

Package: censuspvrID (via r-universe)

May 18, 2026

Type Package

Title Explorer of Indonesian Population Pyramids from Harmonized and Non-Harmonized Census Data

Version 1.0.2

Date 2025-10-09

Description Provides harmonized and non-harmonized population pyramid datasets from the Indonesian population censuses (1971–2020), along with tools for visualization and an interactive 'shiny'-based explorer application. Data are processed from IPUMS International (1971–2010) and the Population Census 2020 (BPS Indonesia).

License GPL-3

Depends R (>= 4.1)

Imports shiny (>= 1.11.1), shinythemes, shinyWidgets, shinyjs, dplyr, tidyr, DT, ggplot2, ggthemes, scales, networkD3

Suggests tibble

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

URL <https://github.com/aripurwantosp/censuspvrID>

BugReports <https://github.com/aripurwantosp/censuspvrID/issues>

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libuv1-dev libxml2-dev zlib1g-dev

Repository <https://aripurwantosp.r-universe.dev>

Date/Publication 2025-10-20 08:47:28 UTC

RemoteUrl <https://github.com/aripurwantosp/censuspvrID>

RemoteRef HEAD

RemoteSha 2e941ae53192141f81461adb4c8f1743018e16c6

Contents

ageprof	2
area_trends	3
censuspyrID_explorer	4
data_for_table	5
get_code_label	6
is_expanded	7
load_pop_data	8
pop_data_by_reg	9
pop_data_by_year	10
pop_summary	11
pop5	12
pyr_single	13
pyr_trends	14
year_range	15
Index	17

ageprof	<i>Build Age-Profile Plot by Sex</i>
---------	--------------------------------------

Description

Create a line plot of population age profiles (5-year age groups) for a given province and year, with optional logarithmic scale. The plot is faceted by sex.

Usage

```
ageprof(data, log_scale = FALSE, color = "Fresh and bright")
```

Arguments

data	A data frame of population data for a specific province and year, containing at least the variables: pop (population count), sex (coded as 1 = male, 2 = female), age5 (5-year age groups).
log_scale	Logical; whether to use a logarithmic scale for the Y-axis. Default is FALSE.
color	Character; the name of a Canva color palette available in <code>ggthemes::canva_palettes</code> . Default is "Fresh and bright".

Details

The function produces an age-profile line chart where:

- X-axis: Age (5-year groups).
- Y-axis: Population counts (in thousands by default).
- Separate lines are drawn for males and females.
- Users can choose logarithmic scaling of the Y-axis.

Value

A ggplot2 object representing the age-profile plot, faceted by sex.

See Also

[pyr_single\(\)](#), [load_pop_data\(\)](#), [pop_data_by_reg\(\)](#), [pop_data_by_year\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:
# Example: age profile for Indonesia, 2020
data_idn <- pop_data_by_year(load_pop_data(), 2020) |>
  pop_data_by_reg(0) # Indonesia
ageprof(data_idn)

# Example with log scale
ageprof(data_idn, log_scale = TRUE)

## End(Not run)
```

area_trends

Plot Population Area Trends by Age Group and Sex

Description

This function builds an area plot showing the proportion of population distributed across three broad age groups (young, working-age, old) over census years. The plot can be displayed separately by sex or combined.

Usage

```
area_trends(data, sex = 1, color = "Fresh and bright")
```

Arguments

data	A data frame containing population trends data for a specific region over years. Must include variables year, sex, age5, and pop.
sex	Integer indicating which sex to include in the plot: <ul style="list-style-type: none"> • 1 = All sexes • 2 = Male • 3 = Female • 4 = Male+Female Default is 1 (all sexes).
color	Character string specifying the palette name from <code>ggthemes::canva_palettes</code> . Default is "Fresh and bright".

Details

The function aggregates population into three age groups:

- 0–14 years (Young)
- 15–64 years (Working age)
- 65+ years (Old)

It then calculates the proportion of each age group within each sex and year. The result is plotted as a stacked area chart, optionally faceted by sex.

Value

A ggplot2 object showing the population area trends.

See Also

[pyr_trends\(\)](#), [load_pop_data\(\)](#), [pop_data_by_reg\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:
# Example: area trends for Indonesia
data_idn <- load_pop_data(harmonized = TRUE, smoothing = 1) |>
  pop_data_by_reg(0) #Indonesia
area_trends(data_idn, sex = 1) #All sexes
area_trends(data_idn, sex = 2) #Male
area_trends(data_idn, sex = 3) #Female
area_trends(data_idn, sex = 4) #Male+Female

## End(Not run)
```

censuspyrID_explorer *Explore Harmonized and Non-Harmonized Population Pyramids from Indonesia's Censuses (1971–2020)*

Description

Launches **censuspyrID Explorer**, a Shiny application for visualizing harmonized and non-harmonized population pyramids from Indonesia's population censuses (1971–2020).

