Package ‘states’

January 11, 2019

Type Package

Title Create Panels of Independent States

Version 0.2.2

Maintainer Andreas Beger <adbeger@gmail.com>

Description Create panel data consisting of independent states from 1816 to the present. The package includes the Gleditsch & Ward (G&W) and Correlates of War (COW) lists of independent states, as well as helper functions for working with state panel data and standardizing other data sources to create country-year/month/etc. data.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

Imports dplyr

Suggests testthat, ggplot2, stringr, knitr, rmarkdown, covr

RoxygenNote 6.1.1

VignetteBuilder knitr


BugReports https://github.com/andybega/states/issues

NeedsCompilation no

Author Andreas Beger [cre, aut] (<https://orcid.org/0000-0003-1883-3169>)

Repository CRAN

Date/Publication 2019-01-11 14:10:03 UTC

R topics documented:

.r WARNERS 2
 cowstates ........................................................................ 2
 gwstates ........................................................................ 3
Temporary helper function to warn about argument order change

Usage

`.warner(data, x)`

Arguments

data  data.frame
x  string

cowstates

Correlates of War list of independent states

Description

A list of independent states and microstates from 1816 on by the Correlates of War project.

Usage

cowstates

Format

Data frame

code  Gleditsch and Ward country code.
iso3c  ISO 3 character country code.
country_name  Long form country name
start  Country start of independence.
end  Country end of independence.
microstate  Logical flag for whether state is a microstates with less than 250,000 population.
**Source**

**Examples**
```r
data(cowstates)
head(cowstates)
```

---

**gwstates**  
*Gleditsch and Ward list of independent states*

**Description**
A list of independent states and microstates from 1816 on by Gleditsch and Ward

**Usage**
gwstates

**Format**
Data frame
- gwcode  Gleditsch and Ward country code.
- iso3c  ISO 3 character country code.
- country_name  Long form country name
- start  Country start of independence.
- end  Country end of independence.
- microstate  Logical flag for whether state is a microstates with less than 250,000 population.

**Source**

**Examples**
```r
data(gwstates)
head(gwstates)
```
id_date_sequence  Identify date sequences

Description
For correctly plotting country-time period spells

Usage
id_date_sequence(x, pd)

Arguments
x a Date sequence
pd what is the time aggregation period in the data?

Examples
library("ggplot2")
d1 <- as.Date("2018-01-01")
d2 <- as.Date("2025-01-01")
seq1 <- seq(d1, d2, by = "year")
data.frame(seq1, id=id_date_sequence(seq1, "year"))
# With a gap, should be two ids
df <- data.frame(date = seq1[-4], id=id_date_sequence(seq1[-4], "year"), cowcode = 999)
df

# The point is to plot countries with interrupted independence correctly:
df$y <- c(rep(1, 3), rep(2, 4))
df$id <- paste0(df$cowcode, df$id)
df
ggplot(df, aes(x = date, y = y, group = cowcode)) + geom_line()
ggplot(df, aes(x = date, y = y, group = id)) + geom_line()

plot_missing  Visualize missing and non-proper cases for state panel data

Description
Plot missing values by country and date, and additionally identify country-date cases that do or do not match an independent state list.
Usage

plot_missing(data, x = NULL, space = NULL, time = "date",
period = "year", statelist = "none", skip_labels = 5,
partial = "exact")

missing_info(data, x = NULL, space = NULL, time = "date",
period = "year", statelist = "none", partial = "exact")

Arguments

data State panel data frame
x Variable names(s), e.g. "x" or c("x1", "x2"). Default is NULL, in which case all columns expect the space and time ID columns will be used.
space Name of variable identifying state country codes. If NULL (default) and one of "gwcode" or "cowcode" is a column in the data, it will be used.
time Name of time identifier, the corresponding variables needs to be Date class. Default is "date".
period Temporal resolution character string, e.g. "year" or "month". See details in seq.Date. Default is "year".
statelist Check not only missing values, but presence or absence of observations against a list of independent states? "none" or "GW" or "COW".
skip_labels Only plot the label for every n-th country on the y-axis to avoid overplotting.
partial Option for how to handle edge cases where a state is independent for only part of a time period (year, month, etc.). Options include "exact", and "any". See state_panel() for details.

Details

missing_info provides the information that is plotted with plot_missing. The latter returns a ggplot, and thus can be chained with other ggplot functions as usual.

Value

plot_missing returns a ggplot2 object.
missing_info returns a data frame with components:
space Space identifier, with name equal to the "space" argument, e.g. "ccode".
time Time identifier, with name equal to the "time" argument, e.g. "date".
independent A logical vector, is the statelist argument is none, NA.
missing_value A logical vector indicating if that record has missing values
status The label used for plotting, combining the independence and missing value information for a case as appropriate.
Examples

# Create an example data frame with missing values
cy <- state_panel(as.Date("1980-06-30"), as.Date("2015-06-30"), by = "year",
useGW = TRUE)
cy$myvar <- rnorm(nrow(cy))
cy$myvar[sample(1:nrow(cy), nrow(cy)*.1, replace = FALSE)] <- NA
str(cy)

