R topics documented:

clusOpt2fixedPSU ........................................... 15
clusOpt3 ...................................................... 16
clusOpt3fixedPSU ........................................... 18
CVcalc2 ....................................................... 20
CVcalc3 ....................................................... 21
deff ......................................................... 23
deffCR ......................................................... 25
deffH ......................................................... 27
deffK ......................................................... 29
deffS ......................................................... 30
Domainy1y2 ................................................... 31
dub ........................................................... 32
gamEst ......................................................... 33
gammaFit ....................................................... 34
HMT ........................................................... 35
hospital ....................................................... 36
labor .......................................................... 37
MDarea.pop .................................................... 38
mibrfss ....................................................... 39
nAuditAttr .................................................... 41
nAuditMUS ..................................................... 43
nCont ........................................................... 45
nContMoe ..................................................... 46
nDep2sam ...................................................... 48
nDomain ....................................................... 49
nhis ............................................................ 50
nhis.large ..................................................... 52
nhispart ....................................................... 53
nLogOdds ...................................................... 55
nProp .......................................................... 56
nProp2sam ..................................................... 57
nPropMoe ...................................................... 59
NRadjClass .................................................... 60
NRFUopt ....................................................... 61
nWilson ....................................................... 63
pclass ......................................................... 64
smho.N874 ................................................... 66
smho98 ......................................................... 67
strAlloc ....................................................... 68
ThirdGrade .................................................... 70
wtdvar ......................................................... 71
**Description**

Compute components of relvariance for a sample design where primary sampling units (PSUs) are selected with probability proportional to size (pps) and elements are selected via simple random sampling (srs). The input is an entire sampling frame.

**Usage**

```r
BW2stagePPS(X, pp, psuID, lonely.SSU = "mean")
```

**Arguments**

- `X`: data vector; length is the number of elements in the population.
- `pp`: vector of one-draw probabilities for the PSUs; length is number of PSUs in population.
- `psuID`: vector of PSU identification numbers. This vector must be as long as `X`. Each element in a given PSU should have the same value in `psuID`. PSUs must be in the same order as in `X`.
- `lonely.SSU`: indicator for how singleton SSUs should be handled when computing the within PSU unit relvariance. Allowable values are "mean" and "zero".

**Details**

`BW2stagePPS` computes the between and within population relvariance components appropriate for a two-stage sample in which PSUs are selected with varying probabilities and with replacement. Elements within PSUs are selected by simple random sampling. The components are appropriate for approximating the relvariance of the probability-with-replacement (pwr)-estimator of a total when the same number of elements are selected within each sample PSU. The function requires that an entire frame of PSUs and elements be input.

If a PSU contains multiple SSUs, some of which have missing data, or contains only one SSU, a value is imputed. If `lonely.SSU = "mean"`, the mean of the non-missing PSU contributions is imputed. If `lonely.SSU = "zero"`, a 0 is imputed. The former would be appropriate if a PSU contains multiple SSUs but one or more of them has missing data in which case R will normally calculate an NA. The latter would be appropriate if the PSU contains only one SSU which would be selected with certainty in any sample.

(Use `BW2stagePPSe` if only a sample of PSUs and elements is available.)

**Value**

List object with values:

- `B2`: between PSU unit relvariance
- `W2`: within PSU unit relvariance
unit relvar  unit relvariance for population
B2+W2    sum of between and within relvariance estimates
k        ratio of \(B^2 + W^2\) to unit relvariance
delta    measure of homogeneity with PSUs estimated as \(B^2 / (B^2 + W^2)\)

Author(s)
Richard Valliant, Jill A. Dever, Frauke Kreuter

References

See Also
BW2stagePPSe, BW2stageSRS, BW3stagePPS, BW3stagePPSe

Examples
```r
data(MDarea.pop)
MDsub <- MDarea.pop[1:100000,]
  # Use PSU and SSU variables to define psu's
pp.PSU <- table(MDsub$PSU) / nrow(MDsub)
pp.SSU <- table(MDsub$SSU) / nrow(MDsub)
  # components with psu's defined by the PSU variable
BW2stagePPS(MDsub$y1, pp=pp.PSU, psuID=MDsub$PSU, lonely.SSU="mean")
  # components with psu's defined by the SSU variable
BW2stagePPS(MDsub$y1, pp=pp.SSU, psuID=MDsub$SSU, lonely.SSU="mean")

  # Use census tracts and block groups to define psu's
trtBG <- 10*MDsub$TRACT + MDsub$BLKGROUP
pp.trt <- table(MDsub$TRACT) / nrow(MDsub)
pp.BG <- table(trtBG) / nrow(MDsub)
  # components with psu's defined by tracts
BW2stagePPS(MDsub$ins.cov, pp=pp.trt, psuID=MDsub$TRACT, lonely.SSU="mean")
  # components with psu's defined by block groups
BW2stagePPS(MDsub$ins.cov, pp=pp.BG, psuID=trtBG, lonely.SSU="mean")
```
Description

Estimate components of relvariance for a sample design where primary sampling units (PSUs) are selected with \textit{pps} and elements are selected via \textit{srs}. The input is a sample selected in this way.

Usage

\texttt{BW2stagePPSe(Ni, ni, X, psuID, w, m, pp, lonely.SSU = "mean")}

Arguments

- \texttt{Ni} vector of number of elements in the population of each sample PSU; length is the number of PSUs in the sample.
- \texttt{ni} vector of number of sample elements in each sample PSU; length is the number of PSUs in the sample. PSUs must be in the same order in \texttt{ni} and in \texttt{X}.
- \texttt{X} data vector for sample elements; length is the number of elements in the sample. These must be in PSU order. PSUs must be in the same order in \texttt{ni} and in \texttt{X}.
- \texttt{psuID} vector of PSU identification numbers. This vector must be as long as \texttt{X}. Each element in a given PSU should have the same value in \texttt{psuID}.
- \texttt{w} vector of full sample weights. This vector must be as long as \texttt{X}. Vector must be in the same order as \texttt{X}.
- \texttt{m} number of sample PSUs
- \texttt{pp} vector of 1-draw probabilities for the PSUs. The length of this vector is the number of PSUs in the sample. Vector must be in the same order as \texttt{Ni} and \texttt{ni}.
- \texttt{lonely.SSU} indicator for how singleton SSUs should be handled when computing the within PSU unit relvariance. Allowable values are "mean" and "zero".

Details

\texttt{BW2stagePPSe} computes the between and within population variance and relvariance components appropriate for a two-stage sample in which PSUs are selected with varying probabilities and with replacement. Elements within PSUs are selected by simple random sampling. The number of elements selected within each sample PSU can vary but must be at least two. The estimated components are appropriate for approximating the relvariance of the \textit{pwr}-estimator of a total when the same number of elements are selected within each sample PSU. This function can also be used if PSUs are selected by \textit{srswr} by appropriate definition of \texttt{pp}.

If a PSU contains multiple SSUs, some of which have missing data, or contains only one SSU, a value is imputed. If \texttt{lonely.SSU = "mean"}, the mean of the non-missing PSU contributions is imputed. If \texttt{lonely.SSU = "zero"}, a 0 is imputed. The former would be appropriate if a PSU contains multiple SSUs but one or more of them has missing data in which case R will normally calculate an NA. The latter would be appropriate if the PSU contains only one SSU which would be selected with certainty in any sample.
**Value**

List with values:

- **Vpsu**: estimated between PSU unit variance
- **Vssu**: estimated within PSU unit variance
- **B**: estimated between PSU unit relvariance
- **W**: estimated within PSU unit relvariance
- **k**: estimated ratio of **B+W** to estimated unit relvariance of the analysis variable
- **delta**: intraclass correlation estimated as **B/(B+W)**

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**


**See Also**

`BW2stagePPS, BW2stageSRS, BW3stagePPS, BW3stagePPSe`

**Examples**

```r
## Not run:
require(sampling)
require(plyr) # has function that allows renaming variables
data(MDarea.pop)
Ni <- table(MDarea.pop$TRACT)
m <- 20
probi <- m*Ni / sum(Ni)
# select sample of clusters
sam <- cluster(data=MDarea.pop, clustername="TRACT", size=m, method="systematic",
pik=probi, description=TRUE)
# extract data for the sample clusters
samclus <- getdata(MDarea.pop, sam)
samclus <- rename(samclus, c(Prob = "pi1"))

# treat sample clusters as strata and select srswor from each
s <- strata(data = as.data.frame(samclus), stratanames = "TRACT",
size = rep(50,m), method="srswor")
# extracts the observed data
samdat <- getdata(samclus,s)
samdat <- rename(samdat, c(Prob = "pi2"))

# extract pop counts for PSUs in sample
```
pick <- names(Ni) %in% sort(unique(samdat$TRACT))
Ni.sam <- Ni[pick]
pp <- Ni.sam / sum(Ni)
wt <- 1/samdat$pi1/samdat$pi2

BW2stagePPSe(Ni = Ni.sam, ni = rep(50,20), X = samdat$y1,
            psuID = samdat$TRACT, w = wt,
            m = 20, pp = pp, lonely.SSU="mean")

## End(Not run)

---

**BW2stageSRS**  
Relvariance components for 2-stage sample

**Description**

Compute components of relvariance for a sample design where primary sampling units (PSUs) and elements are selected via *srs*. The input is an entire sampling frame.

**Usage**

BW2stageSRS(X, psuID, lonely.SSU)

**Arguments**

- **X**  
data vector; length is the number of elements in the population.

- **psuID**  
vector of PSU identification numbers. This vector must be as long as X. Each element in a given PSU should have the same value in psuID. PSUs must be in the same order as in X.

- **lonely.SSU**  
indicator for how singleton SSUs should be handled when computing the within PSU unit relvariance. Allowable values are "mean" and "zero".

**Details**

**BW2stageSRS** computes the between and within population relvariance components appropriate for a two-stage sample in which PSUs are selected via *srs* (either with or without replacement). Elements within PSUs are assumed to be selected by *srswor*. The same number of elements is assumed to be selected within each sample PSU. The function requires that an entire frame of PSUs and elements be input.

If a PSU contains multiple SSUs, some of which have missing data, or contains only one SSU, a value is imputed. If lonely.SSU = "mean", the mean of the non-missing PSU contributions is imputed. If lonely.SSU = "zero", a 0 is imputed. The former would be appropriate if a PSU contains multiple SSUs but one or more of them has missing data in which case R will normally calculate an NA. The latter would be appropriate if the PSU contains only one SSU which would be selected with certainty in any sample.

(Use **BW2stagePPSe** if only a sample of PSUs and elements are available.)
Value

List with values:

- B2: between PSU unit relvariance
- W2: within PSU unit relvariance
- unit relvar: unit relvariance for population
- B2+W2: \( B^2 + W^2 \)
- k: ratio of \( B^2 + W^2 \) to unit relvariance
- delta full: intraclass correlation estimated as \( \frac{B^2}{(B^2 + W^2)} \)

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

- `BW2stagePPS`, `BW2stagePPSe`, `BW3stagePPS`, `BW3stagePPSe`

Examples

```r
data(MDarea.pop)
MDsub <- MDarea.pop[1:100000,]
# psu's are defined by PSU variable
BW2stageSRS(abs(MDsub$Hispanic-2), psuID=MDsub$PSU, lonely.SSU="mean")
# psu's are defined by SSU variable
BW2stageSRS(abs(MDsub$Hispanic-2), psuID=MDsub$SSU, lonely.SSU="mean")
```

Description

Compute components of relvariance for a sample design where primary sampling units (PSUs) are selected with `ppswr` and secondary sampling units (SSUs) and elements within SSUs are selected via `srs`. The input is an entire sampling frame.

Usage

```r
BW3stagePPS(X, pp, psuID, ssuID, lonely.SSU = "mean", lonely.TSU = "mean")
```
Arguments

**X**
data vector; length is the number of elements in the population.

**pp**
vector of one-draw probabilities for the PSUs; length is number of PSUs in population.

**psuID**
vector of PSU identification numbers. This vector must be as long as **X**. Each element in a given PSU should have the same value in psuID. PSUs must be in the same order as in **X**.

**ssuID**
vector of SSU identification numbers. This vector must be as long as **X**. Each element in a given SSU should have the same value in ssuID. PSUs and SSUs must be in the same order as in **X**. ssuID should have the form psuID||ssuID within PSU.

**lonely.SSU**
indicator for how singleton SSUs should be handled when computing the within PSU unit relvariance. Allowable values are "mean" and "zero".

**lonely.TSU**
indicator for how singleton third-stage units (TSUs) should be handled when computing the within SSU unit relvariance. Allowable values are "mean" and "zero".

Details

**BW3stagePPS** computes the between and within population relvariance components appropriate for a three-stage sample in which PSUs are selected with varying probabilities and with replacement. SSUs and elements within SSUs are selected by simple random sampling. The components are appropriate for approximating the relvariance of the pwr-estimator of a total when the same number of SSUs are selected within each PSU, and the same number of elements are selected within each sample SSU. The function requires that an entire sampling frame of PSUs and elements be input.

If a PSU contains multiple SSUs, some of which have missing data, or contains only one SSU, a value is imputed. If **lonely.SSU** = "mean", the mean of the non-missing PSU contributions is imputed. If **lonely.SSU** = "zero", a 0 is imputed. The former would be appropriate if a PSU contains multiple SSUs but one or more of them has missing data in which case R will normally calculate an NA. The latter would be appropriate if the PSU contains only one SSU which would be selected with certainty in any sample. **lonely.TSU** has a similar purpose for third-stage units.

(Use **BW2stagePPSe** if only a sample of PSUs, SSUs, and elements is available.)

