Package: censusapi (via r-universe)

September 2, 2024

Version 0.9.0.9000

Title Retrieve Data from the Census APIs

Description A wrapper for the U.S. Census Bureau APIs that returns data frames of Census data and metadata. Available datasets include the Decennial Census, American Community Survey, Small Area Health Insurance Estimates, Small Area Income and Poverty Estimates, Population Estimates and Projections, and more.

URL https://www.hrecht.com/censusapi/, https://github.com/hrecht/censusapi

BugReports https://github.com/hrecht/censusapi/issues

Depends R (>= 3.5)

License GPL-3

LazyData true

RoxygenNote 7.3.1

Roxygen list(markdown = TRUE)

Imports httr, jsonlite, rlang

Suggests knitr, rmarkdown

Encoding UTF-8

VignetteBuilder knitr

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

RemoteUrl https://github.com/hrecht/censusapi

RemoteRef HEAD

RemoteSha 74334d4d180ff79477456a777235afdde164f2cd

Contents

`fips` *List of state fips codes - 50 states plus DC*

Description

Some small geographies in some Census APIs can only be used under a state hierarchy. This is a list of fips codes that may be looped over to retrieve data for all states.

Usage

```
fips
```

Format

A list of fips codes for 50 states and the District of Columbia.

Examples

```
fips
```

`getCensus` *Retrieve Census data from a given API*

Description

Retrieve Census data from a given API

Usage

```
getCensus(
  name,
  vintage = NULL,
  key = NULL,
  vars,
  region = NULL,
  regionin = NULL,
  time = NULL,
  show_call = FALSE,
  convert_variables = TRUE,
  year = NULL,
```

```
  date = NULL,
  period = NULL,
  monthly = NULL,
  category_code = NULL,
  data_type_code = NULL,
  naics = NULL,
  pscode = NULL,
  naics2012 = NULL,
  naics2007 = NULL,
  naics2002 = NULL,
  naics1997 = NULL,
  sic = NULL,
  ...
)
```

Arguments

name	The programmatic name of your dataset, e.g. "timeseries/poverty/saipe" or "acs/acs5". Use listCensusApis() to see valid dataset names. Required.
vintage	Vintage (year) of dataset, e.g. 2014. Not required for timeseries APIs.
key	A Census API key, obtained at https://api.census.gov/data/key_signup.html. If you have a CENSUS_KEY or CENSUS_API_KEY stored in your .Renviron file, getCensus() will automatically use that key. Using a key is recommended but not required.
vars	List of variables to get. Required.
region	Geography to get.
regionin	Optional hierarchical geography to limit region.
time	Time period of data to get. Required for most timeseries APIs.
show_call	Display the underlying API call that was sent to the Census Bureau. Default is FALSE.
convert_variables	Convert columns that are likely numbers into numeric data. Default is TRUE. If false, all columns will be characters, which is the type returned by the Census Bureau.
year, date, period, monthly, category_code, data_type_code, naics, pscode, naics2012, naics2007, naics2002, naics1997, sic	Optional arguments used in some timeseries data APIs.
...	Other valid arguments to pass to the Census API. Note: the APIs are case sensitive.

Value

A data frame with results from the specified U.S. Census Bureau dataset.

Examples

```
# Get total population and median household income for Census places
# (cities, towns, villages) in a single state from the 5-year American Community Survey.
acs_simple <- getCensus(
  name = "acs/acs5",
  vintage = 2022,
  vars = c("NAME", "B01001_001E", "B19013_001E"),
  region = "place:*",
  regionin = "state:01")
head(acs_simple)

# Get all data from the B08301 variable group, "Means of Transportation to Work."
# This returns estimates as well as margins of error and annotation flags.
acs_group <- getCensus(
  name = "acs/acs5",
  vintage = 2022,
  vars = "group(B08301)",
  region = "state:*")
head(acs_group)

# Retreive 2020 Decennial Census block group data within a specific Census tract,
# using the regionin argument to precisely specify the Census tract, county,
# and state.
decennial_block_group <- getCensus(
 name = "dec/dhc",
 vintage = 2020,
 vars = c("NAME", "P1_001N"),
 region = "block group:*",
 regionin = "state:36+county:027+tract:220300")
head(decennial_block_group)

# Get poverty rates for children and for people of all ages beginning in 2000 using the
# Small Area Income and Poverty Estimates API
saipe <- getCensus(
  name = "timeseries/poverty/saipe",
  vars = c("NAME", "SAEPOVRT0_17_PT", "SAEPOVRTALL_PT"),
  region = "state:01",
  time = "from 2000")
head(saipe)

# Get the number of employees and number of establishments in the construction sector,
# NAICS2017 code 23, using the County Business Patterns API
cbp <- getCensus(
 name = "cbp",
 vintage = 2021,
 vars = c("EMP", "ESTAB", "NAICS2017_LABEL"),
 region = "county:*",
 NAICS2017 = 23)
head(cbp)
```

get_api_key *Retrieve a Census API key stored the .Renivron file*

Description

Retrieve a Census API key stored the .Renivron file

Usage

```
get_api_key()
```

Value

A CENSUS_KEY or CENSUS_API_KEY string stored in the user's .Renviron. file, or a warning message printed once per R session if none is found.

See Also

Other helpers: `has_api_key()`

Examples

```
## Not run:
get_api_key()

## End(Not run)
```

has_api_key *Is there a saved Census API key in the .Renivron file?*

Description

Is there a saved Census API key in the .Renivron file?

Usage

```
has_api_key()
```

Value

TRUE or FALSE.

See Also

Other helpers: `get_api_key()`

Examples

```
has_api_key()
```

`listCensusApis` *Get general information about available datasets*

Description

Scrapes https://api.census.gov/data.json and returns a dataframe that includes columns for dataset title, description, name, vintage, url, dataset type, and other useful fields.

