Package 'deprivateR'

July 22, 2025

Title Calculating and Analyzing Measures of Deprivation in the United States

Version 0.1.0

Description Provides a unified framework to building Area Deprivation Index (ADI), Social Vulnerability Index (SVI), and Neighborhood Deprivation Index (NDI) deprivation measures and accessing related data from the U.S. Census Bureau such as Gini coefficient data. Tools are also available for calculating percentiles, quantiles, and for creating clear map breaks for data visualization.

Depends R (>= 3.5)

License Apache License (>= 2)

URL <https://pfizer-opensource.github.io/deprivateR/>

BugReports <https://github.com/pfizer-opensource/deprivateR/issues>

Encoding UTF-8

RoxygenNote 7.3.2

Imports classInt, dplyr, english, ndi, rlang, sf, sociome, stringr, tibble, tidycensus, tidyselect, tidyr, zippeR

Suggests measurements, testthat, tigris

NeedsCompilation no

Author Christopher Prener [aut, cre] (ORCID: <https://orcid.org/0000-0002-4310-9888>), Timothy Wiemken [aut] (ORCID: <https://orcid.org/0000-0002-8251-3007>)

Maintainer Christopher Prener <christopher.prener@pfizer.com>

Repository CRAN

Date/Publication 2025-01-29 17:10:02 UTC

Contents

---

dep_build_varlist   *Create Variable Lists*

Description

This function creates a vector or `tibble` containing variables included in particular calls.

Usage

```
dep_build_varlist(geography, index, year, survey = "acs5", output = "vector")
```

Arguments

geography	A character scalar; one of `"state"`, `"county"`, or `"tract"`
index	A character scalar or vector listing deprivation measures to return. These include the area deprivation index (`"adi"`), the gini coefficient (`"gini"`), two versions of the neighborhood deprivation index by Messer (`"ndi_m"`) and Powell and Wiley (`"ndi_pw"`), and four versions of the social vulnerability index (`"svi10"`, `"svi14"`, `"svi20"`, and `"svi20s"`).
year	A numeric scalar between 2010 and 2020
survey	A character scalar representing the Census product. It can be any American Community Survey product (either `"acs1"`, `"acs3"`, or `"acs5"`). Note that `"acs3"` was discontinued after 2013.
output	A character scalar; either `"vector"` (default) or `tibble`. See Return below.

Value

A vector of variable names or a `tibble` containing both variable names, labels, and the measure(s) they are associated with.

Examples

```
# Gini coefficient at the Census tract level
dep_build_varlist(geography = "tract", index = "gini", year = 2019)
```

---

dep_calc_index   *Perform Deprivation Calculations*

Description

Calculates various measures of deprivation on data you have. Data cannot be automatically downloaded with this option, and the output options are more limited. See Details under dep_get_index for more information. For information about structuring your data prior to using this function, see Details below.

Usage

```
dep_calc_index(.data, geography, index, year, survey = "acs5",
    return_percentiles = FALSE, keep_subscales = FALSE, keep_components = FALSE,
    output = "wide")
```

Arguments

.data	A data frame, tibble, or sf object that contains all of the columns needed to calculate your selected deprivation measure(s). See Details below.
geography	A character scalar; one of "county", "zcta3", "zcta5", or "tract"
index	A character scalar or vector listing deprivation measures to return. These include the area deprivation index ("adi"), the gini coefficient ("gini"), two versions of the neighborhood deprivation index by Messer ("ndi_m") and Powell and Wiley ("ndi_pw"), and four versions of the social vulnerability index ("svi10", "svi14", "svi20", and "svi20s"). See Details.
year	A numeric scalar between 2010 and 2022.
survey	A character scalar representing the Census product. It can be any American Community Survey product (either "acs1", "acs3", or "acs5"). Note that "acs3" was discontinued after 2013.
return_percentiles	
	A logical scalar; if TRUE, scales (and their subscales) will be returned as percentiles instead of in raw scores. If FALSE (default), raw scores will be returned. Note that SVI is returned as a percentile regardless of what return_percentiles is set to.
keep_subscales	A logical scalar; if FALSE (default), only the full ADI and/or SVI scores (depending on what is passed to the index argument) will be returned. If TRUE and "svi" is listed for the index argument, the four SVI "themes" (see Details) will be returned along with the full SVI score. Similarly, if "adi" is listed for the index argument, the three ADI subscales (see Details) will be returned.
keep_components	
	A logical scalar; if FALSE (default), none of the components used to calculate the deprivation measures will be returned. If TRUE, all of the demographic variables used to calculate ADI and/or SVI will be returned.

