

# Package ‘SPEI’

January 20, 2025

**Type** Package

**Version** 1.8.1

**Title** Calculation of the Standardized Precipitation-Evapotranspiration Index

**Description**

A set of functions for computing potential evapotranspiration and several widely used drought indices including the Standardized Precipitation-Evapotranspiration Index (SPEI).

**Depends** R (>= 3.5.0)

**Imports** lmomco, lmom, TLMoments, reshape, ggplot2, checkmate, zoo, lubridate

**License** GPL-2

**URL** <https://spei.csic.es>, <https://github.com/sbegueria/SPEI>

**Language** en-US

**BugReports** <https://github.com/sbegueria/SPEI/issues>

**LazyLoad** true

**Encoding** UTF-8

**Suggests** covr, testthat

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Santiago Beguería [aut, cre],  
Sergio M. Vicente-Serrano [aut]

**Maintainer** Santiago Beguería <santiago.begueria@csic.es>

**Repository** CRAN

**Date/Publication** 2023-03-02 13:00:05 UTC

## Contents

Datasets . . . . .	2
Drought-indices . . . . .	4

Generalized-Logistic . . . . .	10
Generic-methods-for-spei-objects . . . . .	11
Kernel-functions . . . . .	12
Potential-evapotranspiration . . . . .	13
SPEI . . . . .	19
SPEINews . . . . .	20

<b>Index</b>	<b>21</b>
--------------	-----------

---

Datasets	<i>Data sets for illustrating the functions in the SPEI package.</i>
----------	--

---

### Description

Data used in the examples of the SPEI package: wichita dataset: monthly climate in Wichita (Kansas, lat=37.6475, elevation=402.6 m. a.s.l.) since January 1980; balance dataset: monthly climatic water balance (precipitation minus potential evapotranspiration) at eleven locations around the World, since January 1900; cabinda: one year of data for computing Penman-Monteith ETO from Allen et al. (1998); cruts4: 120 years of monthly climatic water balance (precipitation minus reference evapotranspiration) data at six grid points from CRU TS 4.05.

### Format

wichita dataset: a data frame with:

**YEAR** monthly precipitation totals, in mm.

**MONTH** monthly precipitation totals, in mm.

**PRCP** monthly precipitation totals, in mm.

**TMAX** monthly mean daily maximum temperature, in °C.

**TMIN** monthly mean daily minimum temperature, in °C.

**TMED** monthly mean temperature, in °C.

**AWND** monthly mean wind speed, in km h<sup>-1</sup>

**ACSH** monthly mean sun hours, in h.

**ACSH** monthly mean cloud cover, in %.

balance dataset: a data frame with monthly climatic water balance (precipitation minus potential evapotranspiration) at Indore (India), Kimberley (South Africa), Albuquerque (US), Valencia (Spain), Wien (Austria), Abashiri (Japan), Tampa (US), Sao Paulo (Brazil), Lahore (India), Punta Arenas (Chile) and Helsinki (Finland), in mm. cabinda dataset: a data frame with one year of monthly climatic data at Cabinda (Angola, -5.33S 12.11E 20 m), with:

**mon** month of the year

**Tmin** monthly mean daily minimum temperature, in °C.

**Tmax** monthly mean daily maximum temperature, in °C.

**RH** monthly mean relative humidity, in %.

**U2** monthly mean wind speed, in km h<sup>-1</sup>

**tsun** monthly mean sunshine hours, in h.

**Rs** monthly mean daily incoming solar radiation, MJ m<sup>-2</sup> d<sup>-1</sup>.

**ET0** monthly ET0 from the original publication, in mm.

cruts4 dataset: an array with 120 years of monthly climatic water balance (precipitation minus reference evapotranspiration) data at six grid points from CRU TS 4.05 data set. The array has dimensions [time=1440, longitude=2, latitude=3], with time starting in January 1900. Longitudes are (0.25, 0.75), and latitudes (42.25, 42.75, 43.25), corresponding to the Central Pyrenees between Spain and France.

### Details

See description.

### Author(s)

Data ported to R by S. Beguería.

### Source

The wichita data were obtained from the Global Historical Climatology Network (GHCN, <http://www.ncei.noaa.gov/>). Data for the balance dataset were extracted from CRU TS V3.1 and from the 20th Century Reanalysis V2 data set. Data for the balance dataset were taken from Allen et al. (1998), page 69, figure 18. The cruts4 data were obtained from the CRU (Climatic Research Unit, University of East Anglia <https://crudata.uea.ac.uk/>) TS V4.05 data set.

### References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

R.G. Allen, L.S. Pereira, D. Raes, M. Smith. 1998. *ICrop evapotranspiration - Guidelines for computing crop water requirements - FAO Irrigation and drainage paper 56*. FAO, Rome. ISBN 92-5-104219-5.

