

# Package ‘optimStrat’

October 14, 2022

**Type** Package

**Title** Choosing the Sample Strategy

**Version** 2.3

**Date** 2020-11-11

**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** 2020-11-11 11:40:02 UTC

## R topics documented:

optimStrat-package . . . . .	2
desvar . . . . .	2
expgreg . . . . .	3
expvar . . . . .	7
optiallo . . . . .	9
optimApp . . . . .	10
pinc . . . . .	10
simulatey . . . . .	11
skewness . . . . .	12
stratify . . . . .	13
vargreg . . . . .	14

<b>Index</b>	<b>17</b>
--------------	-----------

---

optimStrat-package      *optimStrat*

---

### Description

OptimStrat is a package intended to assist in the choice of the sample strategy to implement in a survey. It allows for calculating the variance and the expected variance of several sampling strategies.

### Details

The package includes a function to calculate the design variance of several sampling strategies. It also includes a function to calculate the expected variance under a superpopulation model and a web-based application where the user can compare five sampling strategies in order to determine which one to implement in a survey.

### Author(s)

Edgar Bueno

### References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proportional-to-size Sampling*. Research Report, Department of Statistics, Stockholm University 2018:6. [http://gauss.stat.su.se/rr/RR2018\\_6.pdf](http://gauss.stat.su.se/rr/RR2018_6.pdf).

---

desvar      *Design variance*

---

### Description

Compute the design variance of six sampling strategies.

### Usage

```
desvar(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 giving the desired number of strata/poststrata.
d2	a number giving the <i>assumed</i> shape of the trend term in the superpopulation model.
d4	a number giving the <i>assumed</i> shape of the spread term in the superpopulation model.

## Details

The design variance of a sample of size  $n$  is computed for six sampling strategies (stsi-HT,  $\pi$ ps-HT, stsi-pos,  $\pi$ ps-pos, stsi-reg and  $\pi$ ps-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 six with the variance of the six sampling strategies.

## References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proportional-to-size Sampling*. Research Report, Department of Statistics, Stockholm University 2018:6. [http://gauss.stat.su.se/rr/RR2018\\_6.pdf](http://gauss.stat.su.se/rr/RR2018_6.pdf).

## See Also

[expvar](#) for the expected variance of five sampling strategies.

## Examples

```
f<- function(x,b0,b1,b2,...) {b0+b1*x^b2}
g<- function(x,b3,...) {x^b3}
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
y<- simulatey(x,f,g,dist="gamma",b0=10,b1=1,b2=1.25,b3=0.5,rho=0.90)

desvar(y,x,n=500,H=6,d2=1.25,d4=0.50)
desvar(y,x,n=500,H=6,d2=1.00,d4=1.00)
```

---

expgreg

*Expected variance of the general regression estimator*

---

## Description

Compute the expected design variance of the general regression estimator of the total of a study variable under different sampling designs.

## Usage

