

Package ‘maSAE’

July 2, 2014

Type Package

Title Mandallaz' model-assisted small area estimators

Version 0.1-2

Date 2014-04-27

Description

an S4 implementation of the unbiased extension of the model-assisted synthetic-regression estimator proposed by Mandallaz (2013), Mandallaz et al. (2013) and Mandallaz (2014).
It yields smaller variances than the standard bias correction, the generalised regression estimator.

License GPL (>= 2)

Depends methods

Suggests nlme

Author Andreas Dominik Cullmann [aut, cre], Daniel Mandallaz [ctb], Alexander Francis Massey [ctb]

Maintainer Andreas Dominik Cullmann <r-package_masae@arcor.de>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-04-28 07:10:28

R topics documented:

maSAE-package	2
predict	3
s0	4
s1	5
s2	6
sadObj-class	6
saeObj-class	7
saObj	8
savObj-class	9

Index	11
--------------	-----------

Description

an S4 implementation of the unbiased extension of the model-assisted' synthetic-regression estimator proposed by Mandallaz (2013), Mandallaz et al. (2013) and Mandallaz (2014). It yields smaller variances than the standard bias correction, the generalised regression estimator.

Details

This package provides Mandallaz' extended synthetic-regression estimator for two- and three-phase sampling designs with or without clustering. See `vignette('maSAE', package = 'maSAE')` and `demo('maSAE', package = 'maSAE')` for introductions, "`class?maSAE::saeObj`" and "`?maSAE::predict`" for help on the main feature.

Note

Model-assisted estimators use models to improve the efficiency (i.e. reduce prediction error compared to design-based estimators) but need not assume them to be correct as in the model-based approach, which is advantageous in official statistics.

References

Mandallaz, D. 2013 Design-based properties of some small-area estimators in forest inventory with two-phase sampling. *Canadian Journal of Forest Research* **43**(5), pp. 441–449. doi: [10.1139/cjfr-2012-0381](https://doi.org/10.1139/cjfr-2012-0381).

Mandallaz, and Breschan, J. and Hill, A. 2013 New regression estimators in forest inventories with two-phase sampling and partially exhaustive information: a design-based Monte Carlo approach with applications to small-area estimation. *Canadian Journal of Forest Research* **43**(11), pp. 1023–1031. doi: [10.1139/cjfr-2013-0181](https://doi.org/10.1139/cjfr-2013-0181).

Mandallaz, D. 2014 A three-phase sampling extension of the generalized regression estimator with partially exhaustive information. *Canadian Journal of Forest Research* **44**(4), pp. 383–388. doi: [10.1139/cjfr-2013-0449](https://doi.org/10.1139/cjfr-2013-0449).

See Also

There are a couple packages for model-based small area estimation, see [sae](#), [rsae](#), [hbsae](#) and [JoSAE](#).

Examples

```
## Not run: vignette('maSAE', package = 'maSAE')
## Not run: demo('design', package = 'maSAE')
## Not run: demo('maSAE', package = 'maSAE')
```

predict	<i>Methods for Function predict</i>
---------	-------------------------------------

Description

Calculate small area predictions and their variances

Usage

```
predict(object, ...)
```

```
## S4 method for signature 'sadObj'
```

```
predict(object)
```

```
## S4 method for signature 'saeObj'
```

```
predict(object)
```

Arguments

object a model object for which prediction is desired.
 ... Arguments to be passed to methods.

Details

Based on the structure of the saeObj given, predict decides, which predictor to use:

If a smallAreaMeans-data.frame covering all fixed effects is given, the exhaustive estimator $\hat{Y}_{G, synth}$ is calculated.

If a smallAreaMeans-data.frame not covering all fixed effects is given, the partially exhaustive estimator $\hat{Y}_{G, greg}$ is calculated.

If no smallAreaMeans-data.frame but s1 is given, the three-phase estimator $\hat{Y}_{G, g3reg}$ is calculated.

If neither smallAreaMeans nor s1 are given, the non-exhaustive estimator $\hat{Y}_{G, psynth}$ is calculated.

If a clustering variable is given, the cluster sampling design equivalents of the above estimators are used.

Value

a data frame containing predictions and variances for each small area, attr(..., 'references') gives information on the literature used, attr(...\$prediction, 'reference') and attr(...\$variance, 'reference') specify these.

