Package ‘optimStrat’

March 20, 2020

Type Package

Title Choosing the Sample Strategy

Version 2.1

Date 2020-03-19

Author Edgar Bueno <edgar.bueno@stat.su.se>

Maintainer Edgar Bueno <edgar.bueno@stat.su.se>

Depends shiny, mvtnorm, cubature

Description Intended to assist in the choice of the sampling strategy to implement in a survey.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2020-03-20 08:50:02 UTC

R topics documented:

`optimStrat-package` *optimStrat*

Description

OptimStrat is a package intended to assist in the choice of the sample strategy to implement in a survey. It compares five strategies having into account the information available in an auxiliary variable and two superpopulation models, called working and true models.

Details

The package includes a web-based application where the user can compare five sampling strategies in order to determine which one to implement in a survey.

The package also includes a function to perform the comparison mentioned above, as well as functions for stratifying the auxiliary variable and calculations of the variance of Stratified Simple Random Sampling and Pareto πps.

Author(s)

Edgar Bueno

References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proporional-to-size Sampling.* Research Report, Department of Statistics, Stockholm University 2018:6. http://gauss.stat.su.se/rr/RR2018_6.pdf.

`desmse` *Design Mean Squared Error*

Description

Compute the design Mean Squared Error of five sampling strategies.

Usage

```
desmse(y, x, n, H, d2, d4)
```

Arguments

`y`	a numeric vector giving the values of the study variable.
`x`	a positive numeric vector giving the values of the auxiliary variable.
`n`	a positive integer indicating the desired sample size.
`H`	a positive integer smaller or equal than `length(x)` giving the desired number of strata/poststrata.
`d2`	a number giving the *assumed* shape of the trend term in the superpopulation model.
`d4`	a number giving the *assumed* shape of the spread term in the superpopulation model.

Details

The design Mean Squared Error of a sample of size `n` is computed for five sampling strategies (πps–reg, STSI–reg, STSI–HT, πps–pos and STSI–pos). The strategies are defined assuming that there is an underlying superpopulation model of the form

$$Y_k = \delta_0 + \delta_1 x_k^{\delta_2} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \delta_3^2 x_k^{2\delta_4}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

The number of strata/poststrata is given by `H`.

Value

A vector of length five with the Mean Squared Error of the five sample strategies in the following order: πps–reg, STSI–reg, STSI–HT, πps–pos and STSI–pos.

References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proporional-to-size Sampling.* Research Report, Department of Statistics, Stockholm University 2018:6. http://gauss.stat.su.se/rr/RR2018_6.pdf.

See Also

`expmse` for the anticipated MSE of the five strategies.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
y<- simulatey(x, b0=0, b1=1, b2=1.25, b4=0.5, rho=0.90)
desmse(y,x,n=500,H=6,d2=1.25,d4=0.50)
desmse(y,x,n=500,H=6,d2=1.00,d4=1.00)
```

expmse *Anticipated Mean Squared Error*

Description

Compute the anticipated Mean Squared Error of five sampling strategies.

Usage

