Package ‘geobr’

August 16, 2022

Type Package
Title Download Official Spatial Data Sets of Brazil
Version 1.7.0
Date 2022-08-15
URL https://github.com/ipeaGIT/geobr
BugReports https://github.com/ipeaGIT/geobr/issues

Description Easy access to official spatial data sets of Brazil as ‘sf’ objects in R. The package includes a wide range of geospatial data available at various geographic scales and for various years with harmonized attributes, projection and fixed topology.

License MIT + file LICENSE
Encoding UTF-8
LazyData TRUE

Depends R (>= 3.5.0)
Suggests covr, dplyr (>= 0.8-3), ggplot2 (>= 3.3.1), knitr, rmarkdown (>= 2.6), testthat
Imports curl, data.table, httr (>= 1.4.1), methods, sf (>= 0.9-3), utils

RoxygenNote 7.2.1
VignetteBuilder knitr

NeedsCompilation no

Author Rafael H. M. Pereira [aut, cre]
Caio Nogueira Goncalves [aut],
Paulo Henrique Fernandes de Araujo [ctb],
Guilherme Duarte Carvalho [ctb],
Rodrigo Almeida de Arruda [ctb],
Igor Nascimento [ctb],
Barbara Santiago Pedreira da Costa [ctb],
Welligton Silva Cavedo [ctb],

1
R topics documented:

Pedro R. Andrade [ctb],
Alan da Silva [ctb],
Carlos Kauê Vieira Braga [ctb],
Carl Schmertmann [ctb],
Alessandro Samuel-Rosa [ctb],
Daniel Ferreira [ctb],
Marcus Saraiva [ctb],
Ipea - Institute for Applied Economic Research [cph, fnd]

Maintainer Rafael H. M. Pereira <rafa.pereira.br@gmail.com>

Repository CRAN

Date/Publication 2022-08-16 07:50:02 UTC

R topics documented:

cep_to_state .................................................. 3
geobr .......................................................... 3
grid_state_correspondence_table ......................... 4
list_geobr ...................................................... 4
lookup_muni .................................................... 5
read_amazon .................................................... 6
read_biomes .................................................... 7
read_census_tract ............................................. 8
read_comparable_areas ...................................... 9
read_conservation_units .................................. 10
read_country .................................................. 11
read_disaster_risk_area ................................... 12
read_health_facilities .................................... 13
read_health_region ......................................... 14
read_immediate_region ..................................... 15
read_indigenous_land ....................................... 16
read_intermediate_region ................................ 17
read_meso_region ........................................... 18
readMetro_area .............................................. 19
read_micro_region ......................................... 20
read_municipality .......................................... 21
read_municipal_seat ....................................... 22
read_neighborhood ......................................... 23
read_pop_arrangements ................................... 24
read_region ................................................... 25
read_schools .................................................. 26
read_semiarid ............................................... 26
read_state .................................................... 27
read_statistical_grid ...................................... 28
read_urban_area ............................................. 29
read_urban_concentrations ................................ 30
read_weighting_area ....................................... 31
cep_to_state

Index

cep_to_state Determine the state of a given CEP postal code

Description

Zips codes in Brazil are known as CEP, the abbreviation for postal code address. CEPs in Brazil are 8 digits long, with the format 'xxxxx-xxx'.

Usage

cep_to_state(cep)

Arguments

cep A numeric string with 8 digits in the format xxxxxxxx, or a character with the format 'xxxxx-xxx'.

Value

A character string with a state abbreviation

Examples

uf <- cep_to_state(cep = '69900-000')

# Or:
uf <- cep_to_state(cep = 69900000)

geobr geobr package

Description

Easy access to shapefiles of the Brazilian Institute of Geography and Statistics (IBGE) and other official spatial data sets of Brazil

Details

See the README on GitHub
grid_state_correspondence_table

A correspondence table indicating what quadrants of IBGE’s statistical grid intersect with each Brazilian state

Description

Built-in dataset

- name_state: Title-case name of state (character)
- abbrev_state: Two-letter uppercase abbreviation of a state
- code_grid: Unique code of each quadrant of IBGE’s statistical grid

Usage

data(grid_state_correspondence_table)

Format

A data frame sf with 139 rows and 3 columns

Details

correspondence table indicating what quadrants of IBGE’s statistical grid intersect with each Brazilian state

Note

Last updated 2021-03-21

list_geobr

List all data sets available in the geobr package

Description

Returns a data frame with all datasets available in the geobr package

Usage

list_geobr()

Value

A data.frame
**lookup_muni**

**Examples**

```r
df <- list_geobr()
```

---

**lookup_muni**  
*Look up municipality codes and names*

**Description**

Input a municipality **name or code** and get the names and codes of the municipality’s corresponding state, meso, micro, intermediate, and immediate regions.