### Examples

```
data(wichita)
names(wichita)
summary(wichita)
data(balance)
summary(balance)
data(cruts4)
summary(cruts4)
```

---

Drought-indices

*Calculation of the Standardized Precipitation-Evapotranspiration Index (SPEI) and the Standardized Precipitation Index (SPI).*


---

### Description

Given a time series of the climatic water balance (precipitation minus potential evapotranspiration), gives a time series of the Standardized Precipitation-Evapotranspiration Index (SPEI).

### Usage

```
spei(
  data,
  scale,
  kernel = list(type = 'rectangular', shift = 0),
  distribution = 'log-Logistic',
  fit = 'ub-pwm',
  na.rm = FALSE,
  ref.start=NULL,
  ref.end=NULL,
  keep.x=FALSE,
  params=NULL,
  verbose=TRUE,
  ...)
```

```
spi(
  data,
  scale,
  kernel = list(type = 'rectangular', shift = 0),
  distribution = 'Gamma',
  fit = 'ub-pwm',
  na.rm = FALSE,
  ref.start=NULL,
  ref.end=NULL,
  keep.x=FALSE,
  params=NULL,
  verbose=TRUE,
  ...)
```

```
spi(
  data,
  scale,
  kernel = list(type = "rectangular", shift = 0),
  distribution = "Gamma",
  fit = "ub-pwm",
  na.rm = FALSE,
  ref.start = NULL,
```

```

    ref.end = NULL,
    keep.x = FALSE,
    params = NULL,
    verbose = TRUE,
    ...
)

```

### Arguments

<code>data</code>	a vector, matrix or data frame with time ordered values of precipitation (for the SPI) or of the climatic balance precipitation minus potential evapotranspiration (for the SPEI).
<code>scale</code>	an integer, representing the time scale at which the SPEI / SPI will be computed.
<code>kernel</code>	optional, a list defining the type of kernel used for computing the SPEI / SPI at scales higher than one. Defaults to unshifted rectangular kernel.
<code>distribution</code>	optional, name of the distribution function to be used for computing the SPEI / SPI (one of 'log-Logistic', 'Gamma' and 'PearsonIII'). Defaults to 'log-Logistic' for <code>spei</code> , and to 'Gamma' for <code>spi</code> .
<code>fit</code>	optional, name of the method used for computing the distribution function parameters (one of 'ub-pwm', 'pp-pwm' and 'max-lik'). Defaults to 'ub-pwm'.
<code>na.rm</code>	optional, a logical value indicating whether NA values should be stripped from the computations. Defaults to FALSE, i.e. no NA are allowed in the data.
<code>ref.start</code>	optional, starting point of the reference period used for computing the index. Defaults to NULL, indicating that the first value in data will be used as starting point.
<code>ref.end</code>	optional, ending point of the reference period used for computing the index. Defaults to NULL, indicating that the last value in data will be used as ending point.
<code>keep.x</code>	optional, a logical value indicating whether the data used for fitting the model should be kept. Defaults to FALSE.
<code>params</code>	optional, an array of parameters for computing the <code>spei</code> . This option overrides computation of fitting parameters.
<code>verbose</code>	optional, logical, report the computation options during calculation. Either 'TRUE' (default) or 'FALSE'.
<code>...</code>	other possible parameters.

### Details

The `spei` and `spi` functions allow computing the SPEI and the SPI indices. These are climatic proxies widely used for drought quantification and monitoring. Both functions are identical (in fact, `spi` is just a wrapper for `spei`), but they are kept separated for clarity. Basically, the functions standardize a variable following a log-Logistic (or Gamma, or PearsonIII) distribution function (i.e., they transform it to a standard Gaussian variate with zero mean and standard deviation of one).

## Value

Functions `spei` and `spi` return an object of class `spei`. The generic functions `print` and `summary` can be used to obtain summaries of the results. The generic accessor functions `coefficients` and `fitted` extract useful features of the object.

An object of class `spei` is a list containing at least the following components:

- `call`: the call to `spei` or `spi` used to generate the object.
- `fitted`: time series with the values of the Standardized Precipitation-Evapotranspiration Index (SPEI) or the Standardized Precipitation Index (SPI). If data consists of several columns the function will treat each column as independent data, and the result will be a multivariate time series. The names of the columns in data will be used as column names in `fitted`.
- `coefficients`: an array with the values of the coefficients of the distribution function fitted to the data. The first dimension of the array contains the three (or two) coefficients, the second dimension will typically consist of twelve values corresponding to each month, and the third dimension will be equal to the number of columns (series) in data. If a time scale greater than one has been used then the first elements will have NA value since the kernel can not be applied. The first element with valid data will be the one corresponding to the time scale chosen.
- `scale`: the `scale` parameter used to generate the object.
- `kernel`: the parameters and values of the kernel used to generate the object.
- `distribution`: the distribution function used to generate the object.
- `fit`: the fitting method used to generate the object.
- `na.action`: the value of the `na.action` parameter used.
- `data`: if requested, the input data used.

