

Package ‘eechidna’

May 23, 2016

Version 0.1

Title Exploring Election and Census Highly Informative Data Nationally for Australia

Description Data from the 2013 Australian Federal Election (House of Representatives) and the 2011 Australian Census. Includes tools for visualizing and analysing the data. This package incorporates data that is copyright Commonwealth of Australia (Australian Electoral Commission) 2016.

Depends R (>= 3.2.3)

Imports dplyr, ggplot2 (>= 2.0.0), tidyr, shiny, plotly (>= 3.5.2), ggthemes, shinyjs, methods

Suggests testthat, knitr, rmarkdown, maptools, purrr, sp, GGally, corrplot, broom, plyr, scales, readr, gridExtra

Maintainer Ben Marwick <benmarwick@gmail.com>

LazyData true

ByteCompile TRUE

License GPL (>= 2)

URL <https://github.com/ropenscilabs/eechidna>

VignetteBuilder knitr

RoxygenNote 5.0.1

NeedsCompilation no

Author Di Cook [aut],
Heike Hofmann [aut],
Rob Hyndman [aut],
Thomas Lumley [aut],
Ben Marwick [aut, cre],
Carson Sievert [aut],
Nicholas Tierney [aut],
Nathaniel Tomasetti [aut],
Fang Zhou [aut],
Commonwealth of Australia AEC [cph]

Repository CRAN

Date/Publication 2016-05-23 06:01:37

R topics documented:

eechidna-package	2
abs2011	3
aec2013_2cp	4
aec2013_2cp_electorate	5
aec2013_2pp	6
aec2013_2pp_electorate	7
aec2013_fp	8
aec2013_fp_electorate	9
aec_carto_f	10
aec_carto_join_f	11
aec_extract_f	12
circle	13
complete_color	14
dorling	15
getElectorateShapes	16
hexDat	17
launchApp	17
nat_data	18
nat_data_cart	19
nat_map	20
Index	21

eechidna-package	<i>Exploring Election and Census Highly Informative Data Nationally for Australia</i>
------------------	---

Description

Data from 2013 Australian Federal Election and 2011 Australian Census for each House of Representatives electorate, along with some tools for visualizing and analysing the data.

Author(s)

Di Cook, Heike Hofmann, Rob Hyndman, Thomas Lumley, Ben Marwick, Carson Sievert, Nicholas Tierney, Nathaniel Tomasetti, Fang Zhou.

abs2011

2011 Census data on all 150 electorates

Description

A dataset containing demographic and other information about each electorate. The data were obtained from the Australian Bureau of Statistics, and downloaded from <https://www.censusdata.abs.gov.au/datapacks/>.

Usage

abs2011

Format

A data frame with 150 rows with the following variables:

- ID: Commonwealth Electoral District identifier.
- Electorate: Name of electorate
- State: State containing electorate.
- Population: Total population of electorate.
- Area: Area of electorate in square kilometres.
- MedianIncome: Weekly median income of people within electorate (in \$).
- Unemployed: Percentage of people unemployed.
- Bachelor: Percentage of people whose highest qualification is a Bachelor degree.
- Postgraduate: Percentage of people whose highest qualification is a postgraduate degree.
- Christianity: Percentage of people affiliated with the Christian religion (of all denominations).
- Catholic: Percentage of people affiliated with the Catholic denomination.
- Buddhism: Percentage of people affiliated with the Buddhist religion.
- Islam: Percentage of people affiliated with the Islam religion.
- Judaism: Percentage of people affiliated with the Jewish religion.
- NoReligion: Percentage of people with no religious affiliation.
- Age00_04: Percentage of people aged 0-4.
- Age05_14: Percentage of people aged 5-9.
- Age15_19: Percentage of people aged 15-19.
- Age20_24: Percentage of people aged 20-24.
- Age25_34: Percentage of people aged 25-34.
- Age35_44: Percentage of people aged 35-44.
- Age45_54: Percentage of people aged 45-54.
- Age55_64: Percentage of people aged 55-64.