**Usage**

```r
lookup_muni(name_muni = NULL, code_muni = NULL)
```

**Arguments**

- `name_muni` The municipality name to be looked up
- `code_muni` The municipality code to be looked up

**Details**

Only available from 2010 Census data so far

**Value**

A `data.frame` with 13 columns identifying the geographies information of that municipality

A `data.frame`

**Examples**

```r
## Not run: if (interactive()) {
# Get lookup table for municipality Rio de Janeiro
mun <- lookup_muni(name_muni = "Rio de Janeiro")

# Or you can get a lookup table for the same municipality searching for its code
mun <- lookup_muni(code_muni = 3304557)

# Get lookup table for all municipalities
mun_all <- lookup_muni(name_muni = "all")

# Or:
mun_all <- lookup_muni(code_muni = "all")
}
## End(Not run)
```
Description

This data set covers the whole of Brazil’s Legal Amazon as defined in the federal law n. 12.651/2012). The original data comes from the Brazilian Ministry of Environment (MMA) and can be found at "http://mapas.mma.gov.br/i3geo/datadownload.htm".

Usage

read_amazon(year = 2012, simplified = TRUE, showProgress = TRUE)

Arguments

year  
A date number in YYYY format. Defaults to 2012

simplified  
Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress  
Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: read_biomes(), read_census_tract(), read_comparable_areas(), read_conservation_units(), read_country(), read_health_region(), read Immediate_region(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()
**Description**

This data set includes polygons of all biomes present in Brazilian territory and coastal area. The latest data set dates to 2019 and it is available at scale 1:250,000. The 2004 data set is at the scale 1:5,000,000. The original data comes from IBGE. More information at [https://www.ibge.gov.br/apps/biomas/](https://www.ibge.gov.br/apps/biomas/)

**Usage**

```r
read_biomes(year = 2019, simplified = TRUE, showProgress = TRUE)
```

**Arguments**

- `year` A date number in YYYY format. Defaults to 2019
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other general area functions: `read_amazon()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`

**Examples**

```r
## Not run: if (interactive()) {
# Read biomes
b <- read_biomes(year = 2019)
}
## End(Not run)
```
read_census_tract  
*Description*

Download spatial data of census tracts of the Brazilian Population Census

*Usage*

```r
read_census_tract(
  code_tract,
  year = 2010,
  zone = "urban",
  simplified = TRUE,
  showProgress = TRUE
)
```

*Arguments*

- **code_tract**: The 7-digit code of a Municipality. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all census tracts of that state. If `code_tract="all"`, all census tracts of the country are loaded.
- **year**: Year of the data. Defaults to 2010
- **zone**: For census tracts before 2010, 'urban' and 'rural' census tracts are separate data sets.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

*Value*

An "sf" "data.frame" object

*See Also*

Other general area functions: `read_amazon()`, `read_biomes()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`
Examples

```
## Not run: if (interactive()) {
# Read rural census tracts for years before 2007
  c <- read_census_tract(code_tract=5201108, year=2000, zone="rural")

# Read all census tracts of a state at a given year
  c <- read_census_tract(code_tract=53, year=2010) # or
  c <- read_census_tract(code_tract="DF", year=2010)
  plot(c)

# Read all census tracts of a municipality at a given year
  c <- read_census_tract(code_tract=5201108, year=2010)
  plot(c)

# Read all census tracts of the country at a given year
  c <- read_census_tract(code_tract="all", year=2010)
}
## End(Not run)
```

---

**read_comparable_areas**

*Download spatial data of historically comparable municipalities*

**Description**

This function downloads the shape file of minimum comparable area of municipalities, known in Portuguese as 'Areas minimas comparaveis (AMCs)'. The data is available for any combination of census years between 1872-2010. These data sets are generated based on the Stata code originally developed by doi: 10.1590/0101416147182phePhilipp Ehrl, and translated into \texttt{R} by the \texttt{geobr} team.

