Package 'r2dRue'

July 2, 2014

Type Package

Title 2d Rain Use Efficience model

Version 1.0.4

Date 2013-04-21

Author Gabriel del Barrio, Juan Puigdefabregas, Maria E. Sanjuan & Alberto Ruiz

Maintainer Alberto Ruiz Moreno <aruiz@eeza.csic.es>

Description 2dRUE is a methodology to make a diagnostic of land condition in a large territory during a given time period. The following projects have funded this package: DeSurvey IP (EC FP6 Integrated Project contract No. 003950), DesertWatch (ESA DUE contract No. 18487/04/I-LG) and MesoTopos (Junta de Andalucia PE ref. RNM-4023).

License GPL

LazyLoad yes

Depends rgdal,sp,matrixStats

NeedsCompilation no

Repository CRAN

Date/Publication 2013-06-28 12:22:19

R topics documented:

`r2dRue-package` *2dRUE model functions*

Description

2dRUE is a methodology to make a diagnostic of land condition in a large territory during a given time period. 2dRUE was developed as a low cost and flexible methodology taking full advantage of readily available data from Global Environmental Monitoring satellites and geodatabases.

Assessment and monitoring are explicitly separated to yield estimates of status and trends of land condition respectively. In the assessment module, Rain Use Efficiency is implemented on two time-scales to yield estimates of biomass-maturity and productivity-resilience respectively. In the monitoring module, effects of time and aridity on vegetation growth are discriminated to detect intrinsic trends of aggradation or degradation in the vegetation cover.

2dRUE is based on archived time-series and requires two types of input data. The first one is a vegetation density index such as NDVI, for which several products are available (e.g. GIMMS, MEDOKADS, etc.). The second one are climate fields at a corresponding spatial resolution, involving mean maximum, mean and mean minimum temperatures and precipitation

The r2dRue software library contains all the relevant functions to compute the quantitative maps of assessment (mean and extreme observed RUE) and of monitoring (effects of time and aridity). It also has additional utilities to compute aridity indexes of both 2dRUE and FAO-UNEP formulae, and to make queries to the climate and vegetation archives.

The recommended sequence of functions for a concrete analysis is: rgf.create (optional), petHgsm (optional), edit2dRfile, read2dRfile, showInfo, assessment, monitoring and summarize. Alternatively, individual functions such as rgf.summary can be used to explore the data set and/or conduct a step-by-step analysis.

The following projects have funded this package: DeSurvey IP (EC FP6 Integrated Project contract No. 003950), DesertWatch (ESA DUE contract No. 18487/04/I-LG) and MesoTopos (Junta de Andalucia PE ref. RNM-4023).

Details

Package:	r2dRue
Type:	Package
Version:	1.0.4
Date:	2013-06-28

License: GPL
LazyLoad: yes

Author(s)

Gabriel del Barrio, Juan Puigdefabregas, Maria E. Sanjuan & Alberto Ruiz
Maintainer: Alberto Ruiz, <aruiz@eeza.csic.es>

References

del Barrio, G., Puigdefabregas, J. Sanjuan, M.E., Stellmes, M. & Ruiz, A. (2010). Assessment and monitoring of land condition in the Iberian Peninsula, 1989-2000. Remote Sensing of Environment, 114: 1817-1832 http://dx.doi.org/10.1016/j.rse.2010.03.009

`assessment` *Performs the assessment component of a r2dRue analysis as specified by its associated parameters*

Description

Computes Mean Observed RUE, Mean Observed Aridity, Extreme Observed RUE and Extreme Observed Aridity, and saves the results to corresponding raster files. A batch operation is parameterised from the r2dRue current memory settings (see read2dRfile and showInfo).

Rudimentary bars show the computation in progress.

