Package ‘GUTS’

February 12, 2019

Type Package

Title Fast Calculation of the Likelihood of a Stochastic Survival Model

Version 1.1.0

Date 2019-01-29

Author Carlo Albert <carlo.albert@eawag.ch>, Sören Vogel <soeren.vogel@posteo.ch>, Oliver Jakoby <oliver.jakoby@rifcon.de> and Alexander Singer <alexander.singer@rifcon.de>

Maintainer Oliver Jakoby <oliver.jakoby@rifcon.de>

Description Given exposure and survival time series as well as parameter values, GUTS allows for the fast calculation of the survival probabilities as well as the logarithm of the corresponding likelihood.

License GPL (>= 2)

Depends R (>= 3.0.0), methods, Rcpp (>= 0.12.16)

LinkingTo Rcpp

LazyLoad yes

LazyData no

Encoding UTF-8

Suggests knitr, rmarkdown, adaptMCMC, xlsx, drc

VignetteBuilder knitr

Repository CRAN

Repository/R-Forge/Project guts

Repository/R-Forge/Revision 51

Repository/R-Forge/DateTimeStamp 2019-02-12 14:05:12

Date/Publication 2019-02-12 15:00:03 UTC

NeedsCompilation yes
**Description**

GUTS (General Unified Threshold model of Survival) is a stochastic survival model for ecotoxicology. The package allows for the definition of exposure and survival time series as well as parameter values, and the fast calculation of the survival probabilities as well as the logarithm of the corresponding likelihood.

**Details**

- **Package:** GUTS
- **Type:** Package
- **License:** GPL (>= 2)

A GUTS object is a special list of class “GUTS”. Functions `guts_setup`, `guts_calc_loglikelihood` and `guts_calc_survivalprobs` are available to create and work with GUTS objects. A data set `diazinon` is also included. See links for more details.

**Author(s)**

Carlo Albert <carlo.albert@eawag.ch>, Sören Vogel <soeren.vogel@posteo.ch>, Oliver Jakoby <oliver.jakoby@rifcon.de> and Alexander Singer <alexander.singer@rifcon.de>

Maintainer: Oliver Jakoby <oliver.jakoby@rifcon.de>

**References**


**See Also**

`guts_setup`, `guts_calc_loglikelihood`, `guts_calc_survivalprobs`, `guts_report_damage`, `diazinon`
**Description**

Data of 3 pulsed toxicity tests with the freshwater crustacean Gammarus pulex and diazinon, an organophosphate insecticide.

**Usage**

```r
data("diazinon")
```

**Format**

A list containing 12 data vectors.

**Author(s)**

Carlo Albert <carlo.albert@eawag.ch>, Sören Vogel <soeren.vogel@posteo.ch>, Oliver Jakoby <oliver.jakoby@rifcon.de> and Alexander Singer <alexander.singer@rifcon.de>

**Maintainer**

Oliver Jakoby <oliver.jakoby@rifcon.de>

**Source**


**See Also**

GUTS

---

**GUTS**

*Fast Calculation of the Likelihood of a Stochastic Survival Model*

**Description**

GUTS (General Unified Threshold model of Survival) is a stochastic survival model for ecotoxicology. The package allows for the definition of exposure and survival time series as well as parameter values, and the fast calculation of the survival probabilities as well as the logarithm of the corresponding likelihood.

The package implements the GUTS-SIC (also called GUTS-RED) variants that assume a one-compartment model with first-order toxicokinetics.
Usage

```r
guts_setup(C, Ct, y, yt, dist = "lognormal",
model = "Proper", N = 1000, M = 10000)

guts_calc_loglikelihood(gobj, par, external_dist = NULL)

guts_calc_survivalprobs(gobj, par, external_dist = NULL)

guts_report_damage(gobj)
```

Arguments

- **C**: Numeric vector of concentrations. Vector must contain at least 2 values and be of the same length as Ct.
- **Ct**: Numeric vector of concentration time points. Vector must contain at least 2 values and be of the same length as C. Time points must start at 0, and contain unique values in ascending order.
- **y**: Integer vector (counts) of survivors. Vector must contain at least 2 values and be of the same length as yt. y must not be ascending.
- **yt**: Numeric vector of survivor time points. Vector must contain at least 2 values and be of the same length as y. Time points must start at 0, and contain unique values in ascending order. Survivor information at time points later than the latest concentration time point will be disregarded (with a warning).
- **dist**: Distribution as character, either "lognormal" (default), "loglogistic", "external" or "delta".
- **model**: Model as character, either “Proper” (for full model, the default), “IT” (for individual tolerance), or “SD” (for stochastic death).
- **N**: Integer. Thresholds sample length. Must be greater than 2.
- **M**: Integer. Number of time grid points. Must be greater than 1.
- **gobj**: GUTS object. The object to be updated (and used for the calculation).
- **par**: Numeric vector of parameters. See details below.
- **external_dist**: Numeric vector containing the distribution of individual thresholds. Only used if dist = 'external'. See details below.

