

Package ‘geocodebr’

February 12, 2025

Type Package

Title Geolocalização De Endereços Brasileiros (Geocoding Brazilian Addresses)

Version 0.1.0

Description Método simples e eficiente de geolocalizar dados no Brasil. O pacote é baseado em conjuntos de dados espaciais abertos de endereços brasileiros, utilizando principalmente o Cadastro Nacional de Endereços para Fins Estatísticos (CNEFE). O CNEFE é publicado pelo Instituto Brasileiro de Geografia e Estatística (IBGE), órgão oficial de estatísticas e geografia do Brasil. (A simple and efficient method for geolocating data in Brazil. The package is based on open spatial datasets of Brazilian addresses, primarily using the Cadastro Nacional de Endereços para Fins Estatísticos (CNEFE), published by the Instituto Brasileiro de Geografia e Estatística (IBGE), Brazil's official statistics and geography agency.)

License MIT + file LICENSE

URL <https://github.com/ipeaGIT/geocodebr>,
<https://ipeagit.github.io/geocodebr/>

BugReports <https://github.com/ipeaGIT/geocodebr/issues>

Depends R (>= 4.1.0)

Imports arrow (>= 15.0.1), checkmate, cli, data.table, DBI, dplyr, duckdb, enderecobr, fs, glue, httr2 (>= 1.0.0), parallel, purrr, rlang, sf, sfheaders, tools

Suggests covr, dbplyr, geobr, ggplot2 (>= 3.3.1), knitr, rmarkdown, scales, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Rafael H. M. Pereira [aut, cre]
 (<<https://orcid.org/0000-0003-2125-7465>>),
 Daniel Herszenhut [aut] (<<https://orcid.org/0000-0001-8066-1105>>),
 Ipea - Instituto de Pesquisa Econômica Aplicada [cph, fnd]

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

Repository CRAN

Date/Publication 2025-02-12 20:00:02 UTC

Contents

definir_campos	2
definir_pasta_cache	3
deletar_pasta_cache	4
download_cnefe	5
geocode	5
listar_dados_cache	9
listar_pasta_cache	10

Index	11
--------------	-----------

definir_campos	<i>Especifica as colunas que descrevem os campos dos endereços</i>
----------------	--

Description

Cria um vetor de caracteres especificando as colunas que representam cada campo do endereço na tabela de endereços. Os campos estado e municipio são obrigatórios.

Usage

```
definir_campos(  
  estado,  
  municipio,  
  logradouro = NULL,  
  numero = NULL,  
  cep = NULL,  
  localidade = NULL  
)
```

Arguments

estado	Uma string. O nome da coluna que representa o estado do endereço. Campo obrigatório. Na tabela de endereços, essa coluna pode conter os nomes dos estados por extenso, ou a abreviação oficial dos estados com duas letras, e.g. "AM", "SP", "DF", "RJ".
--------	--

município	Uma string. O nome da coluna que representa o município do endereço. Campo obrigatório. Na tabela de endereços, essa coluna pode conter o nome dos municípios, ou o seu código IBGE de 7 dígitos.
logradouro	Uma string. O nome da coluna que representa o <i>logradouro</i> (endereço da rua) do endereço. Pode ser NULL se o campo não estiver especificado na tabela de endereços. Na tabela de endereços, essa coluna deve incluir o <i>tipo</i> do logradouro, indicando se trata-se de uma "Rua" ou "Avenida" etc, por exemplo "Avenida Presidente Getúlio Vargas". Além disso, essa coluna <i>não</i> deve incluir o número do endereço, pois o número deve ser indicado numa coluna separada.
numero	Uma string. O nome da coluna que representa o número do endereço. Pode ser NULL se o campo não estiver especificado na tabela de endereços. Na tabela de endereços, valores como 0 ou caracteres não numéricos como "S/N" ou "10a" são considerados como NA.
cep	Uma string. O nome da coluna que representa o <i>CEP</i> (Código de Endereçamento Postal) do endereço. Pode ser NULL se o campo não estiver especificado na tabela de endereços.
localidade	Uma string. O nome da coluna que representa a localidade (equivalente ao 'bairro' em áreas urbanas) do endereço. Pode ser NULL se esse campo não estiver presente na tabela de endereços.