Usage

```
assessment(o)
```

Arguments

`o`	A r2dRue Object

Details

The following r2dRue functions are involved in this computation. Local parameters are left with their respective default values, except those specified in the parameters file as displayed by showInfo:

Value

The following raster files are created resulting from the assessment operation. They belong to the SpatialGridDataFrame class

`rueObsMe`	Mean Observed rain Use Efficiency
`rueObsEx`	Extreme Observed Rain Use Efficiency
`aiObsMe`	Mean Observed Aridity Index
`aiObsEx`	Extreme Observed Aridity Index

References

del Barrio, G., Puigdefabregas, J. Sanjuan, M.E., Stellmes, M. & Ruiz, A. (2010). Assessment and monitoring of land condition in the Iberian Peninsula, 1989-2000. Remote Sensing of Environment, 114: 1817-1832 http://dx.doi.org/10.1016/j.rse.2010.03.009

`editr2dRfile` *Enter the parameters of a concrete r2dRue analysis and store them in a file.*

Description

This function facilitates the at once specification of all the parameters corresponding to a given r2dRue analysis. The parameters are saved to a text file that may be subsequently read by read2dRfile() to perform the analysis in batch mode.

Usage

```
editr2dRfile(conf = "")
```

Arguments

`conf`	Name of the parameters file to be created or edited.

Details

1. The user is prompted with questions to enter the following parameters. Default values are suggested if editing an existing parameters file.

   comment Description of this run. 255 characters maximum
   pOut Output directory. Specify the full path
   viRgf Name of the Vegetation Index raster group
   rainRgf Name of the precipitation raster group
   petRgf Name of the Potential Evapo-Transpiration raster group
   sYear Start moment (year) of these raster groups. Format: yyyy
   sMonth Start moment (month) of these raster groups. Format: mm
   mHidro Start month of hydrological year

acum Number of cumulative months for preceding rain

yIni Start year of this run

yEnd End year of this run

driver GIS format for raster images. A GDAL driver name

flag Missing value flag for raster images

2. The output file can be also edited with a text editor.
3. It is recommended to set a new output directory if any parameter is changed for a new analysis of the same dataset. Otherwise, existing results will be overwritten.
4. Raster groups contain sequences of file names associated with their respective time-series. It is assumed here that time-series: start at the same moment and match in length ; are at a monthly resolution; and have no gaps in between.
5. A summary of the available time-series and number of hydrological years is offered just before the user is prompted to enter the parameters of a particular run.

See Also

batchPetHgsm, rgf.create, readr2dRfile, showInfo

`monitoring` *Performs the monitoring component of a r2dRue analysis as specified by its associated parameters*

Description

Compute a stepwise regression of vegetation index over aridity index and year order-number. As many points as available hydrological years are made from annual means of these variables, which are first computed from their respective time-series. The results are then saved to corresponding raster files. A batch operation is parameterised from the r2dRue current memory settings (see read2dRfile and showInfo). Rudimentary bars show the computation in progress.

Usage

```
monitoring(o)
```

Arguments

o	A r2dRue Object

Details

The following r2dRue functions are involved in this computation. Local parameters are left with their respective default values, except those specified in the parameters file as displayed by showInfo:

Value

The following raster files are created resulting from the monitoring operation. They belong to the SpatialGridDataFrame class

`index`	individual label for each grid cell
`effect_time`	rate of change in std.dev. of vegetation index per one std.dev. of time; zero if non significant
`effect_arid`	rate of change in std.dev. of vegetation index per one std.dev. of aridity; zero if non significant
`veg_response`	significant response of vegetation index to time (1), aridity (2), both (3) or none (4)
`ta_single`	single correlation coefficient of time and aridity; zero if non significant
`tv_single`	single correlation coefficient of time and vegetation index; zero if non significant
`av_single`	single correlation coefficient of aridity and vegetation index; zero if non significant
`viMe yyyy`	mean annual vegetation index corresponding to year yyyy
`aiMeyyyy`	mean annual aridity index corresponding to year yyyy
`timeyyyy`	dummy raster containing year yyyy in each grid cell

Author(s)

Gabriel del Barrio, Marieta SanJuan & Alberto Ruiz

References

del Barrio, G., Puigdefabregas, J. Sanjuan, M.E., Stellmes, M. & Ruiz, A. (2010). Assessment and monitoring of land condition in the Iberian Peninsula, 1989-2000. Remote Sensing of Environment, 114: 1817-1832 http://dx.doi.org/10.1016/j.rse.2010.03.009

Sokal, R. R., & Rohlf, F. J. (1995). Biometry. New York: W.H. Freeman and Co.