- Age65_74: Percentage of people aged 65-74.
- Age75_84: Percentage of people aged 75-84.
- Age85plus: Percentage of people aged 85 or higher.
- BornOverseas: Percentage of people born outside Australia.
- Indigenous: Percentage of people who are Indigenous.
- EnglishOnly: Percentage of people who speak only English.
- OtherLanguageHome: Percentage of people who speak a language other than English at home.
- Married: Percentage of people who are married.
- DeFacto: Percentage of people who are in a de facto marriage.
- FamilyRatio: Total number of families to total number of people (times 100).
- Internet: Percentage of people with home internet.
- NotOwned: Percentage of dwellings not owned (either outright or with a mortgage).

aec2013_2cp

2013 General election data for two candidate preferred votes for the House of Representatives for each polling place

Description

A dataset containing two candidate preferred vote counts, polling place locations, and other results for the House of Representatives from the 2013 Australian federal election. Includes the count of votes for the leading two candidates in the electorate after distribution of preferences for each polling place. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_2cp

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- PollingPlace: Polling place name
- PollingPlaceID: Polling place ID
- CandidateID: Candidate ID
- Surname: Candidate surname
- GivenNm: Candidate given name
- BallotPosition: Candidate's position on the ballot

- Elected: Whether the candidate was elected (Y/N)
- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast at the polling place for the candidate
- Swing: Percentage change since the last election

 aec2013_2cp_electorate

2013 General election data for two candidate preferred votes for candidates for the House of Representative for each electorate

Description

A dataset containing two candidate preferred vote counts, and other results for the House of Representatives from the 2013 Australian federal election. Includes the count of votes for the leading two candidates in the electorate after distribution of preferences. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_2cp_electorate

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- CandidateID: Candidate ID
- Surname: Candidate surname
- GivenNm: Candidate given name
- BallotPosition: Candidate's position on the ballot
- Elected: Whether the candidate was elected (Y/N)
- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast for the candidate
- AbsentVotes: Number of absentee votes cast for the candidate
- ProvisionalVotes: Number of provisional votes cast for the candidate

- PrePollVotes: Number of pre-poll votes cast for the candidate
- PostalVotes: Number of postal votes cast for the candidate
- TotalVotes: Total votes cast for the candidate
- Swing: Percentage change since the last election

 aec2013_2pp

2013 General election data for two party preferred votes for the House of Representatives for each polling place

Description

A dataset containing two party preferred vote counts, winning candidate names, polling place locations, and other results for the House of Representatives from the 2013 Australian federal election. Includes the count of votes for the Australian Labor Party and the count of votes for the Liberal-National Coalition for each polling place. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_2pp

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- PollingPlaceID: Polling place ID
- PollingPlace: Polling place name
- CandidateID: Candidate ID
- Surname: Candidate surname
- GivenNm: Candidate given name
- BallotPosition: Candidate's position on the ballot
- Elected: Whether the candidate was elected (Y/N)
- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast at the polling place for the candidate
- Swing:
- State: Abbreviation for state name
- PollingPlaceTypeID:

- Premises Nm:
- PremisesAddress1:
- PremisesAddress2:
- PremisesAddress3:
- PremisesSuburb:
- PremisesStateAb:
- PremisesPostCode:
- Latitude:
- Longitude:
- Australian.Labor.Party.Votes:
- Australian.Labor.Party.Percentage:
- Liberal.National.Coalition.Votes:
- Liberal.National.Coalition.Percentage:
- TotalVotes:

aec2013_2pp_electorate

2013 General election data for two party preferred votes for candidates for the House of Representative for each electorate

Description

A dataset containing two party preferred vote counts, winning candidate names, and other results for the House of Representatives from the 2013 Australian federal election. Includes the count of votes for the Australian Labor Party and the count of votes for the Liberal-National Coalition for each polling place. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_2pp_electorate

