

Package ‘SampleSelectR’

September 12, 2026

Title Randomly Select Samples for Various Probability-Based Methods

Version 1.0.0

Description Randomly select samples using simple random sampling (SRS), systematic sampling, and various probability proportional to size (PPS) methods, including systematic PPS and sequential PPS (i.e., Chromy's method). Also includes functionality to allocate sample sizes across strata using proportional, power, Neyman, and optimal allocation methods, and to select samples within strata. Designed to make survey sample design and selection reproducible, efficient, and transparent for survey statisticians and researchers. Sampling methods follow Kalton (1983) <[doi:10.4135/9781412984683](https://doi.org/10.4135/9781412984683)> and Chromy (1979) <http://www.asasrms.org/Proceedings/papers/1979_081.pdf>.

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 4.1.0)

Imports data.table, rlang, stats, tidytable, utils

LazyData true

URL <https://github.com/RTIInternational/SampleSelectR>,
<https://rtiinternational.github.io/SampleSelectR/>

BugReports <https://github.com/RTIInternational/SampleSelectR/issues>

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Stephanie Zimmer [aut, cre],
John David Bunker, Jr. [aut],
Thomas Burkett [aut],
Philip Lee [aut],
Haby Sow [aut],
RTI International [cph, fnd]

Maintainer Stephanie Zimmer <sazimmer@rti.org>
Repository CRAN
Date/Publication 2026-09-12 12:10:02 UTC

Contents

allocate	2
chromy_pps	6
county_2023	7
ipeds	8
puma_2023	9
select_sample	10
serp_sort	13
srs	14
sys	15
sys_pps	16
Index	17

allocate	<i>Sample allocation</i>
----------	--------------------------

Description

Compute the proportional, power, Neyman, and optimal sample allocations.

Usage

```
allocate(  
  allocation,  
  N.h,  
  n.samp = NULL,  
  S.h = NULL,  
  c.h = NULL,  
  cost = NULL,  
  variance = NULL,  
  power = NULL,  
  lbound = 2,  
  outputs = "rounded"  
)
```

Arguments

allocation	type of allocation, must be one of "proportional", "power", "neyman", or "optimal".
------------	---

N.h	vector of population stratum sizes (N_h , all positive values), for example c(150, 600, 250).
	required for all allocation types.
n.samp	total sample size to be allocated (positive integer of length 1).
	required for the following allocation types: proportional, power, and Neyman, and NULL otherwise.
S.h	vector of stratum unit standard deviations (positive values same length as N.h) (S_h).
	required for the following allocation types: Neyman, and optimal, and NULL otherwise.
c.h	vector of cost per unit in stratum h (positive values same length as N.h) (c_h).
	required for the optimal allocation only, and NULL otherwise.
cost	total variable cost (positive value) ($C - c_0$).
	required for the cost-constrained optimal allocation only, and NULL otherwise.
variance	fixed variance target for estimated mean (positive value) (V_0).
	required for the precision-constrained optimal allocation only, and NULL otherwise.
power	power value for power allocation ($0 \leq \alpha \leq 1$).
	required for the power allocation only, and NULL otherwise.
lbound	minimum stratum-level sample size (positive integer of length 1). Default value is 2. If $N.h < lbound$ for a stratum, the sample size will be limited to N.h.
outputs	character vector representing whether to output: 1. the raw allocations before accounting for N.h, lbound, and n.samp if needed ["raw"], 2. the continuous version after accounting for the above ["adjusted"], and/or 3. the rounded version of the above ["rounded"] Default is to only return the final rounded version ["rounded"]. If one version is requested, the result will be a numeric vector. Otherwise, the result will be a named list matching the requested outputs.

Value

If one output type is requested, a numeric vector of allocations. If multiple output types are requested, a named list containing the requested allocation vectors.

Method

The *allocate* function allocates a sample size n on H strata using one of the following allocation methods:

1. Proportional allocation [n.samp, N.h, allocation = "proportional"]

$$n_h = n \times \frac{N_h}{\sum_{h=1}^H N_h}$$

where

n : total sample size to be allocated (function input is n.samp), and

N_h : population size of stratum h (function input is N.h).

2. Power allocation [n.samp, N.h, power, allocation = "power"]

$$n_h = n \times \frac{N_h^\alpha}{\sum_{h=1}^H N_h^\alpha}$$

where

α : a power value to control over-under-sampling with $0 \leq \alpha \leq 1$ (function input is power).

