Package ‘dfoliatR’

September 2, 2020

Type Package

Title Detection and Analysis of Insect Defoliation Signals in Tree Rings

Version 0.2.0

Description Tools to identify, quantify, analyze, and visualize growth suppression events in tree rings that are often produced by insect defoliation. Described in Guiterman et al. (2020) <doi:10.1016/j.dendro.2020.125750>.

URL https://chguiterman.github.io/dfoliatR/

BugReports https://github.com/chguiterman/dfoliatR/issues

Depends R (>= 3.2)

License GPL (>= 3)

Suggests testthat, vdiffr, knitr, rmarkdown

Imports dplR, dplyr, forcats, ggplot2, ggpubr, magrittr, MASS, plyr, purrr, rlang, stats, tidyr

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Chris Guiterman [aut, cre] (<https://orcid.org/0000-0002-9706-9332>), Ann Lynch [aut] (<https://orcid.org/0000-0002-8043-3855>), Jodi Axelson [aut] (<https://orcid.org/0000-0002-1799-6364>)

Maintainer Chris Guiterman <chguiterman@email.arizona.edu>

Repository CRAN

Date/Publication 2020-09-02 12:30:06 UTC
R topics documented:

as.defol ........................................ 2
as.obr ........................................... 3
as_defol ........................................ 4
as_obr ........................................... 4
defol ............................................ 5
defoliate_trees ................................ 6
defol_stats ...................................... 7
dmj_defol ....................................... 8
dmj_h ............................................. 8
dmj_nh .......................................... 9
dmj_obr .......................................... 9
efk_defol ....................................... 10
efk_h ............................................ 10
efk_nh .......................................... 11
efk_obr .......................................... 11
ef_defol ......................................... 12
ef_h ............................................. 12
ef_nh .......................................... 12
ef_obr .......................................... 13
get_defol_events ................................. 13
gsi ............................................... 13
id_defoliation .................................. 14
is.defol ........................................ 15
is.obr .......................................... 16
obr ............................................... 16
outbreak ........................................ 17
outbreak_stats .................................. 18
plot.defol ...................................... 19
plot_defol ...................................... 20
plot_outbreak ................................... 20
sample_depth .................................... 21
series_names .................................... 21
stack_defoliation ............................... 22

Index 23

as.defol  Alias to as.defol()

Description

Alias to as.defol()

Usage

as.defol(x)
Arguments

x  A data frame or list-like object to cast. Must have named elements for "year", "series", "gsi", "ngsi", and "defol_status".

Value

x cast to a defol object

Examples

data(dmj_defol)
example_data <- as.data.frame(dmj_defol)
is.defol(example_data)
back_to_defol <- as_defol(example_data)
is.defol(back_to_defol)

as.obr

Alias to as_obr()

Description

Alias to as_obr()

Usage

as.obr(x)

Arguments

x  A data frame or list-like object to cast. Must have named elements for "year", "samp_depth", "num_defol", "perc_defol", "num_max_defol", "perc_max_defol", "mean_gsi", "mean_ngsi", "outbreak_status".

Value

x cast to an obr object

Examples

data(dmj_obr)
example_data <- as.data.frame(dmj_obr)
is.obr(example_data)
back_to_obr <- as.obr(example_data)
is.obr(back_to_obr)
as_defol

Cast data frame to list-like defol object

Description

Cast data frame to list-like defol object

Usage

as_defol(x)

Arguments

x

A data frame or list-like object to cast. Must have named elements for "year", "series", "gsi", "ngsi", and "defol_status".

Value

x cast to a defol object

Examples

data(dmj_defol)
example_data <- as.data.frame(dmj_defol)
is.defol(example_data)
back_to_defol <- as_defol(example_data)
is.defol(back_to_defol)

as_obr

Cast data frame to list-like obr object

Description

Cast data frame to list-like obr object

Usage

as_obr(x)

Arguments

x

A data frame or list-like object to cast. Must have named elements for "year", "samp_depth", "num_defol", "perc_defol", "num_max_defol", "perc_max_defol", "mean_gsi", "mean_ngsi", "outbreak_status".
defol

Value

x cast to an obr object

Examples

data(dmj_obr)
example_data <- as.data.frame(dmj_obr)
is.obr(example_data)
back_to_obr <- as_obr(example_data)
is.obr(back_to_obr)

defol

Constructor for S3 defol class

Description

Constructor for S3 defol class

Usage

defol(year, series, gsi, ngsi, defol_status)

Arguments

defol_status

An n-length factor or character vector denoting the defoliation event status of each year. This uses a controlled vocabulary, see dfoliatR::make_defol_status for possible values.