Value

List with values:

- **B**
between PSU unit relvariance
- **W**
within PSU unit relvariance computed as if the sample were two-stage
- **W2**
unit relvariance among SSU totals
- **W3**
unit relvariance among elements within PSU/SSUs
- **unit relvar**
unit relvariance for population
- **k1**
ratio of $B^2 + W^2$ to unit relvariance
- **k2**
ratio of $W_2^2 + W_3^2$ to unit relvariance
delta1  homogeneity measure among elements within PSUs estimated as $B^2/(B^2 + W^2)$

delta2  homogeneity measure among elements within SSUs estimated as $W_2^2/(W_2^2 + W_3^2)$

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

`BW2stagePPS, BW2stagePPSe, BW2stageSRS, BW3stagePPSe`

Examples

```r
data(MDarea.pop)
MDsub <- MDarea.pop[1:100000,]
M <- length(unique(MDsub$PSU))
  # srs/srs/srs design
pp.PSU <- rep(1/M, M)
BW3stagePPS(X=MDsub$y1, pp=pp.PSU, psuID=MDsub$PSU, ssuID=MDsub$SSU,
          lonely.SSU = "mean", lonely.TSU = "mean")
  # ppswr/srs/srs design
pp.PSU <- table(MDsub$PSU) / nrow(MDsub)
BW3stagePPS(X=MDsub$y1, pp=pp.PSU, psuID=MDsub$PSU, ssuID=MDsub$SSU,
          lonely.SSU = "mean", lonely.TSU = "mean")
```

Description

Estimate components of relvariance for a sample design where primary sampling units (PSUs) are selected with probability proportional to size with replacement (ppswr) and secondary sampling units (SSUs) and elements within SSUs are selected via simple random sampling (srs). The input is a sample selected in this way.

Usage

`BW3stagePPSe(dat, v, Ni, Qi, Qij, m, lonely.SSU = "mean", lonely.TSU = "mean")`
Arguments

dat  data frame for sample elements with PSU and SSU identifiers, weights, and analysis variable(s). The data frame should be sorted in hierarchical order: by PSU and SSU within PSU. Required names for columns: psuID = PSU identifier; ssuID = SSU identifier. These must be unique, i.e., numbering should not restart within each PSU. Setting ssuID = psuID || (ssuID within PSU) is a method of doing this. w1i = vector of weights for PSUs; w2ij = vector of weights for SSUs (PSU weight * SSU weight within PSU); w = full sample weight

v  Name or number of column in data frame dat with variable to be analyzed.

N  m-vector of number of SSUs in the population in the sample PSUs; m is number of sample PSUs.

Q  m-vector of number of elements in the population in the sample PSUs

Qij  vector of numbers of elements in the population in the sample SSUs

m  number of sample PSUs

lonely.SSU  indicator for how singleton SSUs should be handled when computing the within PSU unit relvariance. Allowable values are "mean" and "zero".

lonely.TSU  indicator for how singleton third-stage units (TSUs) should be handled when computing the within SSU unit relvariance. Allowable values are "mean" and "zero".

Details

BW3stagePPSe computes the between and within population relvariance components appropriate for a three-stage sample in which PSUs are selected with varying probabilities and with replacement. SSUs and elements within SSUs are selected by simple random sampling. The estimated components are appropriate for approximating the relvariance of the pwr-estimator of a total when the same number of SSUs are selected within each PSU, and the same number of elements are selected within each sample SSU.

If a PSU contains multiple SSUs, some of which have missing data, or contains only one SSU, a value is imputed. If lonely.SSU = "mean", the mean of the non-missing PSU contributions is imputed. If lonely.SSU = "zero", a 0 is imputed. The former would be appropriate if a PSU contains multiple SSUs but one or more of them has missing data in which case R will normally calculate an NA. The latter would be appropriate if the PSU contains only one SSU which would be selected with certainty in any sample. lonely.TSU has a similar purpose for third-stage units.

Value

List with values:

Vpsu  estimated between PSU unit variance

Vssu  estimated second-stage unit variance among SSU totals

Vtsu  estimated third-stage unit variance

B  estimated between PSU unit relvariance

W  estimated within PSU unit relvariance computed as if the sample were two-stage

k1  estimated ratio of B+W to estimated unit relvariance of the analysis variable
$W_2$ estimated unit relvariance among SSU totals

$W_3$ estimated third-stage unit relvariance among elements within PSU/SSUs

$k_2$ estimated ratio of $W_2 + W_3$ to estimated unit relvariance of the analysis variable

$\delta_1$ homogeneity measure among elements within PSUs estimated as $B^2/(B^2 + W^2)$

$\delta_2$ homogeneity measure among elements within SSUs estimated as $W_2^2/(W_2^2 + W_3^2)$

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

`BW2stagePPS, BW2stagePPSe, BW2stageSRS, BW3stagePPS`

Examples

```r
## Not run:
# select 3-stage sample from Maryland population
set.seed(-780087528)
data(MDarea.pop)
MDpop <- MDarea.pop
require(sampling)
require(plyr)  # has function that allows renaming variables
# make counts of SSUs and elements per PSU
xx <- do.call("rbind",list(by(1:nrow(MDpop),MDpop$SSU,head,1)))
pop.tmp <- MDpop[xx,
Ni <- table(pop.tmp$PSU)
Qi <- table(MDarea.pop$PSU)
Qij <- table(MDpop$SSU)
m <- 30  # no. of PSUs to select
probi <- m*Qi / sum(Qi)
# select sample of clusters
sam <- cluster(data=MDpop, clustername="PSU", size=m, method="systematic",
pik=probi, description=TRUE)
# extract data for the sample clusters
samclus <- getdata(MDpop, sam)
samclus <- rename(samclus, c(Prob = "p1i"))
samclus <- samclus[order(samclus$PSU),]
# treat sample clusters as strata and select srswor of block groups from each
# identify psu IDs for 1st instance of each ssuid
xx <- do.call("rbind",list(by(1:nrow(samclus),samclus$SSU,head,1)))
```

SSUs <- cbind(PSU=samclus$PSU[xx], SSU=samclus$SSU[xx])
# select 2 SSUs per tract
n <- 2
s <- strata(data = as.data.frame(SSUs), stratanames = "PSU",
    size = rep(n,m), method="srswor")
s <- rename(s, c(Prob = "p2i"))
# extract the SSU data
# s contains selection probs of SSUs, need to get those onto data file
SSUsam <- SSUs[s$ID_unit, ]
SSUsam <- cbind(SSUsam, s[, 2:3])
# identify rows in PSU sample that correspond to sample SSUs
tmp <- samclus$SSU %in% SSUsam$SSU
SSUdat <- samclus[tmp,]
SSUdat <- merge(SSUdat, SSUsam[, c("p2i","SSU")], by="SSU")
# select srswor from each sample SSU
n.SSU <- m*n
s <- strata(data = as.data.frame(SSUdat), stratanames = "SSU",
    size = rep(50,n.SSU), method="srswor")
s <- rename(s, c(Prob = "p3i"))
samclus <- getdata(SSUdat, s)
del <- (1:ncol(samclus))[dimnames(samclus)[[2]] %in% c("ID_unit","Stratum")]
samclus <- samclus[, -del]
# extract pop counts for PSUs in sample
pick <- names(Qi) %in% sort(unique(samclus$PSU))
Qi.sam <- Qi[pick]
# extract pop counts of SSUs for PSUs in sample
pick <- names(Ni) %in% sort(unique(samclus$PSU))
Ni.sam <- Ni[pick]
# extract pop counts for SSUs in sample
pick <- names(Qij) %in% sort(unique(samclus$SSU))
Qij.sam <- Qij[pick]
# compute full sample weight and wts for PSUs and SSUs
wt <- 1 / samclus$p1i / samclus$p2i / samclus$p3i
w1i <- 1 / samclus$p1i
w2ij <- 1 / samclus$p1i / samclus$p2i
samdat <- data.frame(psuID = samclus$PSU, ssuID = samclus$SSU,
    sampleID = samclus$SSU,
    w1i = w1i, w2ij = w2ij, w = wt, w5i = w5i, w6ij = w6ij, w7i = w7i, w = w,
    samclus[, c("y1","y2","y3","ins.cov","hosp.stay")])
BW3stagePPSe(dat=samdat, v="y1", Ni=Ni.sam, Qi=Qi.sam, Qij=Qij.sam, m,
    lonely.SSU = "mean", lonely.TSU = "mean")

## End(Not run)

clusOpt2

**Compute optimal sample sizes for a two-stage sample**

**Description**

Compute the sample sizes that minimize the variance of the pwr-estimator of a total in a two-stage sample.
Usage
clusOpt2(C1, C2, delta, unit.rv, k=1, CV0=NULL, tot.cost=NULL, cal.sw)

Arguments
- **C1**: unit cost per primary sampling unit (PSU)
- **C2**: unit cost per element
- **delta**: homogeneity measure \( \delta \)
- **unit.rv**: unit relvariance
- **k**: ratio of \( B^2 + W^2 \) to unit relvariance
- **CV0**: target CV
- **tot.cost**: total budget for variable costs
- **cal.sw**: specify type of optimum: 1 = find optimal \( m.\text{opt} \) for fixed total budget; 2 = find optimal \( m.\text{opt} \) for target \( CV_0 \)

Details
clusOpt2 will compute \( m.\text{opt} \) and \( n.\text{opt} \) for a two-stage sample which uses simple random sampling at each stage or ppswr at the first stage and srs at the second.

Value
List with values:
- **C1**: unit cost per PSU
- **C2**: unit cost per element
- **delta**: homogeneity measure
- **unit.rv**: unit relvariance
- **k**: ratio of \( B^2 + W^2 \) to unit relvariance
- **cost**: total budget for variable costs, \( C - C_0 \) if cal.sw=1; or computed cost if cal.sw=2
- **m.opt**: optimum number of sample PSUs
- **n.opt**: optimum number of sample elements per PSU
- **CV**: computed CV if cal.sw=1; or target CV if cal.sw=2

Author(s)
Richard Valliant, Jill A. Dever, Frauke Kreuter

References

clusOpt2fixedPSU

See Also

clusOpt2fixedPSU, clusOpt3, clusOpt3fixedPSU

Examples

# optimum for a fixed total budget
clusOpt2(C1=750, C2=100, delta=0.05, unit.rv=1, k=1, tot.cost=100000, cal.sw=1)
# optimum for a target CV
clusOpt2(C1=750, C2=100, delta=0.01, unit.rv=1, k=1, CV0=0.05, cal.sw=2)

clusOpt2fixedPSU  Optimal number of sample elements per PSU in a two-stage sample when the sample of PSUs is fixed

Description

Compute the optimum number of sample elements per primary sampling unit (PSU) for a fixed set of PSUs

Usage

clusOpt2fixedPSU(C1, C2, m, delta, unit.rv, k=1, CV0=NULL, tot.cost, cal.sw)

Arguments

C1  unit cost per PSU  
C2  unit cost per element  
m  number of sample PSU’s (fixed)  
delta  homogeneity measure  
unit.rv  unit relvariance  
k  ratio of $B^2 + W^2$ to unit relvariance  
CV0  target CV  
tot.cost  total budget for variable costs  
cal.sw  specify type of optimum: 1 = find optimal $\bar{n}$ for fixed total budget; 2 = find optimal $\bar{n}$ for target CV0

Details

clusOpt2fixedPSU will compute $\bar{n}_{opt}$ for a two-stage sample which uses simple random sampling at each stage or ppswr at the first stage and srs at the second. The PSU sample is fixed.
Value

List with values:

- **C1**: unit cost per PSU
- **C2**: unit cost per element
- **m**: number of (fixed) sample PSUs
- **delta**: homogeneity measure
- **unit relvar**: unit relvariance
- **k**: ratio of $B^2 + W^2$ to unit relvariance
- **cost**: total budget for variable costs, $C - C_0$ if cal.sw=1; or computed cost if cal.sw=2
- **n**: optimum number of sample elements per PSU
- **CV**: computed CV if cal.sw=1; or target CV if cal.sw=2

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

clusOpt2, clusOpt3, clusOpt3fixedPSU

Examples

```r
# optima for a vector of budgets
clusOpt2fixedPSU(C1=500, C2=100, m=100, delta=0.05, unit.rv=2, k=1, CV0=NULL, 
tot.cost=c(100000, 500000, 10^6), cal.sw=1)
# optima for a target CV and vector of PSU costs
clusOpt2fixedPSU(C1=c(500,1000,5000), C2=100, m=100, delta=0.05, unit.rv=2, k=1, 
CV0=0.05, tot.cost=NULL, cal.sw=2)
```

clusOpt3

*Compute optimal sample sizes for a three-stage sample*

Description

Compute the sample sizes that minimize the variance of the *pwr*-estimator of a total in a three-stage sample.

Usage

```r
clusOpt3(unit.cost, delta1, delta2, unit.rv, k1=1, k2=1, CV0=NULL, tot.cost=NULL, cal.sw)
```
clusOpt3

Arguments

**unit.cost** vector with three components for unit costs: $C_1 =$ unit cost per primary sampling unit (PSU); $C_2 =$ unit cost per secondary sampling unit (SSU); $C_3 =$ unit cost per element

**delta1** homogeneity measure among elements within PSUs

**delta2** homogeneity measure among elements within SSUs

**unit.rv** population unit relvariance

**k1** ratio of $B^2 + W^2$ to the population unit relvariance

**k2** ratio of $W_2^2 + W_3^2$ to the population unit relvariance

**CV0** target CV

**tot.cost** total budget for variable costs

**cal.sw** specify type of optimum: 1 = find optimal $m$.opt for fixed total budget; 2 = find optimal $m$.opt for target CV0

Details

clusOpt3 will compute $m_{opt}$, $\bar{n}_{opt}$, and $\bar{q}_{opt}$ for a three-stage sample which uses simple random sampling at each stage or ppswr at the first stage and srs at the second and third stages.