```
expmse(b, d, x, n, H, Rxy, estrato1 = NULL, estrato2 = NULL, st = 1:5,
   short = FALSE)
```

Arguments

b	a numeric vector of length two giving the *true* shapes of the trend and spread terms.
d	a numeric vector of length two giving the *assumed* shapes of the trend and spread terms.
x	a positive numeric vector giving the values of the auxiliary variable.
n	a positive integer indicating the desired sample size.
H	a positive integer smaller or equal than `length(x)` giving the desired number of strata/poststrata. Ignored if `estrato1` and `estrato2` are given.
Rxy	a number giving the correlation between the auxiliary variable and the study variable.
estrato1	a list giving stratum and sample sizes per stratum (see 'Details').
estrato2	a list giving stratum and sample sizes per stratum (see 'Details').
st	a numeric vector indicating the strategies for which the anticipated MSE is to be calculated (see 'Details').
short	logical. If `FALSE` (the default) a vector of length five is returned. If `TRUE` only the strategies given by `st` are returned.

Details

The Anticipated Mean Squared Error of a sample of size `n` is computed for five sampling strategies (πps–reg, STSI–reg, STSI–HT, πps–pos and STSI–pos).

The strategies are defined assuming that the underlying superpopulation model is of the form

$$Y_k = \delta_0 + \delta_1 x_k^{\delta_2} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \delta_3^2 x_k^{2\delta_4}$ and $Cov(\epsilon_k, \epsilon_l) = 0$. But the true generating model is of the form

$$Y_k = \beta_0 + \beta_1 x_k^{\beta_2} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \beta_3^2 x_k^{2\beta_4}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

The parameters β_2 and β_4 are given by b. The parameters δ_2 and δ_4 are given by d.

estrato1 and estrato2 are lists with two components (each with length length(x)): stratum indicates the stratum to which each element belongs and nh indicates the sample sizes to be selected in each stratum. They can be created via optiallo. estrato1 gives the stratification for STSI–HT and the poststrata for πps–pos and STSI–pos; whereas estrato2 gives the stratification for STSI–reg and STSI–pos. If NULL, optiallo is used for defining H strata/poststrata.

st indicates which MSEs to be calculated. If 1 in st, the anticipated MSE of πps–reg is calculated. If 2 in st, the anticipated MSE of STSI–reg is calculated, and so on.

Value

If short=FALSE a vector of length five is returned giving the anticipated MSE of the strategies given in st. NA is returned for those strategies not given in st. If short=TRUE, the NAs are omitted.

References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proporional-to-size Sampling.* Research Report, Department of Statistics, Stockholm University 2018:6. http://gauss.stat.su.se/rr/RR2018_6.pdf.

See Also

optiallo for how to stratify an auxiliary variable and allocate the sample size; desmse for calculating the MSE of the five strategies.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
expmse(b=c(1,1),d=c(1,1),x,n=500,H=6,Rxy=0.9)
expmse(b=c(1,1),d=c(1,1),x,n=500,H=6,Rxy=0.9,st=1:3)
expmse(b=c(1,1),d=c(1,1),x,n=500,H=6,Rxy=0.9,st=1:3,short=TRUE)

stratum<- optiallo(n=500,x,H=6)
poststratum<- optiallo(n=500,x^1.5,H=10)
expmse(b=c(1,1),d=c(1,1),x,n=500,H=6,Rxy=0.9,
   estrato1=poststratum,estrato2=stratum)
```

---

expmsepips *Anticipated Mean Squared Error of a PIps design*

Description

Compute the anticipated Mean Squared Error of a PIps design.

Usage

```
expmsepips(x, pik, n, Beta11, Beta12, Beta21, Beta22, Delta12, Rfy, ak = 1)
```

Arguments

- `x` a matrix or data.frame giving the values of the auxiliary variables.
- `pik` a numeric vector giving the variable with respect to which the inclusion probabilities are to be obtained.
- `n` a positive integer indicating the desired sample size.
- `Beta11` a numeric vector of length equal to the number of variables in `x` giving the coefficients of the trend term in the *true* superpopulation model (see 'Details').
- `Beta12` a numeric vector of length equal to the number of variables in `x` giving the exponents of the trend term in the *true* superpopulation model (see 'Details').
- `Beta21` a numeric vector of length equal to the number of variables in `x` giving the coefficients of the spread term in the *true* superpopulation model (see 'Details').
- `Beta22` a numeric vector of length equal to the number of variables in `x` giving the exponents of the spread term in the *true* superpopulation model (see 'Details').
- `Delta12` a numeric vector of length equal to the number of variables in `x` giving the exponents of the trend term in the *assumed* superpopulation model (see 'Details').
- `Rfy` a number giving the square root of the coefficient of determination between the auxiliary variables and the study varible.
- `ak` a vector of weights.

Details

The Anticipated Mean Squared Error of the strategy that couples a PIps design with the general regression estimator of a sample of size `n` is computed.

It is assumed that the underlying superpopulation model is of the form

$$Y_k = \sum_{j=1}^{J} \delta_{1,j} x_{jk}^{\delta_{1,J+j}} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \sigma^2 \sum_{j=1}^{J} \delta_{2,j} x_{jk}^{\delta_{2,J+j}}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

But the true generating model is of the form

$$Y_k = \sum_{j=1}^{J} \beta_{1,j} x_{jk}^{\beta_{1,J+j}} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \sigma^2 \sum_{j=1}^{J} \beta_{2,j} x_{jk}^{\beta_{2,J+j}}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

The coefficients $\beta_{1,j}$ $(j = 1, \cdots, J)$ are given by `Beta11`. The exponents $\beta_{1,j}$ $(j = J+1, \cdots, 2J)$ are given by `Beta12`. The coefficients $\beta_{2,j}$ $(j = 1, \cdots, J)$ are given by `Beta21`. The exponents $\beta_{2,j}$ $(j = J+1, \cdots, 2J)$ are given by `Beta22`.

The exponents $\delta_{1,j}$ $(j = J+1, \cdots, 2J)$ are given by `Delta12`.