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- CandidateID: Candidate ID
- Surname: Candidate surname
- GivenNm: Candidate given name

- BallotPosition: Candidate's position on the ballot
- Elected: Whether the candidate was elected (Y/N)
- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast at the polling place for the candidate
- State: Abbreviation for state name
- Total_Australian_Labor_Party_Votes_per_electorate:
- Average_Australian_Labor_Party_Percentage_in_electorate:
- Total_Liberal_National_Coalition_Votes_per_electorate:
- Average_Liberal_National_Coalition_Percentage_in_electorate:
- Total_2pp_votes_per_electorate:

 aec2013_fp

2013 General election data for first preference votes for candidates for the House of Representatives for each polling place

Description

A dataset containing first preference vote counts, candidate names, polling place locations, and other results for the House of Representatives from the 2013 Australian federal election. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_fp

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- PollingPlaceID: Polling place ID
- PollingPlace: Polling place name
- CandidateID: Candidate ID
- Surname: Candidate surname
- GivenNm: Candidate given name
- BallotPosition: Candidate's position on the ballot
- Elected: Whether the candidate was elected (Y/N)

- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast at the polling place for the candidate
- Swing:
- State: Abbreviation for state name
- PollingPlaceTypeID:
- Premises Nm:
- PremisesAddress1:
- PremisesAddress2:
- PremisesAddress3:
- PremisesSuburb:
- PremisesStateAb:
- PremisesPostCode:
- Latitude:
- Longitude:
- uid:.

aec2013_fp_electorate *2013 General election data for first preference votes for candidates for the House of Representative for each electorate*

Description

A dataset containing first preference vote counts, candidate names, and other results for the House of Representatives from the 2013 Australian federal election. The data were obtained from the Australian Electoral Commission, and downloaded from <http://results.aec.gov.au/17496/Website/HouseDownloadsMenu-17496-csv.htm> and http://www.aec.gov.au/elections/federal_elections/2013/downloads.htm.

Usage

aec2013_fp_electorate

Format

A data frame with the following variables:

- ID: Electoral division ID
- Electorate: Electoral division name
- CandidateID: Candidate ID
- Surname: Candidate surname

- GivenNm: Candidate given name
- BallotPosition: Candidate's position on the ballot
- Elected: Whether the candidate was elected (Y/N)
- HistoricElected:
- PartyAb: Abbreviation for political party name
- PartyNm: Political party name
- OrdinaryVotes: Number of ordinates votes cast at the polling place for the candidate
- State: Abbreviation for state name

 aec_carto_f

aec_carto_f - run dorling ondata centers

Description

The dorling algorithm creates a non-contiguous cartogram by shifting circles to alleviate overlap, while roughly maintaining geographic proximity.

Usage

```
aec_carto_f(aec_data_sub, polygon.vertex = 6, name.text = TRUE,
  dist.ratio = dist.ratio, iteration = 100, xlab = "", ylab = "", ...)
```

Arguments

<code>aec_data_sub</code>	subset of data with centroids of electoral divisions
<code>polygon.vertex</code>	The number of vertice of the circle. Default to be 100. If <code>polygon.vertex=4</code> then diamonds applies. If <code>polygon.vertex=6</code> , then hexagon applies.
<code>name.text</code>	whether to print the region names on the circles or polygons.
<code>dist.ratio</code>	The threshold to determine whether an attract force is added. It is applied to the ratio of the distance between two centroids and the sum of the two radii.
<code>iteration</code>	The limit of the number of iterations. Default to be 9999.
<code>xlab</code>	Label for dorling x axis, intermediate drawing
<code>ylab</code>	Label for dorling y axis, intermediate drawing
<code>...</code>	arguments to dorling function

