| data.all | Original dataset |
| fit.rst | A class IDEM.FIT results of linear regression. See imFitModel . |
| normal | Logical variable indicating whether normality assumption should be made for the residuals |
| n.imp | Number of complete datasets required |
| endponly | Logical variable indicating whether clinical outcomes not used in calculating the final clinical outcome will be imputed. The default is FALSE, indicating that all missing clinical outcomes will be imputed sequentially |
| deltas | Vector of imputation sensitivity parameters |
| update.progress | Parameter reserved for run idem in GUI mode |
| imputeNone | If TRUE, return subjects that do not need imputation |
| ... | options to call STAN sampling. These options include chains, iter, warmup, thin, algorithm. See <code>rstan::sampling</code> for details. |

Value

If imputeNone is TRUE, return a dataset with the original data for the subset of subjects who died at the end of the study or had no missing outcomes.

Otherwise, return a class IDEM.IMP list with components

lst.var List of parameters

complete A dataset with the original data for the subset of subjects who died at the end of the study or had no missing outcomes and the n.imp imputed missing outcomes for subjects who need missing value imputation.

n.imp Number of imputed complete datasets

deltas Imputation sensitivity parameters

org.data Original dataset

normal Normal assumption for the imputation

stan.par parameters in ...

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
                  normal=TRUE, chains = 2, iter = 2000, warmup = 1000);
## End(Not run)
```

imImpSingle

Impute missing data under benchmark assumption

Description

Call STAN model to impute missing data for an individual subject under benchmark assumption

Usage

```
imImpSingle(dsub, fit.rst, normal = TRUE, chains = 4, iter = 5000,
            warmup = 1000, control = list(adapt_delta = 0.95), ...)
```

Arguments

| | |
|---------|--|
| dsub | original individual subject data |
| fit.rst | A class IDEM.FIT results of linear regression. See imFitModel . |
| normal | Logical variable indicating whether normality assumption should be made for the residuals |
| chains | STAN parameter. Number of Markov chains |
| iter | STAN parameter. Number of iterations |
| warmup | STAN parameter. Number of burnin. |
| control | STAN parameter. See <code>rstan::stan</code> for details. |
| ... | other options to call STAN sampling such as <code>thin</code> , <code>algorithm</code> . See <code>rstan::sampling</code> for details. |

Value

NULL if there is no missing data for the current subject.

Otherwise, return a class IDEM.IMPSUB that contains a list with two components

dsub original data of the subject

rst.stan A `stan.fit` class result returned from `rstan::sampling`

complete A dataframe with complete data for the selected subject

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
              endp=c("Y2"), unitTime="days",
              trt.label = c("UC+SBT", "SAT+SBT"),
              cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpSingle(abc[1,], rst.fit, chains = 4, iter = 2000, warmup = 1000);
rstan::traceplot(rst.imp$rst.stan, "YMIS");
```

| | |
|------------|--|
| imMisTable | <i>Generate table of missingness pattern frequencies</i> |
|------------|--|

Description

Generate table of missingness pattern frequencies

Usage

```
imMisTable(data.all, lst.var)
```

Arguments

| | |
|----------|-------------------------------------|
| data.all | Original dataset |
| lst.var | see idem-parameters |

Value

A matrix with frequencies of each missing pattern

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"),
              y0=NULL, trt.label = c("UC+SBT", "SAT+SBT"),
              duration=365);

imMisTable(abc, lst.var);
```

| | |
|-----------|--|
| imNeedImp | <i>Get subjects that need imputation</i> |
|-----------|--|

Description

Get the index of subjects in a dataset that need imputation, i.e. survivors with functional endpoint missing

Usage

```
imNeedImp(data.all, lst.var, endponly = TRUE)
```

Arguments

| | |
|----------|--|
| data.all | Original dataset |
| lst.var | see idem-parameters |
| endponly | Logical variable indicating whether clinical outcomes not used in calculating the final clinical outcome will be imputed. The default is FALSE, indicating that all missing clinical outcomes will be imputed sequentially |

Value

Vector of indices of subjects that need imputation

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
              endp=c("Y2"), unitTime="days",
              trt.label = c("UC+SBT", "SAT+SBT"),
              cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
inx.imp <- imNeedImp(abc, lst.var);
```

| | |
|------------------|-------------------------------|
| imPlotCompleters | <i>Plot data of completer</i> |
|------------------|-------------------------------|

