Package 'palinsol'

August 29, 2016

Version 0.93

Date 2016-02-28

Title Insolation for Palaeoclimate Studies

Author Michel Crucifix

Maintainer Michel Crucifix <michel.crucifix@uclouvain.be>

Depends R (>= 2.10), stats, gsl,

Description
R package to compute Incoming Solar Radiation (insolation) for palaeoclimate studies. Features three solutions: Berger (1978), Berger and Loutre (1991) and Laskar et al. (2004). Computes daily-mean, season-averaged and annual means for all latitudes.

License file LICENSE

URL https://bitbucket.org/mcrucifix/insol

NeedsCompilation no

Repository CRAN

Date/Publication 2016-03-05 00:15:41

R topics documented:

`astro` *Compute astronomical parameters in the past or in the future*

Description

–

Usage

```
astro(t,solution,degree=FALSE)
ber78(t,degree=FALSE)
ber90(t,degree=FALSE)
la04(t,degree=FALSE)
precession(t,solution)
obliquity(t,solution,degree=FALSE)
```

Arguments

`t`	Time, years after 1950
`solution`	solution used. One of `ber78`, `ber90` or `la04`
`degree`	returns angles in degrees if `TRUE`

Details

Both `ber78` and `ber90` compute astronomical elements based on a spectral decomposition (sum of sines and cosines) of obliquity and planetary precession parameters. `ber78` uses the Berger (1978) algorithm and is accurate for +/- 1e6 years about the present. `ber90` uses the Berger and Loutre (1991) algorithm and is accurate for +/- 3e6 years about the present (but with a tiny accuracy over the last 50 kyr, usually negligible for any palaeo application, see example below).

`la04` interpolates tables provided by Laskar (2004), obtained by a simplectic numerical integration of the planetary system, in which the Moon is considered as a planet. This solution is valid for about 50 Myr around the present.

`precession` and `obliquity` do as `astro`, but only return precession (e sin varpi) parameter and obliquity, respectively.

Value

A vector of 3 (la04) or 4 (`ber78` and `ber90`) astronomical elements

`eps`	obliquity,
`ecc`	eccentricity and
`varpi` true solar longitude of the perihelion.	

`ber78` and `ber90` also return `epsp`, the Hilbert transform of obliquity (sines changed in cosines in the spectral decomposition).

Angles are returned in radians unless `degree=TRUE`

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Berger, A. L. (1978). Long-term variations of daily insolation and Quaternary climatic changes, J. Atmos. Sci., 35, 2362-2367, doi:10.1175/1520-0469(1978)035<2362:LTVODI>2.0.CO;2

Berger and M.F. Loutre (1991), Insolation values for the climate of the last 10 million years, Quaternary Science Reviews, 10, 297 - 317, doi:10.1016/0277-3791(91)90033-Q

J. Laskar et al. (2004), A long-term numerical solution for the insolation quantities of the Earth, Astron. Astroph., 428, 261-285, doi:10.1051/0004-6361:20041335

Examples

```
## compare the obliquity over the last 2 Myr with the three solutions

times <- seq(-2e6,0,1e3)
Obl <- function(t) {c(time=t,ber78=ber78(t)['eps'],
       ber90=ber90(t)['eps'], la04=la04(t)['eps'])}

Obls <- data.frame(t(sapply(times,Obl)))
## may take about 10 seconds to run
with(Obls, {
  plot(times/1e3, ber78.eps, type='l', xlab='time (kyr)',
                                       ylab='Obliquity (radians)')
  lines(times/1e3, ber90.eps, type='l', col='red')
  lines(times/1e3, la04.eps, type='l', col='green')
  })

legend('topright', c('ber78','ber90','la04'), col=c('black','red','green'), lty=1)

## same but with a zoom over the last 300 000 years:

T <- which (times > -3e5)
with(Obls, {
  plot(times[T]/1e3, ber78.eps[T], type='l', xlab='time (kyr)',
                                       ylab='Obliquity (radians)')
  lines(times[T]/1e3, ber90.eps[T], type='l', col='red')
  lines(times[T]/1e3, la04.eps[T], type='l', col='green')
  })

legend('topright', c('ber78','ber90','la04'), col=c('black','red','green'), lty=1)
```

BER78 *Tables supplied by BER78 and BER90*

Description

Tables necessary for computation of astronomical solutions by Berger (1978) or Berger and Loutre (1990).

