Package ‘igoR’

February 5, 2024

Title Intergovernmental Organizations Database

Version 0.2.0

Description Tools to extract information from the Intergovernmental Organizations (IGO) Database, version 3, provided by the Correlates of War Project <https://correlatesofwar.org/>. See also Pevehouse, J. C. et al. (2020). Version 3 includes information from 1815 to 2014.

License GPL (>= 3)


BugReports https://github.com/dieghernan/igoR/issues

Depends R (>= 2.10)

Suggests countrycode (>= 1.1.0), dplyr (>= 1.0.2), ggplot2, knitr, markdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/Needs/website dieghernan/gitdevr, reactable, giscoR, devtools

Config/testthat/edition 3

Config/testthat/parallel true

Copyright © 2000-2021 by the Correlates of War

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

X-schema.org-keywords R, IGO, Correlates of War, Intergovernmental Organisations

NeedsCompilation no

Author Diego Hernangómez [aut, cre, cph]

Maintainer Diego Hernangómez <diego.hernangomezherrero@gmail.com>

Repository CRAN

Date/Publication 2024-02-05 15:30:02 UTC
R topics documented:

igo_dyadic 
igo_members 
igo_recode_igoyear 
igo_search 
igo_search_states 
igo_state_membership 
igo_year_format3 
states2016 
state_year_format3

Index

igo_dyadic  Extract the Joint Membership of a pair of Countries across IGOs.

Description

Dyadic version of the data. The unit of observation is a dyad of countries. It provides a summary of the joint memberships of two countries across IGOs over time.

Usage

igo_dyadic(country1, country2, year = 1816:2014, ioname = NULL)

Arguments

country1, country2
A state of vector of states to be compared. It could be any valid name or code of a state as specified on states2016.

year
Year to be assessed, an integer or an array of year.

ioname
Optional. ioname or vector of ioname corresponding to the IGOs to be assessed. If NULL (the default), all IGOs would be extracted. See codes on igo_search().

Details

This function tries to replicate the information contained in the original file distributed by The Correlates of War Project (dyadic_format3.dta). That file is not included in this package due to its size.

The result is a data.frame containing the common years of the states selected via country1, country2, year by rows.

An additional column dyadid, computed as (1000*ccode1)+ccode2 is provided in order to identify relationships.

For each IGO selected via ioname (or all if the default option has been used) a column (using lowercase ioname as identifier) is provided with the following code system:
**igo_dyadic**

<table>
<thead>
<tr>
<th>Category</th>
<th>Numerical Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Joint Membership</td>
<td>0</td>
</tr>
<tr>
<td>Joint Full Membership</td>
<td>1</td>
</tr>
<tr>
<td>Missing data</td>
<td>-9</td>
</tr>
<tr>
<td>State Not System Member</td>
<td>-1</td>
</tr>
</tbody>
</table>

See `igo_recode_dyadic()` section for an easy way to recode the numerical values into factors.

If one state in an IGO is a full member but the other is an associate member or observer, that IGO is not coded as a joint membership.

**Value**

A coded data.frame representing the years and country dyad (rows) and the IGOs selected (columns). See Details.

**Differences with the original dataset**

There are some differences on the results provided by this function and the original dataset on some IGOs regarding the "Missing Data" (-9) and "State Not System Member" (-1). However it is not clear how to fully replicate those values.

See Codebook Version 3 IGO Data

**Source**

Codebook Version 3 IGO Data for full reference.

**References**


**See Also**

state_year_format3, states2016, igo_search().

**Examples**

```r
usa_esp <- igo_dyadic("USA", "Spain")
nrow(usa_esp)
ncol(usa_esp)
dplyr::tibble(usa_esp)
```

# Using custom parameters
custom <- igo_dyadic(
  country1 = c("France", "Germany"),
  country2 = c("Sweden", "Austria"),
  year = 1992:1993,
  ioname = "EU"
)
igo_members

Extract Members of an IGO

Description

Extract all the countries belonging to an IGO on a specific date.

