

Package ‘samplingin’

May 11, 2024

Title Dynamic Survey Sampling Solutions

Version 1.1.0

Description A robust solution employing the SRS (Simple Random Sampling), systematic and PPS (Probability Proportional to Size) sampling methods, ensuring a methodical and representative selection of data. Seamlessly allocate predetermined allocations to smaller levels.

License MIT + file LICENSE

Depends R (>= 2.10)

Imports base, data.table, dplyr, magrittr, rlang, sampling, stats, utils

Suggests knitr, rmarkdown

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

NeedsCompilation no

Author Choerul Afifanto [aut, cre, cph]

Maintainer Choerul Afifanto <choerulafifanto@gmail.com>

Repository CRAN

Date/Publication 2024-05-11 08:50:09 UTC

R topics documented:

| | |
|------------------------------|---|
| alokasi_dt | 2 |
| doSampling | 2 |
| get_allocation | 5 |
| pop_dt | 6 |
| round_preserve_sum | 7 |

| | |
|--------------|----------|
| Index | 8 |
|--------------|----------|

`alokasi_dt`*Example of Allocation Data*

Description

Example of Allocation Data for Sampling Purposes

Usage`alokasi_dt`**Format**`alokasi_dt:`

A data frame with 34 rows and 3 columns:

kdprov province code

jml_kabkota Population or number of regencies/cities

n_primary Sample Allocation ...

`doSampling`*Select Samples Given its Parameters*

Description

Samples selection using systematic or PPS (Probability Proportional to Size) sampling method.

Usage

```
doSampling(  
  pop,  
  alloc,  
  nsample,  
  type,  
  strata = NULL,  
  ident = c("kdprov", "kdkab"),  
  implicitby = NULL,  
  method = "systematic",  
  auxVar = NA,  
  seed = 1,  
  predetermined_rn = NULL,  
  is_secondary = FALSE,  
  verbose = TRUE  
)
```

Arguments

| | |
|------------------|--|
| pop | pop dataframe |
| alloc | allocation dataframe |
| nsample | variable on alloc df as allocation sample |
| type | type value for sample classification ('U' = Primary Samples, 'P' = Secondary Samples) |
| strata | strata variable, must available on both pop and alloc dataframe |
| ident | group by on allocation dataframe |
| implicitby | variable used as implicit stratification |
| method | method of sampling: "systematic" (the default), "srs" or "pps" |
| auxVar | auxiliary variable for pps sampling (method = "pps") |
| seed | seed |
| predetermined_rn | predetermined random number variable on allocation dataframe, the default value is NULL, random number will be generated randomly |
| is_secondary | if the value is TRUE, it will maintains existing primary samples and selects units that have not been selected as samples (FALSE as default) |
| verbose | verbose (TRUE as default) |

Value

list of population data ("pop"), selected samples ("sampledf"), and details of sampling process ("details")

Examples

```
library(samplingin)
library(magrittr)
library(dplyr)

# Simple Random Sampling (SRS)
dtSampling_srs = doSampling(
  pop      = pop_dt
  , alloc  = alokasi_dt
  , nsample = "n_primary"
  , type   = "U"
  , ident  = c("kdprov")
  , method = "srs"
  , auxVar = "Total"
  , seed   = 7892
)

# Population data with flag sample
pop_dt = dtSampling_srs$pop
```

```

# Selected Samples
dsampel = dtSampling_srs$sampledf

# Details of sampling process
rincian = dtSampling_srs$details

# PPS Sampling
dtSampling_pps = doSampling(
  pop      = pop_dt
  , alloc  = alokasi_dt
  , nsample = "n_primary"
  , type   = "U"
  , ident  = c("kdprov")
  , method = "pps"
  , auxVar = "Total"
  , seed   = 1234
)

# Population data with flag sample
pop_dt = dtSampling_pps$pop

# Selected Samples
dsampel = dtSampling_pps$sampledf

# Details of sampling process
rincian = dtSampling_pps$details

# Systematic Sampling
dtSampling_sys = doSampling(
  pop      = pop_dt
  , alloc  = alokasi_dt
  , nsample = "n_primary"
  , type   = "U"
  , ident  = c("kdprov")
  , method = "systematic"
  , seed   = 4321
)

# Population data with flag sample
pop_dt = dtSampling_sys$pop

# Selected Samples
dsampel = dtSampling_sys$sampledf

# Details of sampling process
rincian = dtSampling_sys$details

# Systematic Sampling (Secondary Samples)

alokasi_dt_p = alokasi_dt %>%
  mutate(n_secondary = 2 * n_primary)

dtSampling_sys_p = doSampling(

```

```

    pop          = dtSampling_sys$pop
    , alloc      = alokasi_dt_p
    , nsample    = "n_secondary"
    , type       = "P"
    , ident      = c("kdprov")
    , method     = "systematic"
    , seed       = 6789
    , is_secondary = TRUE
  )

# Population data with flag sample
pop_dt = dtSampling_sys_p$pop

# Selected Samples
dsampel = dtSampling_sys_p$sampelf

# Details of sampling process
rincian = dtSampling_sys_p$details

# Systematic Sampling with predetermined random number (predetermined_rn parameter)
alokasi_dt_rn = alokasi_dt %>% rowwise() %>% mutate(ar = runif(n(),0,1)) %>% ungroup

dtSampling_sys = doSampling(
  pop          = pop_dt
  , alloc      = alokasi_dt_rn
  , nsample    = "n_primary"
  , type       = "U"
  , ident      = c("kdprov")
  , method     = "systematic"
  , predetermined_rn = "ar"
  , seed       = 4321
)

# Population data with flag sample
pop_dt = dtSampling_sys$pop

# Selected Samples
dsampel = dtSampling_sys$sampelf

# Details of sampling process
rincian = dtSampling_sys$details

```

get_allocation

Allocate Predetermined Allocations to Smaller Levels

Description

Allocate predetermined allocations to smaller levels using proportional allocation method

Usage

```
get_allocation(data, n_alloc, group, pop_var = "jml", secondary = 0)
```

Arguments

| | |
|-----------|--|
| data | population tabulation dataframe |
| n_alloc | total allocation dataframe |
| group | group of allocation level to be obtained |
| pop_var | population variable in data |
| secondary | how many times the secondary sample compares to primary sample |

Value

allocation at more detailed level

Examples

```
library(samplingin)
library(magrittr)

contoh_alokasi = alokasi_dt %>%
  dplyr::select(-n_primary) %>%
  dplyr::mutate(nasional = 1)

alokasi_dt = get_allocation(
  data = contoh_alokasi
  , n_alloc = 100
  , group = c("nasional")
  , pop_var = "jml_kabkota"
)
```

pop_dt

Indonesian Population (SP2020)

Description

Tabulation of Indonesia's population based on the results of the 2020 population census by re-gency/city and gender

Usage

```
pop_dt
```

Format

pop_dt:

A data frame with 514 rows and 8 columns:

idkab region id

kdprov province code

kdkab regency/city code

nmprov province name

nmkab regency/city name

Laki-laki Male Population

Perempuan Female Population

Total Total Population ...

Source

<https://sensus.bps.go.id/main/index/sp2020>

round_preserve_sum *round_preserve_sum*

Description

round_preserve_sum

Usage

`round_preserve_sum(x, digits = 0)`

Arguments

x a number

digits 0 (default)

Index

* datasets

alokasi_dt, [2](#)

pop_dt, [6](#)

alokasi_dt, [2](#)

doSampling, [2](#)

get_allocation, [5](#)

pop_dt, [6](#)

round_preserve_sum, [7](#)