## Input data

The input variable is a time ordered series of precipitation values for `spi`, or a series of the climatic water balance (precipitation minus potential evapotranspiration) for `spei`. When used with the default options, it would yield values of both indices exactly as defined in the references given below.

The SPEI and the SPI were defined for monthly data. Since the PDFs of the data are not homogeneous from month to month, the data is split into twelve series (one for each month) and independent PDFs are fit to each series. If data is a vector or a matrix it will be treated as a sequence of monthly values starting in January. If it is a (univariate or multivariate) time series then the function `cycle` will be used to determine the position of each observation within the year (month), allowing the data to start in a month other than January.

## Time scales

An important advantage of the SPEI and the SPI is that they can be computed at different time scales. This way it is possible to incorporate the influence of the past values of the variable in the computation enabling the index to adapt to the memory of the system under study. The magnitude of this memory is controlled by parameter `scale`. For example, a value of six would imply that data from the current month and of the past five months will be used for computing the SPEI or SPI

value for a given month. By default all past data will have the same weight in computing the index, as it was originally proposed in the references below. Other kernels, however, are available through parameter `kernel`. The parameter `kernel` is a list defining the shape of the kernel and a time shift. These parameters are then passed to the function `kern`.

### Probability distributions

Following the original definitions `spei` uses a log-Logistic distribution by default, and `spi` uses a Gamma distribution. This behavior can be modified, however, through parameter `distribution`.

### Fitting methods

The default method for parameter fitting is based on unbiased Probability Weighted Moments (`'ub-pwm'`), but other methods can be used through parameter `fit`. A valid alternative is the plotting-position PWM (`'pp-pwm'`) method. For the log-Logistic distribution, also the maximum likelihood method (`'max-lik'`) is available.

### User-provided parameters

An option exists to override parameter fitting and provide user default parameters. This is activated with the parameter `params`. The exact values provided to this parameter depend on the distribution function being used. For log-Logistic and PearsonIII it should be a three-dimensional array with dimensions (3,number of series in data,12), containing twelve parameter triads ( $\xi$ ,  $\alpha$ ,  $\kappa$ ) for each data series, one for each month. For Gamma, a three-dimensional array with dimensions (2,number of series in data,12), containing twelve parameter pairs ( $\alpha$ ,  $\beta$ ). It is a good idea to look at the coefficients slot of a previously fit `spei` `spei` object in order to understand the structure of the parameter array. The parameter `distribution` is still used under this option in order to know what distribution function should be used.

### Reference period

The default behavior of the functions is using all the values provided in `data` for parameter fitting. However, this can be modified with help of parameters `ref.start` and `ref.end`. These parameters allow defining a subset of values that will be used for parameter fitting, i.e. a reference period. The functions, however, will compute the values of the indices for the whole data set. For these options to work it is necessary that `data` will be a time series object. The starting and ending points of the reference period will then be defined as pairs of year and month values, e.g. `c(1900,1)`.

### Processing large datasets

It is possible to use the `spei` and `spi` functions for processing multivariate datasets at once. If a matrix or data frame is supplied as `data`, with time series of precipitation or precipitation minus potential evapotranspiration arranged in columns, the result would be a matrix (data frame) of `spi` or `spei` series. This makes processing large datasets extremely easy, since no loops need to be used.

### Author(s)

Santiago Beguería and Sergio M. Vicente-Serrano. Maintainer: Santiago Beguería.

## References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

S. Beguería, S.M Vicente-Serrano, F. Reig, B. Latorre. 2014. Standardized precipitation evapotranspiration index (SPEI) revisited: parameter fitting, evapotranspiration models, tools, datasets and drought monitoring. *International Journal of Climatology* **34**(10): 3001-3023.

<http://spei.csic.es>

## See Also

`kern` for different kernel functions available. `thornthwaite`, `hargreaves` and `penman` for ways of calculating potential evapotranspiration. `summary.spei` and `print.spei` for summaries of `spei` objects. `plot.spei` for plotting `spei` objects.

