`morse`: an R-package to analyse toxicity test data

2021-04-07

Summary

The `morse` package is used for the analysis of experimental data collected from standard toxicity tests. It provides ready-to-use functions to visualize a dataset and to estimate several toxicity indices to be further used in support of environmental risk assessment in full compliance with regulatory requirements (OECD 2006). Such toxicity indices are indeed typically requested by standardized regulatory guidelines on which national agencies base their evaluation of applications for regulatory approval of chemical substances; see for example OECD toxicity testing guideline (OECD 2016, 2012). Tools gathered together within the `morse` package involve the most advanced and innovative methods developed for ecotoxicology, such as appropriate stochastic parts in dose-response modelling (Delignette-Muller et al. 2014), and the use of Bayesian statistics to get parameter estimates as posterior probability distribution (Billoir et al. 2008). Hence, these tools are easily accessible for ecotoxicologists and regulators who do not need to deeply invest in the underlying technicalities in order to perform a valuable quantitative environmental risk assessment.

This paper provides an overview of a typical use of the `morse` package with survival data collected over time and at different increasing exposure concentrations. These data are analysed with the reduced version of GUTS models based on the stochastic death hypothesis (namely, the `GUTS-RED-SD` model). This example can be followed, step-by-step, to analyse any new dataset, as long as the dataset format is respected.

Statement of Need

The `morse` package can be used to get estimates of LC_x (x% Lethal Concentration) or EC_x (x% Effective Concentration) by fitting standard exposure-response or effect models on toxicity test data. Toxicity indicator estimates as well as model parameters are provided along with the quantification of their uncertainty. The `morse` package can also be used to get estimates of the NEC (No Effect Concentration) by fitting a toxicokinetic-toxicodynamic (TKTD) model (namely `GUTS` models, that is *General Unified Threshold models of Survival*). Using `GUTS` models also allows a user to get estimates of $LC_{(x,t)}$ (whatever x and t) and $LP_{(x,t)}$, the latter being defined by EFSA as the x% multiplication factor leading to an additional reduction of x% in survival at the end of the exposure profile. Above all, `GUTS` models can be used on data collected under time-variable exposure profiles.

The `morse` package (Baudrot et al. 2021) has been tested using `R` (version 3.5 and later) on macOS, Linux and Windows machines. Regarding the particular case of TKTD models for survival, namely GUTS models, the `morse` package was ring-tested together with nine other `GUTS` implementations under different software platforms; see Appendix A in Jager and Ashauer (2018) for details. Note that we were the only team to perform the ring-test under a Bayesian framework. Other implementations were `BYOM`, `DEBTOX`, `DELPHI`, `EPYTOX`, `GUTS-3S`, `MATHEMATICA`, `MODELMAKER`, `OPENMODEL`, `EASYGUTS` and `GATEAUX`, all under a frequentist statistical framework. All participants to the ring-test received the same datasets and tasks, carried out their simulations independently from each other and sent the results back to the coordinator for analysis. Output estimation was similar to other implementations, therefore, package `morse` was confirmed as fit-for-purpose in fitting GUTS models on survival toxicity test data.

All functions in the `morse` package can be used without a deep knowledge of their underlying probabilistic model or inference methods. Rather, they are designed to behave as well as possible, without requiring the user to provide values for some parameters. Nevertheless, models implemented in `morse` can also be used as

a first step to tailor new models for more specific situations.

Note that `morse` benefits from a web interface, MOSAIC, from which the same analyses can be reproduced directly on-line without having to implement them directly in `R` programming. MOSAIC is freely available at https://mosaic.univ-lyon1.fr/ (Charles et al. 2018) (Figure 1).