Value

Um vetor de caracteres no qual os nomes são os campos do endereço e os valores são as colunas que os representam na tabela de endereços.

Examples

```
definir_campos(  
  logradouro = "Nome_logradouro",  
  numero = "Numero",  
  cep = "CEP",  
  localidade = "Bairro",  
  municipio = "Cidade",  
  estado = "UF"  
)
```

definir_pasta_cache *Define um diretório de cache para o geocodebr*

Description

Define um diretório de cache para os dados do geocodebr. Essa configuração é persistente entre sessões do R.

Usage

```
definir_pasta_cache(path)
```

Arguments

path Uma string. O caminho para o diretório usado para armazenar os dados em cache. Se NULL, o pacote usará um diretório versionado salvo dentro do diretório retornado por `tools::R_user_dir()`.

Value

Retorna de forma invisível o caminho do diretório de cache.

Examples

```
definir_pasta_cache(tempdir())  
  
# retoma pasta padrão do pacote  
definir_pasta_cache( path = NULL)
```

deletar_pasta_cache *Deletar pasta de cache do geocodebr*

Description

Deleta todos arquivos da pasta do cache.

Usage

```
deletar_pasta_cache()
```

Value

Retorna de forma invisível o caminho do diretório de cache.

Examples

```
deletar_pasta_cache()
```

download_cnefe	<i>Faz download dos dados do CNEFE</i>
----------------	--

Description

Faz o download de uma versão pre-processada e enriquecida do CNEFE (Cadastro Nacional de Endereços para Fins Estatísticos) que foi criada para o uso deste pacote.

Usage

```
download_cnefe(verboso = TRUE, cache = TRUE)
```

Arguments

verboso	Um valor lógico. Indica se barras de progresso e mensagens devem ser exibidas durante o download dos dados do CNEFE e a geocodificação dos endereços. O padrão é TRUE.
cache	Um valor lógico. Indica se os dados do CNEFE devem ser salvos ou lidos do cache, reduzindo o tempo de processamento em chamadas futuras. O padrão é TRUE. Quando FALSE, os dados do CNEFE são baixados para um diretório temporário.

Value

Retorna o caminho para o diretório onde os dados foram salvos.

Examples

```
download_cnefe(verboso = FALSE)
```

geocode	<i>Geocaliza endereços no Brasil</i>
---------	--------------------------------------

Description

Geocodifica endereços brasileiros com base nos dados do CNEFE. Os endereços de input devem ser passados como um `data.frame`, no qual cada coluna descreve um campo do endereço (logradouro, número, cep, etc). O resultados dos endereços geocalizados podem seguir diferentes níveis de precisão. Consulte abaixo a seção "Detalhes" para mais informações. As coordenadas de output utilizam o sistema de referência geodésico "SIRGAS2000", CRS(4674).

Usage

```
geocode(
  enderecos,
  campos_endereco = definir_campos(),
  resultado_completo = FALSE,
  resolver_empates = FALSE,
  resultado_sf = FALSE,
  verboso = TRUE,
  cache = TRUE,
  n_cores = 1
)
```

