Package 'psfmi'

February 3, 2020

Type Package

Depends R (>= 3.5.0),

Imports survival (> 2.41-3), car (> 3.0-0), norm (>= 1.0-9.5),
miceadds (> 2.10-14), mitools (>= 2.4), foreign (>= 0.8-72),
pROC (> 1.11.0), rms (> 5.1-2), ResourceSelection (> 0.3-2),
ggplot2 (> 2.2.1), dplyr (>= 0.8.3), magrittr (>= 1.5), rsample
(>= 0.0.5), purrr (>= 0.3.3), tidyr (>= 1.0.0), tibble (>=
2.1.3), lme4 (>= 1.1-21), mice (>= 3.6.0), mitml (>= 0.3-7)

Title Prediction Model Selection and Performance Evaluation in
Multiple Imputed Datasets

Version 0.2.0

Description Provides functions to apply pooling or backward selection of logistic, Cox regression and Multilevel (mixed models) prediction models in multiply imputed datasets. Backward selection can be done from the pooled model using Rubin's Rules (RR), the D1, D2, D3 and promising median p-values method. The model can contain continuous, dichotomous, categorical predictors and interaction terms between all these type of predictors. Continuous predictors can also be introduced as restricted cubic spline coefficients. It is also possible to force (spline) predictors or interaction terms in the model during predictor selection. The package includes a function to evaluate the stability of the models using bootstrapping and cluster bootstrapping. The package further contains functions to generate pooled model performance measures in multiply imputed datasets as ROC/AUC, R-squares, Brier score, fit test values and calibration plots for logistic regression models. A function to apply Bootstrap internal validation is also available where two methods can be used to combine bootstrapping and multiple imputation. One method, boot_MI, first draws bootstrap samples and subsequently performs multiple imputation and with the other method, MI_boot, first bootstrap samples are drawn from each imputed dataset before results are combined. The adjusted intercept after shrinkage of the pooled regression coefficients can be subsequently obtained. Backward selection as part of internal validation is also an option. Also a function to externally validate logistic prediction models in multiple imputed datasets is available.
Eekhout (2017) <doi:10.1186/s12874-017-0404-7>.
Wiel (2009) <doi:10.1093/biostatistics/kxp011>.

Marshall (2009) <doi:10.1186/1471-2288-9-57>.

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

License GPL (>= 2)

URL https://github.com/mwheymans/psfmi

BugReports https://github.com/mwheymans/psfmi/issues

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Martijn Heymans [cre, aut],
Iris Eekhout [ctb]

Maintainer Martijn Heymans <mw.heymans@amsterdamumc.nl>

Repository CRAN

Date/Publication 2020-02-03 07:30:02 UTC

R topics documented:

D1_cox *D1 method called by psfmi_cox*

Description

`D1_cox` D1 pooling method

Usage

```
D1_cox(data, impvar, nimp, fm, names.var)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`fm` regression formula from coxph object

`names.var` list of predictors included in pooled regression model

Examples

```
D1_cox(data=lbpmicox, nimp=5, impvar="Impnr",
fm=survival::Surv(Time, Status) ~ Duration + Radiation + Onset,
names.var=list("Duration", "Radiation", "Onset"))
```

D1_logistic *D1 method called by psfmi_lr*

Description

`D1_logistic` D1 pooling method

Usage

```
D1_logistic(data, impvar, nimp, fm, names.var)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`fm` regression formula from glm object

`names.var` list of predictors included in pooled regression model

Examples

```
D1_logistic(data=lbpmilr, nimp=5, impvar="Impnr",
fm=Chronic ~ Gender + Smoking + Function + JobControl,
names.var=list("Gender", "Smoking", "Function", "JobControl"))
```

ipdna_md *Example dataset for the psfmi_mm function*

Description

5 imputed datasets of the first 10 centres of the IPDNa dataset in the micemd package.

Usage

```
data(ipdna_md)
```

Format

A data frame with 13390 observations on the following 13 variables.

.imp a numeric vector

.id a numeric vector

centre cluster variable

gender dichotomous

bmi continuous

age continuous

sbp continuous

dbp continuous

hr continuous

lvef dichotomous

bnp categorical

afib continuous

bmi_cat categorical

Examples

```
data(ipdna_md)
## maybe str(ipdna_md)

#summary per study
by(ipdna_md, ipdna_md$centre, summary)
```

lbpmicox *Example dataset for psfmi_coxr function*

Description

10 imputed datasets

Usage

```
data(lbpmicox)
```

Format

A data frame with 2650 observations on the following 18 variables.

`Impnr` a numeric vector

`patnr` a numeric vector

`Status` dichotomous event

`Time` continuous follow up time variable

`Duration` continuous

`Previous` dichotomous

`Radiation` dichotomous

`Onset` dichotomous

`Age` continuous

`Tampascale` continuous

`Pain` continuous

`Function` continuous

`Satisfaction` categorical

`JobControl` continuous

`JobDemand` continuous

`Social` continuous

`Expectation` a numeric vector

`Expect_cat` categorical

Examples

```
data(lbpmicox)
## maybe str(lbpmicox)
```

`lbpmilr` *Example dataset for psfmi_lr function*

Description

10 imputed datasets

Usage

```
data(lbpmilr)
```

Format

A data frame with 1590 observations on the following 17 variables.

