Package ‘Evapotranspiration’

July 2, 2014

Version 1.4

Date 2014-06-12

Title Evapotranspiration

Author Danlu Guo <danlu.guo@adelaide.edu.au>, Seth Westra <swestra@civeng.adelaide.edu.au>

Maintainer Danlu Guo <danlu.guo@adelaide.edu.au>

Depends R (>= 2.10), zoo

Description Uses data and constants to calculate potential evapotranspiration (PET) and actual evapotranspiration (AET) from 17 different formulations including Penman, Penman-Monteith FAO 56, Priestley-Taylor and Morton formulations.

NeedsCompilation no

Repository CRAN

Date/Publication 2014-06-12 09:28:08

R topics documented:

`climatedata` *Raw Climate Data Required for Calculating Evapotranspiration at the Kent Town station in Adelaide, Australia*

Description

This data set contains the raw climate data including the variables required for calculating evapotranspiration in function `ET` over the observation period between 1/3/2001 and 08/31/2004 at the Kent Town station in Adelaide, Australia.

Usage

```
climatedata
```

Format

A data frame containing 10240 observations of 13 objects: "Station.Number" - weather station number, "Year" - year of record, "Month" - month of record, "Day" - day of record, "Hour" - hour of record, "Julian" - Julian day of record, "Temp.subdaily" - subdaily temperature data in degree Celcius, "Tdew.subdaily" - subdaily dew point temperature data in degree Celcius, "RH.subdaily" - subdaily relative humidity data in degree Celcius, "n.daily" - daily sunshine hour data in hours, "uz.subdaily" - subdaily wind speed data in kilometres per hour, "Tmin.daily" - daily maximum temperature data in degree Celcius, "Tmax.daily" - daily minimum temperature data in degree Celcius.

Source

Bureau of Meteorology, Kent Town, Adelaide, Australia constants *Constants Required for Calculating Evapotranspriation*

Description

This data set contains the constants required for calculating evapotranspiration in function `ET` for Kent Town station in Adelaide, Australia.

Usage

```
climatedata
```

Format

A list containing 46 constant values including: "lambda", latent heat of evaporisationin = 2.45 MJ.kg^-1 at 20 degree Celcius, "sigma", Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1. "Gsc", solar constant = 0.0820 MJ.m^-2.min^-1, "lat", latitude = -34.9211 degrees for Kent Town station, "lat_rad", latitude in radians = -0.6095 radians for Kent Town station, "as", fraction of extraterrestrial radiation reaching earth on sunless days = 0.23 for Australia (Roderick, 1999, page 181), "bs", difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days = =0.5 for Australia (Roderick, 1999, page 181), "Elev", ground elevation above mean sea level = 48m for Kent Town station, "z", height of wind instrument = 10m for Kent Town station, "Roua", mean density of air = 1.2 kg.m^-3 at 20 degree Celcius, "Ca", specific heat of air = 0.001013 MJ.kg^-1.K^-1, "G", soil heat flux negligible for daily time-step = 0 (Allen et al., 1998, page 68), "alphaA", Albedo for Class-A pan = 0.14, "alphaPT", Priestley-Taylor coefficient = 1.26 for Priestley-Taylor formula (Priestley and Taylor, 1972, Sect. 6; Eichinger et al., 1996, p.163); = 1.31 for Szilagyi-Jozsa formula (Szilagyi and Jozsa, 2008); = 1.28 for Brutsaert-Strickler formula (Brutsaert and Strickler, 1979), "ap", constant in Penpan formula = 2.4, "fz", constant in Morton's procedure = 28.0 W.m^-2.mbar^-1 for CRAE model for T >= 0 degree Celcius, = 28.0*1.15 W.m^-2.mbar^-1 for CRAE model for T < 0 degree Celcius; = 25.0 W.m^-2.mbar^-1 for CRWE model for T >= 0 degree Celcius, = 28.75 W.m^-2.mbar^-1 for CRWE model for T < 0 degree Celcius (Morton, 1983a, page65), "b0", constant in Morton's procedure = 1 (Chiew and McMahon, 1991, Table A1), "b1", constant in Morton's procedure = 14 W.m^-2 (Chiew and McMahon, 1991, Table A1), "b2", constant in Morton's procedure = 1.2 (Chiew and McMahon, 1991, Table A1), "a_0", constant for estimating sunshine hours from cloud cover data = 11.9 for Adelaide (Chiew and McMahon, 1991, Table A1), "b_0", constant for estimating sunshine hours from cloud cover data = -0.15 for Adelaide, "c_0", constant for estimating sunshine hours from cloud cover data = -0.25 for Adelaide, "d_0", constant for estimating sunshine hours from cloud cover data = -0.0107 for Adelaide "e0", constant for Blaney-Criddle formula = 0.81917 (Frevert et al., 1983, Table 1), "e1", constant for Blaney-Criddle formula = -0.0040922 (Frevert et al., 1983, Table 1), "e2", constant for Blaney-Criddle formula = 1.0705 (Frevert et al., 1983, Table 1), "e3", constant for Blaney-Criddle formula = 0.065649 (Frevert et al., 1983, Table 1), "e4", constant for Blaney-Criddle formula = -0.0059864 (Frevert et al., 1983, Table 1), "e5", constant for Blaney-Criddle formula = -0.0005967 (Frevert et al., 1983, Table 1), "gammaps", product of Psychrometric constant and atmospheric pressure as sea level = 0.66 mbar. degree Celcius^-1 for CRAE model for T >= 0 degree Celcius, = 0.66/1.15 mbar. degree Celcius^-1 for CRAE model for T < 0 degree Celcius, "epsilonMo", Land surface emissivity in Morton's procedure = 0.92, "PA", annual precipitation = 285.8mm for Kent Town station, "alphaMo", constant in Morton's procedure = 17.27 when T >= 0 degree Celcius, = 21.88 when T < 0 degree Celcius, "betaMo", constant in Morton's procedure = 237.3 degree Celcius when T >= 0 degree Celcius, = 265.5 degree Celcius when T < 0 degree Celcius, "sigmaMo", Stefan-Boltzmann constant in Morton's procedure = 5.67e-08 W.m^-2.K^-4, "lambdaMo", latent heat of vaporisation in Morton's procedure = 28.5W.day.kg^-1 when T >= 0 degree Celcius, = 28.5*1.15W.day.kg^-1 when T < 0 degree Celcius.

Source

various references

`data` — *Processed Climate Data Required for Calculating Evapotranspiration*

Description

This data set contains the processed climate data including the variables required for calculating evapotranspiration in function `ET` over the observation period between 1/3/2001 and 31/8/2004 at the Kent Town station in Adelaide, Australia.

Usage

```
data
```

Format

A list containing 17 climate variables

Source

Bureau of Meteorology, Kent Town, Adelaide, Australia

`ET` — *ET Formulations*

Description

A generic function for estimating evapotranspiration from different formulations according to class of data.

Usage

```
ET(data, ...)
```

Arguments

`data`	A list of climate data required for estimating evapotranspiration which differs for each evapotranspiration formulations, see specific formulations for details.
`...`	Arguments to be passed to methods which differs for each evapotranspiration formulations, see specific formulations for details.

Details

The class of data must be defined prior to call this function in order to proceed to specific formulation.

Author(s)

Danlu Guo

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Penman" to call function
funname <- "Penman"
class(data) <- funname

# Call generic function ET(data, ...) with class "Penman"
results <- ET(data, constants, solar="sunshine hours", wind=
"yes", windfunction_ver = "1948", alpha = 0.08, z0 = 0.001)
```

`ET.BlaneyCriddle` *Blaney-Criddle Formulation*

Description

Implementing the Blaney-Criddle formulation for estimating reference crop evapotranspiration.

Usage

