Package: pobecu (via r-universe)

November 8, 2026

Type Package

Title Population of Ecuador by Province, Canton and Parish (Población del Ecuador por Provincia, Cantón y Parroquia)

Version 0.3.0

Description Provides the official population estimates and projections of Ecuador published by the National Institute of Statistics and Censuses (INEC, ``Instituto Nacional de Estadística y Censos''), Revision 2024: national (1950-2050), provincial (1990-2035) and cantonal (2010-2035) figures by sex, age and urban/rural area, identified by the codes of the political-administrative division (DPA, ``División Político-Administrativa'') <https://www.ecuadorencifras.gob.ec/proyecciones-poblacionales/>. Because INEC does not publish projections for parishes, the package also offers a parish-level estimate for 2010-2035 that distributes the official cantonal projection among parishes using their shares in the 2010 and 2022 censuses (logistic extrapolation) and their age and sex structure in the 2022 census (iterative proportional fitting); parish figures always add up to the official cantonal ones. Includes helpers to aggregate ages, join population denominators to case tables and compute rates, convert codes of urban parishes and of the 2026 political-administrative division (CONALI, ``Comité Nacional de Límites Internos'') to census units, and native integration with the 'geoecu' package to map population on the boundaries of provinces, cantons and parishes. Documentation is in Spanish. Includes conditional territorial population scenarios for 1900-2100 using structural time series, damped extrapolation and hierarchical compositional allocation with Monte Carlo bands.

License MIT + file LICENSE

URL https://f-slg.github.io/pobecu/, https://github.com/f-slg/pobecu

BugReports https://github.com/f-slg/pobecu/issues

Depends R (>= 4.1.0)

Imports stats, tools, utils

Suggests geoecu (>= 0.4.1), ggplot2, knitr, rmarkdown, scales, sf, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language es

LazyData true

LazyDataCompression xz

Config/roxygen2/version 8.1.1

NeedsCompilation no

Author Ronald Fabricio Salas Guerrero [aut, cre, cph]

Maintainer Ronald Fabricio Salas Guerrero <fabriciosalasguerrero@gmail.com>

Repository https://cran.r-universe.dev

Date/Publication 2026-11-07 22:20:08 UTC

RemoteUrl https://github.com/cran/pobecu

RemoteRef HEAD

RemoteSha e20425ef155a06ae89eac6669c91f2db624c3393

Contents

agrupar_unidades . . . 3
buscar_pob . . . 4
dpa_pob . . . 4
formatear_codigo . . . 5
homologar_canton . . . 6
homologar_parroquia . . . 7
normalizar_nombre . . . 8
pob_ajustar . . . 9
pob_estimar . . . 10
pob_fuentes . . . 11
pob_inec . . . 11
pob_mapa . . . 13
pob_parroquias . . . 14
pob_tasa . . . 16
pob_unir . . . 17
pobecu_cache_dir . . . 18
pobecu_datos_disponibles . . . 18
pobecu_limpiar_cache . . . 19
pobecu_usar_carpeta . . . 19
poblacion . . . 20

Index **22**

agrupar_unidades *Convertir una capa de geoecu a las unidades del Censo 2022*

Description

Fusiona los polígonos de una capa de cantones o parroquias (por ejemplo, `geoecu::read_cantones()` o `geoecu::read_parroquias()`) en las unidades del Censo 2022 que usa pobecu, para unirla con la población sin perder ni repetir territorio. Los códigos se convierten con `homologar_canton()` o `homologar_parroquia()` y los polígonos con el mismo código se unen:

Usage

```
agrupar_unidades(capa)

agrupar_cabeceras(capa)
```

Arguments

capa	Un objeto `sf` con una columna `dpa_parroq` (parroquias) o `dpa_canton` (cantones). Si existen, se usan también `tipo` y `area_km2`.

Details

- Con geoecu 0.4.0 y posteriores (CONALI 2026): los cantones Borbón y Sevilla Don Bosco vuelven a Eloy Alfaro y Morona, y las parroquias creadas después del censo (Sosote, Shuar Pastaza, Juan Montalvo, La Magdalena, La Primavera y Miss Ecuador) se unen a la parroquia censal que las contenía.
- Con geoecu 0.3.0 y anteriores: las parroquias urbanas de cada cantón se unen en su cabecera cantonal (por ejemplo, las 15 de Cuenca).

Las unidades que no existen en pobecu, como las zonas en estudio, se conservan con sus datos originales. `agrupar_cabeceras()` es el nombre anterior de esta función y se mantiene por compatibilidad.

