Package ‘weaana’

September 27, 2021

Maintainer Bangyou Zheng <bangyou.zheng@csiro.au>
Title Analysis the Weather Data
Type Package
Description Functions are collected to analyse weather data for agriculture purposes including to read weather records in multiple formats, calculate extreme climate index.
License MIT + file LICENSE
URL https://weaana.bangyou.me/, https://github.com/bzychgen/weaana
BugReports https://github.com/bzychgen/weaana/issues
Encoding UTF-8
Version 0.1.1
Date 2021-09-06
LazyData true
Depends R (>= 3.5.0)
Imports methods, stats, settings, reshape2, lubridate, magrittr, rlang, dplyr, tibble
RoxygenNote 7.1.2
Suggests testthat
NeedsCompilation no
Author Bangyou Zheng [aut, cre]
Repository CRAN
Date/Publication 2021-09-27 10:00:02 UTC

R topics documented:

  changeWeatherRecords ........................................... 2
  convert2Records .................................................. 3
  createWeaAna ..................................................... 3
  dayLength ...................................................... 4
changeWeatherRecords

Description

Change weather records

Usage

changeWeatherRecords(object, ...)

## S4 method for signature 'WeaAna'
changeWeatherRecords(object, ...)

Arguments

object A WeaAna object.
...
New weather records

Value

A new WeaAna object with updated records
convert2Records

Description
Convert a data frame to weaana class

Usage
convert2Records(infor, records)

Arguments
- infor: A list or data frame of site information
- records: A data frame will convert to records

Value
A new WeaAna object

createWeaAna

Description
create WeaAna class

Usage
createWeaAna(mets)

Arguments
- mets: A list contained information of weather records.

Value
A new WeaAna class
**dayLength**

The time elapsed in hours between the specified sun angle from 90 degree in am and pm. +ve above the horizon, -ve below the horizon.

**Description**

The time elapsed in hours between the specified sun angle from 90 degree in am and pm. +ve above the horizon, -ve below the horizon.

**Usage**

dayLength(doy, lat, angle = -6)

**Arguments**

- **doy**: day of year number
- **lat**: latitude of site (deg)
- **angle**: angle to measure time between, such as twilight (deg). angular distance between 90 deg and end of twilight - altitude of sun. +ve up, -ve down.

**Value**

day length in hours

---

**diurnalT**

Calculate the diurnal variation in air temperature with Parton and Logan, 1981

**Description**


**Usage**

diurnalT(maxt, mint, doy, hour, latitude, A = 1.5, B = 4, C = 1)
**getWeatherRecords**

**Arguments**

- `maxt` maximum daily temperature
- `mint` minimum daily temperature
- `doy` day of year
- `hour` hour from 1 to 24
- `latitude` latitude in radials
- `A` is the time lag in temperature after noon
- `B` is coef that controls temperature decrease at night
- `C` is the time lag for min temperature after sunrise

**Value**

A vector with diurnal air temperature

**Examples**

```r
diurnalT(maxt = 20, mint = 10, doy = 1,
      hour = seq(from = 1, to = 23.99, by = 0.1),
      latitude = -10, A = 1.5, B = 4, C = 1)
```

---

**getWeatherRecords**  
*Get all weather records by year range*

**Description**

Get all weather records by year range

**Usage**

```r
getWeatherRecords(object, ...)
```

### S4 method for signature 'WeaAna'

```r
getWeatherRecords(object, yrange = NULL, vars = "all", ...)
```

**Arguments**

- `object` A WeaAna object.
- `...` Other arguments
- `yrange` Year range.
- `vars` Variable

**Value**

A data frame with all weather records
Examples

```r
library(weaana)
data("WeatherRecordsDemo")
getWeatherRecords( records, yrange = c(2008, 2009), length = 10 )
```

interpolationFunction  \textit{Return a y value from a linear interpolation function}

Description

Return a y value from a linear interpolation function

Usage

```r
interpolationFunction(x, y, values, split = "\\s+")
```

Arguments

- `x`: x
- `y`: y
- `values`: values
- `split`: split

Value

The interpolated values

readWeatherRecords  \textit{Read weather records from a file list and/or a folder list}

Description

Read weather records from a file list and/or a folder list

Usage

```r
readWeatherRecords(
  dataFiles = NULL,
  dataFolders = NULL,
  dataFormat = "APSIM",
  dataWeather = NULL,
  load.later = FALSE,
  ...
)
```
**records**

