Package ‘forestChange’  
February 15, 2020

Type Package  
Title Computing Essential Biodiversity Variables from Global Forest Change Data  
Version 1.0  
Date 2020-02-14  
Author Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>  
Maintainer Wilson Lara <wilarhen@temple.edu>  
Description Metrics and statistics of Essential Biodiversity Variables are computed by processing Global Forest Change data (Hansen et al., 2013) <doi:10.1126/science.1244693>, Canopy Cover data (Sexton et al., 2013) <doi:10.1080/17538947.2013.786146>, and polygon geometries (e.g., GADM, see <https://gadm.org/>).  
License GPL-3  
Depends raster  
Imports rgdal, parallel, curl, gdalUtils, ggplot2, graphics, rvest, stats, landscapemetrics, sp, tibble, utils, xml2, dplyr  
Encoding latin1  
LazyData TRUE  
NeedsCompilation no  
Repository CRAN  
Date/Publication 2020-02-15 22:40:02 UTC  

R topics documented:  

forestChange-package .............................................................. 2  
barplot.EBVstats ................................................................. 3  
CCPolygon ........................................................................... 4  
EBVMask ............................................................................... 5  
EBVmetric ............................................................................. 6  
EBVstats ............................................................................... 7  
FCMask ............................................................................... 7
Description

Metrics and statistics of Essential Biodiversity Variables are computed by processing Global Forest Change data (Hansen et al., 2013) <doi:10.1126/science.1244693>, Canopy Cover data (Sexton et al., 2013) <doi:10.1080/17538947.2013.786146>, and polygon geometries (e.g., GADM, see <https://gadm.org/>).

Details

The DESCRIPTION file:

Package: forestChange
Type: Package
Title: Computing Essential Biodiversity Variables from Global Forest Change Data
Version: 1.0
Date: 2020-02-14
Author: Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>
Maintainer: Wilson Lara <wilarhen@temple.edu>
Description: Metrics and statistics of Essential Biodiversity Variables are computed by processing Global Forest Change data, Canopy Cover data, and polygon geometries (e.g., GADM).
License: GPL-3
Depends: raster
Imports: rgdal,parallel,curl,gdalUtils,ggplot2,graphics,rvest,stats,landscapemetrics,sp,tibble,utils,xml2,dplyr
Encoding: latin1
LazyData: TRUE
NeedsCompilation: no
Packaged: 2019-10-18 14:52:10 UTC; wilar

Index of help topics:

- CCPolygon Canopy-cover polygon
- EBVMask EBV Mask
- EBVmetric EBV metric
- EBVstats EBV Stats
- FCMask Forest-Change Mask
barplot.EBVstats

FCMosaic Forest-Change Mosaic
FCPolygon Forest-Cover Polygon
GFCurls URLs of GFC data
HansenUrltoExtent Extents in GFC links
barplot.EBVstats barplot EBV Stats
forestChange-package Computing Essential Biodiversity Variables from Global Forest Change Data
getGADM Get Geographic Administrative Unit
plot.EBVmetric EBV-metric plot

Maintainer: Wilson Lara <wilarhen@temple.edu>

Author(s)
Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

---

barplot.EBVstats    barplot EBV Stats

Description
A barplot of EBVstats is printed.

Usage
## S3 method for class 'EBVstats'
barplot(height, ...)  

Arguments
  height list of EBVstats.
  ... Additional arguments in barplot.

Value
Plot of EBVstats.

Author(s)
Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

Examples
height <- list(mean = abs(rnorm(4)), sd = abs(rnorm(4)))
class(height) <- 'EBVstats'
barplot(height, main = 'EBV Stats')
**CCPolygon**

**Canopy-cover polygon**

### Description

This function computes regions of interest across GFC areas for tree-cover data (GFC30TC).

