Package 'cnefetools'

February 11, 2026

Title Access and Analysis of Brazilian CNEFE Address Data

Version 0.2.0

Description Download, cache and read municipality-level address data from the Cadastro Nacional de Enderecos para Fins Estatisticos (CNEFE) of the 2022 Brazilian Census, published by the Instituto Brasileiro de Geografia e Estatistica (IBGE) <https://ftp.ibge.gov.br/Cadastro_Nacional_de_Enderecos_para_Fins_Estatisticos/>. Beyond data access, provides spatial aggregation of addresses, computation of land-use mix indices, and dasymetric interpolation of census tract variables using CNEFE dwelling points as ancillary data. Results can be produced on 'H3' hexagonal grids or user-supplied polygons, and heavy operations leverage a 'DuckDB' backend with extensions for fast, in-process execution.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3.9000

URL https://github.com/pedreirajr/cnefetools, https://pedreirajr.github.io/cnefetools/

BugReports https://github.com/pedreirajr/cnefetools/issues

Suggests ggplot2, kableExtra, knitr, leafsync, mapview, odbr, rmarkdown, scales, testthat (>= 3.0.0), zip

Config/testthat/edition 3

Depends R (>= 4.1.0)

Imports arrow, dplyr, sf, geobr, lifecycle, rlang, h3jsr, tidyr, DBI, duckdb, duckspatial, cli (>= 3.6.0), checkmate, fs, httr2, piggyback

LazyData true

NeedsCompilation no

Author Jorge Ubirajara Pedreira Junior [aut, cre, cph] (ORCID:
<https://orcid.org/0000-0002-8243-5395>),
Bruno Mioto [aut],
Kaio Cunha Pedreira [ctb]

Maintainer Jorge Ubirajara Pedreira Junior <jorge.ubirajara@ufba.br>

Repository CRAN

Date/Publication 2026-02-11 20:00:15 UTC

Contents

`cnefe_counts` *Count CNEFE address species on a spatial grid*

Description

`cnefe_counts()` reads CNEFE records for a given municipality, assigns each address point to spatial units (either H3 hexagonal cells or user-provided polygons), and returns per-unit counts of `COD_ESPECIE` as `addr_type1` to `addr_type8`.

Usage

```
cnefe_counts(
  code_muni,
  year = 2022,
  polygon_type = c("hex", "user"),
  polygon = NULL,
  crs_output = NULL,
  h3_resolution = 9,
  verbose = TRUE,
  backend = c("duckdb", "r")
)
```

Arguments

code_muni	Integer. Seven-digit IBGE municipality code.
year	Integer. The CNEFE data year. Currently only 2022 is supported. Defaults to 2022.
polygon_type	Character. Type of polygon aggregation: "hex" (default) uses an H3 hexagonal grid; "user" uses polygons provided via the polygon parameter.
polygon	An sf::sf object with polygon geometries. Required when polygon_type = "user". A warning is issued reporting the percentage of CNEFE points covered by the polygon area. If no CNEFE points fall within the polygon, an error is raised.
crs_output	The CRS for the output object. Only used when polygon_type = "user". Default is NULL, which uses the original CRS of the polygon argument. Can be an EPSG code (e.g., 4326, 31983) or any CRS object accepted by sf::st_transform().
h3_resolution	Integer. H3 grid resolution (default: 9). Only used when polygon_type = "hex".
verbose	Logical; if TRUE, prints messages and timing information.
backend	Character. "duckdb" (default) uses DuckDB with H3/spatial extensions. "r" uses h3jsr and sf in R (slower but no DuckDB dependency).

Details

The counts in the columns addr_type1 to addr_type8 correspond to:

- addr_type1: Private household (Domicílio particular)
- addr_type2: Collective household (Domicílio coletivo)
- addr_type3: Agricultural establishment (Estabelecimento agropecuário)
- addr_type4: Educational establishment (Estabelecimento de ensino)
- addr_type5: Health establishment (Estabelecimento de saúde)
- addr_type6: Establishment for other purposes (Estabelecimento de outras finalidades)
- addr_type7: Building under construction or renovation (Edificação em construção ou reforma)
- addr_type8: Religious establishment (Estabelecimento religioso)

Value

An sf::sf object containing:

- id_hex (when polygon_type = "hex"): H3 cell identifier
- Original columns from polygon (when polygon_type = "user")
- addr_type1 ... addr_type8: counts per address type
- geometry: polygon geometry

When polygon_type = "user", the output CRS matches the original polygon CRS (or crs_output if specified).