Value

Un objeto `sf` con una fila por unidad del censo: el código, los nombres de pobecu (de la parroquia, el cantón y la provincia, según el nivel), `tipo` (`"cabecera cantonal"` o `"parroquia rural"` en el nivel parroquial; el de la capa en el cantonal), `area_km2` (suma) y la geometría.

Examples

```
esmeraldas <- geoecu::read_cantones(provincia = "Esmeraldas")
nrow(esmeraldas)                    # con Borbón
nrow(agrupar_unidades(esmeraldas))  # Borbón dentro de Eloy Alfaro
```

buscar_pob *Buscar en el catálogo de pobecu*

Description

Busca por nombre (sin importar tildes ni mayúsculas) o por código en el catálogo dpa_pob.

Usage

```
buscar_pob(texto, nivel = c("parroquia", "canton", "provincia"))
```

Arguments

texto	Texto a buscar: parte de un nombre o un código.
nivel	Nivel en el que buscar: "parroquia", "canton" o "provincia".

Value

Un data.frame con las coincidencias en el nivel indicado.

Examples

```
buscar_pob("arenal")
buscar_pob("puyango", nivel = "canton")
buscar_pob("11", nivel = "provincia")
```

dpa_pob *Catálogo de parroquias, cantones y provincias de pobecu*

Description

Una fila por parroquia del Censo 2022, con sus códigos DPA, nombres oficiales del INEC, la población censal 2010 y 2022 y el método usado en la estimación parroquial.

Usage

```
dpa_pob
```

Format

Un `data.frame` con 1042 filas y 10 columnas:

dpa_parroq Código DPA de la parroquia (6 dígitos, texto). Las cabeceras cantonales terminan en `50`.

parroquia Nombre de la parroquia según el Censo 2022.

tipo `"cabecera cantonal"` (incluye las parroquias urbanas) o `"parroquia rural"`, con los mismos valores que geoecu.

dpa_canton, canton Código y nombre del cantón.

dpa_provin, provincia Código y nombre de la provincia.

metodo Método de la estimación parroquial: `"logistica"`, `"participacion_constante"` (cantones con parroquias creadas después de 2010) o `"parroquia_unica"` (cantones de una sola parroquia).

censo_2010, censo_2022 Población empadronada en los censos 2010 (en los límites de 2022; `NA` para parroquias creadas después de 2010) y 2022.

Details

Los cantones y parroquias creados después del censo (por ejemplo, el cantón Borbón o la parroquia Sosote) no tienen fila propia: su población está incluida en la unidad censal que los contenía. Ver `homologar_canton()` y `homologar_parroquia()`.

Source

INEC, VIII Censo de Población y VII de Vivienda 2022, tablas 2.1 y 5.2 de *Estructura poblacional*. Los códigos de las parroquias rurales se asignaron por nombre a partir de la División Político-Administrativa oficial.

Examples

```
head(dpa_pob)
table(dpa_pob$provincia, dpa_pob$tipo)
```

`formatear_codigo` *Formatear códigos DPA*

Description

Devuelve códigos DPA como texto con el número correcto de dígitos, recuperando los ceros a la izquierda que se pierden cuando un código se guarda como número en Excel (por ejemplo, `701` en lugar de `"0701"`).

Usage

```
formatear_codigo(x, nivel = c("parroquia", "canton", "provincia"))
```

Arguments

x	Vector de códigos (número o texto).
nivel	"provincia" (2 dígitos), "canton" (4) o "parroquia" (6).

Value

Vector de texto. Los valores que no son códigos válidos se devuelven como NA con una advertencia.

Examples

```
formatear_codigo(c(1, 11, "7"), "provincia")
formatear_codigo(c(701, 1101), "canton")
formatear_codigo(110150, "parroquia")
```

---

homologar_canton *Homologar códigos de cantón a las unidades del Censo 2022*

Description

Las proyecciones cantonales del INEC usan los 221 cantones del Censo 2022. La Organización Territorial del Estado del CONALI (2026), que usa geoecu desde la versión 0.4.0, tiene 223: incluye Borbón (0809), creado a partir de Eloy Alfaro (0802), y Sevilla Don Bosco (1413), creado a partir de Morona (1401). Esta función convierte esos códigos al del cantón censal que los contiene, para agregar los datos y unirlos con la población. Los demás códigos se devuelven sin cambios.

Usage

```
homologar_canton(codigo)
```

Arguments

codigo	Vector de códigos de cantón (número o texto).

Value

Vector de texto con códigos de 4 dígitos.

See Also

homologar_parroquia() para los códigos de parroquia.