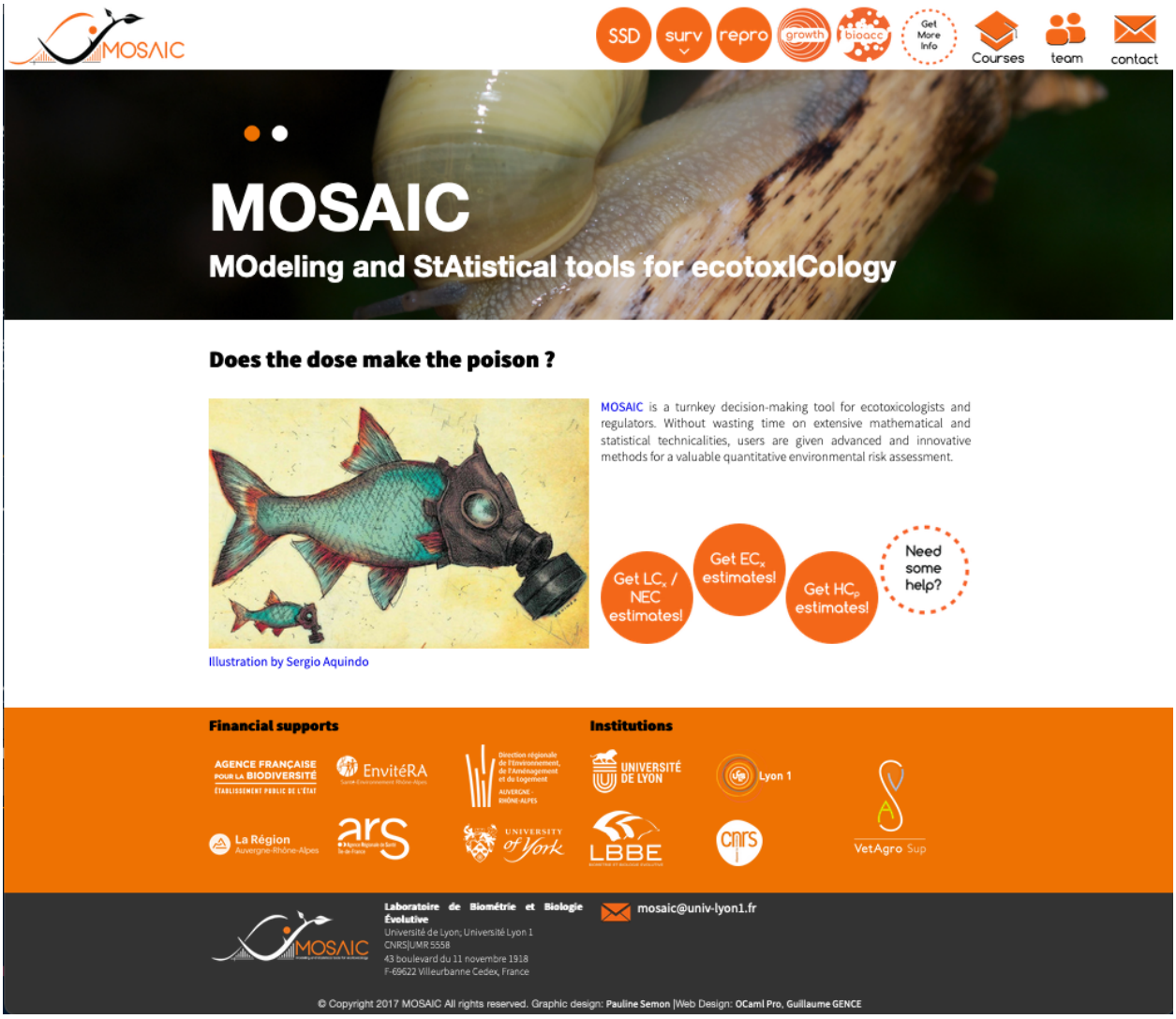

Figure 1: Homepage of the MOSAIC web platform (https://mosaic.univ-lyon1.fr/)..

Availability

The `morse` package is available as an `R` package; it can be directly downloaded from CRAN https://CRAN.R-project.org/package=morse, where package dependencies and system requirements are also documented. The development version can be found on GitHub https://github.com/pveber/morse, where code contributions, bug reports, fixes and feature requests are more than welcome by opening issues and pull requests.

Main features

The main functions in the `morse` package are `survData()`, `reproData()`, and `plotDoseResponse()` to visualize raw data. The `survFitTT()`, `reproFitTT()`, and `survFit()` functions allow a user to fit a model on data in order to estimate toxicity indicators, the choice of which depends on the type of data. Fitting outputs can either be displayed with `plot()` or synthesized with `summary()`. Functions are available to check the goodness-of-fit, namely `ppc()` and `plot_prior_post()`. Predictions can be performed with `predict()`, `predict_ode()`, `predict_Nsurv()`, and `predict_Nsurv_ode()`. Finally, function `LCx()`and `MFx()` allow a user to get x% lethal concentrations or profiles, respectively.