## Examples

```
# Load data
data(wichita)

# Compute potential evapotranspiration (PET) and climatic water
# balance (BAL).
wichita$PET <- thornthwaite(wichita$TMED, 37.6475)
wichita$BAL <- wichita$PRCP - wichita$PET

# Convert to a ts (time series) for convenience
wichita <- ts(wichita[, -c(1, 2)], end = c(2011, 10), frequency = 12)
plot(wichita)

# One and twelve-months SPEI
spei1 <- spei(wichita[, "BAL"], 1)
spei12 <- spei(wichita[, "BAL"], 12)
class(spei1)
plot(spei1)
plot(spei12)

# Extract information from `spei` object: summary, call function,
# fitted values, and coefficients
summary(spei1)
names(spei1)
spei1$call
spei1$fitted
spei1$coefficients

# Plot `spei` object
par(mfrow = c(2, 1))
plot(spei1, main = "Wichita, SPEI-1")
plot(spei12, main = "Wichita, SPEI-12")

# One and twelve-months SPI
spi_1 <- spi(wichita[, "PRCP"], 1)
```



```

spi_12 <- spi(wichita[, "PRCP"], 12)

par(mfrow = c(2, 1))
plot(spi_1, "Wichita, SPI-1")
plot(spi_12, "Wichita, SPI-12")

# Time series not starting in January
plot(spei(ts(wichita[, "BAL"], freq = 12, start = c(1980, 6)), 12))

# Using a particular reference period (1980-2000) for computing the
# parameters. This may result in unexpected values (Inf, NaN) if data
# outside the reference period are way higher or lower than those within
# the reference period.
plot(spei(ts(wichita[, "BAL"], freq = 12, start = c(1980, 6)), 12,
  ref.start = c(1980, 1), ref.end = c(2000, 1)
))

# Using different kernels
spei24 <- spei(wichita[, "BAL"], 24)
spei24_gau <- spei(wichita[, "BAL"], 24,
  kernel = list(type = "gaussian", shift = 0)
)
par(mfrow = c(2, 1))
plot(spei24)
plot(spei24_gau)
dev.off()

# Using different methods (distributions)
spi_gam <- spi(wichita[, "PRCP"], 12, distribution = "Gamma")
spi_pe3 <- spi(wichita[, "PRCP"], 12, distribution = "PearsonIII")
plot(spi_gam$fitted, spi_pe3$fitted)
grid()

# Using custom (user provided) parameters
coe <- spei1$coefficients
dim(coe)
spei(wichita[, "BAL"], 1, params = coe)

# Matrix input (computing data from several stations at one)
# Dataset `balance` contains time series of the climatic water balance at
# 12 locations. Note that input must be provided as matrix.
data(balance)
head(balance)
bal_spei12 <- spei(as.matrix(balance), 12)
plot(bal_spei12)

# 3-d array input (computing data from a gridded spatio-temporal dataset)
# Dataset cruts4 contains monthly time series of the climatic water balance
# at six locations, in a gridded format (3-d array).
data(cruts4)
dim(cruts4)
spei_12 <- spei(cruts4, 12)
dim(spei_12$fitted)

```

```

# Modding the plot
# Since plot.spei() returns a ggplot object, it is possible to add or tweak
# parts of the plot.
require(ggplot2)
plot(spei(wichita[, "BAL"], 12)) +
  ggtitle("SPEI1 at Wichita") +
  scale_fill_manual(values = c("blue", "red")) + # classic SPEI look
  scale_color_manual(values = c("blue", "red")) + # classic SPEI look
  theme_classic() +
  theme(legend.position = "bottom")

```

---

Generalized-Logistic    *Generalized Logistic maximum likelihood function*

---

### Description

Maximum likelihood fitting function for generalized logistic distribution.

### Usage

```
parglo.maxlik(x, ini)
```

### Arguments

`x`                    vector of quantiles for which to evaluate the PDF.  
`ini`                    a vector of initial values of the parameters to be fit.

### Details

This function is used internally by `spei` and `spi` and is supposed to never be needed by the regular user. Initial values for maximum likelihood estimation can be provided by `parglo`.

### Value

a list of parameters of a generalized Logistic distribution function

### Author(s)

Santiago Beguería

### References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

---

Generic-methods-for-spei-objects

*Generic methods for spei objects.*

---

## Description

Generic methods for extracting information and plotting spei objects.

See print.spei

See print.spei

## Usage

```
## S3 method for class 'spei'  
print(x, ...)  
## S3 method for class 'spei'  
summary(object, ...)  
## S3 method for class 'spei'  
plot(x, ...)
```

```
## S3 method for class 'spei'  
summary(object, ...)
```

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

## Arguments

x	an object of class spei.
...	additional parameters, not used at present.
object	an object of class spei.

## Details

These functions allow extracting information and plotting spei objects. `print` yields the fitted values, i.e. a time series of SPEI or SPI values. `summary` reports the function call, the parameters of the PDF used, and the time series of SPEI or SPI values. `plot` produces a plot of the time series of SPEI or SPI values, with blue and red colors for positive and negative values, respectively. If a reference period was used in the function call it is shown by a shaded area. In the event that NA or Inf values were produced, these are shown by circles.

