

Package ‘censosbo’

August 7, 2026

Type Package

Title Access and Analysis of Bolivian Census Microdata

Version 2.0.0

Description Programmatic access to the microdata of the Bolivian population and housing censuses of 1976, 1992, 2001, 2012 and 2024, published by the National Statistics Institute of Bolivia (INE, <https://www.ine.gob.bo/>). Data files in Apache Parquet format are downloaded on demand from a companion data repository, cached locally, and can be filtered by department, province or municipality. Supports 'dplyr' workflows through Apache Arrow and SQL queries through 'DuckDB'. Includes variable dictionaries for every census year with a thematic taxonomy and contextual metadata (reference population, questionnaire item number and provenance of each variable), derived from the census questionnaires and from the Data Documentation Initiative (DDI) files of the INE ANDA catalogue; functions to harmonise variables across censuses for temporal comparison; and choropleth maps at the department and municipality level. Also includes the 2024 census aggregates for urban blocks and rural communities, with their geometries. Documentation and messages are in Spanish, the language of the source data.

License MIT + file LICENSE

Encoding UTF-8

Language es

LazyData true

LazyDataCompression xz

URL <https://lab-tecnosocial.github.io/censosbo/>,
<https://github.com/lab-tecnosocial/censosbo>

BugReports <https://github.com/lab-tecnosocial/censosbo/issues>

Depends R (>= 4.2)

Imports arrow (>= 14.0.0), curl, cli, fs, rlang, tools, stats, dplyr,
sf

Suggests duckdb (>= 0.9.0), DBI, ggplot2, DT, testthat (>= 3.0.0),
withr, knitr, rmarkdown, tidyr, scales

Config/testthat/edition 3
Config/censosbo/data-version 1.1.0
VignetteBuilder knitr
Config/roxygen2/version 8.0.0
NeedsCompilation no
Author Alex Ojeda Copa [aut, cre, cph] (organization: Lab TecnoSocial),
Lab TecnoSocial [cph, fnd]
Maintainer Alex Ojeda Copa <alex@labtecnicosocial.org>
Repository CRAN
Date/Publication 2026-08-07 16:30:12 UTC

Contents

censosbo_cache_clear	3
censosbo_cache_dir	4
censosbo_cache_info	4
censo_bloques_meta	5
censo_temas	5
censo_temas_meta	6
codebook_1976	8
codebook_docs	10
codebook_docs_meta	11
codebook_historico_meta	12
codebook_meta	13
codebook_valores	14
departamentos	15
etiquetar_geografia	15
etiquetar_valores	16
etiquetar_variables	18
geo_bolivia	19
geo_departamentos	20
geo_municipios	20
get_emigracion_2024	21
get_fichas_2024	23
get_geo_comunidades	25
get_geo_manzanos	26
get_mortalidad_2024	27
get_personas_2024	28
get_poblacion_1976	30
get_temporal	34
get_temporal_vivienda	37
get_unidades_2024	38
get_viviendas_2024	40
grupos_variables	42
mapa_dep	42

censosbo_cache_clear3

mapa_man43

mapa_mun45

municipios46

provincias47

tipos_vivienda47

variables_armonizadas48

variable_temporal_map49

vars_tema50

Index52

censosbo_cache_clear	<i>Limpia el caché local de datos</i>
----------------------	---------------------------------------

Description

Elimina todos los archivos Parquet descargados localmente. Los datos se pueden volver a descargar usando las funciones ‘get_*()’.

Usage

```
censosbo_cache_clear(ask = TRUE)
```

Arguments

ask Lógico. Si ‘TRUE’ (defecto), pide confirmación antes de borrar.

Value

Invisible ‘NULL’.

Examples

```
## Not run:
censosbo_cache_clear()

## End(Not run)
```

censosbo_cache_dir	<i>Directorio de caché local del paquete</i>
--------------------	--

Description

Devuelve la ruta donde se guardan los archivos Parquet descargados. Por defecto usa el directorio estándar del sistema operativo, pero puede redirigirse a cualquier ruta local (por ejemplo, dentro del proyecto actual) estableciendo la opción ‘censosbo.cache_dir’ antes de llamar a ‘get_*()’.

Usage

```
censosbo_cache_dir()
```

Details

Para guardar el caché dentro de tu proyecto en lugar del directorio del sistema, añade esto al inicio de tu script o en tu ‘.Rprofile’:

```
““r options(censosbo.cache_dir = "data/censosbo") ““
```

El directorio se crea automáticamente si no existe.

Value

Ruta al directorio de caché (cadena de caracteres).

Examples

```
censosbo_cache_dir()

# Redirigir el caché a una carpeta del proyecto
anterior <- options(censosbo.cache_dir = file.path(tempdir(), "censosbo"))
censosbo_cache_dir()
options(anterior)
```

censosbo_cache_info	<i>Información sobre los archivos en caché</i>
---------------------	--

Description

Muestra los archivos Parquet descargados localmente, con sus tamaños y fechas de descarga.

Usage

```
censosbo_cache_info()
```

Value

Un data.frame con columnas 'archivo', 'tamanio' y 'modificado', o 'NULL' invisible si el caché está vacío.

Examples

```
censosbo_cache_info()
```

censo_bloques_meta	<i>Bloques temáticos de los indicadores de manzano y comunidad</i>
--------------------	--

Description

Los 15 bloques con que el INE organiza las fichas censales del CPV-2024 en su geoportal. Es un desglose más fino que el tema: 'salud_lugar' y 'salud_seguro' son dos bloques dentro del tema 'salud_seguridad_social'.

Usage

```
censo_bloques_meta
```

Format

Un data.frame de 15 filas con columnas 'bloque', 'etiqueta', 'tema', 'capitulo' y 'orden'.

Details

Se publica porque hasta ahora esta agrupación vivía duplicada a mano en los proyectos que consumen el paquete. La fuente única son 'data-raw/fichas/campos.csv' y 'campos_vivienda.csv'.

Source

INE Bolivia, fichas censales del geoportal del CPV-2024.

censo_temas	<i>Consulta los temas de un censo, con el número de variables de cada uno</i>
-------------	---

Description

Devuelve el catálogo de [censo_temas_meta] contando, en vivo, cuántas variables tiene cada tema en el censo y la tabla que se pidan. El conteo se calcula sobre el codebook en el momento de la llamada, así que nunca se desincroniza.

Usage

```
censo_temas(tema = NULL, capitulo = NULL, tabla = NULL, anio = 2024)
```

Arguments

tema	Caracteres. Filtra a uno o varios temas por su slug.
capitulo	Caracteres. Filtra por capítulo del cuestionario del CPV-2024: la letra ("C") o parte de su nombre ("vivienda").
tabla	Caracteres. Restringe el conteo a una tabla (e.g. "persona"). Con este argumento se omiten los temas que no tienen ninguna variable ahí.
anio	Entero. '2024' (defecto), '2012', '2001', '1992' o '1976'.

Value

Un data.frame con 'tema', 'etiqueta', 'capitulo', 'capitulo_etiqueta', 'fuente', 'n_variables' y 'descripcion', ordenado por capítulo y cuestionario.

See Also

[vars_tema()] para obtener los nombres de las variables de un tema, y [codebook()] para sus etiquetas y categorías.

Examples

```
# Los 20 temas del CPV-2024, con cuántas variables tiene cada uno
censo_temas()

# Solo los temas presentes en la tabla de vivienda
censo_temas(tabla = "vivienda")

# Qué temas cubre el censo 2001
censo_temas(anio = 2001)
```

censo_temas_meta	Catálogo de temas de los censos de Bolivia
------------------	--

Description

Los 21 temas con los que 'censosbo' agrupa las variables censales. Diecisiete son los 'topics' oficiales que el INE declara en el catálogo ANDA del CPV-2024; los otros cuatro ('ubicacion_geografica', 'identificacion', 'materiales_construccion' y 'religion') son extensiones del paquete, marcadas en la columna 'fuente'.

Usage

```
censo_temas_meta
```

Format

Un data.frame de 21 filas:

tema Identificador en snake_case; es la clave que acepta `'codebook(tema = )'` y `[vars_tema()]`

etiqueta Nombre legible del tema

capitulo Capítulo principal del cuestionario CPV-2024 ("A"–"G")

capitulo_etiqueta Nombre del capítulo

capitulos Todos los capítulos en que aparece el tema, separados por coma

anios Censos en los que el tema tiene variables

fuelle "INE-ANDA" si es un topic oficial, "censosbo" si es una extensión del paquete

orden Orden de presentación: por capítulo y luego por cuestionario

descripcion Qué incluye el tema y, cuando la asignación es discutible, por qué se decidió así

Details

El mismo vocabulario se aplica a los cinco censos, de modo que `'codebook(tema = "educacion", anio = ...)` sea comparable entre ellos. Los ocho temas del vocabulario antiguo del INE (idénticos en 2012, 2001, 1992 y 1976) se mapearon a estos veintiuno: seis literales coinciden y dos son renombrados ('Hogar y/o Vivienda' y 'Empleo, Ocupación y Actividad Económica').

Capítulo y tema son dos facetas independientes

Cada variable tiene exactamente un 'capitulo' y un 'tema', pero el tema ****no**** está anidado en el capítulo: 'v01_tipoviv' está en el capítulo B y 'v17_tenencia' en el C, y los dos son 'vivienda_hogar'. En esta tabla, 'capitulo' es el capítulo principal del tema (donde vive la mayoría de sus variables) y 'capitulos' lista todos aquellos en los que aparece.

Los capítulos son los del cuestionario del CPV-2024 y solo se aplican a ese censo; en los censos anteriores la columna 'capitulo' del codebook queda a 'NA' y la estructura oficial de cada año vive en 'grupo_ine'.

Source

INE Bolivia — catálogo ANDA, estudios 132 (CPV-2024), 8 (CPV-2012), 10 (CNPV-2001), 47 (CNPV-1992) y 46 (CNPV-1976), elemento 'topcClas' del DDI.

See Also

`[censo_temas()]` para consultarlo con conteos de variables.

codebook_1976*Consulta el diccionario de variables de un censo de Bolivia*

Description

Permite buscar variables del censo por nombre, tabla o texto libre en las etiquetas.