```
## S3 method for class 'BlaneyCriddle'
ET(data, constants, solar, height, ...)
```

Arguments

`data`	A list of data in class "BlaneyCriddle" which contains the following items (climate variables) required by Blaney-Criddle formulation: "Tmax", "Tmin", "RHmin", "n", "u2" or "uz"

constants — A list named "constants" consists of constants required for the calculation of Blaney-Criddle formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lat_rad" - latitude in radians, "z" - height of wind instrument in m, "e0", "e1", "e2", "e3", "e4" - recommended values of 0.81917, -0.0040922, 1.0705, 0.065649, -0.0059684, -0.0005967 respectively (Table 1 in Frevert et al., 1983).

solar — Must be either "sunshine hours" or "cloud". "sunshine hours" indicates that solar radiation is to be calculated using the data of sunshine hours; for "cloud" sunshine hours is to be estimated from cloud data.

height — Must be TRUE or FALSE, indicating if adjustment for site elevation for arid and semi-arid regions is applied in Blaney-Criddle formulation (Allen and Brockway, 1983).

... — Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details. Height adjustment for the estimations is available through argument `height`, please see "Arguments" for details.

Value

This function returns a object of class `BlaneyCriddle`. Such objects are list with components:

ET.Daily — A `zoo` object containing daily aggregated estimations of Blaney-Criddle reference crop evapotranspiration.

ET.Monthly — A `zoo` object containing monthly aggregated estimations of Blaney-Criddle reference crop evapotranspiration.

ET.Annual — A `zoo` object containing annually aggregated estimations of Blaney-Criddle reference crop evapotranspiration.

ET.MonthlyAve — A `zoo` object containing monthly averaged estimations of daily Blaney-Criddle reference crop evapotranspiration.

ET.AnnualAve — A `zoo` object containing annually averaged estimations of daily Blaney-Criddle reference crop evapotranspiration.

ET_formulation — A character string containing the name of the formulation used which equals to "Blaney-Criddle".

ET_type — A character string containing the type of the estimation obtained which is "Reference Crop Evapotranspiration".

message1 — A message to inform the users about how solar radiation has been calculated by using which data.

message3 — A message to inform the users about if height adjustment has been applied to calculated Blaney-Criddle reference crop evapotranspiration.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

ALLEN, R. & PRUITT, W. 1986. Rational Use of The FAO Blaney-Criddle Formula. Journal of Irrigation and Drainage Engineering, 112, 139-155.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "BlaneyCriddle" to call function
funname <- "BlaneyCriddle"
class(data) <- funname

# Call generic function ET(data, ...) with class "BlaneyCriddle"
results <- ET(data, constants, solar="sunshine.hours", height = TRUE)
```

`ET.BrutsaertStrickler` *Brutsaert-Strickler Formulation*

Description

Implementing the Brutsaert-Strickler formulation for actual areal evapotranspiration

Usage

```
## S3 method for class 'BrutsaertStrickler'
ET(data, constants, solar, alpha, ...)
```

Arguments

`data`	A list of data in class "BrutsaertStrickler" which contains the following items (climate variables) required by Brutsaert-Strickler formulation: "Tmax", "Tmin", "RHmax", "RHmin", "RS", "n" or "Cd", "u2" or "uz"
`constants`	A list named "constants" consists of constants required for the calculation of Brutsaert-Strickler formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1.

`solar`	Must be either "data", "sunshine hours", cloud" or "monthly precipitation". "data" indicates that solar radiation data is used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.
`alpha`	Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.
`...`	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details. User-defined evaporative surface is allowed through argument `alpha`, please see "Arguments" for details.

Value

This function returns a object of class `BrutsaertStrickler`. Such objects are list with components:

`ET.Daily`	A `zoo` object containing daily aggregated estimations of Brutsaert-Strickler actual areal evapotranspiration.
`ET.Monthly`	A `zoo` object containing monthly aggregated estimations of Brutsaert-Strickler actual areal evapotranspiration.
`ET.Annual`	A `zoo` object containing annually aggregated estimations of Brutsaert-Strickler actual areal evapotranspiration.
`ET.MonthlyAve`	A `zoo` object containing monthly averaged estimations of daily Brutsaert-Strickler actual areal evapotranspiration.
`ET.AnnualAve`	A `zoo` object containing annually averaged estimations of daily Brutsaert-Strickler actual areal evapotranspiration.
`ET_formulation`	A character string containing the name of the formulation used which equals to "Brutsaert-Strickler".
`ET_type`	A character string containing the type of the estimation obtained which is "Actual Areal Evapotranspiration".
`message1`	A message to inform the users about how solar radiation has been calculated by using which data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "BrutsaertStrickler" to call function
funname <- "BrutsaertStrickler"
class(data) <- funname

# Call generic function ET(data, ...) with class "BrutsaertStrickler"
results <- ET(data, constants, solar="sunshine.hours", alpha = 0.08)
```

ET.ChapmanAustralian *Chapman Formulation*

Description

Implementing the Chapman formulation for estimating equivalent Penmen-Monteith reference crop evapotranspiration.

Usage

```
## S3 method for class 'ChapmanAustralian'
ET(data, constants, Penpan, solar, alpha, ...)
```

Arguments

`data` A list of data in class "ChapmanAustralian" which contains the following items (climate variables) required by Chapman formulation: "Epan" or "Tmax" and "Tmin" and "RHmax" and "RHmin" and "u2" or "uz" and "Rs" or "n" or "Cd"

`constants` A list named "constants" consists of constants required for the calculation of Penman formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "lat" - latitude in degrees, "alphaA" - albedo for Class-A pan, "ap" - a constant in Penpan = 2.4.

`Penpan` Must be TRUE or FALSE, indicating if the Penpan formulation is used for estimating Class-A pan evaporation required in Chapman formulation. If TRUE Penpan will be used and if FALSE the actual data of Class-A pan evaporation will be used.

`solar` Must be either "data", "sunshine hours", or "cloud". "data" indicates that solar radiation data is used directly for calculating evapotranspiration, "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data.

`alpha`	Must be defined if Penpan = TRUE. Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.
`...`	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through arguments `Penpan` and `solar`, please see "Arguments" for details.

Value

This function returns a object of class `ChapmanAustralian`. Such objects are list with components:

`ET.Daily`	A `zoo` object containing daily aggregated estimations of Chapman equivalent Penmen-Monteith reference crop evapotranspiration.
`ET.Monthly`	A `zoo` object containing monthly aggregated estimations of Chapman equivalent Penmen-Monteith reference crop evapotranspiration.
`ET.Annual`	A `zoo` object containing annually aggregated estimations of Chapman equivalent Penmen-Monteith reference crop evapotranspiration.
`ET.MonthlyAve`	A `zoo` object containing monthly averaged estimations of daily Chapman equivalent Penmen-Monteith reference crop evapotranspiration.
`ET.AnnualAve`	A `zoo` object containing annually averaged estimations of daily Chapman equivalent Penmen-Monteith reference crop evapotranspiration.
`ET_formulation`	A character string containing the name of the formulation used which equals to "Chapman".
`ET_type`	A character string containing the type of the estimation obtained which is "Equivalent Penmen-Monteith Reference Crop Evapotranspiration".
`message1`	A message to inform the users about how solar radiation has been calculated by using which data.
`message5`	A message to inform the users about if the Class-A pan evaporation is from actual data or from Penpan estimation.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "ChapmanAustralian" to call function
funname <- "ChapmanAustralian"
class(data) <- funname

# Call generic function ET(data, ...) with class "ChapmanAustralian"
results <- ET(data, constants, Penpan = TRUE, solar = "sunshine hours",
alpha = 0.23)
```

ET.GrangerGray *Granger-Gray Formulation*

Description

Implementing the Granger-Gray formulation for estimating actual areal evapotranspiration.

Usage

```
## S3 method for class 'GrangerGray'
ET(data, constants, solar, windfunction_ver, alpha, ...)
```

Arguments

`data`	A list of data in class "GrangerGray" which contains the following items (climate variables) required by Granger-Gray formulation: "Tmax", "Tmin", "RHmax", "RHmin", "Rs", "n" or "Cd", "u2" or "uz"
`constants`	A list named "constants" consists of constants required for the calculation of Granger-Gray formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "G" - soil heat flux in MJ.m^-2.day^-1, = 0 when using daily time step.
`solar`	Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation data is to be used to calculate evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.

windfunction_ver
: The version of Penman wind function that will be used within the Penman formulation. Must be either "1948" or "1956". "1948" is for applying the Penman's 1948 wind function (Penman, 1948); "1956" is for applying the Penman's 1956 wind function (Penman, 1956)

alpha
: Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.

...
: Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through arguments `solar` and `windfunction_ver`, please see "Arguments" for details. User-defined evaporative surface is allowed through argument `alpha`, please see "Arguments" for details.

Value

This function returns a object of class `GrangerGray`. Such objects are list with components:

ET.Daily
: A `zoo` object containing daily aggregated estimations of Granger-Gray actual areal evapotranspiration.

ET.Monthly
: A `zoo` object containing monthly aggregated estimations of Granger-Gray actual areal evapotranspiration.

ET.Annual
: A `zoo` object containing annually aggregated estimations of Granger-Gray actual areal evapotranspiration.

ET.MonthlyAve
: A `zoo` object containing monthly averaged estimations of daily Granger-Gray actual areal evapotranspiration.

ET.AnnualAve
: A `zoo` object containing annually averaged estimations of daily Granger-Gray actual areal evapotranspiration.

ET_formulation
: A character string containing the name of the formulation used which equals to "Granger-Gray".