**Usage**

```
read_comparable_areas(
  start_year = 1970,
  end_year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```

**Arguments**

- **start_year**: Numeric. Start year to the period.
- **end_year**: Numeric. End year to the period. (defaults to 2010)
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using \texttt{st_simplify(sf)} preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.

Details
These data sets are generated based on the original Stata code developed by Philipp Ehrl. If you use these data, please cite:


Value
An "sf" "data.frame" object

See Also
Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_conservation_units(), read_country(), read_health_region(), read_immediate_region(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()

Examples
```r
## Not run: if (interactive()) {

  amc <- read_comparable_areas(start_year=1970, end_year=2010)

} # End(Not run)
```

---

**read_conservation_units**

*Download spatial data of Brazilian environmental conservation units*

**Description**

This data set covers the whole of Brazil and it includes the polygons of all conservation units present in Brazilian territory. The last update of the data was 09-2019. The original data comes from MMA and can be found at "http://mapas.mma.gov.br/i3geo/datadownload.htm".

**Usage**

```r
read_conservation_units(date = 201909, simplified = TRUE, showProgress = TRUE)
```
Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674).

Usage

```r
read_country(year = 2010, simplified = TRUE, showProgress = TRUE)
```

Arguments

- **year**: Year of the data. Defaults to 2010
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`
Value

An "sf" "data.frame" object

See Also

Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), read_conservation_units(), read_health_region(), read_immediate_region(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()

Examples

## Not run: if (interactive()) {
# Read specific year
br <- read_country(year = 2018)
}
## End(Not run)

read_disaster_risk_area

Download spatial data of disaster risk areas

Description

This function reads the official data of disaster risk areas in Brazil. It specifically focuses on geodynamic and hydro-meteorological disasters capable of triggering landslides and floods. The data set covers the whole country. Each risk area polygon (known as 'BATER') has unique code id (column 'geo_bater'). The data set brings information on the extent to which the risk area polygons overlap with census tracts and block faces (column "acuracia") and number of risk areas within each risk area (column 'num'). Original data were generated by IBGE and CE-MADEN. For more information about the methodology, see details at https://www.ibge.gov.br/geociencias/organizacao-do-territorio/tipologias-do-territorio/21538-populacao-em-areas-de-risco-no-brasil.html

Usage

read_disaster_risk_area(year = 2010, simplified = TRUE, showProgress = TRUE)

Arguments

year
A year number in YYYY format. Defaults to 2010

simplified
Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress
Logical. Defaults to TRUE display progress bar.
read_health_facilities

Value
An "sf" "data.frame" object

Examples

```r
## Not run: if (interactive()) {
 # Read all disaster risk areas in an specific year
 d <- read_disaster_risk_area(year=2010)
}
## End(Not run)
```

---

**Description**

Data comes from the National Registry of Healthcare facilities (Cadastro Nacional de Estabelecimentos de Saude - CNES), originally collected by the Brazilian Ministry of Health. According to the Ministry of Health: "The coordinates of each facility were obtained by CNES and validated by means of space operations. These operations verify if the point is in the municipality, considering a radius of 5,000 meters. When the coordinate is not correct, further searches are done in other systems of the Ministry of Health and in web services like Google Maps. Finally, if the coordinates have been correctly obtained in this process, the coordinates of the municipal head office are used. The geocode source used is registered in the database in a specific column data_source. Periodically the coordinates are revised with the objective of improving the quality of the data." The date of the last data update is registered in the database in the columns date_update and year_update. More information available at [https://dados.gov.br/dataset?q=CNES](https://dados.gov.br/dataset?q=CNES). These data use Geodetic reference system "SIRGAS2000" and CRS(4674).

**Usage**

```r
read_health_facilities(showProgress = TRUE)
```

**Arguments**

- `showProgress` Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**Examples**