The inclusion probabilities are calculated as $n \times x_k / t_x$ and corrected, if necessary, to ensure that they are smaller or equal than one.

Value

A numeric value giving the anticipated Mean Squared Error.

References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proporional-to-size Sampling*. Research Report, Department of Statistics, Stockholm University 2018:6. http://gauss.stat.su.se/rr/RR2018_6.pdf.

Examples

```
x1<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x2<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x3<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x<- cbind(x1,x2,x3)
expmsepips(x,pik=x3,n=150,Beta11=c(1,-1,0),Beta12=c(1,1,0),Beta21=c(0,0,1),
   Beta22=c(0,0,0.5),Delta12=c(1,1,0),Rfy=0.8)
expmsepips(x,pik=x2,n=150,Beta11=c(1,-1,0),Beta12=c(1,1,0),Beta21=c(0,0,1),
   Beta22=c(0,0,0.5),Delta12=c(1,1,0),Rfy=0.8)
```

---

expmsestsi — *Anticipated Mean Squared Error of a STSI design*

Description

Compute the anticipated Mean Squared Error of a Stratified Simple Random Sampling design.

Usage

```
expmsestsi(x, stratum, nh, Beta11, Beta12, Beta21, Beta22, Delta12, Rfy, ak = 1)
```

Arguments

`x`	a matrix or data.frame giving the values of the auxiliary variables.
`stratum`	a vector indicating the stratum to which each element belongs.
`nh`	a vector indicating the sample size of the stratum to which each element belongs.
`Beta11`	a numeric vector of length equal to the number of variables in `x` giving the coefficients of the trend term in the *true* superpopulation model (see 'Details').
`Beta12`	a numeric vector of length equal to the number of variables in `x` giving the exponents of the trend term in the *true* superpopulation model (see 'Details').
`Beta21`	a numeric vector of length equal to the number of variables in `x` giving the coefficients of the spread term in the *true* superpopulation model (see 'Details').
`Beta22`	a numeric vector of length equal to the number of variables in `x` giving the exponents of the spread term in the *true* superpopulation model (see 'Details').
`Delta12`	a numeric vector of length equal to the number of variables in `x` giving the exponents of the trend term in the *assumed* superpopulation model (see 'Details').

`Rfy` a number giving the square root of the coefficient of determination between the auxiliary variables and the study varible.

`ak` a vector of weights.

Details

The Anticipated Mean Squared Error of the strategy that couples a STSI design with the general regression estimator is computed.

It is assumed that the underlying superpopulation model is of the form

$$Y_k = \sum_{j=1}^{J} \delta_{1,j} x_{jk}^{\delta_{1,J+j}} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \sigma^2 \sum_{j=1}^{J} \delta_{2,j} x_{jk}^{\delta_{2,J+j}}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

But the true generating model is of the form

$$Y_k = \sum_{j=1}^{J} \beta_{1,j} x_{jk}^{\beta_{1,J+j}} + \epsilon_k$$

with $E\epsilon_k = 0$, $V\epsilon_k = \sigma^2 \sum_{j=1}^{J} \beta_{2,j} x_{jk}^{\beta_{2,J+j}}$ and $Cov(\epsilon_k, \epsilon_l) = 0$.

The coefficients $\beta_{1,j}$ ($j = 1, \cdots, J$) are given by `Beta11`. The exponents $\beta_{1,j}$ ($j = J+1, \cdots, 2J$) are given by `Beta12`. The coefficients $\beta_{2,j}$ ($j = 1, \cdots, J$) are given by `Beta21`. The exponents $\beta_{2,j}$ ($j = J+1, \cdots, 2J$) are given by `Beta22`.

The exponents $\delta_{1,j}$ ($j = J+1, \cdots, 2J$) are given by `Delta12`.

Value

A numeric value giving the anticipated Mean Squared Error.

References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proporional-to-size Sampling.* Research Report, Department of Statistics, Stockholm University 2018:6. http://gauss.stat.su.se/rr/RR2018_6.pdf.

Examples

```
x1<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x2<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x3<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
x<- cbind(x1,x2,x3)
stratum1<- optiallo(n=150,x=x3,H=6)
expmsestsi(x,stratum1$stratum,stratum1$nh,Beta11=c(1,-1,0),Beta12=c(1,1,0),
   Beta21=c(0,0,1),Beta22=c(0,0,0.5),Delta12=c(1,1,0),Rfy=0.8)
expmsestsi(x,stratum1$stratum,stratum1$nh,Beta11=c(1,-1,0),Beta12=c(1,1,0),
   Beta21=c(0,0,1),Beta22=c(0,0,0.5),Delta12=c(1,0,1),Rfy=0.8)
```

gk *Computes the gk*

Description

Compute the values of the function gk.