The `morse` package currently handles binary and count data, for example survival and reproduction data. Functions dedicated to binary (respectively, count) data analysis start with a `surv` (respectively, `repro`) prefix. `morse` provides a similar workflow in both cases:

1. create and validate a dataset;
2. explore a dataset;
3. plot a dataset;
4. fit a model on data and get parameter estimates;
5. check goodness-of-fit with Posterior Predictive Check plot (PPC).

In addition, for binary data handled with `GUTS` models, the `morse` package also allows a user to:

1. calculate and plot $LC_{(x,t)}$ and $LP_{(x,t)}$;
2. compute goodness-of-fit criteria: the PPC percentage, the Normalized Root Mean Square Error (NRMSE), and the Survival Probability Prediction Error (SPPE) at the end of the exposure profile.

See EFSA PPR Panel (2018) for details.

Those steps are presented in depth in the `Tutorial` available at https://cran.r-project.org/web/packages/morse/vignettes/tutorial.html, with all necessary details to implement all `morse` features. A more formal description of the models and the estimation procedures are provided in a document called "Models in morse package" available at https://cran.r-project.org/web/packages/morse/vignettes/modelling.pdf. Please refer to this documentation for further introduction to the use of the `morse` package.

Minimal Working Example

Installation

JAGS

The `morse` package is linked to JAGS http://mcmc-jags.sourceforge.net/, a Bayesian sampler used to perform inference with all implemented models. So, you need to download and install JAGS at https://sourceforge.net/projects/mcmc-jags/. Then you must test that your `R` graphical user interface has access to JAGS, and, if not, to specify where JAGS can be found on your computer. Indeed, once installed, JAGS can be lost in the PATH. To help solve this issue, you can use package `runjags` which is installed and loaded as follows.

```
### install the `runjags` package, if needed
if(is.element('runjags', installed.packages()[,1]) == FALSE){
  install.packages('runjags')
}
### load the `runjags` package
library("runjags")
### run test
testjags()
```

The output should look like this:

```
You are using R version 4.0.2 (2020-06-22) on a windows machine, with the RStudio GUI
```

```
JAGS version 4.3.0 found successfully using the command
'C:/Program Files/JAGS/JAGS-4.3.0/x64/bin/jags-terminal.exe'
The rjags package is installed
```

Otherwise, you can specify to your R graphical user interface where the JAGS executable is located in your computer (somewhere in 'C:/Program Files/JAGS/JAGS-4.3.0/x64/bin/jags-terminal.exe' on Windows machines):

```
testjags(jags=runjags.getOption('jagspath'))
### replace `jagspath` by your own PATH to JAGS
### For instance
### 'C:/Program Files/JAGS/JAGS-4.3.0/x64/bin/jags-terminal.exe'
```

Loading morse and its dependencies

In order to use the morse package, you need to install it with all its dependencies on other R-packages: mandatory ones (coda, deSolve, dplyr, epitools, graphics, grDevices, ggplot2 ($\geqslant$ 2.1.0), grid, gridExtra, magrittr, methods, reshape2, rjags ($\geqslant$ 4.0), stats, tibble, tidyr, zoo) and suggested ones (knitr, rmarkdown, testthat). For this purpose, you can use the two classical R commands:

```
### install the `morse` package, if needed
if(is.element('morse', installed.packages()[,1]) == FALSE){
  install.packages('morse')
}
### load the `morse` package
library(morse)
```

C++

Note that the morse package is also linked to C++. C++ is used for running simulations leading to predictions. In R, you should not have issues with C++ requirements since it is very well integrated (many R functions are simple interfaces to C++ functions). Feel free to report any trouble at https://github.com/pveber/morse/issues by opening a new issue for the morse package.

Survival analysis

We assume hereafter that morse and all the above dependencies have been corrected installed. To illustrate the use of morse, we will use a standard survival dataset coming from a chronic laboratory toxicity test with *Gammarus pulex*, a freshwater invertebrate, exposed to increasing concentrations of propiconazole (a fungicide) during four days. Eight concentrations were tested with two replicates of 10 organisms per concentration. Survival was monitored at five time points (at day 0, 1, 2, 3 and 4) (Nyman, Schirmer, and Ashauer 2012).