Description

Spaghetti plot for subjects alive at the end of the study without missing data

Usage

```
imPlotCompleters(data.all, lst.var, fname = NULL, ...)
```

Arguments

| | |
|----------|---|
| data.all | Original dataset |
| lst.var | see idem-parameters |
| fname | File name of the result pdf file. If fname is null, result pdf file will not be generated |
| ... | Options for pdf function |

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"),
              y0=NULL, trt.label = c("UC+SBT", "SAT+SBT"),
              duration=365);

imPlotCompleters(abc, lst.var);
```

| | |
|-----------------|------------------------|
| imPlotComposite | <i>Cumulative Plot</i> |
|-----------------|------------------------|

Description

Generate cumulative plot of the composite survival and functional outcome

Usage

```
imPlotComposite(imp.rst, delta = 0, buffer = 0.05, at.surv = NULL,
                at.z = NULL, p.death = NULL, seg.lab = c("Survival", "Functional"),
                fname = NULL, cols = rep(c("cyan", "red"), 3), ltys = rep(1, 6),
                main = "", ...)
```

Arguments

| | |
|---------|---|
| imp.rst | A class IDEM.IMP list containing complete data with relevant missing values imputed. See imImpAll . |
| delta | Imputation sensitivity parameter for which to generate the results |
| buffer | Small horizontal gap used to better visually distinguish the transition from survival to functional outcome. |
| at.surv | Sets the range of the survival times to plot in the cumulative distribution function. By default the range is the range of survival values up to the duration of the study. |
| at.z | Sets the range of the functional outcome to plot in the cumulative distribution function. By default this is the range of the functional outcomes plus the buffer amount to improve visibility in the transition from survival to functional outcome. |

| | |
|---------|---|
| p.death | Proportion of the plot width devoted to Survival. By default the cumulative distribution will devote horizontal space to the survival portion that is proportional to the number of subjects who die prior to duration. |
| seg.lab | Labels for the two components of the composite outcome. |
| fname | File name of the result pdf file. If fname is null, result pdf file will not be generated |
| cols | plot options |
| ltys | plot options |
| main | plot options |
| ... | Options for pdf function |

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
                 normal=TRUE, chains = 4, iter = 2000, warmup = 1000);
imPlotComposite(rst.imp);
## End(Not run)
```

imPlotContour

Contour plot of the sensitivity analysis results

Description

Generate contour plot of p-values for sensitivity analysis results

Usage

```
imPlotContour(test.rst, con.v = 0.05, nlevels = 30, ...)
```

Arguments

| | |
|----------|--|
| test.rst | A class IDEM.TEST list generated by imTest |
| con.v | Levels of contour plot |
| nlevels | Levels of color scale |
| ... | Options for filled.contour |

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
                  normal=TRUE, chains = 4, iter = 2000, warmup = 1000);
rst.final <- imTest(rst.boot);
rst.boot <- imBs(rst.imp, n.boot = 10, n.cores = 5);
imPlotContour(rst.final, con.v=0.05, nlevels = 30);
## End(Not run)
```

| | |
|---------------|---------------------------------------|
| imPlotImputed | <i>Plot density of imputed values</i> |
|---------------|---------------------------------------|

Description

Plot density of imputed values and the density of the observed outcomes

Usage

```
imPlotImputed(imp.rst, deltas = 0, endp = FALSE, fname = NULL,
              adj = 1.5, cols = c("red", "cyan", "blue", "green", "brown"),
              ltys = rep(1, 6), xlim = NULL, ylim = NULL, mfrow = NULL, ...)
```

Arguments

| | |
|---------|---|
| imp.rst | A class IDEM.IMP list containing complete data with relevant missing values imputed. See imImpAll . |
| deltas | Imputation sensitivity parameter for which to generate the results |
| endp | If TRUE, plot the densities of the imputed functional outcomes. Otherwise, plot the densities of the imputed outcomes |
| fname | File name of the result pdf file. If fname is null, result pdf file will not be generated |
| adj | density estimation option |
| cols | plot options |
| ltys | plot options |
| xlim | plot options |
| ylim | plot options |
| mfrow | plot options |
| ... | Options for pdf function |

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
                  normal=TRUE, chains = 4, iter = 2000, warmup = 1000);
imPlotImputed(rst.imp, deltas=c(-0.25,0,0.25), xlim=c(0,100), endp=FALSE);
## End(Not run)
```

| | |
|------------------|------------------------------|
| imPlotMisPattern | <i>Plot missing patterns</i> |
|------------------|------------------------------|