`Impnr` a numeric vector

`ID` a numeric vector

`Chronic` dichotomous

`Gender` dichotomous

`Carrying` categorical

`Pain` continuous

`Tampascale` continuous

`Function` continuous

`Radiation` dichotomous

`Age` continuous

`Smoking` dichotomous

`Satisfaction` categorical

`JobControl` continuous

`JobDemands` continuous

`SocialSupport` continuous

`Duration` continuous

`BMI` continuous

Examples

```
data(lbpmilr)
## maybe str(lbpmilr)
```

lbpmilr_dev *Example dataset for mivalext_lr function*

Description

1 development dataset

Usage

```
data(lbpmilr_dev)
```

Format

A data frame with 108 observations on the following 16 variables.

`ID` a numeric vector

`Chronic` dichotomous

`Gender` dichotomous

`Carrying` categorical

`Pain` continuous

`Tampascale` continuous

`Function` continuous

`Radiation` dichotomous

`Age` continuous

`Smoking` dichotomous

`Satisfaction` categorical

`JobControl` continuous

`JobDemands` continuous

`SocialSupport` continuous

`Duration` continuous

`BMI` continuous

Examples

```
data(lbpmilr_dev)
## maybe str(lbpmilr_dev)
```

lbp_orig *Example dataset for psfmi_perform function, method boot_MI*

Description

Original dataset with missing values

Usage

```
data(lbp_orig)
```

Format

A data frame with 159 observations on the following 15 variables.

`Chronic` dichotomous

`Gender` dichotomous

`Carrying` categorical

`Pain` continuous

`Tampascale` continuous

`Function` continuous

`Radiation` dichotomous

`Age` continuous

`Smoking` dichotomous

`Satisfaction` categorical

`JobControl` continuous

`JobDemands` continuous

`SocialSupport` continuous

`Duration` continuous

`BMI` continuous

Examples

```
data(lbp_orig)
## maybe str(lbp_orig)
```

mivalext_lr *External Validation of logistic prediction models in multiply imputed datasets*

Description

`mivalext_lr` External validation of logistic prediction models

Usage

```
mivalext_lr(
  data.val = NULL,
  data.orig = NULL,
  nimp = 5,
  impvar = NULL,
  Outcome,
  predictors = NULL,
  lp.orig = NULL,
  cal.plot = FALSE,
  plot.indiv = FALSE,
  val.check = FALSE,
  g = 10
)
```

Arguments

`data.val`	Data frame with stacked multiply imputed validation datasets. The original dataset that contains missing values must be excluded from the dataset. The imputed datasets must be distinguished by an imputation variable, specified under impvar, and starting by 1.
`data.orig`	A single data frame containing the original dataset that was used to develop the model. Used to estimate the original regression coefficients in case lp.orig is not provided.
`nimp`	A numerical scalar. Number of imputed datasets. Default is 5.
`impvar`	A character vector. Name of the variable that distinguishes the imputed datasets.
`Outcome`	Character vector containing the name of the outcome variable.
`predictors`	Character vector with the names of the predictor variables of the model that is validated.
`lp.orig`	Numeric vector of the original coefficient values that are externally validated.
`cal.plot`	If TRUE a calibration plot is generated. Default is FALSE.
`plot.indiv`	If TRUE calibration plots of each imputed dataset are generated. Default is FALSE.
`val.check`	logical vector. If TRUE the names of the predictors of the LP are provided and can be used as information for the order of the coefficient values as input for lp.orig. If FALSE (default) validation procedure is executed with coefficient values fitted in the order as used under lp.orig.

g A numerical scalar. Number of groups for the Hosmer and Lemeshow test. Default is 10.

Details

The following information of the externally validated model is provided: `ROC` pooled ROC curve (median and back transformed after pooling log transformed ROC curves), `R2_fixed` and `R2_calibr` pooled Nagelkerke R-Square value (median and back transformed after pooling Fisher transformed values), `HLtest` pooled Hosmer and Lemeshow Test (using miceadds package), `coef_pooled` pooled coefficients when model is freely estimated in imputed datasets and `LP_pooled_ext` the pooled linear predictor (LP), after the externally validated LP is estimated in each imputed dataset (provides information about miscalibration in intercept and slope). In addition information is provided about `nimp`, `impvar`, `Outcome`, `val_ckeck`, `g` and `coef_check`. When the external validation is very poor, the R2 fixed can become negative due to the poor fit of the model in the external dataset (in that case you may report a R2 of zero).

Value

A `mivalext_lr` object from which the following objects can be extracted: ROC results as `ROC`, R squared results (fixed and calibrated) as `R2 (fixed)` and `R2 (calibr)`, Hosmer and Lemeshow test as `HL_test`, coefficients pooled as `coef_pooled`, linear predictor pooled as `LP_pooled ext`, and `Outcome`, `nimp`, `impvar`, `val.check`, `g` and `coef.check`.

References

F. Harrell. Regression Modeling Strategies. With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis. Springer, New York, NY, 2015.

