Package ‘litterfitter’

May 8, 2026

Type Package

Title Fits a Collection of Curves to Single-Cohort Decomposition Data

Version 0.1.4

Date 2025-08-19

Description Fit different model forms to single-cohort litter decomposition data (mass remaining through time) using likelihood-based estimation. Models span simple empirical to process-motivated forms with differing numbers of free parameters. Provides parameter estimates, uncertainty, and tools for model comparison/selection. Based on Cornwell & Weedon (2013) <doi:10.1111/2041-210X.12138>.

Depends R (>= 3.5.0)

LazyData true

LazyDataCompression xz

Encoding UTF-8

Imports stats, graphics, grDevices, methods

Suggests testthat, knitr, roxygen2, devtools, rmarkdown

License MIT + file LICENSE

URL `http://traitecoevo.github.io/litterfitter/`, `https://github.com/traitecoevo/litterfitter`

RoxygenNote 7.3.2

VignetteBuilder knitr

BugReports `https://github.com/traitecoevo/litterfitter/issues`

Config/testthat/edition 3

NeedsCompilation no

Author Will Cornwell [aut, cre] (ORCID: <`https://orcid.org/0000-0003-4080-4073`>),
James Weedon [aut],
Guofang Liu [ctb]

Maintainer Will Cornwell <`wcornwell@gmail.com`>

Repository CRAN

Date/Publication 2025-08-21 08:20:09 UTC

Contents

`bootstrap_parameters` *Create a bootstrap distribution of a particular coefficient from a model fit*

Description

Create a bootstrap distribution of a particular coefficient from a model fit

Usage

```
bootstrap_parameters(x,nboot,upper,lower,...)
```

Arguments

`x`	an object of class "`litfit`"
`nboot`	number of bootstrap replications
`upper`	optional vector of upper bounds for the bootstrap replicates
`lower`	optional vector of lower bounds for the bootstrap replicates
`...`	passed to `optim`

Value

litfit_bootstrap object

Examples

```
fit <- fit_litter(time=pineneedles$Year,
mass.remaining=pineneedles$Mass.remaining, model='neg.exp', iters=100)
boot1 <- bootstrap_parameters(fit, nboot = 500)
```

`fit_litter` *Fit different models to single cohort decomposition data*

Description

Non-linear fits of different models to the decomposition trajectory of one cohort (as in typical litterbag studies) data. Models range from very simple (and easy to fit with limited data) to more complex.

Usage

```
fit_litter(
  time,
  mass.remaining,
 model = c("neg.exp", "weibull", "discrete.parallel", "discrete.series", "cont.quality",
   "neg.exp.limit"),
  iters = 500,
  upper = NULL,
  lower = NULL,
  ...
)
```

Arguments

`time`	time since decomposition began, that is, ti-t0
`mass.remaining`	proportional mass loss, that is, mi/m0
`model`	there are five models currently implemented (see below)
`iters`	Number of random starts for the fitting. Use higher numbers for models with larger numbers of parameters and for models that inherently tend to be less identifiable.
`upper`, `lower`	**Optional** user specified values for the upper and lower bounds used by `optim` in the fitting procedure. Use with care, only minimal sanity checking is currently implemented.
`...`	Additional arguments passed to `optim`

Details

the model likelihood is maximized using methods available in `optim`. Optimization methods to be used within optim can be specified through the control object (i.e., control$method). The default method is L-BFGS-B with bounds specific to each model. Each model

- **weibull** The Weibull residence time model–two parameters (Frechet 1927)
- **discrete.parallel** Two pools in parallel with a term for the fraction of initial mass in each pool–three parameter (Manzoni et al. 2012)
- **discrete.series** A three parameter model in which there is the possibility of two sequential pools (Manzoni et al. 2012)

- **cont.quality** (Ågren and Bosatta 1996, see also Manzoni et al. 2012)

Warning: difficulty in finding the optimal solution is determined by an interaction between the nature and complexity of the likelihood space (which is both data- and model-dependent) as well as the optimization methods. There is can never be a guarantee that the optimal solution is found, but using many random starting points will increase these odds. It should be noted that there is significant variation among models in identifiability, with some models inherently less identifiable likely due to a tendency to form for flat ridges in likelihood space. The confidence in the fit should be very low in these cases (see Cornwell and Weedon 2013). A number of random starting points are used in optimization and are given through the iters. The function checks whether the the top 10 optimizations have converged on the same likelihood, and if they have not this function will return a warning.