Description

Plot the missing patterns of the observed data

Usage

```
imPlotMisPattern(data.all, lst.var, cols = c("blue", "gray"), fname = NULL,
...)
```

Arguments

| | |
|----------|---|
| data.all | Original dataset |
| lst.var | see idem-parameters |
| cols | Color of observed and missing values |
| fname | File name of the result pdf file. If fname is null, result pdf file will not be generated |
| ... | Options for pdf function |

Examples

```
lst.var <- list(trt="TRT", outcome=c("Y1","Y2"),
               trt.label = c("UC+SBT", "SAT+SBT"));

imPlotMisPattern(abc, lst.var);
```

| | |
|------------|-----------------------------|
| imPlotSurv | <i>Plot survival curves</i> |
|------------|-----------------------------|

Description

Plot Kaplan-Meier survival curves

Usage

```
imPlotSurv(data.all, lst.var, cols = c("black", "blue"), fname = NULL, ...)
```

Arguments

| | |
|----------|---|
| data.all | Original dataset |
| lst.var | see idem-parameters |
| cols | Curve colors of the treatment and control arm |
| fname | File name of the result pdf file. If fname is null, result pdf file will not be generated |
| ... | Options for pdf function |

Examples

```
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"),
               y0=NULL, trt.label = c("UC+SBT", "SAT+SBT"),
               duration=365);
imPlotSurv(abc, lst.var);
```

| | |
|---------|---------------------------------------|
| imShiny | <i>Run Web-Based idem application</i> |
|---------|---------------------------------------|

Description

Call Shiny to run idem as a web-based application. A web browser will be brought up.

Usage

```
imShiny()
```

Examples

```
## Not run:
run.idem()
## End(Not run)
```

imTest

*Hypothesis testing***Description**

Hypothesis testing using the estimation for the original dataset and Summarize Bootstrap analysis results

Usage

```
imTest(bs.rst, quantiles = c(0.025, 0.975))
```

Arguments

bs.rst A class IDEM.BOOT result list from [imBs](#) for bootstrap analysis
quantiles Quantiles for extracting bootstrap confidence intervals

Value

A class IDEM.TEST containing two datasets

list.var List of parameters

theta With columns

- Delta0: Sensitivity parameter for control arm,
- Delta1: Sensitivity parameter for intervention arm
- Theta: Estimated θ
- SD: Standard deviation
- PValue: p-value

quantiles With columns

- Delta: Sensitivity parameter
- TRT: Treatment arm
- Q: Quantiles of the composite endpoint to be estimated
- Quant: Estimation
- LB: Lower bound of the specified confidence interval
- UB: Upper bound of the specified confidence interval

Examples

```
## Not run:
lst.var <- list(trt="TRT", surv="SURV", outcome=c("Y1","Y2"), y0=NULL,
               endp=c("Y2"), unitTime="days",
               trt.label = c("UC+SBT", "SAT+SBT"),
               cov=c("AGE"), endfml="Y2", duration=365, bounds=c(0,100));
rst.fit <- imFitModel(abc, lst.var);
rst.imp <- imImpAll(abc, rst.fit, deltas=c(-0.25,0,0.25),
```

```
                                normal=TRUE, chains = 4, iter = 2000, warmup = 1000);  
rst.boot  <- imBs(rst.imp, n.boot = 10, n.cores = 5);  
rst.final <- imTest(rst.boot);  
## End(Not run)
```

Index

abc, [3](#)

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

idem-package, [2](#)

idem-parameters, [4](#)

imBs, [5](#), [18](#)

imChkPars, [6](#)

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

imFitModel, [8](#), [9](#), [10](#)

imImpAll, [5](#), [7](#), [9](#), [13](#), [15](#)

imImpSingle, [10](#)

imMisTable, [11](#)

imNeedImp, [12](#)

imPlotCompleters, [12](#)

imPlotComposite, [13](#)

imPlotContour, [14](#)

imPlotImputed, [15](#)

imPlotMisPattern, [16](#)

imPlotSurv, [17](#)

imShiny, [3](#), [17](#)

imTest, [14](#), [18](#)