Van Buuren S. (2018). Flexible Imputation of Missing Data. 2nd Edition. Chapman & Hall/CRC Interdisciplinary Statistics. Boca Raton.

http://missingdatasolutions.rbind.io/

Examples

```
mivalext_lr(data.val=lbpmilr, nimp=10, impvar="Impnr", Outcome="Chronic",
predictors=c("Gender", "factor(Carrying)", "Function", "Tampascale",  "Age"),
lp.orig=c(-9.2, -0.34, 0.92, 1.5, 0.5, 0.26, -0.02),
cal.plot=TRUE, plot.indiv=TRUE, val.check = TRUE)

mivalext_lr(data.val=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
predictors=c("Gender", "factor(Carrying)", "Function", "Tampascale", "Age"),
lp.orig=c(-9.2, -0.34, 0.92, 1.1, -0.05, 0.26, -0.02),
cal.plot=TRUE, plot.indiv=TRUE, val.check = FALSE)
```

pool_intadj — *Provides pooled adjusted intercept after shrinkage of pooled coefficients in multiply imputed datasets*

Description

`pool_intadj` Provides pooled adjusted intercept after shrinkage of the pooled coefficients in multiply imputed datasets for models selected with the `psfmi_lr` function and internally validated with the `psfmi_perform` function.

Usage

```
pool_intadj(pobj, shrinkage_factor)
```

Arguments

`pobj` An object of class `smodsmi` (selected models in multiply imputed datasets), produced by a previous call to `psfmi_lr`.

`shrinkage_factor` A numerical scalar. Shrinkage factor value as a result of internal validation with the `psfmi_perform` function.

Details

The function provides the pooled adjusted intercept after shrinkage of pooled regression coefficients in multiply imputed datasets. The function is only available for logistic regression models without random effects.

Value

A `pool_intadj` object from which the following objects can be extracted: `int_adj`, the adjusted intercept value, `coef_shrink_pooled`, the pooled regression coefficients after shrinkage, `coef_orig_pooled`, the (original) pooled regression coefficients before shrinkage and `nimp`, the number of imputed datasets.

References

F. Harrell. Regression Modeling Strategies. With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis (2nd edition). Springer, New York, NY, 2015.

EW. Steyerberg (2019). Clinical Prediction MOdels. A Practical Approach to Development, Validation, and Updating (2nd edition). Springer Nature Switzerland AG.

http://missingdatasolutions.rbind.io/

Examples

```
 res_psfmi <- psfmi_lr(data=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
           predictors=c("Gender", "Pain","Tampascale","Smoking","Function",
           "Radiation", "Age"), p.crit = 1, method="D1")
 res_psfmi$RR_Model

## Not run:
 set.seed(100)
 res_val <- psfmi_perform(res_psfmi, method = "MI_boot", nboot=10,
   int_val = TRUE, p.crit=1, cal.plot=FALSE, plot.indiv=FALSE)
 res_val$intval

 res <- pool_intadj(res_psfmi, shrinkage_factor = 0.9774058)
 res$int_adj
 res$coef_shrink_pooled

## End(Not run)
```

pool_performance   *Pooling performance measures over multiply imputed datasets*

Description

pool_performance Pooling performance measures

Usage

```
pool_performance(
  data,
  nimp,
  impvar,
  Outcome,
  predictors,
  cal.plot,
  plot.indiv,
  groups_cal = 10
)
```

Arguments

data — Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset.

nimp — A numerical scalar. Number of imputed datasets. Default is 5.

impvar — A character vector. Name of the variable that distinguishes the imputed datasets.

Outcome — Character vector containing the name of the outcome variable.

`predictors` Character vector with the names of the predictor variables as used in the formula part of an glm object.

`cal.plot` If TRUE a calibration plot is generated. Default is FALSE. Can be used in combination with int_val = FALSE.

`plot.indiv` If TRUE calibration plots for each separate imputed dataset are generated, otherwise all calibration plots are plotted in one figure.

`groups_cal` A numerical scalar. Number of groups used on the calibration plot. Default is 10. If the range of predicted probabilities is too low 5 groups can be chosen.

Examples

```
pool_performance(data=lbpmilr, nimp=5, impvar="Impnr",
Outcome = "Chronic", predictors = c("Gender", "Pain", "rcs(Tampascale, 3)",
"Smoking", "Function", "Radiation", "Age", "factor(Carrying)"),
cal.plot=TRUE, plot.indiv=FALSE)
```

`psfmi_coxr` *Pooling and predictor selection function for Cox regression models in multiply imputed datasets*

Description

`psfmi_coxr` Pooling and backward selection for Cox regression models in multiply imputed datasets using different selection methods.