Usage

igo_members(ioname, year = NULL, status = "Full Membership")

Arguments

ioname
Any valid ioname of an IGO as specified on igo_year_format3. It could be also
a vector of IGOs.

year
Year to be assessed, an integer or an array of year. If NULL the latest year avail-
able of the IGO would be extracted.

status
Character or vector with the membership status to be extracted. See Details on
state_year_format3.

Value

A data.frame.

See Also

igo_year_format3, igo_search(), state_year_format3.

Examples

library(dplyr)
igo_members("EU", year = 1993) %>% as_tibble()
igo_members("EU") %>% as_tibble()
igo_members("NAFTA", year = c(1995:1998)) %>% as_tibble()

# Extract different status
igo_members("ACCT", status = c("Associate Membership", "Observer")) %>%
  as_tibble()

# States no members of the UN
igo_members("UN", status = "No Membership") %>%
  as_tibble()

# Vectorized
igo_members(c("NAFTA", "EU"), year = 1993) %>%
  as_tibble() %>%
**igo_recode_igoyear**

Helper functions to recode categories

**Description**

These functions convert the numerical code of `igo_year_format3` and `state_year_format3` into factors.

- `igo_recode_igoyear()` is intended to work with values on `igo_year_format3`.
- `igo_recode_stateyear()` is intended to work with values on `state_year_format3`.
- `igo_recode_dyadic()` is intended to work with values on `igo_dyadic()`.

**Usage**

```r
igo_recode_igoyear(x)
igo_recode_stateyear(x)
igo_recode_dyadic(x)
```

**Arguments**

- `x` Numerical value (or vector of values) to recode.

**Value**

The recoded values as factors.

**See Also**

Other datasets: `igo_year_format3`, `state_year_format3`, `states2016`
Examples

```r
data("igo_year_format3")

# Recode memberships for some countries

library(dplyr)

samp <- igo_year_format3 %>%
  select(ioname:year, spain, france) %>%
  filter(year > 2000) %>%
  as_tibble()

glimpse(samp)

# Recode

samp %>%
  mutate(
    spain = igo_recode_igoyear(spain),
    france = igo_recode_igoyear(france)
  ) %>%
  glimpse()
```

igo_search

**Search and Find an IGO**

**Description**

Search any IGO by name or string pattern.

**Usage**

```r
igo_search(pattern = NULL, exact = FALSE)
```

**Arguments**

- `pattern` regex pattern. If `NULL` the function returns a dataset with all the IGOs on `igo_year_format3`. Integer values are accepted.
- `exact` Logical. When `TRUE` only exact matches are returned.

**Details**

The information of each IGO is retrieved based on the last year available on `igo_year_format3`. An additional column `label` is provided. This column is a clean version of `longorgname`.

**Value**

A `data.frame`. 

See Also

igo_year_format3

Examples

# All values
library(dplyr)
all <- igo_search()
all %>% tibble()

# Search by pattern
igo_search("EU") %>%
  select(ionum:orgname) %>%
  tibble()

igo_search("EU", exact = TRUE) %>%
  select(ionum:orgname) %>%
  tibble()

# With integers
igo_search(10) %>%
  select(ionum:orgname) %>%
  tibble()

igo_search(10, exact = TRUE) %>%
  select(ionum:orgname) %>%
  tibble()

# Several patterns (regex style)
igo_search("NAFTA|UN|EU") %>%
  select(ionum:orgname) %>%
  tibble()

# Several patterns Exact (regex style)
igo_search("^NAFTA$|^UN$|^EU$") %>%
  select(ionum:orgname) %>%
  tibble()

igo_search_states

Finds codes and names of a state

Description

Extract all the memberships of a state on a specific date.

Usage

igo_search_states(state)
igo_state_membership

Arguments

state  Any valid name or code of a state as specified on states2016(). It could be also an array of states.