```
expgreg(x, b11, b12, b21, b22, d12, Rfy, n, design = NULL,
        stratum = NULL, x_des = NULL, inc.p = NULL, ...)
```

**Arguments**

x	design matrix with the variables to be used into the GREG estimator.
b11	a numeric vector of length equal to the number of variables in x giving the coefficients of the trend term in the <i>true</i> superpopulation model (see ‘Details’).
b12	a numeric vector of length equal to the number of variables in x giving the exponents of the trend term in the <i>true</i> superpopulation model (see ‘Details’).
b21	a numeric vector of length equal to the number of variables in x giving the coefficients of the spread term in the <i>true</i> superpopulation model (see ‘Details’).
b22	a numeric vector of length equal to the number of variables in x giving the exponents of the spread term in the <i>true</i> superpopulation model (see ‘Details’).
d12	a numeric vector of length equal to the number of variables in x giving the exponents of the trend term in the <i>assumed</i> superpopulation model (see ‘Details’).
Rfy	a number giving the square root of the coefficient of determination between the auxiliary variables and the study variable.
n	either a positive number indicating the (expected) sample size (when design is one of ‘srs’, ‘poi’, ‘pips’ or NULL) or a numeric vector indicating the sample size of the strata to which each element belongs (when design is ‘stsi’) (see ‘Examples’).
design	a character string giving the sampling design. It must be one of ‘srs’ (simple random sampling without replacement), ‘poi’ (Poisson sampling), ‘stsi’ (stratified simple random sampling), ‘pips’ (Pareto $\pi$ ps sampling) or NULL (see ‘Details’).
stratum	a vector indicating the stratum to which every unit belongs. Only used if design is ‘stsi’.
x_des	a positive numeric vector giving the values of the auxiliary variable that is used for defining the inclusion probabilities. Only used if design is ‘poi’ or ‘pips’.
inc.p	a matrix giving the first and second order inclusion probabilities. Only used if design is NULL.
...	other arguments passed to <code>lm</code> (see ‘Details’).

**Details**

The expected variance of the general regression estimator under different sampling designs is computed.

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

$$Y_k = f(x_k|\delta_1) + \epsilon_k$$

with  $E\epsilon_k = 0$ ,  $V\epsilon_k = \sigma_0^2 g^2(x_k|\delta_2)$  and  $Cov(\epsilon_k, \epsilon_l) = 0$ .

But the true generating model is in fact of the form

$$Y_k = f(x_k|\beta_1) + \epsilon_k$$

with  $E\epsilon_k = 0$ ,  $V\epsilon_k = \sigma^2 g^2(x_k|\beta_2)$  and  $Cov(\epsilon_k, \epsilon_l) = 0$ .

Where

$$f(x_k|\delta_1) = \sum_{j=1}^J \delta_{1,j} x_{jk}^{\delta_{1,j}},$$

$$g(x_k|\delta_2) = \sum_{j=1}^J \delta_{2,j} x_{jk}^{\delta_{2,j}},$$

$$f(x_k|\beta_1) = \sum_{j=1}^J \beta_{1,j} x_{jk}^{\beta_{1,j}},$$

$$g(x_k|\beta_2) = \sum_{j=1}^J \beta_{2,j} x_{jk}^{\beta_{2,j}}.$$

- the coefficients  $\beta_{1,j}$  ( $j = 1, \dots, J$ ) are given by b11;
- the exponents  $\beta_{1,j}$  ( $j = J + 1, \dots, 2J$ ) are given by b12;
- the coefficients  $\beta_{2,j}$  ( $j = 1, \dots, J$ ) are given by b21;
- the exponents  $\beta_{2,j}$  ( $j = J + 1, \dots, 2J$ ) are given by b22;
- the exponents  $\delta_{1,j}$  ( $j = J + 1, \dots, 2J$ ) are given by d12.

The expected variance of the GREG estimator is approximated by

$$E(V(\hat{t})) = V(\hat{t}_z) + \hat{\sigma}^2 \sum_{k=1}^N \left( \frac{1}{\pi_k} - 1 \right) g^2(x_k|\beta_2)$$

where

$$V(\hat{t}_z) = \sum_{k=1}^N \sum_{l=1}^N \pi_{kl} \frac{z_k}{\pi_k} \frac{z_l}{\pi_l} - \left( \sum_{k=1}^N z_k \right)^2$$

and

$$\hat{\sigma}^2 = \frac{S_f^2}{\bar{g}^2} \left( \frac{1}{R_{fy}^2} - 1 \right),$$

$$z_k = \left( x_k^\beta - x_k^\delta A \right) \beta_1^{**},$$

$$S_f^2 = \sum_{k=1}^N (f(x_k|\beta_1) - \bar{f})^2 / N,$$

$$\bar{g}^2 = \sum_{k=1}^N g(x_k|\beta_2)^2 / N,$$

$$x_k^\beta = \left( x_{1k}^{\beta_{1,J+1}}, \dots, x_{Jk}^{\beta_{1,2J}} \right),$$

$$x_k^\delta = \left( x_{1k}^{\delta_{1,J+1}}, \dots, x_{Jk}^{\delta_{1,2J}} \right),$$

$$\beta_1^{**} = (\beta_{1,1}, \dots, \beta_{1,J})',$$

$$A = \left( \sum_{k=1}^N w_k x_k^{\delta'} x_k^{\delta} \right)^{-1} \sum_{k=1}^N w_k x_k^{\delta'} x_k^{\beta}.$$

$N$  is the population size and  $\pi_k$  and  $\pi_{kl}$  are, respectively, the first and second order inclusion probabilities.  $w_k$  is a weight associated to each element and it represents the inverse of the conditional variance (up to a scalar) of the underlying superpopulation model (see ‘Examples’).

If `design=NULL`, the matrix of inclusion probabilities is obtained proportional to the matrix `p.inc`. If `design` is other than `NULL`, the formula for the variance is simplified in such a way that the inclusion probabilities matrix is no longer necessary. In particular:

- if `design='srs'`, only the sample size `n` is required;
- if `design='stsi'`, both the stratum ID `stratum` and the sample size per stratum `n`, are required;
- if `design` is either `'pips'` or `'poi'`, the inclusion probabilities are obtained proportional to the values of `x_des`, corrected if necessary.

### Value

A numeric value giving the expected variance of the general regression estimator for the desired design under the working and true models.

### References

Bueno, E. (2018). *A Comparison of Stratified Simple Random Sampling and Probability Proportional-to-size Sampling*. Research Report, Department of Statistics, Stockholm University 2018:6. [http://gauss.stat.su.se/rr/RR2018\\_6.pdf](http://gauss.stat.su.se/rr/RR2018_6.pdf).

### See Also

`expvar` for the simultaneous calculation of the expected variance of five sampling strategies under a superpopulation model; `vargreg` for the variance of the GREG estimator; `desvar` for the simultaneous calculation of the variance of six sampling strategies; `optimApp` for an interactive application of `expgreg`.

### 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)
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,1,0),Rfy=0.8,n=150,"pips",x_des=x3)
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,1,0),Rfy=0.8,n=150,"pips",x_des=x2)
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,1,0),Rfy=0.8,n=150,"pips",x_des=x2,weights=1/x1)

st1<- optiallo(n=150,x=x3,H=6)
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,1,0),Rfy=0.8,n=st1$nh,"stsi",stratum=st1$stratum)
```

```
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,0,1),Rfy=0.8,n=st1$nh,"stsi",stratum=st1$stratum)
expgreg(x,b11=c(1,-1,0),b12=c(1,1,0),b21=c(0,0,1),b22=c(0,0,0.5),
        d12=c(1,0,1),Rfy=0.8,n=st1$nh,"stsi",stratum=st1$stratum,weights=1/x1)
```

