Package 'hatsurvey'

April 29, 2026

Type Package

Title Survey Indicator Estimation for Complex Survey Designs

Version 1.1.1

Description Estimates survey indicators using complex survey designs.
Supports mean, proportion, and ratio estimation with multi-stage stratified sampling,
weights, and finite population correction. The output is designed to be comparable
to results from 'SPSS' (Statistical Package for the Social Sciences) Complex Samples procedures.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Imports survey, stats

NeedsCompilation no

Author Asy-Syaja'ul Haqqul Amin [aut, cre]

Maintainer Asy-Syaja'ul Haqqul Amin <haqqul.amin06@gmail.com>

Repository CRAN

Date/Publication 2026-04-29 18:40:13 UTC

Contents

`datause` *Example Survey dataset*

Description

A sample dataset derived from Household Survey used for demonstrating survey estimation functions.

Usage

```
datause
```

Format

A data frame with several variables:

CR509 School participation indicator

R101 Province (factor)

JMLH_PDDK Population count

CRCOB Eligibility indicator

IDSUBSLS Primary Sampling Unit (PSU) identifier. This variable represents the first-stage sampling unit (e.g., census block or sub-subsample area) selected during the first stage of sampling. Each PSU is uniquely identified within a stratum.

IDRUTA Secondary Sampling Unit (SSU) identifier. This variable represents the second-stage sampling unit (household level). Households are selected within each PSU during the second stage of sampling.

IDIDV Tertiary Sampling Unit (TSU) identifier. This variable represents the third-stage sampling unit (individual level). Individuals are selected within households during the third stage of sampling.

STRATA Stratification variable. Defines the survey strata, typically based on geographic or administrative regions. Stratification improves the precision of estimates and ensures representation across regions.

W_FINAL Final sampling weight. This weight reflects the inverse probability of selection, adjusted for non-response and calibrated to known population totals. It must be applied to produce unbiased estimates.

FPC1 Finite Population Correction (FPC) for the first stage. Represents the total number of PSUs in each stratum. Used to adjust variance estimation under sampling without replacement at the first stage.

FPC2 Finite Population Correction (FPC) for the second stage. Represents the total number of households within each PSU. Used for variance correction at the second sampling stage.

FPC3 Finite Population Correction (FPC) for the third stage. Represents the total number of individuals within each household. Used for variance correction at the third sampling stage.

The survey design follows a three-stage stratified cluster sampling scheme:

1. First stage: selection of PSUs (IDSUBSLS) within strata (STRATA)
2. Second stage: selection of households (IDRUTA) within PSUs
3. Third stage: selection of individuals (IDIDV) within households

The inclusion of FPC variables ensures correct variance estimation under without-replacement sampling assumptions.

Source

Simulated Household Survey Data

`hatsurvey` *hatsurvey*

Description

Computes survey indicator estimates using complex survey design from the 'survey' package. It supports three types of estimation:

- `"mean"`: mean or simple proportion (svymean)
- `"prop"`: ratio-based proportion (svyratio, returned in percentage)
- `"ratio"`: ratio of two variables (e.g., GER, NER, LFPR)

Usage

```
hatsurvey(
  x,
  y,
  denom = NULL,
  design,
  denom_value = NULL,
  success_value = NULL,
  data,
  survey.type
)
```

Arguments

`x`	Character. Name of the target variable (numerator).
`y`	Character. Name of the disaggregation (grouping) variable.
`denom`	Character. Name of the denominator variable (only for `"prop"` and `"ratio"`).
`design`	A survey design object created using `svydesign`.
`denom_value`	A vector of values used to filter the denominator (optional).
`success_value`	A vector of values considered as "success" in the numerator (optional).
`data`	Original data frame used to preserve factor level ordering of `y`.

`survey.type` Character. Type of estimation:

- `"mean"`
- `"prop"`
- `"ratio"`

Details

The output includes estimates, standard errors, relative standard errors, confidence intervals, variance, design effect, and unweighted counts for numerator and denominator.

Important notes:

- For `"mean"`, the variable `x` should be numeric or binary (0/1).
- For `"prop"` and `"ratio"`, ensure that `x` and `denom` are properly defined (e.g., 1 = event, 0 = non-event).
- The function uses `svyby`, so results follow the complex survey design.
- Category ordering follows the factor levels in `data[[y]]`.
- For `"prop"`, the estimate is computed as a ratio of totals, not as a simple mean. This is useful for population-based indicators.

Value

A data frame containing:

- `Variable` : Name of the target variable
- `Disaggregation` : Disaggregation category
- `Estimation` : Estimated value
- `SE` : Standard error
- `RSE` : Relative standard error (%)
- `Lower Conf.Int` : Lower bound of confidence interval
- `Upper Conf.Int` : Upper bound of confidence interval
- `Variance` : Variance of the estimate
- `DEFF` : Design effect
- `n_denom` : Unweighted denominator count
- `n_num` : Unweighted numerator count (for prop and ratio)

Examples

```
# --- Simple toydata
df <- data.frame(
  x = c(100, 0, 100, 100, 0, 100),
  denom = c(100, 100, 100, 100, 100, 100),
  y = factor(c("Urban","Urban","Rural","Rural","Urban","Rural")),
  w = c(2,1,3,1,2,1)
)
```

```
# Build simple survey design
dsgn <- survey::svydesign(id = ~1, data = df, weights = ~w)

# --- Proportion using proportion estimator
hatsurvey(
  x = "x",
  y = "y",
  denom = "denom",
  design = dsgn,
  denom_value = 100,
  success_value = 100,
  data = df,
  survey.type = "prop"
)

# --- Full example (complex survey)

data("datause")

# Prepare data
datause$R101 <- as.factor(datause$R101)
options(survey.lonely.psu = "certainty")
# Build complex survey design (3-stage, stratified, with FPC)
snlik.design <- survey::svydesign(
  id = ~IDSUBSLS + IDRUTA + IDIDV,
  strata = ~STRATA,
  data = subset(datause, !is.na(CR509)),
  weights = ~W_FINAL,
  fpc = ~FPC1 + FPC2 + FPC3,
  nest = TRUE
)

# --- Proportion (percentage via ratio)
# Example: proportion of CR509 == 100 over total population
hatsurvey(
  x = "CR509",
  y = "R101",
  denom = "JMLH_PDDK",
  design = snlik.design,
  denom_value = NULL,
  success_value = 100,
  data = subset(datause, !is.na(CR509)),
  survey.type = "prop"
)

# --- Ratio (e.g., conditional rate)
# Example: CR509 == 100 over CRCOB == 1
hatsurvey(
  x = "CR509",
  y = "R101",
  denom = "CRCOB",
  design = snlik.design,
  denom_value = 1,
```

```
  success_value = 100,
  data = subset(datause, !is.na(CR509)),
  survey.type = "ratio"
)

# --- Mean
hatsurvey(
  x = "CR509",
  y = "R101",
  denom = NULL,
  design = snlik.design,
  denom_value = NULL,
  success_value = NULL,
  data = subset(datause, !is.na(CR509)),
  survey.type = "mean"
)
```

Index