Usage

```
codebook_1976(  
  variable = NULL,  
  tabla = NULL,  
  buscar = NULL,  
  tema = NULL,  
  origen = NULL  
)
```

```
codebook_1992(  
  variable = NULL,  
  tabla = NULL,  
  buscar = NULL,  
  tema = NULL,  
  origen = NULL  
)
```

```
codebook_2001(  
  variable = NULL,  
  tabla = NULL,  
  buscar = NULL,  
  tema = NULL,  
  origen = NULL  
)
```

```
codebook_2012(  
  variable = NULL,  
  tabla = NULL,  
  buscar = NULL,  
  tema = NULL,  
  origen = NULL  
)
```

```
codebook_2024(  
  variable = NULL,  
  tabla = NULL,  
  buscar = NULL,  
  tema = NULL,  
  capítulo = NULL,
```

```

    origen = NULL
  )

codebook(
  variable = NULL,
  tabla = NULL,
  buscar = NULL,
  anio = 2024,
  tema = NULL,
  capitulo = NULL,
  origen = NULL
)
```

Arguments

variable	Vector de caracteres. Nombre(s) de variable a consultar. Si 'NULL', devuelve todas.
tabla	Caracteres. Filtra por tabla (e.g., "persona", "vivienda"). Si 'NULL', devuelve todas las tablas.
buscar	Caracteres. Texto libre para buscar en las etiquetas y nombres de variables, y también en el tema y el capítulo (no distingue mayúsculas/minúsculas). Se interpreta como expresión regular.
tema	Caracteres. Filtra por tema, con los slugs de [censo_temas_meta] (e.g. "educacion", "servicios_basicos"). Acepta varios. Disponible en los cinco censos; ver [censo_temas()].
origen	Caracteres. Filtra por procedencia de la variable: "cuestionario" (pregunta directa), "derivada" (construida por el INE), "geografia", "identificador" o "indicador" (agregados de las fichas de manzano y comunidad).
capitulo	Caracteres. Filtra por capítulo del cuestionario del CPV-2024: la letra ("C") o parte de su nombre ("vivienda"). Solo aplica a 2024: los cuestionarios anteriores tienen otra estructura y varios numeran las secciones de vivienda y de persona en paralelo. El eje comparable entre censos es 'tema'.
anio	Entero. Año del censo: '2024' (defecto), '1976', '1992', '2001' o '2012'.

Value

Un data.frame con las variables que coinciden con los filtros. Las columnas son siempre las mismas para los cinco censos.

See Also

[censo_temas()] para el catálogo de temas, [vars_tema()] para obtener los nombres de variable de un tema, y [codebook_docs()] para la definición conceptual y la pregunta literal de cada variable.

Examples

```

# Ver etiqueta de una variable especifica del CPV-2024
codebook("p25-sexo")
```

```
# Variables de sexo en el censo 2012
codebook(buscar = "sexo", anio = 2012)

# Todas las variables de vivienda del censo 1992
codebook(tabla = "vivienda", anio = 1992)

# Por tema, y comparando entre censos
codebook(tema = "educacion", tabla = "persona")
codebook(tema = "educacion", tabla = "persona", anio = 2012)

# Por capítulo del cuestionario, solo las preguntas directas
codebook(capitulo = "C", origen = "cuestionario")
```

codebook_docs	<i>Definición, universo y pregunta literal de una variable</i>
---------------	--

Description

Accesor de [codebook_docs_meta]: los textos oficiales del INE sobre una variable. Útil para resolver qué mide exactamente, a quién se le preguntó, o cómo se construyó una variable derivada.

Usage

```
codebook_docs(variable, tabla = NULL, campos = NULL, anio = 2024)
```

Arguments

variable	Caracteres. Nombre(s) de variable.
tabla	Caracteres. Desambigua cuando la variable existe en varias tablas.
campos	Caracteres. Devuelve solo estas columnas de documentación (e.g. "definicion", "pregunta_literal", "regla_derivacion").
anio	Entero. Censo o censos: '2024' (defecto), '2012', '2001', '1992' o '1976'. Acepta varios, que es la forma de comparar cómo cambió una definición entre censos.

Value

Un data.frame con una fila por variable y año encontrados.

See Also

[codebook()] para las etiquetas y categorías, y [codebook_valores()] para los códigos de una variable categórica.

Examples

```
# ¿Qué mide exactamente y a quién se le preguntó?
codebook_docs("p40_lee", campos = c("definicion", "universo_literal"))

# ¿Cómo construyó el INE esta variable derivada?
codebook_docs("nivel_edu", campos = "regla_derivacion")
```

codebook_docs_meta	<i>Documentación conceptual de las variables censales</i>
--------------------	---

Description

Textos oficiales del INE para cada variable: qué mide, a quién se le preguntó, la pregunta tal como se leyó en campo, las instrucciones que recibió el censista y —para las variables que el INE construyó— cómo se calcularon.

Usage

```
codebook_docs_meta
```

Format

Un data.frame de 445 filas:

anio Censo: 2024, 2012, 2001, 1992 o 1976

variable, tabla Clave; coinciden con [codebook_meta]

variable_ddi Nombre en el diccionario del ANDA, que no siempre es el de los microdatos (en 2012 el ANDA usa ‘P23_PARENTES’ y los datos ‘P23’)

definicion Definición conceptual de la variable

universo_literal Población de referencia, en las palabras del INE. La versión normalizada y comparable está en ‘codebook_meta\$universo’

pregunta_literal La pregunta como se formuló, con sus opciones y saltos

regla_derivacion Cómo construyó el INE la variable; solo en las derivadas de 2024 y 1992

notas Advertencias del INE, como el significado de los códigos de omisión. Solo disponible en 2024

informante Quién respondía: “jefe_hogar”, “persona_misma”, “empadronador” u “observacion”

instruccion Instrucciones al censista. Es donde el INE define en términos operativos conceptos como hogar o residencia habitual

Details

Va en una tabla aparte de [codebook_meta] a propósito: algunos de estos textos pasan de los 4.000 caracteres y harían ilegible la salida de [codebook()]. Consúltese con [codebook_docs()].

Cubre las variables de los cinco censos que existen en el paquete (445 de las 539 que documenta el ANDA; el resto son campos de texto abierto e identificadores que los microdatos publicados no incluyen).

Source

INE Bolivia — catálogo ANDA, DDI de los estudios 132 (CPV-2024), 8 (CPV-2012), 10 (CNPV-2001), 47 (CNPV-1992) y 46 (CNPV-1976). El atributo "ddi" del objeto registra las URL, fechas de descarga y sha256 de los archivos usados. Los textos se reproducen literalmente; el INE los publica bajo la condición «Uso público».

codebook_historico_meta
<i>Diccionarios de variables de los censos históricos de Bolivia</i>

Description

Lista nombrada con los metadatos de variables de los censos 1976, 1992, 2001 y 2012. Cada elemento es un data.frame con la misma estructura que [codebook_meta].

Usage

codebook_historico_meta

Format

Una lista con elementos "1976", "1992", "2001" y "2012", cada uno un data.frame con las mismas columnas que [codebook_meta]. Estos objetos guardan 'tipo' y 'valores_codigos' en orden inverso al de 'codebook_meta', por razones históricas; [codebook()] reordena las columnas al devolver, así que consultado por ahí el esquema es idéntico en los cinco censos. Tres particularidades de contenido:

- variable** Conserva el nombre original del censo, que en los primeros años son códigos cortos ('p10', 'anioes1') y no nombres descriptivos
- tabla** Es la entidad REDATAM de origen, así que varía entre censos ("poblacion" en 1976 donde 2024 usa "persona")
- capitulo** Siempre 'NA': los capítulos son los del cuestionario del CPV-2024. La estructura oficial de cada año está en 'grupo_ine'

Source

INE Bolivia. Diccionarios Parquet generados por open-redatam, con el tema y el universo añadidos desde los DDI del catálogo ANDA (estudios 8, 10, 47 y 46) y los cuestionarios de cada censo.

codebook_meta	<i>Diccionario de variables del CPV-2024</i>
---------------	--

Description

Metadatos de todas las variables del Censo de Población y Vivienda 2024 de Bolivia, extraídos del Diccionario de Variables oficial del INE.

Usage

codebook_meta

Format

Un data.frame con las siguientes columnas:

variable Nombre de la variable (minúsculas, igual que en los datos)

etiqueta Descripción en español de la variable

tabla Tabla a la que pertenece: "persona", "vivienda", "emigracion", "mortalidad", "unidad" o "ficha"

valores_codigos Lista de data.frames con los códigos y etiquetas para variables categóricas; 'NULL' para variables numéricas o de texto

tipo Tipo de dato: "categorica", "numerica" o "texto". Una variable es "categorica" si sus valores son códigos con etiquetas (aunque sean números, como 'sexo' 1/2) o si su nombre indica un código de clasificación (sufijo 'cod': códigos geográficos, de ocupación, etc.); "texto" si almacena texto libre; "numerica" en los demás casos (conteos y medidas)

tema Uno de los 21 temas de [censo_temas_meta]. Es el eje comparable entre censos

capitulo Capítulo del cuestionario del CPV-2024 ("A"–"G"); 'NA' en los censos anteriores, cuya estructura oficial está en 'grupo_ine'

pregunta, pregunta_num Número de pregunta del formulario, como texto y como entero, para recorrer el censo en el orden en que se aplicó

origen Procedencia: "cuestionario" (pregunta directa), "derivada" (construida por el INE o por REDATAM), "geografia", "identificador" o "indicador" (agregados de las fichas de manzано y comunidad)

universo Población de referencia normalizada ("personas_5_mas", "mujeres_12_mas", "viviendas_presentes"...). Es el denominador correcto de la variable; 'NA' cuando el INE no lo declara

grupo_ine Agrupación oficial del censo de origen; 'NA' en 2024, que no la publica

bloque, denominador Bloque de [censo_bloques_meta] y denominador de los indicadores de manzано y comunidad; 'NA' en el resto de variables

valores_fuente De dónde salen las etiquetas de valor: "redatam", "ddi" o el diccionario oficial

Source

INE Bolivia, CPV-2024. Diccionario de Variables CPV 2024.xlsx, los cuestionarios censales y el DDI del estudio 132 del catálogo ANDA.

