Package 'ejscreen'

August 7, 2015

Type Package

Title EJSCREEN Tools for US EPA Environmental Justice Mapping and Screening

Version 0.1

Date 2015-07-19

Author info@ejanalysis.com

Maintainer ejanalyst <info@ejanalysis.com>

Description Data and tools related to the United States Environmental Protection Agency's screening and mapping tool for environmental justice, EJSCREEN

License MIT + file LICENSE

LazyData TRUE

URL http://ejanalysis.github.io

http://www.ejanalysis.com/

http://www.epa.gov/ejscreen

R topics documented:

`ejscreen` *Tools for EJSCREEN, US EPA's Environmental Justice (EJ) Screening and Mapping Tool*

Description

This R package provides tools related to environmental justice (EJ) analysis, specifically related to the United States Environmental Protection Agency (EPA) screening and mapping/GIS tool called EJSCREEN. See http://www.epa.gov/ejscreen This package facilitates development of the EJSCREEN dataset, based on user-provided environmental indicators. The resulting dataset is a data.frame that contains data on demographics (e.g., percent of residents who are low-income) and user-provided local environmental indicators (e.g., an air quality index), and calculated indicators called EJ Indexes, which combine environmental and demographic indicators. The dataset also provides each key indicator as a national population-percentile that represents what percentage of the US population have equal or lower raw values for the given indicator. The dataset has one row per spatial location (e.g., Census block group).

Details

Key functions include

- ejscreen.create
- ejscreen.lookuptables
- Various functions from the **ejanalysis** package.

References

http://ejanalysis.github.io
http://www.ejanalysis.com/
http://www.epa.gov/ejscreen

---

ejscreen.acs.calc *Create Calculated EJSCREEN Variables*

Description

Use specified formulas to create calculated, derived variables such as percent low income. Relies upon calc.fields from **analyze.stuff** package.

Usage

```
ejscreen.acs.calc(bg, folder = getwd(), keep.old, keep.new, formulafile)
```

Arguments

bg	Data.frame of raw demographic data counts, and environmental indicators, for each block group, such as population or number of Hispanics.
folder	Default is getwd(). Specifies path for where to read from (if formulafile specified) and write to.
keep.old	Vector of variables names from names(bg), indicating which to return (retain, not drop). Default is to keep only the ones that match the list of default names in this code.
keep.new	Vector of variables names of new created variables, indicating which to return (retain, not drop). Default is to keep all.
formulafile	Name of optional csv file with column called formula, providing formulas as character fields. If not specified, function loads this as data(ejscreenformulas).

Value

Returns a data.frame with some or all of input fields, plus calculated new fields.

Examples

```
set.seed(99)
 envirodata=data.frame(FIPS=analyze.stuff::lead.zeroes(1:1000, 12),
   air=rlnorm(1000), water=rlnorm(1000)*5, stringsAsFactors=FALSE)
 demogdata=data.frame(FIPS=analyze.stuff::lead.zeroes(1:1000, 12),
   pop=rnorm(n=1000, mean=1400, sd=200), mins=runif(1000, 0, 800),
   num2pov=runif(1000, 0,500), stringsAsFactors=FALSE)
 demogdata$povknownratio <- demogdata$pop
 x=ejscreen.acs.calc(bg=demogdata)
```

---

ejscreen.acs.rename *Rename Fields of ACS Data for Use in EJSCREEN*

Description

Start with raw counts from demographic survey data, and environmental data, and rename fields to use friendly variable names.

Usage

```
ejscreen.acs.rename(acsraw, folder = getwd(), formulafile)
```

Arguments

acsraw	Data.frame of raw data counts for each block group, such as population or number of Hispanics.
folder	Default is getwd(). Specifies path for where to read from (if formulafile specified) and write to.
formulafile	Default if this is blank is to use data(ejscreenformulas). Otherwise filename must be specified. If not specified, function loads this as data().

Value

Returns a data.frame with some or all of input fields, plus calculated new fields.

Examples

```
# (no examples yet)
```

`ejscreen.create` *Create EJSCREEN Dataset from Environmental Indicators*

Description

Start with raw environmental indicator data, and create full EJSCREEN dataset. This code also contains an outline of steps involved.