Examples

```
homologar_canton(c("0809", "1413", 1101))

# Agregar casos de Borbón y Eloy Alfaro antes de calcular la tasa
casos <- data.frame(dpa_canton = c("0802", "0809"), anio = 2025,
                    casos = c(7, 5))
casos$dpa_canton <- homologar_canton(casos$dpa_canton)
casos <- aggregate(casos ~ dpa_canton + anio, casos, sum)
pob_tasa(casos, "casos")
```

---

homologar_parroquia *Homologar códigos parroquiales a las unidades del Censo 2022*

Description

El Censo 2022 y las estimaciones parroquiales de pobecu usan las parroquias vigentes en el censo: cada **cabecera cantonal** es una sola unidad (código terminado en `50`), sin separar las parroquias urbanas. Otras fuentes usan códigos que no existen en el censo:

Usage

```
homologar_parroquia(codigo)
```

Arguments

`codigo`	Vector de códigos de parroquia (número o texto).

Details

- **Parroquias urbanas** (sufijos `01` a `49`, como las 15 de Cuenca), que aparecen en muchos registros administrativos y en geoecu 0.3.0 y anteriores. Se convierten al código de su cabecera cantonal.
- **Códigos de la Organización Territorial del Estado del CONALI (2026)**, que usa geoecu desde la versión 0.4.0: las parroquias de los cantones creados después del censo (Borbón, antes en Eloy Alfaro, y Sevilla Don Bosco, antes en Morona) y Bella Vista (Santa Cruz) recuperan su código del censo, y las parroquias creadas después del censo (Sosote, Shuar Pastaza, Juan Montalvo, La Magdalena, La Primavera y Miss Ecuador) se convierten al de la parroquia censal que las contenía.

Así se pueden agregar los datos y unirlos con la población. Los demás códigos se devuelven sin cambios.

Casos especiales:

- `220254` (Tiputini, Aguarico): no figura como parroquia rural en el Censo 2022; su población está incluida en la cabecera cantonal `220250`.
- La Magdalena (`210456`) se formó con territorio de Shushufindi, Limoncocha y San Roque; se asigna a Shushufindi (`210450`), de donde salió la mayor parte.
- Las zonas en estudio de geoecu (`139090`, `140190`, `900651`) no tienen población en pobecu y se devuelven sin cambios.

Value

Vector de texto con códigos de 6 dígitos.

See Also

`homologar_canton()` para los códigos de cantón.

Examples

```
# Parroquias urbanas de Cuenca -> cabecera cantonal de Cuenca
homologar_parroquia(c("010101", "010115", "010150", "010151"))

# Códigos de geoecu >= 0.4.0 (CONALI 2026): cabecera de Borbón y Sosote
homologar_parroquia(c("080950", "131251"))

# Agregar casos por parroquia urbana a la unidad del censo
casos <- data.frame(dpa_parroq = c("010101", "010102", "010151"),
                    casos = c(3, 5, 1))
casos$dpa_parroq <- homologar_parroquia(casos$dpa_parroq)
aggregate(casos ~ dpa_parroq, casos, sum)
```

---

normalizar_nombre *Normalizar nombres geográficos*

Description

Convierte nombres a una forma comparable: mayúsculas, sin tildes ni diéresis, Ñ como N, sin signos de puntuación y con espacios simples. Útil para unir tablas que escriben los nombres de forma distinta ("Cañar", "CANAR", "cañar ").

Usage

```
normalizar_nombre(x)
```

Arguments

x	Vector de texto.

Value

Vector de texto normalizado.

Examples

```
normalizar_nombre(c("Cañar", "  san  José de Chimbo", "CHECA (JIDCAY)"))
```

pob_ajustar *Ajustar un modelo territorial de población para 1900-2100*

Description

Ajusta tendencias locales lineales mediante máxima verosimilitud y filtro de Kalman: logaritmo del total nacional y coordenadas log-ratio centradas de las participaciones de cada territorio en su padre. La extrapolación amortigua la pendiente y la distribución descendente conserva los totales.

Usage

```
pob_ajustar(
  entrenamiento = NULL,
  ventana = 20,
  phi_total = 0.98,
  phi_partes = 0.95
)
```

Arguments

entrenamiento	Años de las tablas publicados que se pueden utilizar. NULL usa todos. Deben quedar al menos ocho años consecutivos por nivel.
ventana	Máximo de años utilizados en cada extremo, al menos ocho.
phi_total	Amortiguación anual de la pendiente nacional, en (0,1].
phi_partes	Amortiguación de las participaciones, en (0,1].

