Package ‘TDPanalysis’

February 28, 2020

Type Package
Title Granier's Sap Flow Sensors (TDP) Analysis
Version 1.0
Date 2020-02-28
Author Maxime Durand
Maintainer Maxime Durand <duran1211@gmail.com>
Description Set of functions designed to help in the analysis of TDP sensors. Features includes dates and time conversion, weather data interpolation, daily maximum of tension analysis and calculations required to convert sap flow density data to sap flow rates at the tree and plot scale (For more information see : Granier (1985) <DOI:10.1051/forest:19850204> & Granier (1987) <DOI:10.1093/treephys/3.4.309>).
Imports stats, plyr, graphics
Depends R (>= 2.10)
Encoding UTF-8
LazyData true
License GPL-2
RoxygenNote 7.0.2
NeedsCompilation no
Repository CRAN
Date/Publication 2020-02-28 07:40:02 UTC

R topics documented:

  date.to.DOY ................................................................. 2
  datetime ................................................................. 2
  remove.fun ............................................................ 3
  SpFl ..................................................................... 4
  SpWd_Area_calc ......................................................... 4
  tens.to.sapflow ........................................................ 5
  timecont .............................................................. 6
  Tmax.find .............................................................. 6
Tmax.mean .............................................................. 7
Tmaxplot ............................................................. 8
Wat.transp ......................................................... 8

Index 10

date.to.DOY Date conversion

Description
Convert dates from the DD/MM/YYYY format to day of the year (DOY)

Usage
date.to.DOY(dates, format = "dd/mm/yyyy")

Arguments
dates Vector with dates to convert.
format Format of the date (support DD/MM/YYYY MM/DD/YYYY and YYYY/MM/DD).

Value
Return a vector containing the corresponding DOY.

Examples
dates = c("01/01/2000", "03/03/2000", "03/03/1999")
date.to.DOY(dates=dates)

datetime Time & dates conversion

Description
Convert DOY and time into a single numerical variable

Usage
datetime(dates, Time)

Arguments
dates Vector with dates in the DOY format.
Time Vector with time
Details
time vector should be numerical (e.g. as outputed by the time.to.cont function)

Value
Return a vector containing DOY and time as a single numerical variable

Examples
dates = c(102,102,102,102,103,103,103,103)
Time = c(22, 22.5, 23, 23.5, 0, 0.5, 1, 1.5)
datetime(dates=dates, Time=Time)

Description
Remove all data for the corresponding date argument

Usage
remove.fun(df, dates)

Arguments
df Data frame containing a DOY column named "DOY".
dates Character vector containing the DOY to remove from the data frame.

Details
This function is primarily used to remove days for which Tmax is too extreme.

Value
Return the inputed data frame without the date corresponding the the "dates" argument.

Examples
DOY = c(rep(102, times=10), rep(103, times=10))
ID = c(rep("A", times=5), rep("B", times=5), rep("A", times=5), rep("B", times=5))
Tmax = c(rep(2.5, times=5), rep(2.7, times=5), rep(3.2, times=5), rep(3.4, times=5))
df <- data.frame(DOY, ID, Tmax)
dates = c("103")
remove.fun(df=df, dates=dates)
**SpF1**  
*Sap flow dataset*

**Description**

Exemple dataset exemple for the TDPanalysis package

**Usage**

SpF1

**Format**

An object of class data.frame with 432 rows and 4 columns.

**Details**

"DATE" is dates in dd/mm/yyyy format. "TIME" is time in hh:mm:ss format, "ID" is sub-groups and "tension" is the measured tension from the TDP probe.

---

**SpWd_Area_calc**  
*Sapwood area calculation*

**Description**

Calculate sapwood area based on diameter, heartwood diameter and sapwood fraction

**Usage**

SpWd_Area_calc(diam, SpWd_frac = 1, HtWd_diam = 0)

**Arguments**

- **diam** Vector with diameter.
- **SpWd_frac** Numerical (from 0 to 1). Indicate the fraction of the diameter which is sapwood
- **HtWd_diam** Vector with diameter of the heartwood.

**Details**

If SpWD_frac and HtWd_diam are both entered, the function will return an error. Units of "diam" and "HtWd_diam" should be the same.

**Value**

Return a numerical vector containing the sapwood area
### tens.to.sapflow

**Convert tension into sap flow density**

**Description**

Use the Granier formula to convert tension into sap flow density using daily or mean Tmax.

**Usage**

```r
tens.to.sapflow(tension, Tmax)
```

**Arguments**

- `tension`: Vector with tension.
- `Tmax`: Vector with corresponding maximums of tension.

**Value**

Return a numerical vector containing the sap flow density.