Value

returns a litfit object with the following elements:

- **optimFit:** a list generated by the `optim` function
- **logLik:** the log-likelihood of the model
- **time:** vector of time (same as input)
- **mass:** vector os mass remaining (same as input)
- **predicted:** predicted values from the model for each of the points within time
- **model:** name of the model
- **nparams:** number of fit parameters in the model
- **AIC:** AIC of the model fit
- **AICc:** AICc of the model fit
- **BIC:** BIC of the model fit
- and some other potentially useful things

Author(s)

Will Cornwell and James Weedon

References

- Ågren, G. and Bosatta, E. (1996) Quality: a bridge between theory and experiment in soil organic matter studies. Oikos, 76, 522–528.
- Cornwell, W. K., and J. T. Weedon. (2013). Decomposition trajectories of diverse litter types: a model selection analysis. Methods in Ecology and Evolution.
- Frechet, M. (1927) Sur la loi de probabilite de lecart maximum. Ann de la Soc polonaise de Math, 6, 93–116.
- Manzoni, S., Pineiro, G., Jackson, R. B., Jobbagy, E. G., Kim, J. H., & Porporato, A. (2012). Analytical models of soil and litter decomposition: Solutions for mass loss and time-dependent decay rates. Soil Biology and Biochemistry, 50, 66-76.
- Olson, J.S. (1963) Energy storage and the balance of producers and decomposers in ecological systems. Ecology, 44, 322–331.

See Also

`optim`, `steady_state`, `plot.litfit`

Examples

```
data(pineneedles)
fit<-fit_litter(time=pineneedles$Year,mass.remaining=pineneedles$Mass.remaining,
model='neg.exp',iters=1000)
```

`litterfitter` *litterfitter: methods for fitting curves to litter decomposition trajectories*

Description

There is a long tradition of studying the flux of carbon from the biosphere to the atmosphere by following a particular cohort of litter (wood, leaves, roots, or other organic material) through time. The resulting data are mass remaining and time. A variety of functional forms may be used to fit the resulting data. Some work better empirically. Some are better connected to a process-based understanding. Some have a small number of free parameters; others have more. This package matches decomposition data to a set of these curves using likelihood–based fitting.

Author(s)

Maintainer: Will Cornwell <`wcornwell@gmail.com`> (ORCID)

Authors:

- James Weedon

Other contributors:

- Guofang Liu [contributor]

References

Cornwell, W. K., and J. T. Weedon. (2013). Decomposition trajectories of diverse litter types: a model selection analysis. Methods in Ecology and Evolution.

Manzoni, S., Pineiro, G., Jackson, R. B., Jobbagy, E. G., Kim, J. H., & Porporato, A. (2012). Analytical models of soil and litter decomposition: Solutions for mass loss and time-dependent decay rates. Soil Biology and Biochemistry, 50, 66-76.

See Also

Useful links:

- `http://traitecoevo.github.io/litterfitter/`
- `https://github.com/traitecoevo/litterfitter`
- Report bugs at `https://github.com/traitecoevo/litterfitter/issues`

`pineneedles` *decomposition trajectory for pine needles*

Description

data from Hobbie et al means of pine needle decomposition

Usage

```
data(pineneedles)
```

Format

a data.frame with two columns

Author(s)

Will Cornwell

Source

Hobbie et al.

`plot.litfit` *Plot Decomposition Trajectory and Curve Fit*

Description

Plot a Litter Decomposition Trajectory with Curve Fit

Usage

```
## S3 method for class 'litfit'
plot(x, formulae.cex = 1, ...)
```

Arguments

`x`	A `litfit` object.
`formulae.cex`	Size scaling factor for the formula display on the plot.
`...`	Additional arguments passed to `plot.default`.

Details

Visualizes the litter decomposition trajectory data and its curve fit derived from a `litfit` object. This function is designed to provide a quick visual check on the adequacy of model fitting.

The plot displays data points from the `litfit` object along with the curve fit. The formula for the fit is displayed on the plot.

Value

A plot visualizing the data and curve fit from a `litfit` object. The result is returned invisibly.

Author(s)

Will Cornwell

See Also

`fit_litter` for generating `litfit` objects.

Examples

```
fit <- fit_litter(
  time=c(0,1,2,3,4,5,6),
  mass.remaining=c(1,0.9,1.01,0.4,0.6,0.2,0.01),
  'neg.exp',
  iters=250
)
plot(fit)
```

`plot.litfit_bootstrap` *Plot the bootstrap distribution for a parameter from a litfit object*

Description

Plot a bootstrap distribution of a particular coefficient

Usage

```
## S3 method for class 'litfit_bootstrap'
plot(x,coef.index,bw,...)
```

Arguments

`x`	litfit object
`coef.index`	coefficient number to plot from the `litfit` object, see order of coefficients for that particular model. Default is to plot the first parameter for that model
`bw`	bandwidth (or bandwidth algorithm see `density`) for the density plot
`...`	additional arguments passed to plot.default

Details

The grey fill goes from 0.025 quantile to the 0.975 quantile of the distribution. Red line shows the mean. Blue line shows the median.