Examples

```

library(dplyr)
library(ggplot2)
data(nat_map)
data(nat_data)
adelaide <- aec_extract_f(nat_data, ctr=c(138.6, -34.9), expand=c(2,3))
adelaide_carto <- aec_carto_f(adelaide) %>% rename(id=region)
ggplot(data=nat_map) +
  geom_path(aes(x=long, y=lat, group=group, order=order),
            colour="grey50") +
  geom_point(data=adelaide_carto, aes(x=x, y=y), size=4, alpha=0.4,
            colour="#f0027f") +
  xlim(c(136, 140)) + ylim(-36, -33) +
  coord_equal()
adelaide_all <- merge(adelaide, adelaide_carto, by="id")
ggplot(data=nat_map) +
  geom_path(aes(x=long, y=lat, group=group, order=order),
            colour="grey50") +
  geom_point(data=adelaide_all, aes(x=long_c, y=lat_c), size=2, alpha=0.4,
            colour="#f0027f") +
  geom_point(data=adelaide_all, aes(x=x, y=y), size=2, alpha=0.4,
            colour="#f0027f") +
  geom_segment(data=adelaide_all,
              aes(x=long_c, xend=x, y=lat_c, yend=y), colour="#f0027f") +
  xlim(c(136, 140)) + ylim(-37, -33) +
  coord_equal()

```

aec_carto_join_f *aec_carto_join_f - bind the cartogram coordinates to original data*

Description

Add the cartogram locations as new variables to original data and make any of these that were not made equal to the original centroids

Usage

```
aec_carto_join_f(aec_data, aec_carto)
```

Arguments

aec_data	subset of data with centroids of electoral divisions
aec_carto	centers

Examples

```

library(dplyr)
library(ggplot2)
data(nat_map)
data(nat_data)
cities <- list(c(151.2, -33.8), # Sydney
c(153.0, -27.5), # Brisbane
c(145.0, -37.8), # Melbourne
c(138.6, -34.9), # Adelaide,
c(115.9, -32.0)) # Perth
expand <- list(c(2,3.8), c(2,3), c(2.6,4.1), c(4,3), c(12,6))
nat_carto <- purrr::map2(.x=cities, .y=expand,
.f=aec_extract_f, aec_data=nat_data) %>%
  purrr::map_df(aec_carto_f) %>%
  mutate(region=as.integer(as.character(region))) %>%
  rename(id=region)
nat_data_cart <- aec_carto_join_f(nat_data, nat_carto)
# Map theme
theme_map <- theme_bw()
theme_map$line <- element_blank()
theme_map$strip.text <- element_blank()
theme_map$axis.text <- element_blank()
theme_map$plot.title <- element_blank()
theme_map$axis.title <- element_blank()
theme_map$panel.border <- element_rect(colour = "grey90", size=1, fill=NA)

ggplot(data=nat_map) +
  geom_polygon(aes(x=long, y=lat, group=group, order=order),
fill="grey90", colour="white") +
  geom_point(data=nat_data_cart, aes(x=x, y=y), size=2, alpha=0.4,
colour="#572d2c") +
  geom_text(data=nat_data_cart, aes(x=x, y=y, label=id), size=0.5) +
  coord_equal() + theme_map

```

aec_extract_f

aec_extract_f - extract subsets geographically

Description

The dorling algorithm doesn't work on the entire country, because it is very clustered at the cities. To get a reasonable cartogram we need to extract out the cities, expand these with dorling independently. This function does the extraction.

Usage

```
aec_extract_f(aec_data, ctr = c(151.2, -33.8), expand = c(3, 4.5), ...)
```

Arguments

aec_data	data with centroids of electoral divisions
ctr	centroids of subset
expand	how large a chunk to cut out
...	other arguments

Examples

```
library(dplyr)
library(ggplot2)
data(nat_map)
data(nat_data)
adelaide <- aec_extract_f(nat_data, ctr=c(138.6, -34.9), expand=c(2,3))
ggplot(data=nat_map) +
  geom_polygon(aes(x=long, y=lat, group=group, order=order),
    fill="grey90", colour="white") +
  geom_point(data=adelaide, aes(x=long_c, y=lat_c), size=2, alpha=0.4,
    colour="#f0027f") +
  xlim(c(136, 142)) + ylim(-36, -33) +
  coord_equal()
```

circle

*Draw a circle***Description**

##' From <https://github.com/chxy/cartogram/blob/master/R/dorling.R> Not exported here, but needed for aec_carto_f