See Also

[censo_temas()] y [vars_tema()] para navegar por tema, y [codebook_docs_meta] para los textos conceptuales del INE.

codebook_valores	<i>Muestra los valores codificados de una variable categórica</i>
------------------	---

Description

Muestra los valores codificados de una variable categórica

Usage

```
codebook_valores(variable, anio = 2024)
```

Arguments

- variable Caracteres. Nombre de la variable (e.g., "p25_sexo", "P23").
- anio Entero. Año del censo: '2024' (defecto), '1976', '1992', '2001' o '2012'.

Value

Un data.frame con columnas 'codigo' y 'etiqueta', o un mensaje si la variable no tiene categorías.

Examples

```
# Los diccionarios vienen dentro del paquete: esto no descarga nada.
codebook_valores("p25_sexo")
codebook_valores("P23", anio = 2012)
```

departamentos	<i>Lista los departamentos de Bolivia</i>
Description	
Lista los departamentos de Bolivia	
Usage	
departamentos()	
Value	
Un [tibble][dplyr::tibble] con columnas 'idep' y 'nombre_dep'.	
Examples	
departamentos()	
etiquetar_geografia	<i>Añade nombres geográficos legibles a los microdatos</i>

Description

Une un data frame de microdatos con [geo_bolivia] para agregar los nombres de departamento, provincia y municipio ('nombre_dep', 'nombre_prov', 'nombre_mun') a partir de los códigos geográficos ('idep', 'ipro', 'imun'). Es el equivalente geográfico de [etiquetar_valores()] y evita el 'left_join' manual con [municipios()] que antes hacía falta para trabajar por municipio.

Usage

etiquetar_geografia(df)

Arguments

df Un data.frame ya materializado (tras 'collect()') que contenga al menos la columna 'idep'. El nivel de detalle se detecta automáticamente según las columnas presentes: solo 'idep' agrega 'nombre_dep'; 'idep'+ 'ipro' agrega también 'nombre_prov'; 'idep'+ 'ipro'+ 'imun' agrega los tres.

Details

Los códigos se normalizan a 2 dígitos antes de unir, por lo que funciona aunque 'idep'/'ipro'/'imun' vengan como enteros. Los nombres provienen de la geografía del CPV-2024 ([geo_bolivia]); en censos históricos algunos códigos de municipio pueden no tener correspondencia y quedar como 'NA'.

Value

El mismo ‘df’ con las columnas de nombre añadidas. Si ya existían columnas de nombre, se reemplazan.

See Also

[etiquetar_valores()], [municipios()]

Examples

```
## Not run:
library(dplyr)
get_personas_2024(departamento = "Cochabamba") |>
  count(idep, improv, imun) |>
  collect() |>
  etiquetar_geografia()

## End(Not run)
```

etiquetar_valores	<i>Etiqueta los valores de las variables categóricas</i>
-------------------	--

Description

Convierte los códigos numéricos de las columnas categóricas en factores con las etiquetas en español del diccionario oficial del INE. Detecta automáticamente el censo (1976, 1992, 2001, 2012 o 2024) a partir de los nombres de columna del data frame.

Usage

```
etiquetar_valores(df, columnas = NULL, anio = NULL)
```

Arguments

df	Un data.frame (resultado de ‘collect()’, ‘DBI::dbGetQuery()’ u otro).
columnas	Vector de caracteres con los nombres de columnas a etiquetar. Si ‘NULL’ (por defecto), etiqueta todas las columnas categóricas presentes.
anio	Entero. Año del censo: ‘1976’, ‘1992’, ‘2001’, ‘2012’ o ‘2024’. Si ‘NULL’ (por defecto), se detecta automáticamente a partir de los nombres de columna del data frame.

Details

La detección automática del censo funciona comparando los nombres de columna del data frame con los de cada codebook. Dado que cada censo usa una convención de nombrado única (p.ej. 'p25_sexo' en 2024 vs 'P25' en 1992), la detección es confiable incluso después de 'count()' o 'filter()'.

Los valores que no coinciden con ningún código del diccionario (incluyendo 'NA') quedan como 'NA' en el factor resultante. Para ver los códigos disponibles de una variable usa 'codebook_valores()'.

Con datos de [get_temporal()] / [get_temporal_vivienda()] se usan las etiquetas ****armonizadas**** (no las de un censo concreto). Esto se detecta por la columna 'anio'; conviene conservarla al resumir. Para variables cuyo nombre también existe como columna cruda ('nivel_edu', 'area', 'pea', 'pet'), la columna 'anio' es necesaria para distinguir datos temporales de datos crudos del CPV-2024.

Para volver de etiquetas a códigos: `“r as.integer(df$p25_sexo) # → 1, 2 as.character(df$p25_sexo) # → "Mujer", "Hombre" “`

Value

El mismo 'df' con las columnas categóricas convertidas a 'factor' con las etiquetas del diccionario. Las columnas no encontradas en el diccionario se devuelven sin cambios.

See Also

[etiquetar_variables()] para renombrar columnas con sus descripciones.

Examples

```
# El diccionario viene en el paquete, así que etiquetar no descarga nada: basta
# que las columnas se llamen como en el censo correspondiente.
conteo <- data.frame(p25_sexo = c(1, 2), n = c(2894112, 2762418))
etiquetar_valores(conteo)

# Censo 1992: los nombres de sus variables identifican el año por sí solos
etiquetar_valores(data.frame(P25 = c(1, 2), n = c(3199490, 3221302)))

# Año explícito, cuando los nombres son ambiguos entre censos
etiquetar_valores(data.frame(P25 = c(1, 2), n = c(10, 12)), anio = 1992)

# Sobre datos reales (requiere descarga)
## Not run:
get_personas_2024(departamento = "Pando", as = "tibble") |>
  etiquetar_valores() |>
  head(3)

## End(Not run)
```

etiquetar_variables *Etiqueta los nombres de las variables (columnas)*

Description

Reemplaza los nombres técnicos de las columnas (e.g., 'p25_sexo') por sus descripciones en español del diccionario oficial del INE (e.g., "25. Es mujer u hombre"). Detecta automáticamente el censo (1976, 1992, 2001, 2012 o 2024) a partir de los nombres de columna. Útil para tablas y reportes destinados a lectores no técnicos.

Usage

```
etiquetar_variables(df, anio = NULL)
```

Arguments

df	Un data.frame.
anio	Entero. Año del censo: '1976', '1992', '2001', '2012' o '2024'. Si 'NULL' (por defecto), se detecta automáticamente a partir de los nombres de columna del data frame.

Details

Las descripciones pueden contener espacios y caracteres especiales. En RMarkdown/Quarto se muestran directamente en tablas. Para referenciarlas en código R usa backticks: `df$`25. Es mujer u hombre``.

La detección automática del censo compara nombres de columna con los codebooks disponibles. Usa 'anio' explícito si el data frame tiene muy pocas columnas o solo columnas geográficas.

Value

El mismo 'df' con los nombres de columnas reemplazados por sus descripciones. Las columnas no encontradas en el diccionario —y las que están pero sin descripción, como algunas variables derivadas del censo 1976— conservan su nombre original.

See Also

[`etiquetar_valores()`] para convertir códigos numéricos a etiquetas.

Examples

```
# Renombrar columnas de un resumen del diccionario
codebook_meta[1:5, c("variable", "etiqueta", "tabla")] |>
  etiquetar_variables()

# Valores y nombres etiquetados a la vez
data.frame(p25_sexo = c(1, 2), n = c(2894112, 2762418)) |>
  etiquetar_valores() |>
```

```
etiquetar_variables()

# El mismo flujo sobre datos reales (requiere descarga)
## Not run:
get_personas_2024(departamento = "Pando") |>
  dplyr::count(p25_sexo, nivel_edu) |>
  dplyr::collect() |>
  etiquetar_valores() |>
  etiquetar_variables()

## End(Not run)
```

geo_bolivia	Tabla de geografía de Bolivia
-------------	-------------------------------

Description

Tabla con los 343 municipios de Bolivia y sus códigos geográficos oficiales del CPV-2024, extraídos del diccionario Redatam del INE.

Usage

```
geo_bolivia
```

Format

- Un data.frame con 343 filas y 6 columnas:
- idep** Código de departamento (2 dígitos, con cero a la izquierda)
 - nombre_dep** Nombre del departamento
 - ipro** Código de provincia
 - nombre_prov** Nombre de la provincia
 - imun** Código de municipio
 - nombre_mun** Nombre del municipio

Source

INE Bolivia, CPV-2024. Diccionario de variables Redatam (CEN24.dicX).

geo_departamentos	<i>Geometrías de los departamentos de Bolivia</i>
-------------------	---

Description

Objeto ‘sf’ con los 9 departamentos de Bolivia y sus geometrías poligonales. Compatible con ‘ggplot2::geom_sf()’ y la función [mapa_dep()].

Usage

geo_departamentos

Format

Un ‘sf’ data.frame con 9 filas y 3 columnas (más geometría):

- idep** Código de departamento (2 dígitos, con cero a la izquierda)
- nombre_dep** Nombre del departamento
- geometry** Geometría de polígono (CRS: WGS84 / EPSG:4326)

Source

Derivado por disolución de [geo_municipios], de modo que los bordes departamentales coinciden exactamente con los municipales.

See Also

[geo_municipios], [mapa_dep()]

geo_municipios	<i>Geometrías de los municipios de Bolivia</i>
----------------	--

Description

Objeto ‘sf’ con los 343 municipios del CPV-2024 y sus geometrías poligonales, incluidos los cuatro Gobiernos Autónomos Indígena Originario Campesinos (GAIOC) creados entre 2016 y 2023.

Usage

geo_municipios

Format

Un 'sf' data.frame con 343 filas y 8 columnas (más geometría):

idep Código de departamento

nombre_dep Nombre del departamento

iprov Código de provincia

nombre_prov Nombre de la provincia

imun Código de municipio

nombre_mun Nombre del municipio

capital Capital del municipio

superficie_km2 Superficie en kilómetros cuadrados

geometry Geometría de polígono (CRS: WGS84 / EPSG:4326)

Details

Las geometrías están simplificadas preservando la topología, así que los bordes entre municipios vecinos son idénticos y no quedan franjas vacías entre ellos. Los únicos huecos interiores del país son cuerpos de agua excluidos de la división municipal: el Salar de Uyuni y los lagos Poopó y Uru Uru.

