

Package ‘MAMI’

October 17, 2016

Type Package

Title Model Averaging and Model Selection after Multiple Imputation

Version 0.9.4

Date 13 October 2016

Author Michael Schomaker

Maintainer Michael Schomaker <michael.schomaker@uct.ac.za>

Description Performs model selection/averaging on multiply imputed datasets and combines the resulting estimates. The package also provides access to less frequently used model averaging techniques and offers integrated bootstrap estimation.

Depends R (>= 3.3.1), lasso2, quadprog, corpcor, boot, MASS, survival, MuMIn, zoo, lme4, Amelia, BMA

License GPL-2

LazyData yes

R topics documented:

mami-package	2
HIV	2
lae	3
mami	5
mma	9
plot.lae	10
plot.mami	11
print.lae	12
print.mami	13
print.mma	13
summary.lae	14
summary.mami	15

Index	16
--------------	-----------

mami-package	<i>Model averaging (and model selection) after multiple imputation</i>
--------------	--

Description

Performs model selection/averaging on multiply imputed data sets and combines the resulting estimates. The package also provides access to less frequently used model averaging techniques and offers integrated bootstrap estimation. The main function is "mami".

Details

Package: mami
 Type: Package
 Version: 0.9.4
 Date: 2016-10-13
 License: GPL-2
 Depends: R (>= 3.3.1), lasso2, quadprog, corpcor, boot, MASS, survival, MuMIn, zoo, lme4, Amelia, BMA

Author(s)

Michael Schomaker

Maintainer: Michael Schomaker <michael.schomaker@uct.ac.za>

References

Schomaker, M., Heumann, C. (2014) *Model Selection and Model Averaging after Multiple Imputation*, Computational Statistics & Data Analysis, 71:758-770

HIV	<i>HIV treatment data</i>
-----	---------------------------

Description

A hypothetical, simulated dataset which provides typical data from HIV treatment research.

Usage

data(HIV)

Format

A data frame with 5000 observations on the following 14 variables:

patient The patient's id number

hospital The treatment facility

futime Follow-up time of the respective patient
 dead Outcome at the end of follow-up: (1=dead, 0=alive)
 sex The patients's sex (1=female, 0=male)
 age The patient's age
 cd4 The patient's baseline CD4 cell count
 cd4slope6 The patients CD4 slope (average CD4 gain/loss per week) at 6 month of follow up
 weight The patient's baseline weight (in kg)
 period Categorical variable specifying the period of treatment initiation: before 2004, between 2004 and 2007, or after 2007
 haem The patient's baseline haemoglobin level
 stage The patient's baseline WHO stage indicating disease severity, ordinal variable
 tb The patient's baseline tuberculosis status (1=yes,0=no)
 cm The patient's baseline cryptococcal meningitis status (1=yes,0=no)

References

Schomaker, M. et al. (2013) *Immune Recovery After Starting ART in HIV-infected Patients Presenting and not Presenting with Tuberculosis in South Africa*, Journal of Acquired Immune Deficiency Syndroms, 63:142-145

Hoffmann, C. et al. (2013) *CD4 Count Slope and Mortality in HIV-infected Patients on Antiretroviral Therapy: Multi-cohort Analysis from South Africa*, Journal of Acquired Immune Deficiency Syndroms, 63:34-41

Examples

```
data(HIV)
str(HIV)
```

 lae

Lasso Averaging Estimation

Description

Lasso (least absolute shrinkage and selection operator) estimation is performed and evaluated for different tuning parameter choices. To address tuning parameter selection uncertainty a weighted average of these estimators is calculated. The weight vector is chosen such that a k-fold cross validation criterion is minimized.

Usage

```
lae(X, ycol = 1, B.var = 50, nolambda = 100, kfold = 5, my.formula = NULL,
    standardize = TRUE, calc.variance = TRUE)
```

Arguments

<code>X</code>	A dataframe or matrix containing the data to be analyzed.
<code>ycol</code>	An integer specifying the column of the outcome variable.
<code>B.var</code>	An integer specifying the number of bootstrap replications to be used to estimate the standard error of the Lasso estimator.
<code>nolambda</code>	An integer specifying the number of candidate complexity parameters to consider.
<code>kfold</code>	An integer specifying the kfold cross validation criterion to (i) use for tuning parameter selection (ii) be minimized for Lasso averaging estimation.
<code>my.formula</code>	A formula specifying the full model.
<code>standardize</code>	A logical value specifying whether the covariate data should be standardized.
<code>calc.variance</code>	A logical value specifying whether the standard error of the estimates should be estimated at all (by means of bootstrapping).