Usage

```
circle(xvec, yvec, rvec, vertex = 100, border = 1, col = NULL,
  add = TRUE, square = FALSE, ...)
```

Arguments

xvec	X-coordinates
yvec	Y-coordinates
rvec	Radii
vertex	The number of vertices of the circle
border	Color of border
col	Color to render in circle
add	Whether the circles are added to another plot.
square	A logical value to determine whether to draw squares.
...	other things

Details

This function is used to compute the locations of the circle border and draw multiple circles. It borrows the code from `plotrix::draw.circle`

Examples

```
## Not run:
x=y=1:5
r=5:1/5
circle(x,y,r,add=FALSE,asp=1)
circle(x,y,r,vertex=6,add=TRUE) # hexagon
circle(x,y,r,vertex=4,add=TRUE) # diamond
circle(x,y,r,square=TRUE,add=TRUE) # square

## End(Not run)
```

complete_color	<i>Auto complete (or cut) a vector to a fixed length</i>
----------------	--

Description

From <https://github.com/chxy/cartogram/blob/master/R/dorling.R> Not exported here, but needed for `aec_carto_f`

Usage

```
complete_color(cl, targetlen)
```

Arguments

<code>cl</code>	a vector
<code>targetlen</code>	the target length

Value

a vector of completed `cl` with length `n`

Examples

```
## Not run:
complete_color('red',5)
complete_color(c('red','blue'),5)
complete_color(c('red','blue','green','yellow','pink','grey'),5)

## End(Not run)
```

 dorling

Produce a Pseudo-Dorling Cartogram.

Description

From <https://github.com/chxy/cartogram/blob/master/R/dorling.R> Not exported here, but needed for `aec_carto_f`

Usage

```
dorling(name, centroidx, centroidy, density, nbr = NULL,
        shared.border = NULL, color = NULL, tolerance = 0.1, dist.ratio = 1.2,
        iteration = 9999, polygon.vertex = 100, animation = FALSE,
        sleep.time = 0.3, nbredge = ifelse(is.null(nbr), FALSE, TRUE),
        name.text = TRUE, ggplot2 = FALSE, ...)
```

Arguments

<code>name</code>	A vector of region names.
<code>centroidx</code>	A vector of x-coordinates of the regions.
<code>centroidy</code>	A vector of y-coordinates of the regions.
<code>density</code>	A vector of the variable of interest. It will be used as the radii of the circles.
<code>nbr</code>	A list of the neighbors of every region. Each element is a vector of all the neighbor names of a region. If <code>nbr=NULL</code> , then it is assumed that no region has any neighbors. If <code>nbr</code> is not <code>NULL</code> , then names should be given to all the elements of the list, for matching the neighbors with the host region name, otherwise the parameter "name" (a character vector) will be used as the element names of <code>nbr</code> . Besides, any values in <code>nbr</code> that are not in "name" will be removed. The length of <code>nbr</code> could be different from the length of "name", but any element in <code>nbr</code> whose name is not in "name" will be removed too.
<code>shared.border</code>	A matrix of the counts of shared borders, typically generated from the function <code>border_summary_length()</code> . It is used to scale the attract force.
<code>color</code>	a vector of color to fill in the circles or polygons. Auto-completed if the length does not match with <code>name</code> .
<code>tolerance</code>	Tolerant value for the sum of overlapped radii.
<code>dist.ratio</code>	The threshold to determine whether an attract force is added. It is applied to the ratio of the distance between two centroids and the sum of the two radii.
<code>iteration</code>	The limit of the number of iterations. Default to be 9999.
<code>polygon.vertex</code>	The number of vertice of the circle. Default to be 100. If <code>polygon.vertex=4</code> then diamonds applies. If <code>polygon.vertex=6</code> , then hexagon applies.
<code>animation</code>	Whether to show the movements of centroids.
<code>sleep.time</code>	Only works when <code>animation=TRUE</code> .
<code>nbredge</code>	whether to draw the lines between neighbor regions.

