Package 'samplingEstimates'

February 20, 2015

Version 0.1-3

Date 2013-09-20

Title Sampling Estimates

Author Emilio Lopez Escobar [aut, cre]

Maintainer Emilio Lopez Escobar <emilio@numerika.mx>

Description Functions to estimate from survey data. This package is a user-friendly wrapper of the samplingVarEst package. It considers that the user is more familiar with practical survey data rather than with research on survey sampling (variance estimation). More functionalities are on the way.

Classification/MSC 62D05, 62F40, 62G09

Classification/JEL C13, C83

Classification/ACM G.3

Depends R (>= 3.0.0), samplingVarEst

License GPL (>= 2)

URL http://www.numerika.mx

NeedsCompilation no

Repository CRAN

Date/Publication 2014-05-13 21:11:37

R topics documented:

Estimate.Total.NHT 2
Sample1 7

Index **8**

`Estimate.Total.NHT` *Narain-Horvitz-Thompson estimates for a total from survey data*

Description

Produces estimates for population totals that are estimated using the Narain (1951); Horvitz-Thompson (1952) point estimator and from survey data obtained from a single-stage sampling design, i.e. direct element sampling.

Usage

```
Estimate.Total.NHT(MatY.s                 ,
                   VecWk.s                ,
                   VarEst         = "SYG",
                   MatPkl.s       = NULL ,
                   PopSize        = NULL ,
                   VecStratLb.s   = NULL ,
                   VecStratSize.s = NULL ,
                   ShowStrata     = FALSE,
                   VecDomainLb.s  = NULL )
```

Arguments

`MatY.s` matrix (dataframe or vector) with n rows (observations) and Q columns (variables of interest), where n is the overall sample size of elements. The argument `MatY.s` can also be a vector which is internally treated as a matrix with $Q = 1$. There must not be any missing value.

`VecWk.s` vector of the elements sampling weights; its length is equal to n, the sample size. Values in `VecWk.s` must be greater than or equal to one. Columns of `MatY.s` and length of `VecWk.s` must be the same. There must not be any missing value.

`VarEst` string indicating the mathematical expression for estimating the variance. Available options are: `"HT"`, `"SYG"` or `"Hajek"`. If `VarEst` argument is omitted, the default is `"SYG"`, which requires the provision of the matrix of joint inclusion probabilities `MatPkl.s`. In the case that the argument `MatPkl.s` is not provided, `VarEst` is set to `"Hajek"`. When using `VarEst="Hajek"`, it is assumed a high-entropy sampling design, see Hajek (1964); care should be taken with highly-stratified samples, e.g. Berger (2005).

`MatPkl.s` matrix of the second-order inclusion probabilities; its number of rows and columns is equal to n, the overall sample size of observed elements. Values in `MatPkl.s` must be greater than zero and less than or equal to one. There must not be any missing value.

`PopSize` population size N. This argument may be optional; if it is not provided the computations are made using $\hat{N} = \sum_{k \in s} w_k$, which estimates the total of elements in the population.

`VecStratLb.s` vector of the strata labels; its length is equal to n, the sample size. Values in the argument `VecStratLb.s` can be numeric (integers), strings or a factor. It does not need to be sorted, however, all other arguments (variables, vectors, matrices) of size n must follow the same order of `VecStratLb.s` correspondingly. This argument is optional, if it is not provided the computations are made assuming that there is no stratification. There must not be any missing value.

`VecStratSize.s` vector of the strata population sizes; its length is equal to n, the sample size. This vector contains, for each of the n observations, the size of the stratum each observation belongs to. The argument `VecStratSize.s` does not need to be sorted, however, all other arguments (variables, vectors, matrices) of size n must follow the same order of `VecStratSize.s` correspondingly. There must not be any missing value.

`ShowStrata` logical. If `TRUE` partial results from each stratum is displayed. This is an optional argument; default is `FALSE`.

`VecDomainLb.s` vector of the domains (sub-groups) labels; its length is equal to n, the sample size. Values in the argument `VecDomainLb.s` can be numeric (integers) or strings (characters). They do not need to be sorted, however this variable (column) must follow the same order of `VecStratLb.s` correspondingly with the other variables, vectors or matrices of size n. This argument is optional and there must not be any missing value.

