Package 'mase'

October 13, 2018

Type Package

Title Model-Assisted Survey Estimators

Version 0.1.2

Date 2018-10-12

Encoding UTF-8

Maintainer Kelly McConville <mcconville@reed.edu>

Description A set of model-assisted survey estimators and corresponding variance estimators for single stage, unequal probability, without replacement sampling designs. All of the estimators can be written as a generalized regression estimator with the Horvitz-Thompson, ratio, post-stratified, and regression estimators summarized by Sarndal et al. (1992, ISBN:978-0-387-40620-6). Two of the estimators employ a statistical learning model as the assisting model: the elastic net regression estimator, which is an extension of the lasso regression estimator given by McConville et al. (2017) <doi:10.1093/jssam/smw041>, and the regression tree estimator described in McConville and Toth (2017) <arXiv:1712.05708>. The variance estimators which approximate the joint inclusion probabilities can be found in Berger and Tille (2009) <doi:10.1016/S0169-7161(08)00002-3> and the bootstrap variance estimator is presented in Mashreghi et al. (2016) <doi:10.1214/16-SS113>.

License GPL-2

LazyData TRUE

Imports glmnet, Matrix, foreach, survey, dplyr, magrittr, rpms, boot, stats, Rdpack

Suggests roxygen2, testthat

Depends R (>= 3.1)

Collate 'gregt.R' 'varMase.R' 'GREG.R' 'gregElasticNett.R' 'gregElasticNet.R' 'gregTree.R' 'gregtreet.R' 'htt.R' 'horvitzThompson.R' 'logisticGregElasticNett.R' 'logisticGregt.R' 'postStratt.R' 'postStrat.R' 'ratioEstimatort.R' 'ratioEstimator.R' 'treeDesignMatrix.R'

RoxygenNote 6.1.0

RdMacros Rdpack

NeedsCompilation no

Author Kelly McConville [aut, cre, cph],
Becky Tang [aut],
George Zhu [aut],
Sida Li [ctb],
Shirley Chueng [ctb],
Daniell Toth [ctb, cph] (Author and copyright holder of
treeDesignMatrix helper function)

Repository CRAN

Date/Publication 2018-10-12 23:00:03 UTC

R topics documented:

greg *Compute a generalized regression estimator*

Description

Calculates a generalized regression estimator for a finite population mean/proportion or total based on sample data collected from a complex sampling design and auxiliary population data.

Usage

```
greg(y, x_sample, x_pop, pi = NULL, model = "linear", pi2 = NULL,
  var_est = FALSE, var_method = "lin_HB", data_type = "raw",
  N = NULL, model_select = FALSE, lambda = "lambda.min", B = 1000,
  strata = NULL)
```

Arguments

`y`	A numeric vector of the sampled response variable.
`x_sample`	A data frame of the auxiliary data in the sample.
`x_pop`	A data frame of population level auxiliary information. It must contain the same names as x_sample. If data_type = "raw", must contain unit level data. If data_type = "totals" or "means", then contains one row of aggregated, population totals or means for the auxiliary data. Default is "raw".

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`model` A string that specifies the regression model to utilize. Options are "linear" or "logistic".

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`var_est` A logical indicating whether or not to compute a variance estimator. Default is FALSE.

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement. The default is "lin_HB".

`data_type` A string that specifies the form of population auxiliary data. The possible values are "raw", "totals" or "means" for whether the user is providing population data at the unit level, aggregated to totals, or aggregated to means. Default is "raw".

`N` A numeric value of the population size. If NULL, it is estimated with the sum of the inverse of the pis.

`model_select` A logical for whether or not to run lasso regression first and then fit the model using only the predictors with non-zero lasso coefficients. Default is FALSE.

`lambda` A string specifying how to tune the lasso hyper-parameter. Only used if model_select = TRUE and defaults to "lambda.min". The possible values are "lambda.min", which is the lambda value associated with the minimum cross validation error or "lambda.1se", which is the lambda value associated with a cross validation error that is one standard error away from the minimum, resulting in a smaller model.

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean
- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate
- weights: Survey weights produced by greg (linear model only)
- coefficients: Survey-weighted model coefficients

References

Cassel C, Sarndal C, Wretman J (1976). “Some results on generalized difference estimation and generalized regression estimation for finite populations.” *Biometrika*, **63**, 615–620.

Sarndal C, Swensson B, Wretman J (1992). *Model Assisted Survey Sampling*. Springer-Verlag, New York.

See Also

`gregElasticNet` for a penalized regression model.

Examples

```
library(survey)
data(api)
greg(y = apisrs$api00, x_sample = apisrs[c("col.grad", "awards")],
x_pop = apipop[c("col.grad", "awards")], pi = apisrs$pw^(-1),
var_est = TRUE)

#To estimate a proportion
y <- 0 + (apisrs$both == "Yes")
greg(y = y, x_sample = apisrs[c("col.grad")],
x_pop = apipop[c("col.grad")], pi = apisrs$pw^(-1),
var_est = TRUE, model = "logistic")
```

`gregElasticNet` *Compute an elastic net regression estimator*

Description

Calculates a lasso, ridge or elastic net generalized regression estimator for a finite population mean/proportion or total based on sample data collected from a complex sampling design and auxiliary population data.