Value

List with values:

**C1** unit cost per PSU

**C2** unit cost per SSU

**C3** unit cost per element

**delta1** homogeneity measure among elements within PSUs

**delta2** homogeneity measure among elements within SSUs

**unit.relvar** unit relvariance

**k1** ratio of $B^2 + W^2$ to the population unit relvariance

**k2** ratio of $W_2^2 + W_3^2$ to the population unit relvariance

**cost** total budget for variable costs if cal.sw=1; or computed cost if cal.sw=2

**m.opt** optimum number of sample PSUs

**n.opt** optimum number of sample SSUs per PSU

**q.opt** optimum number of sample elements per SSU

**CV** achieved CV if cal.sw=1 or target CV if cal.sw=2

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter
References


See Also

clusOpt2, clusOpt2fixedPSU, clusOpt3fixedPSU

Examples

# optima for a fixed total budget
clusOpt3(unit.cost=c(500, 100, 120), delta1=0.01, delta2=0.10, unit.rv=1, k1=1, k2=1, tot.cost=100000, cal.sw=1)

# optima for a target CV
clusOpt3(unit.cost=c(500, 100, 120), delta1=0.01, delta2=0.10, unit.rv=1, k1=1, k2=1, CV0=0.01, cal.sw=2)

clusOpt3fixedPSU

Compute optimal number of sample secondary sampling units (SSUs) and elements per SSU for a fixed set of primary sampling units (PSUs) in a three-stage sample

Description

Compute the sample sizes that minimize the variance of the pwr-estimator of a total in a three-stage sample when the PSU sample is fixed.

Usage

clusOpt3fixedPSU(unit.cost, m, delta1, delta2, unit.rv, k1=1, k2=1, CV0=NULL, tot.cost=NULL, cal.sw)

Arguments

unit.cost 3-vector of unit costs: C1 = unit cost per PSU; C2 = unit cost per SSU; C3 = unit cost per element
m number of sample PSUs (fixed)
delta1 homogeneity measure among elements within PSUs
delta2 homogeneity measure among elements within SSUs
unit.rv unit relvariance
k1 ratio of $B^2 + W^2$ to unit relvariance
k2 ratio of $W_2^2 + W_3^2$ to unit relvariance
CV0 target CV
clusOpt3fixedPSU

**tot.cost**
- total budget for variable costs, including PSU costs

**cal.sw**
- specify type of optimum: 1 = find optimal $m_{opt}$ for fixed total budget; 2 = find optimal $m_{opt}$ for target CV0

**Details**
clusOpt3 will compute $\bar{n}_{opt}$ and $\bar{q}_{opt}$ for a three-stage sample which uses simple random sampling at each stage or ppswr at the first stage and srs at the second and third stages. The set of sample PSUs is assumed to be fixed. "Variable costs" in tot.cost includes the budget for all costs that vary with the number of sample PSUs, SSUs, and elements, i.e., $C_1 m + C_2 \bar{n} + C_3 \bar{q}$.

**Value**
- List with values:
  - C1: unit cost per PSU
  - C2: unit cost per SSU
  - C3: unit cost per element
  - $m$: number of sample PSUs (fixed)
  - delta1: homogeneity measure among elements within PSUs
  - delta2: homogeneity measure among elements within SSUs
  - unit relvar: unit relvariance
  - k1: ratio of $B^2 + W^2$ to unit relvariance
  - k2: ratio of $W^2_2 + W^2_3$ to unit relvariance
  - cost: budget constraint, tot.cost if cal.sw=1; computed cost if cal.sw=2
  - n: optimum number of sample SSUs per PSU
  - q: optimum number of sample elements per PSU
  - CV: achieved CV, used if cal.sw=1; or target CV, used if cal.sw=2
  - CV check: computed CV based on optimal sample sizes; used only if cal.sw=2

**Author(s)**
- Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**

**See Also**
- clusOpt2, clusOpt2fixedPSU, clusOpt3
Examples

```r
# optima for a fixed total budget
clusOpt3fixedPSU(unit.cost=c(500, 100, 120), m=100, delta1=0.01, delta2=0.05, unit.rv=1,
                  k1=1, k2=1, tot.cost=5000000, cal.sw=1)
# optima for a target CV
clusOpt3fixedPSU(unit.cost=c(500, 100, 120), m=100, delta1=0.01, delta2=0.05, unit.rv=1,
                  k1=1, k2=1, CV0=0.05, cal.sw=2)
```

CVcalc2

**Coefficient of variation of an estimated total in a 2-stage sample**

**Description**

Compute the coefficient of variation of an estimated total in a two-stage design. Primary sampling units (PSUs) can be selected either with probability proportional to size (pps) or with equal probability. Elements are selected via simple random sampling (srs).

**Usage**

```r
CVcalc2(V=NULL, m=NULL, nbar=NULL, k=1, delta=NULL, Bsq=NULL, Wsq=NULL)
```

**Arguments**

- `V`: unit relvariance of analysis variable in the population
- `m`: number of sample PSUs
- `nbar`: number of sample elements per PSU
- `k`: ratio of $B^2 + W^2$ to $V$. Default value is 1.
- `delta`: measure of homogeneity equal to $B^2/(B^2 + W^2)$
- `Bsq`: unit relvariance of PSU totals
- `Wsq`: within PSU relvariance

**Details**

`CVcalc2` computes the coefficient of variation of an estimated total for a two-stage sample. PSUs can be selected either with varying probabilities and with replacement or with equal probabilities and with replacement. Elements within PSUs are selected by simple random sampling. The CV formula is appropriate for approximating the relvariance of the probability-with-replacement (pwr)-estimator of a total when the same number of elements is selected within each sample PSU. See Sections 9.2.1–9.2.3 of Valliant, Dever, and Kreuter (2013) for details of formulas.

**Value**

Value of the coefficient of variation of an estimated total

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter
References


See Also

CVcalc3

Examples

CVcalc2(V=1, m=20 , nbar=5, k=1, delta=0.05)
CVcalc2(V=10, m=20 , nbar=5, k=1, delta=0.5)
CVcalc2(V=2.5, m=20 , nbar=5, k=2, Bsq=1, Wsq=4)

CVcalc3 Coefficient of variation of an estimated total in a 3-stage sample

Description

Compute the coefficient of variation of an estimated total in a three-stage design. Primary sampling units (PSUs) can be selected either with probability proportional to size (pps) or with equal probability. Secondary units and elements within SSUs are selected via simple random sampling (srs).

Usage

CVcalc3(V=NULL, m=NULL , nbar=NULL, qbar=NULL, k1=1, k2=1, delta1=NULL, delta2=NULL, Bsq=NULL, Wsq=NULL, W2sq=NULL, W3sq=NULL)

Arguments

V unit relvariance of analysis variable in the population
m number of sample PSUs
nbar number of sample secondary units per PSU
qbar number of sample elements per SSU
k1 ratio of $B^2 + W^2$ to $V$. Default value is 1.
k2 ratio of $W_2^2 + W_3^2$ to $V$. Default value is 1.
delta1 measure of homogeneity between PSUs equal to $B^2/(B^2 + W^2)$
delta2 measure of homogeneity between SSUs within PSUs, equal to $W_2^2/(W_2^2 + W_3^2)$
Bsq unit relvariance of PSU totals, equal to population variance of totals divided by $\bar{t}_U^2$
CVcalc3 computes the coefficient of variation of an estimated total for a three-stage sample. PSUs can be selected either with varying probabilities and with replacement or with equal probabilities and with replacement. SSUs and elements within SSUs are selected by simple random sampling. The \( CV \) formula is appropriate for approximating the relvariance of the probability-with-replacement (\( pwr \))-estimator of a total when the same number of SSUs is selected in each PSU and the same number of elements is selected within each sample SSU. See Sect. 9.2.4 of Valliant, Dever, and Kreuter (2018) for details of formulas.

Value

Value of the coefficient of variation of an estimated total

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

CVcalc3

Examples

```
CVcalc3(V=1, m=20, nbar=5, qbar=10, delta1=0.02, delta2=0.10)
CVcalc3(V=1, m=20, nbar=5, qbar=10, delta1=0.02, delta2=0.09)
CVcalc3(V=2, m=20, nbar=5, qbar=10, k1=5, k2=10, Bsq=1, Wsq=9, W2sq=2, W3sq=18)
```
**deff**

*Design effects of various types*

**Description**

Compute the Kish, Henry, Spencer, or Chen-Rust design effects.

**Usage**

```
deff(w, x=NULL, y=NULL, p=NULL, strvar=NULL, clvar=NULL, Wh=NULL, nest=FALSE, type)
```

**Arguments**

- `w`: vector of weights for a sample
- `x`: matrix of covariates used to construct a GREG estimator of the total of `y`. This matrix does not include the intercept. Used only for Henry `deff`.
- `y`: vector of the sample values of an analysis variable
- `p`: vector of 1-draw selection probabilities, i.e., the probability that each unit would be selected in a sample of size 1. Used only for Spencer `deff`.
- `strvar`: vector of stratum identifiers; equal in length to that of `w`. Used only for Chen-Rust `deff`.
- `clvar`: vector of cluster identifiers; equal in length to that of `w`. Used only for Chen-Rust `deff`.
- `Wh`: vector of the proportions of elements that are in each stratum; length is number of strata. Used only for Chen-Rust `deff`.
- `nest`: Are cluster IDs numbered within strata (TRUE or FALSE)? If TRUE, cluster IDs can be restarted within strata, e.g., 1,2,3,1,2,3,...
- `type`: type of allocation; must be one of "kish", "henry", "spencer", "cr"

**Details**

deff calls one of deffK, deffH, deffS, or deffCR depending on the value of the type parameter.

The Kish design effect is the ratio of the variance of an estimated mean in stratified simple random sampling without replacement (stsrswor) to the variance of the estimated mean in srswor, assuming that all stratum unit variances are equal. In that case, proportional allocation with equal weighting is optimal. deffK equals 1 + relvar(w) where relvar is relvariance of the vector of survey weights. This measure is not appropriate in samples where unequal weighting is more efficient than equal weighting.

The Henry design effect is the ratio of the variance of the general regression (GREG) estimator of a total of `y` to the variance of the estimated total in srswr. Calculations for the Henry `deff` are done as if the sample is selected in a single-stage and with replacement. Varying selection probabilities can be used. The model for the GREG is assumed to be \( y = \alpha + \beta x + \epsilon \), i.e., the model has an intercept.
The Spencer design effect is the ratio of the variance of the \textit{pwr}-estimator of the total of \( y \), assuming that a single-stage sample is selected with replacement, to the variance of the total estimated in \textit{srswr}. Varying selection probabilities can be used.

The Chen-Rust \textit{deff} accounts for stratification, clustering, and unequal weights, but does not account for the use of any auxiliary data in the estimator of a mean. The Chen-Rust \textit{deff} returned here is appropriate for stratified, two-stage sampling.

**Value**

Numeric design effect for types \textit{kish}, \textit{henry}, \textit{spencer}. For type \textit{cr} a list with components:

- \texttt{strata components}
  - Matrix with \textit{deff}'s due to weighting, clustering, and stratification for each stratum
- \texttt{overall deff}
  - Design effect for full sample accounting for weighting, clustering, and stratification

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**


**See Also**

\texttt{deffK, deffH, deffS, deffCR}

**Examples**

```r
require(reshape) # has function that allows renaming variables
require(sampling)

set.seed(-500398777)
# generate population using HMT function
```
deffCR <- as.data.frame(HMT())
mos <- pop.dat$x
pop.dat$prbs.1d <- mos / sum(mos)
# select pps sample
n <- 80
pk <- n * pop.dat$prbs.1d
sam <- UPrandomsystematic(pk)
sam <- sam==1

sam.dat <- pop.dat[sam, ]
dsgn.wts <- 1/pk[sam]
deff(w=dsgn.wts, type="kish")
deff(w=dsgn.wts, y=sam.dat$y, p=sam.dat$prbs.1d, type="spencer")
deff(w=dsgn.wts, x=sam.dat$x, y=sam.dat$y, type="henry")

data(MDarea.pop)
Ni <- table(MDarea.pop$TRACT)
m <- 10
probi <- m*Ni / sum(Ni)
# select sample of clusters
set.seed(-780087528)
sam <- cluster(data=MDarea.pop, clustername="TRACT", size=m, method="systematic",
pik=probi, description=TRUE)
# extract data for the sample clusters
samclus <- getdata(MDarea.pop, sam)
samclus <- rename(samclus, c(Prob = "pi1"))
# treat sample clusters as strata and select srswor from each
tbar <- 4
s <- strata(data = as.data.frame(samclus), stratanames="TRACT",
size = rep(tbar,m), method="srswor")
# extracts the observed data
samdat <- getdata(samclus,s)
samdat <- rename(samdat, c(Prob = "pi2"))
# add a fake stratum ID
H <- 2
nh <- m * tbar / H
stratum <- NULL
for (h in 1:H){
  stratum <- c(stratum, rep(h,nh))
}
wt <- 1/(samdat$pi1*samdat$pi2) * runif(m*tbar)
samdat <- cbind(subset(samdat, select = -c(Stratum)), stratum, wt)
deff(w = samdat$wt, y=samdat$y2, strvar = samdat$stratum, clvar = samdat$TRACT, Wh=NULL, type="cr")

deffCR  

Chen-Rust design effect

Description

Chen-Rust design effect for stratified, clustered, two-stage samples
Usage

deffCR(w, strvar=NULL, clvar=NULL, Wh=NULL, nest=FALSE, y)

Arguments

- **w**: vector of weights for a sample
- **strvar**: vector of stratum identifiers; equal in length to that of *w*.
- **clvar**: vector of cluster identifiers; equal in length to that of *w*.
- **Wh**: vector of the proportions of elements that are in each stratum; length is number of strata.
- **nest**: Are cluster IDs numbered within strata (TRUE or FALSE)? If TRUE, cluster IDs can be restarted within strata, e.g., 1,2,3,1,2,3,...
- **y**: vector of the sample values of an analysis variable

Details

The Chen-Rust *deff* accounts for stratification, clustering, and unequal weights, but does not account for the use of any auxiliary data in the estimator of a mean. The Chen-Rust *deff* returned here is appropriate for stratified, two-stage sampling.