**References**


**Examples**

```r
tmax = c(rep(2.5, times=5), rep(2.7, times=5), rep(3.2, times=5), rep(3.4, times=5))
tension = c(5:25)
tens.to.sapflow(tension=tension, Tmax=tmax)
```
timecont

Description

Convert time from the HH:MM:SS format to a numerical

Usage

\[
\text{timecont}(\text{Time}, \text{sep} = ":\)"
\]

Arguments

- \text{Time} : Vector with time to convert.
- \text{sep} : Character element containing regular expression(s) to use to splitting.

Details

Time vector should be in the HH:MM:SS format.

Value

Return a vector containing the corresponding time.

Examples

\[
\text{Time} = \text{c("14:30:00", "20:45:00", "05:00:00")}
\]

\[
\text{timecont}(\text{Time}=\text{Time})
\]

Tmax.find

Description

Find the daily maximum of tension

Usage

\[
\text{Tmax.find}(\text{tension, dates, ID})
\]

Arguments

- \text{tension} : Vector with tension.
- \text{dates} : Vector with dates in the DOY format.
- \text{ID} : Character vector for specifying which group the tension is assigned to (e.g. trees)
Value
Return a vector containing daily Tmax for each group specified in the ID argument

Examples
```
tension = c(1:20)
dates = c(rep(102, times=10), rep(103, times=10))
ID = c(rep("A", times=5), rep("B", times=5), rep("A", times=5), rep("B", times=5))
Tmax.find(tension=tension, dates=dates, ID=ID)
```

---

**Tmax.mean**

*Calculate a mean of Tmax*

Description

Calculate a mean Tmax for each sub-group

Usage

```
Tmax.mean(df)
```

Arguments

df  Data frame containing all Tmax for each sub-group.

Details

The data frame should contain a column named "Tmax" whith all Tmax and a column named "ID" to identify which Tmax belong to which sub-group.

Value

Return the inputed data frame with a new column names "Tmax_mean".

Examples
```
ID = c(rep("A", times=5), rep("B", times=5), rep("A", times=5), rep("B", times=5))
Tmax = c(rep(2.5, times=5), rep(2.7, times=5), rep(3.2, times=5), rep(3.4, times=5))
DOY = c(rep(102, times=10), rep(103, times=10))
df <- data.frame(DOY, ID, Tmax)
Tmax.mean(df)
```
Tmaxplot

Plot the Tmax

Description

Plot the Tmax with indications of extreme values

Usage

Tmaxplot(df)

Arguments

df Data frame containing Tmax, identification of sub-groups and DOY.

Details

The dataframe should contain at least 3 columns named "Tmax" (daily maximums of tension), "DOY" (day of the year) and "ID" (sub-groups). The red horizontal lines represent 3 times the inter-quartile range (3*IQR) of all the Tmax of the data. The blue horizontal line represent the 1.5*IQR without the Tmax outside the red lines.

Value

Return a plot of Tmax by days for each sub-group

Examples

DOY = c(rep(102, times=10), rep(103, times=10))
ID = c(rep("A", times=5), rep("B", times=5), rep("A", times=5), rep("B", times=5))
Tmax = c(rep(0.7512, times=5), rep(0.7359, times=5), rep(0.7644, times=5), rep(0.7666, times=5))
df <- data.frame(DOY, ID, Tmax, stringsAsFactors = FALSE)
Tmaxplot(df)

Wat.transp

Calculate daily transpiration

Description

Calculate daily transpiration for each sub-group inputed

Usage

Wat.transp(Sapflow, days, ID)
**Wat.transp**

**Arguments**
- **Sapflow** Vector with sap flow.
- **days** Vector containing the days for which to calculate transpiration
- **ID** Character vector containing identification for each sub-group

**Details**

!!Beware of the units!! The Granier formula usually convert tension into sap flow density (in kg.dm\(^{-2}\).h\(^{-1}\)). So, you should first convert sap flow density into sap flow (in kg.h\(^{-1}\)). Moreover, if you take measurement every 30 minutes sap flow should be corrected by dividing the value by 2.

**Value**

Return a data frame with transpiration for each day and sub-group inputed

**Examples**

```r
ID = c(rep("A", times=5), rep("B", times=5), rep("A", times=5), rep("B", times=5))
Sapflow = c(rep(2.5, times=5), rep(2.7, times=5), rep(3.2, times=5), rep(3.4, times=5))
days = c(rep(102, times=10), rep(103, times=10))
Wat.transp(Sapflow=Sapflow, days=days, ID=ID)
```
Index

*Topic datasets
  SpFl, 4

  date.to.DOF, 2
datetime, 2

remove.fun, 3

SpFl, 4
SpWd_Area_calc, 4
tens.to.sapflow, 5
timecont, 6
Tmax.find, 6
Tmax.mean, 7
Tmaxplot, 8
Wat.transp, 8