3. Neyman allocation [n.samp, N.h, S.h, allocation = "neyman"]

$$n_h = n \times \frac{N_h S_h}{\sum_{h=1}^H N_h S_h}$$

where

S_h : standard deviation of stratum h (function input is S.h).

4. Optimal allocation

- cost-constrained [N.h, S.h, c.h, cost, allocation = "optimal"]

$$n_h = (C - c_0) \times \frac{N_h S_h / \sqrt{c_h}}{\sum_{h=1}^H N_h S_h \sqrt{c_h}}$$

where

c_h : cost per unit in stratum h (function input is c.h), and

$(C - c_0)$: total variable cost (function input is cost)

- precision-constrained [N.h, S.h, c.h, variance, allocation = "optimal"]

$$n_h = N_h S_h / \sqrt{c_h} \times \frac{\sum_{h=1}^H N_h S_h \sqrt{c_h}}{V_0 \left(\sum_{h=1}^H N_h \right)^2 + \sum_{h=1}^H N_h S_h^2}$$

where

V_0 : fixed variance target for estimated mean (function input is variance)

The table below presents the relevant inputs for each type; when irrelevant inputs are entered, an error message will be displayed.

allocation	N.h	n.samp	S.h	c.h	cost	variance	lbound	power
proportional	X	X					X	
power	X	X					X	X
neyman	X	X	X				X	
optimal: cost-constrained	X		X	X	X		X	
optimal: precision-constrained	X		X	X		X	X	

Examples

```
# The first step is getting a frame summary
# Summarize the IPEDS dataset by OBEREG
# - N: number of universities per region
# - SD_ENRTOT: standard deviation of total enrollment per region
# - Filter out rows with missing ENRTOT to ensure accurate variance estimates

ipeds_summary <- ipeds |>
  tidytable::filter(!is.na(ENRTOT)) |>
  tidytable::group_by(OBEREG) |>
  tidytable::summarize(
    N = tidytable::n(),
    SD_ENRTOT = stats::sd(ENRTOT)
  ) |>
  tidytable::ungroup()

# Example of proportional allocation
ipeds_summary |>
  tidytable::mutate(
    n = allocate("proportional", N.h = N, n.samp = 500)
  )

# Example of power allocation
ipeds_summary |>
  tidytable::mutate(
    n = allocate("power", N.h = N, power = 0.5, n.samp = 500)
  )

# Example of Neyman allocation
ipeds_summary |>
  tidytable::mutate(
    n = allocate("neyman", N.h = N, n.samp = 500, S.h = SD_ENRTOT)
  )

# Example of Neyman allocation with a lower bound of 5
ipeds_summary |>
  tidytable::mutate(
    n = allocate("neyman", N.h = N, n.samp = 500, S.h = SD_ENRTOT, lbound = 5)
  )
```

chromy_pps

Select a sequential PPS sample

Description

Draws a sequential sample of size n . Each unit's probability of selection is proportional to its size measure. This is a minimum replacement method as discussed in Chromy (1979).

Usage

```
chromy_pps(frame, n, mos, outall = FALSE, curstrat = NULL)
```

Arguments

frame	The input data frame for the function to work on.
n	The sample size, the parameter expects an integer of length 1. The function will check if n is less than or equal to the number of rows in the input frame.
mos	The measure of size, the parameter expects a character string to indicate the variable to be used as the measure of size. The variable must exist on the frame and be a non-missing and non-negative numeric variable.
outall	Output all records or selected records. If outall is TRUE, then all records are returned and the following variables are created: SelectionIndicator, SamplingWeight, NumberHits, and ExpectedHits. If outall is FALSE, then the selected records are returned and the following variables are created: SamplingWeight, NumberHits, ExpectedHits.
curstrat	A character variable that specifies the current strata, only used as an assertion for the $n == N$ test.