We will use the reduced version of the GUTS model based on the stochastic death hypothesis (namely, the GUTS-RED-SD model), as recommended by the *European Food Safety Authority* (EFSA) for the environmental risk assessment (ERA) of plant protection products potentially toxic for aquatic living organisms (EFSA PPR Panel 2018). This model can also be fitted on-line with the MOSAIC web platform (Baudrot, Veber, et al. 2018). Below is the *modus operandi* with the morse package to be followed step-by-step in order to be in full compliance with the EFSA workflow for ERA (EFSA PPR Panel 2018).

Calibration step

```
### load package `morse`
library(morse)
### load a dataset
data("propiconazole")
### create a morse object for binary data analysis
```

```
survData_PRZ <- survData(propiconazole)
### fit a reduced GUTS model (GUTS-RED) with option "SD" (Stochastic Death)
fit_cstSD <- survFit(survData_PRZ, model_type = "SD")
### plot the fitting result
plot(fit_cstSD)
```

Get the x% lethal concentration

Using a GUTS model with `morse` allows the user to get a probability distribution on the x% lethal concentration whatever the exposure duration t, namely the $LC_{(x,t)}$. By default, t corresponds to the last time point in the dataset and $x = 50\%$.

```
### run function LCx()
LCX_cstSD <- LCx(fit_cstSD, X = 50)
### plot the output as a concentration-response curve
plot(LCX_cstSD)
```

Validation step

Validation consists in predicting the number of survivors over time under pulsed-exposure profiles for which observations have also been collected. Predictions are then compared to observations and their adequacy is checked according to several validation criteria defined by EFSA (EFSA PPR Panel 2018). The aim of this step is to choose an appropriate model for the following step.

```
### load data collected under pulsed exposure profiles
data("propiconazole_pulse_exposure")
### predict the number of survivors for all profiles
predict_Nsurv <- predict_Nsurv_ode(
  object = fit_cstSD,
  data_predict = propiconazole_pulse_exposure
)
### plot results
plot(predict_Nsurv)
```

Once the predictions are visually checked (Figure 4), quantitative validation criteria need to be calculated.

```
### check for adequacy between predictions and observations
predict_Nsurv_check(predict_Nsurv)

check <- predict_Nsurv_check(predict_Nsurv)
check$Percent_PPC_global
check$Percent_NRMSE_global
```

This reveals that, in total, 84% of the observations lie within the uncertainty band of the predictions, while the global variability of data around the predictions is 16.2%. For both criteria, a maximum value of 50% is expected, what means here that we do not expect specific risk for the species and the chemical compound under consideration.

Prediction step

Risk assessors are interested in testing various exposure scenarios, having a certain environmental realism that is varying over time. Risk assessors expect to evaluate the potential impact of these profiles on survival of target species to protect. Typically, they want to compute the multiplication factor $MF_{(x,t)}$ that could be applied to the exposure profile without reducing more than by x% the survival probability at a specified test duration t (default being the last time point of the exposure profile). This is the so-called x% lethal profile,