Value

A list with components:

- **strata components**: Matrix with *deff*’s due to weighting, clustering, and stratification for each stratum
- **overall deff**: Design effect for full sample accounting for weighting, clustering, and stratification

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

- `deff`, `deffH`, `deffK`, `deffS`
**deffH**

Henry design effect for pps sampling and GREG estimation of totals

---

**Examples**

```r
require(sampling)
require(reshape)
data(MDarea.pop)
Ni <- table(MDarea.pop$TRACT)
m <- 20
probi <- m*Ni / sum(Ni)
  # select sample of clusters
set.seed(-780087528)
sam <- sampling::cluster(data=MDarea.pop, clustername="TRACT", size=m, method="systematic",
       pik=probi, description=TRUE)
  # extract data for the sample clusters
samclus <- getdata(MDarea.pop, sam)
samclus <- rename(samclus, c(Prob = "pi1"))
  # treat sample clusters as strata and select srswor from each
nbar <- 8
s <- sampling::strata(data = as.data.frame(samclus), stratanames = "TRACT",
       size = rep(nbar,m), method="srswor")
  # extracts the observed data
samdat <- getdata(samclus,s)
samdat <- rename(samdat, c(Prob = "pi2"))
  # add a fake stratum ID
H <- 2
nh <- m * nbar / H
stratum <- NULL
for (h in 1:H){
  stratum <- c(stratum, rep(h,nh))
}
wt <- 1/(samdat$pi1*samdat$pi2) * runif(m*nbar)
samdat <- cbind(subset(samdat, select = -c(Stratum)), stratum, wt)
deffCR(w = samdat$wt, strvar = samdat$stratum, clvar = samdat$TRACT, Wh=NULL, y=samdat$y2)
```

---

**Description**

Compute the Henry design effect for single-stage samples when a general regression estimator is used for a total.

**Usage**

```
deffH(w, y, x)
```

**Arguments**

- `w`  
  vector of inverses of selection probabilities for a sample
- `y`  
  vector of the sample values of an analysis variable
- `x`  
  matrix of covariates used to construct a GREG estimator of the total of `y`. This matrix does not include the intercept.
Details

The Henry design effect is the ratio of the variance of the general regression (GREG) estimator of a total of $y$ to the variance of the estimated total in srswr. Calculations for the Henry defH are done as if the sample is selected in a single-stage and with replacement. Varying selection probabilities can be used. The model for the GREG is assumed to be $y = \alpha + \beta x + \epsilon$, i.e., the model has an intercept.

Value

numeric design effect

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

deff, deffCR, deffK, deffS

Examples

```r
set.seed(-500398777)
# generate population using HMT function
pop.dat <- as.data.frame(HMT())
mos <- pop.dat$x
pop.dat$prbs.1d <- mos / sum(mos)
# select pps sample
require(sampling)
n <- 80
pk <- n * pop.dat$prbs.1d
sam <- UPrandomsystematic(pk)
sam <- sam==1
sam.dat <- pop.dat[sam, ]
dsgn.wts <- 1/pk[sam]
deffH(w=dsgn.wts, y=sam.dat$y, x=sam.dat$x)
```

**Description**

Compute the Kish design effect due to having unequal weights.

**Usage**

```r
defK(w)
```

**Arguments**

- `w` vector of inverses of selection probabilities for a sample

**Details**

The Kish design effect is the ratio of the variance of an estimated mean in stratified simple random sampling without replacement (`srswor`) to the variance of the estimated mean in `srswor`, assuming that all stratum unit variances are equal. In that case, proportional allocation with equal weighting is optimal. `deffK` equals `1 + relvar(w)` where `relvar` is relvariance of the vector of survey weights. This measure is not appropriate in samples where unequal weighting is more efficient than equal weighting.

**Value**

numeric design effect

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**


**See Also**

`deff`, `deffCR`, `deffH`, `deffS`

**Examples**

```r
data(nhis)
w <- nhis$svywt
defK(w)
```
The Spencer design effect is the ratio of the variance of the \textit{pwr}-estimator of the total of \(y\), assuming that a single-stage sample is selected with replacement, to the variance of the total estimated in \textit{srswr}. Varying selection probabilities can be used.

Value

numeric design effect

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

deff, deffCR, deffH, deffK
Examples

```r
set.seed(-500398777)  # generate population using HMT function
pop.dat <- as.data.frame(HMT())
mos <- pop.dat$x
pop.dat$prbs.1d <- mos / sum(mos)
# select pps sample
require(sampling)
n <- 80
pk <- n * pop.dat$prbs.1d
sam <- UPrandomsystematic(pk)
sam <- sam==1
sam.dat <- pop.dat[sam, ]
dsgn.wts <- 1/pk[sam]
deffS(p=sam.dat$prbs.1d, w=dsgn.wts, y=sam.dat$y)
```

---

**Domainy1y2**  
*Domain data*

**Description**

*Domainy1y2* is a small data file used for an exercise in sample size calculations.

**Usage**

```r
data(Domainy1y2)
```

**Format**

A data frame with 30 observations on 2 variables.

- **y1** an artificial variable
- **y2** an artificial variable

**References**


**Examples**

```r
data(Domainy1y2)
str(Domainy1y2)
summary(Domainy1y2)
```
Sample sizes for a double sampling design

Description

Compute sample sizes at each phase of a two-phase design where strata are created using the first phase.

Usage

dub(c1, c2, Ctot, Nh, Sh, Yh.bar)

Arguments

c1  cost per unit in phase-1

c2  cost per unit in phase-2

Ctot  Total variable cost

Nh  Vector of stratum population counts or proportions

Sh  Vector of stratum population standard deviations

Yh.bar  Vector of stratum population means

Details

Compute the first and second phase sample sizes for a double sampling design. A first phase sample is selected by simple random sampling (srs). Strata are formed based on information collected in the first phase. The Neyman allocation to strata of the second phase sample is computed ignoring costs. Optimal total sample sizes are computed for the first and second phases, given per-unit costs for the first and second phases and a fixed total budget for both phases combined.

Value

A list object with elements:

V1  Variance component associated with phase-1

V2  Variance component associated with phase-2

n1  Phase-1 sample size

n2  Total phase-2 sample across all strata

"n2/n1"  Fraction that phase-2 is of phase-1

ney.alloc  Vector of stratum sample sizes for phase-2 sample

Vopt  Variance of mean with the calculated phase-1 and phase-2 sample sizes

nsrs  Size of an srs that has cost Ctot, assuming each unit costs c2

Vsrs  Variance of mean in an srs of cost Ctot, assuming each unit costs c2

Vratio  Ratio of Vopt to Vsrs

Ctot  Input value of total cost

cost.chk  Computed value of phase-1 plus phase-2 sample with optimal sample sizes; should agree with Ctot
gamEst

Author(s)
Richard Valliant, Jill A. Dever, Frauke Kreuter

References

Examples
\begin{verbatim}
Wh <- rep(0.25,4)
Ph <- c(0.02,0.12,0.37,0.54)
Sh <- sqrt(Ph*(1-Ph))
c1 <- 10
c2 <- 50
Ctot <- 20000
dub(c1, c2, Ctot, Nh=Wh, Sh, Yh.bar=Ph)
\end{verbatim}

\begin{verbatim}
  gamEst
  Estimate variance model parameter $\gamma$
\end{verbatim}

Description
Regresses a $y$ on a set of covariates $X$ where $\text{Var}_M(y) = \sigma^2 x^\gamma$ and then regresses the squared residuals on $\log(x)$ to estimate $\gamma$.

Usage
gamEst(X1, x1, y1, v1)

Arguments
X1 matrix of predictors in the linear model for $y1$
x1 vector of $x$'s for individual units in the assumed specification of $\text{Var}_M(y)$
y1 vector of dependent variables for individual units
v1 vector proportional to $\text{Var}_M(y)$

Details
The function gamEst estimates the power $\gamma$ in a model where the variance of the errors is proportional to $x^\gamma$ for some covariate $x$. Values of $\gamma$ are typically in [0,2]. The function is iteratively called by gammaFit, which is normally the function that an analyst should use.
gammaFit

Value
The estimate of $\gamma$.

Author(s)
Richard Valliant, Jill A. Dever, Frauke Kreuter

References

See Also
gammaFit

Examples
```r
data(hospital)
x <- hospital$x
y <- hospital$y

X <- cbind(sqrt(x), x)
gamEst(X1 = X, x1 = x, y1 = y, v1 = x)
```

__gammaFit__

Iteratively estimate variance model parameter $\gamma$

Description
Iteratively computes estimate of $\gamma$ in a model with $E_M(y) = x^T \beta$ and $Var_M(y) = \sigma^2 x^\gamma$.

Usage
gammaFit(X, x, y, maxiter = 100, show.iter = FALSE, tol = 0.001)

Arguments
- **X**: matrix of predictors in the linear model for $y$
- **x**: vector of $x$'s for individual units in the assumed specification of $Var_M(y)$
- **y**: vector of dependent variables for individual units
- **maxiter**: maximum number of iterations allowed
- **show.iter**: should values of $\gamma$ be printed of each iteration? TRUE or FALSE
- **tol**: size of relative difference in $\hat{\gamma}$'s between consecutive iterations used to determine convergence. Algorithm terminates when relative difference is less than tol.
Details

The function `gammaFit` estimates the power $\gamma$ in a model where the variance of the errors is proportional to $x^\gamma$ for some covariate $x$. Values of $\gamma$ are typically in [0,2]. The function calls `gamEst`.

Value

A list with the components:

- `g.hat`: estimate of $\gamma$ when iterative procedure stopped
- `converged`: TRUE or FALSE depending on whether convergence was obtained
- `steps`: number of steps used by the algorithm

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

`gamEst`

Examples

```r
data(hospital)
x <- hospital$x
y <- hospital$y
X <- cbind(sqrt(x), x)
gammaFit(X = X, x = x, y = y, maxiter=100, tol=0.001)
```

---

**HMT**

*Generate an HMT population*

Description

Generate a population that follows the model in Hansen, Madow, and Tepping (1983)

Usage

```r
HMT(N=5000, H=10)
```

Arguments

- `N`: population size
- `H`: number of strata
Details

HMT generates a population based on the model: $E(y) = \alpha + \beta x$, $Var(y) = \sigma^2 x^{3/2}$. Both $x$ and $y$ have gamma distributions. Strata are formed to have approximately the same total of $x$.

Value

N x 3 matrix with columns:

<table>
<thead>
<tr>
<th>stratum ID</th>
<th>auxiliary variable $x$</th>
<th>analysis variable $y$</th>
</tr>
</thead>
</table>

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


Examples

```r
# generate HMT population with 1000 units and 5 strata and plot results
pop <- HMT(N=1000, H=5)
plot(pop[, "x"], pop[, "y"])
```

hospital

<table>
<thead>
<tr>
<th>hospital</th>
<th>Hospital Data</th>
</tr>
</thead>
</table>

Description

The hospital data file is a national sample of short-stay hospitals with fewer than 1000 beds.

Usage

data(hospital)

Format

A data frame with 393 observations on the following 2 variables.

$y$ Number of patients discharged by the hospital in January 1968

$x$ Number of inpatient beds in the hospital
Details

The hospital data are from the National Hospital Discharge Survey conducted by the U.S. National Center for Health Statistics. The survey collects characteristics of inpatients discharged from non-Federal short-stay hospitals in the United States. This population is from the January 1968 survey and contains observations on 393 hospitals.

Source

National Center for Health Statistics Hospital Discharge Survey of 1968.

References


Examples

data(hospital)
str(hospital)

labor

*Labor force population*

Description

A clustered population of persons extracted from the September 1976 Current Population Survey (CPS)

Usage

data(labor)

Format

A data frame with 478 persons on the following variables:

- h stratum
- cluster cluster (or segment) number
- person person number
- age age of person
- agecat age category (1 = 19 years and under; 2 = 20-24; 3 = 25-34; 4 = 35-64; 5 = 65 years and over)
- race race (1 = non-Black; 2 = Black)
- sex Gender (1=Male; 2=Female)
- HoursPerWk Usual number of hours worked per week
- WklyWage Usual amount of weekly wages (in 1976 U.S. dollars)
- y An artificial variable generated to follow a model with a common mean. Persons in the same cluster are correlated. Persons in different clusters are uncorrelated under the model.
Details
This population is a clustered population of 478 persons extracted from the September 1976 Current Population Survey (CPS) in the United States. The clusters are compact geographic areas used as one of the stages of sampling in the CPS and are typically composed of about 4 nearby households. The elements within clusters for this illustrative population are individual persons.