ET_type
: A character string containing the type of the estimation obtained which is "Actual Areal Evapotranspiration".

message1
: A message to inform the users about how solar radiation has been calculated by using which data.

message2
: A message to inform the users about which version of the Penman wind function has been used.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

PENMAN, H. L. 1948. Natural evaporation from open water, bare soil and grass. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 193, 120-145.

PENMAN, H. L. 1956. Evaporation: An introductory survey. Netherlands Journal of Agricultural Science, 4, 9-29.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "GrangerGray" to call function
funname <- "GrangerGray"
class(data) <- funname

# Call generic function ET(data, ...) with class "GrangerGray"
results <- ET(data, constants, solar="sunshine hours", windfunction_ver =
"1948", alpha = 0.23)
```

`ET.HargreavesSamani` *Hargreaves-Samani Formulation*

Description

Implementing the Hargreaves-Samani formulation for estimating reference crop evapotranspiration.

Usage

```
## S3 method for class 'HargreavesSamani'
ET(data, constants, ...)
```

Arguments

`data`	A list of data in class "HargreavesSamani" which contains the following items (climate variables) required by Hargreaves-Samani formulation: "Tmax", "Tmin"
`constants`	A list named "constants" consists of constants required for the calculation of Hargreaves-Samani formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1.
`...`	Dummy for generic function, no need to define.

Details

This formulation provides a single calculation method with no alternatives available.

Value

This function returns a object of class `HargreavesSamani`. Such objects are list with components:

- `ET.Daily` A zoo object containing daily aggregated estimations of Hargreaves-Samani reference crop evapotranspiration.
- `ET.Monthly` A zoo object containing monthly aggregated estimations of Hargreaves-Samani reference crop evapotranspiration.
- `ET.Annual` A zoo object containing annually aggregated estimations of Hargreaves-Samani reference crop evapotranspiration.
- `ET.MonthlyAve` A `zoo` object containing monthly averaged estimations of daily Hargreaves-Samani reference crop evapotranspiration.
- `ET.AnnualAve` A `zoo` object containing annually averaged estimations of daily Hargreaves-Samani reference crop evapotranspiration.
- `ET_formulation` A character string containing the name of the formulation used which equals to "Hargreaves-Samani".
- `ET_type` A character string containing the type of the estimation obtained which is "Reference Crop Evapotranspiration".

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

HARGREAVES, G. H. & SAMANI, Z. A. 1985. Reference crop evapotranspiration from ambient air temperature. American Society of Agricultural Engineers.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "HargreavesSamani" to call function
funname <- "HargreavesSamani"
class(data) <- funname

# Call generic function ET(data, ...) with class "HargreavesSamani"
results <- ET(data, constants)
```

`ET.JensenHaise` *Jensen-Haise Formulation*

Description

Implementing the Jensen-Haise formulation for estimating potential evapotranspiration.

Usage

```
## S3 method for class 'JensenHaise'
ET(data, constants, ...)
```

Arguments

`data`	A list of data in class "JensenHaise" which contains the following items (climate variables) required by Jensen-Haise formulation: "Tmax", "Tmin"
`constants`	A list named "constants" consists of constants required for the calculation of Penman formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1.
`...`	Dummy for generic function, no need to define.

Details

This formulation provides a single calculation method with no alternatives available.

Value

This function returns a object of class `JensenHaise`. Such objects are list with components:

`ET.Daily`	A `zoo` object containing daily aggregated estimations of Jensen-Haise potential evapotranspiration.
`ET.Monthly`	A `zoo` object containing monthly aggregated estimations of Jensen-Haise potential evapotranspiration.
`ET.Annual`	A `zoo` object containing annually aggregated estimations of Jensen-Haise potential evapotranspiration.
`ET.MonthlyAve`	A `zoo` object containing monthly averaged estimations of daily Jensen-Haise potential evapotranspiration.
`ET.AnnualAve`	A `zoo` object containing annually averaged estimations of daily Jensen-Haise potential evapotranspiration.
`ET_formulation`	A character string containing the name of the formulation used which equals to "Jensen-Haise".
`ET_type`	A character string containing the type of the estimation obtained which is "Potential Evapotranspiration".

Author(s)

Danlu Guo

References

JENSEN, M. E. & HAISE, H. R. 1963. Estimating evapotranspiration from solar radiation. Proceedings of the American Society of Civil Engineers, Journal of the Irrigation and Drainage Division, 89, 15-41.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "JensenHaise" to call function
funname <- "JensenHaise"
class(data) <- funname

# Call generic function ET(data, ...) with class "JensenHaise"
results <- ET(data, constants)
```

ET.Makkink *Makkink Formulation*

Description

Implementing the Makkink formulation for estimating potential evaporation.

Usage

```
## S3 method for class 'Makkink'
ET(data, constants, solar, ...)
```

Arguments

data	A list of data in class "Makkink" which contains the following items (climate variables) required by Makkink formulation: "Tmax", "Tmin", "Rs", "n" or "Cd"
constants	A list named "constants" consists of constants required for the calculation of Makkink formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days.

`solar` Must be either "sunshine hours", "data", "cloud" or "monthly precipitation". "sunshine hours" indicates that solar radiation data will be used directly for calculating evapotranspiration; "data" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.

`...` Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details.

Value

This function returns a object of class `Makkink`. Such objects are list with components:

`ET.Daily` A `zoo` object containing daily aggregated estimations of Makkink potential evaporation.

`ET.Monthly` A `zoo` object containing monthly aggregated estimations of Makkink potential evaporation.

`ET.Annual` A `zoo` object containing annually aggregated estimations of Makkink potential evaporation.

`ET.MonthlyAve` A `zoo` object containing monthly averaged estimations of daily Makkink potential evaporation.

`ET.AnnualAve` A `zoo` object containing annually averaged estimations of daily Makkink potential evaporation.

`ET_formulation` A character string containing the name of the formulation used which equals to "Makkink".

`ET_type` A character string containing the type of the estimation obtained which is "Potential Evaporation".

`message1` A message to inform the users about how solar radiation has been calculated by using which data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Makkink" to call function
funname <- "Makkink"
class(data) <- funname

# Call generic function ET(data, ...) with class "Makkink"
results <- ET(data, constants, solar="sunshine hours")
```

ET.MattShuttleworth *Matt-Shuttleworth Formulation*

Description

Implementing the Matt-Shuttleworth formulation for reference crop evapotranspiration

Usage

```
## S3 method for class 'MattShuttleworth'
ET(data, constants, solar, alpha, r_s, CH, ...)
```

Arguments

`data`	A list of data in class "MattShuttleworth" which contains the following items (climate variables) required by Matt-Shuttleworth formulation: "Tmax", "Tmin", "RHmax", "RHmin" "n" or "Cd" or "Precip", "u2" or "uz"
`constants`	A list named "constants" consists of constants required for the calculation of Matt-Schuttleworth formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "Roua" - mean air density = 1.20 kg.m^-3, "Ca" - specific heat of air = 0.001013 MJ.kg^-1.oC^-1.
`solar`	Must be either "sunshine hours", "data", "cloud" or "monthly precipitation". "data" indicates that solar radiation data will be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.

alpha	Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.
r_s	Any value (seconds per metre), surface resistance depends on the type of reference crop.
CH	Any value (metres), crop height depends on the reference crop.
...	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details. User-defined evaporative surface is allowed through arguments `alpha`, `r_s` and `CH`, please see "Arguments" for details.

Value

This function returns a object of class `MattShuttleworth`. Such objects are list with components:

ET.Daily	A `zoo` object containing daily aggregated estimations of Matt-Shuttleworth reference crop evapotranspiration.
ET.Monthly	A `zoo` object containing monthly aggregated estimations of Matt-Shuttleworth reference crop evapotranspiration.
ET.Annual	A `zoo` object containing annually aggregated estimations of Matt-Shuttleworth reference crop evapotranspiration.
ET.MonthlyAve	A `zoo` object containing monthly averaged estimations of daily Matt-Shuttleworth reference crop evapotranspiration.
ET.AnnualAve	A `zoo` object containing annually averaged estimations of daily Matt-Shuttleworth reference crop evapotranspiration.
ET_formulation	A character string containing the name of the formulation used which equals to "Matt-Shuttleworth".
ET_type	A character string containing the type of the estimation obtained which is "Reference Crop Evapotranspiration".
message1	A message to inform the users about how solar radiation has been calculated by using which data.

Author(s)

Danlu Guo

References

SHUTTLEWORTH, W. & WALLACE, J. 2009. Calculating the water requirements of irrigated crops in Australia using the Matt-Shuttleworth approach. Transactions of the ASABE, 52, 1895-1906.

