Package: ascotraceR (via r-universe)

September 3, 2024

Title Simulate the Spread of Ascochyta Blight in Chickpea

Version 0.0.1.9000

DOI 10.5281/zenodo.5771290

Description A spatiotemporal model that simulates the spread of Ascochyta blight in chickpea fields based on location-specific weather conditions. This model is adapted from a model developed by Diggle et al. (2002) <doi:10.1094/PHYTO.2002.92.10.1110> for simulating the spread of anthracnose in a lupin field.

URL `https://github.com/IhsanKhaliq/ascotraceR`

BugReports `https://github.com/IhsanKhaliq/ascotraceR/issues`

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Depends R (>= 3.5.0)

Imports data.table (>= 1.13.0), lubridate (>= 1.7.9.2), lutz (>= 0.3.1), circular (>= 0.4-93), stats, sf, terra

Suggests ggplot2, knitr, rmarkdown, spelling, testthat (>= 3.0.1)

VignetteBuilder knitr

Language en-US

X-schema.org-applicationCategory Tools

X-schema.org-keywords chickpea, botanical-epidemiology, plant-disease-model, agricultural-modelling, agricultural-modeling, crop-protection, agricultural-research, model, modelling, modeling, Ascochyta-rabiei

Repository https://ihsankhaliq.r-universe.dev

RemoteUrl https://github.com/ihsankhaliq/ascotracer

RemoteRef HEAD

RemoteSha 64d2f8987333c006356b4adc1e095aefceca7467

Contents

`format_weather` *Format weather data into an an object suitable for use in ascotraceR spore dispersal models*

Description

Formats raw weather data into an object suitable for use in the `trace_asco()` function ensuring that the supplied weather data meet the requirements of the model to run.

Usage

```
format_weather(
  x,
  YYYY = NULL,
  MM = NULL,
  DD = NULL,
  hh = NULL,
  mm = NULL,
  POSIXct_time = NULL,
  time_zone = NULL,
  temp,
  rain,
  ws,
  wd,
  wd_sd,
  station,
  lon = NULL,
  lat = NULL,
  r = NULL,
  lonlat_file = NULL
)
```

Arguments

`x`	A data.frame object of weather station data for formatting. `character`.
`YYYY`	Column name or index in `x` that refers to the year when the weather was logged. `character`.
`MM`	Column name or index in `x` that refers to the month (numerical) when the weather was logged. `character`.

`DD` Column name or index in `x` that refers to the day of month when the weather was logged. `character`.

`hh` Column name or index in `x` that refers to the hour (24 hour) when the weather was logged. `character`.

`mm` Column name or index in `x` that refers to the minute when the weather was logged. `character`.

`POSIXct_time` Column name or index in `x` which contains a `POSIXct` formatted time. This can be used instead of arguments `YYYY`, `MM`, `DD`, `hh`, `mm. character`.

`time_zone` Time zone (Olsen time zone format) where the weather station is located. May be in a column or supplied as a character string. Optional, see also `r. character`. See details.

`temp` Column name or index in `x` that refers to temperature in degrees Celsius. `character`.

`rain` Column name or index in `x` that refers to rainfall in mm. `character`.

`ws` Column name or index in `x` that refers to wind speed in km / h. `character`.

`wd` Column name or index in `x` that refers to wind direction in degrees. `character`.

`wd_sd` Column name or index in `x` that refers to wind speed columns standard deviation .`character`. This is only applicable if weather data is already summarised to hourly increments. See details.

`station` Column name or index in `x` that refers to the weather station name or identifier. `character`. See details.

`lon` Column name or index in `x` that refers to weather station's longitude. `character`. See details.

`lat` Column name or index in `x` that refers to weather station's latitude. `character`. See details.

`r` Spatial raster which is intended to be used with this weather data for use in the blackspot model. Used to set `time_zone` if it is not supplied in data. `character`. Optional, see also `time_zone`.

`lonlat_file` A file path to a CSV which included station name/id and longitude and latitude coordinates if they are not supplied in the data. `character`. Optional, see also `lon` and `lat`.

Details

`time_zone` All weather stations must fall within the same time zone. If the required stations are located in differing time zones, separate `ascotraceR.weather` objects must be created for each time zone. If a raster object, `r`, of previous crops is provided that spans time zones, an error will be emitted.

`wd_sd` If weather data is provided in hourly increments, a column with the standard deviation of the wind direction over the hour is required to be provided. If the weather data are sub-hourly, the standard deviation will be calculated and returned automatically.