Examples

```
# Count addresses per H3 hexagon (resolution 9)
hex_counts <- cnefe_counts(code_muni = 2929057)

# Count addresses per user-provided polygon (neighborhoods of Lauro de Freitas-BA)
# Using geobr to download neighborhood boundaries
library(geobr)
nei_ldf <- subset(
  read_neighborhood(year = 2022),
  code_muni == 2919207
)
hex_counts <- cnefe_counts(
  code_muni = 2919207,
  polygon_type = "user",
  polygon = nei_ldf
)
```

---

cnefe_dictionary *Open the official CNEFE data dictionary*

Description

Opens the bundled Excel data dictionary in the system's default spreadsheet viewer (e.g., Excel, LibreOffice).

Usage

```
cnefe_dictionary(year = 2022)
```

Arguments

year	Integer. The CNEFE data year. Currently only 2022 is supported.

Value

Invisibly, the path to the Excel file inside the installed package.

Examples

```
cnefe_dictionary()
```

cnefe_doc *Open the official CNEFE methodological note*

Description

Opens the bundled PDF methodological document in the system's default PDF viewer.

Usage

```
cnefe_doc(year = 2022)
```

Arguments

year	Integer. The CNEFE data year. Currently only 2022 is supported.

Value

Invisibly, the path to the PDF file inside the installed package.

Examples

```
cnefe_doc()
```

compute_lumi *Compute land-use mix indicators on a spatial grid*

Description

`compute_lumi()` reads CNEFE records for a given municipality, assigns each address point to spatial units (either H3 hexagonal cells or user-provided polygons), and computes the residential proportion (`p_res`) and land-use mix indices, such as the Entropy Index (`ei`), the Herfindahl-Hirschman Index (`hhi`), the Balance Index (`bal`), the Index of Concentration at Extremes (`ice`), the adapted HHI (`hhi_adp`), and the Bidirectional Global-centered Index (`bgbi`), following the methodology proposed in Pedreira Jr. et al. (2025).

Usage

```
compute_lumi(
  code_muni,
  year = 2022,
  polygon_type = c("hex", "user"),
  polygon = NULL,
  crs_output = NULL,
  h3_resolution = 9,
  verbose = TRUE,
  backend = c("duckdb", "r")
)
```

Arguments

code_muni	Integer. Seven-digit IBGE municipality code.
year	Integer. The CNEFE data year. Currently only 2022 is supported. Defaults to 2022.
polygon_type	Character. Type of polygon aggregation: "hex" (default) uses an H3 hexagonal grid; "user" uses polygons provided via the polygon parameter.
polygon	An sf::sf object with polygon geometries. Required when polygon_type = "user". A warning is issued reporting the percentage of CNEFE points covered by the polygon area. If no CNEFE points fall within the polygon, an error is raised.
crs_output	The CRS for the output object. Only used when polygon_type = "user". Default is NULL, which uses the original CRS of the polygon argument. Can be an EPSG code (e.g., 4326, 31983) or any CRS object accepted by sf::st_transform().
h3_resolution	Integer. H3 grid resolution (default: 9). Only used when polygon_type = "hex".
verbose	Logical; if TRUE, prints messages and timing information.
backend	Character. "duckdb" (default) uses DuckDB + H3 extension reading directly from the cached ZIP. "r" computes H3 in R using h3jsr.

Value

An sf::sf object containing:

When polygon_type = "hex":
- id_hex: H3 cell identifier
- p_res, ei, hhi, bal, ice, hhi_adp, bgbi: land-use mix indicators
- geometry: hexagon geometry (CRS 4326)

When polygon_type = "user":
- Original columns from polygon
- p_res, ei, hhi, bal, ice, hhi_adp, bgbi: land-use mix indicators
- geometry: polygon geometry (in the original or crs_output CRS)

References

Pedreira Jr., J. U.; Louro, T. V.; Assis, L. B. M.; Brito, P. L. Measuring land use mix with address-level census data (2025). *engrXiv*. https://engrxiv.org/preprint/view/5975

Booth, A.; Crouter, A. C. (Eds.). (2001). *Does It Take a Village? Community Effects on Children, Adolescents, and Families*. Psychology Press.