Usage

```
gk(x, Beta21, Beta22)
```

Arguments

x	a matrix or data.frame giving the values of the auxiliary variables.
Beta21	a numeric vector of length equal to the number of variables in x giving the coefficients of the spread term in the *true* superpopulation model (see 'Details').
Beta22	a numeric vector of length equal to the number of variables in x giving the exponents of the spread term in the *true* superpopulation model (see 'Details').

Details

Compute the values of

$$g(x_k|\beta) = \sum_{j=1}^{J} \beta_{2,j} x_{jk}^{\beta_{2,J+j}}$$

The coefficients $\beta_{2,j}$ $(j = 1, \cdots, J)$ are given by Beta21. The exponents $\beta_{2,j}$ $(j = J+1, \cdots, 2J)$ are given by Beta22.

Value

A numeric vector giving the values of the function.

Examples

```
x1<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x2<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x3<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x<- cbind(x1,x2,x3)
gk(x,Beta21=c(1,2,-1),Beta22=c(1,0.75,0.5))
```

optiallo *Optimal allocation in stratified simple random sampling*

Description

Allocates a sample of size n using Neyman optimal allocation in Stratified Simple Random Sampling.

Usage

```
optiallo(n, x, stratum = NULL, ...)
```

Arguments

n	a positive integer indicating the desired sample size.
x	a positive numeric vector giving the values of the auxiliary variable.
stratum	a vector indicating the stratum to which every unit belongs (see 'Details').
...	other arguments passed to stratify (see 'Details').

Details

Allocates a sample of size n using Neyman optimal allocation in Stratified Simple Random Sampling.

If stratum==NULL, the stratification is generated via stratify. Then at least the number of strata should be passed to stratify using the argument H.

Value

A list with two elements:

stratum	a vector indicating the stratum to which each element belongs.
nh	a vector indicating the sample size of the strata to which each element belongs.

See Also

stratify for defining the stratification using the cum-sqrt-rule.

Examples

```
x<- 1 + sort( rgamma(100, shape=4/9, scale=108) )
st1<- stratify(x,H=6)
optiallo(n=30,x,stratum=st1)

optiallo(n=30,x,H=6)
```

optimApp — *Interactive Web-based Application of optimStrat*

Description

Call Shiny to run a web-based application of `optimStrat`.

Usage

```
optimApp()
```

Author(s)

Edgar Bueno, <edgar.bueno@stat.su.se>

pinc — *Inclusion probabilities in a PIps design*

Description

Compute the inclusion probabilities to be used in a PIps design with sample size equal to `n`.

Usage

```
pinc(n, x)
```

Arguments

`n`	a positive integer indicating the desired sample size.
`x`	a positive numeric vector giving the values of the auxiliary variable.

Details

The inclusion probabilities are calculated as $n \times x_k/t_x$ and corrected, if necessary, to ensure that they are smaller or equal than one.

Value

A numeric vector giving the inclusion probability of each element.

Examples

```
x<- 1 + sort( rgamma(100, shape=4/9, scale=108) )
pinc(n=30,x)
```

simulatey *Simulate the Study Variable*

Description

Simulate values for the study variable based on the auxiliary variable `x` and the parameters of a superpopulation model.

Usage

```
simulatey(x, b0, b1, b2, b4, rho=NULL, b3=NULL)
```

Arguments

`x`	a positive numeric vector giving the values of the auxiliary variable.
`b0`	a number giving the intercept of the trend term in the superpopulation model.
`b1`	a number giving the scale of the trend term in the superpopulation model.
`b2`	a number giving the shape of the trend term in the superpopulation model.
`b4`	a number giving the shape of the spread term in the superpopulation model.
`rho`	a number giving the absolute value of the desired correlation between `x` and the vector to be simulated.
`b3`	a nonnegative number giving the scale of the spread term in the superpopulation model. Ignored if `rho` is given (see 'Details').

Details

The values of the study variable `y` are simulated using a superpopulation model defined as follows:

$$Y_k = \beta_0 + \beta_1 x_k^{\beta_2} + \epsilon_k$$

with ϵ_k $N(0, \beta_3 x_k^{\beta_4})$.

Note that `b3` defines the degree of association between `x` and `y`: the larger `b3`, the smaller the correlation, `rho`, and vice versa. For this reason only one of them should be defined. If both are defined, `b3` will be ignored.