Details

Se incluyen proyecciones publicadas en el entrenamiento por defecto. Se condiciona a esas trayectorias; no se ajustan fecundidad, mortalidad ni migración por cohortes. Las unidades son las de referencia del Censo 2022, incluso para años en los que no existían administrativamente. La retrospectiva ajusta el modelo a la serie invertida: no es un suavizador histórico ni evidencia censal del siglo XX. Consulte la metodología del README antes de interpretar horizontes lejanos.

Value

Objeto pob_modelo con datos de referencia, modelos en ambos sentidos temporales, parámetros y tabla diagnosticos de cada ajuste.

Examples

```
modelo <- pob_ajustar()
head(modelo$diagnosticos)
```

pob_estimar *Reconstruir y proyectar población territorial entre 1900 y 2100*

Description

Extiende las trayectorias con un modelo estructural y composicional jerárquico. Conserva exactamente las referencias utilizadas por el modelo.

Usage

```
pob_estimar(
  nivel = c("provincia", "canton", "parroquia", "nacional", "todos"),
  anio = 1900:2100,
  modelo = NULL,
  provincia = NULL,
  canton = NULL,
  simulaciones = 500,
  semilla = 2026,
  nivel_intervalo = 0.95
)
```

Arguments

nivel	Nivel territorial o "todos" para la jerarquía completa.
anio	Años enteros entre 1900 y 2100; por defecto todo el periodo.
modelo	Resultado de pob_ajustar(). NULL ajusta un modelo nuevo.
provincia	Filtro provincial por nombre o código.
canton	Filtro cantonal por nombre o código, solo cantones/parroquias.
simulaciones	Número de trayectorias Monte Carlo, de 100 a 10000.
semilla	Semilla local reproducible; se restaura el estado aleatorio. Cada grupo territorial usa una semilla propia derivada de esta, por lo que filtrar territorios o años no cambia sus bandas.
nivel_intervalo	Cobertura nominal de bandas condicionales (0,1).

Details

Las bandas simulan innovaciones gaussianas con varianzas estimadas, condicionadas a los estados finales y parámetros ajustados. No incluyen incertidumbre de parámetros, censos, límites ni modelos fuente. En años de referencia las bandas son NA. Los resultados extrapolados no son cifras oficiales. No se modifican las tablas consultadas por poblacion(). La amortiguación es una hipótesis de escenario aplicada al extrapolar, no un parámetro estimado por máxima verosimilitud. No hay desagregación por sexo, edad o residencia en este modelo de totales territoriales.

Value

Tabla con códigos, nombres, población entera, origen, dirección, horizonte, referencia temporal y bandas marginales. El atributo `metodologia` registra parámetros y supuestos. Los cuantiles marginales no se suman. La trayectoria central sí suma exactamente a sus padres.

Examples

```
m <- pob_ajustar()
p <- pob_estimar("parroquia", c(1900, 2022, 2100), m, canton = "Loja")
p[c("nombre", "anio", "poblacion", "origen", "horizonte")]
```

---

pob_fuentes *Fuentes y versión de los datos*

Description

Muestra de dónde provienen los datos de pobecu, qué periodo cubre cada nivel y qué cifras son oficiales del INEC y cuáles son estimaciones del paquete.

Usage

```
pob_fuentes()
```

Value

Un `data.frame` de clase `pob_fuentes` con una fila por nivel. Al imprimirlo se muestran también la versión de los datos y las fuentes.

Examples

```
pob_fuentes()
```

---

pob_inec *Población nacional, provincial y cantonal del INEC*

Description

Estimaciones (1950-2022) y proyecciones (2023 en adelante) oficiales de población del Instituto Nacional de Estadística y Censos (INEC), Revisión 2024, por año, área de residencia, sexo al nacer y grupos quinquenales de edad. Las cifras corresponden al 30 de junio de cada año.

Usage

```
pob_nacional

pob_provincias

pob_cantones
```

Format

`data.frame` en formato largo con las columnas:

dpa_provin, dpa_canton Código DPA (factor; solo en `pob_provincias` y `pob_cantones`).

anio Año (entero).

area `"total"`, `"urbana"` o `"rural"` (factor).

sexo `"hombre"` o `"mujer"` (factor).

grupo_edad Grupo quinquenal: `"0-4"`, `"5-9"`, ..., `"95-99"`, `"100+"` (factor ordenado por edad).

poblacion Número de personas (entero).

An object of class `data.frame` with 5334 rows and 5 columns.

An object of class `data.frame` with 72576 rows and 6 columns.

An object of class `data.frame` with 482664 rows and 6 columns.

Details

Para consultarlas de forma cómoda (filtrar, agregar, cambiar los grupos de edad o agregar nombres) usa `poblacion()`.

