Package: AOI (via r-universe)

December 4, 2024

Type Package

Title Areas of Interest

Version 0.3.0

BugReports https://github.com/mikejohnson51/AOI/issues

Description A consistent tool kit for forward and reverse geocoding
and defining boundaries for spatial analysis.

Depends R(>= 3.5.0)

Imports datasets, dplyr, fipio, htmlwidgets, jsonlite, leaflet,
leaflet.extras, rnaturalearth, rvest, sf, shiny, terra,
tidygeocoder, units

Suggests climateR, distill, FedData, gdalio, knitr, mapview,
nhdplusTools, osmdata, raster, rmarkdown, testthat, zonal

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

URL https://github.com/mikejohnson51/AOI/

VignetteBuilder knitr

Config/pak/sysreqs libgdal-dev gdal-bin libgeos-dev make libicu-dev
libpng-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev
libudunits2-dev zlib1g-dev

Repository https://mikejohnson51.r-universe.dev

RemoteUrl https://github.com/mikejohnson51/AOI

RemoteRef HEAD

RemoteSha f821d499b80371c7c389937072ea041312a8efcd

Contents

`.domain` *Build Domain*

Description

Build Domain

Usage

```
.domain(xy, wh, units = default_units, crs = default_crs, bbox = FALSE)
```

Arguments

`xy`	a orign specified as a numeric vector
`wh`	width and height (can be a single number) in units (see units arg)
`units`	units of wh expansion
`crs`	output crs
`bbox`	return bbox object?

Value

vector or sf object

`.geocode` *.geocode*

Description

.geocode

Usage

```
.geocode(
  geo,
  pt = FALSE,
  bbox = FALSE,
  all = FALSE,
  method = default_method,
  crs = default_crs
)
```

Arguments

`geo`	`character`. Place name(s)
`pt`	`logical`. If TRUE point geometry is created.
`bbox`	`logical`. If TRUE bounding box geometry is created
`all`	`logical`. If TRUE the point, bbox and xy representations are returned as a list
`method`	the geocoding service to be used. See ?tidygeocoder::geocode
`crs`	desired CRS. Defaults to AOI::default_crs

Value

a data.frame, sf object, or vector

`alt_page` *Alternate Page Finder*

Description

Find linked pages to a wikipedia call

Usage

```
alt_page(loc, pt = FALSE)
```

Arguments

`loc`	a wikipedia structured call
`pt`	`logical`. If TRUE point geometery is appended to the returned list()

Value

at minimum a data.frame of lat, long

Examples

```
## Not run:
alt_page("Twin_towers")

## End(Not run)
```

`aoi_describe` *Describe an AOI*

Description

Describe a spatial (sf/sp/raster) object in terms of a reproducible AOI (e.g. `aoi_get`) parameters.

Usage

```
aoi_describe(AOI)
```

Arguments

`AOI`	a spatial object (`raster`, `sf`, `sp`).

Value

a data.frame of AOI descriptors

Examples

```
{
  fname <- system.file("shape/nc.shp", package = "sf")
  nc <- sf::read_sf(fname)
  aoi_describe(AOI = nc[1, ])
}
```

`aoi_draw` *AOI Draw*

Description

Interactively draw an Area of Interest (AOI) using a shiny app. Once an object is drawn and the "Save AOI" button pressed, a new sf object called 'aoi' will appear in your environment.

Usage

```
aoi_draw()
```

Value

An sf object called 'aoi'.

Examples

```
## Not run:
aoi_draw()

## End(Not run)
```

`aoi_ext` *AOI extent*

Description

Build an extent surrinding by location point (longitude, latitude) based on a width and height.

Usage

```
aoi_ext(
  geo = NULL,
  xy = NULL,
  wh = NULL,
  units = default_units,
  crs = default_crs,
  bbox = FALSE
)
```

Arguments

`geo`	an origion specificed by a name
`xy`	a orign specified as a numeric vector
`wh`	width and height (can be a single number) in units (see units arg)
`units`	units of wh expansion
`crs`	output crs
`bbox`	return bbox object?

Value

vector or sf object

`aoi_get` *Get Area of Interest (AOI) geometry*

Description

Generate a spatial geometry from:

Usage

```
aoi_get(
  x = NULL,
  country = NULL,
  state = NULL,
  county = NULL,
  fip = NULL,
  zipcode = NULL,
  union = FALSE
)
```

Arguments

`x`	`sf`, or a `Spat*` object
`country`	`character`. Full name, ISO 3166-1 2 or 3 digit code. Not case sensitive. Data comes from Natural Earth.
`state`	`character`. Full name or two character abbreviation. Not case sensitive. If `state = 'conus'`, the lower 48 states will be returned. If `state = 'all'`, all states will be returned.
`county`	`character`. County name(s). Requires `state` input. Not case sensitive If 'all' then all counties in a state are returned
`fip`	a 2 or 5 digit US fip code
`zipcode`	a US zip code. Will return a centroid.
`union`	`logical`. If TRUE objects are unioned into a single object

Value

a sf geometry projected to *EPSG:4326*.