See Also

`regStepRaster`, `editr2dRfile`, `readr2dRfile`, `showInfo`

`petHgsm` *Computes Potential Evapo-Transpiration (PET)*

Description

PET (mm) is computed from mean maximum, mean and mean minimum air temperatures in Celsius degree and extra-terrestrial solar radiation (cal/cm2/day) using the Hargreaves-Samani equation.

batchPet () provides a batch process to compute PET over sets of images, typically time-series, specified as raster groups.

Usage

```
petHgsm(Tmin,Tmax,Tmed,Rad,month)
batchPetHgsm(outFl, monthIni, Tmin, Tmed, Tmax, Rad, ...)
```

Arguments

monthIni	The month order-number of the first image in the time-series. An integer between 1 and 12.
month	The target month order-number to compute PET An integer between 1 and 12.
Tmin	In petHgsm, a SpatialGridDataFrame with mean minimum air temperature in Celsius degrees. In batchpetHgsm, a raster group with the file names containing mean minimum air temperature.
Tmax	In petHgsm a SpatialGridDataFrame with mean maximum air temperature in Celsius degrees. In batchpetHgsm, a raster group with the file names containing mean maximum air temperature.
Tmed	In petHgsm a SpatialGridDataFrame with mean air temperature (Celsius degree). In batchpetHgsm, a raster group with the file names containing mean air temperature.
Rad	In petHgsm a SpatialGridDataFrame with extra-terrestial solar radiation (cal/cm2/day). In batchpetHgsm a raster group with the file names containing extra-terrestial solar radiation.
outFl	A raster group with file names to save the results
...	Any unmatched parameter will be passed to the writeGDAL routine.

Value

petHgsm returns a spatialGridDataFrame representing the potential evapotranspiration. batchEtp writes to disc the computed rasters as SpatialGridDataFrame objects.

Author(s)

Gabriel del Barrio, Marieta E. SanJuan & Alberto Ruiz

References

Estimating potential evapotranspiration. Hargreaves, G.H., Samani, Z.A. Journal of the Irrigation & Drainage Division - ASCE Volume 108, Issue IR3, 1982, Pages 225-230

Examples

```
#not run
## Read  the temperature and radiation data
#tmin=readGDAL('Tmin_Ene1990')
#tmax=readGDAL('Tmax_Ene1990')
#tmea=readGDAL('Tmea_Ene1990')
#rad=readGDAL('SolarRad_Ene1990')
## Calc the PET for Ene 1990
```

```
#PetEne=petHgsm(tmin,tmax,tmea,rad, 1)
#writeGDAL(PetEne,'PetEne1990')
## Or to calculate and write to disk, in a batch process, a set of PETs
## create the Input file list
#Flmin=c('Tmin_Ene1990','Tmin_Feb1990', 'Tmin_Mar1990')
#Flmax=c('Tmax_Ene1990','Tmax_Feb1990', 'Tmax_Mar1990')
#Flmea=c('Tmea_Ene1990','Tmea_Feb1990', 'Tmea_Mar1990')
#Flrad=c('Rad_Ene','Rad_Feb', 'Rad_Mar')
#PETFileNames=c('PET_Ene1990','PET_Feb1990', 'PET_Mar1990')
##define the date of the first image
#iniDate=c(1,1990)
#batchPET(iniDate, Flmin, Flmax, Flmea, Flrad, PETFileNames)
```

`r2dRplot` *functions to plot several results of 2dRue analisys*

Description

incomplete help

Usage

```
r2dRplot(o, type = "rain", scope = "run", var = "vi", pixel = 1,
col = c("blue", "green4", "salmon", "gray"))
```