Value

a tree-level defol object
defoliate_trees

Identify defoliation events in host trees

Description

defoliate_trees() is the starting point for most analyses of insect defoliation signals preserved in the growth patterns of trees. It requires individual-tree standardized measurements from potential host trees and a tree-ring chronology from a nearby non-host species. First, defoliate_trees() combines these tree-ring indices by calling gsi() to perform a “correction” of the host-tree indices to remove the climatic influences on tree growth as represented by the non-host chronology. This should isolate a disturbance-related signal. Second, defoliate_trees(), runs id_defoliation(), which completes a runs analyses to evaluate sequences of negative departures in the host tree growth series (ngsi) for potential defoliation events.

Usage

defoliate_trees(
  host_tree,
  nonhost_chron,
  duration_years = 8,
  max_reduction = -1.28,
  bridge_events = FALSE,
  series_end_event = FALSE,
  list_output = FALSE
)

Arguments

defoliate_trees (host_tree, nonhost_chron, duration_years = 8, max_reduction = -1.28, bridge_events = FALSE, series_end_event = FALSE, list_output = FALSE)

host_tree A dplR::rwl object containing the tree-level growth series for all host trees to be compared to the non-host chronology.
nonhost_chron A dplR::rwl object containing a single non-host chronology.
duration_years The minimum length of time in which the tree is considered to be in defoliation.
max_reduction The minimum value of ngsi required to be considered a defoliation event. If a sequence of negative ngsi values does not reach this threshold, the potential event is rejected. Defaults to -1.28.
bridge_events Binary, defaults to FALSE. This option allows for successive events that are separated by a single year to be bridged and become one event. It should be used cautiously and closely evaluated to ensure the likelihood that the two (or more) events are actually one long event.
series_end_event Binary, defaults to FALSE. This option allows the user to identify an event occurring at the time of sampling as a defoliation event, regardless of duration. Including it will help to quantify periodicity and extent of an outbreak. This should only be used if the user has direct knowledge of an ongoing defoliation event when the trees were sampled.
defol_stats

- list_output: Defaults to FALSE. This option is to output a long list object containing a separate data frame for each series in host_tree that includes the input series and the nonhost_chron, the corrected series, and the character string identifying the defoliation events.

**Value**

By default this returns a long-form data frame of tree-level growth suppression indices and identified defoliation events. If list_output = TRUE, it returns a list object with each element containing a data.frame rwl object of the host and non-host series, plus the outputs from gsi(). The list object is useful for assessing the effects of running gsi() on the host and nonhost data.

**Note**

Other functions in dfoliatR, like outbreak() and plot_defol(), require a long-form data frame identifiable as a defol() object. Selecting list_output = TRUE will trigger errors in running other functions.

**Examples**

```r
# Load host and non-host data
data("dmj_h") # Host trees
data("dmj_nh") # Non-host chronology
dmj_defol <- defoliate_trees(dmj_h, dmj_nh)
```

---

defol_stats

*Descriptive statistics for defoliation trees*

**Description**

Descriptive statistics for defoliation trees

**Usage**

```r
defol_stats(x)
```

**Arguments**

- `x`: A defol object after running defoliate_trees().

**Value**

A data frame containing tree/series-level statistics.
Note

If series-end-events are present, they are omitted from calculations of total event years and mean duration.

Examples

data("dmj_defol")
defol_stats(dmj_defol)

---

 dmj_defol  Demi John dfol object

Description

Produced by running defoliate_trees(dmj_h,dmj_nh,series_end_events=TRUE)

Usage

dmj_defol

Format

A defol object with 4267 rows and 5 columns

---

 dmj_h  Demi John Douglas-fir

Description

Western spruce budworm host series

Usage

dmj_h

Format

An rwl object with 17 tree-level series, standardized in ARSTAN. Dates range from 1620-1997.
**dmj_nh**

---

**dmj_nh**  
*Demi John area ponderosa pine*

---

**Description**

Non-host pair to Demi John Douglas-fir dmj_h

**Usage**

```r
dmj_nh
```

**Format**

An `rwl` object with 1 series, 1675-1997.

---

**dmj_obr**  
*Demi John outbreak object*

---

**Description**

Produced by running `outbreak(dmj_defol)`

**Usage**

```r
dmj_obr
```

**Format**

An object of class `obr` (inherits from `data.frame`) with 323 rows and 9 columns.