Los códigos 'idep'/'iprov'/'imun' se verificaron uno a uno contra los puntos de comunidades del CPV-2024 ([get_geo_comunidades()]) y contra la población municipal de los microdatos, que coincide exactamente en los 343 municipios.

Source

Varias fuentes, detalladas en la viñeta **Mapas coropléticos**. Límites, capital y superficie: SDSN Bolivia (2025), **Límites Municipales Bolivia 2025** \[Archivo shapefile\], <<https://sdsnbolivia.org/datos-espaciales/>>. Códigos y nombres: INE Bolivia (Redatam, CPV-2024).

See Also

[geo_departamentos], [mapa_mun()], [geo_bolivia]

get_emigracion_2024 *Accede a los microdatos de emigración internacional del CPV-2024*

Description

Descarga y/o carga desde caché los datos de emigración internacional del Censo de Población y Vivienda 2024 de Bolivia.

Usage

```
get_emigracion_2024(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Vector de caracteres. Código(s) de departamento ("01"- "09") o nombre(s) (e.g., "La Paz", "Santa Cruz"). Si 'NULL', incluye todos los departamentos (descarga ~282 MB).
provincia	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cercado") de provincia. Si 'NULL', incluye todas. Los valores no existentes producen un error (en vez de un resultado vacío silencioso).
municipio	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cochabamba") de municipio. Si 'NULL', incluye todos. Si se especifica 'provincia'/'municipio' sin 'departamento', este se infiere del catálogo para no descargar todo el país. Un nombre repetido entre departamentos requiere indicar 'departamento' para desambiguar.
variables	Vector de caracteres. Nombres de columnas a seleccionar. Si 'NULL', devuelve todas (119 columnas). Las columnas geográficas ('idep', 'iprov', 'imun', 'i00') siempre se incluyen.
as	Formato de retorno: "arrow" (por defecto), "tibble" o "duckdb" (tabla "emigracion").
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

Registra los miembros del hogar que emigraron al exterior en los últimos 5 años. Variables principales: sexo, año de salida, edad al salir y país de destino ('pais_destino_cod').

Para emigración del censo 2012 usa [get_emigracion_2012()] o 'get_censo(2012, "emigracion")'.

Value

Ver [get_personas_2024()].

Examples

```
## Not run:
# Emigración por país de destino
library(dplyr)
```

```
get_emigracion_2024() |>
  count(pais_destino_cod, sort = TRUE) |>
  collect()

## End(Not run)
```

get_fichas_2024	Accede a los indicadores del CPV-2024 por manzano y comunidad
-----------------	---

Description

Devuelve las 194 variables de la ficha resumen del INE para cada unidad censal que la tenga disponible: población por edad y sexo, educación, salud, migración, empleo, actividad económica, vivienda, servicios básicos, TIC, materiales de construcción, hacinamiento y tipo de hogar.

Usage

```
get_fichas_2024(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  area = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Nombre o código del departamento (e.g., "Cochabamba" o "03"). 'NULL' (defecto) devuelve todo el país.
provincia	Nombre o código de la provincia. Opcional.
municipio	Nombre o código del municipio. Opcional.
area	Filtra por tipo de unidad: "urbano" (manzanos) o "rural" (comunidades). 'NULL' (defecto) devuelve ambas. La columna 'area' se almacena con el mismo código que en los microdatos (1 = Urbana, 2 = Rural), así que también acepta '1' o '2'.
variables	Vector de nombres de variables a devolver. Las columnas geográficas se conservan siempre. 'NULL' (defecto) devuelve todas.
as	Formato de retorno: "arrow" (por defecto), "tibble" o "duckdb" (tabla "fichas").
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar progreso.

Details

Contiene 150.744 unidades: las que el INE libera ('ficha == TRUE' en [get_unidades_2024()]). Para el universo completo, con población y viviendas de todas las unidades, usa esa función.

Todas las variables son conteos de personas, viviendas u hogares. Cada bloque temático trae su propio total, que sirve de denominador:

'pob_*' Población por grupo de edad y sexo. Total: 'pob_total_*'.

'edu_*' Nivel de instrucción de la población de 19 o más años.

'salud_lugar_*' Dónde acude por problemas de salud. ****Respuesta múltiple****: las categorías suman más que el total.

'salud_seguro_*' Registro al SUS o afiliación a seguros.

'nac_*', 'res_*' Lugar de nacimiento y residencia habitual.

'ocup_*', 'act_*' Categoría ocupacional y actividad económica de la población de 14 o más años.

'viv_*' Tipo, condición de ocupación y tenencia de la vivienda.

'serv_*' Energía eléctrica, agua, desagüe, combustible y basura.

'tic_*' Equipamiento del hogar. ****Respuesta múltiple****.

'mat_*' Material de paredes, revoque, techo y piso.

'hac_*' Hacinamiento por dormitorio: sin, medio o alto.

'hogar_*' Tipología del hogar.

Los bloques 'mat_*', 'hac_*' y 'hogar_*' tienen como base las viviendas particulares con personas presentes, es decir 'viv_tipo_presentes'.

Los sufijos '_h' y '_m' son hombres y mujeres; '_ns' es "sin especificar". Usa 'codebook(tabla = "ficha")' para la lista completa con sus etiquetas.

Value

Ver [get_unidades_2024()].

Source

Geoportal del INE Bolivia, <<https://geoportal.ine.gob.bo/>>.

Examples

```
## Not run:
# Acceso a agua por cañería en los manzanos de El Alto
library(dplyr)
get_fichas_2024(municipio = "El Alto", as = "tibble") |>
  mutate(pct_caneria = 100 * serv_agua_caneria / serv_agua_total) |>
  select(codigo, pct_caneria)

# Solo algunas variables, para no traer las 194
get_fichas_2024(departamento = "Oruro",
  variables = c("pob_total_h", "pob_total_m", "tic_internet"))

## End(Not run)
```

get_geo_comunidades *Descarga las geometrías de las comunidades rurales del CPV-2024*

Description

Descarga las geometrías de las comunidades rurales del CPV-2024

Usage

```
get_geo_comunidades(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  overwrite = FALSE,  
  verbose = TRUE  
)
```

Arguments

departamento	Nombre o código del departamento (e.g., "Cochabamba" o "03"). 'NULL' (defecto) devuelve todo el país.
provincia	Nombre o código de la provincia. Opcional.
municipio	Nombre o código del municipio. Opcional.
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar progreso.

Details

El INE publica la mayoría de las comunidades rurales como **puntos** (un centro aproximado), no como polígonos. Para mapas de coropletas rurales no hay superficie que rellenar: conviene usar [ggplot2::geom_sf()] con puntos graduados por tamaño o color.

Value

Un objeto 'sf' con columnas 'codigo', 'nombre', 'idep', 'ipro', 'imun' y 'geometria' (EPSG:4326).

Examples

```
## Not run:  
comunidades <- get_geo_comunidades(departamento = "Pando")  
plot(comunidades[,"nombre"])  
  
## End(Not run)
```

get_geo_manzanos

Descarga las geometrías de los manzanos urbanos del CPV-2024

Description

Descarga las geometrías de los manzanos urbanos del CPV-2024

Usage

```
get_geo_manzanos(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Nombre o código del departamento (e.g., "Cochabamba" o "03"). 'NULL' (defecto) devuelve todo el país.
provincia	Nombre o código de la provincia. Opcional.
municipio	Nombre o código del municipio. Opcional.
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar progreso.

Details

Los archivos están partidos por departamento (0,3 a 6,7 MB cada uno). Sin el argumento 'departamento' se descargan los nueve, unos 25 MB en total; para un mapa de un municipio conviene acotar.

Value

Un objeto 'sf' con columnas 'codigo', 'nombre', 'idep', 'iprov', 'imun' y 'geometria' (polígonos, EPSG:4326).

See Also

[mapa_man()] para dibujarlas, [get_geo_comunidades()] para el área rural.

Examples

```
## Not run:
manzanos <- get_geo_manzanos(municipio = "Sucre")
plot(manzanos[,"nombre"])

## End(Not run)
```

get_mortalidad_2024	Accede a los microdatos de mortalidad del CPV-2024
---------------------	--

Description

Descarga y/o carga desde caché los datos de mortalidad del Censo de Población y Vivienda 2024 de Bolivia.

Usage

```
get_mortalidad_2024(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)
```

Arguments

departamento	Vector de caracteres. Código(s) de departamento ("01"- "09") o nombre(s) (e.g., "La Paz", "Santa Cruz"). Si 'NULL', incluye todos los departamentos (descarga ~282 MB).
provincia	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cercado") de provincia. Si 'NULL', incluye todas. Los valores no existentes producen un error (en vez de un resultado vacío silencioso).
municipio	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cochabamba") de municipio. Si 'NULL', incluye todos. Si se especifica 'provincia'/'municipio' sin 'departamento', este se infiere del catálogo para no descargar todo el país. Un nombre repetido entre departamentos requiere indicar 'departamento' para desambiguar.
variables	Vector de caracteres. Nombres de columnas a seleccionar. Si 'NULL', devuelve todas (119 columnas). Las columnas geográficas ('idep', 'ipro', 'imun', 'i00') siempre se incluyen.
as	Formato de retorno: "arrow" (por defecto), "tibble" o "duckdb" (tabla "mortalidad").
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

Registra los fallecimientos ocurridos en el hogar durante los últimos 12 meses. Variables principales: mes y año de fallecimiento, edad, causa COVID-19 ('m214_cov'), sexo y si fue parto ('m216_parto').
Para mortalidad del censo 1992 usa [get_mortalidad_1992()] o 'get_censo(1992, "mortalidad")'.

Value

Ver [get_personas_2024()].

Examples

```
## Not run:
# Fallecimientos por departamento
library(dplyr)
get_mortalidad_2024() |>
  count(idep) |>
  collect()

## End(Not run)
```

get_personas_2024	Accede a los microdatos de personas del CPV-2024
-------------------	--

Description

Descarga y/o carga desde caché los datos de personas del Censo de Población y Vivienda 2024 de Bolivia, con filtros geográficos opcionales.

Usage

```
get_personas_2024(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Vector de caracteres. Código(s) de departamento ("01"- "09") o nombre(s) (e.g., "La Paz", "Santa Cruz"). Si 'NULL', incluye todos los departamentos (descarga ~282 MB).
provincia	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cercado") de provincia. Si 'NULL', incluye todas. Los valores no existentes producen un error (en vez de un resultado vacío silencioso).
municipio	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cochabamba") de municipio. Si 'NULL', incluye todos. Si se especifica 'provincia'/'municipio' sin 'departamento', este se infiere del catálogo para no descargar todo el país. Un nombre repetido entre departamentos requiere indicar 'departamento' para desambiguar.

variables	Vector de caracteres. Nombres de columnas a seleccionar. Si 'NULL', devuelve todas (119 columnas). Las columnas geográficas ('idep', 'ipro', 'imun', 'i00') siempre se incluyen.
as	Formato de retorno: "arrow" (Dataset Arrow, por defecto — lazy, no carga en RAM), "tibble" (trae a memoria RAM; precaución con datos grandes), o "duckdb" (conexión DBI activa con tabla "personas").
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

Los datos se descargan por departamento bajo demanda. Especificando 'departamento = "07"' solo se descarga 'persona_dep07.parquet' (~77 MB). Sin especificar departamento se descargan los 9 archivos (~282 MB en total).