See print.spei

See print.spei

## Author(s)

Santiago Beguería

## References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

## Examples

```
# See examples of use in the help page of the spei() function.
```

---

Kernel-functions      *Time kernel for computing the SPEI at different time scales.*

---

## Description

Function kern is used internally by [spei](#) and [spi](#) for computing drought indices at different time scales.

See kern

## Usage

```
kern(scale, type = "rectangular", shift = 0)
```

```
kern.plot(scale = 12, shift = 0)
```

## Arguments

scale	numeric, time scale or length of the kernel.
type	character, shape of the kernel function.
shift	numeric, shifting of the kernel peak.

## Details

Drought indices, such as the SPEI or the SPI, are usually computed at different time scales to adapt to the different response times of systems affected by drought. This is accomplished by applying a kernel function to the data prior to computation of the SPEI. Application of a kernel has the effect of smoothing the temporal variability of the resulting SPEI, allowing for the major patterns to emerge from the noise. Other way of considering it is that the kernel allows incorporating information of previous time steps into the calculation of the current time step, so the resulting values of the SPEI adapt to the memory of the system under study.

The most common kernel function is rectangular, i.e. all the data of the previous  $n$  time steps are given equal weight. This was the way the Standardized Precipitation Index (SPI) was defined, and it is also the way the SPEI is computed. This would be the default option for the kern function. However, data from the past can be thought of as having a decreasing influence in the current state of the system as the temporal lag between them increases. The function kern allows weighting

the past data as a function of the time lapse, according to a series of pre-defined shapes. Available options are 'rectangular' (default), 'triangular', 'circular' and 'gaussian'.

By default the highest weight will be given to the observation of the current month. However, it is possible to modify this by setting the `shift` parameter to a value higher than zero. This will cause the highest weight be given to the  $n$  antecedent observation.

`kern.plot` produces plots of the weighting factor against the time lag for the four different kernel shapes so they can be compared.

See `kern`

### Value

A vector of length equal to `scale` with weights used for computing the drought index.

### Author(s)

Santiago Beguería

### References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

### Examples

```
# A rectangular kernel with a time scale of 12 and no shift
kern(12)
# A gaussian kernel with a time scale of 12 and no shift
kern(12, 'gaussian')
# Comparison of the four kernels, with and without shift
kern.plot(12)
kern.plot(12, 2)
```

---

Potential-evapotranspiration

*Computation of potential and reference evapotranspiration.*

---

### Description

Potential evapotranspiration (PET) is the amount of evaporation and transpiration that would occur if a sufficient water source were available. Reference evapotranspiration (ET<sub>o</sub>) is the amount of evaporation and transpiration from a reference vegetation of grass. They are usually considered equivalent. This set of functions calculate PET or ET<sub>o</sub> according to the Thornthwaite, Hargreaves or Penman-Monteith equations.

See `hargreaves`

See `hargreaves`

**Usage**

```
thornthwaite(Tave, lat, na.rm = FALSE, verbose=TRUE)
```

```
hargreaves(Tmin, Tmax, Ra = NULL, lat = NULL, Pre = NULL, na.rm = FALSE,
verbose=TRUE)
```

```
penman(Tmin, Tmax, U2, Ra = NULL, lat = NULL, Rs = NULL, tsun = NULL,
      CC = NULL, ed = NULL, Tdew = NULL, RH = NULL, P = NULL, P0 = NULL,
      CO2 = NULL, z = NULL, crop='short', na.rm = FALSE, method='ICID',
      verbose=TRUE)
```

```
penman(
  Tmin,
  Tmax,
  U2 = NULL,
  Ra = NULL,
  lat = NULL,
  Rs = NULL,
  tsun = NULL,
  CC = NULL,
  ed = NULL,
  Tdew = NULL,
  RH = NULL,
  P = NULL,
  P0 = NULL,
  CO2 = NULL,
  z = NULL,
  crop = "short",
  na.rm = FALSE,
  method = "ICID",
  verbose = TRUE
)
```

```
thornthwaite(Tave, lat, na.rm = FALSE, verbose = TRUE)
```

**Arguments**

Tmin	a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily minimum temperatures, °C.
Tmax	a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily maximum temperatures, °C.
Ra	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily external radiation, MJ m <sup>-2</sup> d <sup>-1</sup> .
lat	a numeric vector or matrix with the latitude of the site or sites, in degrees.
Pre	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly total precipitation, mm.