`lon`, `lat` and `lonlat_file` If `x` provides longitude and latitude values for station locations, these may be specified in the `lon` and `lat` columns. If the coordinates are not relevant to the study location `NA` can be specified and the function will drop these column variables. If these data are not included, (`NULL`) a separate file may be provided that contains the longitude, latitude and matching station name to provide station locations in the final `ascotraceR.weather` object that is created by specifying the file path to a CSV file using `lonlat_file`.

Value

A `ascotraceR.weather` object (an extension of **data.table**) containing the supplied weather aggregated to each hour in a suitable format for use with `trace_asco()` containing the following columns:

times:	Time in POSIXct format
rain:	Rainfall in mm
ws:	Wind speed in km / h
wd:	Wind direction in compass degrees
wd_sd:	Wind direction standard deviation in compass degrees
lon:	Station longitude in decimal degrees
lat:	Station latitude in decimal degrees
station:	Unique station identifying name
YYYY:	Year
MM:	Month
DD:	Day
hh:	Hour
mm:	Minute

Examples

```
# Weather data files for Newmarracara for testing and examples have been
# included in ascotraceR. The weather data files both are of the same format,
# so they will be combined for formatting here.

Newmarracarra <- read.csv(
 system.file("extdata",
             "1998_Newmarracarra_weather_table.csv",
             package = "ascotraceR")
)

station_data <- system.file("extdata",
                            "stat_dat.csv",
                            package = "ascotraceR")

weather <- format_weather(
  x = Newmarracarra,
  POSIXct_time = "Local.Time",
  temp = "mean_daily_temp",
  rain = "rain_mm",
  ws = "ws",
  wd = "wd",
  wd_sd = "wd_sd",
  station = "Location",
  time_zone = "Australia/Perth",
  lonlat_file = station_data
)

# Saving weather data and reimporting can lose the object class
# Reimported data can be quickly reformatted, adding the 'asco.weather' class
#  with this same function
temp_file_path <- paste0(tempdir(),"weather_file.csv")
write.csv(weather, file = temp_file_path, row.names = FALSE)
weather_imported <- read.csv(temp_file_path)
weather <- format_weather(weather_imported,
                          time_zone = "Australia/Perth")
unlink(temp_file_path) # remove temporary weather file
```

`summarise_trace` *Summarise a trace_asco output nested list as a single data.frame object*

Description

Creates a paddock-level summary **data.table** from the output of `trace_asco()` on a daily time-step where each row represents one day for the entire paddock.

Usage

```
summarise_trace(trace)

summarize_trace(trace)
```

Arguments

`trace`	a nested list output from `trace_asco()`

Value

A **data.table** summarising the model's output for a paddock on a daily time-step with the area under the disease progress curve (AUDPC) at the paddock level for the simulation's run with the following columns:

i_day:	Model iteration day (day)
new_gp:	New growing points on `i_day` (n)
susceptible_gp:	Susceptible growing points on `i_day` (n)
exposed_gp:	Exposed growing points on `i_day` (n)
i_date:	Calendar date corresponding to model's `i_day`
day:	Julian day or numeric day of year (day)
cdd:	Cumulative degree days (day)
cwh:	Cumulative wet hours (h)
cr:	Cumulative rainfall (mm)
gp_standard:	standard growing points assuming growth is not impeded by infection on `i_day` (n)
AUDPC:	Area under the disease progress curve (AUDPC) for the duration of the model's run.

See Also

`trace_asco()`, `tidy_trace()`

Examples

```
Newmarracarra <-
   read.csv(system.file("extdata",
            "1998_Newmarracarra_weather_table.csv", package = "ascotraceR"))
station_data <-
   system.file("extdata", "stat_dat.csv", package = "ascotraceR")

weather_dat <- format_weather(
   x = Newmarracarra,
   POSIXct_time = "Local.Time",
   temp = "mean_daily_temp",
   ws = "ws",
   wd_sd = "wd_sd",
   rain = "rain_mm",
   wd = "wd",
   station = "Location",
   time_zone = "Australia/Perth",
   lonlat_file = station_data)

traced <- trace_asco(
  weather = weather_dat,
  paddock_length = 100,
  paddock_width = 100,
  initial_infection = "1998-06-10",
  sowing_date = "1998-06-09",
  harvest_date = "1998-06-30",
  time_zone = "Australia/Perth",
  primary_infection_foci = "centre")

summarised <- summarise_trace(traced)



Newmarracarra <-
   read.csv(system.file("extdata",
            "1998_Newmarracarra_weather_table.csv", package = "ascotraceR"))
station_data <-
   system.file("extdata", "stat_dat.csv", package = "ascotraceR")

weather_dat <- format_weather(
   x = Newmarracarra,
   POSIXct_time = "Local.Time",
   temp = "mean_daily_temp",
   ws = "ws",
   wd_sd = "wd_sd",
   rain = "rain_mm",
   wd = "wd",
   station = "Location",
   time_zone = "Australia/Perth",
   lonlat_file = station_data)

traced <- trace_asco(
  weather = weather_dat,
  paddock_length = 100,
  paddock_width = 100,
  initial_infection = "1998-06-10",
  sowing_date = as.POSIXct("1998-06-09"),
  harvest_date = as.POSIXct("1998-06-09") + lubridate::ddays(100),
  time_zone = "Australia/Perth",
  primary_infection_foci = "centre")

summarised <- summarise_trace(traced)
```