Value

Returns an object of type tidytable that contains the weight, expected hits (selection probability for nonreplacement designs), and number of hits plus all original variables. Include a SelectionIndicator variable if outall=TRUE

References

Chromy, J. R. (1979). "Sequential Sample Selection Methods." In *Proceedings of the Survey Research Methods Section*, 401–406. Washington, DC: American Statistical Association. http://www.asasrms.org/Proceedings/papers/1979_081.pdf

Examples

```
# PPS sample of counties using population size as MOS
# LA county will be selected two or three times based on expected hits
# Cook, Harris, and Maricopa will be selected one or two times based on expected hits

county_2023 |>
```

```

tidytable::select(GEOID, Name, Pop_Tot) |>
chromy_pps(n = 75, mos = "Pop_Tot") |>
tidytable::arrange(desc(ExpectedHits))

county_2023 |>
tidytable::select(GEOID, Name, Pop_Tot) |>
chromy_pps(n = 75, mos = "Pop_Tot", outall = TRUE) |>
tidytable::arrange(desc(ExpectedHits))

```

county_2023	<i>American Community Survey (ACS) 5-Year Estimates, 2023 for counties</i>
-------------	--

Description

A select set of estimates from the ACS for all counties in the 50 states and District of Columbia.

Usage

```
county_2023
```

Format

A tibble/data.frame with 3,144 rows and 25 columns:

GEOID Geographic Identifier (GEOID) of county (character)
Name Name of county (character)
State State (Abbreviation) (character)
Region Census Region (factor)
Divison Census Division (factor)
Pop_Tot Total population (B03002) (numeric)
Pop_Pct_White_NH Percent of population who is White alone, non-Hispanic (B03002) (numeric)
Pop_Pct_Black_NH Percent of population who is Black or African American alone, non-Hispanic (B03002) (numeric)
Pop_Pct_AIAN_NH Percent of population who is American Indian and Alaska Native alone, non-Hispanic (B03002) (numeric)
Pop_Pct_Asian_NH Percent of population who is Asian alone, non-Hispanic (B03002) (numeric)
Pop_Pct_NHPI_NH Percent of population who is Native Hawaiian and Other Pacific Islander alone, non-Hispanic (B03002) (numeric)
Pop_Pct_Other_NH Percent of population who is another race (including 2 or more races), non-Hispanic (B03002) (numeric)
Pop_Pct_Hispanic Percent of population who is Hispanic (B03002) (numeric)
HU_Tot Total number of housing units (B25002) (numeric)

HU_Pct_Occupied Percent of housing units that are occupied (B25002) (numeric)
HU_Pct_Vacant Percent of housing units that are vacant (B25002) (numeric)
Pop_Pct_0004 Percent of population that are 0-4 years of age (S0101) (numeric)
Pop_Pct_0509 Percent of population that are 5-9 years of age (S0101) (numeric)
Pop_Pct_1014 Percent of population that are 10-14 years of age (S0101) (numeric)
Pop_Pct_2544 Percent of population that are 15-17 years of age (S0101) (numeric)
Pop_Pct_4564 Percent of population that are 18-24 years of age (S0101) (numeric)
Pop_Pct_6574 Percent of population that are 25-44 years of age (S0101) (numeric)
Pop_Pct_75plus Percent of population that are 45-64 years of age (S0101) (numeric)
Pop_Pct_1517 Percent of population that are 65-74 years of age (S0101) (numeric)
Pop_Pct_1824 Percent of population that are 75 years of age or older (S0101) (numeric)

Source

U.S. Census Bureau, "American Community Survey 5-Year Estimates" 2023, <https://api.census.gov/data/2023/acs/acs5>, accessed on May 30, 2025. Tables B03002, B25002, and S0101. Note - the tidycensus package was used to download the data from the Census API. Region and division are coded based on https://www2.census.gov/geo/pdfs/maps-data/maps/reference/us_regdiv.pdf

ipeds

IPEDS - Post secondary education institutions, Fall 2023

Description

A select set of characteristics for postsecondary educational institutions from the Integrated Postsecondary Education Data System (IPEDS) for operational institutions reporting fall enrollment data for 2023.

Usage

ipeds

Format

A tibble with 5914 rows and 16 columns:

UNITID Unique identification number of the institution (numeric)
INSTNM Institution (entity) name (character)
STABBR State abbreviation (character)
FIPS FIPS state code (numeric)
OBereg Bureau of Economic Analysis (BEA) regions (factor)
ICLEVEL Level of institution (factor)

SECTOR Sector of institution (factor)
LOCALE Degree of urbanization (Urban-centric locale) (factor)
DEGRANT Degree-granting status (factor)
HLOFFER Highest level of offering (factor)
ENRTOT Total enrollment (numeric)
EFUG Undergraduate enrollment (numeric)
EFUG1ST First-time degree/certificate-seeking undergraduate enrollment (numeric)
EFUGFT Full-time undergraduate enrollment (numeric)
EFGRAD Graduate enrollment (numeric)
EFGRADFT Full-time graduate enrollment (numeric)

Source

National Center for Education Statistics, "Integrated Postsecondary Education Data System " 2023, <https://nces.ed.gov/ipeds/datacenter/DataFiles.aspx?year=2023&surveyNumber=-1&sid=4737d338-5121-4355-bb91-01ffa92243ef&rtid=7>, downloaded on May 2, 2025.

puma_2023	<i>American Community Survey (ACS) 5-Year Estimates, 2023 for Public Use Microdata Areas (PUMAs)</i>
-----------	--

Description

A select set of estimates from the ACS for all PUMAs in the 50 states and District of Columbia.