**Details**

@format An outbreak object with 323 rows and 9 columns
Description

Produced by running `defoliate_trees(efk_h, efk_nh, series_end_event = TRUE)`

Usage

```
efk_defol
```

Format

A `defol` object with 5142 rows and 5 columns

---

**efk_h**

*East Fork Jemez River Douglas-fir*

Description

Western spruce budworm host series

Usage

```
efk_h
```

Format

An `rwl` object with 37 tree-level series, standardized in ARSTAN. Dates range from 1776-1987.

Source

Description

Non-host pair chronology for East Fork Douglas-fir efk_nh

Usage

efk_nh

Format

An rwl chronology object with 1 series, 1612-1987. Standardized in ARSTAN.

Source


Description

Produced by running outbreak(efk_defol)

Usage

efk_obr

Format

An outbreak object with 221 rows and 9 columns
**ef_defol**  
*East Fork defol object*

**Description**
Produced by running `defoliate_trees(ef_h, ef_nh, series_end_event = TRUE)`

**Usage**
`ef_defol`

**Format**
A `defol` object with 2500 rows and 5 columns

---

**ef_h**  
*East Fork Jemez River Douglas-fir*

**Description**
Western spruce budworm host series

**Usage**
`ef_h`

**Format**

---

**ef_nh**  
*East Fork Jemez area ponderosa pine*

**Description**
Non-host pair chronology for East Fork Douglas-fir `ef_h`

**Usage**
`ef_nh`

**Format**
An `rwl` chronology object with 1 series, 1612-1987. Standardized in ARSTAN.
**ef_obr**

**Description**
Produced by running `outbreak(ef_defol)`

**Usage**
`ef_obr`

**Format**
An outbreak object with 221 rows and 9 columns

---

**get_defol_events**

**Description**
Defoliation event list

**Usage**
`get_defol_events(x)`

**Arguments**

- `x` a `defol` object

---

**gsi**

**Description**
This function removes the nonhost growth signal from a host tree-ring series.

**Usage**
`gsi(input_series)`
**Arguments**

`input_series` A `dplyr::rwl` object with the host tree series as the first column and the non-host chronology as the second. Years should be the row names. This is specifically created by `defoliate_trees()` and passed to `gsi()`.

**Details**

The growth suppression index (GSI) is referred to as the "corrected" series in OUTBREAK. It is calculated as:

\[
GSI(i) = H(i) - (NH(i) - mean(NH))sd(H)/sd(NH)
\]

where H and NH are the host and nonhost tree-ring series as standardized index values; i is the year, and the functions `mean()` and `sd()` are applied to the common period.

`gsi()` will rarely be run directly by a user. It is called from `defoliate_trees()`.

**Value**

A data frame with the two input columns (host and nonhost series) and 3 added columns:

1. The mean/sd adjusted non-host chronology,
2. The growth suppression index ("gsi") of the host series after subtraction of the adjusted non-host chronology,
3. The normalized growth suppression index ("ngsi") generated by applying `scale()` to the gsi.