Value

plot of litfit_bootstrap object, returns invisibly

Author(s)

James Weedon

See Also

`fit_litter` `bootstrap_parameters` `density`

Examples

```
fit <- fit_litter(time=pineneedles$Year,
mass.remaining=pineneedles$Mass.remaining, model='neg.exp', iters=200)
boot1 <- bootstrap_parameters(fit, nboot = 500)
plot(boot1)
```

`plot_multiple_fits` *Plot multiple fits on one graph with model selection results displayed*

Description

Plot multiple fits of decomposition trajectories on one graph with model selection results displayed

Usage

```
plot_multiple_fits(time,mass.remaining,model,color,iters,bty,...)
```

Arguments

`time`	vector of time points
`mass.remaining`	vector of mass remaining
`model`	vector of models to fit and plot (see `fit_litter`)
`color`	a vector of colors the same length as the number of models
`iters`	parameter passed to `fit_litter`
`bty`	bty
`...`	additional parameters passed to `plot`

Details

this function is designed to compare a variety of curve shapes visually and with AIC and BIC simultaneously

Value

plot of multiple fits, returns invisibly

Author(s)

Liu Guofang

See Also

`fit_litter` `plot.litfit`

Examples

```
data(pineneedles,package='litterfitter')

plot_multiple_fits(time = pineneedles$Year,
mass.remaining = pineneedles$Mass.remaining,
bty = 'n', model = c('neg.exp', 'weibull'),
xlab = 'Time', ylab = 'Proportion mass remaining',iters=200)
```

`predict.litfit` *Predict method for litfit objects*

Description

Generated predicted values for (new) time points from a litfit model fit

Usage

```
## S3 method for class 'litfit'
predict(object,newdata=NULL,...)
```

Arguments

`object`	litfit object
`newdata`	optional vector of new Time points at which to predict mass remaining. If not specified, Time points from the original fit are used.
`...`	further arguments passed to or from other methods.

Details

to do

Value

predicted values from a litfit object

Author(s)

Will Cornwell

James Weedon

See Also

`fit_litter`

Examples

```
fit<-fit_litter(time=c(0,1,2,3,4,5,6),mass.remaining=c(1,0.9,1.01,0.4,0.6,0.2,0.01),
'neg.exp',iters=250)
predict(fit, newdata=1:10)
```

`steady_state` *Estimate Steady State Biomass*

Description

Estimate Steady State Biomass

Usage

```
steady_state(x = NULL, pars = NULL, model = NULL)
```

Arguments

`x`	A `litfit` object. If provided, `pars` and `model` parameters are extracted from this object.
`pars`	A numeric vector of parameters for the model. Only needed if `x` is not provided.
`model`	A character string specifying the decomposition model. Must be one of the following: "neg.exp", "weibull", "discrete.parallel", "discrete.series", or "cont.quality2". Only needed if `x` is not provided.

Details

Computes the steady state biomass, as a proportion of the annual input, based on a given model fit or parameters.

Currently, the function supports a subset of decomposition models. New model support is planned for future updates.

Value

A named numeric value representing the estimated steady state biomass from the specified model.

Author(s)

Will Cornwell

See Also

`fit_litter` for generating `litfit` objects.

Examples

```
# Example with litfit object
fit <- fit_litter(
  time = c(0,1,2,3,4,5,6),
  mass.remaining = c(1,0.9,1.01,0.4,0.6,0.2,0.01),
  model = 'neg.exp',
  iters = 250
)
steady_state(fit)

# Example with specific model and parameter values
steady_state(pars = c(6,2), model = "weibull")
```

`time_to_prop_mass_remaining` *Get the predicted time until half mass loss for a litter decomposition trajectory*

Description

Get estimated time to 0.5 (or an alternate threshold) mass loss from a particular fit to a litter decomposition trajectory

Usage

```
time_to_prop_mass_remaining(x,threshold.mass=0.5)
```

Arguments

`x`	a litfit object
`threshold.mass`	mass loss threshold in proportion, default is 0.5

Details

this function finds the time to a specified mass loss percentage

Value

numeric value that represents time to a specified mass loss percentage

Author(s)

Will Cornwell

See Also

`fit_litter` `plot.litfit`

Examples

```
fit<-fit_litter(time=pineneedles$Year,mass.remaining=pineneedles$Mass.remaining,
model='neg.exp',iters=1000)
time_to_prop_mass_remaining(fit, threshold.mass = 0.5)
```

Index