**Arguments**
- `dataFiles`: A character vector to specify the path of weather data files.
- `dataFolders`: A character vector to specify the path of weather data folders.
- `dataFormat`: The format of weather data file.
- `dataWeather`: A data.frame for existing data.
- `load.later`: Whether load weather records now or later. "dataFormat" should be One of "APSIM" and "RDATA".
- `...`: Other arguments

**Value**
A WeaAna class which contains all weather data.

---

<table>
<thead>
<tr>
<th>records</th>
<th>Demo weather records</th>
</tr>
</thead>
</table>

---

**Description**
Demo weather records

**Usage**
records

**Format**
An object of class WeaAna of length 1.

---

<table>
<thead>
<tr>
<th>result-class</th>
<th>Define the class for statistics results</th>
</tr>
</thead>
</table>

---

**Description**
Define the class for statistics results

**Slots**
- `name`: Name of result
- `type`: Type of result
show, WeaAna-method

Show basic information of class WeaAna

Description

Show the name, number, latitude, longitude of all weather stations.

Usage

```r
## S4 method for signature 'WeaAna'
show(object)
```

Arguments

- `object` WeaAna objects

Examples

```r
library(weaana)
data("WeatherRecordsDemo")
show(records)
records
```

siteInfor

Get site information

Description

Get site information

Usage

```r
siteInfor(object, ...)
```

---

## S4 method for signature 'WeaAna'
siteInfor(object, load.now = FALSE)

---

## S4 method for signature 'WeaAnaSite'
siteInfor(object, load.now = FALSE)
sphericalDistance

Arguments

- object: A WeaAnaSite object.
- ...: Not used
- load.now: Whether load site information

Value

- Site information in the WeaAna object
- Site information in the WeaAnaSite object

Examples

```r
library(weaana)
data("WeatherRecordsDemo")
siteInfor(records)
siteInfor(records, load.now = TRUE)
```

---

sphericalDistance  
*Calculate the sphere distance*

Description

Calculate the sphere distance

Usage

```r
sphericalDistance(lat1, lon1, lat2, lon2)
```

Arguments

- lat1: Latitude
- lon1: Longitude
- lat2: Latitude
- lon2: Longitude

Value

Distance in km
thermalTime

Calculate thermal time using cardinal temperatures

Description

Calculate thermal time using cardinal temperatures

Usage

thermalTime(weather, x_temp, y_temp, method = NULL)

Arguments

weather: WeaAna object
x_temp: The cardinal temperatures
y_temp: The effective thermal time
method: The method to calculate thermal time. The default method is \((\text{maxt} + \text{mint}) / 2\)

- base. The three hour temperature methods will be used if method = '3hr'

Value

A data.frame with three columns: year, day and thermalTime.

Examples

```r
met_file <- system.file("extdata/WeatherRecordsDemo1.met", package = "weaana")
records <- readWeatherRecords(met_file)
x_temp <- c(0, 26, 34)
y_temp <- c(0, 26, 0)
res <- thermalTime(records, x_temp, y_temp)
head(res)
res <- thermalTime(records, x_temp, y_temp, method = "3hr")
head(res)
```

thermalTimeDaily

Calculate thermal time using cardinal temperatures

Description

Calculate thermal time using cardinal temperatures

Usage

thermalTimeDaily(mint, maxt, x_temp, y_temp, method = NULL)
thermalTimeHourly

Arguments

mint  The minimum temperature
maxt  The maximum temperature
x_temp  The cardinal temperatures
y_temp  The effective thermal time
method  The method to calculate thermal time. The default method is \((\text{maxt} + \text{mint}) / 2\) - base. The three hour temperature methods will be used if method = '3hr'.

Value

The thermal time.

Examples

```r
mint <- c(0, 10)
maxt <- c(30, 40)
x_temp <- c(0, 20, 35)
y_temp <- c(0, 20, 0)
thermalTimeDaily(mint, maxt, x_temp, y_temp)
thermalTimeDaily(mint, maxt, x_temp, y_temp, method = '3hr')
```

---

thermalTimeHourly  Calculate thermal time using the hourly temperature (non daily temperature)

Description

Calculate thermal time using the hourly temperature (non daily temperature)

Usage

thermalTimeHourly(timestamp, temperature, x_temp, y_temp)

Arguments

timestamp  The timestamp of weather records
temperature  The temperature
x_temp  The cardinal temperatures
y_temp  The effective thermal time

