Package ‘splash’

October 12, 2022

Title Simple Process-Led Algorithms for Simulating Habitats

Version 1.0.2

Description This program calculates bioclimatic indices and fluxes (radiation, evapotranspiration, soil moisture) for use in studies of ecosystem function, species distribution, and vegetation dynamics under changing climate scenarios. Predictions are based on a minimum of required inputs: latitude, precipitation, air temperature, and cloudiness.
Davis et al. (2017) <doi:10.5194/gmd-10-689-2017>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.1

Depends R (>= 3.2.3)

Language en-GB

URL https://github.com/villegar/splash/,
https://splash.robertovillegas-diaz.com/,
https://bitbucket.org/labprentice/splash/

BugReports https://github.com/villegar/splash/issues/

NeedsCompilation no

Author Tyler W. Davis [aut] (<https://orcid.org/0000-0003-4312-919X>),
Iain Colin Prentice [aut] (<https://orcid.org/0000-0002-1296-6764>),
Benjamin D. Stocker [aut] (<https://orcid.org/0000-0003-2697-9096>),
Rebecca T. Thomas [aut],
Rhys J. Whitley [aut],
Han Wang [aut] (<https://orcid.org/0000-0003-2482-1818>),
Bradley J. Evans [aut],
Angela V. Gallego-Sala [aut],
Martin T. Sykes [aut],
Wolfgang Cramer [aut] (<https://orcid.org/0000-0002-9205-5812>),
Roberto Villegas-Diaz [cre] (<https://orcid.org/0000-0001-5036-8661>)

Maintainer Roberto Villegas-Diaz <r.villegas-diaz@outlook.com>

Repository CRAN

Date/Publication 2022-10-12 15:20:05 UTC

R topics documented:

`calc_daily_evap` *Calculate daily evaporation fluxes*

Description

This function calculates daily radiation, condensation, and evaporation fluxes.

Usage

```
calc_daily_evap(
  lat,
  n,
  elv = 0,
  y = 0,
  sf = 1,
  tc = 23,
  sw = 1,
  ke = 0.0167,
  keps = 23.44,
  komega = 283,
  kw = 0.26
)
```

Arguments

lat	double, decimal degrees.
n	double, day of year.
elv	double, elevation, m A.S.L. Default: 0.
y	double, year. Default: 0.
sf	double, fraction of sunshine hours. Default: 1.
tc	double, mean daily air temperature, degrees C. Default: 23.0.
sw	double, evaporative supply rate, mm/hr. Default: 1.0.
ke	double, eccentricity of earth's orbit. Default: 0.01670, 2000CE (Berger, 1978).
keps	double, obliquity of earth's elliptic. Default: 23.44, 2000CE (Berger, 1978).

komega	double, lon. of perihelion, degrees Default: 283, 2000CE (Berger, 1978).
kw	double, PET entrainment, $(1 + kw) * EET$ Default: 0.26 (Priestley-Taylor, 1972)

Value

Returns a `list` object with the following variables:

- nu_deg .... true anomaly, degrees
- lambda_deg .... true longitude, degrees
- dr .... distance factor, unitless
- delta_deg .... declination angle, degrees
- hs_deg .... sunset angle, degrees
- ra_j.m2 .... daily extraterrestrial radiation, J/m^2
- tau .... atmospheric transmittivity, unitless
- ppfd_mol.m2 .... daily photosyn photon flux density, mol/m^2
- hn_deg .... net radiation hour angle, degrees
- rn_j.m2 .... daily net radiation, J/m^2
- rnn_j.m2 .... daily nighttime net radiation, J/m^2
- econ_m3.j .... water to energy conversion, m^3/J
- cond_mm .... daily condensation, mm
- eet_mm .... daily equilibrium evapotranspiration, mm
- pet_mm .... daily potential evapotranspiration, mm
- hi_deg .... intersection hour angle, degrees
- aet_mm .... daily actual evapotranspiration, mm

References

Berger, A.L., 1978. Long-term variations of daily insolation and Quaternary climatic changes. Journal of Atmospheric Sciences, 35(12), pp.2362-2367. doi:10.1175/15200469(1978)035<2362:LTVODI>2.0.CO;2

Priestley, C.H.B. and Taylor, R.J., 1972. On the assessment of surface heat flux and evaporation using large-scale parameters. Monthly weather review, 100(2), pp.81-92. doi:10.1175/1520-0493(1972)100<0081:OTAOSH>2.3.CO;2