Usage

```
listCensusApis(name = NULL, vintage = NULL)
```

Arguments

`name`	Optional complete or partial API dataset programmatic name. For example, "acs", "acs/acs5", "acs/acs5/subject". If using a partial name, this needs to be the left-most part of the dataset name before `/`, e.g. "timeseries/eits" or "dec" or "acs/acs5".
`vintage`	Optional vintage (year) of dataset.

Value

A data frame with the following columns:

- title: Short written description of the dataset.
- name: Programmatic name of the dataset.
- vintage: Year of the survey, for use with microdata and aggregate datasets.
- type: Dataset type, which is either "Aggregate", "Microdata", or "Timeseries".
- temporal: Time period of the dataset. Warning: not always documented.
- spatial: Spatial region of the dataset. Warning: not always documented.
- url: Base URL of the dataset endpoint.
- modified: Date last modified. Warning: sometimes out of date.
- description: Long written description of the dataset.
- contact: Email address for specific questions about the Census Bureau survey.

See Also

Other metadata: `listCensusMetadata()`, `makeVarlist()`

Examples

```
# Get information about every dataset available in the APIs
apis <- listCensusApis()
head(apis)

# Get information about all vintage 2022 datasets
apis_2022 <- listCensusApis(vintage = 2022)
head(apis_2022)

# Get information about all timeseries datasets
apis_timeseries <- listCensusApis(name = "timeseries")
head(apis_timeseries)

# Get information about 2020 Decennial Census datasets
apis_decennial_2020 <- listCensusApis(name = "dec", vintage = 2020)
head(apis_decennial_2020)

# Get information about one particular dataset
api_sahie <- listCensusApis(name = "timeseries/healthins/sahie")
head(api_sahie)
```

---

listCensusMetadata *Get metadata about a specified API endpoint*

Description

Get information about a Census Bureau API dataset, including its available variables, geographies, variable groups, and value labels

Usage

```
listCensusMetadata(
  name,
  vintage = NULL,
  type = "variables",
  group = NULL,
  variable_name = NULL,
  include_values = FALSE
)
```

Arguments

name	API programmatic name - e.g. acs/acs5. Use `listCensusApis()` to see valid dataset names.
vintage	Vintage (year) of dataset. Not required for timeseries APIs.
type	Type of metadata to return. Options are:

- "variables" (default) - list of variable names and descriptions for the dataset.
- "geographies" - available geographies.
- "groups" - available variable groups. Only available for some datasets.
- "values" - encoded value labels for a given variable. Pair with "variable_name". Only available for some datasets.

`group` An optional variable group code, used to return metadata for a specific group of variables only. Variable groups are not used for all APIs.

`variable_name` A name of a specific variable used to return value labels for that variable. Value labels are not used for all APIs.

`include_values` Use with `type = "variables"`. Include value metadata for all variables in a dataset if value metadata exists. Default is "FALSE".

Value

A data frame with metadata about the specified API endpoint.

See Also

Other metadata: `listCensusApis()`, `makeVarlist()`

Examples

```
# type: variables # List the variables available in the Small Area
# Health Insurance Estimates.
variables <- listCensusMetadata(
  name = "timeseries/healthins/sahie", type = "variables")
head(variables)

# type: variables for a single variable group
# List the variables that are included in the B17020 group in the
# 5-year American Community Survey.
variable_group <- listCensusMetadata(
  name = "acs/acs5", vintage = 2022,
  type = "variables", group = "B17020")
head(variable_group)

# type: variables, with value labels
# Create a data dictionary with all variable names and encoded values
# for a microdata API.
variable_values <- listCensusMetadata(
  name = "cps/voting/nov",
  vintage = 2020,
  type = "variables",
  include_values = TRUE)
head(variable_values)

# type: geographies
# List the geographies available in the 5-year American Community Survey.
geographies <- listCensusMetadata(
  name = "acs/acs5",
```

```
  vintage = 2022,
  type = "geographies")
head(geographies)

# type: groups
# List the variable groups available in the 5-year American
# Community Survey.
groups <- listCensusMetadata(
  name = "acs/acs5",
  vintage = 2022,
  type = "groups")
head(groups)

# type: values for a single variable
# List the value labels of the NAICS2017 variable in the County
# Business Patterns dataset.
naics_values <- listCensusMetadata(
  name = "cbp",
  vintage = 2021,
  type = "values",
  variable = "NAICS2017")
head(naics_values)
```

`makeVarlist` *Use variable metadata to find variables containing a given string*

Description

Return a list of variable names or data frame of variable metadata containing a given string. This can be used create a list of variables to later pass to getCensus, or a data frame documenting variables used in a given project.

Usage

```
makeVarlist(name, vintage = NULL, find, varsearch = "all", output = "list")
```

Arguments

`name`	API programmatic name - e.g. acs/acs5. Use `listCensusApis()` to see valid dataset names.
`vintage`	Vintage (year) of dataset. Not required for timeseries APIs.
`find`	A string to find in the variable metadata.
`varsearch`	Optional argument specifying which fields to search. Default is "all". Options are "all", "name", "label", or "concept".
`output`	Optional argument, specifying output to "list" or "dataframe". Default is "list".

Value

A data frame containing variable metadata

See Also

Other metadata: `listCensusApis()`, `listCensusMetadata()`

Examples

```
# Return a list, and then use getCensus function to retrieve those variables
myvars <- makeVarlist(name = "timeseries/poverty/saipe",
  find = "Ages 0-4",
  varsearch = "label")
myvars
saipe_dt <- getCensus(name = "timeseries/poverty/saipe",
  time = 2016,
  vars = myvars,
  region = "state:*")
head(saipe_dt)
```

Index