- Periodos: nacional 1950-2050, provincial 1990-2035, cantonal 2010-2035.
- Área: en 2010-2035 cada combinación aparece dos veces, `"urbana"` y `"rural"`, que suman el total. Fuera de ese periodo (solo nacional y provincial) hay una sola fila con `area = "total"`. Según el INEC, la población por área es una estimación derivada de las proyecciones cantonales.
- Método del INEC: componentes demográficos por cohortes para los niveles nacional y provincial; extrapolación logística de la participación de cada cantón en su provincia para el nivel cantonal.
- Los tres niveles son coherentes entre sí: la suma de los cantones de una provincia es igual a la provincia, y la suma de las provincias es igual al total nacional.

Source

INEC, *Estimaciones y Proyecciones de Población, Revisión 2024* (tabulados con áreas de residencia, 2025), https://www.ecuadorencifras.gob.ec/proyecciones-poblacionales/. Ver `data-raw/` en el repositorio para el procesamiento completo.

Examples

```
head(pob_cantones)
# Población nacional por año
tapply(pob_nacional$poblacion, pob_nacional$anio, sum)[c("1950", "2025", "2050")]
```

---

pob_mapa *Mapa con población: capas de geoecu unidas a pobecu*

Description

Devuelve la capa geográfica del nivel indicado (del paquete geoecu) con la población de pobecu ya unida, lista para mapear. Ambos paquetes usan los códigos DPA, así que no hace falta preparar nada.

Usage

```
pob_mapa(
  nivel = c("canton", "provincia", "parroquia"),
  anio = NULL,
  provincia = NULL,
  canton = NULL,
  sexo = FALSE,
  area = FALSE,
  edad = NULL,
  resolucion = c("baja", "alta"),
  crs = 4326
)
```

Arguments

nivel	"canton" (por defecto), "provincia" o "parroquia".
anio	Año o años. Por defecto, el año en curso si está disponible; si no, el último año disponible.
provincia	Provincias a incluir, por nombre (sin importar tildes ni mayúsculas) o código: "Loja", "11" o 11. NULL incluye todo el país.
canton	Cantones a incluir (niveles "canton" y "parroquia"), por nombre o código. Si un nombre se repite en varias provincias (por ejemplo, Bolívar u Olmedo), usa el código o indica también provincia.
sexo	Si es TRUE, desagrega por sexo al nacer ("hombre", "mujer").
area	Si es TRUE, desagrega por área de residencia ("urbana", "rural"). Solo para los niveles nacional, provincial y cantonal, y para los años 2010-2035.
edad	Desagregación por edad: • NULL (por defecto): sin desagregar. • "quinquenal": grupos de 5 años ("0-4", ..., "100+"; en el nivel parroquial el último grupo es "85+").

- `"simple"`: años cumplidos, de 0 a 100 (100 = 100 y más).
- `"etapas"`: etapas de vida del INEC (0-11, 12-17, 18-29, 30-64, 65+).
- Un vector numérico con las edades de inicio de cada grupo, por ejemplo `c(0, 1, 5, 15, 65)` para 0, 1-4, 5-14, 15-64 y 65+.

Los grupos quinquenales (y los cortes múltiplos de 5) vienen dentro del paquete. Las edades simples, las etapas y los demás cortes usan datos que se descargan la primera vez (ver `pobecu_datos_disponibles()`) y no están disponibles para parroquias.

`resolucion`, `crs` Resolución y sistema de coordenadas de la capa, como en `geoecu::read_cantones()`.

Details

Las cifras de población usan las unidades del Censo 2022, y geoecu (desde la versión 0.4.0) la Organización Territorial del Estado del CONALI (2026). Para que cada cifra se dibuje sobre el territorio al que corresponde, la capa se convierte primero a las unidades del censo con `agrupar_unidades()`: por ejemplo, el cantón Borbón se dibuja dentro de Eloy Alfaro y la parroquia Sosote dentro de Rocafuerte. Con geoecu 0.3.0 y anteriores, las parroquias urbanas se fusionan en su cabecera cantonal.

Las zonas en estudio de geoecu (territorios con límites aún no resueltos) se conservan para que el mapa no tenga huecos, con población `NA` en cada año y desagregación.

Value

Un objeto `sf` con las columnas de `agrupar_unidades()`, las de `poblacion()` (`anio`, `poblacion` y las desagregaciones pedidas) y `densidad` (habitantes por km²). Con varios años o desagregaciones, cada polígono se repite una vez por combinación (útil para `facet_wrap()`).

See Also

`agrupar_unidades()`, `pob_tasa()`.