Source

Examples
```r
data(labor)
str(labor)
table(labor$h)
hist(labor$WklyWage)
```

Description
An artificial population of census tracts, block groups, and persons

Usage
data(MDarea.pop)

Format
A data frame with 403,997 persons on the following variables:

- PSU: A grouping of block groups (BLKGROUP) which has about 5050 persons
- SSU: A grouping of block groups which has about 1010 persons
- TRACT: A geographic area defined by the Census Bureau. Tracts generally have between 1,500 and 8,000 people but have a much wider range in Anne Arundel county.
- BLKGROUP: Block group. A geographic area defined by the Census Bureau. Block groups generally have between 600 and 3,000 people.
- Hispanic: Hispanic ethnicity (1=Hispanic; 2=Non-Hispanic)
- Gender: Gender (1=Male; 2=Female)
- Age: 23 level age category (1 = Under 5 years; 2 = 5 to 9 years; 3 = 10 to 14 years; 4 = 15 to 17 years; 5 = 18 and 19 years; 6 = 20 years; 7 = 21 years; 8 = 22 to 24 years; 9 = 25 to 29 years; 10 = 30 to 34 years; 11 = 35 to 39 years; 12 = 40 to 44 years; 13 = 45 to 49 years; 14 = 50 to 54 years; 15 = 55 to 59 years; 16 = 60 and 61 years; 17 = 62 to 64 years; 18 = 65 and 66 years; 19 = 67 to 69 years; 20 = 70 to 74 years; 21 = 75 to 79 years; 22 = 80 to 84 years; 23 = 85 years and over)
person Counter for person within tract/block group/Hispanic/Gender/Age combination
y1 Artificial continuous variable
y2 Artificial continuous variable
y3 Artificial continuous variable
ins.cov Medical coverage (0 = person does not have medical insurance coverage; 1 = person has medical insurance coverage)
hosp.stay Overnight hospital stay (0 = person did not have an overnight hospital stay in last 12 months; 1 = person did have an overnight hospital stay in last 12 months)

Details
A dataset of 403,997 persons based on the 2000 decennial U.S. Census for Anne Arundel County in the state of Maryland. Person records were generated based on counts from the 2000 census. Individual values for each person were generated using models. Groupings to form the variables PSU and SSU were done after sorting the census file by tract and block group within tract.

Source

Examples
data(MDarea.pop)
str(MDarea.pop)
table(MDarea.pop$TRACT)
table(MDarea.pop$TRACT, MDarea.pop$Hispanic)

mibrfss

Description
Demographic and health related variables from a U.S. household survey in the state of Michigan

Usage
data(mibrfss)

Format
A data frame with 2485 observations on persons for the following 21 variables.

SMOKE100 Smoked 100 or more cigarettes in lifetime (1 = Yes; 2 = No)
BMICAT3 Body mass index category (1 = Neither overweight nor obese (BMI < 25); 2 = Overweight (25 <= BMI <= 30); 3 = Obese (BMI > 30))
AGECAT Age group (1 = 18-24 years; 2 = 25-34 years; 3 = 35-44 years; 4 = 45-54 years; 5 = 55-64 years; 6 = 65+)
GENHLTH General health (self-reported) (1 = Excellent; 2 = Very good; 3 = Good; 4 = Fair; 5 = Poor)

PHYSACT Physical activity: In last month participated in activities such as running, calisthenics, golf, gardening, or walking for exercise (1 = Yes; 2 = No)

HIGHBP High blood pressure: Have you ever been told by a doctor, nurse, or other health professional that you have high blood pressure? (1 = Yes; 2 = No)

ASTHMA Asthma: Have you ever been told by a doctor, nurse, or other health professional that you have asthma? (1 = Yes; 2 = No)

HISPANIC Hispanic ethnicity (1 = Yes; 2 = No; 7 = Missing)

WEIGHT Body weight in pounds

GENDER Gender (1 = Male; 2 = Female)

CELLPHON Has a wireless phone (1 = Yes; 2 = No)

INETHOME Has access to the Internet at home (1 = Yes; 2 = No)

WEBUSE How often do you use the Internet at home? Would you say, at least once a day, five to six times a week, two to four times a week, about once a week, less than once a week, or have you not used the Internet in the last month? (1 = At least once a day; 2 = 5-6 times a week; 3 = 2-4 times a week; 4 = About once a week; 5 = Less than once a week; 6 = Not in the last month)

RACECAT Race (1 = White; 2 = African American; 3 = Other)

EDCAT Education level (1 = Did not graduate high school; 2 = Graduated high school; 3 = Attended college or technical school; 4 = Graduated from college or technical school)

INCOMC3 Income category (1 = Less than $15000; 2 = $15000 to less than $25000; 3 = $25000 to less than $35000; 4 = $35000 to less than $50000; 5 = $50000 or more)

DIABETE2 Diabetes: Have you ever been told by a doctor, nurse, or other health professional that you have diabetes? (1 = Yes; 2 = No)

CHOLCHK Cholesterol check: Blood cholesterol is a fatty substance found in the blood. Have you ever had your blood cholesterol checked? (1 = Yes; 2 = No)

BMI Body mass index (continuous)

BINGE2 Binge drinking: At risk for binge drinking based on alcohol consumption responses (1 = Yes; 2 = No)

ARTHRIT Arthritis: Have you ever been told by a doctor, nurse, or other health professional that you have some form of arthritis, rheumatoid arthritis, gout, lupus, or fibromyalgia, or have joint symptoms of arthritis? (1 = Yes; 2 = No; 3 = Don’t know, not sure, or refused)

Details

The Michigan Behavioral Risk Factor Surveillance Survey (MIBRFSS) is part of a national state-by-state system of surveys used to monitor health conditions in the U.S. Data are collected through telephone household interviews. Demographic variables and a few health related variables are included in this subset. The mibrfss data set contains observations on 2845 persons and is extracted from the 2003 U.S. survey. The file contains only persons 18 years and older.
Source

See Also
nhis, nhis.large

Examples

data(mibrfss)
str(mibrfss)
table(mibrfss$SMOKE100, useNA = "always")
table(mibrfss$BMICAT3, useNA="always")

nAuditAttr Sample sizes for an attribute sample in an audit

Description
Compute a sample size for an audit where the goal is to control the probability of observing only a small number of errors given an underlying error rate in the population. Auditors refer to this as an attribute sample.

Usage
nAuditAttr(TolRate=0.05, AccDev, CL, N=5000)

Arguments

TolRate Proportion of units in the population with an attribute, e.g., errors in an audit. Auditors term this the 'tolerable rate of deviation' in the population to be tested.

AccDev Acceptable deviation, which is the number of units with the attribute (i.e., the number of errors) that would be acceptable in the sample. The largest proportion of errors that would be deemed to be acceptable in an audit would be AccDev/N.

CL Probability that the sample will contain an acceptable number of errors. Auditors refer to this as 'confidence level'. The probability that the sample will contain AccDev errors or fewer is 1-CL.

N Size of the population of records to be audited.
Details

`nAuditAttr` computes the minimum sample size required so that the probability, $1-\text{CL}$ of detecting less than or equal to a specified number of errors in the sample, is controlled. The sample is assumed to be selected with equal probabilities. `AccDev` is the largest number of errors in the sample that will be considered as meeting the audit standards. `TolRate` is the underlying population error rate, which is typically set to be larger than \( \frac{\text{AccDev}}{N} \). The sample size is computed in two ways: (1) using the hypergeometric distribution, which accounts for the size of the population and (2) with the binomial distribution, which will be appropriate if the population is very large. When \( N \) is large and the sampling fraction is small, both sample sizes will be approximately the same.

Value

List object with values:

- **Pop.Size**: population size
- **Tol.Dev.Rate**: proportion of records with errors in population
- **Acceptable.Errors**: largest number of errors, found in the sample, that will meet audit standards
- **Sample.Size.Hypergeometric**: minimum sample size needed to detect `AccDev` errors in the sample computed via the hypergeometric distribution
- **Sample.Size.Binomial**: minimum sample size needed to detect `AccDev` errors in the sample computed via the binomial distribution

Author(s)

George Zipf, Richard Valliant

References


Examples

# Examples from the US GAO Financial Audit Manual (2020), Figure 450.1, Table 1
```
nAuditAttr(AccDev = 0, CL = .90)
nAuditAttr(AccDev = 1, CL = .90)
nAuditAttr(AccDev = 2, CL = .90)
nAuditAttr(AccDev = 3, CL = .90)
nAuditAttr(AccDev = 4, CL = .90)
```
Sample sizes for a Monetary Unit Sample in an audit

**Description**

Compute a sample size for an audit where the goal is to control the probability of observing only a small number of errors given an underlying error rate in the population. The sample will be selected with probabilities proportional to a measure of size (MOS). When the MOS of each record is a monetary unit, auditors refer to this as an monetary unit sampling or dollar unit sampling.

**Usage**

\[
nAuditMUS(MUSVar, Value.sw, CL = 0.90, Error.sw, Tol.Error, Exp.Error = 0)
\]

**Arguments**

- **MUSVar**
  - The measure of size for monetary unit sampling (MUS)

- **Value.sw**
  - Determines whether the monetary unit sample is based on positive values, negative values, or absolute values. If `Value.sw = "Positive"` or "Pos", only positive values of `MUSVar` are used; if "Negative" or "Neg" only negative values are used; if "Absolute" or "Abs", all values of `MUSVar` are used after taking their absolute values.

- **CL**
  - Probability that the sample will contain an acceptable number of errors. Auditors refer to this as 'confidence level'. The probability that the sample will contain the tolerable number of errors or fewer is 1-CL. The range of CL is 0 to 1.

- **Error.sw**
  - Determines whether errors are based on monetary amounts or percentages, i.e., whether `Tol.Error` is interpreted as a dollar amount (`Error.sw = "Absolute"` or "Amt") or as a percent (`Error.sw = "Percent"` or "Pct").

- **Tol.Error**
  - The amount of error expressed as a value or a percentage that the auditor considers tolerable. If `Error.sw` is "Percent" or "Pct", `Tol.Error` is a percent between 0 and 100. If `Error.sw = "Absolute"` or "Amt", `Tol.Error` is interpreted as a dollar amount.

- **Exp.Error**
  - The amount of error, expressed as a value or a percentage, that the auditor expects in the population. If `Error.sw` is "Percent" or "Pct", `Exp.Error` is a percent between 0 and 100. If `Error.sw = "Absolute"` or "Amt", `Exp.Error` is interpreted as a dollar amount.

**Details**

`nAuditMUS` computes the minimum sample size required for a given population, tolerable error rate or misstatement, and desired confidence level. If the expected error or misstatement is 0, (i.e. `Exp.Error = 0`), then the sample size is computed using the hypergeometric distribution where the acceptable number of deviations is 0. If the expected error is greater than 0, then sample size is computed by first calculating the maximum sample size where the number of deviations divided by the sample size is less than the expected error, then calculating the minimum sample size where
the number of deviations divided by the sample size is greater than the expected error, and finally performing a straight line interpolation between these two values where the interpolated value is the specified expected error. The returned sample size calculation is the ceiling of that interpolated sample size.

**Value**

List object with values:

- **Value.Range** Whether the MUS variable is for positive, negative, or absolute values as defined by `Value.sw`
- **Error.Type** Amount or Percent as defined by `Error.sw`
- **Tol.Error.Rate** The tolerable error expressed as a percentage of items if `Error.sw = "Percent"` or "Pct" or as a proportion of total monetary units otherwise
- **Exp.Error.Rate** The expected error expressed as a percentage of items if `Error.sw = "Percent"` or "Pct" or as a proportion of total monetary value otherwise
- **Number.Records** The population count of records in the value range based on selecting ones with positive, negative or absolute value of `MUSVar`
- **Sample.Size** Minimum sample size needed to meet tolerable and expected error rate requirements
- **Number.HighVal** Number of records that are high value (exceed the interval used for systematic sampling) and will be certainties in the sample
- **Positive.Pop.Dollars** The absolute value of the total dollar (or other monetary unit) amount in the population in the value range
- **Conf.level** Probability that the sample will meet MUS requirements
- **Sampling.Interval** Spacing or skip interval that would be used in a systematic probability proportional to monetary unit sampling

**Author(s)**

George Zipf, Richard Valliant

**References**


**See Also**

-nAuditAttr
Examples

```r
# generate an artificial population with some negative monetary amounts
EX <- 1000
relvar <- 2
alpha <- 1/relvar
sigma <- EX * relvar
lowval <- 100  # minimum positive X's allowed
prop.neg <- 0.05  # proportion of pop with negative values
N.neg <- floor(1000 * prop.neg)  # number of negative X's allowed
X <- rgamma(n=1000, shape=alpha, scale=sigma)
Xlow <- sort(X)[1:N.neg]
xneg <- -Xlow - lowval
xpos <- X[N.neg:length(X)]
X <- c(xneg, xpos)

```

---

**nCont**

*Compute a simple random sample size for an estimated mean*

**Description**

Compute a simple random sample size using either a target coefficient of variation, \( CV_0 \), or target variance, \( V_0 \), for an estimated mean.