### Usage

```r
CCPolygon(pol = NULL, path, int.patt = "[[:digit:].tif", mc.cores = detectCores(), ...)
```

### Arguments

- **pol** SpatialPolygonsDataFrame, character or NULL. Polygon geometry, GADM, stack such as the produced by FCPolygon, or NULL. If NULL then a list of GADM units is printed.
- **path** character. File path(s) to GFC30TC layers (.tif), or to zip archives containing GFC30TC.
- **int.patt** character. If files in path are compressed then common pattern in the extracted files. Default '[[:digit:].tif'.
- **mc.cores** logical. Use parallel execution. Default TRUE. Ignored in Windows machines.
- **...** Additional arguments in FCMask.

### Value

RasterStack, set of GADM units, or NULL.

### Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

### References


### Examples

```r
mun <- CCPolygon(NULL)
```
Description

This function combines EBV distribution maps with forest/non-forest masks from FCMask.

Usage

EBVMask(pol = NULL, path, int.patt = "[[:digit:].tif", mc.cores = detectCores(), ...

Arguments

- **pol**: SpatialPolygonsDataFrame, character or NULL. User-defined polygon, the name of a GADM, or such a name plus its corresponding higher-level unit. If NULL then a list of GADM units is printed.
- **path**: character. File path(s) to the distribution maps maps (.tif). Compressed files (.zip) are extracted.
- **int.patt**: character. If files in path are compressed then common pattern in the extracted files. Default '[[:digit:].tif'.
- **mc.cores**: numeric. The number of cores.
- **...**: additional arguments in FCMask. This argument only works if pol is not an object produced by FCMask.

Value

RasterStack, set of GADM units, or NULL.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

Examples

mun <- EBVMask(NULL)
Description

This function formats GFC and computes structural metrics of Essential Biodiversity Variables (EBV metrics): forest-cover areas and landscape metrics, in predefined polygons or GADM.

Usage

EBVmetric(pol, what = "lsm_l_tafc", mc.cores = detectCores(),
(...)

Arguments

pol 
RasterStack.Stack such as these produced by FCMask.
what 
character. The metrics. These include 'lsm_l_tafc' to compute total forest-cover areas (ha) and other landscape metrics in calculate_lsm. Default ('lsm_l_tafc').
mc.cores 
numeric. The number of cores, see mclapply.
...
additional arguments in calculate_lsm.

Details

Coordinate system of the spatial units should be UTM. Metrics other than 'lsm_l_tafc' are calculated implementing calculate_lsm.

Value

tibble.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

References


Examples

mpio <- 'Uribia'
msk <- FCMask(mpio, year = 10:17)
met <- EBVmetric(msk, what = 'lsm_l_frac_mn')
plot(met)
**EBVstats**

**EBV Stats**

**Description**

This function is a wrapper of cellStats to compute statistics for CCPolygon objects.

**Usage**

```r
EBVstats(ccp, stats, mc.cores = detectCores(), ...)
```

**Arguments**

- `ccp` RasterStack or NULL. Stack such as that produced by CCPolygon. If NULL then NULL is returned.
- `stats` character vector of stats defined in cellStats. If missing then 'mean', 'sd' and 'max' are computed.
- `mc.cores` numeric. The number of cores, see mclapply.
- `...` Additional arguments in cellStats

**Value**

list of EBVstats.

**Author(s)**

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

**Examples**

```r
EBVstats(NULL)
```

**FCMask**

**Forest-Change Mask**

**Description**

This function can format GFC into forest-distribution regions of interest over time.