Song, Y.; Merlin, L.; Rodriguez, D. (2013). Comparing measures of urban land use mix. *Computers, Environment and Urban Systems*, 42, 1–13. https://doi.org/10.1016/j.compenvurbsys.2013.08.001

Examples

```
# Compute land-use mix indices on H3 hexagons
lumi <- compute_lumi(code_muni = 2929057)

# Compute land-use mix indices on user-provided polygons (neighborhoods of Lauro de Freitas-BA)
# Using geobr to download neighborhood boundaries
```

```
library(geobr)
nei_ldf <- subset(
  read_neighborhood(year = 2022),
  code_muni == 2919207
)
lumi_poly <- compute_lumi(
  code_muni = 2919207,
  polygon_type = "user",
  polygon = nei_ldf
)
```

`read_cnefe` *Read CNEFE data for a given municipality*

Description

Downloads and reads the CNEFE CSV file for a given IBGE municipality code, using the official IBGE FTP structure. The function relies on an internal index linking municipality codes to the corresponding ZIP URLs. Data are returned either as an Arrow Table (default) or as an sf object with SIRGAS 2000 coordinates.

Usage

```
read_cnefe(
  code_muni,
  year = 2022,
  verbose = TRUE,
  cache = TRUE,
  output = c("arrow", "sf")
)
```

Arguments

`code_muni`	Integer. Seven-digit IBGE municipality code.
`year`	Integer. The CNEFE data year. Currently only 2022 is supported. Defaults to 2022.
`verbose`	Logical; if `TRUE`, print informative messages about download, extraction, and reading steps.
`cache`	Logical; if `TRUE`, cache the downloaded ZIP file in a user-level cache directory specific to this package. If `FALSE`, a temporary file is used and removed after reading.
`output`	Character. Output format. `"arrow"` (default) returns an arrow::Table, whereas `"sf"` returns an sf point object with coordinates built from `LONGITUDE` / `LATITUDE` in CRS 4674.

Details

When `output = "arrow"` (default), the function does not perform any spatial conversion and simply returns the Arrow table. When `output = "sf"`, the function converts the result to an sf point object using the `LONGITUDE` and `LATITUDE` columns, with CRS EPSG:4674 (SIRGAS 2000), keeping these columns in the final object (`remove = FALSE`).

Value

If `output = "arrow"`, an arrow::Table containing all CNEFE records for the given municipality.

If `output = "sf"`, an sf object with point geometry in EPSG:4674 (SIRGAS 2000), using the `LONGITUDE` and `LATITUDE` columns.

Caching

When `cache = TRUE` (the default), the downloaded ZIP file is stored in a user-level cache directory specific to this package, created via `tools::R_user_dir()` with `which = "cache"`. This avoids re-downloading the same municipality file across sessions.

When `cache = FALSE`, the ZIP file is stored in a temporary location and removed when the function exits.

Examples

```
# Read CNEFE data as an Arrow table
cnefe <- read_cnefe(code_muni = 2929057)

# Read as an sf spatial object
cnefe_sf <- read_cnefe(code_muni = 2929057, output = "sf")
```

`tracts_to_h3` *Convert census tract aggregates to an H3 grid using CNEFE points*

Description

`tracts_to_h3()` performs a dasymetric interpolation with the following steps:

1. census tract totals are allocated to CNEFE dwelling points inside each tract;
2. allocated values are aggregated to an H3 grid at a user-defined resolution.

The function uses DuckDB with the spatial and H3 extensions for the heavy work.

Usage

```
tracts_to_h3(
  code_muni,
  year = 2022,
  h3_resolution = 9,
  vars = c("pop_ph", "pop_ch"),
  cache = TRUE,
  verbose = TRUE
)
```

Arguments

code_muni	Integer. Seven-digit IBGE municipality code.
year	Integer. The CNEFE data year. Currently only 2022 is supported. Defaults to 2022.
h3_resolution	Integer. H3 resolution (0 to 15). Defaults to 9.
vars	Character vector. Names of tract-level variables to interpolate. Supported variables: • pop_ph: population in private households (*Domicílios particulares*). • pop_ch: population in collective households (*Domicílios coletivos*). • male: total male population. • female: total female population. • age_0_4, age_5_9, age_10_14, age_15_19, age_20_24, age_25_29, age_30_39, age_40_49, age_50_59, age_60_69, age_70m: population by age group. • race_branca, race_preta, race_amarela, race_parda, race_indigena: population by race/color (*cor ou raça*). • n_resp: number of household heads (*Pessoas responsáveis por domicílios*). • avg_inc_resp: average income of the household heads. For a reference table mapping these variable names to the official IBGE census tract codes and descriptions, see tracts_variables_ref. Allocation rules: • pop_ph is allocated only to private dwellings. • pop_ch is allocated only to collective dwellings. • n_resp is allocated only to private dwellings (same rule as pop_ph). • Demographic variables (male, female, age_*, race_*) are allocated to private dwellings when the tract has any; if the tract has zero private dwellings but has collective dwellings, they are allocated to collective. • avg_inc_resp is assigned (not split) to each private dwelling point; tracts with no private dwellings receive no allocation.
cache	Logical. Whether to use the existing package cache for assets and CNEFE zips.
verbose	Logical. Whether to print step messages and timing.