Arguments

`o`	a r2dRue object
`type`	valids type are: "rain", "ndvi", "vimax", "box", "density", "assessment1", assessment2", "monitoring", or "pixel"
`scope`	unused in this implementation
`var`	"vi" or "rain"
`pixel`	unused in this implementation
`col`	unused in this implementation

`rasterStack` *Stacks bands from raster image files*

Description

Concatenates multiple raster files into a single file. Provides support to several interleaved schemas: BIP (Bands Interleaved by Pixel), BIL (Band Interleaved by Line) or BSQ (Band Sequential). The output concatenated file may be saved in Ascii or binary format.

Usage

```
rasterStack(inFl, outFN, asc = FALSE, zip = FALSE, dec = 3,
interleave = "BIP", silent = FALSE)
```

Arguments

`inFl`	List with the name of the files to be concatenated
`outFN`	Name for the output file
`asc`	Logical. If TRUE the output is an ASCII file
`zip`	Logical. If TRUE the output is zipped. Only used if asc is TRUE.
`dec`	Integer. If present, the output values are rounded to the specified number of decimal places. Only used if asc is TRUE.
`interleave`	The interleave format. BIP, BIL or BSQ.
`silent`	Logical. If TRUE, comment and non-fatal CPL driver errors are suppressed.

Details

In binary mode (default), the data is stored as 32bit floating point number. Only the first band of each image is used. This function is not intended to be of general utility. It is used in internal processes to arrange image data into a format that is optimum for the process.

Value

An ASCII or binary file.

Author(s)

Alberto Ruiz. aruiz@eeza.csic.es

`readr2dRfile` *Reads metadata for a concrete r2dRue analysis as specified in its corresponding parameters file*

Description

The analysis parameters (names of the raster groups, number of images per time-series, available and analysis periods, number of hydrological years, etc.) are read from the parameters file. Then, the file names within each raster group are read to memory. All that information is also reported to the R console along with basic spatial data such as raster dimensions and resolution, and the PROJ.4 coordinate reference system specification.

Usage

```
readr2dRfile(conf)
```

Arguments

`conf`	Name of an existing parameters file

Details

This function is intended to be executed at the beginning of an analysis. For that reason, the sections on assessment and monitoring always report "results not updated". If read2dRfile is executed again within the same session, and a previous analysis has yielded any of these sections updated (which can be checked using showInfo), such information will be erased.

Value

Data are read to memory. Nothing is returned.

Author(s)

Alberto Ruiz, aruiz@eeza.csic.es

See Also

`showInfo`, `editr2dRfile`, `assessment`, `monitoring`

`regStepRaster` *Stepwise regression over raster images*

Description

This function performs a stepwise regression with two predictors.

The regression is made in standard form, therefore the reported regression coefficients are in standard deviation units. Multiple regression is accepted only if the second variable produces a significant increment of determination. Else, single correlations are explored and correlation coefficients are reported if significant. Significance is set to p<=0.1.

In r2dRue, the predictors are typically time (year order-number) and mean annual Aridity Index, whilst mean annual Vegetation Index is the dependent variable. The regression is fitted using as many points as available hydrological years. However, other variables could be used whilst the format and rationale of the procedure is respected.

Usage

```
regStepRaster(ndviFl, timeFl, aridFl, outFl, silent = FALSE, ...)
```

Arguments

`ndviFl`	Raster group with the file names of the annual Vegetation Index rasters.
`timeFl`	Raster group with the file names of the annual year number-order rasters.
`aridFl`	Raster group with the file names of the annual Aridity Index rasters.
`outFl`	A list of 7 file names to save the results.
`silent`	Logical. If TRUE, comment and non-fatal CPL driver errors are suppressed.
`...`	Any unmatched parameter will be passed to the writeGDAL routine. Usually drivername=..., or mvFlag=...

