Package ‘bbw’

January 17, 2018

Type Package

Title Blocked Weighted Bootstrap

Version 0.1.3

Description The blocked weighted bootstrap (BBW) is an estimation technique for use with data from two-stage cluster sampled surveys in which either prior weighting (e.g. population-proportional sampling or PPS as used in Standardized Monitoring and Assessment of Relief and Transitions or SMART surveys) or posterior weighting (e.g. as used in rapid assessment method or RAM and simple spatial sampling method or S3M surveys). The method was developed by Accion Contra la Faim, Brixton Health, Concern Worldwide, Global Alliance for Improved Nutrition, UNICEF Sierra Leone, UNICEF Sudan and Valid International. It has been tested by the Centers for Disease Control (CDC) using infant and young child feeding (IYCF) data. See Cameron et al (2008) <doi:10.1162/rest.90.3.414> for application of bootstrap to cluster samples. See Aaron et al (2016) <doi:10.1371/journal.pone.0163176> and Aaron et al (2016) <doi:10.1371/journal.pone.0162462> for application of the blocked weighted bootstrap to estimate indicators from two-stage cluster sampled surveys.

Imports car

Depends R (>= 3.0.1)

Suggests knitr, rmarkdown, testthat

License AGPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

URL https://github.com/validmeasures/bbw

BugReports https://github.com/validmeasures/bbw/issues

VignetteBuilder knitr

NeedsCompilation no

Author Mark Myatt [aut],
Ernest Guevarra [ctb, cre]

Maintainer Ernest Guevarra <ernestgmd@gmail.com>

Repository CRAN

Date/Publication 2018-01-17 09:18:12 UTC

R topics documented:

---

bbw *Blocked Weighted Bootstrap*

Description

The blocked weighted bootstrap (BBW) is an estimation technique for use with data from two-stage cluster sampled surveys in which either prior weighting (e.g. population proportional sampling or PPS as used in Standardized Monitoring and Assessment of Relief and Transitions or SMART surveys) or posterior weighting (e.g. as used in Rapid Assessment Method or RAM and Simple Spatial Sampling Method or S3M surveys). The method was developed by Accion contra la Faim (ACF), Brixton Health, Concern Worldwide, Global Alliance for Improved Nutrition (GAIN), UNICEF Sierra Leone, UNICEF Sudan and Valid International. It has been tested by the Centers for Disease Control (CDC) using infant and young child feeding (IYCF) data.

Details

The bootstrap technique is summarised in this article. The BBW used in RAM and S3M is a modification to the percentile bootstrap to include blocking and weighting to account for a complex sample design.

With RAM and S3M surveys, the sample is complex in the sense that it is an unweighted cluster sample. Data analysis procedures need to account for the sample design. A blocked weighted bootstrap (BBW) can be used:

Blocked The block corresponds to the primary sampling unit ($PSU = cluster$). PSUs are resampled with replacement. Observations within the resampled PSUs are also sampled with replacement.

`Weighted RAM` and `S3M` samples do not use `population proportional sampling (PPS)` to weight the sample prior to data collection (e.g. as is done with `SMART` surveys). This means that a posterior weighting procedure is required. `BBW` uses a `"roulette wheel"` algorithm to weight (i.e. by population) the selection probability of `PSUs` in bootstrap replicates.

In the case of prior weighting by `PPS` all clusters are given the same weight. With posterior weighting (as in `RAM` or `S3M`) the weight is the population of each `PSU`. This procedure is very similar to the fitness proportionate selection technique used in evolutionary computing.

A total of m `PSUs` are sampled with replacement for each bootstrap replicate (where m is the number of `PSUs` in the survey sample).

The required statistic is applied to each replicate. The reported estimate consists of the `0.025th (95% LCL)`, `0.5th (point estimate)`, and `0.975th (95% UCL)` quantiles of the distribution of the statistic across all survey replicates.

Early versions of the `BBW` did not resample observations within `PSUs` following:

Cameron AC, Gelbach JB, Miller DL, Bootstrap-based improvements for inference with clustered errors, Review of Economics and Statistics, 2008:90;414–427 `doi.org/10.1162/rest.90.3.414`

and used a large number (e.g. 3999) survey replicates. Current versions of the `BBW` resample observations within `PSUs` and use a smaller number of survey replicates (e.g. $n = 400$). This is a more computationally efficient approach

bootBW   `bootBW` : *Blocked Weighted Bootstrap*

Description

The `blocked weighted bootstrap (BBW)` is an estimation technique for use with data from two-stage cluster sampled surveys in which either prior weighting (e.g. `population proportional sampling` or `PPS` as used in `SMART` surveys) or posterior weighting (e.g. as used in `RAM` and `S3M` surveys).