Details

**Functions:**

- Use `guts_setup` to define (or alter) a GUTS object. Various checks are applied to the data. On success, a GUTS object will be created.
- Use `guts_calc_loglikelihood` to calculate the survival probabilities and the corresponding loglikelihood for a given set of parameters. The function is very fast and can be used in routines for parameter estimation. The function returns the loglikelihood, however it also updates the fields par, S and LL of the object.
- `guts_calc_survivalprobs` is a convenience wrapper that can be used for predictions; it returns the survival probabilities.
guts_report_damage returns a data.frame with time grid points and the damage for each of these. The function reports the damage that was calculated in the previous call to guts_calc_loglikelihood or guts_calc_survivalprobs.

**Models, Parameters, and Distributions:**

The GUTS package provides three model types:

- **Proper**: a GUTS-SIC-Proper (also called GUTS-RED-Proper) model using random individual tolerances and a stochastic death process, when individual tolerances are exceeded.
- **IT**: a GUTS-SIC-IT (GUTS-RED-IT) individual tolerance model using random individual tolerances. If an individual's tolerance threshold is exceeded, the individual dies.
- **SD**: a GUTS-SIC-SD (GUTS-RED-SD) stochastic death model using a stochastic death process above a population-wide tolerance threshold. The tolerance-threshold is the same for all individuals.

The Proper GUTS model requires the following parameters par, while variants IT and SD are based on a reduced subset (as indicated in brackets). Parameter values in par must be ordered as listed here:

- hb: background mortality rate (Proper, IT, SD)
- ke: dominant rate constant (Proper, IT, SD)
- kk: killing rate (Proper, SD)
- further parameters for the tolerance threshold (in SD) or the threshold distribution dist (in Proper and IT)

For model type “SD” (stochastic death), required parameters par[1:4] are kb, ke, kk and mn, which is the population-wide tolerance threshold. For backwards compatibility this model type can be initiated setting dist = "Delta" and model = "Proper". For model type “IT” (individual tolerance), required parameters par[1:2] are kb, ke, as well as respective distribution parameters (continued from par[3]). Parameter (kk) is set internally to infinity and must not be provided. For model type “Proper”, all parameters are needed. par[1:3] take kb, ke, kk, distribution parameters follow from par[4]).

For model types “Proper” and “IT” individual tolerance thresholds are created internally. Individual tolerances are drawn from the specified distribution dist:

- "lognormal": requires the parameters mn and sd which are the mean and standard deviation of the lognormal random distribution. In contrast to parameters meanlog and sdlog of function dlnorm, these parameters are not on the logscale. They relate in the following way:

  \[
  sdlog = \sqrt{\ln \frac{1 + sd^2}{mn^2}} \\
  meanlog = \ln mn - \frac{1}{2} * sdlog^2
  \]

- "loglogistic": requires the parameters mn = scale = median and beta = shape.
- "external": uses random variates provided to external_dist. With this option GUTS can be run with arbitrarily distributed individual tolerance thresholds. With the option “external” only parameters hb, ke and kk are required. Further, the thresholds sample length N is internally adjusted to the length of the external vector of random variates external_dist. The adjustment of N is notified by a warning.
For performance reasons the implemented distributions “lognormal” and “loglogistic” are approximated using importance sampling. The option “external” generally performs well, but might require a larger thresholds sample (i.e. length(external_dist) should be large).

The number of parameters is checked according to dist and model. Wrong number of parameters invoke an error, wrong parameter values (e.g., negative values) invoke a warning, and the loglikelihood is set to -Inf.

**Field and Attribute Access:**

Fields and attributes of an object of class “GUTS” are read-only. To prevent accidental change of fields or attributes, replacement functions were rewritten throwing an error when used. Always use function guts_setup to create objects or modify fields on existing objects. However, guts_calc_loglikelihood and guts_calc_survivalprobs update an object’s fields par (parameters), D (damage), S (survival probabilities) and LL (the loglikelihood).

**Value**

- guts_setup returns a list of class “GUTS” with the following fields:
  - C: Concentrations.
  - Ct: Concentration time points.
  - y: Survivors.
  - yt: Survivor time points.
  - dist: Distribution.
  - model: Model.
  - N: Sample length.
  - M: Time grid points.
  - par: Parameters.
  - S: Vector of survivor probabilities.
  - D: Vector of internal damage for each of the M time grid points.
  - LL: The loglikelihood.