Details

The rasters specified in argument timeFL are dummy images with the year number-order, or simply the year, in each grid cell. This is used as one of the predictors.

The rationale is explained in del Barrio et al. below.

The regression procedure has been taken from Box 9.1 of Sokal below.

Value

Returns no value, but saves to disk the following raster files. They belong to the SpatialGridDataFrame class

`index`	individual label for each grid cell
`effect_time`	rate of change in std.dev. of vegetation index per one std.dev. of time; zero if non significant

effect_arid	rate of change in std.dev. of vegetation index per one std.dev. of aridity; zero if non significant
veg_response	significant response of vegetation index to time (1), aridity (2), both (3) or none (4)
ta_single	single correlation coefficient of time and aridity; zero if non significant
tv_single	single correlation coefficient of time and vegetation index; zero if non significant
av_single	single correlation coefficient of aridity and vegetation index; zero if non significant
viMeyyyy	mean annual vegetation index corresponding to year yyyy
aiMeyyyy	mean annual aridity index corresponding to year yyyy
timeyyyy	dummy raster containing year yyyy in each grid cell

Author(s)

Gabriel del Barrio, Marieta SanJuan & Alberto Ruiz

References

del Barrio, G., Puigdefabregas, J. Sanjuan, M.E., Stellmes, M. & Ruiz, A. (2010). Assessment and monitoring of land condition in the Iberian Peninsula, 1989-2000. Remote Sensing of Environment, 114: 1817-1832 http://dx.doi.org/10.1016/j.rse.2010.03.009

Sokal, R. R., & Rohlf, F. J. (1995). Biometry. New York: W.H. Freeman and Co.

See Also

rgf.create

`rgf.create` *Functions to read and write raster group files.*

Description

Raster group files are essential elements for managing information on time-series in r2dRue. They are simply text lists containing the names of the files of a given time-series. Because names in such lists follow many times regular and sequential patterns, these functions may facilitate their creation.

Usage

```
rgf.read(inFl)
rgf.create(prefix,suffix='',ini,fin=ini,monthini=1,output)
```

Arguments

inFl	Filename of the raster group file to be read.
prefix	Prefix for the output filenames.
suffix	Suffix for the output filenames.
ini	Start year.
fin	End year.
monthini	Number-order of the start month (integer in the 1 to 12 range).
output	Filename to be written to disk.

Details

rgf.read reads a raster group file and returns a character vector with the read values.

rgf.create is useful to generate sequences of file names that contain an indicator of the month and year. For example, take the sequence:

Ndvi199001.tif, Ndvi199002.tif ... Ndvi199012.tif, Ndvi199101.tif, Ndvi199102.tif ...

In it, a prefix ("Ndvi") is concatenated to a sequence of years (1990,1991) and months (01,02,...,12), and it ends with a suffix (".tif").

Value

Rgf.read returns a character vector with the filenames in the rgf file.

Rgf.create returns a character vector. If an output file name parameter is specified, then the list is writen to disc.

Author(s)

Alberto Ruiz, aruiz@eeza.csic.es

Examples

```
rgf.create('ndvi','.rst',1970,1975,output='ndvi.rgf')
ndvifiles=rgf.read('ndvi.rgf')
```

---

rgf.plot *functions to plot raster groups*

Description

functions to plot raster groups

Usage

```
rgf.plot(o, type = "rain")
```

Arguments

o	a r2dRue object
type	valid type are "box", "density", "rain"

`rgf.summary` *Extract summaries from time-series specified as raster group files.*

Description

These functions query the time-series specified in a raster group file and save the selected summary in a new raster. Summaries refer to the population of values of each grid cell across the time-series.

rgf.summary computes summaries, and rgf.when finds at which moment (ie. order No. in the time-series) the selected summary is located.

