Package ‘tidycensuskr’

November 10, 2025

Title Easy Access for South Korea Census Data and Boundaries

Version 0.2.3

Description Census and administrative data in South Korea are a basic source of quantitative and mixed-methods research for social and urban scientists. This package provides a 'sf' (Pebesma et al., 2024 <doi:10.32614/CRAN.package.sf>) based standardized workflow based on direct open API access to the major census and administrative data sources and pre-generated files in South Korea.

License MIT + file LICENSE

LazyData true

Depends R (>= 4.2.0)

RoxygenNote 7.3.3

Imports sf, dplyr, tidyr, kosis, utils, rlang

Suggests knitr, withr, purrr, rmarkdown, ggplot2, biscale, cowplot, tmap, geodata, testthat (>= 3.0.0), ggspatial, spdep

Config/testthat/edition 3

Encoding UTF-8

VignetteBuilder knitr

URL `https://github.com/sigmafelix/tidycensuskr`, `https://sigmafelix.github.io/tidycensuskr/`, `https://sigmafelix.r-universe.dev/tidycensuskr/`

BugReports `https://github.com/sigmafelix/tidycensuskr/issues`

NeedsCompilation no

Author Insang Song [aut, cre] (ORCID: <`https://orcid.org/0000-0001-8732-3256`>), Sohyun Park [aut, ctb] (ORCID: <`https://orcid.org/0000-0002-1231-5662`>), Hyesop Shin [aut, ctb] (ORCID: <`https://orcid.org/0000-0003-2637-7933`>)

Maintainer Insang Song <`geoissong@snu.ac.kr`>

Repository CRAN

Date/Publication 2025-11-10 09:00:02 UTC

Contents

`adm2_sf_2020` *South Korea Census Boundary in 2020*

Description

District level boundary data in South Korea in 2020. `adm2_code` column can be used to join with an `anycensus()` output.

Usage

```
adm2_sf_2020
```

Format

A sf object with 250 rows and 3 variables:

Details

- year Year of the census data, e.g., 2010, 2015, or 2020
- adm2_code Code of the district/municipal-level (Sigungu) administrative unit
- geometry Geometry list-column

Source

- Statistical Geographic Information Service (SGIS) anycensus *Query Korean census data by admin code (province or municipality) and year*

Description

The function queries a long format census data frame (censuskor) for specific administrative codes (if provided)

Usage

```
anycensus(
  year = 2020,
  codes = NULL,
  type = c("population", "housing", "tax", "mortality", "economy", "medicine",
    "migration", "environment"),
  level = c("adm2", "adm1"),
  aggregator = sum,
  ...
)
```

Arguments

year	integer(1). One of 2010, 2015, or 2020.
codes	integer vector of admin codes (e.g. c(11, 26)) or character administrative area names (e.g. c("Seoul", "Daejeon")).
type	character(1). "population", "housing", "tax", "economy", "medicine", "migration", "environment" or "mortality" Defaults to "population".
level	character(1). "adm1" for province-level or "adm2" for municipal-level. Defaults to "adm2".
aggregator	function to aggregate values when level = "adm1".
...	additional arguments passed to the aggregator function. (e.g., na.rm = TRUE).

Value

A data.frame object containing census data for the specified codes and year.

Note

Using characters in codes has a side effect of returning all rows in the dataset that match year and type. The 'wide' table is returned with separate columns for each class1 and class2 and unit (abbreviated whereof) combination.

Examples

```
# Query mortality data for adm2_code 21 (Busan)
anycensus(codes = 21, type = "mortality")

# Query population data for adm1 "Seoul" or "Daejeon"
anycensus(codes = c("Seoul", "Daejeon"), type = "housing", year = 2015)

# Aggregate to adm1 level tax (province-level) using sum
anycensus(
  codes = c(11, 23, 31),
  type = "tax",
  year = 2020,
  level = "adm1",
  aggregator = sum,
  na.rm = TRUE
)
```

censuskor *South Korea Census Data*

Description

District level data including tax, population, business entities, housing, economy, medicine and mortality in South Korea in 2010, 2015, and/or 2020. The availble years and variables depend on the type of data.

Usage

```
censuskor
```

Format

A data.frame with 54218 rows and 10 variables:

Details

- year Year of the census data, e.g., 2010, 2015, or 2020
- adm1 Name of the province-level (Sido) administrative unit
- adm1_code Code of the province-level (Sido) administrative unit
- adm2 Name of the district/municipal-level (Sigungu) administrative unit
- adm2_code Code of the district/municipal-level (Sigungu) administrative unit
- type Type of variable, e.g., "population", "tax", "mortality", "housing", "medicine", "migration", "environment" or "economy"
- class1 First-level classification of the variable depending on the type
- class2 Second-level classification of the variable depending on the type
- unit Unit of measurement for the variable
- value Value of the variable

Note

NA values in the value field indicate that the data was omitted or suppressed. We kept these NA values as-is to reflect the original data from the source. For temporal comparison, province names in adm1 field are standardized to the common names with no suffix in metropolitan cities and "-do" suffix in provinces. For example, "Seoul" instead of "Seoul Metropolitan City", and "Jeollabuk-do" instead of "Jeonbuk State". "KRW" in the unit field stands for South Korean Won. Values are as-is unless otherwise noted in the unit field (e.g., "per 100k population" or "million KRW").

Source

- KOSIS (Korean Statistical Information Service)

`kr_grid_adm2_sgis_2020` *geofacet Grid for South Korea Administrative Districts (SGIS Standard, 2020)*

Description

A geofacet grid for South Korea administrative districts (*Si-Gun-Gu*) based on the Statistical Geographic Information Service (SGIS) standard in 2020. Non-autonomous districts in cities are retained as separate entities. This grid can be used with the geofacet package to create faceted visualizations based on geographic layout.

Usage

```
kr_grid_adm2_sgis_2020
```

Format

A data.frame with 250 rows and 6 variables

Details

- name Name of the district/municipal-level (Sigungu) administrative unit
- code SGIS code of the district/municipal-level (Sigungu) administrative unit
- row Row position in the geofacet grid
- col Column position in the geofacet grid

Source

- Statistical Geographic Information Service (SGIS)
- GitHub username chichead in GitHub geofacet issue page

load_districts *Load district boundaries for a specific year*

Description

Load district boundaries for a specific year

Usage

```
load_districts(year = 2020)
```

Arguments

`year`	The year for which to load district boundaries (2010, 2015, or 2020)

Value

An `sf` object containing district boundaries for the specified year

Note

This function requires the `tidycensussfkr` package to be installed. No explicit dependency is defind; but users should install the package following the instructions at vignette('v01_intro') or more succinctly: `install.packages('tidycensussfkr', repos = 'https://sigmafelix.r-universe.dev')`

set_kosis_key *Set KOSIS API Key from a File*

Description

This function reads a KOSIS API key from a specified file and sets it for use in KOSIS API calls.

Usage

```
set_kosis_key(file)
```

Arguments

`file`	A character string specifying the path to the file containing the KOSIS API key.

Details

The file should contain the API key as a single line of text. If the file does not exist, an error will be raised.

Value

No return value. A message will be printed to confirm that the key has been set.

Index