**Usage**

```r
nCont(CV0=NULL, V0=NULL, S2=NULL, ybarU=NULL, N=Inf, CVpop=NULL)
```

**Arguments**

- `CV0`: target value of coefficient of variation of \( \bar{y}_s \)
- `V0`: target value of variance of \( \bar{y}_s \)
- `S2`: unit (population) variance
- `ybarU`: population mean of target variable
- `N`: number of units in finite population
- `CVpop`: unit (population) coefficient of variation

**Details**

If \( CV_0 \) is the desired target, then the unit CV, CVpop, or the population mean and variance, \( ybarU \) and \( S2 \), must also be provided. If \( V_0 \) is the constrained value, then \( S2 \) must be also be included in the function call.
**Value**
numeric sample size

**Author(s)**
Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**

**See Also**
nLogOdds, nProp, nPropMoe, nWilson

**Examples**

```r
nCont(CV0=0.05, CVpop=2)
nCont(CV0=0.05, CVpop=2, N=500)
nCont(CV0=0.10/1.645, CVpop=1)

# Compute sample size for a ratio estimator in smho98 population
# Identify large units to select with certainty first
data(smho98)
cert <- smho98[, "BEDS"] > 2000
tmp <- smho98[, !cert, ]
tmp <- tmp[tmp[, "BEDS"] > 0, ]
x <- tmp[, "BEDS"]
y <- tmp[, "EXPTOTAL"]
m <- lm(y ~ 0 + x, weights = 1/x)
ybarU <- mean(y)
S2R <- sum(m$residuals^2/(length(x)-1))
nCont(CV0=0.15, S2=S2R, ybarU=ybarU)
```

---

**nContMoe**

*Compute a simple random sample size for an estimated mean of a continuous variable based on margin of error*

**Description**
Compute a simple random sample size using a margin of error specified as the half-width of a normal approximation confidence interval or the half-width relative to the population mean.

**Usage**

```r
nContMoe(moe.sw, e, alpha=0.05, CVpop=NULL, S2=NULL, ybarU=NULL, N=Inf)
```
Arguments

moe.sw switch for setting desired margin of error (1 = CI half-width on the mean; 2 = CI half-width on the mean divided by \( \bar{y_U} \))
e desired margin of error; either \( e = z_{1-\alpha/2}/\sqrt{V(\bar{y}_s)} \) or \( e = z_{1-\alpha/2}CV(\bar{y}_s) \)
alpha 1 - (confidence level)
CVpop unit (population) coefficient of variation
S2 population variance of the target variable
ybarU population mean of target variable
N number of units in finite population

Details

If moe.sw=1, then S2 must be provided. If moe.sw=2, then either (i) CVpop or (ii) S2 and ybarU must be provided.

Value

numeric sample size

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

nCont, nLogOdds, nProp, nPropMoe, nWilson

Examples

nContMoe(moe.sw=1, e=0.05, alpha=0.05, S2=2)
nContMoe(moe.sw=1, e=0.05, alpha=0.05, S2=2, N=200)
nContMoe(moe.sw=2, e=0.05, alpha=0.05, CVpop=2)
nContMoe(moe.sw=2, e=0.05, alpha=0.05, CVpop=2, N=200)
nContMoe(moe.sw=2, e=0.05, alpha=0.05, S2=4, ybarU=2)
nDep2sam

Simple random sample size for difference in means

Description

Compute a simple random sample size for estimating the difference in means when samples overlap

Usage

nDep2sam(S2x, S2y, g, r, rho, alt, del, sig.level=0.05, pow=0.80)

Arguments

S2x  unit variance of analysis variable x in sample 1
S2y  unit variance of analysis variable y in sample 2
g   proportion of sample 1 that is in the overlap with sample 2
r   ratio of the size of sample 1 to that of sample 2
rho unit-level correlation between x and y
alt should the test be 1-sided or 2-sided; allowable values are alt="one.sided" or alt="two.sided".
del size of the difference between the means to be detected
sig.level significance level of the hypothesis test
pow desired power of the test

Details

nDep2sam computes sample sizes in two groups that are required for testing whether the difference in group means is significant. The power of the test is one of the input parameters. The samples have a specified proportion of units in common. Both samples are assumed to be selected via simple random sampling.

Value

List with values:

n1 sample size in group 1
n2 sample size in group 2
S2x.S2y unit variances in groups 1 and 2
delta difference in group means to be detected
gamma proportion of sample 1 that is in the overlap with sample 2
r ratio of the size of sample 1 to that of sample 2
rho unit-level correlation between analysis variables in groups 1 and 2
alt type of test: one-sided or two-sided
sig.level significance level of test
power power of the test
nDomain

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter

**References**


**See Also**

nProp2sam

**Examples**

```r
nDep2sam(S2x=200, S2y=200,
          g=0.75, r=1, rho=0.9,
          alt="one.sided", del=5,
          sig.level=0.05, pow=0.80)
```

---

**nDomain**

Compute a simple random sample size for an estimated mean or total for a domain

**Description**

Compute a simple random sample size using either a target coefficient of variation, $CV_0(d)$, or target variance, $V_0(d)$, for an estimated mean or total for a domain.

**Usage**

```r
nDomain(CV0d=NULL, V0d=NULL, S2d=NULL, ybarUd=NULL, N=Inf, CVpopd=NULL, Pd, est.type)
```

**Arguments**

- `CV0d`: target value of coefficient of variation of estimated domain mean or total
- `V0d`: target value of variance of estimated domain mean or total
- `S2d`: unit (population) variance for domain units
- `ybarUd`: population mean of target variable for domain units
- `N`: number of units in full finite population (not just the domain population)
- `CVpopd`: unit (population) coefficient of variation for domain units
- `Pd`: proportion of units in the population that are in the domain
- `est.type`: type of estimate; allowable values are "mean" or "total"
Details

If CV0d is the desired target, then the unit CV, CVpopd, or the domain population mean and variance, ybarUd and S2d, must also be provided. If V0d is the constrained value, then ybarUd must be also be included in the function call. CV0d will then be computed as \( \sqrt{V0d}/ybarUd \).

Value

umeric sample size

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

\nCont, nLogOdds, nProp, nPropMoe, nWilson

Examples

```r
nDomain(CV0d=0.05, N=Inf, CVpopd=1, Pd=0.5, est.type="total")
nDomain(CV0d=0.05, N=Inf, CVpopd=1, Pd=0.5, est.type="mean")
nDomain(V0d=50, ybarUd=50, S2d=100, N=Inf, Pd=0.5, est.type="total")
nDomain(CV0d=0.05, ybarUd=50, S2d=100, N=Inf, Pd=0.5, est.type="total")
nDomain(CV0d=0.05, ybarUd=50, S2d=100, N=Inf, Pd=0.5, est.type="mean")
```

---

*nhis*  
*National Health Interview Survey: Demographic variables*

Description

Demographic variables from a U.S. national household survey

Usage

data(nhis)
Format

A data frame with 3,911 observations on the following 16 variables.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Identification variable</td>
</tr>
<tr>
<td>stratum</td>
<td>Sample design stratum</td>
</tr>
<tr>
<td>psu</td>
<td>Primary sampling unit, numbered within each stratum (1,2)</td>
</tr>
<tr>
<td>svywt</td>
<td>Survey weight</td>
</tr>
<tr>
<td>sex</td>
<td>Gender (1 = male; 2 = female)</td>
</tr>
<tr>
<td>age</td>
<td>Age, continuous</td>
</tr>
<tr>
<td>age_r</td>
<td>Recoded age (3 = 18-24 years; 4 = 25-44 years; 5 = 45-64 years; 6 = 65-69 years; 7 = 70-74 years; 8 = 75 years and older)</td>
</tr>
<tr>
<td>hisp</td>
<td>Hispanic ethnicity (1 = Hispanic; 2 = Non-Hispanic)</td>
</tr>
<tr>
<td>marital</td>
<td>Marital status (1 = Separated; 2 = Divorced; 3 = Married; 4 = Single/never married; 5 = Widowed; 9 = Unknown marital status)</td>
</tr>
<tr>
<td>parents</td>
<td>Parent(s) of sample person present in the family (1 = Mother, no father; 2 = Father, no mother; 3 = Mother and father; 4 = Neither mother nor father)</td>
</tr>
<tr>
<td>parents_r</td>
<td>Parent(s) of sample person present in the family recode (1 = Yes; 2 = No)</td>
</tr>
<tr>
<td>educ</td>
<td>Education (1 = 8th grade or less; 2 = 9-12th grade, no high school diploma; 3 = High school graduate; 4 = General education development (GED) degree recipient; 5 = Some college, no degree; 6 = Associate’s degree, technical or vocational; 7 = Associate’s degree, academic program; 8 = Bachelor’s degree (BA, BS, AB, BBA); 9 = Master’s, professional, or doctoral degree)</td>
</tr>
<tr>
<td>educ_r</td>
<td>Education recode (1 = High school, general education development degree (GED), or less; 2 = Some college; 3 = Bachelor’s or associate’s degree; 4 = Master’s degree &amp; higher)</td>
</tr>
<tr>
<td>race</td>
<td>Race (1 = White; 2 = Black; 3 = Other)</td>
</tr>
<tr>
<td>resp</td>
<td>Respondent (0 = nonrespondent; 1 = respondent)</td>
</tr>
</tbody>
</table>

Details

The National Health Interview Survey (NHIS) is used to monitor health conditions in the U.S. Data are collected through personal household interviews. Only demographic variables are included in this subset which was collected in 2003. The nhis data set contains observations for 3,911 persons. The file contains only persons 18 years and older.

Source

National Health Interview Survey of 2003 conducted by the U.S. National Center for Health Statistics. [https://www.cdc.gov/nchs/nhis.htm](https://www.cdc.gov/nchs/nhis.htm)

See Also

nhis.large
Examples

```r
data(nhis)
str(nhis)
table(nhis$sex, nhis$age_r)
```

---

**nhis.large**

*National Health Interview Survey: Demographic and health variables*

---

**Description**

Demographic and health related variables from a U.S. national household survey

**Usage**

```r
data(nhis.large)
```

**Format**

A data frame with 21,588 observations on the following 18 variables.

- **ID**: Identification variable
- **stratum**: Sample design stratum
- **psu**: Primary sampling unit, numbered within each stratum (1,2)
- **svywt**: Survey weight
- **sex**: Gender (1 = male; 2 = female)
- **age.grp**: Age group (1 = < 18 years; 2 = 18-24 years; 3 = 25-44 years; 4 = 45-64 years; 5 = 65+)
- **hisp**: Hispanic ethnicity (1 = Hispanic; 2 = Non-Hispanic White; 3 = Non-Hispanic Black; 4 = Non-Hispanic All other race groups)
- **parents**: Parents present in the household (1 = mother, father, or both present; 2 = neither present)
- **educ**: Highest level of education attained (1 = High school graduate, graduate equivalence degree, or less; 2 = Some college; 3 = Bachelor’s or associate’s degree; 4 = Master’s degree or higher; NA = missing)
- **race**: Race (1 = White; 2 = Black; 3 = All other race groups)
- **inc.grp**: Family income group (1 = < $20K; 2 = $20000-$24999; 3 = $25000-$34999; 4 = $35000-$44999; 5 = $45000-$54999; 6 = $55000-$64999; 7 = $65000-$74999; 8 = $75K+; NA = missing)
- **delay.med**: Delayed medical care in last 12 months because of cost (1 = Yes; 2 = No; NA = missing)
- **hosp.stay**: Had an overnight hospital stay in last 12 months (1 = Yes; 2 = No; NA = missing)
- **doc.visit**: During 2 WEEKS before interview, did person see a doctor or other health care professional at a doctor’s office, a clinic, an emergency room, or some other place? (excluding overnight hospital stay)? (1 = Yes; 2 = No)
- **medicaid**: Covered by medicaid, a governmental subsidy program for the poor (1 = Yes; 2 = No; NA = missing)
notcov  Not covered by any type of health insurance (1 = Yes; 2 = No; NA = missing)

doing.lw  What was person doing last week? (1 = Working for pay at a job or business; 2 = With a job or business but not at work; 3 = Looking for work; 4 = Working, but not for pay, at a job or business; 5 = Not working and not looking for work; NA = missing)

limited  Is the person limited in any way in any activities because of physical, mental or emotional problems? (1 = Limited in some way; 2 = Not limited in any way; NA = missing)

Details

The National Health Interview Survey (NHIS) is used to monitor health conditions in the U.S. Data are collected through personal household interviews. Demographic variables and a few health related variables are included in this subset. The nhis.large data set contains observations on 21,588 persons extracted from the 2003 U.S. NHIS survey. The file contains only persons 18 years and older.

Source


See Also

   nhis

Examples

   data(nhis.large)
   str(nhis.large)
   table(nhis.large$stratum, nhis.large$psu)
   table(nhis.large$delay.med, useNA="always")

Description

Socioeconomic variables from a U.S. national household survey

Usage

   data(nhispart)
Format

A data frame with 3,924 observations on the following variables.