**Usage**

```r
FCMask(pol = NULL, year = 1:2, cummask = TRUE, deforest = FALSE, perc = 80:100, mc.cores = detectCores(), ...)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pol</td>
<td>RasterStack; SpatialPolygonsDataFrame; Extent; character, or NULL. Spatial object such as any of these produced by getGADM or by FCPolygon. If NULL then a list of unit names is printed, see getGADM.</td>
</tr>
<tr>
<td>year</td>
<td>numeric. Years between 0 and 18 (or between 2000 and 2018).</td>
</tr>
<tr>
<td>cummask</td>
<td>logical. Compute cumulative masks instead of discrete masks. Default TRUE.</td>
</tr>
<tr>
<td>deforest</td>
<td>logical. Process non-forest areas instead of forest areas. Default FALSE.</td>
</tr>
<tr>
<td>perc</td>
<td>numeric. Percentage of canopy closure. Default 80:100. This argument is ignored if deforest = TRUE.</td>
</tr>
<tr>
<td>mc.cores</td>
<td>numeric. The number of cores, see mclapply.</td>
</tr>
<tr>
<td>...</td>
<td>Additional arguments in FCPolygon.</td>
</tr>
</tbody>
</table>

Value

RasterBrick of forest/non-forest masks over time. or character vector with GADM.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

Examples

```r/themeset,palette=rainbow
## A mask of GFC layers in the municipality of 'Uribia' is computed:
UribiaMask <- FCMask('Uribia')
plot(UribiaMask)
```

Description

This function tests whether adjacent layers of GFC can be bounded together using partial matching over the layer names. If it is possible then mosaic is implemented.

Usage

```r
FCMosaic(rst = NULL, lyrs = c("treecover2000", "lossyear"), mc.cores = detectCores())
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rst</td>
<td>list or NULL. List of GFC layers. If NULL then other arguments are ignored and the function returns NULL.</td>
</tr>
<tr>
<td>lyrs</td>
<td>character. Vector of strings matching layer names in GFC data. Defaults 'treecover2000' and 'lossyear'.</td>
</tr>
</tbody>
</table>
Details

The function is implemented by `FCPolygon` to cut GFC data.

Value

list of rasters.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

Examples

```r
## Printing NULL output:
FCMosaic(NULL)
```

---

### Description

This function can retrieve and crop layers of Global Forest Change (GFC) using polygon geometries (i.e., GADM).

### Usage

```r
FCPolygon(pol = NULL, lyrs = c("treecover2000", "lossyear"),
    path, url, pr.utm = TRUE, mc.cores = detectCores(), ...)
```

### Arguments

- **pol**: `SpatialPolygonsDataFrame`, or character. Polygon geometry, the name of a GADM, or such a name plus its corresponding higher-level unit. If NULL then a list of GADM units is printed, see `getGADM`.
- **lyrs**: character. Vector of strings matching layer names in the GFC. Defaults 'treecover2000' and 'lossyear'.
- **path**: character. Location of a directory with the GFC. This argument overrides the action of `url`.
- **url**: character. Web resource with text files containing lists of URLs for the GFC layers. If missing then data from the application programming interface of GFC is retrieved, see `GFCurls`.
- **pr.utm**: logical. Project to UTM crs.
- **mc.cores**: numeric. The number of cores, see `mclapply`.
- **...**: Additional arguments in `getGADM`. 
Details

The GADM are imported using the in-package `getGADM`. Links to the data sets are obtained using the in-package `GFCurls`. Geographic extents in both the GADM and the GFC are intersected implementing `HansenUrltoExtent`. Common areas between GFC and GADM are cropped using two functions of the `raster` package: `crop` and `rasterize`. Depending on localization of the GADM unit, several GFC layers by data type might be required. This is done implementing the in-package `FCMosaic`. This function could be memory demanding if the extents of the polygons used to cut the GFC are big (30,000 km²). For these cases, machines with RAM of 8 GB or greater should be used. In unix-alike systems, the package implements parallel execution, see `parallel` package.