Value

A data frame with daily thermal time
Examples

```r
met_file <- system.file("extdata/WeatherHourly.csv", package = "weaana")
hourly <- read.csv(met_file, as.is = TRUE)

hourly$timestamp <- as.POSIXct(hourly$timestamp, format = "%Y-%m-%dT%H:%M:%SZ")
x_temp <- c(0, 20, 35)
y_temp <- c(0, 20, 0)
thermalTimeHourly(hourly$timestamp, hourly$temperature, x_temp, y_temp)
```

---

ttest_ts

*Significantly t-test with auto-correlation for time serial data*

Description

Method is presented by Santer et al. 2000

Usage

ttest_ts(y, slope = NULL)

Arguments

- **y**: A vector of time serial data
- **slope**: Whether export slope

Value

- p values of t-test

---

WeaAna-class

*Define the class for multiple sites*

Description

Define the class for multiple sites

Slots

- **num**: total number of weather station
- **records**: A pointer vector to weather records of each site
- **result**: A pointer for all results name and type.
Define the class of WeaAna

Slots

name  Name of weather station
number Station number of weather station
latitude Latitude of weather station
longitude Latitude of weather station
tav  Annual average ambient temperature
amp  Annual amplitude in mean monthly temperature
marker The extra marker for this site
year  A vector of year of weather station
day  A vector of day of weather station
radn  A vector of radiation of weather station
maxt  A vector of maximum temperature of weather station
mint  A vector of minimum temperature of weather station
evap  A vector of evaporation of weather station
rain  A vector of rainfall of weather station
vp  A vector of pressure atmosphere of weather station
code The 6 digit code indicates the source of the 6 data columns
extra  A list of variables need to store
res  All statistics results store in this slot
figures  A list to store all plotted figures.
file.path  The file path for this site.
data.format  The data format for this site.
load.later  Whether are records loaded laterly.
writeWeatherRecords  Write weather records into file

Description
Write weather records into file
Write weather records into file

Usage
writeWeatherRecords(object, ...)

## S4 method for signature 'WeaAna'
writeWeatherRecords(object, file, cols = NULL)

Arguments
object  A WeaAna object.
...  Not used
file  Path of output file.
cols  Columns to export. All columns exported if NULL

Value
No return values

[,]WeaAna-method  Getter to access the weather data at a specific position.

Description
Getter to access the weather data at a specific position.

Usage
## S4 method for signature 'WeaAna'
x[i, j, drop]

Arguments
x  A WeaAna object.
i  the specific position which will access.
j  None use parameter.
drop  None use parameter.
\texttt{\textbackslash \text{\textcdot} WeaAna-method}

**Value**

A \texttt{WeaAnaSite} object at the position \texttt{i}.

**Examples**

```r
library(weaana)
data( "WeatherRecordsDemo" )
records[1]
records[1:2]
records[2:2]
```
Index

* datasets
  records, 7
  [,WeaAna-method, 14
  changeWeatherRecords, 2
  changeWeatherRecords,WeaAna,WeaAna-method
    (changeWeatherRecords), 2
  changeWeatherRecords,WeaAna-method
    (changeWeatherRecords), 2
  convert2Records, 3
  createWeaAna, 3

  dayLength, 4
  diurnalT, 4

  getWeatherRecords, 5
  getWeatherRecords,WeaAna,WeaAna-method
    (getWeatherRecords), 5
  getWeatherRecords,WeaAna-method
    (getWeatherRecords), 5

  interpolationFunction, 6

  readWeatherRecords, 6
  records, 7
  result-class, 7

  show,WeaAna-method, 8
  siteInfor, 8
  siteInfor,WeaAna,WeaAna-method
    (siteInfor), 8
  siteInfor,WeaAna-method(siteInfor), 8
  siteInfor,WeaAnaSite,WeaAnaSite-method
    (siteInfor), 8
  siteInfor,WeaAnaSite-method
    (siteInfor), 8
  sphericalDistance, 9

  thermalTime, 10
  thermalTimeDaily, 10
  thermalTimeHourly, 11

  ttest_ts, 12
  WeaAna-class, 12
  WeaAnaSite-class, 13
  writeWeatherRecords, 14
  writeWeatherRecords,WeaAna,WeaAna-method
    (writeWeatherRecords), 14
  writeWeatherRecords,WeaAna-method
    (writeWeatherRecords), 14