na.rm	optional, a logical value indicating whether NA values should be stripped from the computations.
verbose	optional, logical, report the computation options during calculation. Either 'TRUE' (default) or 'FALSE'.
U2	a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily wind speeds at 2 m height, m s <sup>-1</sup> .
Rs	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily incoming solar radiation, MJ m <sup>-2</sup> d <sup>-1</sup> .
tsun	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily bright sunshine hours, h.
CC	optional, numeric a vector, matrix or time series of monthly mean cloud cover, %.
ed	optional, numeric a vector, matrix or time series of monthly mean actual vapor pressure at 2 m height, kPa.
Tdew	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean daily dewpoint temperature (used for estimating ed), °C.
RH	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean relative humidity (used for estimating ed), %.
P	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean atmospheric pressure at surface, kPa.
P0	optional, a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean atmospheric pressure at sea level (used for estimating P), kPa.
CO2	optional, a single numeric value, a numeric vector, or a tsvector of monthly mean CO2 atmospheric concentration, ppm.
z	optional, a numeric vector or matrix of the elevation of the site or sites, m above sea level.
crop	optional, character string, type of reference crop. Either one of 'short' (default) or 'tall'.
method	optional, character string, Penman-Monteith calculation method. Either one of 'ICID' (default), 'FAO', or 'ASCE'.
Tave	a numeric vector, tsvector, matrix, tsmatrix, or 3-d array of monthly mean temperatures, °C.

### Details

thornthwaite computes the monthly potential evapotranspiration (PE) according to the Thornthwaite (1948) equation. It is the simplest of the three methods, and can be used when only mean temperature data are available.

hargreaves computes the monthly reference evapotranspiration (ET<sub>0</sub>) of a grass crop based on the original Hargreaves equation (1994). However, if precipitation data *Pre* is provided a modified form due to Droogers and Allen (2002) will be used; this equation corrects ET<sub>0</sub> using the amount of rain of each month as a proxy for irradiation. The Hargreaves method requires data on the mean external radiation, *R<sub>a</sub>*. If *R<sub>a</sub>* is not available it can be estimated from the latitude *lat* and the month of the year.

penman calculates the monthly reference evapotranspiration (ET<sub>o</sub>) of a hypothetical reference crop according to the Penman-Monteith equation (Monteith, 1965). This is widely considered the most accurate method for estimating ET<sub>o</sub>, and is the method recommended by the UN Food and Agriculture Organization (FAO). There are several methods which simplify the original Penman-Monteith equation and are used in practice. Here we follow by default the procedure described in Allen et al. (1994), aka the 'ICID' method. However, other versions are also implemented and can be used by setting the method parameter to the appropriate value. The FAO-56 method (Allen et al., 1998) can be used by setting method to 'FAO', while the variation of the American Society of Civil Engineers (Walter et al., 2002) is used when setting it to 'ASCE'. By default the original parameterization corresponding to a short reference crop of 0.12 m height is used, although the parameterization for a tall reference crop of 0.5 m height (Walter et al. (2002)) can also be used, by setting the crop parameter to 'tall'. The method requires data on the incoming solar radiation, R<sub>s</sub>; since this is seldom available, the code will estimate it from data on the bright sunshine duration t<sub>sun</sub>, or alternatively from data on the percent cloud cover CC. Similarly, if data on the saturation water pressure e<sub>s</sub> are not available, it is possible to estimate it from the dew point temperature T<sub>dew</sub>, from the relative humidity RH or even from the minimum temperature T<sub>min</sub> (sorted from least to most uncertain method). Similarly, the atmospheric surface pressure P required for computing the psychrometric constant can be calculated from the atmospheric pressure at sea level P<sub>0</sub> and the elevation z, or else it will be assumed to be constant (101.3 kPa). If no wind speed data U<sub>2</sub> are available, a constant value of 2 m per second is used. Custom CO<sub>2</sub> atmospheric concentration CO<sub>2</sub> can also be provided, following Yang et al. (2019).

The function will produce an error message if a valid combination of input parameters is not provided. If verbose is 'TRUE' (the default value), a message will be produced informing on the computation options selected.

The function accepts data in a variety of formats, including 1-D vectors, 2-D matrices and 3-D arrays. The input data format will determine the output format. Vector input can be used for single-station data. Matrix input can be used to produce output for a number of locations, where each column in the matrix will be considered as one location. 3-D arrays can be used to process gridded data, with the first dimension being time and the two other dimensions representing spatial coordinates. See the examples below for a guidance on how to use these options.

If the main input object (T<sub>ave</sub>, T<sub>min</sub>, T<sub>max</sub>) is a time series then the function `cycle` will be used to determine the position of each observation within the year (month), allowing the data to start in a month different than January. If no time information is provided, then the input data will be treated as a sequence of monthly values starting in January.