---

expvar	<i>Expected variance</i>
--------	--------------------------

---

### Description

Compute the expected variance of five sampling strategies.

### Usage

```
expvar(b, d, x, n, H, Rxy, stratum1 = NULL, stratum2 = NULL, st = 1:5,
       short = FALSE)
```

### Arguments

b	a numeric vector of length two giving the <i>true</i> shapes of the trend and spread terms.
d	a numeric vector of length two giving the <i>assumed</i> 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 giving the desired number of strata/poststrata. Ignored if stratum1 and stratum2 are given.
Rxy	a number giving the correlation between the auxiliary variable and the study variable.
stratum1	a list giving stratum and sample sizes per stratum (see ‘Details’).
stratum2	a list giving stratum and sample sizes per stratum (see ‘Details’).
st	a numeric vector indicating the strategies for which the expected variance 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 expected variance of a sample of size n is computed for five sampling strategies ( $\pi$ ps-reg, STSI-reg, STSI-HT,  $\pi$ ps-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  $\beta_2$  and  $\beta_4$  are given by b. The parameters  $\delta_2$  and  $\delta_4$  are given by d.

stratum1 and stratum2 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`. stratum1 gives the stratification for STSI-HT and the poststrata for  $\pi$ ps-pos and STSI-pos; whereas stratum2 gives the stratification for STSI-reg and STSI-pos. If NULL, `optiallo` is used for defining H strata/poststrata.

st indicates which variances to be calculated. If 1 in st, the expected variance of  $\pi$ ps-reg is calculated. If 2 in st, the expected variance of STSI-reg is calculated, and so on.