Usage

puma_2023

Format

A tibble/data.frame with 2,462 rows and 25 columns:

GEOID Geographic Identifier (GEOID) of PUMA (character)
Name Name of PUMA (character)
State State (Abbreviation) (character)
Region Census Region (factor)
Division Census Division (factor)
Pop_Tot Total population (B03002) (numeric)
Pop_Pct_White_NH Percent of population who is White alone, non-Hispanic (B03002) (numeric)
Pop_Pct_Black_NH Percent of population who is Black or African American alone, non-Hispanic (B03002) (numeric)

Pop_Pct_AIAN_NH Percent of population who is American Indian and Alaska Native alone, non-Hispanic (B03002) (numeric)

Pop_Pct_Asian_NH Percent of population who is Asian alone, non-Hispanic (B03002) (numeric)

Pop_Pct_NHPI_NH Percent of population who is Native Hawaiian and Other Pacific Islander alone, non-Hispanic (B03002) (numeric)

Pop_Pct_Other_NH Percent of population who is another race (including 2 or more races), non-Hispanic (B03002) (numeric)

Pop_Pct_Hispanic Percent of population who is Hispanic (B03002) (numeric)

HU_Tot Total number of housing units (B25002) (numeric)

HU_Pct_Occupied Percent of housing units that are occupied (B25002) (numeric)

HU_Pct_Vacant Percent of housing units that are vacant (B25002) (numeric)

Pop_Pct_0004 Percent of population that are 0-4 years of age (S0101) (numeric)

Pop_Pct_0509 Percent of population that are 5-9 years of age (S0101) (numeric)

Pop_Pct_1014 Percent of population that are 10-14 years of age (S0101) (numeric)

Pop_Pct_2544 Percent of population that are 15-17 years of age (S0101) (numeric)

Pop_Pct_4564 Percent of population that are 18-24 years of age (S0101) (numeric)

Pop_Pct_6574 Percent of population that are 25-44 years of age (S0101) (numeric)

Pop_Pct_75plus Percent of population that are 45-64 years of age (S0101) (numeric)

Pop_Pct_1517 Percent of population that are 65-74 years of age (S0101) (numeric)

Pop_Pct_1824 Percent of population that are 75 years of age or older (S0101) (numeric)

Source

U.S. Census Bureau, "American Community Survey 5-Year Estimates" 2023, <https://api.census.gov/data/2023/acs/acs5>, accessed on May 30, 2025. Tables B03002, B25002, and S0101. Note - the tidycensus package was used to download the data from the Census API. Region and division are coded based on https://www2.census.gov/geo/pdfs/maps-data/maps/reference/us_regdiv.pdf

select_sample

Sample selection function

Description

Selects a random sample using a specified method and sample size. Selection can also optionally be stratified and/or include a measure of size (mos) if a PPS method is used.

Usage

```
select_sample(
  frame,
  method,
  n,
  outall = FALSE,
  strata = NULL,
  mos = NULL,
  sort_vars = NULL,
  sort_method = NULL
)
```

Arguments

frame	A data frame, data.table, or tibble from which to draw the sample. No default.
method	The desired sampling method. Valid options are "srs", "sys_eq", "sys_pps", and "chromy_pps". No default.
n	The sample size to draw from the frame. If strata is NULL, must be a positive integer. If strata is not NULL, must be a data.frame, tibble, or data.table with columns for each stratification variable as the same type and variable names as the frame plus a column with the sample size (sample_size) which is a positive integer. No default.
outall	A logical value indicating whether to return the entire frame with a selection indicator or just the sample. Default is FALSE.
strata	A vector of characters with variable names of strata. Default is NULL.
mos	A character string defining the variable name on the frame for the measure of size. If not NULL, must have method = c("sys_pps", "chromy_pps"). If NULL, must have method=c("srs", "sys"). Default is NULL.
sort_vars	A vector of characters indicating the variables that should be used to sort the frame. If not NULL, cannot have method = "srs". Default is NULL.
sort_method	A character string defining the method to implicitly sort the frame. Valid options are "serpentine" and "nest". Must coincide with sort_var; i.e., both must be NULL or both must be not NULL. Default is NULL.