El caché se almacena en 'censosbo_cache_dir()'. Usa 'censosbo_cache_info()' para ver los archivos descargados.

Para censos históricos usa [get_censo()] o los atajos [get_personas_1992()], [get_personas_2001()], [get_personas_2012()].

Value

Según 'as': - "arrow": un 'arrow::Dataset' (lazy, soporta dplyr y Arrow) - "tibble": un 'data.frame' con los datos en RAM - "duckdb": una conexión 'DBI' con tabla "personas" registrada; recuerda cerrarla con 'DBI::dbDisconnect(con, shutdown = TRUE)'.

Examples

```
## Not run:
# Datos de Santa Cruz como Arrow Dataset (lazy)
personas_sc <- get_personas_2024(departamento = "Santa Cruz")

# Filtrar y agregar sin traer a RAM
library(dplyr)
get_personas_2024(departamento = "07") |>
  filter(p26_edad >= 18) |>
  count(p25_sexo) |>
  collect()

# Seleccionar pocas variables
get_personas_2024(
  departamento = c("02", "03"),
  variables = c("p25_sexo", "p26_edad", "nivel_edu")
)

# Consulta SQL con DuckDB
con <- get_personas_2024(departamento = "07", as = "duckdb")
DBI::dbGetQuery(con, "SELECT p25_sexo, AVG(p26_edad) FROM personas GROUP BY 1")
DBI::dbDisconnect(con, shutdown = TRUE)
```

```
# Equivalente genérico
get_censo(2024, "persona", departamento = "07")

## End(Not run)
```

get_poblacion_1976	<i>Accede a los microdatos de cualquier censo de Bolivia</i>
--------------------	--

Description

Descarga y/o carga desde caché los microdatos de los censos de población de Bolivia de 1976, 1992, 2001, 2012 y 2024, con filtros geográficos opcionales. Es la API genérica por año; para el CPV-2024 delega en [get_personas_2024()] y sus funciones hermanas.

Usage

```
get_poblacion_1976(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)

get_viviendas_1976(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  universo = c("viviendas", "particulares", "colectivas", "todos"),
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)

get_personas_1992(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

```
get_viviendas_1992(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  universo = c("viviendas", "particulares", "colectivas", "todos"),  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)  
  
get_mortalidad_1992(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)  
  
get_personas_2001(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)  
  
get_viviendas_2001(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  universo = c("viviendas", "particulares", "colectivas", "todos"),  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)  
  
get_personas_2012(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  as = c("arrow", "tibble", "duckdb"),
```

```
    overwrite = FALSE,
    verbose = TRUE
)

get_viviendas_2012(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  universo = c("viviendas", "particulares", "colectivas", "todos"),
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)

get_emigracion_2012(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)

get_discapacidad_2012(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)

get_censo(
  anio,
  tabla = "persona",
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  variables = NULL,
  universo = c("viviendas", "particulares", "colectivas", "todos"),
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Vector de caracteres. Código(s) "01"- "09" o nombre(s) del departamento. Si 'NULL', incluye todos.
provincia	Vector de caracteres. Código(s) o nombre(s) de provincia. En 1992/2001/2012 acepta nombres (se resuelven contra el catálogo del CPV-2024). En **1976** solo acepta códigos numéricos (geografía cantonal distinta). Si 'NULL', incluye todas.
municipio	Vector de caracteres. Código(s) o nombre(s) de municipio. En 1992/2001/2012 acepta nombres; en **1976** solo códigos de cantón. Si 'NULL', incluye todos. Si el municipio no existe en el año solicitado, se emite una advertencia y se retorna 'NULL'.
variables	Vector de caracteres. Nombres de columnas a seleccionar. Si 'NULL', devuelve todas las columnas.
as	Formato de retorno: "arrow" (lazy, por defecto), "tibble" o "duckdb".
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.
universo	Solo para 'tabla = "vivienda"'. Qué registros de la entidad devolver: "viviendas" (por defecto, el universo oficial del INE), "particulares", "colectivas" o "todos" (la entidad cruda de REDATAM). Ver [get_viviendas_2024()] y [tipos_vivienda()].
anio	Entero. Año del censo: '1976', '1992', '2001', '2012' o '2024'.
tabla	Caracteres. Nombre de la tabla a consultar. Depende del año: - **1976**: "poblacion" (o "persona" como alias), "vivienda" - **1992**: "persona", "vivienda", "mortalidad" - **2001**: "persona", "vivienda" - **2012**: "persona", "vivienda", "emigracion", "discapacidad" - **2024**: "persona", "vivienda", "emigracion", "mortalidad"

Details

Todas las tablas exponen los códigos geográficos armonizados como columnas directas: 'idep', 'iprov' e 'imun' (2 dígitos, consistentes con el CPV-2024). Esto permite filtrar por geografía sin reconstruir la jerarquía REDATAM (antes requería un join 'persona → vivienda → municipio'). El filtrado se hace directamente sobre estas columnas, igual que en el CPV-2024.

El censo 1976 no tuvo municipios comparables (usó cantones), por lo que solo expone 'idep' e 'iprov'; el filtro de 'municipio' se aplica sobre el cantón.

Con 'anio = 2024' la llamada se redirige a [get_personas_2024()], [get_viviendas_2024()], [get_emigracion_2024()] o [get_mortalidad_2024()] según 'tabla', así que el resultado es idéntico al de esas funciones — incluido el nombre con que se registra la tabla en DuckDB ("personas", "viviendas", en plural, a diferencia de los censos históricos).

Value

Según 'as': - "arrow": un 'arrow::Dataset' o 'arrow::Table' (lazy cuando no hay filtros geo) - "tibble": un 'data.frame' con los datos en RAM - "duckdb": una conexión 'DBI' con la tabla registrada (con el nombre de 'tabla', p.ej. "persona"); cierra con 'DBI::dbDisconnect(con, shutdown = TRUE)'.

Advertencia sobre municipios

El número de municipios cambió entre censos. Un código de municipio válido en 2012 puede no existir en 1992. En ese caso se emite una advertencia y se retorna 'NULL' sin error.

El universo de vivienda

Con 'tabla = "vivienda"', todos los censos desde 1992 traen en la entidad de REDATAM registros que **no son viviendas**: personas censadas en la calle o en tránsito. El defecto 'universo = "viviendas"' los descuenta, que es cómo cuenta el INE. Los códigos afectados de cada censo están en [tipos_vivienda()]:

Censo	Variable	Códigos que no son vivienda	Registros
1976	'v01'	(no se preguntó)	0
1992	'V01'	13 Ambulante	4.939
2001	'V04'	24 Transeúntes	9.392
2012	'P01'	7 En tránsito, 8 Persona que vive en la calle	12.971
2024	'v01_tipoviv'	15 Persona vive en la calle, 16 En tránsito	10.287

La reconciliación con los tabulados oficiales está comprobada para el CPV-2024 (ver [get_viviendas_2024()]). Para los censos anteriores el criterio es el mismo, tomado del diccionario de cada año.

Examples

```
## Not run:
# Personas de Santa Cruz en el censo 2012
get_censo(2012, "persona", departamento = "07")

# Viviendas del censo 1992 en La Paz
get_censo(1992, "vivienda", departamento = "La Paz")

# La entidad cruda de vivienda del censo 2012, con calle y tránsito
get_censo(2012, "vivienda", universo = "todos")

# Todas las personas del censo 1976 (descarga completa ~46 MB)
get_censo(1976, "poblacion")

# El CPV-2024 también: equivale a get_personas_2024(departamento = "07")
get_censo(2024, "persona", departamento = "07")

# Consulta SQL sobre censo 2001
con <- get_censo(2001, "persona", departamento = "03", as = "duckdb")
DBI::dbGetQuery(con, "SELECT P28, COUNT(*) AS n FROM persona GROUP BY P28")
DBI::dbDisconnect(con, shutdown = TRUE)

## End(Not run)
```

get_temporal	Obtiene datos temporales comparables de la tabla persona entre censos
--------------	---

Description

Descarga y armoniza variables clave de múltiples censos para análisis de tendencias y comparaciones históricas. El resultado es un data.frame en formato largo ("tidy"), con una fila por individuo y una columna 'anio' que identifica el censo de origen.