Value

Returns an object of class 'lae':

<code>coefficients</code>	A matrix of coefficients and standard errors for Lasso averaging, Lasso selection, and OLS estimation.
<code>variable.importance</code>	A matrix containing the relative importance of each variable based on model averaging weights.
<code>sae.weights</code>	A vector containing the weights used for Lasso averaging.
<code>sel.weights</code>	A vector indicating the complexity parameter that was chosen for Lasso estimation based on k-fold cross validation.
<code>complexity.parameter</code>	A vector of the actual complexity parameter values used as candidate values for Lasso Averaging Estimation.

Author(s)

Michael Schomaker

References

Schomaker, M. (2012) *Shrinkage Averaging Estimation*, Statistical Papers, 53:1015-1034

See Also

[plot.lae](#) for visualizing the estimation results.

Examples

```
library(lasso2)
data(Prostate)
lae(Prostate, ycol=9, kfold=10, my.formula=lpsa~.)
```

mami

*Model averaging (and model selection) after multiple imputation***Description**

Performs model selection/averaging on multiply imputed data and combines the resulting estimates. The package also provides access to less frequently used model averaging techniques and offers integrated bootstrap estimation.

Usage

```
mami(X, method = c("criterion.average", "criterion.selection", "MMA", "LASSO/LAE", "WAML"), crit
B = 20, X.org = NULL, inference = c("standard", "+bootstrapping"), missing.data = c("imputed",
user.weights = NULL, candidate.models = c("all", "restricted", "very restricted"), model.family
"coxph"), add.factor = NULL, add.interaction = NULL, add.transformation = NULL, ycol = 1, CI =
use.stratum = NULL, report.exp = FALSE, print.time = FALSE, print.warnings = TRUE, ...)
```

Arguments

X	Either a list of multiply imputed datasets (each of them a dataframe), or an object of class 'amelia' created by Amelia II , or a single dataframe.
method	A character string specifying the model selection or model averaging technique: "criterion.average" for model averaging based on exponential AIC or BIC weights, "criterion.selection" for stepwise variable selection based on AIC or BIC, "MMA" for Mallows's model averaging (only linear model), "LASSO/LAE" for shrinkage estimation and averaging with LASSO (only linear model), and "WALS" for weighted-average least squares (linear and logistic model, available soon).
criterion	A character string specifying the model selection criterion used for criterion-based model selection/averaging; currently either "AIC" for Akaike's Information Criterion, "BIC" for the Bayes criterion of Schwarz, "BIC+" for the Bayes criterion of Schwarz with quicker model averaging based on the leaps algorithm of the BMA package, or CV for the cross validation error. CV can currently only be used for GLMs and may be more useful for model selection than model averaging.
B	An integer indicating the number of bootstrap replications to use (when inference = "+bootstrapping" is chosen).
X.org	A dataframe consisting of the original unimputed data (which needs to be specified when inference = "+bootstrapping" is chosen).
inference	A character string, either "standard" for applying multiple imputation combining rules upon the model selection/averaging estimates or "+bootstrapping" if additional bootstrap based inference is required. See reference section for more information.
missing.data	A character string, typically "imputed" when multiply imputed data are provided under X, or "CC" if a complete case analysis is desired, or "none" if there is no missing data.
var.remove	Either a vector of character strings or integers, specifying the variables or columns which are part of the data but not to be considered in the model selection/averaging procedure.