output   A character scalar; if `"wide"` (default), a tibble will be returned with row per
                         jurisdiction where individual measures of deprivation stored in columns. If
                         `"tidy"`, a tibble will be returned with one row for each combination of juris-
                         diction and deprivation measure.

Details

Input data must be "wide" formatted and should have the following columns:

`"GEOID"` The appropriately formatted GEOID values for the geography given in the function. This
         is required.

`"YEAR"` The year that corresponds to the demographic data. For five-year ACS data, this should
         correspond to the final year in the period (e.x. 2021 for the 2017-2021 ACS). This is required
         only if deprivation scores are being generated for more than one year.

Demographic measures All of the necessary columns required for the deprivation scores and years
         given (since the input measures vary between scores and over time for individual scores.)

Value

A tibble object containing the requested deprivation measures.

Examples

```
## load sample data
ndi_m <- dep_sample_data(index = "ndi_m")

## calculate NDI with sample data
ndi_m <- dep_calc_index(ndi_m, geography = "county", index = "ndi_m", year = 2022,
    return_percentiles = TRUE)
```

---

dep_get_index   *Calculate Deprivation Measures*

Description

Downloads raw data and then calculates various measures of deprivation and/or vulnerability, in-
cluding a range of options for structuring output. The included measures include four versions of
the CDC's social vulnerability index, which is a unique offering, along with wrappers that bring
in additional measures from related packages: the area deprivation index (ADI; via `sociome`), gini
coefficient (via `tidycensus`), and the neighborhood deprivation index (NDI; via `ndi`). Both ADI
and NDI contain variations as well. See Details for more information.

Usage

```
dep_get_index(geography, index, year, survey = "acs5",
    return_percentiles = FALSE, keep_subscales = FALSE,
    keep_components = FALSE, output = "wide",
    state = NULL, county = NULL, puerto_rico = FALSE, zcta = NULL,
    zcta_geo_method = NULL, zcta_cb = FALSE, zcta3_method = NULL,
    shift_geo = FALSE, key = NULL)
```

Arguments

geography   A character scalar; one of "county", "zcta3", "zcta5", or "tract"

index   A character scalar or vector listing deprivation measures to return. These include
             the area deprivation index ("adi"), the gini coefficient ("gini"), two versions
             of the neighborhood deprivation index by Messer ("ndi_m") and Powell and
             Wiley ("ndi_pw"), and four versions of the social vulnerability index ("svi10",
             "svi14", "svi20", and "svi20s"). See Details.

year   A numeric scalar or vector. 2010 is earliest year deprivateR supports, while
             2022 is the most recent year.

survey   A character scalar representing the Census product. It can be any American
             Community Survey product (either "acs1", "acs3", or "acs5"). Note that
             "acs3" was discontinued after 2013.

return_percentiles
             A logical scalar; if TRUE, scales (and their subscales) will be returned as per-
             centiles instead of in raw scores. If FALSE (default), raw scores will be returned.
             Note that SVI is returned as a percentile regardless of what return_percentiles
             is set to.

keep_subscales A logical scalar; if FALSE (default), only the full ADI and/or SVI scores (de-
             pending on what is passed to the index argument) will be returned. If TRUE and
             "svi" is listed for the index argument, the four SVI "themes" (see Details) will
             be returned along with the full SVI score. Similarly, if "adi" is listed for the
             index argument, the three ADI subscales (see Details) will be returned.