Usage

```
psfmi_coxr(
  data,
  nimp = 5,
  impvar = NULL,
  time,
  status,
  predictors = NULL,
  p.crit = 1,
  cat.predictors = NULL,
  spline.predictors = NULL,
  int.predictors = NULL,
  keep.predictors = NULL,
  knots = NULL,
  method = "RR",
  print.method = FALSE
)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset. The imputed datasets must be distinguished by an imputation variable, specified under impvar, and starting by 1.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`time` Follow up time.

`status` The status variable, normally 0=censoring, 1=event.

`predictors` Character vector with the names of the predictor variables. At least one predictor variable has to be defined.

`p.crit` A numerical scalar. P-value selection criterium.

`cat.predictors` A single string or a vector of strings to define the categorical variables. Default is NULL categorical predictors.

`spline.predictors` A single string or a vector of strings to define the (restricted cubic) spline variables. Default is NULL spline predictors.

`int.predictors` A single string or a vector of strings with the names of the variables that form an interaction pair, separated by a “:” symbol.

`keep.predictors` A single string or a vector of strings including the variables that are forced in the model during predictor selection. Categorical and interaction variables are allowed. See details.

`knots` A numerical vector that defines the number of knots for each spline predictor separately.

`method` A character vector to indicate the pooling method for p-values to pool the total model or used during predictor selection. This can be "D1", "D2", or "MPR". See details for more information.

`print.method` logical vector. If TRUE full matrix with p-values of all variables according to chosen method (under method) is shown. If FALSE (default) p-value for categorical variables according to method are shown and for continuous and dichotomous predictors Rubin’s Rules are used.

Details

The basic pooling procedure to derive pooled coefficients, standard errors, 95 confidence intervals and p-values is Rubin’s Rules (RR). Specific procedures are available to derive pooled p-values for categorical (> 2 categories) and spline variables. print.method allows to choose between these pooling methods that are: “D1” is pooling of the total covariance matrix, ”D2” is pooling of Chi-square values, and “MPR” is pooling of median p-values (MPR rule). Spline regression coefficients are defined by using the rcs function for restricted cubic splines of the rms package of Frank Harrell. A minimum number of 3 knots as defined under knots is needed.

Value

An object of class `smodsmi` (selected models in multiply imputed datasets) from which the following objects can be extracted: imputed datasets as `data`, selected pooled model as `RR_model`, pooled p-values according to pooling method as `multiparm`, predictors included at each selection step as `predictors_in`, predictors excluded at each step as `predictors_out`, and `impvar`, `nimp`, `time`, `status`, `method`, `p.crit`, `predictors`, `cat.predictors`, `keep.predictors`, `int.predictors`, `spline.predictors`, `knots`, `print.method`, `call`, `model_type` and `predictors_final` for names of predictors in final selection step, `fit.formula` is the regression formula of start model and `predictors_initial` for names of predictors in start model.

References

Eekhout I, van de Wiel MA, Heymans MW. Methods for significance testing of categorical covariates in logistic regression models after multiple imputation: power and applicability analysis. BMC Med Res Methodol. 2017;17(1):129.

Enders CK (2010). Applied missing data analysis. New York: The Guilford Press.

van de Wiel MA, Berkhof J, van Wieringen WN. Testing the prediction error difference between 2 predictors. Biostatistics. 2009;10:550-60.

Marshall A, Altman DG, Holder RL, Royston P. Combining estimates of interest in prognostic modelling studies after multiple imputation: current practice and guidelines. BMC Med Res Methodol. 2009;9:57.

Van Buuren S. (2018). Flexible Imputation of Missing Data. 2nd Edition. Chapman & Hall/CRC Interdisciplinary Statistics. Boca Raton.

http://missingdatasolutions.rbind.io/

Examples

```
  pool_coxr <- psfmi_coxr(data=lbpmicox, nimp=5, impvar="Impnr", time="Time",
  status="Status", predictors=c("Duration", "Radiation", "Onset"), p.crit=1,
  method="D1", cat.predictors=c("Expect_cat"))
  pool_coxr$RR_Model
  pool_coxr$multiparm

 ## Not run:
  pool_coxr <- psfmi_coxr(data=lbpmicox, nimp=5, impvar="Impnr", time="Time",
  status="Status", predictors=c("Previous",  "Radiation", "Onset",
  "Function", "Tampascale" ), p.crit=0.05, cat.predictors=c("Expect_cat"),
  int.predictors=c("Tampascale:Radiation",
  "Expect_cat:Tampascale"), keep.predictors = "Tampascale", method="D2")
  pool_coxr$RR_Model
  pool_coxr$multiparm
  pool_coxr$predictors_in

## End(Not run)
```

psfmi_D3 *Meng & Rubin pooling method called by psfmi_lr*

Description

psfmi_D3 Function to pool using Meng & Rubin pooling method

Usage

```
psfmi_D3(data, nimp, impvar, Outcome, P, p.crit, print.method)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`Outcome` Character vector containing the name of the outcome variable.

`P` Character vector with the names of the predictor variables. At least one predictor variable has to be defined.

`p.crit` A numerical scalar. P-value selection criterium.

`print.method` logical vector. If TRUE full matrix with p-values of all variables according to chosen method (under method) is shown. If FALSE (default) p-value for categorical variables according to method are shown and for continuous and dichotomous predictors Rubin's Rules are used

Examples

```
psfmi_D3(data=lbpmilr, nimp=5, impvar="Impnr",
P=c("Gender", "Smoking", "Function", "JobControl"),
Outcome="Chronic", print.method = FALSE)
```

psfmi_lr *Pooling and Predictor selection function for Logistic regression models in multiply imputed datasets*

Description

psfmi_lr Pooling and backward selection for Logistic regression prediction models in multiply imputed datasets using different selection methods.