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "MattShuttleworth" to call function
funname <- "MattShuttleworth"
class(data) <- funname

# Call generic function ET(data, ...) with class "MattShuttleworth"
results <- ET(data, constants, solar="sunshine hours", alpha = 0.23, r_s = 70, CH =
0.12)
```

ET.McGuinnessBordne *McGuinness-Bordne Formulation*

Description

Implementing the McGuinness-Bordne formulation for estimating potential evapotranspiration.

Usage

```
## S3 method for class 'McGuinnessBordne'
ET(data, constants, ...)
```

Arguments

data	A list of data in class "McGuinnessBordne" which contains the following items (climate variables) required by McGuinness-Bordne formulation: "Tmax", "Tmin"
constants	A list named "constants" consists of constants required for the calculation of Penman formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1.
...	Dummy for generic function, no need to define.

Details

This formulation provides a single calculation method with no alternatives available.

Value

This function returns a object of class `McGuinnessBordne`. Such objects are list with components:

ET.Daily	A zoo object containing daily aggregated estimations of McGuinness-Bordne potential evapotranspiration.
ET.Monthly	A zoo object containing monthly aggregated estimations of McGuinness-Bordne potential evapotranspiration.

`ET.Annual`	A zoo object containing annually aggregated estimations of McGuinness-Bordne potential evapotranspiration.
`ET.MonthlyAve`	A zoo object containing monthly averaged estimations of daily McGuinness-Bordne potential evapotranspiration.
`ET.AnnualAve`	A zoo object containing annually averaged estimations of daily McGuinness-Bordne potential evapotranspiration.
`ET_formulation`	A character string containing the name of the formulation used which equals to "McGuinness-Bordne".
`ET_type`	A character string containing the type of the estimation obtained which is "Potential Evapotranspiration".

Author(s)

Danlu Guo

References

XU, C. Y. & SINGH, V. P. 2000. Evaluation and generalization of radiation-based methods for calculating evaporation. Hydrological Processes, 14, 339-349.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "McGuinnessBordne" to call function
funname <- "McGuinnessBordne"
class(data) <- funname

# Call generic function ET(data, ...) with class "McGuinnessBordne"
results <- ET(data, constants)
```

`ET.MortonCRAE` *Morton CRAE Formulation*

Description

Implementing the Morton CRAE formulation for estimating potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.

Usage

```
## S3 method for class 'MortonCRAE'
ET(data, constants, est, solar, Tdew, ...)
```

Arguments

- `data` A list of data in class "MortonCRAE" which contains the following items (climate variables) required by Morton CRAE formulation: "Tmax", "Tmin", "Tdew", "n" or "Cd".
- `constants` A list named "constants" consists of constants required for the calculation of Morton CRAE formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lat_rad" - latitude in radians, "PA" - annual precipitation in mm, required when precipitation data is not available, "epsilonMo"" - surface emissivity = 0.92 for CRAE model (Morton, 1986) "fz" - A constant in Morton's procedure = 28.0 Wm^-2.mbar^-1 for T >= 0 degree Celcius, and = 28.0*1.15 Wm^-2.mbar^-1 for T >= 0 degree Celcius for CRAE model (Morton, 1983) "b0" - a constants in Morton's procedure, = 1 for CRAE model (Morton, 1983) "b1" - a constant in Morton's procedure, = 14 for CRAE model (Morton, 1983) "b2 - a constant in Morton's procedure, = 1.2 for CRAE model (Morton, 1983) "gammaps" - Produce of Psychrometric constant and atmospheric pressure as sea level, = 0.66 mbar. degree Celcius^-1 for CRAE model for T >= 0 degree Celcius, = 0.66/1.15 mbar. degree Celcius^-1 for CRAE model for T < 0 degree Celcius (Morton, 1983), "lat" - latitude in degrees, "alphaMo" - a constant in Morton's procedure, = 17.27 when T >= 0 degree Celcius, = 21.88 when T < 0 degree Celcius (Morton, 1983), "betaMo" - a constant in Morton's procedure, = 237.3 degree Celcius when T >= 0 degree Celcius,, = 2655. degree Celcius, when T < 0 degree Celcius (Morton, 1983), "sigmaMo" - Stefan-Boltzmann constant in Morton's procedure, = 5.67e-08 W.m^-2.K^-4 (Morton, 1983), "lambdaMo" - Latent heat of vaporisation in Morton's procedure, = 28.5W.day.kg^-1 when T >= 0 degree Celcius, = 28.5*1.15W.day.kg^-1 when T < 0 degree Celcius.
- `solar` Must be either "sunshine hours" or "cloud". "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data.
- `est` Must be either "potential ET", "wet areal ET" or "actual areal ET". "potential ET" proceeds to estimating potential evapotranspiration; "wet areal ET" proceeds to estimating wet-environmental areal evapotranspiration; "actual areal ET" proceeds to estimating actual areal evapotranspiraion
- `Tdew` Logical, must be either TRUE or FALSE, indicating if real data of dew point temperature is used for calculating the radiation in Morton's formulations, if TRUE the data will be used and if FALSE the dew point temperature will be calculated from data of daily vapour pressure.
- `...` Dummy for generic function, no need to define.

Details

The type of evapotranspiration calculated can be selected through argument `est`, please see "Arguments" for details. The alternative calculation options can be selected through argument `solar` and `Tdew`, please see "Arguments" for details.

Value

This function returns a object of class `MortonCRAE`. Such objects are list with components:

- `ET.Daily` A zoo object containing daily aggregated estimations of Morton CRAE potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.
- `ET.Monthly` A zoo object containing monthly aggregated estimations of Morton CRAE potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.
- `ET.Annual` A zoo object containing annually aggregated estimations of Morton CRAE potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.
- `ET.MonthlyAve` A zoo object containing monthly averaged estimations of daily Morton CRAE potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.
- `ET.AnnualAve` A zoo object containing annually averaged estimations of daily Morton CRAE potential evapotranspiration, wet-environment areal evapotranspiration and actual areal evapotranspiration.
- `ET_formulation` A character string containing the name of the formulation used which equals to "MortonCRAE".
- `ET_type` A character string containing the type of the estimation obtained which is either "Potential Evapotranspiration", "Wet-environment Areal Evapotranspiration" and "Actual Areal Evapotranspiration".
- `message1` A message to inform the users about how solar radiation has been calculated by using which data.
- `message6` A message to inform the users about if actual dew point temperature has been used in the calculations or alternative calculations has been performed without dew point temperature data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

MORTON, F. I. 1983. Operational estimates of areal evapotranspiration and their significance to the science and practice of hydrology. Journal of Hydrology, 66, 1-76.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "MortonCRAE" to call function
funname <- "MortonCRAE"
class(data) <- funname
```

```
# Call generic function ET(data, ...) with class "MortonCRAE"
results <- ET(data, constants, est = "potential ET", solar =
"sunshine hours", Tdew = TRUE)
```

`ET.MortonCRWE` *Morton CRWE Formulation*

Description

Implementing the Morton CRWE formulation for estimating potential evapotranspiration or shallow lake evaporation.

Usage