keep_components
             A logical scalar; if FALSE (default), none of the components used to calculate the
             deprivation measures will be returned. If TRUE, all of the demographic variables
             used to calculate ADI and/or SVI will be returned.

output   A character scalar; if "wide" (default), a tibble will be returned with row per
             jurisdiction where individual measures of deprivation stored in columns. If
             "tidy", a tibble will be returned with one row for each combination of juris-
             diction and deprivation measure. If "sf", a "wide" data set will be returned with
             geometric data appeneded to facilitate mapping and/or spatial statistics.

state   A character scalar or vector with character state abbreviations (e.x. "MO") or
             numeric FIPS codes (e.x. 29).

county   A character scalar or vector with character GEOIDs (e.x. "29510")

puerto_rico A logical scalar; if TRUE (default), data for Puerto Rico will be included in cal-
             culations. If FALSE, Puerto Rico will not be included.

zcta   An optional vector of ZCTAs that demographic data are requested for. If this is
                    NULL and geography = "zcta5", data will be returned for all ZCTAs. If a vector
                    is supplied and geography = "zcta5", only data for those requested ZCTAs will
                    be returned. The vector can be created with zippeR::zi_get_geometry() and
                    should only contain five-digit ZCTAs.

zcta_geo_method

                    A character scalar; if geography = "zcta5" or geography = "zcta3", either
                    "intersect" or "centroid", should be supplied. These two options alter how
                    ZCTA overlap with states or counties is defined. See zippeR::zi_get_geometry()
                    for more information.

zcta_cb   A logical scalar; if FALSE, the most detailed TIGER/Line data will be used for
                    style = "zcta5". If TRUE, a generalized (1:500k) version of the data will be
                    used. The generalized data will download significantly faster, though they show
                    less detail. According to the tigris::zctas() documentation, the download
                    size if TRUE is ~65MB while it is ~500MB if cb = FALSE.

                    This argument does not apply to geography = "zcta3", which only returns gen-
                    eralized data. It only applies if output = "sf".

zcta3_method   A character scalar; if geography = "zcta3", a method for aggregating spatially
                    intensive values should be given; either "mean" or "median". In either case, a
                    weighted approach is used where total population for each five-digit ZCTA is
                    used to calculate individual ZCTAs' weights. For American Community Survey
                    Data, this is applied to the margin of error as well.

shift_geo   A logical scalar; if TRUE, Alaska, Hawaii, and Puerto Rico will be re-positioned
                    so that the lie to the southwest of the continental United States. This defaults to
                    FALSE, and can only be used when states are not listed for the state argument.
                    It only applies if output = "sf".

key   A Census API key, which can be obtained at https://api.census.gov/data/key_signup.html. This can be omitted if tidycensus::census_api_key()
                    has been used to write your key to your .Renviron file. You can check whether
                    an API key has been written to .Renviron by using Sys.getenv("CENSUS_API_KEY").

Details

deprivateR provides a unique implementation of the Centers for Disease Control's Social Vulner-
ability Index at a greater range of years and geographies than the CDC originally supported. Four
versions of the SVI are offered:

"svi10" The CDC's 2010 SVI vintage did not include a measure of civilians with a disability,
        unlike their later vintages. This version can be calculated using deprivateR for each year
        from 2010 through 2021.

"svi14" The CDC's 2014, 2016, and 2018 vintages added the measure of civilians with a disability
        to their SVI calculations. The disability measure was added to the American Community
        Survey beginning in 2012, so this version can be calculated using deprivateR for each year
        from 2012 through 2021.

"svi20" The CDC's 2020 vintage made multiple substantive changes to how SVI is calculated
        that changed the underlying data used for the first three of the four themes. In the SES theme:

(1) per capita income was replaced with a measure of housing burden; (2) poverty was converted to 150 insurance. The Household Composition & Disability (HCD) theme was renamed Household Characteristics (HOU), and the English language proficiency measure was moved here from the former Minority Status and Language (MSL) theme. Since the English language measure was removed from MSL theme, it was renamed Racial & Ethnic Minority Status (REM). Though the CDC released this definition with their 2020 data, the underlying data can be accessed from the American Community Survey from 2012 onward. This means that this version can be calculated using deprivateR for each year from 2012 through 2021.