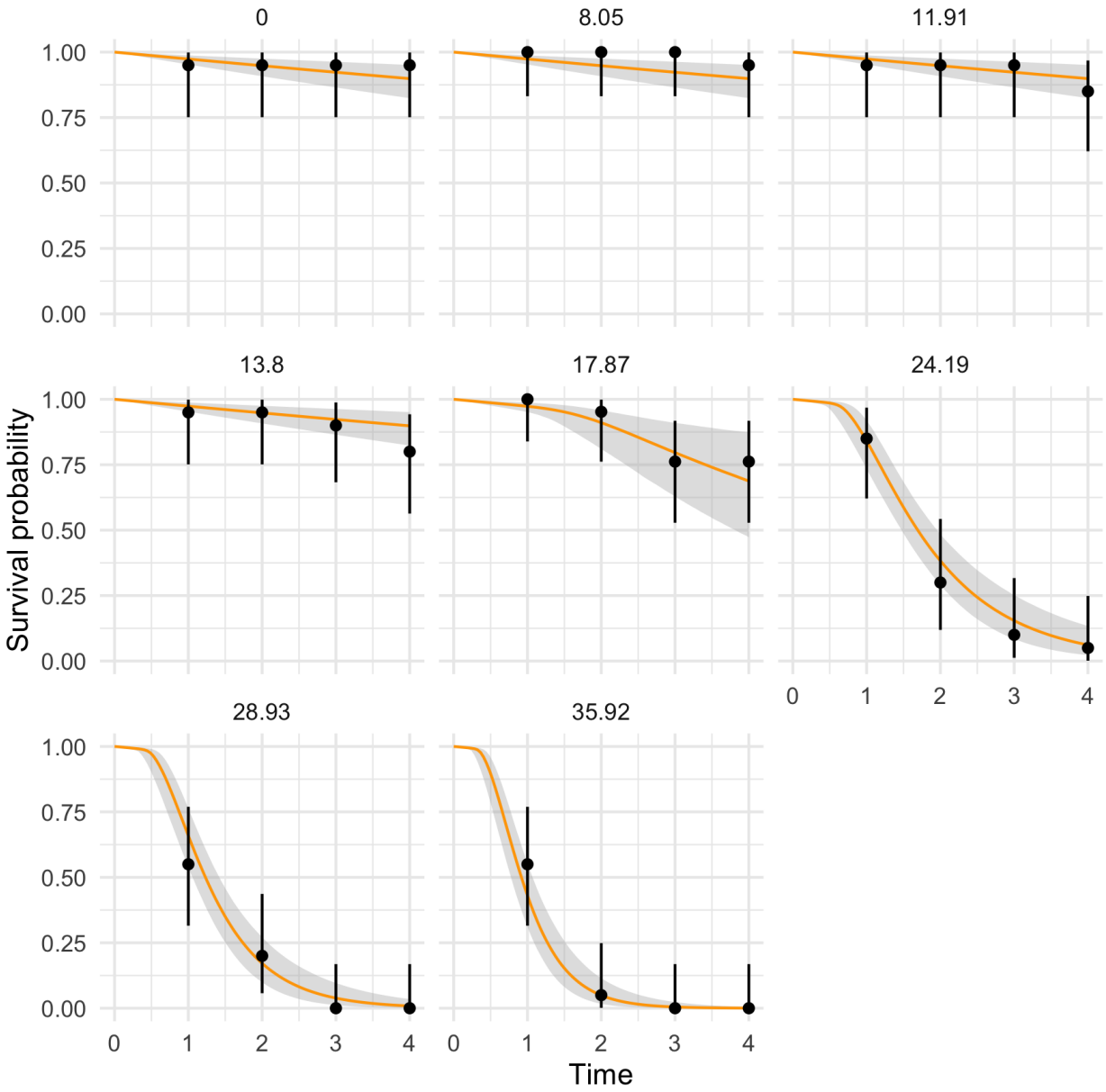

Figure 2: Fitting result with a GUTS-RED-SD model. The median fitted curves are in orange and the 95% uncertainty bands in gray. Black dots are observed data surrounded by their binomial confidence intervals.