```r
## Not run: if (interactive()) {
 # Read all health facilities of the whole country
 h <- read_health_facilities()
}
## End(Not run)
```
read_health_region  

**Description**

Health regions are used to guide the regional and state planning of health services. Macro health regions, in particular, are used to guide the planning of high complexity health services. These services involve larger economics of scale and are concentrated in few municipalities because they are generally more technology intensive, costly and face shortages of specialized professionals. A macro region comprises one or more health regions.

**Usage**

```r
defaulted = 2013,
    macro = FALSE,
    simplified = TRUE,
    showProgress = TRUE
)
```

**Arguments**

- `year`  
  Year of the data. Defaults to 2013, latest available.

- `macro`  
  Logic. If FALSE (default), the function downloads health regions data. If TRUE, the function downloads macro regions data.

- `simplified`  
  Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.

- `showProgress`  
  Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_semiclairid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`
Examples

```r
## Not run: if (interactive()) {
# Read all health regions for a given year
hr <- read_health_region( year=2013 )

# Read all macro health regions
mhr <- read_health_region( year=2013, macro =TRUE)
}
## End(Not run)
```

Description

The Immediate Geographic Areas are part of the geographic division of Brazil created in 2017 by IBGE. These regions were created to replace the "Micro Regions" division. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_immediate_region(
  code_immediate = "all",
  year = 2019,
  simplified = TRUE,
  showProgress = TRUE
)
```

Arguments

- `code_immediate` 6-digit code of an immediate region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all immediate regions of that state. If `code_immediate="all"` (Default), all immediate regions of the country are loaded.
- `year` A date number in YYYY format (defaults to 2019)
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object
read_indigenous_land

Download spatial data of indigenous lands in Brazil

Description

The data set covers the whole of Brazil and it includes indigenous lands from all ethnicities and in different stages of demarcation. The original data comes from the National Indian Foundation (FUNAI) and can be found at http://www.funai.gov.br/index.php/shape. Although original data is updated monthly, the geobr package will only keep the data for a few months per year.

Usage

read_indigenous_land(date = 201907, simplified = TRUE, showProgress = TRUE)

Arguments

date A date number in YYYYMM format. Defaults to 201907

simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress Logical. Defaults to TRUE display progress bar.
read_intermediate_region

Value

An "sf" "data.frame" object

Examples

## Not run: if (interactive()) {
  # Read all indigenous land in an specific date
  i <- read_indigenous_land(date=201907)
}
## End(Not run)

read_intermediate_region

Download spatial data of Brazil’s Intermediate Geographic Areas

Description

The intermediate Geographic Areas are part of the geographic division of Brazil created in 2017 by IBGE. These regions were created to replace the "Meso Regions" division. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

read_intermediate_region(
  code_intermediate = "all",
  year = 2019,
  simplified = TRUE,
  showProgress = TRUE
)

Arguments

code_intermediate

4-digit code of an intermediate region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all intermediate regions of that state. If code_intermediate="all" (Default), all intermediate regions of the country are loaded.

year

A year number in YYYY format. Defaults to 2019

simplified

Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress

Logical. Defaults to TRUE display progress bar.
Value
An "sf" "data.frame" object

See Also
Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), read_conservations_units(), read_country(), read_health_region(), read_immediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()

Examples
## Not run: if (interactive()) {
  # Read an specific intermediate region
  im <- read_intermediate_region(code_intermediate=1202)

  # Read intermediate regions of a state
  im <- read_intermediate_region(code_intermediate=12)
  im <- read_intermediate_region(code_intermediate="AM")

  # Read all intermediate regions of the country
  im <- read_intermediate_region()
  im <- read_intermediate_region(code_intermediate="all")
}
## End(Not run)

read_meso_region

Download spatial data of meso regions

Description
Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage
read_meso_region(
  code_meso = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)

Arguments
code_meso The 4-digit code of a meso region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all meso regions of that state. If code_meso="all", all meso regions of the country are loaded.
The function returns the shapes of municipalities grouped by their respective metro areas. Metropolitan areas are created by each state in Brazil. The data set includes the municipalities that belong to all metropolitan areas in the country according to state legislation in each year. Original data were generated by Institute of Geography. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674).