- **HHX**: Household identification variable
- **PX**: Person identifier within household
- **STRATUM**: Sample design stratum
- **PSU**: Primary sampling unit, numbered within each stratum (1,2)
- **WTFA**: Survey weight
- **SEX**: Gender (1 = male; 2 = female)
- **AGE_P**: Age of persons; values are 18-85 (85 includes age 85 and older)
- **R_AGE1**: Age group (3 = 18-24 years; 4 = 25-44 years; 5 = 45-64 years; 6 = 65-69 years; 7 = 70-74 years; 8 = 75 years and over)
- **ORIGIN_I**: Hispanic ethnicity (1 = Hispanic; 2 = Non-Hispanic)
- **RACERPI2**: Race grouped (1 = White only; 2 = Black/African American only; 3 = American Indian or Alaska native (AIAN) only; 4 = Asian only; 5 = Race group not releasable; 6 = Multiple race)
- **MRACRPI2**: Race detailed (1 = White; 2 = Black/African American; 3 = Indian (American), Alaska Native; 9 = Asian Indian; 10 = Chinese; 11 = Filipino; 15 = Other Asian; 16 = Primary race not releasable; 17 = Multiple race, no primary race selected)
- **RACRECI2**: White/Black (1 = White; 2 Black; 3 All other race groups)
- **R_MARITL**: Marital status (1 = Married - spouse in household; 2 = Married - spouse not in household; 3 = Married - unknown whether spouse in household; 4 = Widowed; 5 = Divorced; 6 = Separated; 7 = Never married; 8 = Living with partner; 9 = Unknown marital status)
- **CDCMSTAT**: CDC marital status (1 = Mother, no father; 2 = Father, no mother; 3 = Mother and father; 4 = Neither mother nor father)
- **INCGRP**: Total combined family income group (1 = 0-$4999; 2 = $5000-$9999; 3 = $10000-$14999; 4 = $15000-$19999; 5 = $20000-$24999; 6 = $25000-$34999; 7 = $35000-$44999; 8 = $45000-$54999; 9 = $55000-$64999; 10 = $65000-$74999; 11 = $75000 and over; 12 = $20000 or more (no detail); 13 = Less than $20000 (no detail); 97 = Refused; 98 = Not ascertained; 99 = Don’t know)
- **PARENTS**: Parent(s) present in the family (1 = Mother, no father; 2 = Father, no mother; 3 = Mother and father; 4 = Neither mother nor father)
- **EDUC_R1**: Highest level of education attained (1 = Less than high school graduate; 3 = High school graduate or general education development degree (GED); 5 = Some college, no degree; 6 = AA degree, technical or vocational or AA degree, academic program or Bachelor’s degree (BA, BS, AB, BBA); 9 = Master’s, professional, or doctoral degree)
- **RAT_CAT**: Ratio of family income to poverty level (1 = Under 0.50; 2 = 0.50 to 0.74; 3 = 0.75 to 0.99; 4 = 1.00 to 1.24; 5 = 1.25 to 1.49; 6 = 1.50 to 1.74; 7 = 1.75 to 1.99; 8 = 2.00 to 2.49; 9 = 2.50 to 2.99; 10 = 3.00 to 3.49; 11 = 3.50 to 3.99; 12 = 4.00 to 4.49; 13 = 4.50 to 4.99; 14 = 5.00 and over; 99 = Unknown)
Details

The National Health Interview Survey (NHIS) is used to monitor health conditions in the U.S. Data are collected through personal household interviews. Socioeconomic variables are included in this subset along with household and person codes. The nhispart data set contains observations on 3,924 persons extracted from the 2003 U.S. survey. The file contains only persons 18 years and older.

Source


Examples

data(nhispart)
str(nhispart)
table(nhispart$STRATUM, nhispart$PSU)
table(nhispart$RACERPI2, nhispart$RACRECI2, useNA="always")

nLogOdds

Calculate simple random sample size for estimating a proportion

Description

Calculate the simple random sample size for estimating a proportion using the log-odds transformation.

Usage

nLogOdds(moe.sw, e, alpha=0.05, pU, N=Inf)

Arguments

moe.sw switch for setting desired margin of error (1 = CI half-width on the proportion; 2 = CI half-width on a proportion divided by \(p_U\))
e desired margin of error
alpha 1 - (confidence level)
pU population proportion
N number of units in finite population

Details

The function accepts five parameters, which are the same ones as accepted by nPropMoe. The desired margin of error can be specified as the CI half-width on the proportion (moe.sw=1) or as the CI half-width as a proportion of the population value \(p_U\) (moe.sw=2).
Value

numeric sample size

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

nProp, nPropMoe, nWilson, nCont

Examples

nLogOdds(moe.sw=1, e=0.05, alpha=0.05, pU=0.2, N=Inf)
nLogOdds(moe.sw=2, e=0.05, alpha=0.05, pU=0.2, N=Inf)

---

**nProp**

*Compute simple random sample size for estimating a proportion*

Description

Compute the simple random sample size for estimating a proportion based on different precision requirements.

Usage

nProp(CV0 = NULL, V0 = NULL, pU = NULL, N = Inf)

Arguments

- **CV0**: target value of coefficient of variation of the estimated proportion
- **V0**: target value of variance of the estimated proportion
- **pU**: population proportion
- **N**: number of units in finite population

Details

The precision requirement of \( p_s \) can be set based on either a target coefficient of variation, \( CV_0 \), or a target variance, \( V_0 \). In either case, a value of \( p_U \) must be supplied.

Value

numeric sample size
**nProp2sam**

**Description**

Compute a simple random sample size for estimating the difference in proportions when samples overlap.

**Usage**

```r
nProp2sam(px, py, pxy, g, r, alt, sig.level=0.05, pow=0.80)
```

**Arguments**

- `px`: proportion in group 1
- `py`: proportion in group 2
- `pxy`: proportion in the overlap has the characteristic in both samples
- `g`: proportion of sample 1 that is in the overlap with sample 2
- `r`: ratio of the size of sample 1 to that of sample 2
- `alt`: should the test be 1-sided or 2-sided; allowable values are `alt="one.sided"` or `alt="two.sided"`.
- `sig.level`: significance level of the hypothesis test
- `pow`: desired power of the test

**Examples**

```r
# srs sample size so that CV of estimated proportion is 0.05
# assuming the population is large and pU=0.01
# Both examples below are equivalent
nProp(V0=0.0005^2, N=Inf, pU=0.01) #or
nProp(CV0=0.05, N=Inf, pU=0.01)

# srswor sample size so that half-width of 2-sided 95% CI is 0.005
nProp(V0=(0.005/1.96)^2, N=Inf, pU=0.01)
```
Details

`nProp2sam` computes sample sizes in two groups that are required for testing whether the difference in group proportions is significant. The power of the test is one of the input parameters. The samples have a specified proportion of units in common.

Value

List with values:

- `n1`: sample size in group 1
- `n2`: sample size in group 2
- `px, py, pxy`: input values of the `px`, `py`, `pxy` parameters
- `gamma`: proportion of sample 1 that is in the overlap with sample 2
- `r`: ratio of the size of sample 1 to that of sample 2
- `alt`: type of test: one-sided or two-sided
- `sig.level`: significance level of test
- `power`: power of the test

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

- `nDep2sam`

Examples

`nProp2sam(px=0.5, py=0.55, pxy=0.45, g=0.5, r=1, alt="two.sided")`
nPropMoe

Simple random sample size for a proportion based on margin of error

Description

Calculates a simple random sample size based on a specified margin of error.

Usage

nPropMoe(moe.sw, e, alpha = 0.05, pU, N = Inf)

Arguments

- **moe.sw**: switch for setting desired margin of error (1 = CI half-width on the proportion; 2 = CI half-width on a proportion divided by \( p_U \))
- **e**: desired margin of error; either \( e = z_{1-\alpha/2} \sqrt{V(p_s)} \) or \( e = z_{1-\alpha/2} CV(p_s) \)
- **alpha**: 1 - (confidence level)
- **pU**: population proportion
- **N**: number of units in finite population

Details

The margin of error can be set as the half-width of a normal approximation confidence interval, \( e = z_{1-\alpha/2} \sqrt{V(p_s)} \), or as the half-width of a normal approximation confidence interval divided by the population proportion, \( e = z_{1-\alpha/2} CV(p_s) \). The type of margin of error is selected by the parameter `moe.sw` where `moe.sw=1` sets \( e = z_{1-\alpha/2} \sqrt{V(p_s)} \) and `moe.sw=2` sets i.e., \( e = \frac{z_{1-\alpha/2} \sqrt{V(p_s)}}{p_U} \).

Value

numeric sample size

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

nCont, nLogOdds, nProp, nWilson
Examples

# srs sample size so that half-width of a 95% CI is 0.01
# population is large and population proportion is 0.04
nPropMoe(moe.sw=1, e=0.01, alpha=0.05, pU=0.04, N=Inf)

# srswor sample size for a range of margins of error defined as
# half-width of a 95% CI
nPropMoe(moe.sw=1, e=seq(0.01,0.08,0.01), alpha=0.05, pU=0.5)

# srswor sample size for a range of margins of error defined as
# the proportion that the half-width of a 95% CI is of pU
nPropMoe(moe.sw=2, e=seq(0.05,0.1,0.2), alpha=0.05, pU=0.5)

NRadjClass

Class-based nonresponse adjustments

Description

Compute separate nonresponse adjustments in a set of classes.

Usage

NRadjClass(ID, NRclass, resp, preds=NULL, wts=NULL, type)

Arguments

- **ID**: identification value for a unit
- **NRclass**: vector of classes to use for nonresponse adjustment. Length is number of respondents plus nonrespondents
- **resp**: indicator for whether unit is a nonrespondent (must be coded 0) or respondent (must be coded 1)
- **preds**: response probabilities, typically estimated from a binary regression model as in pclass
- **wts**: vector of survey weights, typically base weights or base weights adjusted for unknown eligibility
- **type**: type of adjustment computed within each value of NRclass. Allowable codes are 1, 2, 3, 4, or 5. (1 = unweighted average of response propensities, i.e., preds; 2 = weighted average response propensity; 3 = unweighted response rate; 4 = weighted response rate; 5 = median response propensity)

Details

The input vectors should include both respondents and nonrespondents in a sample. A single value between 0 and 1 is computed in each nonresponse adjustment class to be used as a nonresponse adjustment. Five alternatives are available for computing the adjustment based on the value of type. The value of the adjustment is merged with individual unit data and stored in the RR field of the output data frame.
Value

A data frame of respondents only with four columns:

- **NRcl.no**: number of the nonresponse adjustment class for each unit
- **ID**: identification value for a unit
- **resp**: value of the resp variable (always 1)
- **RR**: nonresponse adjustment for each unit

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

`pclass`

Examples

```r
require(PracTools)
data(nhis)
out <- pclass(formula = resp ~ age + as.factor(sex) + as.factor(hisp) + as.factor(race),
               data = nhis, type = "unwtd", link="logit", numcl=5)
# unweighted average of response propensities within each class
zz <- NRadjClass(ID=nhis[,"ID"], NRclass = as.numeric(out$p.class), resp=nhis[,"resp"],
                 preds=out$propensities, wts=NULL, type=1)
```

---

**NRFUopt**

Sample sizes for a nonresponse follow-up study

Description

Compute optimal values of the first-phase sample size and the second-phase sampling fraction in a two-phase sample.

Usage

`NRFUopt(Ctot=NULL, c1, c2, theta, CV0=NULL, CVpop=NULL, N=Inf, type.sw)`
**Arguments**

- **Ctot**: total variable cost
- **c1**: cost per unit in phase-1
- **c2**: cost per unit in phase-2
- **theta**: probability of response for each unit
- **CV0**: target coefficient of variation for the estimated total or mean
- **CVpop**: Unit coefficient of variation
- **N**: Population size; default is Inf
- **type.sw**: type of allocation; "cost" = target total variable cost, "cv" = target coefficient of variation

**Details**

NRFUopt computes the optimal values of the first-phase sample size and the second-phase sampling fraction in a two-phase sample. Both stages are assumed to be selected using simple random sampling without replacement. If **type.sw**="cost", the optima are computed for a target total, expected cost across both phases. If **type.sw**="cv", the optima are computed for a target coefficient of variation for an estimated mean.

**Value**

List object with values:

- **allocation**: type of allocation: either "fixed cost" or "fixed CV"
- **"Total variable cost"**: expected total cost: fixed cost if **type.sw**="cost" or computed cost if **type.sw**="cv"; unrounded sample sizes are used in calculation
- **"Response rate"**: first-phase response rate
- **CV**: anticipated coefficient of variation (CV) if **type.sw**="cost" or target CV if **type.sw**="cv"
- **v.opt**: optimal fraction of first-phase nonrespondents to select for second-phase follow-up
- **n1.opt**: optimal number of units to sample at first-phase
- **"Expected n2"**: expected number of respondents obtained at second-phase
- **"srs sample for same cv"**: size of single-phase simple random sample (srs) needed to obtain same CV as the two-phase sample
- **"Cost Ratio: Two phase to srs"**: ratio of expected cost for two-phase sample to cost of single-phase srs

**Author(s)**

Richard Valliant, Jill A. Dever, Frauke Kreuter
References


Examples

# optima for fixed target CV
NRFUopt(Ctot=NULL, c1=50, c2=200, theta=0.5, CV0=0.05, CVpop=1, type.sw = "cv")

# optima for fixed total cost
NRFUopt(Ctot=100000, c1=50, c2=200, theta=0.5, CV0=NULL, CVpop=1, type.sw = "cost")

nWilson

Calculate a simple random sample size for estimating a proportion

Description

Calculate a simple random sample size for estimating a proportion using the Wilson method.

Usage

nWilson(moe.sw, alpha = 0.05, pU, e)

Arguments

moe.sw switch for setting desired margin of error (1 = CI half-width on the proportion; 2 = CI half-width on a proportion divided by pU)
alpha 1 - (confidence level)
pU population proportion
e desired margin of error; either the value of CI half-width or the value of the half-width divided by pU

Details

Calculate a simple random sample size using the Wilson (1927) method. A margin of error can be set as the CI half-width on the proportion (moe.sw=1) or as the CI half-width as a proportion of the population value pU (moe.sw=2).

Value

n.sam numeric sample size
"CI lower limit" lower limit of Wilson confidence interval with computed sample size
"CI upper limit" upper limit of Wilson confidence interval with computed sample size
"length of CI" length of Wilson confidence interval with computed sample size
pclass

Form nonresponse adjustment classes based on propensity scores

Description

Fit a binary regression model for response probabilities and divide units into a specified number of classes.