"svi20s" The CDC's 2020 vintage changed the variables used to calculate the number of single-parent households. Their new approach does not have the backward compatibility that the other changes made in 2020 do. This version of SVI uses the same underlying data for single-parent households that the CDC's 2020 vintage does, along with the other changes made in 2020. This version can be calculated using deprivateR for each year from 2012 through 2019.

In addition, wrappers to the sociome, ndi, and tidycensus package create a single point of departure for comparative work using multiple measures of deprivation or inequality.

Value

A tibble with the requested deprivation measures. The number of columns and rows depends upon the input arguments. If output = "wide", the number of columns will be equal to the number of deprivation measures requested plus the number of columns needed to store the geographic information. Each unique combination of jurisdiction and year will receive its own row.

If output = "tidy", the number of columns will be equal to the number of deprivation measures requested plus the number of columns needed to store the geographic information. Each unique combination of jurisdiction and year will receive its own row. Each unique combination of jurisdiction, year, and deprivation measure will receive its own row.

Examples

```
# calculate ADI for all US counties
dep_get_index(geography = "county", index = "adi", year = 2022)

# calculate two forms of SVI for all Missouri ZCTAs
dep_get_index(geography = "zcta5", index = c("svi20", "svi20s"), year = 2022,
  state = "MO")

# calculate ADI and two forms of NDI for all US counties over three years
# percentiles are returned to ease comparison
dep_get_index(geography = "county", index = c("adi", "svi14"),
  year = c(2018:2020), return_percentiles = TRUE)
```

---

dep_map_breaks   *Calculating Map Breaks*

Description

Create "bins" for choropleth maps creating using either `ggplot2` or `leaflet`. The function can create the bins automatically or will accept pre-specified breaks.

Usage

```
dep_map_breaks(.data, var, new_var, classes, style, breaks,
    sig_digits = 2, return = "col", show_warnings = TRUE)
```

Arguments

`.data`	A data object, either sf, tibble, or data.frame
`var`	Variable breaks should be based on, can be quoted or unquoted
`new_var`	Optional name of new variable to store breaks in, can be quoted or unquoted. This is required if you are returning a column, but can be omitted if you are returning breaks instead of a column.
`classes`	Optional integer scalar; count of the number of classes to create. If you are supplying breaks manually, this can be omitted.
`style`	String scalar; one of the classes supported by `classInt::classIntervals()`. "jenks" is the ArcGIS default. "quantile" and "fisher" are better alternatives. As with `classes`, this can be omitted if you are supplying breaks manually.
`breaks`	Optional numeric vector if you want to pre-specify the cut points for your breaks. Provide the lower and upper bounds of your distribution. Any values supplied in between the bounds will be the upper bound of individual bins.
`sig_digits`	Integer; how many significant digits should be applied when calculating breaks and constructing labels?
`return`	String scalar; one of either `"col"` (default) or `"breaks"`. The default behavior adds a factor containing the bins to the data object to facilitate mapping. Specifying `"breaks"` will return the calculated breaks instead, which can be modified and passed to the `breaks` argument later in a script to make the final map.
`show_warnings`	Logical scalar; if TRUE, warnings created by `classInt::classIntervals()` if NA values are identified while findings classes.

Value

Either a data object (if `return` is `"col"`) or a vector of breaks (if `return` is `"breaks"`). If a data object is returned, the new column will be placed directly after the input variable specified in `var`.