Value

A data.frame.

See Also

states2016().

Examples

library(dplyr)

igo_search_states("Spain") %>% as_tibble()
igo_search_states(c(20, 150)) %>% as_tibble()
igo_search_states("congo") %>% as_tibble()
igo_search_states(c("Germany", "papal states")) %>% as_tibble()
igo_search_states(c("FRN", "United Kingdom", 240, "italy")) %>% as_tibble()
**igo_year_format3**

**Value**

A data.frame.

**See Also**

igo_year_format3, igo_search_states(), states2016.

**Examples**

```r
# Memberships on two different dates
igo_state_membership("Spain", year = 1850)
igo_state_membership("Spain", year = 1870)
igo_state_membership("Spain", year = 1880:1882)

# Last year
igo_state_membership("ZAN")[, 1:7]

# Use codes to get countries
igo_state_membership("2", year = 1865)

# Extract different status
igo_state_membership("kosovo", status = c(
  "Associate Membership", "Observer",
  "Full Membership"
))

# Vectorized
igo_state_membership(c("usa", "spain"), year = 1870:1871)

# Use countrycodes package to get additional codes
if (requireNamespace("countrycode", quietly = TRUE)) {
  library(countrycode)
  IT <- igo_state_membership("Italy", year = 1880)
  IT$iso3c <- countrycode(IT$ccode, origin = "cown", destination = "iso3c")
  head(IT)
}
```

**Description**

Data on IGOs from 1815-2014, at the IGO-year level. Contains one record per IGO-year (with years listed at 5 year intervals through 1965, and annually thereafter).
Format

`data.frame` with 19,335 rows. Relevant fields:

- `ioname`: Short abbreviation of the IGO name.
- `orgname`: Full IGO name.
- `year`: Calendar Year.
- `afghanistan...zimbabwe`: status of that state in the IGO. See Details.
- `sdate`: start date (year) that the IGO started.
- `deaddate`: dead date (year) that the IGO dead.
- `longorgname`: a longer version of the IGOs name (including previous names)
- `ionum`: IGO id number in v2.1 and v3.0 of the data.
- `version`: COW version number.

See [Codebook Version 3 IGO Data](#) for full reference.

Details

Possible value of the status of that state in the IGO are:

<table>
<thead>
<tr>
<th>Category</th>
<th>Numerical Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Membership</td>
<td>0</td>
</tr>
<tr>
<td>Full Membership</td>
<td>1</td>
</tr>
<tr>
<td>Associate Membership</td>
<td>2</td>
</tr>
<tr>
<td>Observer</td>
<td>3</td>
</tr>
<tr>
<td>Missing data</td>
<td>-9</td>
</tr>
<tr>
<td>State Not System Member</td>
<td>-1</td>
</tr>
</tbody>
</table>

See `igo_recode_igoyear()` section for an easy way to recode the numerical values into factors.

Note

Raw data used internally by `igoR`.

Source

Intergovernmental Organizations (v3), The Correlates of War Project (IGO Data Stata Files).

References


See Also

Other datasets: `igo_recode_igoyear()`, `state_year_format3`, `states2016`
Examples

data("state_year_format3")

# Show a glimpse
library(dplyr)

state_year_format3 %>%
  select(ccode:afgec) %>%
  filter(year > 1990) %>%
  glimpse()

# Recode numerical to factors: with a sample
sample_state_year <- state_year_format3 %>%
  as_tibble() %>%
  select(ccode:afgec) %>%
  filter(year == 1990)

sample_state_year %>% glimpse()

# Recode
sample_state_year_recoded <- sample_state_year %>%
  mutate(across(-c(ccode:state), igo_recode_stateyear))

sample_state_year_recoded %>% glimpse()

states2016

State System Membership (v2016)

Description

The list of states with COW abbreviations and ID numbers, plus the field state from state_year_format3.