<code>user.weights</code>	A weight vector that is relevant to the analysis model.
<code>candidate.models</code>	A character string specifying whether for criterion based model selection/averaging all possible candidate models should be considered ("all"), or only candidate models with a limited amount of variables ("restricted", "very restricted").
<code>model.family</code>	A character string specifying the model family, either "gaussian" for linear regression models, "binomial" for logistic regression models, "poisson" for Poisson regression models, or "coxph" for Cox's proportional hazards models.
<code>add.factor</code>	Either a vector of character strings or integers, specifying the variables which should be treated as categorical/factors in the analysis. Variables which are already defined to be factors in the data are detected automatically and do not necessarily need to be specified with this option.
<code>add.interaction</code>	A list of vectors of either character strings or integers, specifying the variables which should be added as interactions in the analysis model.
<code>add.transformation</code>	A vector of character strings, specifying transformations of variables which should be added to the analysis models.
<code>ycol</code>	A vector or integer specifying the variable(s) or column(s) which should be treated as outcome variable.
<code>CI</code>	A value greater than 0 and less than 1 specifying the confidence of the confidence interval.
<code>kfold</code>	An integer specifying kfold cross validation; to be used when applying shrinkage estimation (method="LASSO/LAE") or criterion CV.
<code>id</code>	A character vector or integer specifying the variable or column to be used for a random intercept in the analysis model.
<code>use.stratum</code>	A character vector or integer specifying the variable used as a stratum in Cox regression analysis.
<code>report.exp</code>	A logical value specifying whether exponentiated coefficients should be reported or not.
<code>print.time</code>	A logical value specifying whether analysis time and anticipated estimation time for bootstrap estimation should be printed.
<code>print.warnings</code>	A logical value specifying whether warnings and any other output from the function should be printed or not.
<code>...</code>	Further arguments to be passed, i.e. for functions <code>lae</code> , <code>dredge</code> from the MuMIn package or <code>bic.glm</code> and <code>bic.surv</code> from the BMA package.

Details

Model selection/averaging will be performed on each imputed dataset. The results will be combined according to formulae (7)-(10) in Schomaker and Heumann (CSDA, 2014), see *References* below for more details. If `inference="+bootstrapping"` is chosen, then the procedure described in Table 1 will be performed in addition to standard MI inference. For longitudinal data (specified via `id`) the bootstrap is based on the subject/person/id level. To obtain insightful results from bootstrap estimation B should be large, at least $B > 200$ and `plot.mami` may be used.

Note that a variable will be formally selected if it is selected (by means of either model selection or averaging) in at least one imputed set of data, but its overall impact will depend on how often it is chosen. As a result, effects of variables which are not supported throughout imputed datasets and

candidate models will simply be less pronounced. Variable importance measures based on model averaging weights are calculated for each imputed dataset and will then be averaged.

If `method="criterion.average"` is chosen and the number of variables is large, then computation time might be a burden and obtaining results can even become unfeasible. The reason for this is that for criterion based model *averaging* the implementation of package **MuMIn** is used, which considers all possible candidate models, that is 2^p different candidate models for p parameters to estimate. If it is clear that only a subset of variables are relevant then the options `candidate.models="restricted/very restricted"` may be useful which essentially specifies that only up to a half/fourth of the provided variables can be added to a single candidate model. However, this option should be used with caution. Alternatively `criterion="BIC+"` can be used which utilizes efficient Bayesian Model Averaging based on the leaps algorithm of package "BMA". Also, one may consider a model selection or averaging strategy not implemented here and combine estimates "by hand" according to formulae (7)-(10) in the cited reference below.

The function provides access to linear, logistic, Poisson and Cox proportional hazard models; one may add a random intercept to each of these models with the `id` option. Other models are not supported yet. Variables used for the imputation model but not needed for the analysis model can be removed with option `var.remove`.

Value

Returns an object of class 'mami':

<code>coefficients</code>	A matrix of coefficients, standard errors and confidence intervals for model <i>averaging</i> estimators.
<code>coefficients.boot</code>	A matrix of coefficients and bootstrap results (confidence intervals, mean, standard error) for model <i>averaging</i> .
<code>coefficients.s</code>	A matrix of coefficients, standard errors and confidence intervals for model <i>selection</i> estimators.
<code>coefficients.boot.s</code>	A matrix of coefficients and bootstrap results (confidence intervals, mean, standard error) for model <i>selection</i> .
<code>variable.importance</code>	A vector containing the variable importance for each variable based on model averaging weights.
<code>boot.results</code>	A list of detailed estimation results for each bootstrap sample. The first list element refers to the results from model selection the second entry the results from model averaging.

Author(s)

Michael Schomaker

References

Schomaker, M., Heumann, C. (2014) *Model Selection and Model Averaging after Multiple Imputation*, Computational Statistics & Data Analysis, 71:758-770

See Also

[plot.mami](#) to visualize bootstrap results, [lae](#) and [mma](#) for model averaging techniques.