Usage

```
data(BER78)
data(BER90)
```

Format

text files reproducing the published tables with original units.

Note

Table 2 is reconstructed based on the electronic files supplied by the authors, based on the method provided by Berger and Loutre (1990) below.

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

Source

ftp://ftp.elic.ucl.ac.be/berger/berger78/ ftp://ftp.elic.ucl.ac.be/loutre/QSR/

References

Berger, A. L. (1978). Long-term variations of daily insolation and Quaternary climatic changes, J. Atmos. Sci., 35, 2362-2367, doi:10.1175/1520-0469(1978)035<2362:LTVODI>2.0.CO;2

A. Berger and M. F. Loutre (1990), Origine des fr\'equences des \'el\'ements astronomiques intervenant dans l'insolation, Bull. Classe des Sciences, 1-3, 45-106

Berger and M.F. Loutre (1991), Insolation values for the climate of the last 10 million years, Quaternary Science Reviews, 10, 297 - 317, doi:10.1016/0277-3791(91)90033-Q

Examples

```
data(BER78)
data(BER90)
```

calins *Caloric insolation*

Description

Computes caloric summer insolation for a given astronomical configuration and latitude.

Usage

```
calins (orbit,lat=65*pi/180,...)
```

Arguments

`orbit`	Output from a solution, such as `ber78`, `ber90` or `la04`
`lat`	latitude
`...`	Other arguments passed to Insol

Details

The caloric summer is a notion introduced by M. Milankovitch. It is defined as the halve of the tropical year during for which daily mean insolation are greater than all days of the other halves. The algorithm is an original algorithm by M. Crucifix, but consistent with earlier definitions and algorithms by A. Berger (see examples). Do not confuse this Berger (1978) reference with the Berger (1978), J. Atm. Sci. of the astronomical solution.

Value

Time-integrated insolation in kJ/m2 during the caloric summer.

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Berger (1978) Long-term variations of caloric insolation resulting from the earth's orbital elements, Quaternary Research, 9, 139 - 167.

Examples

```
## reproduces Table 2 of Berger 1978
lat <- seq(90, 0, -10) * pi/180. ## angles in radiants.
orbit_1 = ber78(0)
orbit_2 = orbit_1
orbit_2 ['eps'] = orbit_2['eps'] + 1*pi/180.

T <-  sapply(lat, function(x) c(lat = x * 180/pi,
                          calins(orbit_2, lat=x, S0=1365) / (4.18 * 1e1)
```

```
                        - calins(orbit_1, lat=x, S0=1365) / (4.18 * 1e1) ) )
data.frame(t(T))
# there are still some differences, of the order of 0.3 %, that are probably related to
# the slightly different methods.
# 41.8 is the factor from cal/cm2 to  kJ/m2
```

---

day2l *Converts calendar day into true solar longitude and vice-versa*

Description

Converts calendar day into true solar longitude for a given astronomical configuration and vice-versa

Usage

```
day2l  (orbit,day)
l2day  (orbit,l)
date_of_perihelion (orbit)
```

Arguments

orbit	Output from a solution, such as `ber78`, `ber90` or `la04`
l	true solar longitude, in radiants
day	calendar day, in a 360-d year

Details

The 360-d calendar is a conventional calendar, for which day 80 is the day of NH spring equinoxe. The tropic year, which in reality is 365.24219876 * 86400 seconds was the practical reference to define the Gregorian Calendar since this is the time needed to go through all the seasons. More discussion of calendars and conversions in Berger et al. (2010) appendix D.

The `day2l` and `l2day` is based on algoritms given in Berger (1978), but which can be traced back to expansions of the mean and true anomaly by Brouwer and Clemente (1961), pp. 65 and 77 (see code for further details).

Value

day of year (360-d cal.) or true solar longitude (in radiants).

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Brouwer D. and G. M. Clemence, (1961), Methods of celestial mechanics, Academic Press, New York.