tidy_trace *Tidy up a trace_asco output nested list*

Description

Creates a tidy **data.table** from the output of `trace_asco()`.

Usage

```
tidy_trace(trace)
```

Arguments

`trace` a nested list output from `trace_asco()`

Value

A tidy **data.table** of `trace_asco()` output.

See Also

`summarise_trace()`, `trace_asco()`

Examples

```
Newmarracarra <-
   read.csv(system.file("extdata",
            "1998_Newmarracarra_weather_table.csv", package = "ascotraceR"))
station_data <-
   system.file("extdata", "stat_dat.csv", package = "ascotraceR")

weather_dat <- format_weather(
   x = Newmarracarra,
   POSIXct_time = "Local.Time",
   temp = "mean_daily_temp",
   ws = "ws",
   wd_sd = "wd_sd",
    rain = "rain_mm",
    wd = "wd",
    station = "Location",
    time_zone = "Australia/Perth",
    lonlat_file = station_data)

traced <- trace_asco(
  weather = weather_dat,
  paddock_length = 20,
  paddock_width = 20,
  initial_infection = "1998-06-10",
  sowing_date = as.POSIXct("1998-06-09"),
  harvest_date = as.POSIXct("1998-06-09") + lubridate::ddays(100),
  time_zone = "Australia/Perth",
  primary_infection_foci = "centre")

tidied <- tidy_trace(traced)

# take a look at the infectious growing points on day 102
library(ggplot2)
ggplot(data = subset(tidied, i_day == 102),
       aes(x = x, y = y, fill = infectious_gp)) +
  geom_tile()
```

trace_asco *Simulates the spread of Ascochyta blight in a chickpea field*

Description

Simulate the spatiotemporal development of Ascochyta blight in a chickpea paddock over a growing season. Both host and pathogen activities are simulated in one square metre cells.

Usage

```
trace_asco(
  weather,
  paddock_length,
  paddock_width,
  sowing_date,
  harvest_date,
  initial_infection,
  seeding_rate = 40,
  gp_rr = 0.0065,
  max_gp_lim = 5000,
  max_new_gp = 350,
  latent_period_cdd = 150,
  time_zone = "UTC",
  primary_infection_foci = "random",
  primary_inoculum_intensity = 1,
  n_foci = 1,
  spores_per_gp_per_wet_hour = 0.22,
  splash_cauchy_parameter = 0.5,
  wind_cauchy_multiplier = 0.015,
  daily_rain_threshold = 2,
  hourly_rain_threshold = 0.1,
  susceptible_days = 2,
  rainfall_multiplier = FALSE
)
```

Arguments

`weather` weather data for a representative chickpea paddock for a complete chickpea growing season for the model's operation.

`paddock_length` length of a paddock in metres (y).

`paddock_width` width of a paddock in metres (x).

`sowing_date` a character string of a date value indicating sowing date of chickpea seed and the start of the 'ascotraceR' model. Preferably in ISO8601 format (YYYY-MM-DD), *e.g.* "2020-04-26". Assumes there is sufficient soil moisture to induce germination and start the crop growing season.

`harvest_date` a character string of a date value indicating harvest date of chickpea crop, which is also the last day to run the 'ascotraceR' model. Preferably in ISO8601 format (YYYY-MM-DD), *e.g.*, "2020-04-26".

`initial_infection`
a character string of a date value referring to the initial or primary infection on seedlings, resulting in the production of infectious growing points.

`seeding_rate` indicate the rate at which chickpea seed is sown per square metre. Defaults to `40`.

`gp_rr` refers to rate of increase in chickpea growing points per degree Celsius per day. Defaults to `0.0065`.

`max_gp_lim` maximum number of chickpea growing points (meristems) allowed per square metre. Defaults to `5000`.

`max_new_gp` Maximum number of new chickpea growing points (meristems), which develop per day, per square metre. Defaults to `350`.