Usage

```
rgf.summary(inFl, outFl, step= length(inFl), fun = "SUM", silent = FALSE, ...)
rgf.when(inFl, ref, order = "FIRST", silent = FALSE)
```

Arguments

inFl	Raster group file with the file names of the input time-series.
outFl	Raster group file with the file names of the output time-series.
step	Number of images taken for each summary. If a value is not supplied, the summary is made on the complete time-series.
fun	Summary function SUM, MAX, MIN, MEAN, COUNT, RANGE, SD, VAR,MEDIAN.
ref	The reference image.
order	"FIRST" or "LAST"; to find the first or the last occurrence in the time-series.
silent	logical; if TRUE, comment and non-fatal CPL driver errors are suppressed.
...	Any unmatched parameter will be passed to the writeGDAL function.

Details

Any summary can be computed for the entire set, or for a subset of images. For example, let a time-series made of 36 images representing monthly NDVI. The maximum NDVI found at each grid cell could be computed for the entire series (step not specified), for each year of the series (step = 12), or quarterly (step = 3). The corresponding summary images are then saved to raster files.

rgf.when compares a reference image with the images in the time-series (or a subset of it), and returns a composite image where each grid cell contains the order-number at which its reference value is found.

Value

rgf.summary does not returns any value, but writes to disk the solicited summaries.

rgf.when returns a composite raster image as a SpatialGridDataFrame.

Author(s)

Alberto Ruiz, aruiz@eeza.csic.es

See Also

rgf.create

Examples

```
#not work...
#rgf.create('ndvi','.rst',1970,1975,output='ndvi.rgf')
#ndvifiles=rgf.read('ndvi.rgf')
#rgf.summary(ndvifiles,'maxNdvi.rst',fun='MAX', drivername='RST')
#Anualnames=paste('maxNdvi',1970:1975,sep='_')
#maxNdvi=rgf.summary(ndvifiles,AnualNames, step=12,fun='MAX',drivername='RST')
```

`rueObsEx` *Functions to perform the assessment component of a r2dRue analysis through direct specification of parameters*

Description

These functions compute Mean Observed RUE, Mean Observed Aridity, Extreme Observed RUE and Extreme Observed Aridity, and save the results to corresponding raster files.

Their arguments can be manually specified, or they are piped from the assessment function.

Usage

```
rueObsMe(rainFl, viFl, silent = FALSE)
aiObsMe(rainFl, petFl, FAO = FALSE, silent = FALSE)
rueObsEx(rainFl, viFl, preRainFl, nMonths = 6, silent = FALSE)
aiObsEx(rainFl, viFl, petFl, preRainFl, prePetFl, FAO=FALSE, nMonths = 6, silent = FALSE)
```

Arguments

`rainFl`	Raster group file with the file names of the precipitation grids
`viFl`	Raster group file with the file names of the vegetation index grids
`petFl`	Raster group file with the file names of the evapotranspiration grids
`preRainFl`	Raster group file with the file names of the precipitation files preceding the given period
`prePetFl`	Raster group file with the file names of the evapotranspiration files preceding the given period
`FAO`	logical; If TRUE the index is according with the FAO definition

nMonths	A integer value representing the number of cumulative months for preceding precipitation. If a valid image filename is supplied, the value is read from the file.
silent	Logical. If TRUE, comment and non-fatal CPL driver errors are suppressed

Details

rueObsMe compute mean observed RUE. aiObsMe compute mean observed aridity index. rueObsEx compute extreme observed RUE. aiObsEx compute extreme observed aridity index.

Value

The following raster files are created resulting from the assessment operation. They belong to the SpatialGridDataFrame class

rueObsMe	Mean Observed rain Use Efficiency
rueObsEx	Extreme Observed Rain Use Efficiency
aiObsMe	Mean Observed Aridity Index
aiObsEx	Extreme Observed Aridity Index

Author(s)

Gabriel del Barrio, Juan Puigdefabregas, Maria E. Sanjuan & Alberto Ruiz

References

del Barrio, G., Puigdefabregas, J. Sanjuan, M.E., Stellmes, M. & Ruiz, A. (2010). Assessment and monitoring of land condition in the Iberian Peninsula, 1989-2000. Remote Sensing of Environment, 114: 1817-1832 http://dx.doi.org/10.1016/j.rse.2010.03.009