The sign of the correlation should be given through `b1` (see 'Examples').

Depending on the value of `b2`, some correlations cannot be reached, e.g. if `b2=2` it is pointless to set `rho=1`. In those cases, `b3` will automatically be set to zero and `rho` will be ignored (see 'Examples').

Value

A numeric vector giving the simulated value of `y` associated to each value in `x`.

Examples

```
#Linear trend and homocedasticity
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
y<- simulatey(x, b0=0, b1=1, b2=1, b4=0, rho=0.90)
plot(x, y)

#Linear trend and heterocedasticity
y<- simulatey(x, b0=0, b1=1, b2=1, b4=1, rho=0.90)
plot(x, y)

#Quadratic trend and homocedasticity
y<- simulatey(x, b0=0, b1=1, b2=2, b4=0, rho=0.80)
plot(x, y)

#Correlation of minus one
y<- simulatey(x, b0=0, b1=-1, b2=1, b4=0, rho=1)
cor(x, y)
plot(x, y)

#Desired correlation cannot be attained
y<- simulatey(x, b0=0, b1=1, b2=3, b4=0, rho=0.99)
cor(x, y)
plot(x, y)
```

`skewness` *Sample Skewness*

Description

Calculate the sample skewness.

Usage

```
skewness(x, na.rm = FALSE)
```

Arguments

`x`	a numeric vector.
`na.rm`	a logical value indicating whether `NA` values should be stripped before the computation proceeds.

Details

Compute the sample skewness of `x` as

$$\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-\bar{x})^3}{\left[\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\bar{x})^2\right]^{3/2}}$$

Value

A vector of length one giving the sample skewness of `x`.

Examples

```
x<- rnorm(1000)
skewness(x)
```

`stratify` *Stratification of an Auxiliary Variable*

Description

Stratify the auxiliary variable `x` into `H` strata using the cum-sqrt-rule.

Usage

```
stratify(x, H, forced = FALSE, J = NULL)
```

Arguments

`x`	a positive numeric vector giving the values of the auxiliary variable.
`H`	a positive integer smaller or equal than `length(x)` giving the desired number of strata.
`forced`	a logical value indicating if the number of strata *must* be exactly equal to `H` (see 'Details').
`J`	a positive integer indicating the number of bins used for the cum-sqrt-rule.

Details

The cum-sqrt-rule is used in order to define `H` strata from the auxiliary vector `x`.

Depending on some characteristics of `x`, e.g. high skewness, few observations or too many ties, the resulting stratification may have a number of strata other than `H`. Using `forced = TRUE` tries its best to obtain exactly `H` strata.

Note that if `length(x) < H` then `forced` will be set to `FALSE`.

Value

A numeric vector giving the stratum to which each observation in `x` belongs.

References

Sarndal, C.E., Swensson, B. and Wretman, J. (1992). *Model Assisted Survey Sampling*. Springer.

See Also

optiallo for allocating the sample into the strata using Neyman optimal allocation; varstsi for computing the variance of Stratified Simple Random Sample.

Examples

```
x<- 1 + sort( rgamma(100, shape=4/9, scale=108) )
stratify(x, H=3)
```

---

varpips *Variance of Pareto PIps Sampling with the HT Estimator*

Description

Compute the design variance of the Horvitz-Thompson estimator of the total of y under Pareto probability proportional-to-size Sampling, where the size variable is indicated by x and the sample size is n.

Usage

```
varpips(y,x,n)
```

Arguments

y	a numeric vector giving the values of the study variable.
x	a positive numeric vector giving the values of the auxiliary variable that is used in order to define the inclusion probabilities.
n	a positive integer indicating the desired sample size.

Details

Target inclusion probabilities are computed as $\pi_k = n \cdot x_k / \sum x_k$.

If $\pi_k > 1$ for at least one element, π_k is set equal to one for those elements and the inclusion probabilities are calculated again for the remaining elements with the remaining sample size.

Once the π_k are obtained, the variance of the Horvitz-Thompson estimator under Pareto probability proportional-to-size Sampling is computed as: $V_{\pi ps}\left[\hat{t}_{HT}\right] = \frac{N}{N-1}(t_1 - \frac{t_2^2}{t_3})$ with

$$t_1 = \sum \frac{y_k^2(1-\pi_k)}{\pi_k}$$

$$t_2 = \sum y_k(1-\pi_k)$$

$$t_3 = \sum \pi_k(1-\pi_k)$$

Value

A numeric value giving the variance of the Horvitz-Thompson estimator under Pareto probability proportional-to-size Sampling.