`latent_period_cdd`
latent period in cumulative degree days (sum of daily temperature means) is the period between infection and production of lesions on susceptible growing points. Defaults to `150`.

`time_zone` refers to time in Coordinated Universal Time (UTC).

`primary_infection_foci`
refers to the inoculated coordinates where the infection starts. Accepted inputs are: `centre`/`center` or `random` (Default) or a `data.frame` with column names 'x', 'y' and 'load'. The `data.frame` inputs inform the model of specific grid cell/s coordinates where the epidemic should begin. The 'load' column is optional and can specify the `primary_inoculum_intensity` for each coordinate.

`primary_inoculum_intensity`
: Refers to the amount of primary infection as lesions on chickpea plants at the time of `initial_infection`. On the date of initial infection in the experiment. The sources of primary inoculum can be infected seed, volunteer chickpea plants or infested stubble from the previous seasons. Defaults to `1`.

`n_foci`
: Quantifies the number of primary infection foci. The value is 1 when `primary_infection_foci = "centre"` and can be greater than 1 if `primary_infection_foci = "random`.

`spores_per_gp_per_wet_hour`
: number of spores produced per infectious growing point during each wet hour. Also known as the `spore_rate`. Value is dependent on the susceptibility of the host genotype.

`splash_cauchy_parameter`
: a parameter used in the Cauchy distribution and describes the median distance spores travel due to rain splashes. Default to `0.5`.

`wind_cauchy_multiplier`
: a scaling parameter to estimate a Cauchy distribution which resembles the possible distances a conidium travels due to wind driven rain. Defaults to `0.015`.

`daily_rain_threshold`
: minimum cumulative rainfall required in a day to allow hourly spore spread events. See also `hourly_rain_threshold`. Defaults to `2`.

`hourly_rain_threshold`
: minimum rainfall in an hour to trigger a spore spread event in the same hour (assuming daily_rain_threshold is already met). Defaults to `0.1`.

`susceptible_days`
: the number of days for which conidia remain viable on chickpea after dispersal. Defaults to `2`. Conidia remain viable on the plant for at least 48 hours after a spread event

`rainfall_multiplier`
: logical values will turn on or off rainfall multiplier default method. The default method increases the number of spores spread per growing point if the rainfall in the spore spread event hour is greater than one. Numeric values will scale the number of spores spread per growing point against the volume of rainfall in the hour. Defaults to `FALSE`.

Value

a nested `list` object where each sub-list contains daily data for the day `i_day` (the model's iteration day) generated by the model including: * **paddock**, an 'x' 'y' **data.table** containing: * **x**, location of quadrat on x-axis in paddock, * **y**, location of quadrat on y-axis in paddock, * **new_gp**, new growing points produced in the last 24 hours, * **susceptible_gp**, susceptible growing points in the last 24 hours, * **exposed_gp**, exposed growing points in the last 24 hours, * **infectious_gp**, infectious growing points in the last 24 hours,

- **i_day**, model iteration day, * **cumulative daily weather data**, a **data.table** containing: * **cdd**, cumulative degree days, * **cwh**, cumulative wet hours, * **cr**, cumulative rainfall in mm, * **gp_standard**, standard growing points assuming growth is not impeded by infection, * **infected_coords**, a **data.table** of only infectious growing point coordinates, * **new_infections**, a **data.table** of newly infected growing points, * **exposed_gps**, a **data.table** of exposed growing points in the latent period phase of infection.

See Also

tidy_trace(), summarise_trace()

Examples

```
# First weather data needs to be imported and formatted with `format_weather`
Newmarracarra <-
   read.csv(system.file("extdata",
            "1998_Newmarracarra_weather_table.csv", package = "ascotraceR"))
station_data <-
   system.file("extdata", "stat_dat.csv", package = "ascotraceR")

weather_dat <- format_weather(
   x = Newmarracarra,
   POSIXct_time = "Local.Time",
   temp = "mean_daily_temp",
   ws = "ws",
   wd_sd = "wd_sd",
   rain = "rain_mm",
   wd = "wd",
   station = "Location",
   time_zone = "Australia/Perth",
   lonlat_file = station_data)

# Now the `trace_asco` function can be run to simulate disease spread
traced <- trace_asco(
  weather = weather_dat,
  paddock_length = 100,
  paddock_width = 100,
  initial_infection = "1998-06-10",
  sowing_date = "1998-06-09",
  harvest_date = "1998-06-30",
  time_zone = "Australia/Perth",
  gp_rr = 0.0065,
  primary_inoculum_intensity = 40,
  spores_per_gp_per_wet_hour = 0.22,
  primary_infection_foci = "centre")

traced[[23]] # extracts the model output for day 23
```

Index