Details

For the population total of the variable y:

$$t = \sum_{k\in U} y_k$$

the unbiased Narain (1951); Horvitz-Thompson (1952) estimator of t is given by:

$$\hat{t}_{NHT} = \sum_{k\in s} w_k y_k$$

where w_k denotes the sampling weight of the k-th element in the sample s, $w_k = 1/\pi_k$ with π_k denoting the inclusion probability of the k-th element in the sample. Let π_{kl} denotes the joint-inclusion probabilities of the k-th and l-th elements in the sample s. The variance of $\hat{t}_{NHT}$ is given by:

$$V(\hat{t}_{HT}) = \sum_{k\in U}\sum_{l\in U}(\pi_{kl} - \pi_k\pi_l) w_k y_k w_l y_l$$

which can therefore be estimated by the Horvitz-Thompson variance estimator (implemented by the current function if `VarEst="HT"`):

$$\hat{V}_{HT}(\hat{t}_{NHT}) = \sum_{k\in s}\sum_{l\in s}\frac{\pi_{kl} - \pi_k\pi_l}{\pi_{kl}} w_k y_k w_l y_l$$

If the utilised sampling design is of fixed-size, the variance $V(\hat{t}_{NHT})$ can be estimated by the Sen-Yates-Grundy variance estimator (implemented by the current function if `VarEst="SYG"`):

$$\hat{V}_{SYG}(\hat{t}_{NHT}) = \frac{-1}{2}\sum_{k\in s}\sum_{l\in s}\frac{\pi_{kl} - \pi_k\pi_l}{\pi_{kl}}\left(w_k y_k - w_l y_l\right)^2$$

For large-entropy sampling designs, the variance of $\hat{t}_{NHT}$ is approximated by the Hajek (1964) variance:

$$V_{Hajek}(\hat{t}_{NHT}) \doteq \frac{N}{N-1}\left[\sum_{k\in U} w_k y_k^2 \left(\frac{w_k-1}{w_k}\right) - dG^2\right]$$

with $d = \sum_{k\in U} w_k^{-2}(w_k-1)$ and $G = d^{-1}\sum_{k\in U} w_k^{-1}(w_k-1)y_k$.

This approximate variance can therefore be estimated by the variance estimator (implemented by the current function if `VarEst="Hajek"`):

$$\hat{V}_{Hajek}(\hat{t}_{NHT}) = \frac{n}{n-1}\left[\sum_{k\in s} w_k^2 y_k^2 \left(\frac{w_k-1}{w_k}\right) - \hat{d}\hat{G}^2\right]$$

where $\hat{d} = \sum_{k\in s} w_k^{-1}(w_k-1)$ and $\hat{G} = \hat{d}^{-1}\sum_{k\in s}(w_k-1)y_k$.

The Hajek (1964) variance approximation is designed for large-entropy sampling designs and large populations, i.e. care should be taken with highly-stratified samples, e.g. Berger (2005).

Value

The function returns a dataframe with Q rows (the number of variables of interest) and some columns depending on input information and used expressions in computations. The results in the returned columns are:

`Statistic` the utilised point estimator.

`VariableName` the name of the variable of interest.

`Estimate` the point estimate obtained from evaluating the sample data.

`Variance` the estimated variance of the point estimator.

`StdErr` the estimated standard error of the point estimator.

`AbsErr` the estimated absolute error of the point estimator.

`LInfCI95` the lower limit of the 95 percent confidence interval.

`LSupCI95` the upper limit of the 95 percent confidence interval.

`Range95` the range (width) of the 95 percent confidence interval.

`PctCVE` the estimated coefficient of variation (in percentage).

`DEff` the estimated design effect.

`n` the overall sample size.

`Nhat` an estimate of the population size (number of elements in the population) $\hat{N} = \sum_{k\in s} w_k$.

`fhat` an estimate of the overall sampling fraction $\hat{f} = n/\hat{N}$.

`N` the population size (total of elements in the population).

`f` the overall sampling fraction.

If a stratified sampling design was specified and if `ShowStrata=TRUE` some further columns are displayed with partial results. Note that these per-stratum partial results are NOT returned by the function, they are only on-screen information.

`h` stratum counter.

`Stratum` stratum label (integer, character).

`nh` the sample size for the stratum h.

`Nh` the size of the stratum h (total of elements in the stratum h).

`fh` the sampling fraction for the stratum h.

`Wh` the relative weight of the stratum h among all strata $W_h = n_h/N_h$.

If domains of study were specified these extra columns are displayed. Note that these per-domain results are NOT returned by the function, they are only on-screen information.

`d` domain counter.

`Domain` domain label.

`nd` the sample size in the domain d.

`Ndhat` an estimate of the population size (number of elements) for the domain d.

`fdhat` an estimate of the sampling fraction for the domain d.

`Wdhat` an estimate of the relative weight of the domain d among all domains.

References

Berger, Y. G. (2005) Variance estimation with highly stratified sampling designs with unequal probabilities. *Australian & New Zealand Journal of Statistics*, **47**, 365–373.

Hajek, J. (1964) Asymptotic theory of rejective sampling with varying probabilities from a finite population. *The Annals of Mathematical Statistics*, **35**, 4, 1491–1523.

Horvitz, D. G. and Thompson, D. J. (1952) A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, **47**, 663–685.

Narain, R. D. (1951) On sampling without replacement with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, **3**, 169–175.