### Value

If short=FALSE a vector of length five is returned giving the expected variance 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 Proportional-to-size Sampling*. Research Report, Department of Statistics, Stockholm University 2018:6. [http://gauss.stat.su.se/rr/RR2018\\_6.pdf](http://gauss.stat.su.se/rr/RR2018_6.pdf).

### See Also

`optiallo` for how to stratify an auxiliary variable and allocate the sample size; `desvar` for calculating the variance of the five strategies.

### Examples

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

st1<- optiallo(n=500,x,H=6)
post1<- optiallo(n=500,x^1.5,H=10)
expvar(b=c(1,1),d=c(1,1),x,n=500,H=6,Rxy=0.9,
       stratum1=post1,stratum2=st1)
```



---

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

<code>n</code>	a positive integer indicating the desired sample size.
<code>x</code>	a positive numeric vector giving the values of the auxiliary variable.
<code>stratum</code>	a vector indicating the stratum to which every unit belongs (see ‘Details’).
<code>...</code>	other arguments passed to <a href="#">stratify</a> (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:

<code>stratum</code>	a vector indicating the stratum to which each element belongs.
<code>nh</code>	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	<i>Simulate the Study Variable</i>
-----------	------------------------------------

---

**Description**

Simulate values for the study variable based on the auxiliary variable  $x$  and an assumed superpopulation model.

**Usage**

```
simulatey(x, f, g, dist = "normal", rho = NULL, Sigma = NULL, ...)
```

**Arguments**

$x$	a numeric vector giving the values of the auxiliary variable.
$f$	the name of the function defining the desired trend (see ‘Details’).
$g$	the name of the function defining the desired spread (see ‘Details’).
$dist$	the desired distribution of the study variable conditioned on the auxiliary variable. Either ‘normal’ or ‘gamma’ (see ‘Details’).
$\rho$	a number giving the absolute value of the desired correlation between $x$ and the vector to be simulated.
$\Sigma$	a nonnegative number giving the scale of the spread term in the superpopulation model. Ignored if $\rho$ is given (see ‘Details’).
...	other arguments passed to $f$ and $g$ (see ‘Details’).

**Details**

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

$$Y_k = f(x_k) + \epsilon_k$$

with  $E(\epsilon_k) = 0$ ,  $V(\epsilon_k) = \sigma^2 g^2(x_k)$  and  $Cov(\epsilon_k, \epsilon_l) = 0$  if  $k \neq l$ . Also  $Y_k | f(x_k)$  is distributed according to  $dist$ .

$f$  and  $g$  should return a vector of the same length of  $x$ . Their first argument should be  $x$  and they should not share the name of any other argument. Both  $f$  and  $g$  should have the ... argument (see ‘Examples’).

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

Depending on the trend function  $f$ , some correlations cannot be reached. In those cases,  $\Sigma$  will automatically be set to zero,  $dist$  will automatically be set to ‘normal’ and  $\rho$  will be ignored (see ‘Examples’).

If the trend term takes negative values,  $dist$  will be automatically set to ‘normal’.

**Value**

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

**Examples**

```
f<- function(x,b0,b1,b2,...) {b0+b1*x^b2}
g<- function(x,b3,...) {x^b3}

x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )

#Linear trend and homocedasticity
y1<- simulatey(x,f,g,dist="normal",b0=0,b1=1,b2=1,b3=0,rho=0.90)
y2<- simulatey(x,f,g,dist="gamma",b0=0,b1=1,b2=1,b3=0,rho=0.90)