Examples

```
m <- pob_mapa("canton", anio = 2025, provincia = "Loja")
plot(m["densidad"], logz = TRUE)

# Parroquias de Cuenca
pob_mapa("parroquia", anio = 2030, canton = "Cuenca")
```

`pob_parroquias` *Población parroquial estimada, 2010-2035*

Description

Estimación de la población de las 1042 parroquias del Censo 2022 (221 cabeceras cantonales y 821 parroquias rurales) por año, sexo al nacer y grupos quinquenales de edad. **No es una cifra oficial**: el INEC no publica proyecciones parroquiales. pobecu la construye a partir de la proyección oficial de cada cantón, de modo que la suma de las parroquias de un cantón es exactamente igual a la cifra oficial del INEC para ese cantón, año, sexo y grupo de edad.

Usage

```
pob_parroquias
```

Format

`data.frame` con las columnas `dpa_parroq` (factor), `anio`, `area` (siempre `"total"`), `sexo`, `grupo_edad` (`"0-4"`, ..., `"80-84"`, `"85+"`) y `poblacion`.

Details

Unidades. Son las parroquias del Censo 2022. Cada cabecera cantonal (código terminado en `50`) es una sola unidad que incluye a todas las parroquias urbanas del cantón. Para unir datos que usan códigos de parroquias urbanas o de la Organización Territorial del Estado del CONALI (2026), como las capas de geoecu 0.4.0 y posteriores, conviértelos con `homologar_parroquia()`.

Método.

1. *Participación en el cantón.* Se calcula la proporción de la población del cantón que vive en cada parroquia en los censos 2010 y 2022, y se interpola y extrapola con una función logística (lineal en la escala logit), el mismo método que usa el INEC para estimar los cantones a partir de su provincia. Las proporciones se reescalan para que sumen 1. En los cantones con parroquias creadas después de 2010 se mantiene constante la proporción del Censo 2022 (ver `metodo` en dpa_pob).
2. *Total parroquial* = proporción x población oficial del cantón.
3. *Sexo y edad.* Ajuste proporcional iterativo (IPF o *raking*): se parte de la estructura por sexo y edad de cada parroquia en el Censo 2022 y se ajusta a dos márgenes, el total parroquial (paso 2) y la estructura oficial del cantón por sexo y edad.
4. *Redondeo* a enteros por el método del mayor residuo, conservando exactamente los totales cantonales.

Precauciones. La extrapolación prolonga la tendencia observada entre los dos censos: parroquias que crecieron o decrecieron mucho entre 2010 y 2022 siguen haciéndolo hasta 2035. Las cifras de 2022 no coinciden con el conteo censal porque la proyección del INEC corrige la omisión censal y se refiere al 30 de junio. El grupo de edad más alto es `"85+"`.

Source

Elaboración propia a partir de INEC, *Estimaciones y Proyecciones de Población, Revisión 2024*, y del VIII Censo de Población y VII de Vivienda 2022 (tabulados de estructura poblacional), https://www.censoecuador.gob.ec/resultados-censo/.

Examples

```
head(pob_parroquias)
```

---

pob_tasa *Calcular tasas con la población de pobecu*

Description

Une la población con `pob_unir()` y calcula una tasa por cada `por` habitantes (por defecto, por 100 000).

Usage

```
pob_tasa(
  datos,
  casos,
  nivel = c("canton", "provincia", "parroquia"),
  por = 1e+05,
  ...
)
```

Arguments

`datos`	Un `data.frame` con una columna de códigos DPA.
`casos`	Nombre de la columna de `datos` con el numerador (casos, defunciones, nacimientos, etc.).
`nivel`	`"canton"` (por defecto), `"provincia"` o `"parroquia"`.
`por`	Base de la tasa: un número finito mayor que cero. Por defecto 100 000 habitantes.
`...`	Otros argumentos para `pob_unir()` (`codigo`, `anio`, `sexo`, `edad`).

Value

`datos` con las columnas `poblacion` y `tasa`. Si la población es cero, la tasa es `NA_real_` y se emite una advertencia. Los casos deben ser numéricos, finitos y no negativos; se permiten valores `NA` y numeradores fraccionarios (por ejemplo, ponderados).

Examples

```
casos <- data.frame(dpa_provin = c("01", "11", "17"), anio = 2024,
                    casos = c(35, 52, 210))
pob_tasa(casos, "casos", nivel = "provincia")
```

pob_unir *Unir la población con una tabla propia*

Description

Agrega a tu tabla (por ejemplo, casos por cantón y año) una columna con la población correspondiente, usando el código DPA y, si existe, el año. Los códigos se limpian con `formatear_codigo()`, así que funcionan aunque vengan de Excel sin ceros a la izquierda.