Methods

signature(object = saeObj) Calculate predictions and variances according to the auxilliary information given, see Details above.

signature(object = sadObj) Calculate design-based predictions and variances.

See Also

```
demo('maSAE')
```

Examples

```
library('maSAE')
## ## design-based estimation
## load data
data('s2')
## create object
sae0 <- saObj(data = s2, f = y ~ NULL | g)
## design-based estimation for all small areas given by g
predict(sae0)
## ## model-assisted estimation
## load s1 data
data('s1'); str(s1)
## add sample indicators to s2
s2$s1 <- s2$s2 <- TRUE
## add sample indicators to s1
s1$s1 <- TRUE
s1$s2 <- FALSE
## prepare s1 data
eval(parse(text=(paste('s1$', setdiff(names(s2), names(s1)), ' <- NA' , sep = '))))
## union s1 and s2 data
s12 <- rbind(s1, s2)
## create object
sae0 <- saObj(data = s12, f = y ~x1 + x2 + x3 | g, s2 = 's2')
## small area estimation
p <- predict(sae0)
## print p and view its attributes set by predict()
p;
str(p)
cat(sep = '\n', attr(p, 'references')[2])
attributes(p$prediction)
attributes(p$variance)
```

s0

Example s0 data set.

Description

Artificial null phase sampling data used for examples in the maSAE package.

Usage

```
data(s0)
```

Format

A data frame with 9008 observations on the following 5 variables.

Details

clustid See "?maSAE::s2"

x1 See "?maSAE::s2"

x2 See "?maSAE::s2"

x3 See "?maSAE::s2"

inclusion See "?maSAE::s2"

g See "?maSAE::s2"

s1

Example s1 data set.

Description

Artificial first phase sampling data used for examples in the maSAE package.

Usage

```
data(s1)
```

Format

A data frame with 786 observations on the following 5 variables.

Details

clustid See "?maSAE::s2"

x1 See "?maSAE::s2"

x2 See "?maSAE::s2"

x3 See "?maSAE::s2"

inclusion See "?maSAE::s2"

g See "?maSAE::s2"

s2	<i>Example s2 data set.</i>
----	-----------------------------

Description

Artificial second phase sampling data used for examples in the maSAE package.

Usage

```
data(s2)
```

Format

A data frame with 206 observations on the following 7 variables.

Details

clustid index giving the clusters.

x1 a potential fixed effect.

x2 another potential fixed effect.

x3 yet another potential fixed effect.

inclusion a logical vector indicating whether or not to include the current observation. All TRUE.

y the predictand

sadObj-class	<i>Class "sadObj"</i>
--------------	-----------------------

Description

a class for design-based estimation only

Details

See "[saeObj](#)". The fixed effects part of f has to be NULL: design-based estimation knows no fixed effects.

Slots

data See "[saeObj](#)".

f See "[saeObj](#)".

cluster See "[saeObj](#)".

include See "[saeObj](#)".

Extends

Class "savObj", directly.

Objects from the Class

Objects can be created by calls of the form `new("sadObj", ...)` or via the constructor function `"?maSAE::saObj"`.

Methods

`predict`

Note

the slots are described in `"class?maSAE::saeObj"`, since this is the main class of the package.

See Also

`"saeObj" "?maSAE::saObj"`

Examples

```
showClass("sadObj")
```

saeObj-class

Class "saeObj"

Description

the class for small area estimation, the one you're probably looking for.

Details

`cluster` optionally gives the name of a variable in slot data from which the cluster information for clustered sample designs is to be read. See Manadallaz 2013, p. 445 for Details.

`include` optionally gives the name of a variable in slot data from which the inclusion indicator for cluster points is to be read. See Manadallaz 2013, p. 445 for Details on I_f .

Also see the **Details** for `predict`.

Slots

`smallAreaMeans` An *optional* "data.frame" giving the true means of fixed effects for the small areas. Must have a column with the random effect defining the small areas in slot data.

`s1` An *optional* "character" string giving the name of a variable in slot data indicating that an observation (a row in slot data) belongs to subset 1.

`s2` An *optional* "character" string giving the name of a variable in slot data indicating that an observation (a row in slot data) belongs to subset 2.

data Object of class "data.frame" to use for prediction, typically consisting of a predictand and one or more predictors (zero or more fixed effects and one random effect defining the small areas). See **Details** for optional clustering variable and/or inclusion indicator.

f Object of class "formula" a linear mixed effects model formula.

cluster An *optional* "character" string giving the name of the clustering variable in slot data.

include An *optional* "character" string giving the name of the inclusion indicator in slot data.