Value

An sf object (CRS 4326) with an H3 grid and the requested interpolated variables.

Examples

```
# Interpolate population to H3 hexagons
hex_pop <- tracts_to_h3(
  code_muni = 2929057,
  vars = c("pop_ph", "pop_ch")
)
```

`tracts_to_polygon` *Convert census tract aggregates to user-provided polygons using CNEFE points*

Description

`tracts_to_polygon()` performs a dasymetric interpolation with the following steps:

1. census tract totals are allocated to CNEFE dwelling points inside each tract;
2. allocated values are aggregated to user-provided polygons (neighborhoods, administrative divisions, custom areas, etc.).

The function uses DuckDB with spatial extensions for the heavy work.

Usage

```
tracts_to_polygon(
  code_muni,
  polygon,
  year = 2022,
  vars = c("pop_ph", "pop_ch"),
  crs_output = NULL,
  cache = TRUE,
  verbose = TRUE
)
```

Arguments

`code_muni`	Integer. Seven-digit IBGE municipality code.
`polygon`	An `sf::sf` object with polygon geometries (POLYGON or MULTIPOLYGON). The function will automatically align CRS and issue a warning reporting the percentage of the polygon area that falls outside the municipality.
`year`	Integer. The CNEFE data year. Currently only 2022 is supported. Defaults to 2022.
`vars`	Character vector. Names of tract-level variables to interpolate. Supported variables: • `pop_ph`: population in private households (*Domicílios particulares*).

- pop_ch: population in collective households (*Domicílios coletivos*).
- male: total male population.
- female: total female population.
- age_0_4, age_5_9, age_10_14, age_15_19, age_20_24, age_25_29, age_30_39, age_40_49, age_50_59, age_60_69, age_70m: population by age group.
- race_branca, race_preta, race_amarela, race_parda, race_indigena: population by race/color (*cor ou raça*).
- n_resp: number of household heads (*Pessoas responsáveis por domicílios*).
- avg_inc_resp: average income of the household heads.

For a reference table mapping these variable names to the official IBGE census tract codes and descriptions, see tracts_variables_ref.

Allocation rules:

- pop_ph is allocated only to private dwellings.
- pop_ch is allocated only to collective dwellings.
- n_resp is allocated only to private dwellings (same rule as pop_ph).
- Demographic variables (male, female, age_*, race_*) are allocated to private dwellings when the tract has any; if the tract has zero private dwellings but has collective dwellings, they are allocated to collective.
- avg_inc_resp is assigned (not split) to each private dwelling point; tracts with no private dwellings receive no allocation.

crs_output — The CRS for the output object. Default is NULL, which uses the original CRS of the polygon argument. Can be an EPSG code (e.g., 4326, 31983) or any CRS object accepted by sf::st_transform().

cache — Logical. Whether to use the existing package cache for assets and CNEFE zips.

verbose — Logical. Whether to print step messages and timing.

Value

An sf object with the user-provided polygons and the requested interpolated variables. The output CRS matches the original polygon CRS (or crs_output if specified).

Examples

```
# Interpolate population to user-provided polygons (neighborhoods of Lauro de Freitas-BA)
# Using geobr to download neighborhood boundaries
library(geobr)
nei_ldf <- subset(
  read_neighborhood(year = 2022),
  code_muni == 2919207
)
poly_pop <- tracts_to_polygon(
  code_muni = 2919207,
  polygon = nei_ldf,
  vars = c("pop_ph", "pop_ch")
)
```

`tracts_variables_ref` *Reference table for tracts_to_* function variables*

Description

A data frame that maps variable names used in `tracts_to_h3()` and `tracts_to_polygon()` to the official IBGE census tract dataset codes and descriptions.

Usage

```
tracts_variables_ref
```

Format

A data frame with 22 rows and 4 columns:

var_cnefetools Variable name used in cnefetools functions.

code_var_ibge Official IBGE variable code from the census tract aggregates.

desc_var_ibge Official IBGE variable description in Portuguese.

table_ibge Name of the IBGE census tract table where the variable is found (Domicilios, Pessoas, or ResponsavelRenda).

Source

IBGE - Censo Demografico 2022, Agregados por Setores Censitarios.

Examples

```
# View the reference table
tracts_variables_ref

# Find the IBGE code for a specific variable
tracts_variables_ref[tracts_variables_ref$var_cnefetools == "pop_ph", ]
```

Index