Value

`RasterStack`, or set of GADM units.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

References


Examples

```r
## A list of departments of Colombia is printed:

dep <- FCPolygon(level = 1)
head(dep)

## Two adjacent layers of GFC must be bounded together before cropping
## the GFC data using the boundaries of the the municipality of
## "Cumaribo" in Colombia. This is automatically developed by
## FCPolygon:

cumariboArea <- FCPolygon(pol = "Cumaribo")

## The name 'Mosquera' matches two municipalities of Colombia. A
## corresponding department should be specified in the argument 'pol'
## of FCPolygon:

mosquera <- FCPolygon('Mosquera')
mosqueraNarinho <- FCPolygon(pol = c('Mosquera','Narino'))
```
getGADM

Get Geographic Administrative Unit

Description

This function is a wrapper of `getData` used to import levels in Geographic Administrative Units (GADM).

Usage

```r
getGADM(unit.nm = NULL, level = 2, country = "COL")
```

Arguments

- `unit.nm` character or NULL. Name of an administrative unit (e.g. municipality), or the name of such a name plus its corresponding higher-level unit (e.g. department/state). If NULL then a list of unit names is printed.
- `level` numeric. A number between zero and two, indicating any of the levels of administrative subdivisions in GADM: 0=country, 1=first level of subdivision, and 2=second level of subdivision).
- `country` character. ISO code specifying a country. Default 'COL'

Value

SpatialPolygonsDataFrame or character vector of GADM units.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

References

https://gadm.org/

Examples

```r
## Printing municipalities of Colombia:

muni <- getGADM()
head(muni)
```
### GFCurls

**URLs of GFC data**

**Description**

This function retrieves URL of Global Forest Change (GFC) data.

**Usage**

```r
gFCurls(lyrs = c("treecover2000", "lossyear"), url)
```

**Arguments**

- `lyrs` character. Name(s) of the layers. Default 'treecover2000', and 'lossyear'
- `url` character. Path to the html file containing the files. Default NULL retrieves URLs stored in the application programming interface of GFC, see References.

**Value**

character vector.

**Author(s)**

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

**References**


**Examples**

```r
gainLayers <- gFCurls(lyrs = 'gain')
head(gainLayers)
```

---

### HansenUrltoExtent

**Extents in GFC links**

**Description**

This function can extract extents of Global Forest Change data (GFC) using a corresponding URL.

**Usage**

```r
HansenUrltoExtent(x, path. = "[[[:digit:]]{1,3}[N|S|E|W]]")
```
plot.EBVmetric

Arguments

x character. URL to the GFC such as any of these produced by `GFCurls`.

path character. Pattern in the URL to extract the extent. Default extracts the 3 digits nearest to any of the cardinal directions: N, S, E, or W.

Details

The function is implemented by `FCPolygon`.

Value

extent.

Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

References

http://earthenginepartners.appspot.com/science-2013-global-forest

Examples

```r
HansenUrltoExtent(urtt)
```

Description

A plot of `EBVmetric` is printed.

Usage

```r
## S3 method for class 'EBVmetric'
plot(x, ...)
```

Arguments

x tibble. Data set of metrics such as that produced by `EBVmetric`.

... Further arguments not implemented here.

Value

plot.
Author(s)

Wilson Lara <wilarhen@temple.edu>, Victor Gutierrez-Velez <victorhugo@temple.edu>

Examples

```r
mpio <- 'Uribia'
msk <- FCMask(mpio, year = 10:17)
met <- EBVmetric(msk, what = 'lsm_l_frac_mn')
plot(met)
```
Index

*Topic package
  forestChange-package, 2

barplot, 3
barplot.EBVstats, 3

calculate_lsm, 6
CCPolygon, 4, 7
cellStats, 7
crop, 10

EBVMask, 5
EBVmetric, 6, 13
EBVstats, 3, 7

FCMask, 4–6, 7
FCMosaic, 8, 10
FCPolygon, 8, 9, 9, 13
forestChange (forestChange-package), 2
forestChange-package, 2

dataGet, 11
dataGetGADM, 9, 10, 11
dataGFCurls, 9, 10, 12, 13

HansenUrlToExtent, 10, 12

mclapply, 6–9
mosaic, 8

parallel, 10
plot.EBVmetric, 13

raster, 10
rasterize, 10

tibble, 13