Value

A tidytable object containing the entire frame with a selection indicator or just the sample, dependent on the value of outall. Selection probability and sampling weight are also included. May include various summary messages to the console when applicable for certain sampling methods.

Examples

```
# SRS of 100 US counties, using geographic region as strata
# n is a data frame containing the strata values and corresponding desired sample size
# Sample size column must be titled 'sample_size'

n_df_srs <- data.frame(
```

```

    Region = as.factor(c("Northeast", "Midwest", "South", "West")),
    sample_size = c(25, 25, 25, 25)
  )

county_2023 |>
  select_sample(method = "srs", n = n_df_srs, strata = "Region")

# Systematic sample of 250 US universities. Each unit has an equal probability of being selected
# Includes a nested sort of enrollment total within sector
# Returns all obs from original data frame with a selection indicator column

sample_sys_eq <- ipeds |>
  select_sample(
    method = "sys_eq", n = 250, outall = TRUE,
    sort_vars = c("SECTOR", "ENRTOT"), sort_method = "nest"
  )

# For samples taken with outall = TRUE, the sample size can be verified by summing
# the SelectionIndicator column.

sample_sys_eq
sum(sample_sys_eq$SelectionIndicator)

# Systematic PPS sample of 250 US universities. Each unit's probability of selection
# is proportional to its size measure.
# Using enrollment total as MOS
# Includes a nested sort of enrollment total within sector

sample_sys_pps <- ipeds |>
  select_sample(
    method = "sys_pps", n = 250, mos = "ENRTOT",
    sort_vars = c("SECTOR", "ENRTOT"), sort_method = "nest"
  )

# For pps samples, it is possible for a single sampling unit to be selected multiple times
# due to a large mos value. This is especially true as desired sample size increases. The
# result is the final sample may not meet the desired sample size. To verify the pps sample,
# the NumberHits column can be summed and should total to the desired sample size.

sample_sys_pps
sum(sample_sys_pps$NumberHits)

# Sequential aka Chromy's method PPS sample of 500 PUMAs, using geographic region as strata
# Includes a serpentine sort of geographic division then state
# Using population total as MOS, each unit's probability of selection is proportional to its
# size measure.
# Note that there may be a discrepancy between the desired and final sample sizes. The final
# sample size can be verified by totaling NumberHits.

n_df_chr <- data.frame(

```

```

    Region = as.factor(c("Northeast", "Midwest", "South", "West")),
    sample_size = c(125, 125, 125, 125)
  )

puma_2023 |>
  select_sample(
    method = "chromy_pps", n = n_df_chr, strata = "Region", mos = "Pop_Tot",
    sort_vars = c("Division", "State"), sort_method = "serpentine"
  )

```

serp_sort

*Serpentine sort function***Description**

Sorts the rows in a given data frame in a hierarchical fashion given a list of variables (...)

Usage

```
serp_sort(.data, ..., naorder = TRUE, sortID = FALSE)
```

Arguments

.data	A data frame. No default.
...	Variables to sort by - must be listed as character variables. No default.
naorder	Logical value for whether to put NA values at the end of the sorted data frame. Default value is TRUE and treats NA values as the largest values when sorting. A value of FALSE treats NA Values as the smallest values when sorting.
sortID	Logical value that indicates whether the sort group information used to implement the sorting should be added to .data. By default, sort group is FALSE which means the sort group information is not added to .data. When TRUE, and there are k variables to sort by, a set of k-1 sort group variables with names sortSerpj (j=1 to k-1) are added to .data.

Value

A copy of .data that is hierarchically sorted and, if specified, includes the sort group variables created for sorting

Examples

```

# Sort counties by Region, then Division, in a serpentine pattern
county_2023 |>
  serp_sort("Region", "Division")

# Keep the intermediate sort-group columns (sortSerp1, sortSerp2)
county_2023 |>
  serp_sort("Region", "Division", sortID = TRUE)

```

srs	<i>Simple random sampling function</i>
-----	--

Description

Draws a simple random sample of size `n`.