References

Rosen, B. (1997). *On Sampling with Probability Proportional to Size.* Journal of Statistical Planning and Inference **62**, 159-191.

See Also

`varstsi` for the variance of the Horvitz-Thompson estimator under stratified simple random sampling; `varpipspos` for the variance of the poststratified estimator under probability proportional-to-size sampling; `varstsipos` for the variance of the poststratified estimator under stratified simple random sampling; `varpipsreg` for the variance of the regression estimator under probability proportional-to-size sampling; `varstsireg` for the variance of the regression estimator under stratified simple random sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varpips(y,x=x^1.25,n=500)
```

---

varpipspos — *Design variance of a PIps–pos sampling strategy.*

Description

Compute the design variance of the poststratified estimator of the total of `y` under Pareto probability proportional-to-size Sampling, where the size variable is indicated by `x_des` and the sample size is `n`.

Usage

```
varpipspos(y, x_des, n, poststratum)
```

Arguments

`y`	a numeric vector giving the values of the study variable.
`x_des`	a positive numeric vector giving the values of the auxiliary variable that is used in order to define the inclusion probabilities.
`n`	a positive integer indicating the desired sample size.
`poststratum`	a vector indicating the poststratum to which each element belongs.

Details

Target inclusion probabilities are computed as $\pi_k = n \cdot x_k / \sum x_k$.

If $\pi_k > 1$ for at least one element, π_k is set equal to one for those elements and the inclusion probabilities are calculated again for the remaining elements with the remaining sample size.

Once the π_k are obtained, the variance of the poststratified estimator under Pareto probability proportional-to-size Sampling is computed as: $V_{\pi ps}\left[\hat{t}_{HT}\right] = \frac{N}{N-1}(t_1 - \frac{t_2^2}{t_3})$ with

$$t_1 = \sum \frac{E_k^2(1-\pi_k)}{\pi_k}$$

$$t_2 = \sum E_k(1-\pi_k)$$

$$t_3 = \sum \pi_k(1-\pi_k)$$

with $E_k = y_k - B_g$ and $B_g = \bar{y}_g$.

Value

A numeric value giving the variance of the poststratified estimator under Pareto probability proportional-to-size Sampling.

References

Rosen, B. (1997). *On Sampling with Probability Proportional to Size.* Journal of Statistical Planning and Inference **62**, 159-191.

See Also

varpips for the variance of the Horvitz-Thompson estimator under probability proportional-to-size sampling; varstsi for the variance of the Horvitz-Thompson estimator under stratified simple random sampling; varstsipos for the variance of the poststratified estimator under stratified simple random sampling; varpipsreg for the variance of the regression estimator under probability proportional-to-size sampling; varstsireg for the variance of the regression estimator under stratified simple random sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
postst1<- stratify(x^1.25,H=6)
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varpipspos(y, x_des=x^0.75, n=500, poststratum=postst1)
```

`varpipsreg` *Design variance of a PIps–reg sampling strategy.*

Description

Compute the design variance of the regression estimator of the total of `y` under Pareto probability proportional-to-size Sampling, where the size variable is indicated by `x_des` and the sample size is `n`.

Usage

```
varpipsreg(y, x_des, n, x_est)
```

Arguments

`y`	a numeric vector giving the values of the study variable.
`x_des`	a positive numeric vector giving the values of the auxiliary variable that is used for defining the inclusion probabilities.
`n`	a positive integer indicating the desired sample size.
`x_est`	a positive numeric vector giving the values of the auxiliary variable that is used at the estimation stage.

Details

Target inclusion probabilities are computed as $\pi_k = n \cdot x_k / \sum x_k$.

If $\pi_k > 1$ for at least one element, π_k is set equal to one for those elements and the inclusion probabilities are calculated again for the remaining elements with the remaining sample size.

Once the π_k are obtained, the variance of the poststratified estimator under Pareto probability proportional-to-size Sampling is computed as: $V_{\pi ps}\left[\hat{t}_{HT}\right] = \frac{N}{N-1}(t_1 - \frac{t_2^2}{t_3})$ with

$$t_1 = \sum \frac{E_k^2(1-\pi_k)}{\pi_k}$$

$$t_2 = \sum E_k(1-\pi_k)$$

$$t_3 = \sum \pi_k(1-\pi_k)$$

with $E_k = y_k - \hat{y}_k$.

Value

A numeric value giving the variance of the regression estimator under Pareto probability proportional-to-size Sampling.