Usage

```
psfmi_lr(
  data,
  nimp = 5,
  impvar = NULL,
  Outcome,
  predictors = NULL,
  p.crit = 1,
  cat.predictors = NULL,
  spline.predictors = NULL,
  int.predictors = NULL,
  keep.predictors = NULL,
  knots = NULL,
  method = "RR",
  print.method = FALSE
)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset. The imputed datasets must be distinguished by an imputation variable, specified under impvar, and starting by 1.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`Outcome` Character vector containing the name of the outcome variable.

`predictors` Character vector with the names of the predictor variables. At least one predictor variable has to be defined.

`p.crit` A numerical scalar. P-value selection criterium. A value of 1 provides the pooled model without selection.

`cat.predictors` A single string or a vector of strings to define the categorical variables. Default is NULL categorical predictors.

`spline.predictors` A single string or a vector of strings to define the (restricted cubic) spline variables. Default is NULL spline predictors. See details.

`int.predictors` A single string or a vector of strings with the names of the variables that form an interaction pair, separated by a “:” symbol.

`keep.predictors` A single string or a vector of strings including the variables that are forced in the model during predictor selection. Categorical and interaction variables are allowed.

`knots` A numerical vector that defines the number of knots for each spline predictor separately.

`method` A character vector to indicate the pooling method for p-values to pool the total model or used during predictor selection. This can be "D1", "D2", "D3" or "MPR". See details for more information.

`print.method` logical vector. If TRUE full matrix with p-values of all variables according to chosen method (under method) is shown. If FALSE (default) p-value for categorical variables according to method are shown and for continuous and dichotomous predictors Rubin's Rules are used.

Details

The basic pooling procedure to derive pooled coefficients, standard errors, 95 confidence intervals and p-values is Rubin's Rules (RR). Specific procedures are available to derive pooled p-values for categorical (> 2 categories) and spline variables. print.method allows to choose between the pooling methods: "D1" is pooling of the total covariance matrix, "D2" is pooling of Chi-square values, "D3" is pooling Likelihood ratio statistics (method of Meng and Rubin) and "MPR" is pooling of median p-values (MPR rule). Spline regression coefficients are defined by using the rcs function for restricted cubic splines of the rms package. A minimum number of 3 knots as defined under knots is required.

Value

An object of class `smodsmi` (selected models in multiply imputed datasets) from which the following objects can be extracted: imputed datasets as `data`, selected pooled model as `RR_model`, pooled p-values according to pooling method as `multiparm`, predictors included at each selection step as `predictors_in`, predictors excluded at each step as `predictors_out`, and `impvar`, `nimp`, `Outcome`, `method`, `p.crit`, `predictors`, `cat.predictors`, `keep.predictors`, `int.predictors`, `spline.predictors`, `knots`, `print.method`, `call`, `model_type`, `predictors_final` for names of predictors in final selection step, `fit.formula` is the regression formula of start model and `predictors_initial` for names of predictors in start model.

References

Eekhout I, van de Wiel MA, Heymans MW. Methods for significance testing of categorical covariates in logistic regression models after multiple imputation: power and applicability analysis. BMC Med Res Methodol. 2017;17(1):129.

Enders CK (2010). Applied missing data analysis. New York: The Guilford Press.

Meng X-L, Rubin DB. Performing likelihood ratio tests with multiply-imputed data sets. Biometrika.1992;79:103-11.

van de Wiel MA, Berkhof J, van Wieringen WN. Testing the prediction error difference between 2 predictors. Biostatistics. 2009;10:550-60.

Marshall A, Altman DG, Holder RL, Royston P. Combining estimates of interest in prognostic modelling studies after multiple imputation: current practice and guidelines. BMC Med Res Methodol. 2009;9:57.

Van Buuren S. (2018). Flexible Imputation of Missing Data. 2nd Edition. Chapman & Hall/CRC Interdisciplinary Statistics. Boca Raton.

EW. Steyerberg (2019). Clinical Prediction MOdels. A Practical Approach to Development, Validation, and Updating (2nd edition). Springer Nature Switzerland AG.

http://missingdatasolutions.rbind.io/

Examples

```
pool_lr <- psfmi_lr(data=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
predictors=c("Gender", "Smoking", "Function", "JobControl",
"JobDemands", "SocialSupport"), method="D1")
pool_lr$RR_Model
pool_lr$multiparm

pool_lr <- psfmi_lr(data=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
predictors=c("Gender", "Smoking", "Function", "JobControl",
"JobDemands", "SocialSupport"), p.crit = 0.05, method="D1")
pool_lr$RR_Model
pool_lr$multiparm
pool_lr$predictors_in
```

---

psfmi_mm *Pooling and Predictor selection function for multilevel models in multiply imputed datasets*

Description

psfmi_mm Pooling and backward selection for 2 level (generalized) linear mixed models in multiply imputed datasets using different selection methods.

Usage