Usage

```
srs(frame, n, outall = FALSE, curstrat = NULL)
```

Arguments

<code>frame</code>	A data frame, <code>data.table</code> , or <code>tibble</code> from which to draw the sample. No default.
<code>n</code>	The sample size to draw from the frame. Must be a positive integer. No default.
<code>outall</code>	A logical value indicating whether to return the entire frame with a selection indicator or just the sample. Default is <code>FALSE</code> .
<code>curstrat</code>	A character string indicating the current stratum. Used only for printing messages when <code>n = N</code> . Default is <code>NULL</code> .

Value

A `tidytable` object containing the entire frame with a selection indicator or just the sample, dependent on the value of `outall`. Selection probability and sampling weight are also included. The sample size, `n`, and the population size, `N`, are printed to the console.

Examples

```
# Random sample of 200 universities, only sampled rows returned
ipeds |>
  tidytable::filter(!is.na(ENRTOT)) |>
  srs(n = 200, outall = FALSE)

# Return full dataset with selection indicators
ipeds |>
  tidytable::filter(!is.na(ENRTOT)) |>
  srs(n = 200, outall = TRUE)
```

sys

Systematic Sampling Without Replacement

Description

Draws a systematic sample of size `n` from a data frame. Each unit has an equal probability of being selected.

Usage

```
sys(frame, n, curstrat = NULL, outall = FALSE)
```

Arguments

<code>frame</code>	a <code>data.frame</code> , <code>tibble</code> , or <code>data.table</code> containing the sampling frame. Must have at least one row.
<code>n</code>	Integer. The desired sample size. Must be less than or equal to the number of rows in <code>frame</code> .
<code>curstrat</code>	Character or <code>NULL</code> . Optional stratum name for printing messages.
<code>outall</code>	logical indicator for whether full frame is returned or just the sample

Value

a `data.table` with the original columns plus:

SelectionProbability Equal to n / N for all units.

SamplingWeight Equal to N / n for all units.

SelectionIndicator TRUE if selected, FALSE otherwise.-only included if `outall=TRUE`

NumberHits 1 if selected, 0 otherwise.

ExpectedHits Equal to `SelectionProbability`.

References

Kalton, G. (1983). *Introduction to Survey Sampling*. SAGE Publications. <https://doi.org/10.4135/9781412984683>

Examples

```
# Sort by REGION, DIVISION, and Pop_Tot, then take a sample
puma_2023 |>
  tidytable::arrange(Region, Division, Pop_Tot) |>
  sys(n = 50, outall = FALSE)

# Return full dataset with selection indicators
puma_2023 |>
  tidytable::arrange(Region, Division, Pop_Tot) |>
  sys(n = 50, outall = TRUE)
```

sys_pps

Select a systematic PPS sample

Description

Draws a systematic sample of size n . Each unit's probability of selection is proportional to its size measure.

Usage

```
sys_pps(frame, n, mos, outall = FALSE, curstrat = NULL)
```

Arguments

frame	The input data frame for the function to work on.
n	The sample size, the parameter expects an integer of length 1. The function will check if n is less than or equal to the number of rows in the input frame.
mos	The measure of size, the parameter expects a character string to indicate the variable to be use as the measure of size. The variable must exists on the frame and be non-missing and non-negative numeric variable.
outall	Output all records or selected records. If outall is TRUE, then all records are return and the following variables are created: SelectionIndicator, SamplingWeight, NumberHits, and ExpectedHits. If outall is FALSE, then the selected records are return and the following variables are created: SamplingWeight, NumberHits, ExpectedHits.
curstrat	A character variable that specifies the current strata, only used as an assertion for the $n == N$ test.

Value

Returns an object of type tidytable that contains the weight, selection probability, number of hits, etc plus all original variables.

Examples

```
# PPS sample of 75 counties using Pop_Tot as the measure of size
# Return only the sampled counties
sys_pps(county_2023, mos = "Pop_Tot", n = 75, outall = FALSE)

# Return the full dataset with selection indicators
sys_pps(county_2023, mos = "Pop_Tot", n = 75, outall = TRUE)
```

Index

* **datasets**

county_2023, [7](#)

ipeds, [8](#)

puma_2023, [9](#)

allocate, [2](#)

chromy_pps, [6](#)

county_2023, [7](#)

ipeds, [8](#)

puma_2023, [9](#)

select_sample, [10](#)

seq_pps (chromy_pps), [6](#)

serp_sort, [13](#)

srs, [14](#)

sys, [15](#)

sys_pps, [16](#)