Examples

```
#####
# Example 1: Freetrade example from Amelia package #
#           Cross-Section-Time-Series Data         #
#           Linear and linear mixed model          #
#####
set.seed(24121980)
library(Amelia)
data(freetrade)
freetrade$pop <- log(freetrade$pop) # in line with original publication
freetrade_imp <- amelia(freetrade, ts = "year", cs = "country", noms="signed", polytime = 2,
                        intercs = TRUE, empri = 2)

# AIC based model averaging and model selection in a linear model after MI
# (with and without bootstrapping)
mami(freetrade_imp, method="criterion.selection", ycol="tariff",add.factor=c("country"))
mami(freetrade_imp, method="criterion.average", ycol="tariff",add.factor=c("country"))
m1 <- mami(freetrade_imp, method="criterion.selection", ycol="tariff",add.factor=c("country"),
           inference="+bootstrapping",B=25,X.org=freetrade,print.time=TRUE)
m1      # be patient with bootstrapping, increase B>=200 for better results
plot(m1, plots.p.page="4")

# For comparison: Mallows model averaging (MMA) and complete case analysis
mami(freetrade_imp, method="MMA", ycol="tariff",add.factor=c("country"))
mami(freetrade, method="criterion.selection", missing.data="CC", ycol="tariff",
     add.factor=c("country")) #Note the difference to imputed analysis (e.g. usheg)

# Use linear mixed model with random intercept for country as alternative, just specify "id" option
# Note: same procedure for random intercepts (frailty) in logistic, Poisson, or Cox Model
mami(freetrade_imp, ycol="tariff", id="country")

#####
# Example 2: HIV treatment data, linear model and Cox model #
#####

# Impute with Amelia
data(HIV)
HIV_imp <- amelia(HIV, m=5, idvars="patient",noms=c("hospital","sex","dead","tb","cm"),
                 ords=c("period","stage"),logs=c("fuptime","cd4"),
                 bounds=matrix(c(3,7,9,11,0,0,0,0,3000,5000,200,150),ncol=3,nrow=4))

# i) Cox PH model
# Model selection (with AIC) to select risk factors for the hazard of death,
# reported as hazard ratios
# Also: add transformations and interaction terms to candidate models
mami(HIV_imp, method="criterion.selection",model.family="coxph", ycol=c("fuptime","dead"),
     add.factor=c("hospital","stage","period"), add.transformation=c("cd4^2","age^2"),
     add.interaction=list(c("cd4","age")), report.exp=TRUE, var.remove=c("patient","cd4slope6"))
# Similar as above (= same but no hazard ratios reported, no interaction, hospitals as stratum),
# but with bootstrap CI and visualization of bootstrap distribution (be patient...its worth it)
m2 <- mami(HIV_imp, method="criterion.selection",model.family="coxph", inference="+bootstrapping",
           X.org=HIV, ycol=c("fuptime","dead"), add.factor=c("stage","period"),
           add.transformation=c("cd4^2","age^2"), use.stratum="hospital", B=25,
           var.remove=c("patient","cd4slope6"),print.time=TRUE,print.warnings=FALSE)

m2
```



```

plot(m2)

# ii) Linear model
# Model selection and averaging to identify predictors for immune recovery 6 months
# after starting antiretroviral therapy, presented as CD4 slope which is the average
# change in number of CD4 cells per week (deaths are ignored for this example)

# AIC based model selection (stepAIC) after multiple imputation
mami(HIV_imp, method="criterion.selection", ycol="cd4slope6",
      add.factor=c("hospital","stage","period"), var.remove=c("patient","dead","fuptime"))
# Model averaging (AIC weights) for variables typically captured
mami(HIV_imp, ycol="cd4slope6", add.factor=c("hospital","stage","period"),
      var.remove=c("patient","dead","fuptime","tb","cm","haem"))
# Mallows model averaging
mami(HIV_imp, method="MMA", ycol="cd4slope6", add.factor=c("hospital","stage","period"),
      var.remove=c("patient","dead","fuptime"))

#####
# Example 3: Model selection/averaging with no missing data, using shrinkage #
# Example from Tibshirani, R. (1996) Regression shrinkage and selection via the lasso, #
# Journal of the Royal Statistical Society, Series B 58(1), 267-288. #
# Useful to use mami to obtain Bootstrap CI after model selection/averaging #
#####

library(lasso2)
data(Prostate)
mami(Prostate, method="LASSO/LAE", missing.data="none", ycol="lpsa"
      , kfold=10) # LASSO (selection/averaging) based on 10-fold CV
mami(Prostate, missing.data="none", ycol="lpsa") # AIC based averaging
m3 <- mami(Prostate, missing.data="none", ycol="lpsa", inference="+bootstrapping",
           B=50, print.time=TRUE) # with Bootstrap CI

m3
plot(m3) # a few bimodal distributions: effect or not?