Examples

```
evap <- splash::calc_daily_evap(lat = 37.7,
                                n = 172,
                                elv = 142,
                                y = 2000,
                                sf = 1,
                                tc = 23.0,
                                sw = 0.9)
cat(sprintf("Evaporation values:\n"))
cat(sprintf("  s: %0.6f Pa/K\n", evap$s_pa.k))
```

```
cat(sprintf("  Lv: %0.6f MJ/kg\n", (1e-6) * evap$lv_j.kg))
cat(sprintf("  Patm: %0.6f bar\n", (1e-5) * evap$patm_pa))
cat(sprintf("  pw: %0.6f kg/m^3\n", evap$pw_kg.m3))
cat(sprintf("  gamma: %0.6f Pa/K\n", evap$gam_pa.k))
cat(sprintf("  Econ: %0.6f mm^3/J\n", (1e9) * evap$econ_m3.j))
cat(sprintf("  Cn: %0.6f mm\n", evap$cond_mm))
cat(sprintf("  rx: %0.6f\n", evap$rx))
cat(sprintf("  hi: %0.6f degrees\n", evap$hi_deg))
cat(sprintf("  EET: %0.6f mm\n", evap$eet_mm))
cat(sprintf("  PET: %0.6f mm\n", evap$pet_mm))
cat(sprintf("  AET: %0.6f mm\n", evap$aet_mm))
```

`calc_daily_solar` *Calculate daily solar radiation fluxes*

Description

This function calculates daily solar radiation fluxes.

Usage

```
calc_daily_solar(
  lat,
  n,
  elv = 0,
  y = 0,
  sf = 1,
  tc = 23,
  ke = 0.0167,
  keps = 23.44,
  komega = 283,
  kA = 107,
  kalb_sw = 0.17,
  kalb_vis = 0.03,
  kb = 0.2,
  kc = 0.25,
  kd = 0.5,
  kfFEC = 2.04,
  kGsc = 1360.8
)
```

Arguments

`lat`	double, decimal degrees.
`n`	double, day of year.
`elv`	double, elevation, m A.S.L. Default: 0.
`y`	double, year. Default: 0.

`sf`	double, fraction of sunshine hours. Default: 1.
`tc`	double, mean daily air temperature, degrees C. Default: 23.0.
`ke`	double, eccentricity of earth's orbit. Default: 0.01670, 2000CE (Berger, 1978).
`keps`	double, obliquity of earth's elliptic. Default: 23.44, 2000CE (Berger, 1978).
`komega`	double, lon. of perihelion, degrees Default: 283, 2000CE (Berger, 1978).
`kA`	double, empirical constant, degrees Celsius. Default: 107 (Monteith and Unsworth, 1990).
`kalb_sw`	double, shortwave albedo. Default: 0.17 (Federer, 1968).
`kalb_vis`	double, visible light albedo. Default: 0.03 (Sellers, 1985).
`kb`	double, empirical constant. Default: 0.20 (Linacre, 1968).
`kc`	double, cloudy transmittivity. Default: 0.25 (Linacre, 1968).
`kd`	double, angular coefficient of transmittivity. Default: 0.50 (Linacre, 1968).
`kfFEC`	double, flux-to-energy conversion, umol/J. Default: 2.04 (Meek et al., 1984).
`kGsc`	double, solar constant, W/m^2. Default: 1360.8 (Kopp and Lean, 2011).

Value

Returns a `list` object with the following variables:

- nu_deg .... true anomaly, degrees
- lambda_deg .... true longitude, degrees
- dr .... distance factor, unitless
- delta_deg .... declination angle, degrees
- hs_deg .... sunset angle, degrees
- ra_j.m2 .... daily extraterrestrial radiation, J/m^2
- tau .... atmospheric transmittivity, unitless
- ppfd_mol.m2 .... daily photosyn. photon flux density, mol/m^2
- hn_deg .... net radiation hour angle, degrees
- rn_j.m2 .... daily net radiation, J/m^2
- rnn_j.m2 .... daily nighttime net radiation, J/m^2

References

Berger, A.L., 1978. Long-term variations of daily insolation and Quaternary climatic changes. Journal of Atmospheric Sciences, 35(12), pp.2362-2367. doi:10.1175/15200469(1978)035<2362:LTVODI>2.0.CO;2

Federer, C.A., 1968. Spatial variation of net radiation, albedo and surface temperature of forests. Journal of Applied Meteorology and Climatology, 7(5), pp.789-795. doi:10.1175/15200450(1968)007<0789:SVONRA>2.0.