Examples

```
## Not run:
# Get AOI for a country
aoi_get(country = "Brazil")
# Get AOI defined by a state(s)
aoi_get(state = "CA")
aoi_get(state = c("CA", "nevada"))

# Get AOI defined by all states, or the lower 48
aoi_get(state = "all")
aoi_get(state = "conus")

# Get AOI defined by state & county pair(s)
aoi_get(state = "California", county = "Santa Barbara")
aoi_get(state = "CA", county = c("Santa Barbara", "ventura"))

# Get AOI defined by state & all counties
aoi_get(state = "California", county = "all")

## End(Not run)
```

---

aoi_inside *Is Inside*

Description

A check to see if one object is inside another

Usage

```
aoi_inside(AOI, obj, total = TRUE)
```

Arguments

`AOI`	object 2
`obj`	object 1
`total`	boolean. If `TRUE` then check if obj is completely inside the AOI (and vice versa: order doesn't matter). If `FALSE`, then check if at least part of obj is in the AOI.

Value

boolean value

aoi_map *Generate Leafet map and tool set for AOI*

Description

Provides a precanned leaflet layout for checking, and refining AOI queries. Useful `leaflet` tools allow for the marking of points, measuring of distances, and panning and zooming.

Usage

```
aoi_map(AOI = NULL, returnMap = FALSE)
```

Arguments

`AOI`	any spatial object (`raster`, `sf`, `sp`). Can be piped (%>%) from `aoi_get`. If `AOI = NULL`, base map of CONUS will be returned.
`returnMap`	`logical`. If `FALSE` (default) the input AOI is returned and the leaflet map printed. If `TRUE` the leaflet map is returned and printed.

Value

a `leaflet` html object

Examples

```
## Not run:
## Generate an empty map:
aoi_map()

## Check a defined AOI:
AOI <- getAOI(clip = list("UCSB", 10, 10))
aoi_map(AOI)

## Chain to AOI calls:
getAOI(clip = list("UCSB", 10, 10)) %>% aoi_map()

## Add layers with standard leaflet functions:
r <- getAOI("UCSB") %>% # get AOI
  HydroData::findNWIS() # get SpatialPointsDataframe of local USGS gages

aoi_map(r$AOI) %>%
  addMarkers(data = r$nwis, popup = r$nwis$site_no)

## Save map for reference:
m <- getAOI("Kansas City") %>% aoi_map()
htmlwidgets::saveWidget(m, file = paste0(getwd(), "/myMap.html"))

## End(Not run)
```

`bbox_coords` *Return bounding box coordinates of a spatial (sp/sf/raster) object*

Description

This function provides a simple wrapper around sf::st_bbox that instead returns a named data.frame containing (xmin, ymin, xmax, ymax)

Usage

```
bbox_coords(x)
```

Arguments

`x`	a spatial object (sp/sf/raster)

Value

a data.frame

`bbox_get` *Get Spatial Bounding Box*

Description

Get spatial (sf) representation of bounding box of an input feature type. Input can be data.frame, numeric, character, or spatial (sp/sf/raster). If numeric or character order of inputs should be (xmin, xmax, ymin, ymax)

Usage

```
bbox_get(x)
```

Arguments

`x`	input feature

Value

a sf polygon

`check_pkg` *Check for a package*

Description

Check for a package

Usage

```
check_pkg(pkg)
```

Arguments

`pkg`	package name

`default_crs` *AOI Package*

Description

An area of interest (AOI) is a geographic extent. The aim of this package is to help users create these - turning locations, place names, and political boundaries into servicable representation for spatial analysis. The package defaults to EPSG:4326

See the README on github, and the project webpage for examples here.

Usage

```
default_crs
```

Format

An object of class `numeric` of length 1.

`discritize` *Materialize Grid from File or inputs*

Description

Materialize Grid from File or inputs

Usage

```
discritize(
  ext = NULL,
  dim = default_dim,
  in_crs = default_crs,
  out_crs = default_crs,
  spatrast = FALSE,
  fillValue = NULL,
  showWarnings = TRUE
)
```

Arguments

`ext`	extent (xmin, xmax, ymin, ymax) in some coordinate system
`dim`	dimension (number of columns, number of rows)
`in_crs`	projection of input ext
`out_crs`	projection of output object
`spatrast`	should a SpatRaster object be returned? Default is FALSE
`fillValue`	in spatrast is TRUE, what values should fill the object
`showWarnings`	should warnings be shown?