Usage

```
gregElasticNet(y, x_sample, x_pop, pi = NULL, alpha = 1,
  model = "linear", pi2 = NULL, var_est = FALSE,
  var_method = "lin_HB", data_type = "raw", N = NULL,
  lambda = "lambda.min", B = 1000, cvfolds = 10, strata = NULL)
```

Arguments

`y`	A numeric vector of the sampled response variable.
`x_sample`	A data frame of the auxiliary data in the sample.

`x_pop` A data frame of population level auxiliary information. It must contain the same names as x_sample. If data_type = "raw", must contain unit level data. If data_type = "totals" or "means", then contains one row of aggregated, population totals or means for the auxiliary data. Default is "raw".

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`alpha` A numeric value between 0 and 1 which signifies the mixing parameter for the lasso and ridge penalties in the elastic net. When alpha = 1, only a lasso penalty is used. When alpha = 0, only a ridge penalty is used. Default is alpha = 1.

`model` A string that specifies the regression model to utilize. Options are "linear" or "logistic".

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`var_est` A logical indicating whether or not to compute a variance estimator. Default is FALSE.

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement. The default is "lin_HB".

`data_type` A string that specifies the form of population auxiliary data. The possible values are "raw", "totals" or "means" for whether the user is providing population data at the unit level, aggregated to totals, or aggregated to means. Default is "raw".

`N` A numeric value of the population size. If NULL, it is estimated with the sum of the inverse of the pis.

`lambda` A string specifying how to tune the lambda hyper-parameter. Only used if model_select = TRUE and defaults to "lambda.min". The possible values are "lambda.min", which is the lambda_value associated with the minimum cross validation error or "lambda.1se", which is the lambda value associated with a cross validation error that is one standard error away from the minimum, resulting in a smaller model.

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`cvfolds` The number of folds for the cross-validation to tune lambda.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean

- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate
- coefficients: Survey-weighted model coefficients

References

McConville K, Breidt F, Lee T, Moisen G (2017). “Model-Assisted Survey Regression Estimation with the Lasso.” *Journal of Survey Statistics and Methodology*, **5**, 131-158.

See Also

`greg` for a linear or logistic regression model.

Examples

```
library(survey)
data(api)
gregElasticNet(y = apisrs$api00,
x_sample = apisrs[c("col.grad", "awards", "snum", "dnum", "cnum", "pcttest", "meals", "sch.wide")],
x_pop = apipop[c("col.grad", "awards", "snum", "dnum", "cnum", "pcttest", "meals", "sch.wide")],
pi = apisrs$pw^(-1), var_est = TRUE, alpha = .5)
```

`gregTree` *Compute a regression tree estimator*

Description

Calculates a regression tree estimator for a finite population mean or total based on sample data collected from a complex sampling design and auxiliary population data.

Usage

```
gregTree(y, x_sample, x_pop, pi = NULL, pi2 = NULL, var_est = FALSE,
  var_method = "lin_HB", B = 1000, p_value = 0.05, perm_reps = 500,
  bin_size = NULL, strata = NULL)
```

Arguments

`y` A numeric vector of the sampled response variable.

`x_sample` A data frame of the auxiliary data in the sample.

`x_pop` A data frame of population level auxiliary information. It must contain the same names as x_sample.

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`var_est` A logical indicating whether or not to compute a variance estimator. Default is FALSE.

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement. The default is "lin_HB".

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`p_value` Designated p-value level to reject null hypothesis in permutation test used to fit the regression tree. Default value is 0.05.

`perm_reps` An integer specifying the number of permutations for each permutation test run to fit the regression tree. Default value is 500.

`bin_size` A integer specifying the minimum number of observations in each node.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean
- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate
- weights: Survey weights produced by regression tree
- tree: rpms object

References

McConville K, Toth D (2018). “Automated selection of post-strata using a model-assisted regression tree estimator.” *Scandinavian Journal of Statistics*.

See Also

`greg` for a linear or logistic regression model.

Examples

```
library(survey)
data(api)
gregTree(y = apisrs$api00,
x_sample = apisrs[c("col.grad", "awards", "snum", "dnum", "cnum", "pcttest", "meals", "sch.wide")],
x_pop = apipop[c("col.grad", "awards", "snum", "dnum", "cnum", "pcttest", "meals", "sch.wide")])
```

horvitzThompson *Compute the Horvitz-Thompson Estimator*

Description

Calculate the Horvitz-Thompson Estimator for a finite population mean/proportion or total based on sample data collected from a complex sampling design.

Usage

```
horvitzThompson(y, pi = NULL, N = NULL, pi2 = NULL,
  var_est = FALSE, var_method = "lin_HB", B = 1000, strata = NULL)
```

Arguments

`y` A numeric vector of the sampled response variable.

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`N` A numeric value of the population size. If NULL, it is estimated with the sum of the inverse of the pis.

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`var_est` A logical indicating whether or not to compute a variance estimator. Default is FALSE.

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement. The default is "lin_HB".