Berger, (1978) Long-term variations of daily insolation and Quaternary climatic changes, J. Atmos. Sci., 35, 2362-2367 1978, doi:10.1175/1520-0469(1978)035<2362:LTVODI>2.0.CO;2

Berger, A. Loutre, M.F. and Yin Q. (2010), Total irradiation during any time interval of the year using elliptic integrals, Quaternary Science Reviews, 29, 1968 - 1982, doi:10.1016/j.quascirev.2010.05.007

Examples

```
## date of perihelion throughout today
orbit=c(eps=0.409214, ecc=0.01672393, varpi=4.92251)
date_of_perihelion(orbit)
## date of winter solstice)
l2day(orbit, 270*pi/180.)
```

Insol — *Computes incoming solar radiation (insolation)*

Description

Computes incoming solar radiation (insolation) for a given astronomical configuration, true solar longitude and latitude

Usage

```
Insol (orbit,long=pi/2, lat=65*pi/180,S0=1365)
```

Arguments

`orbit`	Output from a solution, such as `ber78`, `ber90` or `la04`
`long`	true solar longitude
`lat`	latitude
`S0`	Total solar irradiance

Details

True solar longitude is measured in radians:

pi/2	for June solstice
pi	for September equinox
3 * pi/2	for December solstice
0	for Spring equinox

It may be obtained for a given day in the year using the function `day2l`.

Value

Daily-mean insolation (assuming fixed astronomical parameters during a true solar day)

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Berger, A. L. (1978). Long-term variations of daily insolation and Quaternary climatic changes, J. Atmos. Sci., 35, 2362-2367.

Examples

```
## make a little wrapper, with all default values

insolation <- function(times, astrosol=ber78,...)
  sapply(times, function(tt) Insol(orbit=astrosol(tt)))

tts <- seq(from = -400e3, to = 0, by = 1e3)
isl <- insolation(tts, ber78)
plot(tts, isl, typ='l')
```

Insol_l1l2 *Time-integrated insolation*

Description

Computes time-integrated incoming solar radiation (Insol) either between given true solar longitudes (`Insol_l1l2`) or days of year (`Insol_d1d2`) for a given orbit and latitude

Usage

```
Insol_l1l2 (orbit,l1=0,l2=2*pi,lat=65*pi/180,avg=FALSE,ell=TRUE,...)
Insol_d1d2 (orbit,d1,d2,lat=65*pi/180,avg=FALSE,...)
```

Arguments

`orbit`	Output from a solution, such as `ber78`, `ber90` or `la04`
`lat`	latitude
`l1`	lower true solar longitude bound of the time-integral
`l2`	upper true solar longitude bound of the time-integral
`d1`	lower calendar day (360-day-year) of the time-integral
`d2`	upper calendar day (360-day-year) of the time-integral

`avg`	performs a time-average.
`ell`	uses elliptic integrals for the calculation (much faster)
`...`	other arguments to be passed to `Insol`

Details

All angles input measured in radiants.

Note that in contrast to Berger (2010) we consider the tropic year as the reference, rather than the sideral year, which partly explains some of the small differences with the original publication

Value

Time-integrated insolation in kJ/m2 if `avg=TRUE`, else time-average in W/m2

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Berger, A., Loutre, M.F. and Yin Q. (2010), Total irradiation during any time interval of the year using elliptic integrals, Quaternary Science Reviews, 29, 1968 - 1982, doi:10.1016/j.quascirev.2010.05.007

Examples

```
## reproduces Table 1a of Berger et al. 2010:
lat <- seq(85, -85, -10) * pi/180. ## angles in radiants.
orbit=c(eps=  23.446 * pi/180., ecc= 0.016724, varpi= (102.04 - 180)* pi/180. )
T <-  sapply(lat, function(x) c(lat = x * 180/pi,
        m1 =  Insol_l1l2(orbit, 0, 70 * pi/180, lat=x, ell= TRUE, S0=1368) / 1e3,
        m2 =  Insol_l1l2(orbit, 0, 70 * pi/180, lat=x, ell=FALSE, S0=1368) / 1e3) )
data.frame(t(T))
 ## reproduces Table 1b of Berger et al. 2010:
lat <- c(85, 55, 0, -55, -85) * pi/180. ## angles in radiants.
T <-  sapply(lat, function(x) c(lat = x * 180/pi,
         m1 =  Insol_l1l2(orbit, 30 * pi/180. , 75 * pi/180,
               lat=x, ell= TRUE, S0=1368) / 1e3,
         m2 =  Insol_l1l2(orbit, 30 * pi/180. , 75 * pi/180,
               lat=x, ell=FALSE, S0=1368) / 1e3) )
 ## reproduces Table 2a of Berger et al. 2010:
lat <- seq(85, -85, -10) * pi/180. ## angles in radiants.

## 21 march in a 360-d year. By definition : day 80 = 21 march at 12u
d1 = 79.5
d2 = 79.5 + (10 + 30 + 30 ) * 360/365.2425 ## 30th May in a 360-d year

T <-  sapply(lat, function(x) c(lat = x * 180/pi,
        m1 =  Insol_d1d2(orbit, d1,d2, lat=x, ell= TRUE, S0=1368) / 1e3,
        m2 =  Insol_d1d2(orbit, d1,d2, lat=x, ell= FALSE, S0=1368) / 1e3))

## I did not quite get the same results as on the table
```

```
## on this one; probably a matter of calendar
## note : the authors in fact used S0=1368 (pers. comm.)
## 1366 in the paper is a misprint

data.frame(t(T))
```