Usage

```
censuspyrID_explorer(host = NULL, ...)
```

Arguments

host Character string passed to [runApp](#). Default is "0.0.0.0".
... Additional arguments passed to [runApp](#).

Details

The application provides interactive tools to explore demographic structures across provinces and census years. See the *Help* menu within the application for a navigation guide.

Value

The function launches the Shiny application. It does not return a value.

Examples

```
## Not run:  
censuspvrID_explorer()  
  
## End(Not run)
```

data_for_table	<i>Prepare Population Data for Tabular Display</i>
----------------	--

Description

Prepares population data for tabular display (e.g., in reports or Shiny apps). The function reshapes the data by sex, adds total population, and computes the sex ratio, while also attaching province names and labels.

Usage

```
data_for_table(data, reg_code, harmonized = TRUE)
```

Arguments

data	A data frame containing population data for a specific province and year. Must include columns: year, province_id, sex, age5, and pop.
reg_code	Integer or character. Province code used to retrieve the province name.
harmonized	Logical. If TRUE (default), province codes are treated as harmonized; if FALSE, non-harmonized codes are used.

Details

The function performs the following steps:

- Adds the province name using `reg_name()`.
- Relabels sex and age5 using reference tables in `ref_label`.
- Reshapes data into wide format with separate columns for Male and Female.
- Adds a Male+Female total population column.
- Computes the sex ratio (Male/Female * 100).

Value

A data frame in wide format with columns:

- province_id — province identifier
- province — province name
- year — census year
- age5 — five-year age group label
- Male — male population
- Female — female population
- Male+Female — total population
- sex_ratio — ratio of males to females (per 100 females)

See Also

[load_pop_data\(\)](#), [pop_data_by_year\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:
data_idn <- pop_data_by_year(load_pop_data(), 2020) |>
  pop_data_by_reg(0) #Indonesia
tab <- data_for_table(data_idn, reg_code = 0, harmonized = TRUE)
head(tab)

## End(Not run)
```

get_code_label

Retrieve Reference Codes and Labels

Description

This function returns reference tables for codes and labels used in the package. It can provide mappings for census years, sex, age groups, and province codes (harmonized or non-harmonized).

Usage

```
get_code_label(what = 4)
```

Arguments

what	Integer indicating which reference table to return: <ul style="list-style-type: none">• 1 = Census year and label• 2 = Sex code and label• 3 = Age (5-year group) code and label• 4 = Harmonized province code and label• 5 = Non-harmonized province code and label
------	--

Details

The function retrieves data from internal reference object `re_label`, which stores standardized coding schemes and their associated labels.

Value

A data frame (or tibble) containing codes and labels for the selected reference category.

Examples

```
# Get harmonized province codes and labels
get_code_label(4)

# Get sex codes and labels
get_code_label(2)
```

is_expanded	<i>Check Province Expansion Status</i>
-------------	--

Description

This function checks whether a given province code corresponds to a province that has been expanded (i.e., administratively split or modified).

Usage

```
is_expanded(reg_code)
```

Arguments

`reg_code` non-harmonized province code (character or numeric).

Details

The function looks up the internal dataset `prov_coverage`. Expansion status is determined by the field `expanded`.

Value

A logical value:

- TRUE if the province is marked as expanded,
- FALSE otherwise.

See Also

[get_code_label\(\)](#)

Examples

```
# Example: check expansion status of a province
get_code_label(5) #returns list of non-harmonized province code
is_expanded(1400) # returns TRUE/FALSE for Riau province
```

load_pop_data	<i>Load Population Data</i>
---------------	-----------------------------

Description

Load census population data with options for harmonization and smoothing. Returns population counts by year, province, sex, and five-year age group, with raw or smoothed estimates depending on the selected method.

Usage