Usage

read_metro_area(year = 2018, simplified = TRUE, showProgress = TRUE)
Arguments

- **year**: A year number in YYYY format. Defaults to 2018
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

Examples

```r
## Not run: if (interactive()) {
# Read all official metropolitan areas for a given year
m <- read_metro_area(2005)

m <- read_metro_area(2018)
}
## End(Not run)
```

Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_micro_region(
  code_micro = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```

Arguments

- **code_micro**: 5-digit code of a micro region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all micro regions of that state. If code_micro="all", all micro regions of the country are loaded.
- **year**: Year of the data. Defaults to 2010.
read_municipality

```
simplified  Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object

See Also
Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), read_conservation_units(), read_country(), read_health_region(), read_immediate_region(), read_intermediate_region(), read_meso_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()

Examples
```## Not run: if (interactive()) {
  # Read an specific micro region a given year
  micro <- read_micro_region(code_micro=11008, year=2018)

  # Read micro regions of a state at a given year
  micro <- read_micro_region(code_micro=12, year=2017)
  micro <- read_micro_region(code_micro="AM", year=2000)

  # Read all micro regions at a given year
  micro <- read_micro_region(code_micro="all", year=2010)
}## End(Not run)
```
Arguments

- **code_muni**: The 7-digit identification code of a municipality. If `code_muni = "all"` (default), all municipalities of the country will be downloaded. Alternatively, if the two-digit identification code or a two-letter uppercase abbreviation of a state is passed, e.g. 33 or "RJ", all municipalities of that state will be downloaded. Municipality identification codes are defined in [https://www.ibge.gov.br/explica/codigos-dos-municipios.php](https://www.ibge.gov.br/explica/codigos-dos-municipios.php).

- **year**: Year of the data. Defaults to 2010.

- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.

- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_neighborhood()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`

Examples

```r
## Not run: if (interactive()) {
# Read specific municipality at a given year
mun <- read_municipality(code_muni = 1200179, year = 2017)

# Read all municipalities of a state at a given year
mun <- read_municipality(code_muni = 33, year = 2010)
mun <- read_municipality(code_muni = "RJ", year = 2010)

# Read all municipalities of the country at a given year
mun <- read_municipality(code_muni = "all", year = 2018)
}
## End(Not run)
```

---

**read_municipal_seat**  
*Download spatial data of municipal seats (sede dos municipios) in Brazil*
Description

This function reads the official data on the municipal seats (sede dos municipios) of Brazil. The data brings the geographical coordinates (lat lon) of municipal seats for various years between 1872 and 2010. Original data were generated by Brazilian Institute of Geography and Statistics (IBGE).

Usage

read_municipal_seat(year = 2010, showProgress = TRUE)

Arguments

year A year number in YYYY format. Defaults to 2010
showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

Examples

## Not run: if (interactive()) {
# Read municipal seats in an specific year
m <- read_municipal_seat(year = 1991)
}
## End(Not run)

Description

This data set includes the neighborhood limits of 720 Brazilian municipalities. It is based on aggregations of the census tracts from the Brazilian census. Only 2010 data is currently available.

Usage

read_neighborhood(year = 2010, simplified = TRUE, showProgress = TRUE)

Arguments

year Year of the data. Defaults to 2010
simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.
read_pop_arrangements

Value
An "sf" "data.frame" object

See Also
Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), read_conservaation_units(), read_country(), read_health_region(), read_immediate_region(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_region(), read_semiarid(), read_state(), read_statistical_grid(), read_weighting_area()

Examples
## Not run: if (interactive()) {
  # Read neighborhoods of Brazilian municipalities
  n <- read_neighborhood(year=2010)
}  
## End(Not run)

read_pop_arrangements  Download population arrangements in Brazil

Description
This function reads the official data on population arrangements (Arranjos Populacionais) of Brazil. Original data were generated by the Institute of Geography and Statistics (IBGE) For more information about the methodology, see details at https://www.ibge.gov.br/apps/arranjos_populacionais/2015/pdf/publicacao.pdf

Usage
read_pop_arrangements(year = 2015, simplified = TRUE, showProgress = TRUE)

Arguments
year A year number in YYYY format. Defaults to 2015
simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object
Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

`read_region(year = 2010, simplified = TRUE, showProgress = TRUE)`

Arguments

- **year**: Year of the data (defaults to 2010)
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`

Examples