```

mma

MMA: Mallow's Model Averaging

Description

Performs model averaging on a set of nested candidate models with the weights vector chosen such that a specific Mallow's criterion is minimized.

Usage

```
mma(somedata, modelformula = NULL, ycol = 1)
```

Arguments

somedata	A dataframe or matrix of data.
modelformula	Formula of the full model.
ycol	Either a character string or integer specifying the column with the outcome variable.

Details

Mallow's Model Averaging (MMA) considers all nested submodels of the full model as candidate models, i.e if there are 7 variables there are 7 candidate models. The weight vector used to combine the respective estimates is chosen such that a certain Mallow's type criterion is minimized. The final weighted estimate targets to minimize the mean squared prediction error and is optimal in some sense, see Theorem 1 and Lemma 3 in Hansen, B. (2007, *Least Squares Model Averaging*, Econometrica, 75:1175-1189).

Note however that the results of MMA depend on the ordering of the regressors.

Value

Returns an object of class 'mma':

`coefficients` A matrix of estimated coefficients and standard errors.

`averaging.weights`

A matrix containing the weights for each models used in the model averaging procedure.

Author(s)

Michael Schomaker

References

Hansen, B. (2007), *Least Squares Model Averaging*, Econometrica, 75:1175-1189

Examples

```
library(lasso2)
data(Prostate)
mma(Prostate,modelformula=lpsa~.,ycol="lpsa")
```

plot.lae

Visualize estimates from Lasso Averaging Estimation

Description

The function plots (i) the weights for each candidate model relating to different candidate complexity parameters and (ii) the Lasso averaging, Lasso, OLS estimation, and variable importance results.

Usage

```
## S3 method for class lae
plot(x, xaxis = c("index", "realnumbers"), legend.place = "topright",
      display.importance = TRUE, ...)
```

Arguments

x	An object of class "lae".
xaxis	A string of either "index" or "realnumbers" to specify whether the x-axis should show the real values of the candidate complexity parameters or not.
legend.place	A string defining where the legend should be placed.
display.importance	A logical value defining whether the variable importance measure should be plotted or not.
...	Further graphical arguments to be passed.

Author(s)

Michael Schomaker

See Also

[lae](#) for Lasso Averaging Estimation.

Examples

```
library(lasso2)
data(Prostate)
l1 <- lae(Prostate, ycol=9, kfold=10, my.formula=lpsa~.)
plot(l1,xaxis="realnumbers")
```

plot.mami	<i>Plot bootstrap results when utilizing model selection/averaging after multiple imputation</i>
-----------	--

Description

Plots the bootstrap distribution of model selection/averaging estimators after multiple imputation. By default, the areas of the bootstrap distribution that are left and right of zero are shaded in different colors to better identify the importance of each variable.

Usage

```
## S3 method for class mami
plot(x, shade.areas = TRUE, plots.p.page = c("1", "4", "9", "16"),
     color = c("lightgrey", "darkgrey"), adj.bw = 1, ...)
```

Arguments

x	An object of class "mami".
shade.areas	A logical value indicating whether areas left and right of zero should be shaded or not.
plots.p.page	A string of either "1", "4", "9", or "16" indicating the number of plots per page.
color	A string vector of length 2 indicating the colors used for the shaded areas.
adj.bw	An integer to adjust the bandwidth.
...	Further graphical arguments to be passed.

Author(s)

Michael Schomaker

Examples

```
library(Amelia)
data(freetrade)
freetrade$pop <- log(freetrade$pop) # in line with original publication
freetrade_imp <- amelia(freetrade, ts = "year", cs = "country", noms="signed", polytime = 2,
                        intercs = TRUE, empri = 2)
m1 <- mami(freetrade_imp, method="criterion.selection", ycol="tariff", add.factor=c("country"),
           inference="+bootstrapping", B=25, X.org=freetrade, print.time=TRUE)
m1 # be patient with bootstrapping, increase B>=200 for better results
plot(m1)
```

print.lae

Printing results of Lasso Averaging Estimation

Description

Prints estimates and standard errors of Lasso (averaging) estimation and OLS estimation and reports variable importance and weights.

Usage

```
## S3 method for class lae
print(x, ...)
```

Arguments

x	An object of class "lae".
...	Further arguments to be passed.

Author(s)

Michael Schomaker

See Also

[lae](#) for Lasso Averaging Estimation.