#Linear trend and heterocedasticity
y3<- simulatey(x,f,g,dist="normal",b0=0,b1=1,b2=1,b3=1,rho=0.90)
y4<- simulatey(x,f,g,dist="gamma",b0=0,b1=1,b2=1,b3=1,rho=0.90)

#Quadratic trend and homocedasticity
y5<- simulatey(x,f,g,dist="gamma",b0=0,b1=1,b2=2,b3=0,rho=0.80)

#Correlation of minus one
y6<- simulatey(x,f,g,dist="normal",b0=0,b1=-1,b2=1,b3=0,rho=1)

#Desired correlation cannot be attained
y7<- simulatey(x,f,g,dist="normal",b0=0,b1=1,b2=3,b3=0,rho=0.99)

#Negative expectation not possible under gamma distribution
y8<- simulatey(x,f,g,dist="gamma",b0=0,b1=-1,b2=1,b3=0,rho=1)

#Conditional variance of zero not possible under gamma distribution
y9<- simulatey(x,f,g,dist="gamma",b0=0,b1=1,b2=3,b3=0,rho=0.99)
```

---

 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 <code>length(x)</code> giving the desired number of strata.
<code>forced</code>	a logical value indicating if the number of strata <i>must</i> 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.

**Examples**

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

---

vargreg

*Design variance of the general regression estimator.*


---

**Description**

Compute the (approximated) design variance of the general regression estimator of the total of a study variable under different sampling designs.

**Usage**

```
vargreg(formula, design = NULL, n, stratum = NULL,
        x_des = NULL, inc.p = NULL, ...)
```

**Arguments**

formula	an object of class <a href="#">formula</a> : a symbolic description of the model to be fitted. The details of model specification are given under ‘Details’.
design	a character string giving the sampling design. It must be one of ‘srs’ (simple random sampling without replacement), ‘poi’ (Poisson sampling), ‘stsi’ (stratified simple random sampling), ‘pips’ (Pareto $\pi$ ps sampling) or NULL (see ‘Details’).
n	either a positive number indicating the (expected) sample size (when design is one of ‘srs’, ‘poi’, ‘pips’ or NULL) or a numeric vector indicating the sample size of the strata to which each element belongs (when design is ‘stsi’) (see ‘Examples’).
stratum	a vector indicating the stratum to which every unit belongs. Only used if design is ‘stsi’.
x_des	a positive numeric vector giving the values of the auxiliary variable that is used for defining the inclusion probabilities. Only used if design is ‘poi’ or ‘pips’.
inc.p	a matrix giving the first and second order inclusion probabilities. Only used if design is NULL.
...	other arguments passed to <a href="#">lm</a> (see ‘Details’).

## Details

The formula should be of the form  $y \sim x$ , where  $y$  is the study variable and  $x$  are the auxiliary variables used by the general regression (GREG) estimator,  $\hat{t}$ . See [formula](#) for more details and ‘Examples’ for typical expressions for some well-known estimators (e.g. the Horvitz-Thompson, ratio, regression and poststratification estimators).

The variance of the GREG estimator is approximated by

$$AV(\hat{t}) = \sum_{k=1}^N \sum_{l=1}^N \pi_{kl} \frac{E_k}{\pi_k} \frac{E_l}{\pi_l} - \left( \sum_{k=1}^N E_k \right)^2$$

where

$$E_k = y_k - \hat{y}_k \text{ and } \hat{y}_k = x_k B \text{ with } B = \left( \sum_{k=1}^N w_k x_k' x_k \right) \sum_{k=1}^N w_k x_k' y_k$$

$N$  is the population size and  $\pi_k$  and  $\pi_{kl}$  are, respectively, the first and second order inclusion probabilities.  $w_k$  is a weight associated to each element and it represents the inverse of the conditional variance (up to a scalar) of the underlying superpopulation model (see ‘Examples’).

If `design=NULL`, the matrix of inclusion probabilities is obtained proportional to the matrix `p.inc`. If `design` is other than `NULL`, the formula for the variance is simplified in such a way that the inclusion probabilities matrix is no longer necessary. In particular:

- if `design='srs'`, only the sample size `n` is required;
- if `design='stsi'`, both the stratum ID `stratum` and the sample size per stratum `n`, are required;
- if `design` is either `'pips'` or `'poi'`, the inclusion probabilities are obtained proportional to the values of `x_des`, corrected if necessary.

## Value

A numeric value giving the variance of the general regression estimator under the desired design.

## References

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

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

## See Also

[desvar](#) for the simultaneous calculation of the variance of six sampling strategies; [expgreg](#) for the expected variance of the GREG estimator under a superpopulation model; [expvar](#) for the simultaneous calculation of the expected variance of five sampling strategies under a superpopulation model; [optimApp](#) for an interactive application of `expgreg`.

## Examples