name.text	whether to print the region names on the circles or polygons.
ggplot2	whether to use ggplot2 to draw the cartogram.
...	other things

getElectorateShapes *Extract shapefiles (of Australian electorates) from shp file*

Description

Extract polygon information and demographics for each of Australia's electorates. The map and data corresponding to the shapefiles of the 2013 Australian electorates (available at http://www.aec.gov.au/Electorates/gis/gis_datadownload.htm) are part of this package as nat_map.rda and nat_data.rda in the data folder. The function will take several minutes to complete.

Usage

```
getElectorateShapes(shapeFile, keep = 0.05)
```

Arguments

shapeFile	path to the shp file
keep	percent of polygon points to keep, the default is set to 5%.

Value

list with two data frames: map and data; 'map' is a data set with geographic latitude and longitude, and a grouping variable to define each entity. The 'data' data set consists of demographic or geographic information for each electorate, such as size in square kilometers or corresponding state. Additionally, geographic latitude and longitude of the electorate's centroid are added.

Examples

```
## Not run:
url <- "national-esri-16122011/COM20111216_ELB_region.shp"
electorates <- getElectorateShapes(url)
library(ggplot2)
ggplot(data=electorates$data) +
  geom_map(aes(fill=AREA_SQKM, map_id=id), map=electorates$map) +
  expand_limits(
    x=range(electorates$map$long),
    y=range(electorates$map$lat)
  )

## End(Not run)
```

hexDat	<i>Electorate hexagon data in a tidy form</i>
--------	---

Description

Electorate hexagon data in a tidy form

Usage

hexDat

Format

A data frame

References

Thomas Lumley

See Also

Thomas Lumley

Examples

```
data(hexDat)
library(plotly)
p <- ggplot(hexDat, aes(xcent, ycent, text = Electorate)) +
  geom_hex(stat = "identity") +
  lims(x=c(-80, 8), y=c(-40, 50))
ggplotly(p, tooltip = "text")
```

launchApp	<i>Shiny app for exploring census and electorate data</i>
-----------	---

Description

Shiny app for exploring census and electorate data

Usage

```
launchApp(age = c("Age00_04", "Age05_14", "Age15_19", "Age20_24", "Age25_34",
  "Age35_44", "Age45_54", "Age55_64", "Age65_74", "Age75_84", "Age85plus"),
  religion = c("Christianity", "Catholic", "Buddhism", "Islam", "Judaism",
  "NoReligion"), other = c("Population", "MedianIncome", "Unemployed",
  "Bachelor", "Postgraduate", "BornOverseas", "Indigenous", "EnglishOnly",
  "OtherLanguageHome", "Married", "DeFacto", "FamilyRatio", "Internet",
  "NotOwned"), palette = c("#1B9E77", "#F0027F", "#E6AB02", "#66A61E",
  "#7570B3", "#D95F02", "#3690C0"))
```

Arguments

age	Age variables to show (by default, all of them are shown)
religion	Religion variables to show (by default, all of them are shown)
other	Other census variables to show (by default, all of them are shown)
palette	a named character vector of selection colors. The vector names are used as the display in the drop-down control.