```
load_pop_data(harmonized = TRUE, smoothing = 1)
```

Arguments

harmonized	Logical. If TRUE (default), load harmonized data (hpop5). If FALSE, load non-harmonized data (ypop5).
smoothing	Integer. Smoothing method applied to population counts: <ul style="list-style-type: none"> • 1: none (raw) • 2: Arriaga • 3: Karup–King–Newton (KKN)

Details

Data are retrieved from internal census datasets:

- hpop5: harmonized census data
- ypop5: non-harmonized census data

Smoothing methods are applied to the population counts:

- 1: none (raw, default)
- 2: Arriaga method
- 3: Karup–King–Newton (KKN) method

Value

A tibble with columns:

- year: census year
- province_id: province identifier (harmonized or non-harmonized)
- sex: sex code
- age5: five-year age group code
- pop: population count (raw or smoothed)

See Also

[pop_data_by_year\(\)](#), [pop_data_by_reg\(\)](#), [pop5](#)

Examples

```
## Not run:  
# Load harmonized, raw (unsmoothed) population data  
load_pop_data(harmonized = TRUE, smoothing = 1)  
  
# Load non-harmonized, Arriaga-smoothed population data  
load_pop_data(harmonized = FALSE, smoothing = 2)  
  
## End(Not run)
```

pop_data_by_reg

Filter Population Data by Province

Description

Filter population data based on a specified province ID. This function is intended for use with population datasets loaded via [load_pop_data\(\)](#), but can work with any data frame that includes a province_id column.

Usage

```
pop_data_by_reg(data, reg)
```

Arguments

data	A data frame or tibble containing population data. Must include a column named province_id.
reg	Integer or character. The province ID to filter by.

Value

A tibble (or data frame) containing only rows for the specified province.

See Also

[load_pop_data\(\)](#), [pop_data_by_year\(\)](#)

Examples

```
# Load harmonized data
dat <- load_pop_data(harmonized = TRUE, smoothing = 1)

# Filter data for province ID 0 (Indonesia)
pop_data_by_reg(dat, reg = 0)
```

pop_data_by_year *Filter Population Data by Year*

Description

Filter population data for a specific census year. This function is intended for use with population datasets loaded via [load_pop_data\(\)](#), but can work with any data frame that contains a year column.

Usage

```
pop_data_by_year(data, yr)
```

Arguments

data	A data frame or tibble containing population data. Must include a column named year.
yr	Integer or numeric. The census year to filter by.

Value

A tibble (or data frame) containing only rows from the specified year.

See Also

[load_pop_data\(\)](#), [pop_data_by_reg\(\)](#)

Examples

```
# Load harmonized data first
dat <- load_pop_data(harmonized = TRUE, smoothing = 1)

# Filter for the 2000 census year
pop_data_by_year(dat, 2000)
```

`pop_summary`*Print Population Summary Statistics*

Description

Generate and print a formatted summary of population counts, percentages, sex ratio, and dependency ratios from a given dataset of population data for a specific province and year.

Usage

```
pop_summary(data)
```

Arguments

`data` A data frame of population data for a specific province and year, containing at least the variables: `pop` (population count), `sex` (coded as 1 = male, 2 = female), `age5` (5-year age groups).

Details

The function calculates:

- Total population
- Male and female population counts and percentages
- Age group distribution: 0–14, 15–64, and 65+ (counts and percentages)
- Sex ratio (males per 100 females)
- Dependency ratios (0–14, 65+, and total dependency ratio relative to 15–64)

Results are printed directly to the console in a formatted table.

Value

This function does not return an object. It prints formatted summary statistics to the console.

See Also

[load_pop_data\(\)](#), [pop_data_by_reg\(\)](#), [pop_data_by_year\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:  
# Example: population summary for Indonesia, 2020  
data_idn <- pop_data_by_year(load_pop_data(), 2020) |>  
  pop_data_by_reg(0) # Indonesia  
pop_summary(data_idn)  
  
## End(Not run)
```

pop5

Population Counts in 5-Year Age Groups from Indonesian Censuses

Description

Population counts in 5-year age groups at the provincial level (subnational level 1), derived from a series of Indonesian population censuses. Data are available in two versions:

- hpop5 — Harmonized province codes across census years.
- ypop5 — Original (non-harmonized) province codes as reported in each census.