Usage

pclass(formula, data, link="logit", numcl=5, type, design=NULL)

Arguments

formula symbolic description of the binary regression model to be fitted as used in glm
data an optional data frame; must be specified if type=“unwtd”
link a specification for the model link function; allowable values are "logit", "probit", or "cloglog"
numcl number of classes into which units are split based on estimated propensities
type whether an unweighted or weighted binary regression should be fit; allowable values are "unwtd" or "wtd"
design sample design object; required if type="wtd"
Details

A typical formula has the form \( \text{response} \sim \text{terms} \) where \( \text{response} \) is a two-level variable coded as 0 or 1, or is a factor where the first level denotes nonresponse and the second level is response. If \( \text{type} = \text{"unwtd"} \), \text{glm} is used to fit an unweighted regression. If \( \text{type} = \text{"wtd"} \), \text{svyglm} in the \text{survey} package is used to fit a survey-weighted regression.

Value

A list with components:

- \text{p.class} \quad \text{propensity class for each unit}
- \text{propensities} \quad \text{estimated response probability for each unit}

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

\texttt{NRadjClass}

Examples

```r
# classes based on unweighted logistic regression
require(PracTools)
data(nhis)
out <- pclass(formula = resp ~ age + as.factor(sex) + as.factor(hisp) + as.factor(race),
data = nhis, type = "unwtd", link="logit", numcl=5)
table(out$p.class, useNA="always")
summary(out$propensities)
# classes based on survey-weighted logistic regression
require(survey)
nhis.dsgn <- svydesign(ids = ~psu, strata = ~stratum, data = nhis, nest = TRUE, weights = ~svywt)
out <- pclass(formula = resp ~ age + as.factor(sex) + as.factor(hisp) + as.factor(race),
type = "wtd", design = nhis.dsgn, link="logit", numcl=5)
table(out$p.class, useNA="always")
summary(out$propensities)
```
smho.N874  

Survey of Mental Health Organizations Data

Description

Data from the 1998 Survey of Mental Health Organizations (SMHO)

Usage

data(smho.N874)

Format

A data frame with 874 observations on the following 6 variables.

- **EXPTOTAL**  Total expenditures in 1998
- **BEDS**  Total inpatient beds
- **SEENCNT**  Unduplicated client/patient count seen during year
- **EOYCNT**  End of year count of patients on the role
- **FINDIRCT**  Hospital receives money from the state mental health agency (1=Yes; 2=No)
- **hosp.type**  Hospital type (1 = Psychiatric; 2 = Residential or veterans; 3 = General; 4 = Outpatient, partial care; 5 = Multi-service, substance abuse)

Details

The 1998 SMHO was conducted by the U.S. Substance Abuse and Mental Health Services Administration. It collected data on mental health care organizations and general hospitals that provide mental health care services, with an objective to develop national and state level estimates for total expenditure, full time equivalent staff, bed count, and total caseload by type of organization. The population omits one extreme observation in the smho98 population and has fewer variables than smho98. smho.N874 contains observations on 874 facilities.

Source

Substance Abuse and Mental Health Services Administration

References


See Also

smho98
Examples

```r
data(smho98)
str(smho98)
```

---

**Description**

Data from the 1998 Survey of Mental Health Organizations (SMHO)

**Usage**

```r
data(smho98)
```

**Format**

A data frame with 875 observations on the following variables.

- **STRATUM**  Sample design stratum (1 = Psychiatric Hospital, private; 2 = Psychiatric Hospital, public; 3 = Residential, children; 4 = Residential, adults; 5 = General Hospital, public, inpatient or residential care; 6 = General Hospital, public, outpatient care only; 7 = General Hospital, private, inpatient or residential care; 8 = General Hospital, private, outpatient care only; 9 = Military Veterans, inpatient or residential care; 10 = Military Veterans, outpatient care only; 11 = Partial Care; 12 = Outpatient care, private; 13 = Outpatient care, public; 14 = Multi-service, private; 15 = Multi-service, public; 16 = Substance Abuse)
- **BEDS**  Total inpatient beds
- **EXPTOTAL**  Total expenditures in 1998
- **SEENCNT**  Unduplicated client/patient count seen during year
- **EOYCNT**  End of year count of patients on the role
- **Y_IP**  Number of inpatient visits during year
- **OPCSFRST**  Number of outpatients on the rolls on the first day of the reporting year
- **OPCSADDS**  Number of outpatients admitted, readmitted, or transferred to the organization during the reporting year for less than a 24 hour period and not overnight
- **OPCSVIST**  Number of outpatient visits during the reporting year for less than a 24 hour period and not overnight
- **EMGWALK**  Number of emergency walk-ins during the reporting year
- **PSYREHAB**  Number of visits for psychiatric rehabilitation services
- **IPCSADDS**  Number of residential patients added during the reporting year or patients admitted for more than a 24 hour period
Details

The 1998 SMHO was conducted by the U.S. Substance Abuse and Mental Health Services Administration. It collected data on mental health care organizations and general hospitals that provide mental health care services, with an objective to develop national and state level estimates for total expenditure, full time equivalent staff, bed count, and total caseload by type of organization.

Source

Substance Abuse and Mental Health Services Administration

References


See Also

smho.N874

Examples

data(smho98)
str(smho98)
summary(smho98)

allocate a sample to strata

Description

Compute the proportional, Neyman, cost-constrained, and variance-constrained allocations in a stratified simple random sample.

Usage

strAlloc(n.tot = NULL, Nh = NULL, Sh = NULL, cost = NULL, ch = NULL, V0 = NULL, CV0 = NULL, ybarU = NULL, alloc)

Arguments

n.tot fixed total sample size
Nh vector of population stratum sizes \( (N_h) \) or pop stratum proportions \( (W_h) \)
Sh stratum unit standard deviations \( (S_h) \), required unless alloc = "prop"
cost total variable cost
ch vector of costs per unit in stratum \( h \) \( (c_h) \)
strAlloc

V0 fixed variance target for estimated mean
CV0 fixed CV target for estimated mean
ybarU population mean of $y$ ($\bar{y}_U$)
alloc type of allocation; must be one of "prop", "neyman", "totcost", "totvar"

Details

alloc="prop" computes the proportional allocation of a fixed total sample size, $n_{tot}$, to the strata. alloc="neyman" computes the allocation of a fixed total sample size, $n_{tot}$, to the strata that minimizes the variance of an estimated mean. alloc="totcost" computes the allocation of a fixed total sample size, $n_{tot}$, to the strata that minimizes the variance of an estimated mean subject to the fixed total cost. alloc="totvar" computes the allocation that minimizes total cost subject to the target coefficient of variation, $CV_0$, or the target variance, $V_0$, of the estimated mean.

Value

For proportional allocation, a list with values:

alloc type of allocation: "prop", "neyman", "totcost", "totvar"
Nh vector of population sizes ($N_h$) or pop stratum proportions ($W_h$)
h vector of stratum sample sizes
"nh/n" proportion of sample allocated to each stratum

For other allocations, the three components above plus:

Sh stratum unit standard deviations ($S_h$)
"anticipated SE of estimated mean" Anticipated SE of the estimated mean for the computed allocation

Author(s)

Richard Valliant, Jill A. Dever, Frauke Kreuter

References


See Also

nCont, nLogOdds, nProp, nPropMoe, nWilson
Examples

# Neyman allocation
Nh <- c(215, 65, 252, 50, 149, 144)
Sh <- c(26787207, 10645109, 6909676, 11085034, 9817762, 44553355)
strAlloc(n.tot = 100, Nh = Nh, Sh = Sh, alloc = "neyman")

# cost constrained allocation
ch <- c(1400, 200, 300, 600, 450, 1000)
strAlloc(Nh = Nh, Sh = Sh, cost = 100000, ch = ch, alloc = "totcost")

# allocation with CV target of 0.05
strAlloc(Nh = Nh, Sh = Sh, CV0 = 0.05, ch = ch, ybarU = 11664181, alloc = "totvar")

---

**ThirdGrade**

*Third grade population*

---

Description

The ThirdGrade data file is a population of students who participated in the Third International Mathematics and Science Study (TIMSS).

Usage

data(ThirdGrade)

Format

A data frame with 2,427 students on the following variables:

- **region** Geographic region of the U.S. (1 = Northeast; 2 = South; 3 = Central; 4 = West)
- **school.id** School identifier (1 - 135)
- **student.id** Student identifier (1 - 2427)
- **sex** Sex of student (1 = female; 2 = male)
- **language** Is language of test spoken at home? (1 = always; 2 = sometimes; 3 = never)
- **math** Mathematics test score
- **ethnicity** Ethnicity of student (1 = White, non-Hispanic; 2 = Black; 3 = Hispanic; 4 = Asian; 5 = Native American; 6 = Other)
- **science** Science test score
- **community** Type of location of school (2 = village or rural area; 3 = outskirts of a town or city; 4 = close to center of a town or city)
- **enrollment** Number of students in entire school
Details

The Third Grade population consists of 2,427 students in the U.S. who participated in the Third International Mathematics and Science Study (Caslyn, Gonzales, Frase 1999). The methods used in conducting the original study are given in TIMSS International Study Center (1996). Clusters are schools while units within clusters are the students.

Source

TIMSS International Study Center 1996.

References


Examples

data(ThirdGrade)
str(ThirdGrade)

---

**wtdvar**  
*Compute weighted variance*

Description

Compute an estimate of a population unit variance from a complex sample with survey weights.

Usage

wtdvar(x, w, na.rm=TRUE)

Arguments

- **x**: data vector
- **w**: vector of survey weights; must be same length as x
- **na.rm**: remove missing values (TRUE or FALSE)

Details

wtdvar is also used by *BW3stagePPSe* in estimating relvariance components.

Value

numeric estimate of population unit variance
Author(s)
Richard Valliant, Jill A. Dever, Frauke Kreuter

References

Examples
```r
x <- c(1:3)
wts <- c(4, 6, 8)
wtdvar(x=x, w=wts)
```
Index

* datasets
  Domainy1y2, 31
  hospital, 36
  labor, 37
  MDarea.pop, 38
  mibrfss, 39
  nhis, 50
  nhis.large, 52
  nhispart, 53
  smho.N874, 66
  smho98, 67
  ThirdGrade, 70

* methods
  BW2stagePPS, 3
  BW2stagePPSe, 5
  BW2stageSRS, 7
  BW3stagePPS, 8
  BW3stagePPSe, 10
  clusOpt2, 13
  clusOpt2fixedPSU, 15
  clusOpt3, 16
  clusOpt3fixedPSU, 18
  CVcalc2, 20
  CVcalc3, 21
  deff, 23
  deffCR, 25
  deffH, 27
  deffK, 29
  deffS, 30
  dub, 32
  HMT, 35
  nAuditAttr, 41
  nAuditMUS, 43
  nCont, 45
  nContMoe, 46
  nDep2sam, 48
  nDomain, 49
  nLogOdds, 55
  nProp, 56
  nProp2sam, 57
  nPropMoe, 59
  NRAdjClass, 60
  NRFUopt, 61
  nWilson, 63
  pclass, 64
  strAlloc, 68
  wtdvar, 71

* models
  gamEst, 33
  gammaFit, 34

* survey
  BW2stagePPS, 3
  BW2stagePPSe, 5
  BW2stageSRS, 7
  BW3stagePPS, 8
  BW3stagePPSe, 10
  clusOpt2, 13
  clusOpt2fixedPSU, 15
  clusOpt3, 16
  clusOpt3fixedPSU, 18
  CVcalc2, 20
  CVcalc3, 21
  deff, 23
  deffCR, 25
  deffH, 27
  deffK, 29
  deffS, 30
  dub, 32
  HMT, 35
  nAuditAttr, 41
  nAuditMUS, 43
  nCont, 45
  nContMoe, 46
  nDep2sam, 48
  nDomain, 49
  nLogOdds, 55
  nProp, 56
  nProp2sam, 57
nPropMoe, 59
NRadjClass, 60
NRFUopt, 61
nWilson, 63
pclass, 64
strAlloc, 68
wtdvar, 71

BW2stagePPS, 3, 6, 8, 10, 12
BW2stagePPSe, 3, 4, 5, 7–10, 12
BW2stageSRS, 4, 6, 7, 10, 12
BW3stagePPS, 4, 6, 8, 12
BW3stagePPSe, 4, 6, 8, 10, 10, 71
clusOpt2, 13, 16, 18, 19
clusOpt2fixedPSU, 15, 15, 18, 19
clusOpt3, 15, 16, 16, 19
clusOpt3fixedPSU, 15, 16, 18, 18
CVcalc2, 20
CVcalc3, 21, 21, 22
deff, 23, 26, 28–30
deffCR, 24, 25, 28–30
deffH, 24, 26, 27, 29, 30
deffK, 24, 26, 28, 29, 30
deffS, 24, 26, 28, 29, 30
Domainy1y2, 31
dub, 32
gamEst, 33, 35
gammaFit, 33, 34, 34
HMT, 35
hospital, 36
labor, 37
MDarea.pop, 38
mibrfss, 39

nAuditAttr, 41, 44
nAuditMUS, 43
nCont, 45, 47, 50, 56, 57, 59, 64, 69
nContMoe, 46
nDep2sam, 48, 58
nDomain, 49
nhis, 41, 50, 53
nhis.large, 41, 51, 52
nhispart, 53
nLogOdds, 46, 47, 50, 55, 57, 59, 64, 69
nProp, 46, 47, 50, 56, 59, 64, 69
nProp2sam, 49, 57
nPropMoe, 46, 47, 50, 55–57, 59, 64, 69
NRadjClass, 60, 65
NRFUopt, 61
nWilson, 46, 47, 50, 56, 57, 59, 63, 69
pclass, 61, 64
smho.N874, 66, 68
smho98, 66, 67
strAlloc, 68
ThirdGrade, 70
wtdvar, 71