```

f<- function(x,b0,b1,b2,...) {b0+b1*x^b2}
g<- function(x,b3,...) {x^b3}
x<- 1 + sort( rgamma(5000, shape=4/9, scale=108) )
y<- simulatey(x,f,g,dist="gamma",b0=10,b1=1,b2=1,b3=1,rho=0.95)

st1<- optiallo(n=100,x=x,H=6)
vargreg("y~0",design="srs",n=100) #SRS-HT
vargreg("y~0",design="poi",n=100,x_des=x) #Poi-HT
vargreg("y~0",design="stsi",n=st1$nh,stratum=st1$stratum) #STSI-HT
vargreg("y~0",design="pips",n=100,x_des=x) #PIPS-HT

vargreg("y~x-1",design="srs",n=100,weights=1/x) #SRS-ratio
vargreg("y~x-1",design="poi",n=100,x_des=x,weights=1/x) #Poi-ratio
vargreg("y~x-1",design="stsi",n=st1$nh,
        stratum=st1$stratum,weights=1/x) #STSI-ratio
vargreg("y~x-1",design="pips",n=100,x_des=x,weights=1/x) #PIPS-ratio

vargreg("y~x",design="srs",n=100) #SRS-reg
vargreg("y~x",design="poi",n=100,x_des=x) #Poi-reg
vargreg("y~x",design="stsi",n=st1$nh,stratum=st1$stratum) #STSI-reg
vargreg("y~x",design="pips",n=100,x_des=x) #PIPS-reg

x2<- as.factor(st1$stratum)
vargreg("y~x2",design="srs",n=100) #SRS-pos
vargreg("y~x2",design="poi",n=100,x_des=x) #Poi-pos
vargreg("y~x2",design="stsi",n=st1$nh,stratum=st1$stratum) #STSI-pos
vargreg("y~x2",design="pips",n=100,x_des=x) #PIPS-pos

y2<- c(16,21,18)
x2<- y2
inc.probs<- matrix(c(8,5,4,5,7,3,4,3,6),3,3)
vargreg("y2~0",n=2.1,inc.p=inc.probs) #HT
vargreg("y2~x2-1",n=2.1,inc.p=inc.probs,weights=1/x2) #Ratio
vargreg("y2~x2",n=2.1,inc.p=inc.probs) #Regression
x3<- as.factor(c(1,2,2))
vargreg("y2~x3",n=2.1,inc.p=inc.probs) #Post.

```



# Index

- \* **package**
  - optimStrat-package, 2
- \* **survey**
  - desvar, 2
  - expgreg, 3
  - expvar, 7
  - optiallo, 9
  - optimApp, 10
  - optimStrat-package, 2
  - pinc, 10
  - simulatey, 11
  - stratify, 13
  - vargreg, 14
- \* **univar**
  - skewness, 12

desvar, 2, 6, 8, 15

expgreg, 3, 15

expvar, 3, 6, 7, 15

formula, 14, 15

lm, 4, 14

optiallo, 8, 9, 14

optimApp, 6, 10, 15

optimStrat (optimStrat-package), 2

optimStrat-package, 2

pinc, 10

simulatey, 11

skewness, 12

stratify, 9, 13

vargreg, 6, 14