Usage

```
get_temporal(
  variables = NULL,
  grupo = NULL,
  anios = c(1976L, 1992L, 2001L, 2012L, 2024L),
  departamento = NULL,
  verbose = TRUE
)
```

Arguments

variables	Vector de caracteres. Nombres de variables armonizadas a incluir. Usa [variables_armonizadas()] para ver las opciones disponibles. Si se especifica 'grupo', este parámetro se ignora.
grupo	Nombre de un grupo temático predefinido (ver [grupos_variables()]): "demografico", "educacion", "economia", "cultural", "migracion", "fertilidad". Si se especifica, 'variables' se ignora.
anios	Vector de enteros. Años de censo a incluir (cualquier subconjunto de 'c(1976, 1992, 2001, 2012, 2024)'). Por defecto todos.
departamento	Vector de caracteres. Código(s) "01"- "09" o nombre(s) de departamento. Si 'NULL', incluye todo el país.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

****Variables con limitaciones conocidas:**** - 'nivel_edu': la Ley Avelino Siñani (2010) cambió la nomenclatura en 2012. - 'identidad_indigena', 'idioma_materno': NO disponibles en 1976 o 1992. - 'pea', 'pet': no disponibles en 2001. - 'migracion_nac_dpto', 'migracion_rec_dpto': variables derivadas de comparación de departamento de nacimiento/residencia con el actual. Pueden contener NAs cuando la información de origen no fue registrada en el censo. ****Atención:**** 'migracion_rec_dpto' solo tiene los códigos 1 (mismo dpto) y 2 (otro dpto) en 1992 y 2001. Los códigos 3 (exterior) y 4 (no había nacido) solo existen en 1976, 2012 y 2024. Comparar distribuciones entre todos los años puede ser engañoso. - 'idioma_materno' en 1976: captura "idioma que habla", no el materno.

Para variables de vivienda usa [get_temporal_vivienda()].

Value

Un tibble con columnas 'anio', seguida de las variables solicitadas. Las columnas ausentes en un censo aparecen como 'NA' con un aviso. ****Nota:**** 'area' (1=Urbana, 2=Rural) y 'departamento' (código numérico) se incluyen siempre en el resultado, incluso si no fueron solicitados. Son útiles para estratificar o filtrar los datos.

Los universos poblacionales no siempre coinciden

La armonización iguala los **códigos**, no la **población** a la que se preguntó. Comparar distribuciones sin igualar el universo produce conclusiones falsas, y el INE cambió el filtro de edad de varias preguntas entre censos.

get_temporal() lo detecta y avisa: si alguna variable pedida no se preguntó a la misma población en todos los años solicitados, emite un aviso con el universo de cada censo y la edad mínima que los iguala. El dato viene de los diccionarios DDI del catálogo ANDA, así que la advertencia es del INE, no una estimación del paquete.

Siete de las variables armonizadas están afectadas. Los casos más marcados:

- **nivel_edu**: **19 años o más** en el CPV-2024 (es una derivada del INE) frente a 6 años en 1992 y 4 en 2001. Es la mayor divergencia de todas: sin filtrar, 2024 aparece con muchos más **NA** y da la impresión de estar "más educado". - **sabe_leer** y **asistencia_escolar**: el umbral se movió cuatro veces — 5 años (1976), 6 (1992), 4 (2001) y 5 (2024). - **estado_civil**: 12 años en 1976 y 2024, 15 en 2001. - **hijos_nacidos_vivos** y **hijos_sobrevivientes**: mujeres de 12 años o más en 1976, 1992 y 2024; personas de 15 años o más en 2001. - **idioma_materno**: 5 años (1976) frente a 4 (2001) y todas en 2024.

La columna **notas** de **[variables_armonizadas()]** recoge, además, las advertencias sobre los **códigos** de cada variable.

Los dos vocabularios temáticos

grupo acepta los seis grupos de **[grupos_variables()]**, que agrupan nombres **armonizados**, y también los slugs de **[censo_temas()]**, que agrupan las variables **crudas** de cada censo. Los seis originales son alias permanentes; pasar un slug de tema informa del grupo equivalente y devuelve lo mismo.

Examples

```
## Not run:
# Usando grupo temático
datos <- get_temporal(grupo = "educacion", anios = c(1992, 2001, 2012, 2024))
library(dplyr)
datos |> count(anio, asistencia_escolar)

# Evolución del nivel educativo en todo el país
get_temporal(variables = c("nivel_edu", "sexo"),
              anios = c(1976, 1992, 2001, 2012, 2024))

# Identidad cultural (solo 2001-2024)
get_temporal(grupo = "cultural", anios = c(2001, 2012, 2024))

## End(Not run)
```

get_temporal_vivienda *Obtiene datos temporales comparables de la tabla vivienda entre censos*

Description

Descarga y armoniza variables de vivienda de múltiples censos para análisis de tendencias en condiciones habitacionales. El resultado tiene una fila por vivienda y una columna 'anio' que identifica el censo de origen.

Usage

```
get_temporal_vivienda(  
  variables,  
  anios = c(1976L, 1992L, 2001L, 2012L, 2024L),  
  departamento = NULL,  
  verbose = TRUE  
)
```

Arguments

variables	Vector de caracteres. Nombres de variables armonizadas de vivienda. Usa [variables_armonizadas(tabla = "vivienda")] para ver las opciones disponibles.
anios	Vector de enteros. Años de censo a incluir. Por defecto todos.
departamento	Vector de caracteres. Código(s) de departamento ("01"- "09"). Si 'NULL', incluye todo el país.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

Variables disponibles para comparación temporal de vivienda: 'material_paredes', 'material_techo', 'material_piso', 'fuente_agua', 'energia_electrica', 'servicio_sanitario', 'tenencia_vivienda', 'habitaciones_total'.

****Limitaciones conocidas:**** - 'habitaciones_total' en 2024: variable codificada como categorías ordinales (1=Una, ..., 8=Ocho o más), no como número absoluto.

Value

Un tibble con columnas 'anio' + variables solicitadas. Una fila por vivienda. Las variables no disponibles en un año aparecen como 'NA'.

Examples

```
## Not run:  
# Evolución del acceso a agua potable  
agua <- get_temporal_vivienda(  
  variables = c("fuente_agua", "energia_electrica"),
```

```
    años = c(1992, 2001, 2012, 2024)
  )
  library(dplyr)
  agua |> count(año, fuente_agua) |> group_by(año) |>
    mutate(pct = n / sum(n))

## End(Not run)
```

get_unidades_2024	Accede al universo de unidades censales del CPV-2024
-------------------	--

Description

Devuelve todas las unidades censales de Bolivia —manzanos urbanos y comunidades rurales— con su población, sus viviendas y si el INE libera su ficha de indicadores. Es el nivel geográfico más fino disponible del Censo 2024: 268.604 unidades, frente a los 343 municipios de las demás funciones.

Usage

```
get_unidades_2024(
  departamento = NULL,
  provincia = NULL,
  municipio = NULL,
  area = NULL,
  variables = NULL,
  as = c("arrow", "tibble", "duckdb"),
  overwrite = FALSE,
  verbose = TRUE
)
```

Arguments

departamento	Nombre o código del departamento (e.g., "Cochabamba" o "03"). 'NULL' (defecto) devuelve todo el país.
provincia	Nombre o código de la provincia. Opcional.
municipio	Nombre o código del municipio. Opcional.
area	Filtra por tipo de unidad: "urbano" (manzanos) o "rural" (comunidades). 'NULL' (defecto) devuelve ambas. La columna 'area' se almacena con el mismo código que en los microdatos (1 = Urbana, 2 = Rural), así que también acepta '1' o '2'.
variables	Vector de nombres de variables a devolver. Las columnas geográficas se conservan siempre. 'NULL' (defecto) devuelve todas.
as	Formato de retorno: "arrow" (por defecto), "tibble" o "duckdb" (tabla "unidades").
overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar progreso.

Details

La columna 'ficha' indica si esta unidad tiene indicadores en [get_fichas_2024()]. El INE no publica la ficha de las unidades con poca población, por reserva estadística: deja fuera al 47 las 150.744 unidades con ficha cubren el 92 excluidas son las pequeñas.

Los códigos de unidad ('codigo') no son jerárquicos: no contienen el departamento ni el municipio. Por eso las columnas 'idep', 'iprov' e 'imun' vienen desnormalizadas, y se pueden etiquetar con [etiquetar_geografia()].

Agregadas por municipio, 'personas' y 'viviendas' coinciden **exactamente** con [get_personas_2024()] y [get_viviendas_2024()] en los 343 municipios: 11.365.333 personas y 4.480.201 viviendas. Cualquiera de las dos fuentes sirve para el total de un territorio.

Ojo con la comparación de viviendas: el geoportal cuadra con el universo oficial, no con la entidad cruda de REDATAM. Si pides 'get_viviendas_2024(universo = "todos")' saldrán 4.490.488, un 0,23 porque esa entidad incluye 10.287 registros de personas en la calle o en tránsito que no son viviendas (ver [tipos_vivienda()]).

Value

Un 'Dataset' de arrow, un 'tibble' o una conexión DuckDB, según 'as'.

Source

Geoportal del INE Bolivia, <<https://geoportal.ine.gob.bo/>>.

See Also

[get_fichas_2024()] para los indicadores, [get_geo_manzanos()] y [get_geo_comunidades()] para las geometrías.

Examples

```
## Not run:
# Todas las unidades de Cochabamba
get_unidades_2024(departamento = "Cochabamba")

# Comunidades rurales de Potosí, con nombres de la geografía
library(dplyr)
get_unidades_2024(departamento = "Potosí", area = "rural", as = "tibble") |>
  etiquetar_geografia()

# ¿Qué parte de la población queda cubierta por las fichas?
get_unidades_2024(municipio = "Sucre", as = "tibble") |>
  summarise(cobertura = sum(personas[ficha]) / sum(personas))

## End(Not run)
```

get_viviendas_2024	Accede a los microdatos de viviendas del CPV-2024
--------------------	---

Description

Descarga y/o carga desde caché los datos de viviendas del Censo de Población y Vivienda 2024 de Bolivia, con filtros geográficos opcionales.