```
psfmi_mm(
  data,
  nimp = 5,
  impvar = NULL,
  clusvar = NULL,
  Outcome,
  predictors = NULL,
  random.eff = NULL,
  family = "linear",
  p.crit = 1,
  cat.predictors = NULL,
  spline.predictors = NULL,
  int.predictors = NULL,
  keep.predictors = NULL,
  knots = NULL,
  method = "RR",
  print.method = FALSE
)
```

Arguments

`data` Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset. The imputed datasets must be distinguished by an imputation variable, specified under impvar, and starting by 1 and the clusters should be distinguished by a cluster variable, specified under clusvar.

`nimp` A numerical scalar. Number of imputed datasets. Default is 5.

`impvar` A character vector. Name of the variable that distinguishes the imputed datasets.

`clusvar` A character vector. Name of the variable that distinguishes the clusters.

`Outcome` Character vector containing the name of the outcome variable.

`predictors` Character vector with the names of the predictor variables. At least one predictor variable has to be defined.

`random.eff` Character vector to specify the random effects as used by the `lmer` and `glmer` functions of the `lme4` package.

`family` Character vector to specify the type of model, "linear" is used to call the `lmer` function and "binomial" is used to call the `glmer` function of the `lme4` package. See details for more information.

`p.crit` A numerical scalar. P-value selection criterium. A value of 1 provides the pooled model without selection.

`cat.predictors` A single string or a vector of strings to define the categorical variables. Default is NULL categorical predictors.

`spline.predictors`
A single string or a vector of strings to define the (restricted cubic) spline variables. Default is NULL spline predictors. See details.

`int.predictors` A single string or a vector of strings with the names of the variables that form an interaction pair, separated by a “:” symbol.

`keep.predictors`
A single string or a vector of strings including the variables that are forced in the model during predictor selection. Categorical and interaction variables are allowed.

`knots` A numerical vector that defines the number of knots for each spline predictor separately.

`method` A character vector to indicate the pooling method for p-values to pool the total model or used during predictor selection. This can be "D1", "D2", "D3" or "MPR". See details for more information.

`print.method` logical vector. If TRUE full matrix with p-values of all variables according to chosen method (under method) is shown. If FALSE (default) p-value for categorical variables according to method are shown and for continuous and dichotomous predictors Rubin’s Rules are used.

Details

The basic pooling procedure to derive pooled coefficients, standard errors, 95 confidence intervals and p-values is Rubin’s Rules (RR). Specific procedures are available to derive pooled p-values for categorical (> 2 categories) and spline variables. print.method allows to choose between the pooling methods: D1, D2 and D3 and MPR for pooling of median p-values (MPR rule). The D1, D2 and D3 methods are called from the package `mitml`. For Logistic multilevel models (that are estimated using the `glmer` function), the D3 method is not yet available. Spline regression coefficients are defined by using the rcs function for restricted cubic splines of the rms package. A minimum number of 3 knots as defined under knots is required.

Value

An object of class `smodsmi` (selected models in multiply imputed datasets) from which the following objects can be extracted: imputed datasets as `data`, selected pooled model as `RR_model`, pooled p-values according to pooling method as `multiparm`, random effects as `random.eff`, predictors included at each selection step as `predictors_in`, predictors excluded at each step as `predictors_out`, and `family`, `impvar`, `clusvar`, `nimp`, `Outcome`, `method`, `p.crit`, `predictors`, `cat.predictors`, `keep.predictors`, `int.predictors`, `spline.predictors`, `knots`, `print.method`, `model_type`, `call`, `predictors_final` for names of predictors in final step and `fit.formula` is the regression formula of start model.

References

Eekhout I, van de Wiel MA, Heymans MW. Methods for significance testing of categorical covariates in logistic regression models after multiple imputation: power and applicability analysis. BMC Med Res Methodol. 2017;17(1):129.

Enders CK (2010). Applied missing data analysis. New York: The Guilford Press.

Meng X-L, Rubin DB. Performing likelihood ratio tests with multiply-imputed data sets. Biometrika.1992;79:103-11.

van de Wiel MA, Berkhof J, van Wieringen WN. Testing the prediction error difference between 2 predictors. Biostatistics. 2009;10:550-60.

mitml package https://cran.r-project.org/web/packages/mitml/index.html

Van Buuren S. (2018). Flexible Imputation of Missing Data. 2nd Edition. Chapman & Hall/CRC Interdisciplinary Statistics. Boca Raton.

http://missingdatasolutions.rbind.io/

Examples

```
## Not run:
  pool_mm <- psfmi_mm(data=ipdna_md, nimp=5, impvar=".imp", family="linear",
  predictors=c("gender", "afib", "sbp"), clusvar = "centre",
  random.eff="( 1 | centre)", Outcome="dbp", cat.predictors = "bmi_cat",
  p.crit=0.15, method="D1", print.method = FALSE)
  pool_mm$RR_Model
  pool_mm$multiparm

## End(Not run)
```

psfmi_mm_multiparm *Multiparameter pooling methods called by psfmi_mm*

Description

psfmi_mm_multiparm Function to pool according to D1, D2 and D3 methods

Usage