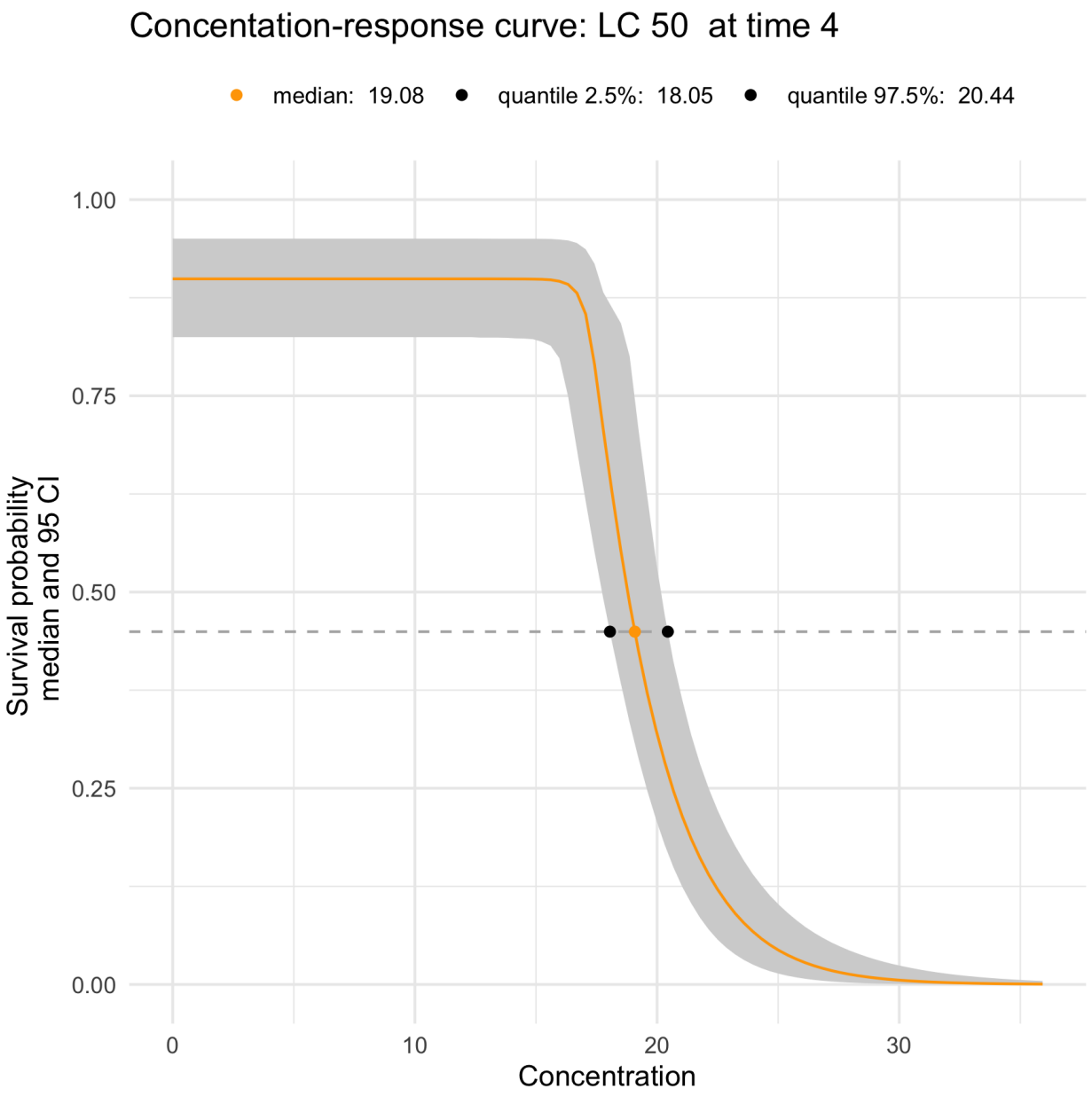

Figure 3: Simulated concentration-response curve corresponding to the previous fitting result on Figure 2.