Usage

```
bootBW(x, w, statistic, params, outputColumns, replicates = 400)
```

Arguments

`x`	A data frame with primary sampling unit (PSU) in column named psu
`w`	A data frame with primary sampling unit (PSU) in column named psu and survey weight (i.e. PSU population) in column named pop
`statistic`	A function operating on data in `x` (see Example)
`params`	Parameters (named columns in `x`) passed to the function specified in `statistic`
`outputColumns`	Names of columns in output data frame
`replicates`	Number of bootstrap replicates

Value

A data frame with:

ncol = length(outputColumns)

nrow = replicates

names = outputColumns

Examples

```
# Example function - estimate a proportion for a binary (0/1) variable):

oneP <- function(x, params) {
  v1 <- params[1]
  v1Data <- x[[v1]]
  oneP <- mean(v1Data, na.rm = TRUE)
  return(oneP)
}

# Example call to bootBW function using RAM-OP test data:

bootP <- bootBW(x = indicatorsHH,
                w = villageData,
                statistic = oneP,
                params = "anc1",
                outputColumns = "anc1",
                replicates = 9)

# Example estimate with 95% CI:

quantile(bootP, probs = c(0.500, 0.025, 0.975), na.rm = TRUE)
```

---

bootClassic *bootClassic*

Description

Simple proportion statistics function for bootstrap estimation

Usage

```
bootClassic(x, params)
```

Arguments

x	A data frame with `primary sampling unit (PSU)` in column named `psu` and with data column/s containing the binary variable/s (0/1) of interest with column names corresponding to `params` values
params	A vector of column names corresponding to the binary variables of interest contained in `x`

Value

A numeric vector of the mean of each binary variable of interest with length equal to `length(params)`

Examples

```
# Example call to bootClassic function

meanResults <- bootClassic(x = indicatorsHH,
                           params = "anc1")
```

bootPROBIT *bootPROBIT*

Description

PROBIT statistics function for bootstrap estimation

Usage

```
bootPROBIT(x, params, threshold = THRESHOLD)
```

Arguments

`x` A data frame with `primary sampling unit (PSU)` in column named `psu` and with data column/s containing the continuous variable/s of interest with column names corresponding to `params` values

`params` A vector of column names corresponding to the continuous variables of interest contained in `x`

`threshold` cut-off value for continuous variable to differentiate case and non-case

Value

A numeric vector of the PROBIT estimate of each continuous variable of interest with length equal to `length(params)`

Examples

```
# Example call to bootBW function:

bootPROBIT(x = indicatorsCH1,
           params = "muac1",
           threshold = 115)
```

indicatorsCH1 *Child Morbidity, Health Service Coverage, Anthropometry*

Description

Child indicators on morbidity, health service coverage and anthropometry calculated from survey data collected in survey conducted in 4 districts from 3 regions in Somalia.

Usage

```
indicatorsCH1
```

Format

A data frame with 14 columns and 3090 rows.

`psu` The PSU identifier. This must use the same coding system used to identify the PSUs that is used in the indicators dataset

`mID` The mother identifier

`cID` The child identifier

`ch1` Diarrhoea in the past 2 weeks (0/1)

`ch2` Fever in the past 2 weeks (0/1)

`ch3` Cough in the past 2 weeks (0/1)

`ch4` Immunisation card (0/1)

`ch5` BCG immunisation (0/1)

`ch6` Vitamin A coverage in the past month (0/1)

`ch7` Anti-helminth coverage in the past month (0/1)

`sex` Sex of child

`muac1` Mid-upper arm circumference in mm

`muac2` Mid-upper arm circumference in mm

`oedema` Oedema (0/1)

Source

Mother and child health and nutrition survey in 3 regions of Somalia

indicatorsCH2 *Infant and Child Feeding Index*

Description

Infant and young child feeding indicators using the infant and child feeding index (ICFI) by Arimond and Ruel. Calculated from survey data collected in survey conducted in 4 districts from 3 regions in Somalia.

Usage

```
indicatorsCH2
```

Format

A data frame with 13 columns and 2083 rows.

`psu` The PSU identifier. This must use the same coding system used to identify the PSUs that is used in the indicators dataset

`mID` The mother identifier

`cID` The child identifier

`ebf` Exclusive breastfeeding (0/1)

`cbf` Continued breastfeeding (0/1)

`ddd` Dietary diversity (0/1)

`mfd` Meal frequency (0/1)

`icfi` Infant and child feeding index (from 0 to 6)

`iycf` Good IYCF

`icfiProp` Good ICFI

`age` Child's age

`bf` Child is breastfeeding (0/1)

`bfStop` Age in months child stopped breastfeeding

Source

Mother and child health and nutrition survey in 3 regions of Somalia

`indicatorsHH` *Mother Indicators Dataset*

Description

Mother indicators for health and nutrition calculated from survey data collected in survey conducted in 4 districts from 3 regions in Somalia.