```
## S3 method for class 'MortonCRWE'
ET(data, constants, est, solar, Tdew, ...)
```

Arguments

`data`	A list of data in class "MortonCRWE" which contains the following items (climate variables) required by Morton CRWE formulation: "Tmax", "Tmin", "Tdew", "n" or "Cd".
`constants`	A list named "constants" consists of constants required for the calculation of Morton CRWE formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lat_rad" - latitude in radians, "PA" - annual precipitation in mm, required when precipitation data is not available, "epsilonMo"" - surface emissivity = 0.92 for CRWE model (Morton, 1983) "fz" - A constant in Morton's procedure = 25.0 Wm^-2.mbar^-1 for T >= 0 degree Celcius, and = 28.75 Wm^-2.mbar^-1 for T >= 0 degree Celcius for CRWE model (Morton, 1983) "b0" - a constants in Morton's procedure, = 1.12 for CRWE model (Morton, 1983) "b1" - a constant in Morton's procedure, = 13 for CRWE model (Morton, 1983) "b2 - a constant in Morton's procedure, = 1.12 for CRWE model (Morton, 1983) "gammaps" - Produce of Psychrometric constant and atmospheric pressure as sea level, = 0.66 mbar. degree Celcius^-1 for CRAE model for T >= 0 degree Celcius, = 0.66/1.15 mbar. degree Celcius^-1 for CRAE model for T < 0 degree Celcius (Morton, 1983), "lat" - latitude in degrees, "alphaMo" - a constant in Morton's procedure, = 17.27 when T >= 0 degree Celcius, = 21.88 when T < 0 degree Celcius (Morton, 1983), "betaMo" - a constant in Morton's procedure, = 237.3 degree Celcius when T >= 0 degree Celcius,, = 2655. degree Celcius, when T < 0 degree Celcius (Morton, 1983), "sigmaMo" - Stefan-Boltzmann constant in Morton's procedure, = 5.67e-08 W.m^-2.K^-4 (Morton, 1983), "lambdaMo" - Latent heat of vaporisation in Morton's procedure, = 28.5W.day.kg^-1 when T >= 0 degree Celcius, = 28.5*1.15W.day.kg^-1 when T < 0 degree Celcius.
`solar`	Must be either "sunshine hours" or "cloud". "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data.

`est`	Must be either "potential" or "shallow lake". "potential" proceeds to estimating potential evapotranspiration; "shallow lake" proceeds to estimating shallow lake evaporation.
`Tdew`	Logical, must be either TRUE or FALSE, indicating if real data of dew point temperature is used for calculating the radiation in Morton's formulations, if TRUE the data will be used and if FALSE the dew point temperature will be calculated from data of daily vapour pressure.
`...`	Dummy for generic function, no need to define.

Details

The type of evapotranspiration calculated can be selected through argument `est`, please see "Arguments" for details. The alternative calculation options can be selected through argument `solar` and `Tdew`, please see "Arguments" for details.

Value

This function returns a object of class `MortonCRWE`. Such objects are list with components:

`ET.Daily`	A `zoo` object containing daily aggregated estimations of MortonCRWE potential evapotranspiration or shallow lake evaporation.
`ET.Monthly`	A `zoo` object containing monthly aggregated estimations of MortonCRWE potential evapotranspiration or shallow lake evaporation.
`ET.Annual`	A `zoo` object containing annually aggregated estimations of MortonCRWE potential evapotranspiration or shallow lake evaporation.
`ET.MonthlyAve`	A `zoo` object containing monthly averaged estimations of daily MortonCRWE potential evapotranspiration or shallow lake evaporation.
`ET.AnnualAve`	A `zoo` object containing annually averaged estimations of daily MortonCRWE potential evapotranspiration or shallow lake evaporation.
`ET_formulation`	A character string containing the name of the formulation used which equals to "MortonCRWE".
`ET_type`	A character string containing the type of the estimation obtained which is either "Potential Evapotranspiration" or "Shallow Lake Evaporation".
`message1`	A message to inform the users about how solar radiation has been calculated by using which data.
`message6`	A message to inform the users about if actual dew point temperature has been used in the calculations or alternative calculations has been performed without dew point temperature data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

MORTON, F. I. 1986. Practical Estimates of Lake Evaporation. Journal of Applied Meteorology, 25, 371-388.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "MortonCRWE" to call function
funname <- "MortonCRWE"
class(data) <- funname

# Call generic function ET(data, ...) with class "MortonCRWE"
results <- ET(data, constants, est = "potential", solar =
"sunshine hours", Tdew = TRUE)
```

`ET.Penman` *Penman Formulation*

Description

Implementing the Penman formulation for estimating open-water evaporation or potential evapotranspiration

Usage

```
## S3 method for class 'Penman'
ET(data, constants, solar, wind, windfunction_ver,
alpha = 0.08, z0 = 0.001, ...)
```

Arguments

`data`	A list of data in class "Penman" which contains the following items (climate variables) required by Penman formulation: "Tmax", "Tmin", "Rs", "n" or "Cd", "u2" or "uz", "RHmax", "RHmin"
`constants`	A list named "constants" consists of constants required for the calculation of Penman formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1, "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1.

solar
: Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation data is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.

wind
: Must be either "yes" or "no". "yes" indicates that the calculation will use real data of wind speed; "no" indicates that the alternative calculation without using wind data will be used in Penman formulation (Valiantzas 2006, Equation33).

windfunction_ver
: The version of Penman wind function that will be used within the Penman formulation. Must be either "1948" or "1956". "1948" is for applying the Penman's 1948 wind function (Penman, 1948); "1956" is for applying the Penman's 1956 wind function (Penman, 1956)

alpha
: Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface. The default is 0.08 for open-water surface which is for the calculation of Penman open-water evaporation, all other values will trigger the calculation of Penman potential evapotranspriation.

z0
: Any value (metres), roughness height of the evaporative surface. The default is 0.08 for open-water surface which is for the calculation of Penman open-water evaporation, all other values will trigger the calculation of Penman potential evapotranspriation.

...
: Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through arguments `solar`, `wind` and `windfunction_ver`, please see "Arguments" for details. User-defined evaporative surface is allowed through arguments `alpha` and `z0`, please see "Arguments" for details.

Value

This function returns a object of class `Penman`. Such objects are list with components:

ET.Daily
: A `zoo` object containing daily aggregated estimations of Penman open-water evaporation or potential evapotranspiration.

ET.Monthly
: A `zoo` object containing monthly aggregated estimations of Penman open-water evaporation or potential evapotranspiration.

ET.Annual
: A `zoo` object containing annually aggregated estimations of Penman open-water evaporation or potential evapotranspiration.

ET.MonthlyAve
: A `zoo` object containing monthly averaged estimations of daily Penman open-water evaporation or potential evapotranspiration.

ET.AnnualAve
: A `zoo` object containing annually averaged estimations of daily Penman open-water evaporation or potential evapotranspiration.

ET_formulation	A character string containing the name of the formulation used which equals to "Penman".
ET_type	A character string containing the type of the estimation obtained which is either "Open-water Evaporation" or "Potential Evapotranspiration".
message1	A message to inform the users about how solar radiation has been calculated by using which data.
message2	A message to inform the users about if actual wind data has been used in the calculations or alternative calculations has been performed without wind data, and which version of the Penman wind function has been used.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

PENMAN, H. L. 1948. Natural evaporation from open water, bare soil and grass. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 193, 120-145.

VALIANTZAS, J. D. 2006. Simplified versions for the Penman evaporation equation using routine weather data. Journal of Hydrology, 331, 690-702.

PENMAN, H. L. 1956. Evaporation: An introductory survey. Netherlands Journal of Agricultural Science, 4, 9-29.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Penman" to call function
funname <- "Penman"
class(data) <- funname

# Call generic function ET(data, ...) with class "Penman"
results <- ET(data, constants, solar="sunshine hours", wind=
"yes", windfunction_ver = "1948", alpha = 0.08, z0 = 0.001)
```

ET.PenmanMonteith *Penman-Monteith Formulation*

Description

Implementing the Penman-Monteith formulation (including the method for FAO-56 hypothetical short grass and the method for ASCE-EWRI Standardised crop) for estimating reference crop evapotranspiration

Usage

```
## S3 method for class 'PenmanMonteith'
ET(data, constants, solar, wind, crop, ...)
```

Arguments

- `data` — A list of data in class "PenmanMonteith" which contains the following items (climate variables) required by Penman-Monteith formulation: "Tmax", "Tmin", "RHmax", "RHmin", "Rs, "n" or "Cd", "u2" or "uz"
- `constants` — A list named "constants" consists of constants required for the calculation of Penman-Monteith formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "G" - soil heat flux in MJ.m^-2.day^-1, = 0 when using daily time step.
- `solar` — Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation data is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.
- `wind` — Must be either "yes" or "no". "yes" indicates that the calculation will use real data of wind speed; "no" indicates that the alternative calculation without using wind data will be used in Penman formulation (Valiantzas 2006, Equation33).
- `crop` — Must be either "short" or "tall". "short" indicates that the method for FAO-56 hypothetical short grass will be applied (Allen et al., 1998, Equation 6); "tall" indicates that the method for ASCE-EWRI Standardised crop will be applied (ASCE, 2005, Equation 1, Table 1)
- `...` — Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through arguments `solar` and `wind`, please see "Arguments" for details. User-defined evaporative surface is allowed through arguments `crop`, please see "Arguments" for details.

Value

This function returns a object of class `PenmanMonteith`. Such objects are list with components:

- `ET.Daily` — A `zoo` object containing daily aggregated estimations of Penman-Monteith rerference crop evapotranspiration.

ET.Monthly	A zoo object containing monthly aggregated estimations of Penman-Monteith rerference crop evapotranspiration.
ET.Annual	A zoo object containing annually aggregated estimations of Penman-Monteith rerference crop evapotranspiration.
ET.MonthlyAve	A zoo object containing monthly averaged estimations of daily Penman-Monteith rerference crop evapotranspiration.
ET.AnnualAve	A zoo object containing annually averaged estimations of daily Penman-Monteith rerference crop evapotranspiration.
ET_formulation	A character string containing the name of the formulation used which equals to either "Penman-Monteith FAO56" or "Penman-Monteith ASCE-EWRI Standardised".
ET_type	A character string containing the type of the estimation obtained which is "Reference Crop Evapotranspiration".
message1	A message to inform the users about how solar radiation has been calculated by using which data.
message2	A message to inform the users about if actual wind data has been used in the calculations or alternative calculations has been performed without wind data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

ALLEN, R. G., PEREIRA, L. S., RAES, D. & SMITH, M. 1998. Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56. FAO, Rome, 300, 6541.

ALLEN, R. G. 2005. The ASCE standardized reference evapotranspiration equation, Amer Society of Civil Engineers.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "PenmanMonteith" to call function
funname <- "PenmanMonteith"
class(data) <- funname

# Call generic function ET(data, ...) with class "PenmanMonteith"
results <- ET(data, constants, solar="sunshine hours", wind="yes", crop="short")
```

ET.Penpan *Penpan Formulation*

Description

Implementing the Penpan formulation for Class-A pan evaporation.