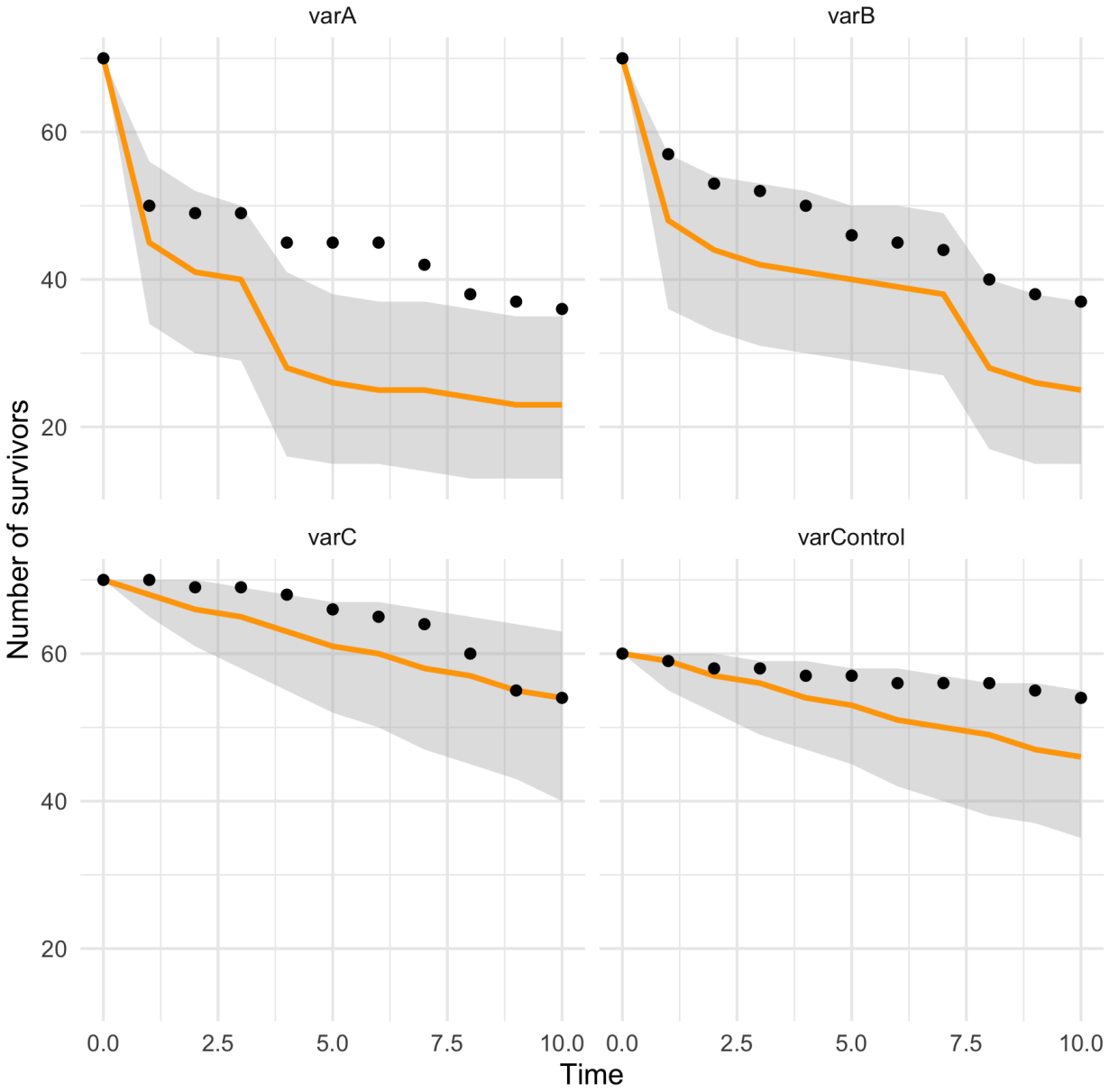

Figure 4: Visual check of adequacy between predictions based on the GUTS-RED-SD model with parameter values estimated in the calibration step (median prediction in orange, uncertainty band in gray), and observations (black dots).

denoted LP_x, and newly proposed by EFSA PPR Panel (2018). This calculation is provided by function `MFx()` in `morse`.

The mathematical definition of the $x\%$ Multiplication Factor at time t (at the end of a time series $T = \{0, \ldots, t\}$) is given by:

$$S\left(MF_{(x,t)} \times C_w(\tau \in T), t\right) = S(C_w(\tau \in T), t) \times \left(1 - \frac{x}{100}\right)$$

where $C_w(\tau \in T)$ is the original exposure profile, and expression $S\left(MF_{(x,t)} \times C_w(\tau \in T), t\right)$ the survival probability after the exposure profile has been translated upward by a multiplication $MF_{(x,t)}$; the new exposure profile thus becomes equal to $MF_{(x,t)} \times C_w(\tau \in T)$.

```
### define an exposure profile (here a theoretical one)
data_4MFx <- data.frame(time = 1:10,
                        conc = c(0,0.5,8,3,0,0,0.5,8,3.5,0))
### run function MFx()
MFx_PRZ_cstSD <- MFx(object = fit_cstSD, data_predict = data_4MFx, ode = TRUE)
### plot the survival probability at the end of the exposure profile
### according to a range of multiplication factors (log-scale)
plot(MFx_PRZ_cstSD, log_scale = TRUE)
```