Kopp, G. and Lean, J.L., 2011. A new, lower value of total solar irradiance: Evidence and climate significance. Geophys. Res. Lett. 38, L01706. doi:10.1029/2010GL045777

Linacre, E.T., 1968. Estimating the net-radiation flux. Agricultural meteorology, 5(1), pp.49-63. doi:10.1016/00021571(68)900228

Meek, D.W., Hatfield, J.L., Howell, T.A., Idso, S.B. and Reginato, R.J., 1984. A generalized relationship between photosynthetically active radiation and solar radiation 1. Agronomy journal, 76(6), pp.939-945. doi:10.2134/agronj1984.00021962007600060018x

Monteith, J., and Unsworth, M., 1990. Principles of Environmental Physics, Butterworth-Heinemann, Oxford.

Sellers, P.J., 1985. Canopy reflectance, photosynthesis and transpiration, International Journal of Remote Sensing, 6:8, 1335-1372, doi:10.1080/01431168508948283

Examples

```
solar <- splash::calc_daily_solar(lat = 37.7,
                                  n = 172,
                                  elv = 142,
                                  y = 2000,
                                  sf = 1,
                                  tc = 23.0)
cat(sprintf("Solar values:\n"))
cat(sprintf("  kn: %d\n", solar$kN))
cat(sprintf("  nu: %0.6f degrees\n", solar$nu_deg))
cat(sprintf("  lambda: %0.6f degrees\n", solar$lambda_deg))
cat(sprintf("  rho: %0.6f\n", solar$rho))
cat(sprintf("  dr: %0.6f\n", solar$dr))
cat(sprintf("  delta: %0.6f degrees\n", solar$delta_deg))
cat(sprintf("  ru: %0.6f\n", solar$ru))
cat(sprintf("  rv: %0.6f\n", solar$rv))
cat(sprintf("  rw: %0.6f\n", solar$rw))
cat(sprintf("  hs: %0.6f degrees\n", solar$hs_deg))
cat(sprintf("  hn: %0.6f degrees\n", solar$hn_deg))
cat(sprintf("  tau_o: %0.6f\n", solar$tau_o))
cat(sprintf("  tau: %0.6f\n", solar$tau))
cat(sprintf("  Qn: %0.6f mol/m^2\n", solar$ppfd_mol.m2))
cat(sprintf("  Rnl: %0.6f w/m^2\n", solar$rnl_w.m2))
cat(sprintf("  Ho: %0.6f MJ/m^2\n", (1.0e-6) * solar$ra_j.m2))
cat(sprintf("  Hn: %0.6f MJ/m^2\n", (1.0e-6) * solar$rn_j.m2))
cat(sprintf("  Hnn: %0.6f MJ/m^2\n", (1.0e-6) * solar$rnn_j.m2))
```

---

julian_day *Calculate Julian day*

Description

This function converts a date in the Gregorian calendar to a Julian day number (i.e., a method of consecutive numbering of days—does not have anything to do with the Julian calendar!)

Usage

```
julian_day(y, m, i)
```

Arguments

`y`	double, year.
`m`	double, month.
`i`	double, day of month.

Details

- valid for dates after -4712 January 1 (i.e., jde >= 0)

Value

double, Julian day.

References

Meeus, J. 1991. Chapter 7 "Julian Day". Astronomical Algorithms. Willmann-Bell.

`read_csv` *Read CSV file*

Description

Reads all three daily input variables (sf, tair, and pn) for a single year from a CSV file that includes a header.

Usage

```
read_csv(fname, y = -1)
```

Arguments

`fname`	String, file name.
`y`	Numeric, year.

Value

List with the following properties:

$file_name File name.

$sf Sunshine fraction.

$tair Air temperature.

$pn Precipitation.

$num_lines Number of data points.

$year Year of data.

`read_txt` *Read plain text file*

Description

Reads plain text file (no header) of one of the input arrays.

Usage

```
read_txt(my_data, fname, var, y = -1)
```

Arguments

`my_data`	List same as the output from `read_csv`.
`fname`	String, file name.
`var`	String, variable name.
`y`	Numeric, year.

Value

List with the following properties:

$file_name File name.

$sf Sunshine fraction.