- guts_calc_loglikelihood returns the loglikelihood.
- guts_calc_survivalprobs returns the survival probabilities.
- guts_report_damage returns the damage.

**Note**

The GUTS project web site can be found here: [http://guts.r-forge.r-project.org](http://guts.r-forge.r-project.org). For questions and discussion, please subscribe to the mailing list there.

**Author(s)**

- Carlo Albert <carlo.albert@eawag.ch>, Sören Vogel <soeren.vogel@posteo.ch>, Oliver Jakoby <oliver.jakoby@rifcon.de> and Alexander Singer <alexander.singer@rifcon.de>

Maintainer: Oliver Jakoby <oliver.jakoby@rifcon.de>
References


See Also
diazinon, GUTS-package, vignettes ../doc/ringTest.html and ../doc/ringTest.html for examples on how to calibrate and project GUTS-models.

Examples

data(diazinon)

# create GUTS object to calculate the Proper model
# using a log-normal distribution of tolerance thresholds
gts.lognormal <- guts_setup(
  C = diazinon$c1, Ct = diazinon$cT1,
  y = diazinon$y1, yt = diazinon$yT1,
  dist = "lognormal", model = "Proper")

guts_calc_loglikelihood(  gts.lognormal,
  c(.051, .126, 1.618, 19.099, 6.495))
gts.lognormal  # show GUTS object

# repeating calculation above
# with threshold values from an external log-normal distribution.
# Note, we need to account for the different parametrisations
# used in the GUTS-package and in rlnorm
sigma2 <- log(1 + 6.405^2 / 19.099^2)
mu <- log(19.099) - 0.5 * sigma2
lognormal.thresholds <- rlnorm(1000, meanlog = mu, sdlog = sqrt(sigma2))
gts.external <- guts_setup(
  C = diazinon$c1, Ct = diazinon$c$t1,
  y = diazinon$y1, yt = diazinon$yt1,
  dist = "external", model = "Proper")
guts.calc_loglikelihood(
  gts.external,
  c(0.051, 0.126, 1.618), external_dist = lognormal.thresholds)
# -> Results using external and internal distributions are comparable

# create GUTS object to calculate the Proper model
# using a log-logistic distribution of tolerance thresholds

gts.loglogistic <- guts_setup(
  C = diazinon$c1, Ct = diazinon$c$t1,
  y = diazinon$y1, yt = diazinon$yt1,
  dist = "loglogistic", model = "Proper")
guts.calc_survivalprobs( # returning survival probabilities
  gts.loglogistic,
  c(0.01, 0.2, 0.3, 3, 2))

str(guts.report_damage(gts.loglogistic)) # returning damage

# calculate survival probabilities with IT model
# using a log-logistic distribution of tolerance thresholds

guts.calc_survivalprobs( 
  guts_setup(
    C = diazinon$c1, Ct = diazinon$c$t1,
    y = diazinon$y1, yt = diazinon$yt1,
    dist = "loglogistic", model = "IT"),
    c(0.01, 0.2, 3, 2))

# calculate survival probabilities with an SD model with a fixed tolerance threshold

guts.calc_survivalprobs(
  guts_setup(
    C = diazinon$c1, Ct = diazinon$c$t1,
    y = diazinon$y1, yt = diazinon$yt1,
    dist = "loglogistic", model = "SD"),
    c(0.01, 0.2, 0.3, 3))

## Not run: guts.calc_survivalprobs(gts.external, rep(.5, 3))
# Warning and no result, because no external distribution was specified

## Not run: guts.calc_survivalprobs(gts.loglogistic, 1:4) # Error.

## Not run: gts.loglogistic[["C"]]<-1:3 # Error.
Index

*Topic datasets
  diazinon, 3
  [[<-.GUTS (GUTS-package), 2
  $<-.GUTS (GUTS-package), 2
  attr<-.GUTS (GUTS-package), 2
  attributes<-.GUTS (GUTS-package), 2
  diazinon, 2, 3, 7
  dlnorm, 5
  GUTS, 3, 3
  guts (GUTS), 3
  GUTS-package, 2
  guts_calc_loglikelihood, 2
  guts_calc_loglikelihood (GUTS), 3
  guts_calc_survivalprobs, 2
  guts_calc_survivalprobs (GUTS), 3
  guts_report_damage, 2
  guts_report_damage (GUTS), 3
  guts_setup, 2
  guts_setup (GUTS), 3
  modguts (GUTS-package), 2
  mostattributes<-.GUTS (GUTS-package), 2
  print.GUTS (GUTS-package), 2
  Rcpp, 3