Both datasets are processed from census samples provided by IPUMS International (1971–2010) and the Population Census 2020. Data processing steps include prorating to allocate missing attributes and smoothing using multiple demographic methods (Arriaga and Karup–King–Newton).

Format

Each dataset is a tibble (data frame) with the following variables:

year Census year.

province_id_h Harmonized province identifier (in hpop5).

province_id_y non-harmonized province identifier (in ypop5).

sex Sex code.

age5 Age group in 5-year intervals.

ns Unsmoothed population count.

arriaga Population count smoothed with the Arriaga method.

kkn Population count smoothed with the Karup–King–Newton method.

- hpop5: 5,500 observations.
- ypop5: 6,146 observations.

Source

Ruggles, S., Cleveland, L., Lovaton, R., Sarkar, S., Sobek, M., Burk, D., Ehrlich, D., Heimann, Q., Lee, J., & Merrill, N. (2025). *Integrated Public Use Microdata Series, International: Version 7.7* (dataset). Minneapolis, MN: IPUMS. doi:10.18128/D020.V7.7

Badan Pusat Statistik (BPS). (2020). *Jumlah Penduduk Menurut Wilayah, Kelompok Umur, dan Jenis Kelamin, di INDONESIA – Sensus Penduduk 2020*. Retrieved September 4, 2025, from <https://sensus.bps.go.id/topik/tabular/sp2020/3>

References

- Ruggles, S., Cleveland, L., Lovaton, R., Sarkar, S., Sobek, M., Burk, D., Ehrlich, D., Heimann, Q., Lee, J., & Merrill, N. (2025). *Integrated Public Use Microdata Series, International: Version 7.7* (dataset). Minneapolis, MN: IPUMS. doi:10.18128/D020.V7.7
- Badan Pusat Statistik (BPS). (2020). *Jumlah Penduduk Menurut Wilayah, Kelompok Umur, dan Jenis Kelamin, di INDONESIA – Sensus Penduduk 2020*. Retrieved September 4, 2025, from <https://sensus.bps.go.id/topik/tabular/sp2020/3>
- Siegel, J. S., Swanson, D. A., & Shryock, H. S. (Eds.). (2004). *The methods and materials of demography* (2nd ed). Elsevier/Academic Press.
- Aburto, J. M., Kashnitsky, I., Pascariu, M., & Riffe, T. (2022). *Smoothing with DemoTools*. Available at: https://timriffe.github.io/DemoTools/articles/smoothing_with_demotools.html#references-1

Examples

```
library(dplyr)

# Harmonized data
data(hpop5)
glimpse(hpop5)
head(hpop5)

# Non-harmonized data
data(ypop5)
glimpse(ypop5)
head(ypop5)
```

pyr_single

Build a Single Population Pyramid

Description

Create a population pyramid for a given dataset (specific province and year), either in absolute counts or in proportions, with customizable color palettes.

Usage

```
pyr_single(data, use_prop = FALSE, color = "Fresh and bright")
```

Arguments

- | | |
|----------|---|
| data | A data frame containing population data for a specific province and year. Must include variables sex, age5, and pop. |
| use_prop | Logical, default FALSE. If TRUE, the pyramid will be shown in proportions instead of absolute counts. |
| color | Character string indicating the color palette name to use for the pyramid. Available palettes come from gthemes::canva_palettes , e.g., "Fresh and bright". |

Value

A ggplot object representing the population pyramid.

See Also

[ageprof\(\)](#), [pyr_trends\(\)](#), [load_pop_data\(\)](#), [pop_data_by_reg\(\)](#), [pop_data_by_year\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:
# Example data for Indonesia, 2020
data_idn <- pop_data_by_year(load_pop_data(), 2020) |>
  pop_data_by_reg(0) #Indonesia

# Absolute count pyramid
pyr_single(data_idn)

# Proportional pyramid with different palette
pyr_single(data_idn, use_prop = TRUE, color = "Professional and modern")

## End(Not run)
```

pyr_trends

Build Population Pyramid Trends

Description

Create trend plots of population pyramids over multiple census years for a given region. Users can choose between a grid layout of pyramids or an overlay of age profiles across years.

Usage