```
psfmi_mm_multiparm(
  data,
  nimp,
  impvar,
  Outcome,
  P,
  p.crit,
  family,
  random.eff,
  method,
  print.method
)
```

Arguments

data	Data frame with stacked multiple imputed datasets. The original dataset that contains missing values must be excluded from the dataset. The imputed datasets must be distinguished by an imputation variable, specified under impvar, and starting by 1 and the clusters should be distinguished by a cluster variable, specified under clusvar.
nimp	A numerical scalar. Number of imputed datasets. Default is 5.
impvar	A character vector. Name of the variable that distinguishes the imputed datasets.
Outcome	Character vector containing the name of the outcome variable.
P	Character vector with the names of the predictor variables. At least one predictor variable has to be defined.
p.crit	A numerical scalar. P-value selection criterium. A value of 1 provides the pooled model without selection.
family	Character vector to specify the type of model, "linear" is used to call the lmer function and "binomial" is used to call the glmer function of the lme4 package. See details for more information.
random.eff	Character vector to specify the random effects as used by the lmer and glmer functions of the lme4 package.
method	A character vector to indicate the pooling method for p-values to pool the total model or used during predictor selection. This can be "D1", "D2", "D3" or "MPR". See details for more information.

`print.method` logical vector. If TRUE full matrix with p-values of all variables according to chosen method (under method) is shown. If FALSE (default) p-value for categorical variables according to method are shown and for continuous and dichotomous predictors Rubin's Rules are used.

Examples

```
## Not run:
 psfmi_mm_multiparm(data=ipdna_md, nimp=5, impvar=".imp", family="linear",
 P=c("gender", "bnp", "dbp", "lvef", "bmi_cat"),
 random.eff="( 1 | centre)", Outcome="sbp",
 p.crit=0.05, method="D1", print.method = FALSE)

## End(Not run)
```

`psfmi_perform` *Evaluate performance of logistic regression models in Multiply Imputed datasets*

Description

`psfmi_perform` Evaluate Performance of logistic regression models selected with the `psfmi_lr` function of the `psfmi` package.

Usage

```
psfmi_perform(
  pobj,
  data_orig = NULL,
  nboot = 10,
  int_val = FALSE,
  method = NULL,
  nimp_boot_MI = NULL,
  p.crit = 1,
  mice_method = NULL,
  mice_niter = 10,
  mice_seed = NA,
  predictorMatrix = NULL,
  cal.plot = FALSE,
  plot.indiv = FALSE,
  groups_cal = 10
)
```

Arguments

`pobj` An object of class `smodsmi` (selected models in multiply imputed datasets), produced by a previous call to `psfmi_lr`.

`data_orig` dataframe of original dataset that contains missing data for method boot_MI

`nboot` The number of bootstrap resamples, default is 10.

`int_val` If TRUE internal validation is conducted in multiply imputed datasets. See `method` for methods that can be used.

`method` Methods for internal validation in multiply imputed datasets. Choose MI_boot for bootstrapping in each imputed dataset and boot_MI for multiple imputation in each bootstrap sample. To use the second method data_orig has to be specified. The first method is faster. See details for more information.

`nimp_boot_MI` Numerical scalar. Number of imputed datasets for method boot_MI. When not defined, the number of multiply imputed datasets is used of the previous call to the function `psfmi_lr`.

`p.crit` A numerical scalar. P-value selection criterium used for backward selection during internal validation. When set at 1, pooling and internal validation is done without backward selection.

`mice_method` The Multiple Imputation method used for each predictor with missing values. For Multiple Imputation the `mice` package is used. See that package for more information.

`mice_niter` Numerical scalar. Default is 10. The number of iterations in Multiple Imputation. See the `mice` package for more information.

`mice_seed` Numerical scalar. Default is random number generator initializeb by computer via set.seed().

`predictorMatrix`
A numeric matrix of nrow(data) rows and ncol(data) columns, containing 0/1 data specifying the imputation models used to impute the predictors with missing data. Default is that each variable is used to impute other variables. See the `mice` package for more information.

`cal.plot` If TRUE a calibration plot is generated. Default is FALSE. Can be used in combination with int_val = FALSE.

`plot.indiv` If TRUE calibration plots for each separate imputed dataset are generated, otherwise all calibration plots are plotted in one figure.

`groups_cal` A numerical scalar. Number of groups used on the calibration plot. Default is 10. If the range of predicted probabilities is too low 5 groups can be chosen.

Details

For internal validation two methods can be used, MI_boot and boot_MI. MI_boot draws for each bootstrap step the same cases in all imputed datasets. With boot_MI first bootstrap samples are drawn from the original dataset with missing values and than multiple imputation is applied. For multiple imputation the `mice` function from the `mice` package is used. It is recommended to use a minumum of 100 bootstrap samples, which may take some time. The method boot_MI is more time consuming than MI_boot.

Value

A psfmi_perform object from which the following objects can be extracted: res_boot, result of pooled performance (in multiply imputed datasets) at each bootstrap step of ROC app (pooled ROC), ROC test (pooled ROC after bootstrap model is applied in original multiply imputed datasets), same for R2 app (Nagelkerke's R2), R2 test, Brier app and Brier test. Information is also provided about testing the Calibration slope at each bootstrap step as interc test and Slope test. The performance measures are pooled by a call to the function pool_performance. Another object that can be extracted is intval, with information of the AUC, R2, Brier score and Calibration slope averaged over the bootstrap samples, in terms of: Orig (original datasets), Apparent (models applied in bootstrap samples), Test (bootstrap models are applied in original datasets), Optimism (difference between apparent and test) and Corrected (original corrected for optimism).

References

Heymans MW, van Buuren S, Knol DL, van Mechelen W, de Vet HC. Variable selection under multiple imputation using the bootstrap in a prognostic study. BMC Med Res Methodol. 2007(13);7:33.

F. Harrell. Regression Modeling Strategies. With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis (2nd edition). Springer, New York, NY, 2015.

Van Buuren S. (2018). Flexible Imputation of Missing Data. 2nd Edition. Chapman & Hall/CRC Interdisciplinary Statistics. Boca Raton.

Harel, O. (2009). The estimation of R2 and adjusted R2 in incomplete data sets using multiple imputation. Journal of Applied Statistics, 36(10), 1109-1118.

Musoro JZ, Zwinderman AH, Puhan MA, ter Riet G, Geskus RB. Validation of prediction models based on lasso regression with multiply imputed data. BMC Med Res Methodol. 2014;14:116.

Wahl S, Boulesteix AL, Zierer A, Thorand B, van de Wiel MA. Assessment of predictive performance in incomplete data by combining internal validation and multiple imputation. BMC Med Res Methodol. 2016;16(1):144.

EW. Steyerberg (2019). Clinical Prediction MOdels. A Practical Approach to Development, Validation, and Updating (2nd edition). Springer Nature Switzerland AG.

http://missingdatasolutions.rbind.io/

Examples