References

Rosen, B. (1997). *On Sampling with Probability Proportional to Size.* Journal of Statistical Planning and Inference **62**, 159-191.

See Also

varpips for the variance of the Horvitz-Thompson estimator under probability proportional-to-size sampling; varstsi for the variance of the Horvitz-Thompson estimator under stratified simple random sampling; varpipspos for the variance of the poststratified estimator under probability proportional-to-size sampling; varstsipos for the variance of the poststratified estimator under stratified simple random sampling; varstsireg for the variance of the regression estimator under stratified simple random sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varpipsreg(y, x_des=x^0.75, n=500, x_est=x^1.25)
```

---

varstsi *Variance of STSI Sampling with the HT Estimator*

Description

Compute the design variance of the Horvitz-Thompson estimator of the total of y under Stratified Simple Random Sampling, where strata are indicated by stratum and the sample sizes by stratum are given by nh.

Usage

```
varstsi(y,stratum,nh)
```

Arguments

y	a numeric vector giving the values of the study variable.
stratum	a vector indicating the stratum to which each element belongs.
nh	a vector indicating the sample size of the stratum to which each element belongs.

Details

The variance of the Horvitz-Thompson estimator under Stratified Simple Random Sampling is computed as: $V_{STSI}\left[\hat{t}_{HT}\right] = \sum_h V_h$ with

$$V_h = \frac{N_h^2}{n_h}\left(1 - \frac{n_h}{N_h}\right) S^2_{y,U_h}$$

where S^2_{y,U_h} is the variance of y in the hth stratum.

The variance of Simple Random Sampling is computed if stratum is a constant.

Value

A numeric value giving the variance of the Horvitz-Thompson estimator under Stratified Simple Random Sampling.

References

Sarndal, C.E., Swensson, B. and Wretman, J. (1992). *Model Assisted Survey Sampling*. Springer.

See Also

stratify for a method to define the strata; optiallo for Neyman optimal allocation of the sample; varpips for the variance of the Horvitz-Thompson estimator under probability proportional-to-size sampling; varpipspos for the variance of the poststratified estimator under probability proportional-to-size sampling; varstsipos for the variance of the poststratified estimator under stratified simple random sampling; varpipsreg for the variance of the regression estimator under probability proportional-to-size sampling; varstsireg for the variance of the regression estimator under stratified simple random sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
st1<- optiallo(n=100,x=x^0.75,H=6)
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varstsi(y,stratum=st1$stratum,nh=st1$nh)
```

---

varstsipos *Design variance of a STSI–pos sampling strategy.*

Description

Compute the design variance of the poststratified estimator of the total of y under Stratified Simple Random Sampling, where strata are indicated by stratum and the sample of size n is allocated using Neyman allocation with respect to x.

Usage

```
varstsipos(y, stratum, nh, poststratum)
```

Arguments

y	a numeric vector giving the values of the study variable.
stratum	a vector indicating the stratum to which each element belongs.
nh	a vector indicating the sample size of the stratum to which each element belongs.
poststratum	a vector indicating the poststratum to which each element belongs.

Details

A sample of size n is allocated into the strata using x-optimal allocation, i.e.

$$n_h \propto N_h S_{x,U_h}$$

where N_h is the size of the hth stratum, S_{x,U_h} is the standard deviation of x in the hth stratum and $propto$ stands for 'proportional to'.

If $n_h > N_h$ for at least one stratum, n_h is set equal to N_h in those strata and optimal allocation is used again for the remaining strata with the remaining sample size.

Once the n_h are obtained, the variance of the poststratified estimator under Stratified Simple Random Sampling is computed as: $V_{STSI}\left[\hat{t}_{HT}\right] = \sum_h V_h$ with

$$V_h = \frac{N_h^2}{n_h}\left(1 - \frac{n_h}{N_h}\right) S_{E,U_h}^2$$

where S_{E,U_h}^2 is the variance of E in the hth stratum with $E_k = y_k - B_g$ and $B_g = \bar{y}_g$.

Value

A numeric value giving the variance of the poststratified estimator under Stratified Simple Random Sampling.