Arguments

enderecos	Um <code>data.frame</code> . Os endereços a serem geolocalizados. Cada coluna deve representar um campo do endereço.
campos_endereco	Um vetor de caracteres. A correspondência entre cada campo de endereço e o nome da coluna que o descreve na tabela <code>enderecos</code> . A função <code>definir_campos()</code> auxilia na criação deste vetor e realiza algumas verificações nos dados de entrada. Campos de endereço passados como <code>NULL</code> serão ignorados, e a função deve receber pelo menos um campo não nulo, além dos campos "estado" e "município", que são obrigatórios. Note que o campo "localidade" é equivalente a 'bairro'.
resultado_completo	Lógico. Indica se o output deve incluir colunas adicionais, como o endereço encontrado de referência. Por padrão, é <code>FALSE</code> .
resolver_empates	Lógico. Alguns resultados da geolocalização podem indicar diferentes coordenadas possíveis (e.g. duas ruas diferentes com o mesmo nome em uma mesma cidade). Esses casos são tratados como 'empate' e o parâmetro <code>resolver_empates</code> indica se a função deve resolver esses empates automaticamente. Por padrão, é <code>FALSE</code> , e a função retorna apenas o caso mais provável.
resultado_sf	Lógico. Indica se o resultado deve ser um objeto espacial da classe <code>sf</code> . Por padrão, é <code>FALSE</code> , e o resultado é um <code>data.frame</code> com as colunas <code>lat</code> e <code>lon</code> adicionais, como o endereço encontrado de referência. Por padrão, é <code>FALSE</code> .
verboso	Um valor lógico. Indica se barras de progresso e mensagens devem ser exibidas durante o download dos dados do CNEFE e a geocodificação dos endereços. O padrão é <code>TRUE</code> .
cache	Um valor lógico. Indica se os dados do CNEFE devem ser salvos ou lidos do cache, reduzindo o tempo de processamento em chamadas futuras. O padrão é <code>TRUE</code> . Quando <code>FALSE</code> , os dados do CNEFE são baixados para um diretório temporário.
n_cores	Um número. O número de núcleos de CPU a serem utilizados no processamento dos dados. O padrão é 1.

Details

Precisão dos resultados:

Os resultados do **geocodebr** são classificados em seis amplas categorias de precisão:

1. "numero"
2. "numero_aproximado"
3. "logradouro"
4. "cep"
5. "localidade"
6. "municipio"

Cada nível de precisão pode ser desagregado em tipos de correspondência mais refinados.

Value

Retorna o `data.frame` de input `enderecos` adicionado das colunas de latitude (`lat`) e longitude (`lon`), bem como as colunas (`precisao` e `tipo_resultado`) que indicam o nível de precisão e o tipo de resultado. Alternativamente, o resultado pode ser um objeto `sf`.

Tipos de resultados

A coluna `tipo_resultado` fornece informações mais detalhadas sobre como exatamente cada endereço de entrada foi encontrado no CNEFE. Em cada categoria, o **geocodebr** calcula a média da latitude e longitude dos endereços incluídos no CNEFE que correspondem ao endereço de entrada, com base em combinações de diferentes campos. No caso mais rigoroso, por exemplo, a função encontra uma correspondência determinística para todos os campos de um dado endereço ("`estado`", "`municipio`", "`logradouro`", "`numero`", "`cep`", "`localidade`"). Pense, por exemplo, em um prédio com vários apartamentos que correspondem ao mesmo endereço de rua e número. Nesse caso, as coordenadas dos apartamentos podem diferir ligeiramente, e o **geocodebr** calcula a média dessas coordenadas. Em um caso menos rigoroso, no qual apenas os campos ("`estado`", "`municipio`", "`logradouro`", "`localidade`") são encontrados, o **geocodebr** calcula as coordenadas médias de todos os endereços no CNEFE ao longo daquela rua e que se encontram na mesma localidade/bairro. Assim, as coordenadas de resultado tendem a ser o ponto médio do trecho daquela rua que passa dentro daquela localidade/bairro.

A coluna `tipo_resultado` fornece informações mais detalhadas sobre os campos de endereço utilizados no cálculo das coordenadas de cada endereço de entrada. Cada categoria é nomeada a partir de um código de quatro caracteres:

- o primeiro caracter, sempre `d` ou `p`, determina se a correspondência foi feita de forma determinística (`d`) ou probabilística (`p`) - a segunda opção ainda não foi implementada no pacote, mas é planejada em versões futuras;
- o segundo faz menção à categoria de `precisao` na qual o resultado foi classificado (`n` para "`numero`", `a` para "`numero_aproximado`", `r` para "`logradouro`", `c` para "`cep`", `b` para "`localidade`" e `m` para "`municipio`");
- o terceiro e o quarto caracteres designam a classificação de cada categoria dentro de seu grupo - via de regra, quanto menor o número formado por esses caracteres, mais precisa são as coordenadas calculadas.