Examples

```
# prep data
## load sample data
ndi_m <- dep_sample_data(index = "ndi_m")

## calculate NDI with sample data
ndi_m <- dep_calc_index(ndi_m, geography = "county", index = "ndi_m", year = 2022,
  return_percentiles = TRUE)

# calculate breaks using a built-in algorithm
dep_map_breaks(ndi_m, var = "NDI_M", new_var = "map_breaks", classes = 5,
  style = "fisher")

# use manually specified breaks
## set breaks
breaks <- c(0, 25, 50, 75, max(ndi_m$NDI_M))

## calculate breaks
dep_map_breaks(ndi_m, var = "NDI_M", new_var = "map_breaks", breaks = breaks)
```

---

dep_percentiles   *Calculate Percentiles*

Description

Calculate percentiles for a given variable in a data frame. This is the method used to calculate ranked percentiles for SVI.

Usage

```
dep_percentiles(.data, source_var, new_var)
```

Arguments

.data	A tibble containing the data to be used for calculating percentiles.
source_var	Required; the quoted or unquoted source variable to be divided into percentiles.
new_var	Required; the quoted or unquoted name of the new variable to be created containing the quantile values.

Value

An updated tibble with the percentiles added as a new column or with replaced values in the source column.

Examples

```
## load sample data
ndi_m <- dep_sample_data(index = "ndi_m")

# calculate percentiles for population 25 years and older
ndi_m <- dep_percentiles(ndi_m, source_var = B06009_001E,
    new_var = pop25_percentile)

# preview the new data
ndi_m[names(ndi_m) %in% c("GEOID", "B06009_001E", "pop25_percentile")]
```

---

dep_quantiles   *Return Quantiles of a Variable*

Description

This helper function can be used to return quantiles of a deprivation index (or any other continuous distribution). This is useful for constructing independent variables for statistical analysis. The function supports splitting a distribution at the median (2 quantiles) through deciles (10 quantiles) if character labels are desired.

Usage

```
dep_quantiles(.data, source_var, new_var, n = 4L, return = "label")
```

Arguments

.data	A tibble containing the data to be used for calculating quantiles.
source_var	Required; the quoted or unquoted source variable to be divided into quantiles.
new_var	Required; the quoted or unquoted name of the new variable to be created containing the quantile values.
n	Required integer scalar; the number of quantiles to divide the source variable into. Defaults to 4L (quartiles), but can be set to any value appropriate for your data as long as it is greater than or equal to 2L.
return	Required character scalar; one of either "label" (default) or "factor". If "label", the function will return a character vector of quantile labels. If "factor", the function will return the underlying factor used in the creation of the quantiles measure.

Value

A copy of .data with a new variable containing the requested quantile.

Examples

```
## load sample data
ndi_m <- dep_sample_data(index = "ndi_m")

## calculate NDI with sample data
ndi_m <- dep_calc_index(ndi_m, geography = "county", index = "ndi_m", year = 2022,
    return_percentiles = TRUE)

## calculate quantiles, return label
ndi_m <- dep_quantiles(ndi_m, source_var = NDI_M, new_var = ndi_m_quartiles_l)

unique(sort(ndi_m$ndi_m_quartiles_l))

## calculate quantiles, return label
ndi_m <- dep_quantiles(ndi_m, source_var = NDI_M, new_var = ndi_m_quartiles_l6,
                       n = 6L)

unique(sort(ndi_m$ndi_m_quartiles_l6))

## calculate quantiles, return factor
ndi_m <- dep_quantiles(ndi_m, source_var = NDI_M, new_var = ndi_m_quartiles_f,
    return = "factor")

levels(ndi_m$ndi_m_quartiles_f)
```

---

dep_sample_data   *Create Sample Data*

Description

Create sample data for testing the package functionality.

Usage

```
dep_sample_data(index)
```

Arguments

index
:   A character scalar or vector listing deprivation measures to return. These include the area deprivation index ("adi"), the gini coefficient ("gini"), two versions of the neighborhood deprivation index by Messer ("ndi_m") and Powell and Wiley ("ndi_pw"), and four versions of the social vulnerability index ("svi10", "svi14", "svi20", and "svi20s").

Value

A tibble containing the raw 2022 American Community Sruvey data for the given index. Each tibble will contain observations for the 115 counties in Missouri.

Examples

```
## load sample data
dep_sample_data(index = "ndi_m")
```

Index