Format

data.frame with 243 rows. Relevant fields:

- **ccode**: COW country number.
- **stateabb**: COW state abbreviation (3 characters).
- **statenme**: COW state name.
- **styear...endday**: Fields to identify the beginning and the end of each tenure.
- **version**: Data file version number.
- **state**: Abbreviated state name as appear in state_year_format3.
Details

This data set contains the list of states in the international system as updated and distributed by the Correlates of War Project.

These data sets identify states, their standard Correlates of War "country code" or state number (used throughout the Correlates of War project data sets), state abbreviations, and dates of membership as states and major powers in the international system.

The Correlates of War project includes a state in the international system from 1816-2016 for the following criteria:

- **Prior to 1920** the entity must have had a population greater than 500,000 and have had diplomatic missions at or above the rank of charge d’affaires with Britain and France.
- **After 1920** the entity must be a member of the League of Nations or the United Nations, or have a population greater than 500,000 and receive diplomatic missions from two major powers.

Note

state variable added to original data to help comparison across datasets on this package.

Source

State System Membership (v2016), The Correlates of War Project.

References


See Also

Other datasets: igo_recode_igoyear(), igo_year_format3, state_year_format3

Examples

```r
# example code
data("states2016")
dplyr::glimpse(states2016)
```

Description

Data on IGOs from 1815-2014, at the country-year level. Contains one record per country-year (with years listed at 5 year intervals through 1965, and annually thereafter).
Format

data.frame with 15,557 rows. Relevant fields:

- ccode: COW country number, see states2016.
- year: Calendar Year.
- state: Abbreviated state name, identical to variable names in igo_year_format3.
- aaid...wassen: IGO variables containing information on state membership status. See Details.

See Codebook Version 3 IGO Data

Details

Possible value of the status of that state in the IGO are:

<table>
<thead>
<tr>
<th>Category</th>
<th>Numerical Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Membership</td>
<td>0</td>
</tr>
<tr>
<td>Full Membership</td>
<td>1</td>
</tr>
<tr>
<td>Associate Membership</td>
<td>2</td>
</tr>
<tr>
<td>Observer</td>
<td>3</td>
</tr>
<tr>
<td>Missing data</td>
<td>-9</td>
</tr>
<tr>
<td>IGO Not In Existence</td>
<td>-1</td>
</tr>
</tbody>
</table>

See igo_recode_stateyear() section for an easy way to recode the numerical values into factors.

Note

Raw data used internally by igoR.

Source

Intergovernmental Organizations (v3), The Correlates of War Project (IGO Data Stata Files)

References


See Also

countrycode::countrycode() to convert between different country code schemes.

Other datasets: igo_recode_igoyear(), igo_year_format3, states2016
Examples

data("state_year_format3")
dplyr::tibble(state_year_format3)
Index

* datasets
  - igo_recode_igoyear, 5
  - igo_year_format3, 9
  - state_year_format3, 12
  - states2016, 11
  - countrycode::countrycode(), 13
  - data.frame, 2–4, 6–11, 13
  - factors, 3, 5, 10, 13
  - igo_dyadic, 2
  - igo_dyadic(), 5
  - igo_members, 4
  - igo_recode_dyadic(igo_recode_igoyear), 5
    - igo_recode_dyadic(), 3, 5
    - igo_recode_igoyear, 5, 10, 12, 13
    - igo_recode_igoyear(), 5, 10
    - igo_recode_stateyear
      - (igo_recode_igoyear), 5
    - igo_recode_stateyear(), 5, 13
  - igo_search, 6
    - igo_search(), 2–4
    - igo_search_states, 7
    - igo_search_states(), 9
    - igo_state_membership, 8
    - igo_year_format3, 4–9, 9, 12, 13
  - regex, 6
  - state_year_format3, 3–5, 10–12, 12
  - states2016, 2, 3, 5, 9, 10, 11, 13
  - states2016(), 8