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean
- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate

References

Horvitz DG, Thompson DJ (1952). “A generalization of sampling without replacement from a finite universe.” *Journal of the American Statistical Association*, **47**, 663-685.

Examples

```
library(survey)
data(api)
horvitzThompson(y = apisrs$api00, pi = apisrs$pw^(-1))
horvitzThompson(y = apisrs$api00, pi = apisrs$pw^(-1), var_est = TRUE, var_method = "lin_HTSRS")
```

`postStrat` *Compute a post-stratified estimator*

Description

Calculates a generalized regression estimator for a finite population mean/proportion or total based on sample data collected from a complex sampling design and auxiliary population data.

Usage

```
postStrat(y, x_sample, x_pop, pi = NULL, N = NULL, var_est = FALSE,
  var_method = "lin_HB", pi2 = NULL, data_type = "raw", B = 1000,
  strata = NULL)
```

Arguments

`y` A numeric vector of the sampled response variable.

`x_sample` A vector containing the post-stratum for each sampled unit.

`x_pop` A vector or data frame, depending on data_type. If data_type = "raw", then a vector containing the post-stratum for each population unit. If data_type = "totals" or "means", then a data frame, where the first column lists the possible post-strata and the second column contains the population total or proportion in each post-stratum.

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`N` A numeric value of the population size. If NULL, it is estimated with the sum of the inverse of the pis.

`var_est` Default to FALSE, logical for whether or not to compute estimate of variance

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement, "unconditional_SRS" = simple random sampling variance estimator which accounts for random strata.

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`data_type` Default to "raw", takes values "raw", "totals" or "means" for whether the user is providing the raw population stratum memberships, the population totals of each stratum, or the population proportions of each stratum.

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean
- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate
- strat_ests: Table of total and mean estimates for each strata
- weights: Survey weights produced by post stratification

References

Cochran W (1977). *Sampling Techniques*, 3rd edition. John Wiley & Sons, New York.

Sarndal C, Swensson B, Wretman J (1992). *Model Assisted Survey Sampling*. Springer-Verlag, New York.

See Also

`greg` for a linear or logistic regression model.

Examples

```
library(survey)
data(api)
postStrat(y = apisrs$api00, x_sample = apisrs$awards,
x_pop = data.frame(table(apipop$awards)), data_type = "totals",
pi = apisrs$pw^(-1))
```

`ratioEstimator` *Compute a ratio estimator*

Description

Calculates a ratio estimator for a finite population mean/proportion or total based on sample data collected from a complex sampling design and auxiliary population data.

Usage

```
ratioEstimator(y, x_sample, x_pop, data_type = "raw", pi = NULL,
  N = NULL, pi2 = NULL, var_est = FALSE, var_method = "lin_HB",
  B = 1000, strata = NULL)
```

Arguments

`y` A numeric vector of the sampled response variable.

`x_sample` A numeric vector of the sampled auxiliary variable.

`x_pop` A numeric vector of population level auxiliary information. Must come in the form of raw data, population total or population mean.

`data_type` A string that specifies the form of population auxiliary data. The possible values are "raw", "total" or "mean". If data_type = "raw", then x_pop must contain a numeric vector of the auxiliary variable for each unit in the population. If data_type = "total" or "mean", then contains either the population total or population mean for the auxiliary variable.

`pi` A numeric vector of inclusion probabilities for each sampled unit in y. If NULL, then simple random sampling without replacement is assumed.

`N` A numeric value of the population size. If NULL, it is estimated with the sum of the inverse of the pis.

`pi2` A square matrix of the joint inclusion probabilities. Needed for the "lin_HT" variance estimator.

`var_est` A logical indicating whether or not to compute a variance estimator. Default is FALSE.

`var_method` The method to use when computing the variance estimator. Options are a Taylor linearized technique: "lin_HB"= Hajek-Berger estimator, "lin_HH" = Hansen-Hurwitz estimator, "lin_HTSRS" = Horvitz-Thompson estimator under simple random sampling without replacement, and "lin_HT" = Horvitz-Thompson estimator or a resampling technique: "bootstrap_SRS" = bootstrap variance estimator under simple random sampling without replacement. The default is "lin_HB".

`B` The number of bootstrap samples if computing the bootstrap variance estimator. Default is 1000.

`strata` A factor vector of the stratum membership. If NULL, all units are put into the same stratum. Must have same length as y.

Value

A list of output containing:

- pop_total: Estimate of population total
- pop_mean: Estimate of the population mean
- pop_total_var: Estimated variance of population total estimate
- pop_mean_var: Estimated variance of population mean estimate
- weights: Survey weights produced by ratio estimator

References

Cochran W (1977). *Sampling Techniques*, 3rd edition. John Wiley & Sons, New York. Sarndal C, Swensson B, Wretman J (1992). *Model Assisted Survey Sampling*. Springer-Verlag, New York.

See Also

greg for a linear or logistic regression model.

Examples

```
library(survey)
data(api)
ratioEstimator(y = apisrs$api00, x_sample = apisrs$meals,
x_pop = sum(apipop$meals), data_type = "total", pi = apisrs$pw^(-1),
N = dim(apipop)[1])
```

Index