LA04 *Astronomical elements supplied by Laskar et al. 2004*

Description

Astronomical elements (longitude of perihelion, obliquity and eccentricity) supplied by Laskar et al. 2004 by stey of 1ka, from -51Ma to present (la04past) and from present to + 21Ma.

Usage

```
data(LA04)
```

Format

text files

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

Source

http://www.imcce.fr/Equipes/ASD/insola/earth/earth.html

References

J. Laskar et al., A long-term numerical solution for the insolation quantities of the Earth, Astron. Astroph., 428, 261-285 200

Examples

```
data(LA04)
```

Milankovitch *Milankovitch graph for a given astronomical configuration*

Description

Computes the distrubition in latitude and longitude of incoming solar radiation, known as a Milankovitch graph, with possibility of plotting with a dedicated plot function

Usage

```
Milankovitch (orbit, S0=1365, lat=seq(-pi/2, pi/2, l=73),
              long=seq(0, 2*pi, l=145), deg=TRUE)
```

Arguments

`orbit`	Output from a solution, such as `ber78`, `ber90` or `la04`
`S0`	Total solar irradiance
`lat`	latitudes, passed as an array
`long`	true solar longitudes, passed as an array
`deg`	If true : the axes of the Milankovitch object are expressed in degrees. Inputs are always in radiants

Value

A object of Milankovitch class, which may be plotted using the regular plot function

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

Berger, A. L. (1978). Long-term variations of daily insolation and Quaternary climatic changes, J. Atmos. Sci., 35, 2362-2367.

Examples

```
orbit <- c(eps=0.409214, ecc=0.01672393, varpi=4.92251)
M <- Milankovitch(orbit)
plot(M, plot=contour)
plot(M, plot=contour, month=FALSE)
```

`plot.Milankovitch` *plot Milankovitch graph*

Description

plot Milankovitch object

Usage

```
## S3 method for class 'Milankovitch'
 plot(x, months=TRUE, plot_function=image,...)
```

Arguments

`x`	Milankovitch object
`months`	if true : x-axis of the plot indicates months conventionnally defined with the true solar longitude; x-axis is simply the true solar longitude otherwise
`...`	Other arguments passed to plotting function
`plot_function`	function used to plot the matrix. Typically `contour` or `image` but may also be `image.plot` if using the `fields` package.

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

`thrins` *Integrated insolation for all days exceeding a threshold*

Description

Integrated insolation over the part during which daily-mean insolation exceeds a threshold, expressed in W/m2

Usage

```
 thrins  (lat=65*pi/180,orbit,threshold=400,...)
```

Arguments

`lat`	latitude
`orbit`	Output from a solution, such as `ber78`, `ber90` or `la04`
`threshold`	threshold insolation ,in W/m2
`...`	other arguments to be passed to Insol

Details

Algorithm is by M. Crucifix, but the idea of thresholded insolation is due to Huybers and Tziperman (2008), reference below.

Value

Time-integrated insolation in kJ/m2 . The quantity is calculated by brute-force integration with a 1-degree time-step in true solar longitude and this can be quite slow if long series are to be calculated.

Author(s)

Michel Crucifix, U. catholique de Louvain, Belgium.

References

P. Huybers and E. Tziperman (2008), Integrated summer insolation forcing and 40,000-year glacial cycles: The perspective from an ice-sheet/energy-balance model, Paleoceanography, 23.

Index