```
res_psfmi <- psfmi_lr(data=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
  predictors=c("Gender", "Pain","Tampascale","Smoking","Function", "Radiation",
  "Age"), p.crit = 1, method="D1")

res_val <- psfmi_perform(res_psfmi, int_val = FALSE, p.crit=1, cal.plot=TRUE,
  plot.indiv=FALSE)
  res_val

## Not run:
set.seed(200)
res_val <- psfmi_perform(res_psfmi, int_val = TRUE, p.crit=0.05, nboot = 10,
 method = "MI_boot", cal.plot=FALSE, plot.indiv=FALSE)
res_val$intval
```

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

psfmi_stab *Function to evaluate bootstrap predictor and model stability in multiply imputed datasets.*

Description

psfmi_stab Stability analysis of predictors and prediction models selected with the psfmi_lr, psfmi_coxr or psfmi_mm functions of the psfmi package.

Usage

```
psfmi_stab(pobj, boot_method = NULL, nboot = 20)
```

Arguments

pobj	An object of class smodsmi (selected models in multiply imputed datasets), produced by a previous call to psfmi_lr, psfmi_coxr or psfmi_mm.
boot_method	A single string to define the bootstrap method. Use "single" after a call to psfmi_lr and psfmi_coxr and "cluster" after a call to psfmi_mm.
nboot	A numerical scalar. Number of bootstrap samples to evaluate the stability. Default is 20.

Details

The function evaluates predictor selection frequency in stratified or cluster bootstrap samples. The stratification factor is the variable that separates the imputed datasets. It uses as input an object of class smodsmi as a result of a previous call to the psfmi_lr, psfmi_coxr or psfmi_mm functions. In combination with the psfmi_mm function a cluster bootstrap method is used where bootstrapping is used on the level of the clusters only.

Value

A psfmi_stab object from which the following objects can be extracted: bootstrap inclusion (selection) frequency of each predictor bif, total number each predictor is included in the bootstrap samples as bif_total, percentage a predictor is selected in each bootstrap sample as bif_perc and number of times a prediction model is selected in the bootstrap samples as model_stab.

References

Heymans MW, van Buuren S. et al. Variable selection under multiple imputation using the bootstrap in a prognostic study. BMC Med Res Methodol. 2007;13:7-33.

Eekhout I, van de Wiel MA, Heymans MW. Methods for significance testing of categorical covariates in logistic regression models after multiple imputation: power and applicability analysis. BMC Med Res Methodol. 2017;17(1):129.

Sauerbrei W, Schumacher M. A bootstrap resampling procedure for model building: application to the Cox regression model. Stat Med. 1992;11:2093–109.

Royston P, Sauerbrei W (2008) Multivariable model-building – a pragmatic approach to regression analysis based on fractional polynomials for modelling continuous variables. (2008). Chapter 8, Model Stability. Wiley, Chichester

Heinze G, Wallisch C, Dunkler D. Variable selection - A review and recommendations for the practicing statistician. Biom J. 2018;60(3):431-449.

http://missingdatasolutions.rbind.io/

Examples

```
 pool_lr <- psfmi_lr(data=lbpmilr, nimp=5, impvar="Impnr", Outcome="Chronic",
                  predictors=c("Gender", "Smoking",  "JobControl", "JobDemands",
                  "Age", "Radiation", "SocialSupport", "Function"),
                  cat.predictors = c("Carrying"), p.crit =0.157, method="D1")
 pool_lr$RR_Model
 pool_lr$multiparm

## Not run:
 stab_res <- psfmi_stab(pool_lr, boot_method = "single", nboot=50)
 stab_res$bif
 stab_res$bif_perc
 stab_res$model_stab

## End(Not run)
```

Index