Usage

```
## S3 method for class 'Penpan'
ET(data, constants, solar, alpha, overest, ...)
```

Arguments

`data`	A list of data in class "Penpan" which contains the following items (climate variables) required by Penpan formulation: "Tmax", "Tmin", "RHmax", "RHmin", "Rs", "n" or "Cd", "u2" or "uz"
`constants`	A list named "constants" consists of constants required for the calculation of Penpan formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "lat" - latitude in degrees, "alphaA" - albedo for Class-A pan, "ap" - a constant in Penpan = 2.4.
`solar`	Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation data is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.
`alpha`	Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.
`overest`	Must be TRUE or FALSE, indicating if adjustment for the overestimation (i.e. divided by 1.078) of Class-A pan evaporation for Australian data is applied in Penpan formulation.
`...`	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details. User-defined evaporative surface is allowed through argument `alpha`, please see "Arguments" for details. Adjustment for overestimation on the estimations are available through argument `height`, please see "Arguments" for details.

Value

This function returns a object of class Penpan. Such objects are list with components:

- `ET.Daily` A zoo object containing daily aggregated estimations of Penpan Class-A pan evaporation.
- `ET.Monthly` A zoo object containing monthly aggregated estimations of Penpan Class-A pan evaporation.
- `ET.Annual` A zoo object containing annually aggregated estimations of Penpan Class-A pan evaporation.
- `ET.MonthlyAve` A zoo object containing monthly averaged estimations of daily Penpan Class-A pan evaporation.
- `ET.AnnualAve` A zoo object containing annually averaged estimations of daily Penpan Class-A pan evaporation.
- `ET_formulation` A character string containing the name of the formulation used which equals to "Penpan".
- `ET_type` A character string containing the type of the estimation obtained which is "Class-A Pan Evaporation".
- `message1` A message to inform the users about how solar radiation has been calculated by using which data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

ROTSTAYN, L. D., RODERICK, M. L. & FARQUHAR, G. D. 2006. A simple pan-evaporation model for analysis of climate simulations: Evaluation over Australia. Geophysical Research Letters, 33.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Penpan" to call function
funname <- "Penpan"
class(data) <- funname

# Call generic function ET(data, ...) with class "Penpan"
results <- ET(data, constants, solar="sunshine hours", alpha = 0.23, overest = TRUE)
```

ET.PriestleyTaylor *Priestley-Taylor Formulation*

Description

Implementing the Priestley-Taylor formulation for potential evaporation

Usage

```
## S3 method for class 'PriestleyTaylor'
ET(data, constants, solar, alpha, ...)
```

Arguments

`data`	A list of data in class "PriestleyTaylor" which contains the following items (climate variables) required by Priestley-Taylor formulation: "Tmax", "Tmin", "RHmax", "RHmin", "RS", "n" or "Cd"
`constants`	A list named "constants" consists of constants required for the calculation of Priestley-Taylor formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1, "alphaPT" - Priestley-Taylor coefficient, "G" - soil heat flux in MJ.m^-2.day^-1, = 0 when using daily time step.
`solar`	Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation data is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.
`alpha`	Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.
`...`	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument `solar`, please see "Arguments" for details. User-defined evaporative surface is allowed through argument `alpha`, please see "Arguments" for details.

Value

This function returns a object of class PriestleyTaylor. Such objects are list with components:

- **ET.Daily** A zoo object containing daily aggregated estimations of Priestley-Taylor potential evaporation.
- **ET.Monthly** A zoo object containing monthly aggregated estimations of Priestley-Taylor potential evaporation.
- **ET.Annual** A zoo object containing annually aggregated estimations of Priestley-Taylor potential evaporation.
- **ET.MonthlyAve** A zoo object containing monthly averaged estimations of daily Priestley-Taylor potential evaporation.
- **ET.AnnualAve** A zoo object containing annually averaged estimations of daily Priestley-Taylor potential evaporation.
- **ET_formulation** A character string containing the name of the formulation used which equals to "Priestley-Taylor".
- **ET_type** A character string containing the type of the estimation obtained which is "Potential Evaporation".
- **message1** A message to inform the users about how solar radiation has been calculated by using which data.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "PriestleyTaylor" to call function
funname <- "PriestleyTaylor"
class(data) <- funname

# Call generic function ET(data, ...) with class "PriestleyTaylor"
results <- ET(data, constants, solar="sunshine hours", alpha = 0.08)
```

`ET.SzilagyiJozsa` *Szilagyi-Jozsa Formulation*

Description

Implementing the Szilagyi-Jozsa formulation for estimating actual evapotranspiration

Usage

```
## S3 method for class 'SzilagyiJozsa'
ET(data, constants, solar, wind,
windfunction_ver, alpha, z0, ...)
```

Arguments

`data` A list of data in class "SzilagyiJozsa" which contains the following items (climate variables) required by Szilagyi-Jozsa formulation: "Tmax", "Tmin", "RHmax", "RHmin", "Rs", "n" or "Cd", "u2" or "uz"

`constants` A list named "constants" consists of constants required for the calculation of Szilagyi-Jozsa formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "z" - height of wind instrument in m, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1.

`solar` Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.

`wind` Must be either "yes" or "no". "yes" indicates that the calculation will use real data of wind speed for calculating Penman potential evapotranspiration, which is required for this formulation; "no" indicates that the alternative calculation without using wind data will be used in Penman formulation (Valiantzas 2006, Equation33).

`windfunction_ver` The version of Penman wind function that will be used within the Penman formulation. Must be either "1948" or "1956". "1948" is for applying the Penman's 1948 wind function (Penman, 1948); "1956" is for applying the Penman's 1956 wind function (Penman, 1956)

`alpha` Any numeric value between 0 and 1 (dimensionless), albedo of evaporative surface (or surrounding surface for Penpan formulation) representing the portion of the incident radiation that is reflected back at the surface.

z0	Any value (metres), roughness height of the evaporative surface.
...	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through arguments solar, wind and windfunction_ver, please see "Arguments" for details. User-defined evaporative surface is allowed through arguments alpha and z0, please see "Arguments" for details.

Value

This function returns a object of class SzilagyiJozsa. Such objects are list with components:

ET.Daily	A zoo object containing daily aggregated estimations of Szilagyi-Jozsa actual evapotranspiration.
ET.Monthly	A zoo object containing monthly aggregated estimations of Szilagyi-Jozsa actual evapotranspiration.
ET.Annual	A zoo object containing annually aggregated estimations of Szilagyi-Jozsa actual evapotranspiration.
ET.MonthlyAve	A zoo object containing monthly averaged estimations of daily Szilagyi-Jozsa actual evapotranspiration.
ET.AnnualAve	A zoo object containing annually averaged estimations of daily Szilagyi-Jozsa actual evapotranspiration.
ET_formulation	A character string containing the name of the formulation used which equals to "Szilagyi-Jozsa".
ET_type	A character string containing the type of the estimation obtained which is "Actual Evapotranspiration".
message1	A message to inform the users about how solar radiation has been calculated by using which data.
message2	A message to inform the users about if actual wind data has been used in the calculations or alternative calculations has been performed without wind data, and which version of the Penman wind function has been used.

Author(s)

Danlu Guo

References

SZILAGYI, J. 2007. On the inherent asymmetric nature of the complementary relationship of evaporation. Geophysical Research Letters, 34, L02405.

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

PENMAN, H. L. 1948. Natural evaporation from open water, bare soil and grass. Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 193, 120-145.

VALIANTZAS, J. D. 2006. Simplified versions for the Penman evaporation equation using routine weather data. Journal of Hydrology, 331, 690-702.