As categorias de tipo_resultado são listadas abaixo, junto às categorias de precisao a qual elas estão associadas:

- precisao "numero"
 - dn01 - logradouro, numero, cep e localidade
 - dn02 - logradouro, numero e cep
 - dn03 - logradouro, numero e localidade
 - dn04 - logradouro e numero
 - pn01 - logradouro, numero, cep e localidade
 - pn02 - logradouro, numero e cep
 - pn03 - logradouro, numero e localidade
 - pn04 - logradouro e numero
- precisao "numero_aproximado"
 - da01 - logradouro, numero, cep e localidade
 - da02 - logradouro, numero e cep
 - da03 - logradouro, numero e localidade
 - da04 - logradouro e numero
 - pa01 - logradouro, numero, cep e localidade
 - pa02 - logradouro, numero e cep
 - pa03 - logradouro, numero e localidade
 - pa04 - logradouro e numero
- precisao "logradouro" (quando o número de entrada está faltando 'S/N')
 - dl01 - logradouro, cep e localidade
 - dl02 - logradouro e cep
 - dl03 - logradouro e localidade
 - dl04 - logradouro
 - pl01 - logradouro, cep e localidade
 - pl02 - logradouro e cep
 - pl03 - logradouro e localidade
 - pl04 - logradouro
- precisao "cep"
 - dc01 - municipio, cep, localidade
 - dc02 - municipio, cep
- precisao "localidade"
 - db01 - municipio, localidade
- precisao "municipio"
 - dm01 - municipio

Endereços não encontrados são retornados com latitude, longitude, precisão e tipo de resultado NA.

Examples

```
library(geocodebr)

# ler amostra de dados
data_path <- system.file("extdata/small_sample.csv", package = "geocodebr")
input_df <- read.csv(data_path)

fields <- geocodebr::definir_campos(
  logradouro = "nm_logradouro",
  numero = "Numero",
  cep = "Cep",
  localidade = "Bairro",
  municipio = "nm_municipio",
  estado = "nm_uf"
)

df <- geocodebr::geocode(
  enderecos = input_df,
  campos_endereco = fields,
  resolver_empates = TRUE,
  verboso = FALSE
)

head(df)
```

listar_dados_cache *Listar dados em cache*

Description

Lista os dados salvos localmente na pasta de cache

Usage

```
listar_dados_cache(print_tree = FALSE)
```

Arguments

print_tree Um valor lógico. Indica se o conteúdo da pasta de cache deve ser exibido em um formato de árvore. O padrão é FALSE.

Value

O caminho para os arquivos em cache

Examples

```
listar_dados_cache()

listar_dados_cache(print_tree = TRUE)
```

listar_pasta_cache *Obtém a pasta de cache usado no geocodebr*

Description

Obtém o caminho da pasta utilizada para armazenar em cache os dados do geocodebr. Útil para inspecionar a pasta configurada com [definir_pasta_cache\(\)](#) em uma sessão anterior do R. Retorna a pasta de cache padrão caso nenhuma pasta personalizado tenha sido configurada anteriormente.

Usage

```
listar_pasta_cache()
```

Value

O caminho da pasta de cache.

Examples

```
listar_pasta_cache()
```

Index

definir_campos, [2](#)
definir_campos(), [6](#)
definir_pasta_cache, [3](#)
definir_pasta_cache(), [10](#)
deletar_pasta_cache, [4](#)
download_cnefe, [5](#)

geocode, [5](#)

listar_dados_cache, [9](#)
listar_pasta_cache, [10](#)

tools::R_user_dir(), [4](#)