auditsampling

May 23, 2019

Title A auxiliary package to calculate some basic things in sampling. Most used para CGU auditors.

Version 0.0.0.9000

Description The package contains some functions the help auditors in sampling process.

Depends R (>= 3.4.3)

Imports flextable

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1.9000

R topics documented:

`infereceSRS` *Outputs a table with SRS based inferences*

Description

This function do Simple Random Sample inferences from a sample data, given a confidence level. It can calculate estimates for means and totals and proportions too. It have option to print a Word `*.docx` table with the estimate parameters.

Usage

```
infereceSRS(sample.data, num.cols = NULL, cat.cols = NULL, alpha,
  N = Inf, type = c("mean", "total"), print.report = FALSE,
  labels = NULL)
```

Arguments

`sample.data`	A data.frame representing a sample with the collected data.
`num.cols`	The number of the columns where the data is continuous.
`cat.cols`	The number of the columns where the data is categorical.
`alpha`	Confidence level.
`type`	If you want means or totals.
`print.report`	`FALSE` by default. If `TRUE`, prints a .docx table.
`labels`	An optional string vector of labels to be putted into printed table. Must have the same length of the sum of numerical columns and all levels of categorical columns.

Value

A data.frame containing the fields:

`parameter`	paremeters estimated
`n`	sample size
`cv`	variation coefficient
`point.estimate`	point estimate
`interval`	confidence interval

If the option `print.report=TRUE`, the function will output a `.docx` file with this table.

Examples

```
data('iris')

infereceSRS(iris,num.cols = 1:4,cat.cols = 5,alpha = 0.9, N=56892,
type = "total")

rotulos <- c("Media da espessura da petala","Media do comprimento da petala",
"Media da espessura da sepala","Media do comprimento da sepala",
"Proporcao de setosas","Proporcao de versicolor", "Proporcao de virginica")
infereceSRS(iris,num.cols = 1:4,cat.cols = 5,alpha = 0.95, N=56892,
            type = "mean",labels = rotulos,print.report = TRUE)
```

`nClusterProp` *Calculates the number os cluster to a cluster sample design*

Description

This function returns the number of clusters to be sampled in a proportion based survey. By default, the argument ‘pq‘ is 0.25 (maximum variance).

Usage

```
nClusterProp(data, clustername, alpha, moe, pq = 0.25)
```

Arguments

`data`	The data.frame containing the variable denoted as the cluster.
`clustername`	The variable name of the considered cluster.
`alpha`	(1 - confidence level).
`moe`	Margin of error.
`pq`	The variance, as $P(1-P)$. By default, it is 0.25.

Value

The value (integer) of the number of clusters to sample in cluster sampling.

Examples

```
library(survey)
data("api")
nClusterProp(data=apipop, clustername = "dnum", alpha = 0.05, moe = 0.1)
nClusterProp(data=apipop, clustername = "dnum", alpha = 0.05, moe = 0.05)
```

`nPilot` *Calculates the remaing sample size when using pilot sample*

Description

Based on a pilot sample, this function calculates the remain elements to the final sample given the variance considered.

Usage

```
nPilot(s2 = NULL, pq = NULL, n1, V)
```

Arguments

`pq`	The variance, as $P(1-P)$. Not necessary if ‘S2‘ is not ‘NULL‘.
`n1`	The size of the pilot sample.
`V`	The target variance.
`S2`	The variance, as σ^2. Not necessary if ‘S2‘ is not ‘NULL‘.

Value

The final sample is

$$n = n_1 + n_2$$

Where n_2 is the value returned by this function.

References

COCHRAN, William Gemmell. Sampling techniques-3. 1977.

Examples

```
nPilot(s2=13.5,n1=30,V=1)
nPilot(pq=0.15,n1=30,V=0.025)
```

nSRS *Calculates the sample size for SRS*

Description

This function calculates de sample size in a SRS design based on one of three possible arguments: σ^2, pq or CV. Provide just one of then.

Usage

```
nSRS(moe, alpha, CV = NULL, S2 = NULL, pq = NULL, N = Inf)
```

Arguments

moe	Margin of error.
alpha	1 - (confidence level).
CV	Variaton coefficient.
S2	σ^2, population variance.
pq	$P(1-P)$, population variance for proportions.
N	Population size.

Value

The value of n.

Examples

```
nSRS(moe=0.05,alpha=0.05,S2=6,N=1500)
nSRS(moe=0.05,alpha=0.05,CV=0.2,N=1500)
nSRS(moe=0.05,alpha=0.05,pq=0.25,N=1500)
```

nStrata *Calculates the sample size in a stratified desgin*

Description

This function returns the number of elements to be sampled in a stratified based survey. If the method choosen is ‘prop‘ or ‘optimum‘, the function return the allocation too.

Usage

```
nStrata(data, stratanames, alpha, moe, S2 = NULL, pq = NULL,
  V = NULL, N = Inf, method = c("none", "prop", "optimum"))
```

Arguments

`data`	The data.frame containing the variable denoted as the cluster.
`stratanames`	The variable name of the considered strata. Can be a vector of variable names.
`alpha`	(1 - confidence level).
`moe`	Margin of error.
`S2`	The variance, as σ^2. Not necessary if ‘pq‘ or ‘V‘ is not ‘NULL‘.
`pq`	The variance, as $P(1-P)$. Not necessary if ‘S2‘ or ‘V‘ is not ‘NULL‘.
`V`	The variance, as $(d/t)^2$. Not necessary if ‘S2‘ or ‘pq‘ is not ‘NULL‘.
`N`	Population size.
`method`	A string with ‘none‘, ‘prop‘ or ‘optimum‘. With ‘optimum‘, it uses Neyman allocation.

Value

The value (integer) of the number of elements to sample in stratified sampling. If ‘prop‘ or ‘optimum‘ where used, the function returns a list with the components

n Total number of elements in the sample.

nh In case of method ‘prop‘ or ‘optimum‘, the allocation in each strata.

Examples

```
library(survey)
data("api")
nStrata(apipop,stratanames = "stype", alpha = 0.05, moe = 0.05,
        pq = 0.25, N = Inf, method = "none")
nStrata(apipop,stratanames = "stype", alpha = 0.05, moe = 0.05,
        pq = 0.25, N = nrow(apipop), method = "prop")
nStrata(apipop,stratanames = "stype", alpha = 0.05, moe = 0.05,
        pq = 0.25, N = nrow(apipop), method = "optimum")
### Cochran's example (Cochran, W. G. (2007). Sampling techniques. John Wiley & Sons. p. 106-107)
data_cochran=data.frame(stratum=rep(c(1:6),c(13,18,26,42,73,24)))
nStrata(data_cochran,stratanames = "stratum", N=196, method = "optimum",
S2=c(325^2,190^2,189^2,82^2,86^2,190^2),V=7974976)
```

NSubpop *Calculates the new subpopulation due elements thar not belong to the populational universe*

Description

This function returns the number of the new subpopulation.

Usage

```
NSubpop(N, n, p, z = 3)
```

Arguments

`N` Population size.

`n` Sample size.

`p` Proportion or the number of elements that belong to the universe intended.

`z` The z-value of α.

`data` The data.frame containing the variable denoted as the cluster.

Value

The value (integer) of the new population (subpopulation) number.

Examples

```
NSubpop(45000,100,59)
NSubpop(45000,100,5)
```

Index