**Usage**

```r
id_defoliation(
  input_series,
  duration_years = 8,
  max_reduction = -1.28,
  bridge_events = FALSE,
  series_end_event = FALSE
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input_series</td>
<td>A data frame with 5 columns, as generated by gsi().</td>
</tr>
<tr>
<td>duration_years</td>
<td>The minimum length of time in which the tree is considered to be in defoliation.</td>
</tr>
<tr>
<td>max_reduction</td>
<td>The minimum value of ngsi required to be considered a defoliation event. If a sequence of negative ngsi values does not reach this threshold, the potential event is rejected. Defaults to -1.28.</td>
</tr>
<tr>
<td>bridge_events</td>
<td>Binary, defaults to FALSE. This option allows for successive events that are separated by a single year to be bridged and become one event. It should be used cautiously and closely evaluated to ensure the likelihood that the two (or more) events are actually one long event.</td>
</tr>
<tr>
<td>series_end_event</td>
<td>Binary, defaults to FALSE. This option allows the user to identify an event occurring at the time of sampling as a defoliation event, regardless of duration. Including it will help to quantify periodicity and extent of an outbreak. This should only be used if the user has direct knowledge of an ongoing defoliation event when the trees were sampled.</td>
</tr>
</tbody>
</table>

Value

After performing runs analyses, the function adds a column to the input data frame that distinguishes years of defoliation and the maximum defoliation year (ie. the year the greatest negative growth departure within the event).

Note

`id_defoliation()` is called by `defoliate_trees()`. It might only be used by the user for troubleshooting.

---

is.defol

Check if object is tree-level defoliation object: defol

Description

Check if object is tree-level defoliation object: defol

Usage

is.defol(x)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Any R object.</td>
</tr>
</tbody>
</table>

Value

Boolean indicating whether x is a defol object.
is.obr  
Check if object is outbreak, meaning site-level outbreak object

Description

Check if object is outbreak, meaning site-level outbreak object

Usage

is.obr(x)

Arguments

x  
Any R object.

Value

Boolean indicating whether x is an outbreak object.

obr  
Constructor for an obr object.

Description

Constructor for an obr object.

Usage

obr(
  year,
  samp_depth,
  num_defol,
  perc_defol,
  num_max_defol,
  perc_max_defol,
  mean_gsi,
  mean_ngsi,
  outbreak_status
)
outbreak

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>year</td>
<td>An n-length numeric vector of observed years.</td>
</tr>
<tr>
<td>samp_depth</td>
<td>An n-length numeric vector of the number of trees.</td>
</tr>
<tr>
<td>num_defol</td>
<td>An n-length numeric vector of the number of trees experiencing defoliation.</td>
</tr>
<tr>
<td>perc_defol</td>
<td>An n-length numeric vector of the percent of trees experiencing defoliation.</td>
</tr>
<tr>
<td>num_max_defol</td>
<td>An n-length numeric vector of the number of trees experiencing their maximum level of defoliation (i.e., their most extreme negative growth departure).</td>
</tr>
<tr>
<td>perc_max_defol</td>
<td>An n-length numeric vector of the percent of trees experiencing their maximum level of defoliation (i.e., their most extreme negative growth departure).</td>
</tr>
<tr>
<td>mean_gsi</td>
<td>An n-length numeric vector of the average growth suppression index across all observed trees.</td>
</tr>
<tr>
<td>mean_ngsi</td>
<td>An n-length numeric vector of the average normalized (scaled) growth suppression index.</td>
</tr>
<tr>
<td>outbreak_status</td>
<td>An n-length factor or character vector that identified whether that year surpasses the designated thresholds for an &quot;outbreak event&quot;. Threshold criteria are provided in outbreak().</td>
</tr>
</tbody>
</table>

Value

An obr object with columns matching the input variables.

Outbreak: Composite defoliation series to determine outbreak events

Description

outbreak() takes a defol object from defoliate_trees() and composites it into a site-level object. Function parameters allow the user to filter the tree-level series in various ways to optimize thresholds of what constitutes an "outbreak" level event recorded by the host trees.

Usage

outbreak(x, filter_perc = 25, filter_min_series = 3, filter_min_defol = 1)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>a defol object</td>
</tr>
<tr>
<td>filter_perc</td>
<td>the minimum percentage of defoliated trees to be considered an outbreak. Default is 25 percent.</td>
</tr>
<tr>
<td>filter_min_series</td>
<td>The minimum number of trees required for an outbreak event. Default is 3 trees.</td>
</tr>
<tr>
<td>filter_min_defol</td>
<td>The minimum number of trees recording a defoliation event. Default is 1 tree.</td>
</tr>
</tbody>
</table>
outbreak_stats

Value
A data.frame obr object for the site that includes all trees in the host defol object. Columns in the obr include:

1. `year` for every year in the set of host trees,
2. `num_defol` the number of trees recording a defoliation event,
3. `percent_defol` the percent of trees recording a defoliation,
4. `num_max_defol` the number of trees recording a maximum growth suppression (or peak of that event on that tree),
5. `perc_max_defol` the percent of trees at maximum defoliation,
6. `mean_gsi` the average of all trees growth suppression index (gsi),
7. `mean_ngsi` the average of all trees normalized growth suppression index (ngsi),
8. `outbreak_status` whether that year constitutes an outbreak based on the filters applied to the function.

Examples
```
data("dmj_defol")
head(outbreak(dmj_defol))
```

<table>
<thead>
<tr>
<th>outbreak_stats</th>
<th>Outbreak statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description
Summary statistics for inferred outbreaks

Usage
```
outbreak_stats(x)
```

Arguments

- `x` An obr object after running outbreak()

Value
A data frame with descriptive statistics for each outbreak event determined by outbreak(), including:

- "start" – first year of outbreak
- "end" – last year of outbreak
- "duration" – length of outbreak (in years)
- "n_df_start" – number of trees defoliated at the start
• "perc_df_start" – percent of trees defoliated at the start
• "max_df_obr" – maximum number of trees in the outbreak during a single year
• "yr_max_df" – year with the maximum number of trees defoliated
• "yr_min_ngsi" – year with the lowest mean normalized growth suppression index (NGSI)
• "min_gsi" – minimum growth suppression index
• "min_ngsi" – minimum normalized gsi

Note

Certain statistics will be set to NA for the final outbreak event if there was an ongoing defoliation event (in which series_end_event = TRUE in defoliate_trees()). This is because the end of the outbreak remains unknown, so statistics such as duration cannot be calculated. Statistics pertaining to the start of the event are provided.

Examples

data("dmj_obr")
outbreak.stats(dmj_obr)

plot.defol

Plot a defol object

Description

Plot a defol object

Usage

## S3 method for class 'defol'
plot(...)

Arguments

... arguments passed to plot_defol()
**plot_defol** *Produce a Gantt plot of individual tree-ring series to show defoliation events in time*

**Description**

Produce a Gantt plot of individual tree-ring series to show defoliation events in time

**Usage**

`plot_defol(x, breaks)`

**Arguments**

- `x` a defol object produced by `defoliate_trees()`.
- `breaks` a vector length two providing threshold (negative) ngsi values to separate minor, moderate, and severe defoliation events. If blank, the mean and 1st quartile are used.

**Examples**