Examples

```
library(lasso2)
data(Prostate)
l1<-lae(Prostate, ycol=9, kfold=10, my.formula=lpsa~.)
print(l1)
```

print.mami	<i>Printing results of model selection and model averaging after multiple imputation</i>
------------	--

Description

Prints estimates, standard errors and confidence intervals for model selection/averaging estimators after multiple imputation and reports variable importance if applicable.

Usage

```
## S3 method for class mami
print(x, string = "x", ...)
```

Arguments

x	An object of class "mami".
string	A string specifying what is printed in the summary table for non-selected variables.
...	Further arguments to be passed.

Author(s)

Michael Schomaker

See Also

[mami](#) for model selection and model averaging after multiple imputation.

Examples

```
library(Amelia)
data(freetrade)
freetrade$pop <- log(freetrade$pop) # in line with original publication
freetrade_imp <- amelia(freetrade, p2s=0, ts = "year", cs = "country", noms="signed",
                        polytime = 2, intercs = TRUE, empri = 2)
m1 <- mami(freetrade_imp, method="criterion.average", ycol="tariff", add.factor=c("country"))
print(m1)
m1
```

print.mma	<i>Printing results of Mallow's Model Averaging</i>
-----------	---

Description

Prints estimates of Mallow's Model Averaging.

Usage

```
## S3 method for class mma
print(x,...)
```

Arguments

`x` An object of class "mma".
`...` Further arguments to be passed.

Author(s)

Michael Schomaker

See Also

[mma](#) for Mallows's Model Averaging.

Examples

```
library(lasso2)
data(Prostate)
m1 <- mma(Prostate,modelformula=lpsa~.,ycol=9)
m1
```

summary.lae

Summarizes results from Lasso Averaging Estimation

Description

Summarizes the estimates and standard errors of Lasso (averaging) estimation and OLS estimation and reports variable importance.

Usage

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

Arguments

`object` An object of class "lae".
`...` Further arguments to be passed.

Author(s)

Michael Schomaker

See Also

[lae](#) for Lasso Averaging Estimation.

Examples

```
library(lasso2)
data(Prostate)
l1<-lae(Prostate, ycol=9, kfold=10, my.formula=lpsa~.)
summary(l1)
```

summary.mami	<i>Summarizes results of model selection and model averaging after multiple imputation</i>
--------------	--

Description

Summarizes estimates, standard errors and confidence intervals for model selection/averaging estimators after multiple imputation and reports variable importance if applicable.

Usage

```
## S3 method for class mami
summary(object, string="x", ...)
```

Arguments

object	An object of class "mami".
string	A string specifying what is printed in the summary table for non-selected variables.
...	Further arguments to be passed.

Author(s)

Michael Schomaker

See Also

[mami](#) for model selection and model averaging after multiple imputation.

Examples

```
library(Amelia)
data(freetrade)
freetrade$pop <- log(freetrade$pop) # in line with original publication
freetrade_imp <- amelia(freetrade, p2s=0, ts = "year", cs = "country", noms="signed",
                        polytime = 2, intercs = TRUE, empri = 2)
m1 <- mami(freetrade_imp, method="criterion.average", ycol="tariff", add.factor=c("country"))
summary(m1)
```

Index

*Topic **Bootstrapping**

mami, [5](#)

*Topic **Datasets**

HIV, [2](#)

*Topic **HIV**

HIV, [2](#)

*Topic **Model Averaging**

lae, [3](#)

mami, [5](#)

mami-package, [2](#)

mma, [9](#)

plot.lae, [10](#)

plot.mami, [11](#)

print.lae, [12](#)

print.mami, [13](#)

print.mma, [13](#)

summary.lae, [14](#)

summary.mami, [15](#)

*Topic **Model Selection**

mami, [5](#)

plot.mami, [11](#)

print.mami, [13](#)

summary.mami, [15](#)

*Topic **Multiple Imputation**

mami, [5](#)

plot.mami, [11](#)

print.mami, [13](#)

summary.mami, [15](#)

*Topic **Shrinkage**

lae, [3](#)

plot.lae, [10](#)

print.lae, [12](#)

summary.lae, [14](#)

HIV, [2](#)

lae, [3](#), [6](#), [7](#), [11](#), [12](#), [14](#)

MAMI (mami-package), [2](#)

mami, [5](#), [13](#), [15](#)

mami-package, [2](#)

mma, [7](#), [9](#), [14](#)

plot.lae, [4](#), [10](#)

plot.mami, [6](#), [7](#), [11](#)

print.lae, [12](#)

print.mami, [13](#)

print.mma, [13](#)

summary.lae, [14](#)

summary.mami, [15](#)