Usage

```
indicatorsHH
```

Format

A data frame with 24 columns and 2136 rows:

`psu` The PSU identifier. This must use the same coding system used to identify the PSUs that is used in the indicators dataset

`mID` The mother identifier

`mMUAC` Mothers with mid-upper arm circumference < 230 mm (0/1)

`anc1` At least 1 antenatal care visit with a trained health professional (0/1)

`anc2` At least 4 antenatal care visits with any service provider (0/1)

`anc3` FeFol coverage (0/1)

`anc4` Vitamin A coverage (0/1)

`wash1` Improved sources of drinking water (0/1)

`wash2` Improved sources of other water (0/1)

`wash3` Probable safe drinking water (0/1)

`wash4` Number of litres of water collected in a day

`wash5` Improved toilet facilities (0/1)

`wash6` Human waste disposal practices / behaviour (0/1)

`wash7a` Handwashing score (from 0 to 5)

`wash7b` Handwashing score of 5 (0/1)

`hhs1` Household hunger score (from 0 to 6)

`hhs2` Little or no hunger (0/1)

`hhs3` Moderate hunger (0/1)

`hhs4` Severe hunger (0/1)

`mfg` Mother's dietary diversity score

`pVitA` Plant-based vitamin A-rich foods (0/1)

`aVitA` Animal-based vitamin A-rich foods (0/1)

`xVitA` Any vitamin A-rich foods (0/1)

`iron` Iron-rich foods (0/1)

Source

Mother and child health and nutrition survey in 3 regions of Somalia

`recode` *Recode*

Description

Utility function that recodes variables based on user recode specifications. Handles both numeric or factor variables.

Usage

```
recode(var, recodes, afr, anr = TRUE, levels)
```

Arguments

`var` Variable to recode

`recodes` Character string of recode specifications:

- Recode specifications in a character string separated by semicolons of the form `input=output` as in: `"1=1;2=1;3:6=2;else=NA"`
- If an input value satisfies more than one specification, then the first (reading from left to right) is applied
- If no specification is satisfied, then the input value is carried over to the result unchanged
- `NA` is allowed on both input and output
- The following recode specifications are supported:

Specification	**Example**	**Notes**
Single values	`9=NA`	
Set of values	`c(1,2,5)=1`	The left-hand-side is any R function call that returns a vector
	`seq(1,9,2)='odd'`	
	`1:10=1`	
Range of values	`7:9=3`	Special values `lo` and `hi` may be used
	`lo:115=1`	
Other values	`else=NA`	

- Character values are quoted as in :
  `recodes = "c(1,2,5)='sanitary' else='unsanitary'"`
- The output may be the (scalar) result of a function call as in:
  `recodes = "999=median(var, na.rm = TRUE)"`
- Users are advised to carefully check the results of `recode()` calls with any outputs that are the results of a function call.
- The output may be the (scalar) value of a variable as in:
  `recodes = "999=scalarVariable"`

- If all of the output values are numeric, and if `'afr'` is `FALSE`, then a numeric result is returned; if `var` is a factor then (by default) so is the result.

`afr` Return a factor. Default is TRUE if `var` is a factor and is FALSE otherwise

`anr` Coerce result to numeric (default is TRUE)

`levels` Order of the levels in the returned factor; the default is to use the sort order of the level names.

Value

Recoded variable

Examples

```
# Recode values from 1 to 9 to varios specifications
var <- sample(x = 1:9, size = 100, replace = TRUE)

# Recode single values
recode(var = var, recodes = "9=NA”)

# Recode set of values
recode(var = var, recodes = "c(1,2,5)=1”)

# Recode range of values
recode(var = var, recodes = "1:3=1;4:6=2;7:9=3”)

# Recode other values
recode(var = var, recodes = "c(1,2,5)=1;else=NA”)
```

`villageData` *Cluster Population Weights Dataset*

Description

Dataset containing cluster population weights for use in performing posterior weighting with the blocked weighted bootstrap approach. This dataset is from a mother and child health and nutrition survey conducted in 4 districts from 3 regions in Somalia.

Usage

```
villageData
```

Format

A data frame with 6 columns and 117 rows:

`region` Region in Somalia from which the cluster belongs to

`district` District in Somalia from which the cluster belongs to

- `psu` The PSU identifier. This must use the same coding system used to identify the PSUs that is used in the indicators dataset
- `lon` Longitude coordinate of the cluster
- `lat` Latitude coordinate of the cluster
- `pop` Population size of the cluster

Source

Mother and child health and nutrition survey in 3 regions of Somalia

Index