Usage

```
ejscreen.create(e, acsraw, folder = getwd(), keep.old, ...)
```

Arguments

`e`	Data.frame of raw data for environmental indicators, one row per block group, one column per indicator.
`acsraw`	Optional data.frame of raw demographic indicators. Downloaded if not provided as parameter.
`folder`	Optional, default is getwd(). Passed to `get.acs` if demog data must be downloaded. Passed to but not currently used by ejscreen.acs.rename which uses `change.fieldnames` in **analyze.stuff** package. Not currently passed to ejscreen.acs.calc which uses `calc.fields` in **analyze.stuff** package.
`keep.old`	optional vector of colnames from e that are to be used/returned. For nondefault colnames, this must be used.
`...`	additional optional parameters to pass to `get.acs` (such as end.year='2013' – otherwise uses default year used by `get.acs`)

Details

**Note that if non-default fieldnames are used in e and/or acsraw, those will cause problems in ejscreen.acs.calc which assumes the standard fields are to be returned! That can be handled using parameter keep.old

Value

Returns a data.frame with full ejscreen dataset of environmental and demographics indicators, and EJ Indexes, as raw values, US percentiles, and text for popups. Output has one row per block group.

Examples

```
## Not run:
 set.seed(99)
 envirodata=data.frame(FIPS=analyze.stuff::lead.zeroes(1:1000, 12),
   air=rlnorm(1000), water=rlnorm(1000)*5, stringsAsFactors=FALSE)
 demogdata=data.frame(FIPS=analyze.stuff::lead.zeroes(1:1000, 12),
   pop=rnorm(n=1000, mean=1400, sd=200), mins=runif(1000, 0, 800),
   num2pov=runif(1000, 0,500), stringsAsFactors=FALSE)
 demogdata$povknownratio <- demogdata$pop
 # downloads ACS demographics and combines with user provided envirodata:
 # bg1=ejscreen.create(envirodata, mystates=c('de','dc'))
 # currently does not work for nonstandard colnames unless keep.old used as follows (work in progress):
```

```
 y=ejscreen.create(e=envirodata, acsraw=demogdata,
  keep.old = c(names(envirodata), names(demogdata)))

## End(Not run)
```

`ejscreen.lookuptables` *Create EJSCREEN Lookup Tables of Pop. Percentiles by Zone*

Description

Start with raw environmental, demographic, and EJ indicator data, and write as csv files to disk a series of lookup tables that show population percentiles and mean values for each indicator.

Usage

```
ejscreen.lookuptables(x, weights, zone, folder = getwd(),
  missingcode = -9999999)
```

Arguments

`x`	Data.frame of indicators, one row per block group, one column per indicator.
`weights`	Weights for percentiles – Default is population count to provide population percentiles.
`zone`	NOT IMPLEMENTED HERE - HELPER FUNCTIONS ASSUME STATES AND REGIONS ARE ZONES NEEDED
`folder`	Default is getwd() - specifies where to save the csv files.
`missingcode`	Leave this unspecified if missing values are set to NA in the input data. Default is -9999999 (but if already NA then do not specify anything for this). The number or value in the input data that designates a missing value.

Value

Creates lookup tables saved as csv files to specified folder.

Examples

```
# (no examples yet)
```

ejscreenformulas *EJSCREEN 2015 Formulas and Fieldnames*

Description

This provides fieldnames and formulas required by the **ejscreen** package.

Usage

```
data('ejscreenformulas')
```

Format

A data.frame:

> str(ejscreenformulas)
'data.frame': 470 obs. of 8 variables:

- $ gdbfieldname : chr NA NA NA NA ...
- $ Rfieldname : chr "ageunder5m" "age5to9m" "age10to14m" "age15to17m" ...
- $ acsfieldname : chr "B01001.003" "B01001.004" "B01001.005" "B01001.006" ...
- $ type : chr "ACS" "ACS" "ACS" "ACS" ...
- $ glossaryfieldname: chr NA NA NA NA ...
- $ formula : chr NA NA NA NA ...
- $ acsfieldnamelong : chr "Under 5 years|SEX BY AGE" "5 to 9 years|SEX BY AGE" "10 to 14 years|SEX BY AGE" "15 to 17 years|SEX BY AGE" ...
- $ universe : chr "Universe: Total population" "Universe: Total population" "Universe: Total population" "Universe: Total population" ...

Source

See related Technical Documentation at http://www.epa.gov/ejscreen

hello *Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

Index