PENMAN, H. L. 1956. Evaporation: An introductory survey. Netherlands Journal of Agricultural Science, 4, 9-29.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "SzilagyiJozsa" to call function
funname <- "SzilagyiJozsa"
class(data) <- funname

# Call generic function ET(data, ...) with class "SzilagyiJozsa"
results <- ET(data, constants, solar="sunshine hours", wind =
"yes", windfunction_ver = "1948", alpha = 0.23, z0 = 0.02)
```

ET.Truc *Truc Formulation*

Description

Implementing the Truc formulation for estimating reference crop evapotranspiration.

Usage

```
## S3 method for class 'Truc'
ET(data, constants, solar, humid, ...)
```

Arguments

`data`	A list of data in class "Truc" which contains the following items (climate variables) required by Truc formulation: "Tmax", "Tmin", "Rs", "n" or "Cd"
`constants`	A list named "constants" consists of constants required for the calculation of Truc formulation which must contain the following items: "Elev" - ground elevation above mean sea level in m, "lambda" - latent heat of vaporisation = 2.45 MJ.kg^-1, "lat_rad" - latitude in radians, "Gsc" - solar constant = 0.0820 MJ.m^-2.min^-1 "as" - fraction of extraterrestrial radiation reaching earth on sunless days, "bs" - difference between fracion of extraterrestrial radiation reaching full-sun days and that on sunless days, "sigma" - Stefan-Boltzmann constant = 4.903*10^-9 MJ.K^-4.m^-2.day^-1.
`solar`	Must be either "data", "sunshine hours", "cloud" or "monthly precipitation". "data" indicates that solar radiation is to be used directly for calculating evapotranspiration; "sunshine hours" indicates that solar radiation is to be calculated using the real data of sunshine hours; "cloud" sunshine hours is to be estimated

	from cloud data; "monthly precipitation" indicates that solar radiation is to be calculated directly from monthly precipitation.
humid	Must be TRUE or FALSE, indicating if adjustment for non-humid conditions is applied in Truc formulation (Alexandris et al., 2008, Equation 5b).
...	Dummy for generic function, no need to define.

Details

The alternative calculation options can be selected through argument solar, please see "Arguments" for details. Humidity adjustment for the estimations is available through argument humid, please see "Arguments" for details.

Value

This function returns a object of class Truc. Such objects are list with components:

ET.Daily	A zoo object containing daily aggregated estimations of Truc reference crop evapotranspiration.
ET.Monthly	A zoo object containing monthly aggregated estimations of Truc reference crop evapotranspiration.
ET.Annual	A zoo object containing annually aggregated estimations of Truc reference crop evapotranspiration.
ET.MonthlyAve	A zoo object containing monthly averaged estimations of daily Truc reference crop evapotranspiration.
ET.AnnualAve	A zoo object containing annually averaged estimations of daily Truc reference crop evapotranspiration.
ET_formulation	A character string containing the name of the formulation used which equals to "Truc".
ET_type	A character string containing the type of the estimation obtained which is "Reference Crop Evapotranspiration".
message1	A message to inform the users about how solar radiation has been calculated by using which data.
message4	A message to inform the users about if adjustment for non-humid conditions has been applied to calculated Truc reference crop evapotranspiration.

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

TURC, L. 1961. Estimation of irrigation water requirements, potential evapotranspiration: a simple climatic formula evolved up to date. Ann. Agron, 12, 13-49.

ALEXANDRIS, S., STRICEVIC, R. & PETKOVIC, S. 2008. Comparative analysis of reference evapotranspiration from the surface of rainfed grass in central Serbia, calculated by six empirical methods against the Penman-Monteith formula. Eur Water, 21, 17-28.

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Truc" to call function
funname <- "Truc"
class(data) <- funname

# Call generic function ET(data, ...) with class "Truc"
results <- ET(data, constants, solar="sunshine hours", humid = TRUE)
```

`ETComparison` *Plot esimtated evapotranspiration with climate variables*

Description

Produce comparison plots for results and statistics from different estimations produced by using different formulations and/or different input data. The number of different sets of results can be between 2 and 7. Plotting type can be selected among daily aggregation, monthly aggregation, annual aggregation, monthly average and annual average. For each type three comparison plots will be produced including time series, non-exceedance probability and box plot.

Usage

```
ETComparison(results1, results2, results3 = NULL, results4 = NULL, results5 = NULL,
results6 = NULL, results7 = NULL, labs, Sdate=NULL, Edate=NULL, type = "Monthly", ylim)
```

Arguments

`results1`	A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor.
`results2`	A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor.
`results3`	A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor. The default is NULL if the user requires the comparison between only two sets of results.
`results4`	A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor. The default is NULL if the user requires the comparison among only three sets of results.

results5
: A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor. The default is NULL if the user requires the comparison among only four sets of results.

results6
: A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor. The default is NULL if the user requires the comparison among only five sets of results.

results7
: A list named "results" which has been derived from function `ET..()` which can be of any class such as "Penman", "PenmanMonteith" and "PriestleyTaylor. The default is NULL if the user requires the comparison among only six sets of results.

labs
: A character vector with the length equal to the number of sets of results to compare, defining the labels for the comparison plots

Sdate
: Only used when type = 'Daily', 'Monthly' and 'Annual' to define the start date for the plotting windows, the default is the first day for the estimate evapotranspiration, but can be defined by user in the format "YYYY-MM-DD".

Edate
: Only used when type = 'Daily', 'Monthly' and 'Annual' to define the end date for the plotting windows, the default is the last day for the estimate evapotranspiration, but can be defined by user in the format "YYYY-MM-DD".

ylim
: A numeric vector of length 2 defining the lower and upper limit of the y-axis for plotting, the default is from 0 to 1.5 times of maximum value from the first set of result that is used to compare with others.

type
: A character string indicating the type of plot produced, can be one of the following: "Daily" - comparison plots of estimated daily evapotranspiration; "Monthly" - comparison plots of monthly aggregated evapotranspiration; "Annual" - comparison plots of annually aggregated evapotranspiration; "MonthlyAve" - comparison plots of monthly averaged daily evapotranspiration; "AnnualAve" - comparison plots of annually averaged daily evapotranspiration.

Value

Three plots are generated for each type of comparison plot selected, including: 1) time series plot of the estimated/aggregated/averaged values from each set of result; 2) non-exceedance plot of the distribution of estimated/aggregated/averaged values from each set of result; 3) box plot of the distribution of estimated/aggregated/averaged values from each set of result.

Author(s)

Danlu Guo

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")
```

```
# Set data class to be "Penman" to call function
funname <- "Penman"
class(data) <- funname

# Call generic function ET(data, constants, ...) with class "Penman"
results_Penman <- ET(data, constants, solar="sunshine hours", wind=
"yes", windfunction_ver = "1948", alpha = 0.26, z0 = 0.02)

# Set data class to be "PenmanMonteith" to call function
funname <- "PenmanMonteith"
class(data) <- funname

# Call generic function ET(data, constants, ...) with class "PenmanMonteith"
results_PenmanMonteith <- ET(data, constants, solar="sunshine hours", wind="yes", crop="short")

# Plot the estimated Penman open-water evaporation against average temperature
ETComparison(results_Penman, results_PenmanMonteith, type = "Monthly",
labs = c("Penman", "Penman-Monteith"))
```

ETForcings *Plot esimtated evapotranspiration with climate variables*

Description

Produce plot of daily, monthly and annual averaged estimated evapotranspiration with selected climate variables of the same time step.

Usage

```
ETForcings(data, results, forcing)
```

Arguments

`data`	A list of data named "data" which must contain a component with the name of a climate variable that the estimated evapotranspiration should be plotted against, see `forcing`.
`results`	A list named "results" which has been derived from function `ET..()`.
`forcing`	A character string as the name of a climate variable that the estimated evapotranspiration should be plotted against, can be any of: "Tmax" - maximum temperature, "Tmin" - minimum temperature, "u2" - average wind speed at 2m, "uz" - average wind speed, "Rs" - solar radiation, "n" - daily sunshine hours, "Precip" - precipitation, "Epan" - Class-A pan evaporation, "RHmax" - maximum relative humidity, "RHmin" - minimum relative humidity, "Tdew" - average dew point temeprature.

Value

Three plots are generated for the response of calculated evapotranspiration to each climate variable, including: 1) daily evapotranspiration estimate vs. daily average temperature; 2) monthly mean daily evaporationion estimate vs. monthly average temperature; 3) annual mean daily evaporationion estimate vs. annual average temperature.

Author(s)

Danlu Guo

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Penman" to call function
funname <- "Penman"
class(data) <- funname

# Call generic function ET(data, constants, ...) with class "Penman"
results <- ET(data, constants, solar="sunshine hours", wind=
"yes", windfunction_ver = "1948", alpha = 0.08, z0 = 0.001)
# Plot the estimated Penman open-water evaporation against average temperature
ETForcings(data, results, forcing = "Tmax")
```

ETPlot — *Plot the daily, monthly and annual aggregations of esimtated evapotranspiration*

Description

Produce plot of aggregated estimations of evapotranspiration in daily, monthly and annual steps, or averaged daily estimations in monthly or annual steps.