Usage

```
pob_unir(
  datos,
  nivel = c("canton", "provincia", "parroquia"),
  codigo = NULL,
  anio = NULL,
  sexo = FALSE,
  edad = NULL
)
```

Arguments

datos	Un `data.frame` con una columna de códigos DPA.
nivel	`"canton"` (por defecto), `"provincia"` o `"parroquia"`.
codigo	Nombre de la columna de `datos` con el código DPA. Por defecto se usa `dpa_provin`, `dpa_canton` o `dpa_parroq` según el nivel.
anio	Nombre de la columna de `datos` con el año, o un año fijo (número) para toda la tabla. Por defecto se usa la columna `anio` si existe. Acepta columnas numéricas, de texto o factor con años enteros.
sexo, edad	Desagregación de la población, como en `poblacion()`. Si la usas, `datos` debe tener columnas `sexo` (`"hombre"`/`"mujer"`) y/o `edad` con los mismos valores que devuelve `poblacion()`.

Value

`datos` con una columna adicional `poblacion`. Las filas sin coincidencia quedan con `NA` y se avisa con una advertencia.

Examples

```
casos <- data.frame(dpa_canton = c(1101, 1102, 701), anio = 2024,
                    casos = c(120, 8, 40))
pob_unir(casos)

# Por sexo
casos_sexo <- data.frame(dpa_provin = "11", anio = 2025,
                         sexo = c("hombre", "mujer"), casos = c(10, 14))
pob_unir(casos_sexo, nivel = "provincia", sexo = TRUE)
```

pobecu_cache_dir *Carpeta de caché de pobecu*

Description

Los datos por edad simple se descargan la primera vez que se piden y se guardan aquí para no volver a descargarlos. La ubicación sigue las convenciones de cada sistema operativo (`tools::R_user_dir()`).

Usage

```
pobecu_cache_dir()
```

Value

Ruta de la carpeta de caché (texto).

Examples

```
pobecu_cache_dir()
```

pobecu_datos_disponibles *¿Están disponibles los datos por edad simple sin descargar?*

Description

Los grupos quinquenales vienen dentro del paquete; las edades simples (0 a 100 y más) se descargan la primera vez que se piden con `poblacion(edad = "simple")` u otros cortes de edad que no son múltiplos de 5. Esta función indica si ya están en la carpeta local o en la caché.

Usage

```
pobecu_datos_disponibles()
```

Value

`TRUE` o `FALSE`.

Examples

```
pobecu_datos_disponibles()
```

pobecu_limpiar_cache *Borrar los datos descargados*

Description

Elimina los datos guardados en la caché. Se volverán a descargar la próxima vez que se necesiten.

Usage

```
pobecu_limpiar_cache()
```

Value

TRUE de forma invisible.

Examples

```
pobecu_limpiar_cache()
```

pobecu_usar_carpeta *Usar datos desde una carpeta local*

Description

Indica a pobecu una carpeta que contiene pobecu_edad_simple.rds. Útil para trabajar sin conexión o en servidores institucionales.

Usage

```
pobecu_usar_carpeta(carpeta)
```

Arguments

carpeta	Ruta a la carpeta. Usa NULL para volver al comportamiento por defecto (caché y descarga).

Details

También puedes definir la variable de entorno POBECU_CARPETA (por ejemplo, en tu .Renviron) para que la carpeta se use siempre.

Value

La carpeta anterior, de forma invisible.

Examples

```
anterior <- pobecu_usar_carpeta(tempdir())
pobecu_datos_disponibles()
pobecu_usar_carpeta(anterior)
```

`poblacion` *Población por provincia, cantón o parroquia*

Description

Consulta la población de un nivel territorial y uno o varios años, con la desagregación que necesites (sexo, área urbana/rural y edad). Es la forma más sencilla de obtener denominadores para tasas, coberturas o indicadores.