Value

list or SpatRaster object

`fip_meta` *Returns a sf data.frame of fipio data*

Description

Returns a sf data.frame of fipio data

Usage

```
fip_meta(state, county = NULL)
```

Arguments

`state`	State names, state abbreviations, or one of the following: "all", "conus", "territories"
`county`	County names or "all"

Value

sf data.frame

Examples

```
## Not run:
fip_meta()

## End(Not run)
```

`geocode` *Geocoding*

Description

A wrapper around the tidygeocoding and Wikipedia services. Users can request a data.frame (default), vector (xy = TRUE), point (pt = TRUE), and/or a bounding box (bbox = TRUE) representation of a place/location (geo) or event. One or more can be given at a time.

If a single entitiy is requested, ‘geocode‘ will provide a data.frame of lat/lon values and, if requested, a point object and the derived bounding box of the geo/event.

If multiple entities are requested, the returned objects will be a data.frame with columns for input name-lat-lon; if requested, a POINT object will be returned. Here, the bbox argument will return the minimum bounding box of all place names.

Usage

```
geocode(
  geo = NULL,
  event = NULL,
  pt = FALSE,
  bbox = FALSE,
  all = FALSE,
  xy = FALSE,
  method = default_method,
  crs = default_crs
)
```

Arguments

`geo`	`character`. Place name(s)
`event`	`character`. a term to search for on Wikipedia
`pt`	`logical`. If TRUE point geometry is created.
`bbox`	`logical`. If TRUE bounding box geometry is created
`all`	`logical`. If TRUE the point, bbox and xy representations are returned as a list
`xy`	`logical`. If TRUE a named xy numeric vector is created
`method`	the geocoding service to be used. See ?tidygeocoder::geocode
`crs`	desired CRS. Defaults to AOI::default_crs

Value

a data.frame, sf object, or vector

See Also

Other geocode: `geocode_rev()`, `geocode_wiki()`

Examples

```
## Not run:
## geocode a single locations
geocode("UCSB")

## geocode a single location and return a POINT object
geocode("UCSB", pt = TRUE)

## geocode a single location and derived bbox of location
geocode(location = "UCSB", bbox = TRUE)

## geocode multiple locations
geocode(c("UCSB", "Goleta", "Santa Barbara"))

## geocode multiple points and generate a minimum bounding box of all locations and spatial points
geocode(c("UCSB", "Goleta", "Santa Barbara"), bbox = T, pt = T)

## End(Not run)
```

`geocode_rev` *Reverse Geocoding*

Description

Describe a location using the ERSI and OSM reverse geocoding web-services. This service provides traditional reverse geocoding (lat/lon to placename) but can also be use to get more information about a place name. xy must contain geographic coordinates!

Usage

```
geocode_rev(xy, pt = FALSE, method = default_method)
```

Arguments

`xy`	`logical`. If TRUE a named xy numeric vector is created
`pt`	`logical`. If TRUE point geometry is created.
`method`	the geocoding service to be used. See ?tidygeocoder::geocode

Value

a data.frame, sf object, or vector

See Also

Other geocode: `geocode_wiki()`, `geocode()`

Examples

```
## Not run:
 geocode_rev(xy = c(38,-115))

## End(Not run)
```

`geocode_wiki` *Geocoding Events*

Description

A wrapper around the Wikipedia API to return geo-coordinates of requested inputs.

Usage

```
geocode_wiki(event = NULL, pt = FALSE)
```

Arguments

`event`	`character`. a term to search for on wikipeida
`pt`	`logical`. If TRUE point geometery is appended to the returned list()

Value

a data.frame of lat/lon coordinates

See Also

Other geocode: `geocode_rev()`, `geocode()`

Examples

```
## Not run:
## geocode an Agency
geocode_wiki("NOAA")

## geocode an event
geocode_wiki("I have a dream speech")

## geocode a n event
geocode_wiki("D day")

## geocode a product
geocode_wiki("New York Times")

## geocode an event
geocode_wiki("Hurricane Harvey")

## End(Not run)
```

`list_states` *Returns a data.frame of valid states with abbreviations and regions*

Description

Returns a data.frame of valid states with abbreviations and regions

Usage

```
list_states()
```

Value

data.frame of states with abbreviation and region

Examples

```
## Not run:
list_states()

## End(Not run)
```

`zipcodes` *USA Zipcode Centroids*

Description

A dataset containing the centriods of USA zipcodes

Usage

```
zipcodes
```

Format

An object of class `data.frame` with 33144 rows and 3 columns.

Source

`USABoundariesDataPackage`

Index