```r
## Not run: if (interactive()) {
# Read specific year
reg <- read_region(year=2018)
}
## End(Not run)
```
Description

Data comes from the School Census collected by INEP, the National Institute for Educational Studies and Research Anisio Teixeira. The date of the last data update is registered in the database in the column ‘date_update’. These data uses Geodetic reference system ”SIRGAS2000” and CRS(4674). The coordinates of each school if collected by INEP. Periodically the coordinates are revised with the objective of improving the quality of the data. More information available at https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/inep-data/catalogo-de-escolas/

Usage

read_schools(year = 2020, showProgress = TRUE)

Arguments

year A year number in YYYY format. Defaults to 2020

showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

Examples

## Not run: if (interactive()) {
# Read all schools in the country
s <- read_schools( year = 2020)
}
## End(Not run)

Description

This data set covers the whole of Brazilian Semiarid as defined in the resolution in 23/11/2017). The original data comes from the Brazilian Institute of Geography and Statistics (IBGE) and can be found at https://www.ibge.gov.br/geociencias/cartas-e-mapas/mapas-regionais/15974-semiarido-brasileiro.html?&t=downloads

Usage

read_semiarid(year = 2017, simplified = TRUE, showProgress = TRUE)
read_state

Arguments

- **year**: A date number in YYYY format (defaults to 2017)
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_state()`, `read_statistical_grid()`, `read_weighting_area()`

Examples

```r
## Not run: if (interactive()) {
  # Read Brazilian semiarid
  a <- read_semiarid(year=2017)
} # End(Not run)
```

Description

Download spatial data of Brazilian states

Usage

```r
read_state(
  code_state = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```
Arguments

code_state  The two-digit code of a state or a two-letter uppercase abbreviation (e.g. 33 or "RJ"). If code_state="all", all states will be loaded.

year  Year of the data. Defaults to 2010

simplified  Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress  Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), read_conservation_units(), read_country(), read_health_region(), read Immediate_region(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), read_neighborhood(), read_region(), read_semiarid(), read_statistical_grid(), read_weighting_area()

Examples

if (interactive()) {
  # Read specific state at a given year
  uf <- read_state(code_state=12, year=2017)

  # Read specific state at a given year
  uf <- read_state(code_state="SC", year=2000)

  # Read all states at a given year
  ufs <- read_state(code_state="all", year=2010)
}

---

read_statistical_grid  Download spatial data of IBGE's statistical grid

Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

read_statistical_grid(code_grid, year = 2010, showProgress = TRUE)
**read_urban_area**

Download spatial data of urbanized areas in Brazil

**Description**

This function reads the official data on the urban footprint of Brazilian cities in the years 2005 and 2015. Original data were generated by the Institute of Geography and Statistics (IBGE) For more information about the methodology, see details at [https://biblioteca.ibge.gov.br/visualizacao/livros/liv100639.pdf](https://biblioteca.ibge.gov.br/visualizacao/livros/liv100639.pdf)

**Usage**

```r
read_urban_area(year = 2015, simplified = TRUE, showProgress = TRUE)
```

**Arguments**

- `code_grid` If two-letter abbreviation or two-digit code of a state is passed, the function will load all grid quadrants that intersect with that state. If `code_grid = "all"`, the grid of the whole country will be loaded. Users may also pass a grid quadrant id to load an specific quadrant. Quadrant ids can be consulted at `geobr::grid_state_correspondence_table`.
- `year` Year of the data. Defaults to 2010. The only year available thus far is 2010.
- `showProgress` Logical. Defaults to `TRUE` display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other general area functions: `read_amazon()`, `read_biomes()`, `read_census_tract()`, `read_comparable_areas()`, `read_conservation_units()`, `read_country()`, `read_health_region()`, `read_immediate_region()`, `read_intermediate_region()`, `read_meso_region()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_weighting_area()`

**Examples**

```r
## Not run: if (interactive()) {
  # Read specific municipality at a given year
  grid <- read_statistical_grid(code_grid = 45, year=2010)
  # Read all municipalities of a state at a given year
  state_grid <- read_statistical_grid(code_grid = "RJ")
} ## End(Not run)
```
Arguments

year       A year number in YYYY format. Defaults to 2015
simplified Logic FALSE or TRUE, indicating whether the function should return the data
set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

Examples

```r
## Not run: if (interactive()) {
  # Read urban footprint of Brazilian cities in an specific year
  d <- read_urban_area(year=2005)
}  
## End(Not run)
```