```r
data("dmj_defol")
plot_defol(dmj_defol)

## Change the values severity classes
plot_defol(dmj_defol, breaks = c(-1.0, -0.5))
```

**plot_outbreak** *Produce a stacked plot to present composited, site-level insect outbreak chronologies*

**Description**

Produce a stacked plot to present composited, site-level insect outbreak chronologies

**Usage**

`plot_outbreak(x, disp_index = c("GSI", "NGSI"))`

**Arguments**

- `x` an 'obr' object produced by `outbreak()`
- `disp_index` Identify the timeseries index to plot. Defaults to NGSI, the average normalized growth suppression index for the site. The only other option is GSI, the average growth suppression index.
sample_depth

Examples

```r
data(dmj_obr)
plot_outbreak(dmj_obr)

# Change middle panel display from the default "NGSI" to "GSI"
plot_outbreak(dmj_obr, disp_index = "GSI")
```

Description

Calculate the sample depth of a defol object

Usage

```
sample_depth(x)
```

Arguments

- `x`: A defol object.

Value

A data.frame containing the years and number of trees

Examples

```r
data("dmj_defol")
head(sample_depth(dmj_defol))
```

series_names

Extract series names from a defol object

Description

Extract series names from a defol object

Usage

```
series_names(x)
```

Arguments

- `x`: a defol object
stack_defoliation

Stack a defoliation list

Description
Stack a defoliation list

Usage
stack_defoliation(x)

Arguments
x a list object created by defoliate_trees()

Value
a defol object (long-format data frame)
## Index

* **datasets**
  - dmj_defol, 8
  - dmj_h, 8
  - dmj_nh, 9
  - dmj_obr, 9
  - ef_defol, 12
  - ef_h, 12
  - ef_nh, 12
  - ef_obr, 13
  - efk_defol, 10
  - efk_h, 10
  - efk_nh, 11
  - efk_obr, 11
  - as_defol, 2
  - as_obr, 3
  - as_defol, 4
  - as_defol(), 2
  - as_obr, 4
  - as_obr(), 3
  - defol, 5
  - defol(), 7
  - defol_stats, 7
  - defoliate_trees, 6
  - defoliate_trees(), 6, 7, 14, 15, 17, 19, 20, 22
  - dmj_defol, 8
  - dmj_h, 8
  - dmj_nh, 9
  - dmj_obr, 9
  - ef_defol, 12
  - ef_h, 12
  - ef_nh, 12
  - ef_obr, 13
  - efk_defol, 10
  - efk_h, 10
  - efk_nh, 11
  - efk_obr, 11

  - get_defol_events, 13
  - gsi, 13
  - gsi(), 5–7, 14, 15
  - id_defoliation, 14
  - id_defoliation(), 6, 14, 15
  - is_defol, 15
  - is_obr, 16
  - mean(), 14
  - obr, 16, 18
  - outbreak, 17
  - outbreak(), 7, 17, 18, 20
  - outbreak_stats, 18
  - plot_defol, 19
  - plot_defol(), 19
  - plot_outbreak, 20
  - sample_depth, 21
  - scale(), 14
  - sd(), 14
  - series_names, 21
  - stack_defoliation, 22