See hargreaves

See hargreaves

## Value

A vector, matrix or 3-d array with the values of monthly potential or reference evapotranspiration, in mm.

A time series with the values of monthly potential or reference evapotranspiration, in mm. If the input is a matrix or a multivariate time series each column will be treated as independent data (e.g., different observatories), and the output will be a multivariate time series.

A time series with the values of monthly potential or reference evapotranspiration, in mm. If the input is a matrix or a multivariate time series each column will be treated as independent data (e.g., different observatories), and the output will be a multivariate time series.



**Author(s)**

Santiago Beguería

**References**

- Thornthwaite, C. W., 1948. An approach toward a rational classification of climate. *Geographical Review* **38**: 55–94. DOI:10.2307/2107309.
- Hargreaves G.H., 1994. Defining and using reference evapotranspiration. *Journal of Irrigation and Drainage Engineering* **120**: 1132–1139.
- Droogers P., Allen R. G., 2002. Estimating reference evapotranspiration under inaccurate data conditions. *Irrigation and Drainage Systems* **16**: 33–45.
- Monteith, J. L., 1965. Evaporation and environment. *Symposia of the Society for Experimental Biology* **19**: 205–234.
- Allen R. G., Smith M., Pereira L. S., Perrier A., 1994. An update for the calculation of reference evapotranspiration. *ICID Bulletin of the International Commission on Irrigation and Drainage*, 35–92.
- Allen R.G., Pereira L.S., Raes D., Smith, M., 1998. *J. Crop evapotranspiration - Guidelines for computing crop water requirements - FAO Irrigation and drainage paper 56*. FAO, Rome. ISBN 92-5-104219-5.
- Walter I.A. and 14 co-authors, 2002. *The ASCE standardized reference evapotranspiration equation*. Rep. Task Com. on Standardized Reference Evapotranspiration July 9, 2002, EWRI–Am. Soc. Civil Engr., Reston, VA, 57 pp.
- Yang, Y., Roderick, M.L., Zhang, S. McVicar, T., Donohue, R.J., 2019. Hydrologic implications of vegetation response to elevated CO<sub>2</sub> in climate projections. *Nature Climate Change* **9**: 44–48.

**Examples**

```
# Load data for Tampa, lat=37.6475N, elevation=402.6 m. a.s.l.
# Data consists on monthly values since January 1980

data(wichita)
attach(wichita)
names(wichita)

# PET following Thornthwaite
tho <- thornthwaite(TMED, 37.6475)

# ETo by Hargreaves
har <- hargreaves(TMIN, TMAX, lat = 37.6475)

# ETo by Penman, based on sun hours, ignore NAs
pen <- penman(TMIN, TMAX, AWND, tsun = TSUN, lat = 37.6475, z = 402.6, na.rm = TRUE)

# Penman, based on cloud cover
pen2 <- penman(TMIN, TMAX, AWND, CC = ACSH, lat = 37.6475, z = 402.6, na.rm = TRUE)

# Penman, with constant wind
pen3 <- penman(TMIN, TMAX, tsun = TSUN, lat = 37.6475, z = 402.6, na.rm = TRUE)
```

```

# Plot them together
plot(ts(cbind(tho, har, pen, pen2, pen3), fr = 12))

# Compare between methods
pairs(cbind(tho, har, pen, pen2, pen3))

# Input data as a time series vector; note that only the first parameter
# needs to be a `ts` object.

thornthwaite(ts(TMED, start = c(1980, 1), frequency = 12), 37.6475)
hargreaves(ts(TMIN, start = c(1980, 1), frequency = 12), TMAX, lat = 37.6475)
penman(ts(TMIN, start = c(1980, 1), frequency = 12), TMAX, AWND,
        tsun = TSUN,
        lat = 37.6475, z = 402.6, na.rm = TRUE
)

# Input data as a time series. Consider the data started in June 1980
thornthwaite(ts(TMED, start = c(1980, 6), frequency = 12), 37.6475)

# Comparison with example from Allen et al. (1998), p. 69, fig. 18:
# Data from Cabinda, Angola (-5.33S, 12.11E, 20 m a.s.l.)

data(cabinda)
pen.cab <- penman(cabinda$Tmin, cabinda$Tmax, cabinda$U2,
                 Rs = cabinda$Rs, tsun = cabinda$tsun, RH = cabinda$RH, lat = -5.33, z = 20
)
plot(cabinda$ET0, pen.cab)
abline(0, 1, lt = "dashed")
summary(lm(pen.cab ~ cabinda$ET0))$r.squared

# Matrix input (data from several stations)
# Replicating Wichita data twice to simulate data at two locations.

tmin <- cbind(TMIN, TMIN + 1.5)
tmax <- cbind(TMAX, TMAX + 1.5)
lat <- c(37.6475, 35.0000)
har <- hargreaves(tmin, tmax, lat = lat, na.rm = TRUE)
plot(har)
abline(0, 1)
plot(ts(har, fr = 12))

# Array input (gridded data)
# Replicating Wichita data to simulate data from a grid. Note that the time
# dimension (`nt`) comes first. Latitude is provided as a 2-d array.

nt <- length(TMIN)
tmin <- array(TMIN, dim = c(nt, 2, 2))
tmax <- array(TMAX, dim = c(nt, 2, 2))
lat <- array(c(40, 30, 40, 30), dim = c(2, 2))
har <- hargreaves(tmin, tmax, lat = lat, na.rm = TRUE)
dim(har)