```
pyr_trends(data, mode = 1, use_prop = FALSE, color = "Fresh and bright")
```

Arguments

data	A data frame of population data for a specific region across census years, containing at least: year, age5, sex, and pop.
mode	Integer; visualization mode: 1 for grid pyramids, 2 for overlaid age profiles. Default is 1.
use_prop	Logical; whether to show proportions instead of absolute counts. Default is FALSE.
color	Character; the name of a Canva color palette available in <code>ggthemes::canva_palettes</code> . Default is "Fresh and bright".

Details

Two visualization modes are available:

- mode = 1: Grid of population pyramids (faceted by year).
- mode = 2: Overlaid age profiles with separate lines by year.

Population counts can be displayed either as absolute numbers (default, in thousands) or as proportions (use_prop = TRUE).

Value

A ggplot2 object representing the population pyramid trend plot.

See Also

[area_trends\(\)](#), [load_pop_data\(\)](#), [pop_data_by_reg\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:  
# Example: pyramid trends for Indonesia  
data_idn <- load_pop_data(harmonized = TRUE, smoothing = 1) |>  
  pop_data_by_reg(0) #Indonesia  
pyr_trends(data_idn, mode = 1) # grid layout  
  
# Overlay mode with proportions  
pyr_trends(data_idn, mode = 2, use_prop = TRUE)  
  
## End(Not run)
```

year_range

Get Census Year Coverage for a Province

Description

This function determines the range of census years available for a given province. Coverage depends on whether harmonized or non-harmonized codes are used, and in the case of non-harmonized data, whether the province has experienced administrative expansion (pemekaran).

Usage

```
year_range(reg_code = NULL, harmonized = TRUE, before_expand = TRUE)
```

Arguments

reg_code	Character or numeric. Province code. Required if harmonized = FALSE.
harmonized	Logical. If TRUE (default), returns harmonized coverage (1971–2020). If FALSE, uses non-harmonized coverage.
before_expand	Logical. Only relevant if harmonized = FALSE and the province has expanded. If TRUE (default), returns coverage before expansion; if FALSE, returns coverage after expansion.

Details

- For harmonized data (harmonized = TRUE), the full coverage of 1971–2020 is returned.
- For non-harmonized data (harmonized = FALSE), coverage is determined based on the internal dataset prov_coverage.
- If the province has expanded, coverage depends on before_expand.
- Census year labels are retrieved from ref_label\$census_label.

Value

An integer vector of census years, with labels as names.

See Also

[is_expanded\(\)](#), [get_code_label\(\)](#)

Examples

```
## Not run:  
# Harmonized coverage (1971–2020)  
year_range(harmonized = TRUE)  
  
# non-harmonized coverage for a province (before expansion)  
get_code_label(5) #returns list of non-harmonized province code  
year_range(reg_code = 1400, harmonized = FALSE, before_expand = TRUE)  
  
# non-harmonized coverage for a province (after expansion)  
year_range(reg_code = 1400, harmonized = FALSE, before_expand = FALSE)  
  
## End(Not run)
```

Index

* datasets

pop5, [12](#)

ageprof, [2](#)

ageprof(), [14](#)

area_trends, [3](#)

area_trends(), [15](#)

censuspyrID_explorer, [4](#)

data_for_table, [5](#)

get_code_label, [6](#)

get_code_label(), [3](#), [4](#), [6](#), [7](#), [11](#), [14–16](#)

ggthemes::canva_palettes, [13](#)

hpop5 (pop5), [12](#)

is_expanded, [7](#)

is_expanded(), [16](#)

load_pop_data, [8](#)

load_pop_data(), [3](#), [4](#), [6](#), [9–11](#), [14](#), [15](#)

pop5, [9](#), [12](#)

pop_data_by_reg, [9](#)

pop_data_by_reg(), [3](#), [4](#), [9–11](#), [14](#), [15](#)

pop_data_by_year, [10](#)

pop_data_by_year(), [3](#), [6](#), [9–11](#), [14](#)

pop_summary, [11](#)

pyr_single, [13](#)

pyr_single(), [3](#)

pyr_trends, [14](#)

pyr_trends(), [4](#), [14](#)

reg_name(), [5](#)

runApp, [4](#)

year_range, [15](#)

ypop5 (pop5), [12](#)