Author(s)

Carson Sievert

Examples

```
## Not run:
launchApp(
  age = c("Age20_24", "Age85plus"),
  religion = c("Christianity", "Catholic", "NoReligion"),
  other = c("Unemployed", "Population", "MedianIncome")
)

## End(Not run)
```

nat_data

(Very small) Map of Australian Electorate from 2013

Description

A dataset containing the map of the all 150 Australian electorates using the 2013 boundaries of the electorates (and downsampled to a 0.5% file to allow fast plotting). The data were obtained from the Australian Electoral Commission, and downloaded from http://www.aec.gov.au/Electorates/gis/gis_datadownload.htm. Data of the Australian Electorate from 2013

Usage

```
nat_data
```

Format

A data frame with 150 rows with the following variables:

- id: numeric identifier that links the electorate with the corresponding polygon in 'nat_map'.
- ELECT_DIV: Electorate division name
- STATE: abbreviation of the state name
- NUMCCDS: AEC variable that might be filled with meaning or a description down the road
- AREA_SQKM: combined square kilometers of each electorate

Details

A dataset containing some demographic information for each of the 150 Australian electorates. The data were obtained from the Australian Electoral Commission, and downloaded from http://www.aec.gov.au/Electorates/gis/gis_datadownload.htm. The data is published

Examples

```
data(nat_data)
nat_map$region <- nat_map$ELECT_DIV
data(abs2011)
abs2011$region <- abs2011$Electorate
both <- intersect(unique(abs2011$region), unique(nat_map$region))
data(nat_data)
library(ggplot2)
library(ggthemes)
ggplot(aes(map_id=region), data=subset(abs2011, region %in% both)) +
  geom_map(aes(fill=MedianIncome), map=subset(nat_map, region %in% both)) +
  expand_limits(x=nat_map$long, y=nat_map$lat) +
  theme_map()
```

nat_data_cart

This is the same as nat_data, but with a few more cols. We can probably replace nat_dat with this one

Description

This is the same as nat_data, but with a few more cols. We can probably replace nat_dat with this one

Usage

```
nat_data_cart
```

Format

An object of class `data.frame` with 150 rows and 10 columns.

`nat_map`*Map of Australian Electorate from 2013*

Description

A dataset containing the map of the all 150 Australian electorates using the 2013 boundaries of the electorates (and downsampled to a 5% file to allow fast plotting). The data were obtained from the Australian Electoral Commission, and downloaded from http://www.aec.gov.au/Electorates/gis/gis_datadownload.htm.

Usage`nat_map`**Format**

An object of class `data.frame` with 27595 rows and 9 columns.

Examples

```
data(nat_map)
# choropleth map with Census data
nat_map$region <- nat_map$ELECT_DIV
data(abs2011)
abs2011$region <- abs2011$Electorate
library(ggplot2)
library(ggthemes)
both <- intersect(unique(abs2011$region), unique(nat_map$region))
ggplot(aes(map_id=region), data=subset(abs2011, region %in% both)) +
  geom_map(aes(fill=MedianIncome), map=subset(nat_map, region %in% both)) +
  expand_limits(x=nat_map$long, y=nat_map$lat) +
  theme_map()
```

Index

*Topic **datasets**

- abs2011, [3](#)
- aec2013_2cp, [4](#)
- aec2013_2cp_electorate, [5](#)
- aec2013_2pp, [6](#)
- aec2013_2pp_electorate, [7](#)
- aec2013_fp, [8](#)
- aec2013_fp_electorate, [9](#)
- hexDat, [17](#)
- nat_data, [18](#)
- nat_data_cart, [19](#)
- nat_map, [20](#)

*Topic **package**

- eechidna-package, [2](#)

- abs2011, [3](#)
- aec2013_2cp, [4](#)
- aec2013_2cp_electorate, [5](#)
- aec2013_2pp, [6](#)
- aec2013_2pp_electorate, [7](#)
- aec2013_fp, [8](#)
- aec2013_fp_electorate, [9](#)
- aec_carto_f, [10](#)
- aec_carto_join_f, [11](#)
- aec_extract_f, [12](#)

- circle, [13](#)

- complete_color, [14](#)

- dorling, [15](#)

- eechidna (eechidna-package), [2](#)

- eechidna-package, [2](#)

- getElectorateShapes, [16](#)

- hexDat, [17](#)

- launchApp, [17](#)

- nat_data, [18](#)

- nat_data_cart, [19](#)

- nat_map, [20](#)