```

```

# Different Penman-Monteith flavors

pen_igid <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  na.rm = TRUE, method = "IGID"
)
pen_asce <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  na.rm = TRUE, method = "ASCE"
)
pen_fao <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  na.rm = TRUE, method = "FAO"
)
plot(ts(cbind(pen_igid, pen_asce, pen_fao), fr = 12))

# Different CO2 concentrations

# Default (300 ppm)
pen_300 <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  na.rm = TRUE
)
# Increased to 450 ppm
pen_450 <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  CO2 = 450, na.rm = TRUE
)
plot(pen_450, pen_300)
abline(0, 1)
# Increasing from 300 to 450
co2 <- seq(300, 450, length.out = length(TMIN))
pen_co2 <- penman(TMIN, TMAX, AWND,
  tsun = TSUN, lat = 37.6475, z = 402.6,
  CO2 = co2, na.rm = TRUE
)
plot(ts(cbind(pen_300, pen_co2), fr = 12))

```

---

SPEI

*SPEI.*


---

### Description

A set of functions for computing potential evapotranspiration and several widely used drought indices including the Standardized Precipitation-Evapotranspiration Index (SPEI).

### Details

Package: SPEI  
Type: Package  
Version: 1.8  
Date: 2023-01-20  
License: GPL version 2 or newer  
LazyLoad: yes

Functions `spei` and `spi` are the workhorse of the SPEI library. Other functions such as `kern`, `parglo.maxlik` is an auxiliary low-level function and will not be used directly by the typical user. Functions for computing potential evapotranspiration are provided, too, for helping computing the SPEI. They are: `thornthwaite`, `hargreaves` and `penman`.

### Author(s)

Santiago Beguería and Sergio M. Vicente-Serrano Maintainer: Santiago Beguería Contributors: Fergus Reig

### References

S.M. Vicente-Serrano, S. Beguería, J.I. López-Moreno. 2010. A Multi-scalar drought index sensitive to global warming: The Standardized Precipitation Evapotranspiration Index – SPEI. *Journal of Climate* **23**: 1696, DOI: 10.1175/2009JCLI2909.1.

<https://spei.csic.es>

---

SPEINews

*SPEINews*

---

### Description

Show the NEWS file of the **SPEI** package.

### Usage

SPEINews()

### Details

(See description)

# Index

- \* **datasets**
  - Datasets, [2](#)
- \* **package**
  - SPEI, [19](#)
- balance (Datasets), [2](#)
- cabinda (Datasets), [2](#)
- cruts4 (Datasets), [2](#)
- cycle, [6](#), [16](#)
- Datasets, [2](#)
- Drought-indices, [4](#)
- Generalized-Logistic, [10](#)
- Generic-methods-for-spei-objects, [11](#)
- hargreaves, [8](#), [20](#)
- hargreaves
  - (Potential-evapotranspiration), [13](#)
- kern, [7](#), [8](#), [20](#)
- kern (Kernel-functions), [12](#)
- Kernel-functions, [12](#)
- parglo.maxlik, [20](#)
- parglo.maxlik (Generalized-Logistic), [10](#)
- penman, [8](#), [20](#)
- penman (Potential-evapotranspiration), [13](#)
- plot.spei, [8](#)
- plot.spei
  - (Generic-methods-for-spei-objects), [11](#)
- Potential-evapotranspiration, [13](#)
- print.spei, [8](#)
- print.spei
  - (Generic-methods-for-spei-objects), [11](#)
- SPEI, [19](#)
- spei, [12](#), [20](#)
- spei (Drought-indices), [4](#)
- spei, (Drought-indices), [4](#)
- SPEINews, [20](#)
- spi, [12](#), [20](#)
- spi (Drought-indices), [4](#)
- summary.spei, [8](#)
- summary.spei
  - (Generic-methods-for-spei-objects), [11](#)
- thornthwaite, [8](#), [20](#)
- thornthwaite
  - (Potential-evapotranspiration), [13](#)
- wichita (Datasets), [2](#)