See Also

varpips for the variance of the Horvitz-Thompson estimator under probability proportional-to-size sampling; varstsi for the variance of the Horvitz-Thompson estimator under stratified simple random sampling; varpipspos for the variance of the poststratified estimator under probability proportional-to-size sampling; varpipsreg for the variance of the regression estimator under probability proportional-to-size sampling; varstsireg for the variance of the regression estimator under stratified simple random sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
strat1<- optiallo(n=150,x^0.75,H=6)
post1<- stratify(x^1.25,H=6)
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varstsipos(y, stratum=strat1$stratum,nh=strat1$nh,poststratum=post1)
```

---

varstsireg *Design variance of a STSI–reg sampling strategy.*

Description

Compute the design variance of the poststratified estimator of the total of y under Stratified Simple Random Sampling, where strata are indicated by stratum and the sample of size n is allocated using Neyman allocation with respect to x.

Usage

```
varstsireg(y, stratum, nh, x)
```

Arguments

y	a numeric vector giving the values of the study variable.
stratum	a vector indicating the stratum to which each element belongs.
nh	a vector indicating the sample size of the stratum to which each element belongs.
x	a positive numeric vector giving the values of the auxiliary variable that is used at the estimation stage.

Details

A sample of size n is allocated into the strata using x-optimal allocation, i.e.

$$n_h \propto N_h S_{x,U_h}$$

where N_h is the size of the hth stratum, S_{x,U_h} is the standard deviation of x in the hth stratum and $propto$ stands for 'proportional to'.

If $n_h > N_h$ for at least one stratum, n_h is set equal to N_h in those strata and optimal allocation is used again for the remaining strata with the remaining sample size.

Once the n_h are obtained, the variance of the poststratified estimator under Stratified Simple Random Sampling is computed as: $V_{STSI}\left[\hat{t}_{HT}\right] = \sum_h V_h$ with

$$V_h = \frac{N_h^2}{n_h}\left(1 - \frac{n_h}{N_h}\right) S_{E,U_h}^2$$

where S_{E,U_h}^2 is the variance of E in the hth stratum with $E_k = y_k - \hat{y}_k$.

Value

A numeric value giving the variance of the regression estimator under Stratified Simple Random Sampling.

See Also

varpips for the variance of the Horvitz-Thompson estimator under probability proportional-to-size sampling; varstsi for the variance of the Horvitz-Thompson estimator under stratified simple random sampling; varpipspos for the variance of the poststratified estimator under probability proportional-to-size sampling; varstsipos for the variance of the poststratified estimator under stratified simple random sampling; varpipsreg for the variance of the regression estimator under probability proportional-to-size sampling.

Examples

```
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) ) #simulating the auxiliary variable
strat1<- optiallo(n=150,x^0.75,H=6)
y<- simulatey(x,b0=10,b1=1,b2=1.25,b4=0.75,rho=0.95)
varstsireg(y, stratum=strat1$stratum,nh=strat1$nh,x=x^1.25)
```

vk *Calculate the values of the function f.*

Description

Calculate the values of the function f under both the true and the misspecified model.

Usage

```
vk(x, Beta11, Beta12, Delta12, ak = 1)
```

Arguments

x	a matrix or data.frame giving the values of the auxiliary variables.
Beta11	a numeric vector of length equal to the number of variables in x giving the coefficients of the trend term in the *true* superpopulation model (see 'Details').
Beta12	a numeric vector of length equal to the number of variables in x giving the exponents of the trend term in the *true* superpopulation model (see 'Details').
Delta12	a numeric vector of length equal to the number of variables in x giving the exponents of the trend term in the *assumed* superpopulation model (see 'Details').
ak	a vector of weights.

Details

Compute the values of

$$f(x_k|\beta) = \sum_{j=1}^{J} \beta_{1,j} x_{jk}^{\beta_{1,J+j}}$$

and

$$f(x_k|\delta) = \sum_{j=1}^{J} \hat{\delta}_{1,j} x_{jk}^{\delta_{1,J+j}}$$

where $\hat{\delta}_1 = A\beta_1$ and

$$A = \left(\sum_U \frac{x_k^{\delta_{12'}} x_k^{\delta_{12}}}{a_k} \right) \sum_U \frac{x_k^{\delta_{12'}} x_k^{\beta_{12'}}}{a_k}$$

.

The coefficients $\beta_{1,j}$ $(j = 1, \cdots, J)$ are given by Beta11. The exponents $\beta_{1,j}$ $(j = J+1, \cdots, 2J)$ are given by Beta12. The exponents $\delta_{1,j}$ $(j = J+1, \cdots, 2J)$ are given by Delta12.

Value

A list with two components

fbk	a vector giving the values of the function f under the true model
fdk	a vector giving the values of the function f under the misspecified model

Examples

```
x1<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x2<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x3<- 1 + sort( rgamma(30, shape=4/9, scale=108) )
x<- cbind(x1,x2,x3)
vk(x,Beta11=c(1,2,-1),Beta12=c(1,0.75,0.5),Delta12=c(1,1,1))
```

Index