Usage

```
poblacion(
  nivel = c("canton", "provincia", "parroquia", "nacional"),
  anio = NULL,
  provincia = NULL,
  canton = NULL,
  sexo = FALSE,
  area = FALSE,
  edad = NULL
)
```

Arguments

`nivel`	`"canton"` (por defecto), `"provincia"`, `"parroquia"` o `"nacional"`.
`anio`	Años a consultar. `NULL` (por defecto) devuelve todos los disponibles: 1950-2050 (nacional), 1990-2035 (provincial) y 2010-2035 (cantonal y parroquial). Las cifras se refieren al 30 de junio de cada año.
`provincia`	Provincias a incluir, por nombre (sin importar tildes ni mayúsculas) o código: `"Loja"`, `"11"` o 11. `NULL` incluye todo el país.
`canton`	Cantones a incluir (niveles `"canton"` y `"parroquia"`), por nombre o código. Si un nombre se repite en varias provincias (por ejemplo, Bolívar u Olmedo), usa el código o indica también `provincia`.
`sexo`	Si es `TRUE`, desagrega por sexo al nacer (`"hombre"`, `"mujer"`).
`area`	Si es `TRUE`, desagrega por área de residencia (`"urbana"`, `"rural"`). Solo para los niveles nacional, provincial y cantonal, y para los años 2010-2035.
`edad`	Desagregación por edad: • `NULL` (por defecto): sin desagregar. • `"quinquenal"`: grupos de 5 años (`"0-4"`, ..., `"100+"`; en el nivel parroquial el último grupo es `"85+"`).

- `"simple"`: años cumplidos, de 0 a 100 (100 = 100 y más).
- `"etapas"`: etapas de vida del INEC (0-11, 12-17, 18-29, 30-64, 65+).
- Un vector numérico con las edades de inicio de cada grupo, por ejemplo `c(0, 1, 5, 15, 65)` para 0, 1-4, 5-14, 15-64 y 65+.

Los grupos quinquenales (y los cortes múltiplos de 5) vienen dentro del paquete. Las edades simples, las etapas y los demás cortes usan datos que se descargan la primera vez (ver `pobecu_datos_disponibles()`) y no están disponibles para parroquias.

Details

Las cifras nacionales, provinciales y cantonales son las estimaciones (hasta 2022) y proyecciones (desde 2023) oficiales del INEC, Revisión 2024. Las parroquiales son una estimación de pobecu que reparte la proyección oficial de cada cantón entre sus parroquias (ver pob_parroquias); no son cifras oficiales.

Value

Un `data.frame` con los códigos y nombres del nivel (`dpa_provin`, `provincia`, `dpa_canton`, `canton`, `dpa_parroq`, `parroquia`, según corresponda), `anio`, las columnas de desagregación pedidas (`sexo`, `area`, `edad`) y `poblacion`.

See Also

`pob_unir()` y `pob_tasa()` para unir la población con tus datos.

Examples

```
# Población total de las provincias en 2025
poblacion("provincia", anio = 2025)

# Cantones de Loja por sexo, 2024 a 2026
poblacion("canton", anio = 2024:2026, provincia = "Loja", sexo = TRUE)

# Menores de 5 años y de 65 y más, por cantón urbano/rural
poblacion("canton", anio = 2025, provincia = "Azuay", area = TRUE,
          edad = c(0, 5, 65))

# Parroquias de un cantón por grupos quinquenales (estimación pobecu)
poblacion("parroquia", anio = 2030, canton = "Puyango", edad = "quinquenal")

# Serie nacional 1950-2050
nac <- poblacion("nacional")
plot(nac$anio, nac$poblacion / 1e6, type = "l",
     xlab = "Año", ylab = "Millones de habitantes")


# Edades simples (se descargan la primera vez)
poblacion("provincia", anio = 2025, provincia = "Pichincha", edad = "simple")
```

Index

∗ **datasets**
    dpa_pob, 4
    pob_inec, 11
    pob_parroquias, 14

agrupar_cabeceras *(*agrupar_unidades*)*, 3
agrupar_unidades, 3
agrupar_unidades(), *14*

buscar_pob, 4

dpa_pob, 4, *4*, *15*

formatear_codigo, 5
formatear_codigo(), *17*

homologar_canton, 6
homologar_canton(), *3*, *5*, *8*
homologar_parroquia, 7
homologar_parroquia(), *3*, *5*, *6*, *15*

normalizar_nombre, 8

pob_ajustar, 9
pob_ajustar(), *10*
pob_cantones *(*pob_inec*)*, 11
pob_estimar, 10
pob_fuentes, 11
pob_inec, 11
pob_mapa, 13
pob_nacional *(*pob_inec*)*, 11
pob_parroquias, 14, *21*
pob_provincias *(*pob_inec*)*, 11
pob_tasa, 16
pob_tasa(), *14*, *21*
pob_unir, 17
pob_unir(), *16*, *21*
pobecu_cache_dir, 18
pobecu_datos_disponibles, 18
pobecu_datos_disponibles(), *14*, *21*
pobecu_limpiar_cache, 19
pobecu_usar_carpeta, 19
poblacion, 20
poblacion(), *10*, *12*, *14*, *17*

tools::R_user_dir(), *18*