Examples

```
#not work...
#frain=rgf.read('rain.rgf')
#fpet=rgf.read('pet.rgf')
#fvi=rgf.read('ndvi.rgf')
#rome=rueObsMe(frain,fvi)
#summary(rome)
#image(rome)
#writeGDAL(rome,'rome.rst',drivername='RST',mvFlag=-99)
#aiome=rueObsMe(frain,fvi,fpet)
#writeGDAL(aiome,'aiome.rst',drivername='RST',mvFlag=-99)

#fprain=read.rgf('prerain.rgf')
#fppet=read.rgf('prepet.rgf')
#roex=rueObsEx(frain,fvi,fprain)
#aioex=rueObsEx(frain,fvi,fprain,fppet)
#writeGDAL(roex,'roex.rst',drivername='RST',mvFlag=-99)
#writeGDAL(aioex,'aioex.rst',drivername='RST',mvFlag=-99)
```

showInfo *Shows the metadata and status of a concrete r2dRue analysis as specified by its associated parameters*

Description

Reports to the R console the details stored in memory (not in a file) of the current r2dRue analysis. Five types of information are displayed: analysis parameters (names of the raster groups, number of images per time-series, available and analysis periods, number of hydrological years, etc.), raster group information (location and names of all the files making the different time-series), spatial information (raster dimensions and resolution, and the PROJ.4 coordinate reference system), assessment results if any (basic statistics of the main four output rasters), and monitoring results if any (basic statistics of the main seven output rasters).

Usage

```
showInfo(o)
```

Arguments

o	A r2dRue Object

Details

In contrast with read2dRfile, showInfo can be executed at any moment within an analysis, and its details will be updated as assessment and monitoring are executed.

The information displayed by showInfo (not the output data, only the status report) will be lost if read2dRfile is executed again.

The status report will be lost anyway after closing the current session. If that information must be preserved, for example for documentation purposes, save it to a text file. If the analysis itself is to be preserved, write the r2dRue object (see Arguments section) to a file.

The information reported by showInfo and read2dRfile is very similar. The difference is that the former mirrors the memory settings, whilst the latter reads data from a file .

See Also

readr2dRfile, editr2dRfile, assessment, monitoring

`solarRad` *Computes extra-terrestrial solar radiation*

Description

solarRad computes the solar radiation received at the top of the Earth's atmosphere on a horizontal surface. This radiation will change during the day and will be different at different latitudes and in different seasons.

Usage

```
solarRad(img, day)
solarRad12M(img, outFl, ...)
```

Arguments

`img`	object of class SpatialGridDataFrame that provided a projection class and a bbox attributes. The nature of the data.frame that contains the attribute data is not relevant. It must be in geographic coordinates.
`day`	The number of the day (from 1 to 365) to compute extra-terrestrial radiation.
`outFl`	A list of 12 file names to save the results.
`...`	Any unmatched parameter will be passed to the writeGDAL routine.

Details

The output is expressed in cal/cm2*day.

This function runs unnoticed within petHgsm, but can be executed manually.

Value

solarRad returns a single image corresponding to the specified day, as a SpatialGridDataFrame class. solarRad12M writes 12 images to disk in the format especified by the GDAL drivername parameter.

Author(s)

Gabriel del Barrio, Marieta SanJuan & Alberto Ruiz

References

Crop evapotranspiration - Guidelines for computing crop water requirements - FAO Irrigation and drainage paper 56

See Also

`petHgsm`

Examples

```
#not run
#aux=readGDAL('iberia_mask')
#calc the extraterrestial radiation the 15-agost
#Rad=solarRad(aux,227)
#writeGDAL(Rad,'solarrad_15ago')
#or to calculate and write to disk, the radiation of the twelve months
#RadFileNames=paste('rad',month.abb,sep='')
#solarRad12M(img,RadFileNames)
```

Index