Extends

Class "savObj", directly.

Objects from the Class

Objects can be created by calls of the form `new("saeObj", ...)` or via the constructor function `"?maSAE::saObj"` (recommended).

Methods

`predict`

References

Mandallaz, D. 2013 *Design-based properties of some small-area estimators in forest inventory with two-phase sampling*. *Canadian Journal of Forest Research* **43**(5), pp. 441–449. doi: [10.1139/cjfr-2012-0381](https://doi.org/10.1139/cjfr-2012-0381).

See Also

`"?stats::formula"`, `"class?maSAE::saObj"`, `"class?maSAE::savObj"`, `"?maSAE::saObj"` and `"?maSAE::predict"`

Examples

```
showClass("saeObj")
```

saObj

An ui-constructor for classes sadObj and saeObj

Description

simple wrapper to `new("sa[de]Obj")`. If missing, it adds an inclusion variable to data; it checks for missing in the clustering variable. Adds comments documenting changes made to the returned object.

Usage

```
saObj(data, f, smallAreaMeans = NULL, s1 = NULL, s2 = NULL,
       cluster = NULL, include = NULL)
```


Arguments

data	See " saeObj ".
f	a linear mixed effects formula, but see Value .
smallAreaMeans	See " saeObj ".
s1	See " saeObj ".
s2	See " saeObj ".
cluster	See " saeObj ".
include	See " saeObj ".

Value

an object of class `sadObj` if `f` is of structure `'x ~ NULL | g'`, an object of class `saeObj` otherwise.

See Also

["saeObj"](#), ["sadObj"](#).

Examples

```
library('maSAE')
## load data
data('s2')
## create sadObj object
sae0 <- saObj(data = s2, f = y ~ NULL | g)
## create saeObj object
s2$s2 <- TRUE
sae0 <- saObj(data = s2, f = y ~x1 + x2 + x3 | g, s2 = 's2')
```

savObj-class

Class "savObj"

Description

Common slots for classes `sadObj` and `saeObj`.

Slots

data See "[saeObj](#)".
 f See "[saeObj](#)".
 cluster See "[saeObj](#)".
 include See "[saeObj](#)".

Objects from the Class

A virtual Class: No objects may be created from it.

Note

the slots are described in "[class?maSAE::saeObj](#)", since this is the main class of the package.

See Also

["?stats::formula](#)", ["class?maSAE::sadObj](#)" and ["class?maSAE::saeObj](#)".

Examples

```
showClass("savObj")
```

Index

- *Topic **area**
 - predict, [3](#)
- *Topic **classes**
 - sadObj-class, [6](#)
 - saeObj-class, [7](#)
 - savObj-class, [9](#)
- *Topic **datasets**
 - s0, [4](#)
 - s1, [5](#)
 - s2, [6](#)
- *Topic **estimation**
 - predict, [3](#)
- *Topic **methods,**
 - predict, [3](#)
- *Topic **package**
 - maSAE-package, [2](#)
- *Topic **small**
 - predict, [3](#)
- ?maSAE::predict, [2](#), [8](#)
- ?maSAE::s2, [5](#)
- ?maSAE::saObj, [7](#), [8](#)
- ?stats::formula, [8](#), [10](#)

- class?maSAE::sadObj, [10](#)
- class?maSAE::saeObj, [2](#), [7](#), [10](#)
- class?maSAE::saObj, [8](#)
- class?maSAE::savObj, [8](#)

- hbsae, [2](#)

- JoSAE, [2](#)

- maSAE-package, [2](#)

- predict, [3](#), [7](#), [8](#)
- predict,sadObj-method (predict), [3](#)
- predict,saeObj-method (predict), [3](#)
- predict-methods (predict), [3](#)

- rsae, [2](#)

- s0, [4](#)
- s1, [5](#)
- s2, [6](#)
- sadObj, [9](#)
- sadObj-class, [6](#)
- sae, [2](#)
- saeObj, [6](#), [7](#), [9](#)
- saeObj-class, [7](#)
- saObj, [8](#)
- savObj, [7](#), [8](#)
- savObj-class, [9](#)

- tmp.maSAE.skeleton (maSAE-package), [2](#)