# Visualize missing values:
plot_missing("myvar", cy, "gwcode", "date", "year", "none")

# missing_info() generates the data underlying plot_missing():
head(missing_info("myvar", cy, "gwcode", "date", "year", "none"))

# if we specify a statelist to check against, 'independent' will have values
# now:
head(missing_info("myvar", cy, "gwcode", "date", "year", "GW"))

# Check data also against G&W list of independent states
head(missing_info("myvar", cy, "gwcode", "date", "year", "GW"))
plot_missing("myvar", cy, "gwcode", "date", "year", "GW")

# To check all variables:
# plot_missing(setdiff(colnames(df), "space", "time"), ...)

# Live example with Polity data
data("polity")
head(polity)
polity$date <- as.Date(paste0(polity$year, "-12-31"))
plot_missing(polity, "polity", "ccode", "date", "year", "COW")
# COW starts in 1816; Polity has excess data for several non-independent
# states after that date, and is missing coverage for several countries.

# The date option is relevant for years in which states gain or lose
# independence, so this will be slightly different:
polity$date <- as.Date(paste0(polity$year, "-01-01"))
plot_missing(polity, "polity", "ccode", "date", "year", "COW")

# plot_missing returns a ggplot2 object, so you can do anything you want
plot_missing(polity, "polity", "ccode", "date", "year", "COW") +
ggplot2::coord_flip()

---

### Polity

**Polity IV combined Polity scores**

**Description**

Polity scores reflect how democratic or autocratic countries are from a scale of -10 (autocratic) to 10 (democratic). There are also three special codes for foreign "interruption" (-66), anarchy (-77), and transition periods (-88).
The data are included here for as an example for use with the missing plot. Thus they do not contain all available Polity indicators, which are available at the Polity project website www.systemicpeace.org.

Usage

polity

Format

Data frame

code  Correlates of War (COW) country code.
year  Year of the observation.
polity  Combined Polity score.

Source


Examples

data("polity")
head("polity")

sfind  

Lookup country codes or names

Description

Helper to look up state list entries by country code or name

Usage

sfind(x, list = "both")

Arguments

x  The search string or number.
list  Which state list to search (both, GW, or COW only)
Examples

```r
# Works with either integer or strings
sfind(325)
sfind("ALG")
sfind("Algeria")

# Search strings are treated as regular expressions (see stringr::str_detect)
sfind("Germany")
sfind("German")
```

---

<table>
<thead>
<tr>
<th>states</th>
<th>State system membership</th>
</tr>
</thead>
</table>

Description

Create data based on the Gleditsch & Ward (G&W) or Correlates of War (COW) state system memberships lists. This is useful as a template for merging other sources of data that have conflicting sets of states.

Details

See static docs at https://andybeger.com/states and the source code at https://www.github.com/andybega/states

References


---

<table>
<thead>
<tr>
<th>state_panel</th>
<th>Create state panel data</th>
</tr>
</thead>
</table>

Description

Create panel data consisting of independent states in the international system.

Usage

```r
state_panel(start, end, by = "year", partial = "exact", useGW = TRUE)
```
Arguments

- **start**: Beginning date for data
- **end**: End date for data
- **by**: Temporal resolution, "year", "month", or "day".
- **partial**: Option for how to handle edge cases where a state is independent for only part of a time period (year, month, etc.). Options include "exact", and "any". See details.
- **useGW**: Use Gleditsch & Ward statelist or Correlates of War state system membership list.

Details

The partial option determines how to handle instances where a country gains or loses independence during a time period specified in the by option:

- "exact": the exact date in start is used for filtering
- "any": a state-period is included if the state was independent at any point in that period.

Examples

```r
# Basic usage with full option set specified:
gwlist <- state_panel("1991-01-01", "2015-01-01", by = "year",
                  partial = "any", useGW = TRUE)
head(gwlist, 3)
                  partial = "any", useGW = FALSE)
head(cowlist, 3)

# For yearly data, a proper date is not needed, and by = "year" and
# partial = "any" are inferred.
gwlist <- state_panel(1990, 1995)
sfind(265, list = "GW")
265 %in% gwlist$gwcode

# Partial
# Focus on South Sudan--is there a record for 2011, first year of independence?
data(gwstates)
dplyr::filter(gwstates, gwcode==626)
# No 2011 because SSD was not independent on January 1st 2011
x <- state_panel("2011-01-01", "2013-01-01", by = "year", partial = "exact")
dplyr::filter(x, gwcode==626)
# Includes 2011 because 12-31 date is used for filtering
x <- state_panel("2011-12-31", "2013-12-31", by = "year", partial = "exact")
dplyr::filter(x, gwcode==626)
# Includes 2011 because partial = "any"
x <- state_panel("2011-01-01", "2013-01-01", by = "year", partial = "any")
dplyr::filter(x, gwcode==626)
```
Index

*Topic datasets
  cowstates, 2
  gwstates, 3
  polity, 6
  .warner, 2

cowstates, 2

gwstates, 3

id_date_sequence, 4

missing_info(plot_missing), 4

plot_missing, 4

polity, 6

seq.Date, 5

sfind, 7

state_panel, 8

state_panel(), 5

states, 8

states-package (states), 8