$tair Air temperature.

$pn Precipitation.

$num_lines Number of data points.

$year Year of data.

`run_one_day` *Runs SPLASH at a single location for one day*

Description

Runs SPLASH at a single location for one day

Usage

```
run_one_day(lat, elv, n, y, wn, sf, tc, pn, kCw = 1.05, kWm = 150)
```

Arguments

`lat`	double, decimal degrees.
`elv`	double, elevation, m A.S.L. Default: 0.
`n`	double, day of year.
`y`	double, year. Default: 0.
`wn`	double, daily soil moisture content, mm (wn).
`sf`	double, fraction of sunshine hours. Default: 1.
`tc`	double, mean daily air temperature, degrees C. Default: 23.0.
`pn`	double, daily precipitation, mm/day.
`kCw`	double, supply constant, mm/hr. Default: 1.05 (Federer, 1982)
`kWm`	double, soil moisture capacity, mm. Default: 150 (Cramer-Prentice, 1988)

Value

List with the following components:

- ho .... daily solar irradiation, J/m2
- hn .... daily net radiation, J/m2
- ppfd .... daily PPFD, mol/m2
- cond .... daily condensation water, mm
- eet .... daily equilibrium ET, mm
- pet .... daily potential ET, mm
- aet .... daily actual ET, mm
- wn .... daily soil moisture, mm
- ro .... daily runoff, mm

References

Cramer, W. and Prentice, I.C., 1988. Simulation of regional soil moisture deficits on a European scale. Norsk Geografisk Tidsskrift - Norwegian Journal of Geography, 42(2-3), pp.149–151. doi:10.1080/00291958808552193

Federer, C.A., 1982. Transpirational supply and demand: plant, soil, and atmospheric effects evaluated by simulation. Water Resources Research, 18(2), pp.355-362. doi:10.1029/WR018i002p00355

Examples

```
soil <- run_one_day(lat = 37.7,
                    elv = 142,
                    n = 172,
                    y = 2000,
                    wn = 75,
                    sf = 1,
                    tc = 23,
                    pn = 5)
```

```
cat(sprintf("Soil moisture (run one day):\n"))
cat(sprintf("  Ho: %0.6f J/m2\n", soil$ho))
cat(sprintf("  Hn: %0.6f J/m2\n", soil$hn))
cat(sprintf("  PPFD: %0.6f mol/m2\n", soil$ppfd))
cat(sprintf("  EET: %0.6f mm/d\n", soil$eet))
cat(sprintf("  PET: %0.6f mm/d\n", soil$pet))
cat(sprintf("  AET: %0.6f mm/d\n", soil$aet))
cat(sprintf("  Cn: %0.6f mm/d\n", soil$cond))
cat(sprintf("  Wn: %0.6f mm\n", soil$wn))
cat(sprintf("  RO: %0.6f mm\n", soil$ro))
```

`spin_up` *Calculate daily totals*

Description

Calculate daily totals updating the soil moisture until equilibrium.

Usage

```
spin_up(mdat, dtot)
```

Arguments

`mdat`	list with meteorological data (see the details section).
`dtot`	list with daily totals (see the details section).

Details

The list with meteorological data, `mdat`, should have the following components:

- num_lines ..... double, length of meteorol. variable lists
- lat_deg .... double latitude (degrees)
- elv_m .... double, elevation (m)
- year .... double, year
- sf .... list, fraction of sunshine hours
- tair .... list, mean daily air temperature (deg. C)
- pn .... list, precipitation (mm/d)

The list with daily totals, `dtot`, should have the following component:

- wm .... list, daily soil moisture (mm)

Value

list, daily totals

Examples

```
daily_totals <- matrix(data = rep(0, 366), nrow = 366, ncol = 1)
daily_totals <- as.data.frame(daily_totals)
names(daily_totals) <- c("wn")
my_file <- system.file("extdata/example_data.csv", package = "splash")
my_data <- splash::read_csv(my_file, 2000)
my_data$lat_deg <- 37.7
my_data$elv_m <- 142
daily_totals <- splash::spin_up(my_data, daily_totals)
cat(sprintf("Spin-Up:\n"))
for (i in seq(from = 1, to = my_data$num_lines, by = 1)) {
  if (i == 1) cat(sprintf("Day\tWn (mm)\n"))
  cat(sprintf("%d\t%0.6f\n", i, daily_totals$wn[i]))
}
```

Index