Usage

```
get_viviendas_2024(  
  departamento = NULL,  
  provincia = NULL,  
  municipio = NULL,  
  variables = NULL,  
  universo = c("viviendas", "particulares", "colectivas", "todos"),  
  as = c("arrow", "tibble", "duckdb"),  
  overwrite = FALSE,  
  verbose = TRUE  
)
```

Arguments

departamento	Vector de caracteres. Código(s) de departamento ("01"- "09") o nombre(s) (e.g., "La Paz", "Santa Cruz"). Si 'NULL', incluye todos los departamentos (descarga ~282 MB).
provincia	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cercado") de provincia. Si 'NULL', incluye todas. Los valores no existentes producen un error (en vez de un resultado vacío silencioso).
municipio	Vector de caracteres. Código(s) ("01") o nombre(s) ("Cochabamba") de municipio. Si 'NULL', incluye todos. Si se especifica 'provincia'/'municipio' sin 'departamento', este se infiere del catálogo para no descargar todo el país. Un nombre repetido entre departamentos requiere indicar 'departamento' para desambiguar.
variables	Vector de caracteres. Nombres de columnas a seleccionar. Si 'NULL', devuelve todas (119 columnas). Las columnas geográficas ('idep', 'ipro', 'imun', 'i00') siempre se incluyen.
universo	Qué registros de la entidad devolver: - "viviendas" (por defecto): el universo oficial del INE, 4.480.201 viviendas. Excluye a las personas censadas en la calle o en tránsito. - "particulares": solo viviendas particulares (4.463.773). - "colectivas": solo viviendas colectivas (16.428). - "todos": los 4.490.488 registros crudos de la entidad de REDATAM, incluidas las personas en la calle y en tránsito.
as	Ver [tipos_vivienda()] para el detalle código a código. Formato de retorno: "arrow" (por defecto), "tibble" o "duckdb" (tabla "viviendas").

overwrite	Lógico. Si 'TRUE', re-descarga aunque exista en caché.
verbose	Lógico. Mostrar mensajes de progreso. Por defecto 'TRUE'.

Details

La tabla de viviendas contiene 48 variables. Se puede unir con [get_personas_2024()] usando la clave 'idep + improv + imun + i00'.

Para viviendas de censos históricos usa [get_viviendas_1992()], [get_viviendas_2001()], [get_viviendas_2012()] o [get_censo()].

Value

Ver [get_personas_2024()].

El universo de vivienda

La entidad 'vivienda' de REDATAM tiene 4.490.488 registros, pero **10.287 no son viviendas**: son personas censadas fuera de una vivienda, marcadas en 'v01_tipoviv' como *persona que vive en la calle* (código 15, 3.311 registros) o *en tránsito: terminal, aeropuerto, puerto u otro* (código 16, 6.976). El INE no las cuenta como viviendas en ningún tabulado, y descontarlas da exactamente su total oficial de **4.480.201 viviendas** —también por área (2.898.140 urbanas y 1.582.061 rurales) y en los 343 municipios, donde coincide al registro con las viviendas del geportal ([get_unidades_2024()]).

Por eso el defecto es 'universo = "viviendas"'. Para contar los registros crudos de la entidad, como los devuelve REDATAM, usa 'universo = "todos"'.

El filtro necesita la columna 'v01_tipoviv': si pasas 'variables' sin incluirla, se añade automáticamente al resultado.

See Also

[tipos_vivienda()] para los códigos de cada grupo.

Examples

```
## Not run:
# Viviendas de Cochabamba (universo oficial)
get_viviendas_2024(departamento = "Cochabamba")

# Solo viviendas colectivas del país
get_viviendas_2024(universo = "colectivas")

# La entidad cruda de REDATAM, con calle y tránsito
get_viviendas_2024(universo = "todos")

# Condiciones de servicios básicos en Oruro
library(dplyr)
get_viviendas_2024(departamento = "04",
  variables = c("urbrur", "v07_aguapro", "v09_energia")) |>
  count(urbrur, v07_aguapro) |>
  collect()
```

```
## End(Not run)
```

grupos_variables	<i>Grupos temáticos predefinidos de variables armonizadas</i>
------------------	---

Description

Devuelve la lista de grupos temáticos disponibles y las variables que contiene cada uno, para usar con el parámetro ‘grupo’ de [get_temporal()].

Usage

```
grupos_variables()
```

Value

Una lista nombrada donde cada elemento es un vector de nombres de variables armonizadas.

Examples

```
grupos_variables()
grupos_variables()$educacion
```

mapa_dep	<i>Visualiza una variable censal a nivel departamental</i>
----------	--

Description

Genera un mapa coroplético de Bolivia a nivel de departamento.

Usage

```
mapa_dep(
  datos,
  variable,
  titulo = NULL,
  etiqueta_fill = NULL,
  paleta = NULL,
  na_color = "grey80",
  mostrar_nombres = FALSE
)
```

Arguments

datos	Data.frame con al menos las columnas 'idep' y 'variable'. Típicamente el resultado de una agregación con dplyr.
variable	Nombre (caracter) de la columna a visualizar.
titulo	Título del mapa. Si 'NULL', usa el nombre de la variable.
etiqueta_fill	Etiqueta de la leyenda. Si 'NULL', usa 'variable'.
paleta	Paleta de color. Por defecto "Blues" (continua) o "Set3" (categórica).
na_color	Color para departamentos sin datos. Por defecto "grey80".
mostrar_nombres	Si 'TRUE', agrega etiquetas con nombres de departamentos.

Details

Compatible con todos los censos (1976–2024): los 9 departamentos son estables. Para el censo 1976, que usa la columna 'dep' en lugar de 'idep', renómbrala: 'datos\$idep <- datos\$dep' (los códigos se normalizan a 2 dígitos automáticamente, así que da igual que vengan como enteros).

Value

Un objeto 'ggplot' modificable con capas adicionales de ggplot2.

Examples

```
## Not run:
library(dplyr)
pob <- get_personas_2024(as = "tibble") |>
  count(idep, name = "poblacion")
mapa_dep(pob, "poblacion", titulo = "Población por departamento (CPV-2024)")

## End(Not run)
```

mapa_man	Visualiza una variable a nivel de manzano o comunidad
----------	---

Description

Genera un mapa de un municipio al máximo detalle disponible del CPV-2024: cada manzano urbano como polígono y cada comunidad rural como punto.

Usage

```
mapa_man(
  datos,
  variable,
  municipio,
  departamento = NULL,
```

```

    area = NULL,
    titulo = NULL,
    etiqueta_fill = NULL,
    paleta = NULL,
    na_color = "grey80",
    tamaño_punto = 0.9
  )

```

Arguments

<code>datos</code>	Data.frame con una columna ‘codigo’ de unidad censal y la columna a visualizar. Típicamente sale de <code>[get_fichas_2024()]</code> o <code>[get_unidades_2024()]</code> .
<code>variable</code>	Nombre (carácter) de la columna a visualizar.
<code>municipio</code>	Nombre o código del municipio a mapear. Obligatorio: son ~268.000 unidades en todo el país y un mapa nacional no se lee.
<code>departamento</code>	Departamento del municipio. Solo hace falta para desambiguar nombres repetidos entre departamentos.
<code>area</code>	Qué unidades incluir: “urbano”, “rural” o ambas (defecto).
<code>titulo</code>	Título del mapa. Si ‘NULL’, usa el nombre de la variable.
<code>etiqueta_fill</code>	Etiqueta de la leyenda. Si ‘NULL’, usa ‘variable’.
<code>paleta</code>	Paleta de color. Por defecto “Blues” (continua) o “Set3” (categórica).
<code>na_color</code>	Color para unidades sin datos. Por defecto “grey80”.
<code>tamaño_punto</code>	Tamaño de los puntos de las comunidades rurales.

Details

Las geometrías se descargan al caché con `[get_geo_manzanos()]` y `[get_geo_comunidades()]` la primera vez; después se reutilizan.

Las unidades del municipio que no estén en ‘datos’ se dibujan con ‘na_color’. Eso es lo normal al mapear `[get_fichas_2024()]`: el 47 tiene ficha por reserva estadística, y salen en gris.

Value

Un objeto ‘ggplot’ modificable con capas adicionales de ggplot2.

See Also

`[mapa_mun()]` para el nivel municipal, `[mapa_dep()]` para el departamental.

Examples

```

## Not run:
library(dplyr)
agua <- get_fichas_2024(municipio = "Sucre", as = "tibble") |>
  mutate(pct_caneria = 100 * serv_agua_caneria / serv_agua_total)
mapa_man(agua, "pct_caneria", municipio = "Sucre",
  titulo = "% viviendas con agua por cañería - Sucre (2024)")

```

```
## End(Not run)
```

mapa_mun	<i>Visualiza una variable censal a nivel municipal</i>
----------	--

Description

Genera un mapa coroplético de Bolivia a nivel de municipio.

Usage

```
mapa_mun(  
  datos,  
  variable,  
  departamento = NULL,  
  titulo = NULL,  
  etiqueta_fill = NULL,  
  paleta = NULL,  
  na_color = "grey80",  
  mostrar_nombres = FALSE  
)
```

Arguments

datos	Data.frame con al menos las columnas ‘idep’, ‘ipro’v’, ‘imun’ y ‘variable’. Típicamente el resultado de una agregación con dplyr.
variable	Nombre (caracter) de la columna a visualizar.
departamento	Código o nombre de departamento para limitar el mapa. Si ‘NULL’ (defecto), muestra Bolivia completa.
titulo	Título del mapa. Si ‘NULL’, usa el nombre de la variable.
etiqueta_fill	Etiqueta de la leyenda. Si ‘NULL’, usa ‘variable’.
paleta	Paleta de color. Por defecto “Blues” (continua) o “Set3” (categórica).
na_color	Color para municipios sin datos. Por defecto “grey80”.
mostrar_nombres	Si ‘TRUE’, agrega etiquetas de nombres de municipios. Recomendado solo al filtrar por un departamento.

Details

Los datos se unen con [geo_municipios] por la clave ‘idep + iprov + imun’; los códigos se normalizan a 2 dígitos, así que también funcionan si vienen como enteros. La capa cubre los 343 municipios del CPV-2024, incluidos los cuatro GAIOC creados entre 2016 y 2023. Los municipios de ‘datos’ que no emparejen generan una advertencia y los que no tengan dato salen en gris (‘na_color’).

Para censos anteriores a 2012, usar con precaución: los códigos de municipio pueden no corresponder a la división actual (en 1976 son cantones).

Value

Un objeto ‘ggplot’ modificable con capas adicionales de ggplot2.