Usage

```
ETPlot(results, type = "Aggregation", OBS, OBSplot = FALSE,
Sdate = time(results$ET.Daily)[1],
Edate = time(results$ET.Daily)[length(results$ET.Daily)])
```

Arguments

`results`	A list named "results" which has been derived from function `ET..()`.
`type`	A character string of either "Aggregation" or "Average" to indicate the type of plot required. The default is "aggregation". For aggregation plot the user can define the start and end date of plotting or by default using the calculation period for plotting. For average plot the plotting period equals to the calculation period.

`OBS` A list named "OBS" which has been derived from function `ReadOBSEvaporation()`.

`OBSplot` Must be eith TRUE or FALSE. TRUE indicates that the observed evaporation will be plotted together with the estimations and FALSE indicates that the observations will not be shown on the plots.

`Sdate` Only used when type = "Aggregation" to define the start date for the plotting windows, the default is the first day for the estimate evapotranspiration, but can be defined by user in the format "YYYY-MM-DD".

`Edate` Only used when type = "Aggregation" to define the end date for the plotting windows, the default is the last day for the estimate evapotranspiration, but can be defined by user in the format "YYYY-MM-DD".

Value

If type = "Aggregation", three plots are displayed in the following order (the next one appears after pressing enter): 1) Daily evapotranspiration estimates; 2) Monthly evapotranspiration estimates aggregated from daily estimates; 3) Annual evapotranspiration estimates aggregated from daily estimates.

If type = "Average", two plots are displayed in the following order 1) Monthly averaged daily estimations of evapotranspiration; 2) Annually averaged daily estiamtions of evapotranspriation.

Author(s)

Danlu Guo

Examples

```
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Set data class to be "Penman" to call function
funname <- "Penman"
class(data) <- funname

# Call generic function Evapotranspiration(data, constants, ...) with class "Penman"
results <- ET(data, constants, solar="sunshine hours", wind=
"yes", windfunction_ver = "1948", alpha = 0.08, z0 = 0.001)

# Read evaporation data
data("E_OBS")
OBS <- ReadOBSEvaporation(E_OBS, data)

# Plot the aggregation of estimated Penman open-water evaporation with observed evaporation
ETPlot(results, type = "Aggregation", OBS, OBSplot = TRUE, Sdate = "2001-05-01",
Edate = "2004-05-01")
```

E_OBS *Observed Class-A Pan Evaporation*

Description

This data set contains the Class-A pan evaporation observed over the period between 1/3/2001 and 31/8/2004 at the Kent Town station in Adelaide, Australia.

Usage

```
climatedata
```

Format

A list containing 48 obserations of 5 variables

Source

Bureau of Meteorology, Kent Town, Adelaide, Australia

ReadInputs *ReadInputs raw date and climate data*

Description

Load raw date and climate data, perform pre-processing, check for missing and error entries and then compile data list of daily time step.

Usage

```
ReadInputs(climatedata, constants, stopmissing)
```

Arguments

`climatedata` A data frame named "climatedata" containing the raw data of date and climate variables. The data frame contain objects named as "Year", "Month" and "Day" to indicate the date. The climate variables will be of the following names while it is not compulsory to have all of them:

"Tmax.daily" - daily maximum temperature in degree Celcius, "Tmin.daily" - daily minimum temperature in degree Celcius, "Temp.subdaily" - subdaily temperature in degree Celcius, "Tdew.subdaily" - subdaily dew point temperature in degree Celcius, "RHmax.daily" - daily maximum relative humidity in percentage, "RHmin.daily" - daily minimum relative humidity in percentage, "RH.subdaily" - subdaily relative humidity in degree Celcius, "Rs.daily" - daily incoming solar radiation in Millijoules per square metres per day, "n.daily" -

daily sunshine hour in hours, "Cd.daily" - daily cloud cover in oktas, "Precip.daily" - daily precipitation in millimitres, "u2.subdaily" - subdaily wind speed measured at 2 metres from the ground surface in kilometres per hour, "uz.subdaily" - subdaily wind speed in kilometres per hour, "Epan.daily" - daily Class-A pan evaporation in millimitres, "Vp.subdaily" - subdaily vapour pressure in hectopascal.

In order to determine which variables to include in "climatedata", please see `Evapotranspiration` for the specific data requirements for different formulations.

`constants` A list named "constants" consists of constants required for data pre-processing which may contain the following items: "a_0", "b_0", "c_0", "d_0". These four constants which are constants required to calculate daily sunshine hours from daily cloud cover (Chiew and McMahon, 1991, described as a0, b0, c0, d0 in Equation S3.10 in McMahon et al., 2012), if such calculation will be performed these constants must be included in "constants".

`stopmissing` A numeric vector of length 2. The first value represents the maximum percentage of missing data that the user can tolerate; the second value represents the maximum percentage of the duration of missing data to the total data duaration that the user can toleratre. Both values should be numbers between 1 and 99. The percentages of the number and duration of missing data in each input variable are compared to the corresponding threshold, below which the missing data will be interpolated and otherwise the program will be terminated due to unsatisfactory data quality.

Value

This function returns a list with all components of class `zoo` which have been processed from the raw data, including:

`Date.daily` A `zoo` object containing the date in daily step in the format of yyyy-mm-dd.

`Date.monthly` A `zoo` object containing the date in daily step in the format of mmm-yyyy.

`J` A `zoo` object containing the Julian Day for every day during the period that the data spans.

`i` A `zoo` object containing the month number for every day during the period that the data spans.

`ndays` A `zoo` object containing the number of days for every month during the period that the data spans.

`Tmax` A `zoo` object containing the daily maximum temperatures in degree Celcius.

`Tmin` A `zoo` object containing the daily minimum temperatures in degree Celcius.

`u2` A `zoo` object containing the daily wind speed at 2m from the ground in m/s.

`uz` A `zoo` object containing the daily wind speed measured at the height of wind instrument in m/s.

`Rs` A `zoo` object containing the daily solar radiation in MJ/m^2/day.

`n` A `zoo` object containing the daily sunshine hours.

`Cd` A `zoo` object containing the daily cloud cover in oktas.

Precip	A zoo object containing the daily precipitation in mm.
Epan	A zoo object containing the daily Class-A pan evaporation in mm.
RHmax	A zoo object containing the daily maximum relative humidity in percentage.
RHmin	A zoo object containing the daily minimum relative humidity in percentage.
Tdew	A zoo object containing the average daily dew temperatures in degree Celcius.

Note that the components might have value of NULL when the corresponding raw data cannot be found in "climatedata".

Author(s)

Danlu Guo

References

MCMAHON, T., PEEL, M., LOWE, L., SRIKANTHAN, R. & MCVICAR, T. 2012. Estimating actual, potential, reference crop and pan evaporation using standard meteorological data: a pragmatic synthesis. Hydrology and Earth System Sciences Discussions, 9, 11829-11910.

CHIEW, F. H. & MCMAHON, T. A. 1991. THE APPLICABILITY OF MORTON'S AND PENMAN'S EVAPOTRANSPIRATION ESTIMATES IN RAINFALL-RUNOFF MODELING1. JAWRA Journal of the American Water Resources Association, 27, 611-620.

See Also

evapotranspiration

Examples

```
# ReadInputs climate data
data("climatedata")
data("constants")
data <- ReadInputs(climatedata, constants, stopmissing = c(10,3))
```

ReadOBSEvaporations *Read Raw Data of Observed Evaporation from File*

Description

Load raw date and evaporation data and then compile data list of daily time step.

Usage

```
ReadOBSEvaporation(E_OBS, data)
```

Arguments

E_OBS — A list of evaporation data named "E_OBS" which must contain the following columns: "Year", "Month", "Day" as the date and, "EVAP.Obs" as the observed evaporation in mm. The observations can be of daily and monthly time steps and must match with the corresponding dates recorded.

data — A list of data named "data" which contains data of climate variables over the same period as the evaporation data

Value

This function returns a list with all components of class zoo which have been processed from the raw data, including:

Date.OBS — A zoo object containing the date data with time step consistent with the raw evaporation data in "E_OBS".

E_obs.Daily — A zoo object containing the daily evaporation data.

E_obs.Monthly — A zoo object containing the monthly aggregaated observed evaporation in mm.

E_obs.Annual — A zoo object containing the annually aggregated observed evaporation in mm.

E_obs.MonthlyAve — A zoo object containing the monthly averaged daily evaporation from observation in mm/day.

E_obs.AnnualAve — A zoo object containing the annually average daily evaporation from observation in mm/day.

Note that the components might have value of NULL when the corresponding raw data cannot be found in "E_OBS".

Author(s)

Danlu Guo

See Also

evapotranspiration

Examples

```
# Get the time period from "data"
# Use processed existing data set from kent Town, Adelaide
data("processeddata")
data("constants")

# Reading obsevations of evaporation within specified time period
data("E_OBS")
OBS <- ReadOBSEvaporation(E_OBS, data)
```

Index