Description

This function reads the official data on the urban concentration areas (Areas de Concentracao de Populacao) of Brazil. Original data were generated by the Institute of Geography and Statistics (IBGE) For more information about the methodology, see details at https://www.ibge.gov.br/apps/arranjos_populacionais/2015/pdf/publicacao.pdf

Usage

```r
read_urban_concentrations(year = 2015, simplified = TRUE, showProgress = TRUE)
```

Arguments

year       A year number in YYYY format. Defaults to 2015
simplified Logic FALSE or TRUE, indicating whether the function should return the data
set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.
read_weighting_area

Value

An "sf" "data.frame" object

Examples

## Not run: if (interactive()) {
# Read urban footprint of Brazilian cities in an specific year
uc <- read_urban_concentrations(year=2015)
}
## End(Not run)

read_weighting_area

Download spatial data of Census Weighting Areas (area de ponderação) of the Brazilian Population Census

Description

Only 2010 data is currently available.

Usage

read_weighting_area(
  code_weighting = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)

Arguments

code_weighting
  The 7-digit code of a Municipality. If the two-digit code or a two-letter upper-case abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all weighting areas of that state. If code_weighting="all", all weighting areas of the country are loaded.

year
  Year of the data. Defaults to 2010

simplified
  Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress
  Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object
See Also

Other general area functions: read_amazon(), read_biomes(), read_census_tract(), read_comparable_areas(), 
read_conservation_units(), read_country(), read_health_region(), read_immediate_region(), 
read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipality(), 
read_neighborhood(), read_region(), read_semiarid(), read_state(), read_statistical_grid()

Examples

## Not run: if (interactive()) {
# Read specific weighting area at a given year
w <- read_weighting_area(code_weighting=5201108005004, year=2010)

# Read all weighting areas of a state at a given year
w <- read_weighting_area(code_weighting=53, year=2010) # or
w <- read_weighting_area(code_weighting="DF", year=2010)
plot(w)

# Read all weighting areas of a municipality at a given year
w <- read_weighting_area(code_weighting=5201108, year=2010)
plot(w)

# Read all weighting areas of the country at a given year
w <- read_weighting_area(code_weighting="all", year=2010)
}
## End(Not run)
Index

* datasets
  grid_state_correspondence_table, 4
* general area functions
  read_amazon, 6
  read_biomes, 7
  read_census_tract, 8
  read_comparable_areas, 9
  read_conservation_units, 10
  read_country, 11
  read_health_region, 14
  read_immediate_region, 15
  read_intermediate_region, 17
  read_meso_region, 18
  read_micro_region, 20
  read_municipality, 21
  read_neighborhood, 23
  read_region, 25
  read_semiarid, 26
  read_state, 27
  read_statistical_grid, 28
  read_weighting_area, 31
* general support functions
  list_geобр, 4
  cep_to_state, 3
  geобр, 3
  grid_state_correspondence_table, 4
  list_geобр, 4
  lookup_muni, 5
  read_amazon, 6, 7, 8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_biomes, 6, 7, 8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_census_tract, 6, 7, 8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_comparable_areas, 6–8, 9, 11, 12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_conservation_units, 6–8, 10, 10, 12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_country, 6–8, 10, 11, 11, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_disaster_risk_area, 12
  read_health_facilities, 13
  read_health_region, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_immediate_region, 6–8, 10–12, 14, 15, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_indigenous_land, 16
  read_intermediate_region, 6–8, 10–12, 14, 16, 17, 19, 21, 22, 24, 25, 27–29, 32
  read_meso_region, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_metro_area, 19
  read_micro_region, 6–8, 10–12, 14, 16, 18, 19, 20, 22, 24, 25, 27–29, 32
  read_municipal_seat, 22
  read_municipality, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_neighborhood, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_pop_arrangements, 24
  read_region, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 32
  read_schools, 26
  read_semiarid, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 26, 28, 29, 32
  read_state, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27, 29, 32
  read_statistical_grid, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27, 28, 29, 32
  read_urban_area, 29
  read_urban_concentrations, 30
  read_weighting_area, 6–8, 10–12, 14, 16, 18, 19, 21, 22, 24, 25, 27–29, 31