Examples

```
## Not run:
library(dplyr)
agua <- get_viviendas_2024(departamento = "07", as = "tibble") |>
  group_by(idep, improv, imun) |>
  summarise(pct_agua = mean(v07_aguapro == 1, na.rm = TRUE) * 100, .groups = "drop")
mapa_mun(agua, "pct_agua", departamento = "07",
  titulo = "% viviendas con agua por cañería - Santa Cruz (2024)")

## End(Not run)
```

municipios	<i>Lista los municipios de Bolivia</i>
------------	--

Description

Lista los municipios de Bolivia

Usage

```
municipios(departamento = NULL, provincia = NULL)
```

Arguments

- departamento Código o nombre del departamento. Opcional.
- provincia Código o nombre de provincia. Opcional.

Value

Un [tibble][dplyr::tibble] con columnas ‘idep’, ‘nombre_dep’, ‘improv’, ‘nombre_prov’, ‘imun’, ‘nombre_mun’.

Examples

```
municipios(departamento = "Cochabamba")
municipios(departamento = "Cochabamba", provincia = "Cercado")
```

provincias	<i>Lista las provincias de un departamento</i>
------------	--

Description

Lista las provincias de un departamento

Usage

provincias(departamento)

Arguments

departamento Código (e.g., "07") o nombre (e.g., "Santa Cruz") del departamento. Acepta vectores.

Value

Un [tibble][dplyr::tibble] con columnas 'idep', 'nombre_dep', 'iprov', 'nombre_prov'.

Examples

```
provincias("Santa Cruz")
provincias("02")
```

tipos_vivienda	<i>Tipos de vivienda de un censo y su grupo</i>
----------------	---

Description

Devuelve los códigos de la variable de tipo de vivienda de un censo, con su etiqueta y el grupo al que pertenecen. Sirve para auditar qué registros entran y salen con cada valor de 'universo' en [get_viviendas_2024()] y [get_censo()].

Usage

tipos_vivienda(anio = 2024)

Arguments

anio Entero. Año del censo: '1976', '1992', '2001', '2012' o '2024'.

Details

El grupo ‘no_vivienda’ es la razón de que la entidad ‘vivienda’ de REDATAM tenga más registros que el total oficial de viviendas: son personas censadas fuera de una vivienda. En el CPV-2024 son 10.287 registros (3.311 en la calle y 6.976 en tránsito) de 4.490.488, y descontarlos da los 4.480.201 del INE.

El censo 1976 no tuvo categorías de calle ni tránsito: ahí ningún código es ‘no_vivienda’ y el universo oficial coincide con la entidad completa.

Los códigos ‘sin_clasificar’ (el 88 de 1976 y el 0 de 1992) **se conservan** en ‘universo = "viviendas"’: no consta que sean calle ni tránsito, así que descartarlos sería una decisión sin respaldo en el diccionario. Sí quedan fuera de “particulares” y “colectivas”, que son selecciones positivas.

Value

Un ‘data.frame’ con las columnas:

‘**codigo**’ Código de la variable de tipo de vivienda.

‘**etiqueta**’ Etiqueta del diccionario del censo.

‘**grupo**’ “particular”, “colectiva”, “no_vivienda” (persona en la calle o en tránsito) o “sin_clasificar” (código que aparece en los microdatos pero no en el diccionario del censo).

‘**en_universo_viviendas**’ ‘TRUE’ si el registro cuenta como vivienda para el INE, es decir si ‘universo = "viviendas"’ lo conserva.

See Also

[get_viviendas_2024()] y [get_censo()] para el argumento ‘universo’.

Examples

```
tipos_vivienda(2024)

# Los códigos que no son vivienda en cada censo
for (a in c(1976, 1992, 2001, 2012, 2024)) {
  t <- tipos_vivienda(a)
  cat(a, ":", paste(t$etiqueta[t$grupo == "no_vivienda"], collapse = "; "), "\n")
}
```

variables_armonizadas *Muestra el mapeo de variables comparables entre censos de Bolivia*

Description

Retorna la tabla de variables armonizadas que pueden usarse en análisis temporales o comparativos entre los censos de 1976, 1992, 2001, 2012 y el CPV-2024.

Usage

```
variables_armonizadas(tabla = NULL)
```

Arguments

tabla Filtrar por tabla de origen: "persona", "vivienda" o 'NULL' para todas. Por defecto 'NULL'.

Value

Un data.frame con las variables armonizadas y sus equivalentes en cada año de censo. La columna 'armonizada' indica si los códigos son comparables entre censos ('TRUE') o crudos de cada año ('FALSE').

Examples

```
variables_armonizadas()
variables_armonizadas(tabla = "vivienda")
```

variable_temporal_map	<i>Variables armonizadas para análisis temporal</i>
-----------------------	---

Description

Tabla de correspondencia entre variables de los censos de Bolivia (1976, 1992, 2001, 2012 y CPV-2024), con las variables que pueden compararse a lo largo del tiempo.

Usage

```
variable_temporal_map
```

Format

Un data.frame con columnas:

variable Nombre armonizado (e.g., "sexo", "nivel_edu")

etiqueta Descripción en español

descripcion Descripción detallada y notas de comparabilidad

tabla Tabla de origen: "persona" o "vivienda"

armonizada 'TRUE' si 'get_temporal()' remapea los códigos a un esquema comparable entre censos; 'FALSE' si devuelve los códigos crudos de cada año (no comparables, p.ej. 'parentesco')

v1976, v1992, v2001, v2012, v2024 Nombre de la columna en cada censo ('NA' si no disponible)

notas Advertencias sobre diferencias metodológicas entre censos

tema Tema al que pertenece, con el vocabulario de [censo_temas_meta]. Permite pasar de la vista armonizada a la temática sin una tabla externa

Source

Elaboración propia a partir de los diccionarios oficiales del INE Bolivia.

vars_tema	<i>Nombres de las variables de un tema</i>
-----------	--

Description

Devuelve un vector de caracteres listo para pasar al argumento ‘variables’ de las funciones ‘get_*()’, o a ‘dplyr::select(dplyr::all_of())’.

Usage

```
vars_tema(  
  tema,  
  tabla = NULL,  
  tipo = NULL,  
  origen = NULL,  
  capitulo = NULL,  
  anio = 2024  
)
```

Arguments

tema	Caracteres. Uno o varios temas, con los slugs de [censo_temas_meta].
tabla	Caracteres. Restringe a una tabla. Conviene indicarla cuando el tema abarca varias, porque pasar variables de ‘vivienda’ a [get_personas_2024()] no devuelve nada útil.
tipo	Caracteres. Filtra por tipo de dato: “categorica”, “numerica” o “texto”.
origen	Caracteres. Filtra por procedencia: “cuestionario”, “derivada”, “geografia”, “identificador” o “indicador”.
capitulo	Caracteres. Filtra por capítulo del cuestionario (solo 2024).
anio	Entero. ‘2024’ (defecto), ‘2012’, ‘2001’, ‘1992’ o ‘1976’.

Details

Las variables salen en el orden del cuestionario (capítulo, luego número de pregunta) y las derivadas al final, para que la selección se lea igual que el formulario original.

Value

Un vector de caracteres, sin duplicados.

See Also

[censo_temas()] para ver qué temas existen.

Examples

```
vars_tema("educacion", tabla = "persona")

# Solo las preguntas directas, sin las derivadas del INE
vars_tema("caracteristicas_economicas", tabla = "persona", origen = "cuestionario")

## Not run:
# Descargar solo las variables de un tema
get_personas_2024(
  departamento = "Cochabamba",
  variables = vars_tema("educacion", tabla = "persona")
)

## End(Not run)
```

Index

* datasets

- censo_bloques_meta, [5](#)
- censo_temas_meta, [6](#)
- codebook_docs_meta, [11](#)
- codebook_historico_meta, [12](#)
- codebook_meta, [13](#)
- geo_bolivia, [19](#)
- geo_departamentos, [20](#)
- geo_municipios, [20](#)
- variable_temporal_map, [49](#)

- censo_bloques_meta, [5](#)
- censo_temas, [5](#)
- censo_temas_meta, [6](#)
- censosbo_cache_clear, [3](#)
- censosbo_cache_dir, [4](#)
- censosbo_cache_info, [4](#)
- codebook (codebook_1976), [8](#)
- codebook_1976, [8](#)
- codebook_1992 (codebook_1976), [8](#)
- codebook_2001 (codebook_1976), [8](#)
- codebook_2012 (codebook_1976), [8](#)
- codebook_2024 (codebook_1976), [8](#)
- codebook_docs, [10](#)
- codebook_docs_meta, [11](#)
- codebook_historico_meta, [12](#)
- codebook_meta, [13](#)
- codebook_valores, [14](#)

- departamentos, [15](#)

- etiquetar_geografia, [15](#)
- etiquetar_valores, [16](#)
- etiquetar_variables, [18](#)

- geo_bolivia, [19](#)
- geo_departamentos, [20](#)
- geo_municipios, [20](#)
- get_censo (get_poblacion_1976), [30](#)
- get_discapacidad_2012 (get_poblacion_1976), [30](#)

- get_emigracion_2012 (get_poblacion_1976), [30](#)
- get_emigracion_2024, [21](#)
- get_fichas_2024, [23](#)
- get_geo_comunidades, [25](#)
- get_geo_manzanos, [26](#)
- get_mortalidad_1992 (get_poblacion_1976), [30](#)
- get_mortalidad_2024, [27](#)
- get_personas_1992 (get_poblacion_1976), [30](#)
- get_personas_2001 (get_poblacion_1976), [30](#)
- get_personas_2012 (get_poblacion_1976), [30](#)
- get_personas_2024, [28](#)
- get_poblacion_1976, [30](#)
- get_temporal, [34](#)
- get_temporal_vivienda, [37](#)
- get_unidades_2024, [38](#)
- get_viviendas_1976 (get_poblacion_1976), [30](#)
- get_viviendas_1992 (get_poblacion_1976), [30](#)
- get_viviendas_2001 (get_poblacion_1976), [30](#)
- get_viviendas_2012 (get_poblacion_1976), [30](#)
- get_viviendas_2024, [40](#)
- grupos_variables, [42](#)

- mapa_dep, [42](#)
- mapa_man, [43](#)
- mapa_mun, [45](#)
- municipios, [46](#)

- provincias, [47](#)

- tipos_vivienda, [47](#)
- variable_temporal_map, [49](#)

variables_armonizadas, [48](#)
vars_tema, [50](#)