

Package: mapineqr (via r-universe)

October 10, 2026

Title Access Mapineq Inequality Indicators via API
Version 0.0.0.9000
Description Access Mapineq inequality indicators via API.
License MIT + file LICENSE
URL <https://github.com/e-kotov/mapineqr/>,
<http://www.ekotov.pro/mapineqr/>
BugReports <https://github.com/e-kotov/mapineqr/issues>
Imports checkmate, dplyr, httr2, jsonlite, purrr, rlang, tibble, tidyr
Suggests biscale, eurostat, nuts
Config/testthat/edition 3
Encoding UTF-8
Roxygen list(markdown = TRUE)
RoxygenNote 7.3.2
Config/pak/sysreqs libicu-dev libssl-dev
Repository <https://e-kotov.r-universe.dev>
Date/Publication 2025-02-26 20:37:22 UTC
RemoteUrl <https://github.com/e-kotov/mapineqr>
RemoteRef HEAD
RemoteSha 3f61adb7a8e4077c456b1722895f881aa8a0a572

Contents

mi_data	2
mi_nuts_levels	3
mi_source_coverage	4
mi_source_filters	5
mi_sources	6
Index	7

`mi_data`*Get univariate or bivariate data for a specific source*

Description

Fetches univariate or bivariate data for a given source, year, NUTS level, and selected filters.

Usage

```
mi_data(  
  x_source,  
  y_source = NULL,  
  year,  
  level,  
  x_filters = list(),  
  y_filters = NULL,  
  limit = 2500  
)
```

Arguments

<code>x_source</code>	A character string specifying the source name for the x variable.
<code>y_source</code>	(Optional) A character string specifying the source name for the y variable.
<code>year</code>	A character or integer specifying the year.
<code>level</code>	A character string specifying the NUTS level ("0", "1", "2", or "3").
<code>x_filters</code>	A named <code>list</code> where the names are the filter fields for the x variable and the values are the selected values for those fields. Default is an empty list. To find out which filters to use, use mi_source_filters with the desired <code>source_name</code> .
<code>y_filters</code>	(Optional) A named <code>list</code> where the names are the filter fields for the y variable and the values are the selected values for those fields. Default is <code>NULL</code> . To find out which filters to use, use mi_source_filters with the desired <code>source_name</code> .
<code>limit</code>	An integer specifying the maximum number of results to return. Default is 2500. This default should be enough for most uses, as it is well above the number of NUTS 3 regions in the EU. The maximum limited by the API is 10000.

Value

A tibble with the following columns:

- `geo`: code for the (NUTS) region at the requested level.
- `geo_name`: name of the (NUTS) region at the requested level.
- `geo_source`: source (type) of the spatial units at the requested level.
- `geo_year`: year of the (NUTS) region at the requested level.

- x_year: The year of the predictor variable (X), included in bivariate requests.
- y_year (optional): The year of the outcome variable (Y), included in bivariate requests (only included when y_source is provided).
- x: the value of the univariate variable.
- y (optional): the value of the y variable (only included when y_source is provided).

Examples

```
# Univariate example
mi_data(
  x_source = "TGS00010",
  year = 2020,
  level = "2",
  x_filters = list(iscd11 = "TOTAL", sex = "F")
)

# Bivariate example
mi_data(
  x_source = "TGS00010",
  y_source = "DEMO_R_MLIFEXP",
  year = 2020,
  level = "2",
  x_filters = list(iscd11 = "TOTAL", sex = "F"),
  y_filters = list(age = "Y2", sex = "F")
)
```

mi_nuts_levels

Get a list of available NUTS levels

Description

Get a list of available NUTS levels

Usage

```
mi_nuts_levels()
```

Value

a character vector of valid NUTS levels that will be accepted by other functions.

Examples

```
mi_nuts_levels()
```

mi_source_coverage	<i>Get NUTS level and Year coverage for a specific source</i>
--------------------	---

Description

Get the NUTS level and Year coverage for a specific data source.

Usage

```
mi_source_coverage(source_name, limit = 2500)
```

Arguments

source_name	name of the data source
limit	An integer specifying the maximum number of results to return. Default is 2500. This default should be enough for most uses, as it is well above the number of NUTS 3 regions in the EU. The maximum limited by the API is 10000.

Value

a tibble containing the following columns:

- nuts_level: NUTS level
- year: year
- source_name: name of the data source (mathces the source_name requested by the user)
- short_description: short description of the data source
- description: description of the data source

Examples

```
mi_source_coverage("BD_HGNACE2_R3")
```

```
mi_source_coverage("ghs_smod")
```

mi_source_filters	<i>Get column values for filtering a source</i>
-------------------	---

Description

Fetches the possible filtering values for a given source, year, and NUTS level.

Usage

```
mi_source_filters(source_name, year, level, filters = list(), limit = 2500)
```

Arguments

source_name	A character string specifying the source name (f_resource).
year	A character or integer specifying the year.
level	A character string specifying the NUTS level ("0", "1", "2", or "3").
filters	A named list where the names are the filter fields and the values are the selected values for those fields. Default is an empty list.
limit	An integer specifying the maximum number of results to return. Default is 2500. This default should be enough for most uses, as it is well above the number of NUTS 3 regions in the EU. The maximum limited by the API is 10000.

Value

A tibble with the fields, labels, and their possible values for filtering.

Examples

```
mi_source_filters(  
  source_name = "DEMO_R_FIND2",  
  year = 2020,  
  level = "2",  
  filters = list(unit = "YR")  
)
```

`mi_sources`*Get a list of available data sources*

Description

Get a list of available data sources

Usage

```
mi_sources(level, year = NULL, limit = 2500)
```

Arguments

<code>level</code>	a character string specifying the NUTS level ("0", "1", "2", or "3"). You can also always check valid NUTS levels using mi_nuts_levels .
<code>year</code>	an integer of length 1, specifying the year. Optional.
<code>limit</code>	An integer specifying the maximum number of results to return. Default is 2500. This default should be enough for most uses, as it is well above the number of NUTS 3 regions in the EU. The maximum limited by the API is 10000.

Value

a tibble of sources with the following columns:

- `source_name`: name of the data source
- `short_description`: short description of the data source
- `description`: description of the data source

Examples

```
# get up to 10 sources for NUTS level 3
mi_sources("3", limit = 10)

# get all sources for NUTS level 3 and year 2020
mi_sources("3", year = 2020)
```

Index

mi_data, [2](#)
mi_nuts_levels, [3](#), [6](#)
mi_source_coverage, [4](#)
mi_source_filters, [2](#), [5](#)
mi_sources, [6](#)