Sen, A. R. (1953) On the estimate of the variance in sampling with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, **5**, 119–127.

Yates, F. and Grundy, P. M. (1953) Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society B*, **15**, 253–261.

Examples

```
##################################
## Setting up data to run examples
##################################
data(Sample1)           ## Loads a data frame with the sample to be used in examples
N        <- 570         ## Defining the population size
## Approximating the 2nd order inclusion probabilities with sample based quantitites
## (Note: this approximation is only suitable for large-entropy sampling designs)
require(samplingVarEst) ## Loading the necessary package
Probs2Mat <- Pkl.Hajek.s(Sample1$InclProbs) ## function from samplingVarEst package
head(Sample1)           ## Showing the first rows of the sample data to be used

###########################################################
```

```
## Example 1: A variable of interest, without stratification
############################################################
Estimate.Total.NHT(MatY.s   = Sample1$y1     ,
                   VecWk.s  = Sample1$Weights)

Estimate.Total.NHT(MatY.s   = Sample1$y1     ,
                   VecWk.s  = Sample1$Weights,
                   VarEst   = "HT"           )

Estimate.Total.NHT(MatY.s   = Sample1$y1     ,
                   VecWk.s  = Sample1$Weights,
                   VarEst   = "SYG"          ,
                   MatPkl.s = Probs2Mat      )

Estimate.Total.NHT(MatY.s   = Sample1$y1     ,
                   VecWk.s  = Sample1$Weights,
                   VarEst   = "SYG"          ,
                   MatPkl.s = Probs2Mat      ,
                   PopSize  = N              )


###################################################################################
## Example 2: A matrix/dataframe of 2 variables of interest, without stratification
###################################################################################
Estimate.Total.NHT(MatY.s   = Sample1[ ,c("y1","y2")],
                   VecWk.s  = Sample1$Weights        ,
                   VarEst   = "SYG"                  ,
                   MatPkl.s = Probs2Mat              ,
                   PopSize  = N                      )


#########################################################
## Example 3: A variable of interest, with stratification
#########################################################
Estimate.Total.NHT(MatY.s         = Sample1$y1              ,
                   VecWk.s        = Sample1$Weights         ,
                   VecStratLb.s   = Sample1$CharStrataNames,
                   VecStratSize.s = Sample1$StrataSizes     )

Estimate.Total.NHT(MatY.s         = Sample1$y1              ,
                   VecWk.s        = Sample1$Weights         ,
                   VecStratLb.s   = Sample1$CharStrataNames,
                   VecStratSize.s = Sample1$StrataSizes     ,
                   ShowStrata     = TRUE                    )


###############################################################################
## Example 4: A matrix/dataframe (2 variables of interest), with stratification
###############################################################################
Estimate.Total.NHT(MatY.s         = Sample1[ ,c("y1","y2")],
                   VecWk.s        = Sample1$Weights         ,
                   VecStratLb.s   = Sample1$CharStrataNames,
                   VecStratSize.s = Sample1$StrataSizes     ,
                   ShowStrata     = TRUE                   )


#################################################################################
## Example 5: A matrix/dataframe (2 variables), no strata, with unplanned domains
#################################################################################
Estimate.Total.NHT(MatY.s        = Sample1[ ,c(”y1”,”y2”)],
                   VecWk.s       = Sample1$Weights        ,
                   VecDomainLb.s = Sample1$CharDoms       )

Estimate.Total.NHT(MatY.s        = Sample1[ ,c(”y1”,”y2”)],
                   VecWk.s       = Sample1$Weights        ,
                   VecDomainLb.s = Sample1$NumDoms        )


###################################################################################
## Example 6: A matrix/dataframe (2 variables), with strata, with unplanned domains
###################################################################################
Estimate.Total.NHT(MatY.s         = Sample1[ ,c(”y1”,”y2”)],
                   VecWk.s        = Sample1$Weights        ,
                   VecStratLb.s   = Sample1$CharStrataNames,
                   VecStratSize.s = Sample1$StrataSizes    ,
                   ShowStrata     = TRUE                   ,
                   VecDomainLb.s  = Sample1$CharDoms       )
```

Sample1 *Sample data 1 to run examples*

Description

Dataset with a sample of size 373 from a population of size 570. It is a dataframe with 373 observations of 8 variables.

Usage

```
data(Sample1)
```

Examples

```
data(Sample1) ###Loads the Sample1 dataset
```

Index

∗Topic **datasets**
    Sample1, 7
∗Topic **domains**
    Estimate.Total.NHT, 2
∗Topic **estimates**
    Estimate.Total.NHT, 2
∗Topic **strata**
    Estimate.Total.NHT, 2
∗Topic **total**
    Estimate.Total.NHT, 2

Estimate.Total.NHT, 2

Sample1, 7