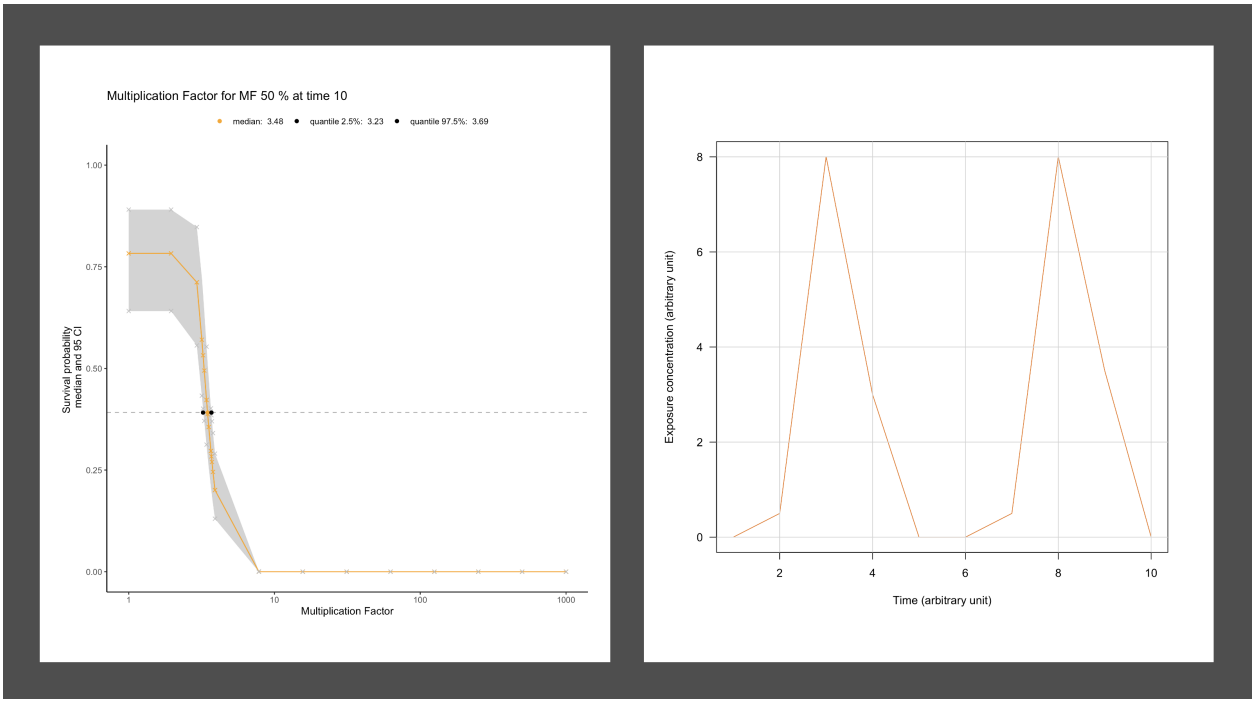

Figure 5: (left) Predicted survival probability according to a range of multiplication factors (log-scale) at the end of a theoretical exposure profile (right).

Predict survival probability under any exposure profile

Finally, it may be useful to predict the survival probability under any exposure profile (time-variable or not), for example when designing new experiments or to better understand what happens in field. Below are some examples that you can use to build your own simulations.

```
### define an exposure profile (here a theoretical one)
### note that here you need to specify a third column `replicate`
data_example <- data.frame(
   time = c(1,1.9,2,15,15.1,20),
   conc = c(0,0,20,20,0,0),
   replicate = rep("Basic example", 6)
)
### perform basic prediction
predict_example_NULL <- predict_ode(
  object = fit_cstSD,
  data_predict = data_example,
  mcmc_size = 10,
  interpolate_length = NULL)
### plot the result for only few exposure time points
plot(predict_example_NULL)
### define the same basic exposure profile
### but by changing the `replicate` value
data_example <- data.frame(
   time = c(1,1.9,2,15,15.1,20),
   conc = c(0,0,20,20,0,0),
   replicate = rep("Basic example (interpolation)", 6)
)
##### perform prediction with interpolation of the exposure profile
predict_example_100 <- predict_ode(
  object = fit_cstSD,
  data_predict = data_example,
  mcmc_size = 10,
  interpolate_length = 100)
# plot the result
plot(predict_example_100)
### load an environmentally realistic profile
data("FOCUSprofile")
FOCUSprofile[,"replicate"] <- "FOCUS example"
### perform prediction
predict_FOCUS <- predict_ode(
  object = fit_cstSD,
  data_predict = FOCUSprofile,
  mcmc_size = 10,
  interpolate_length = NULL)
### plot the result
plot(predict_FOCUS)
```

Research using `morse`

The `morse` package was recently used to evaluate the added value of using TKTD models in comparison with classical dose-response models, based on a case study with the snail *Limnaea stagnalis* when exposed to increasing concentrations of cadmium (Baudrot, Preux, et al. 2018). Also based on the `morse` package, we proposed some recommendations to address TKTD assessment using uncertainties in environmental risk models (Baudrot and Charles 2019).

More recently, benefiting from our experience in developing the `morse` package for ecotoxicology, we strongly contributed to the new `rbioacc` package https://CRAN.R-project.org/package=rbioacc, a turn-key package providing bioaccumulation metrics (BCF/BMF/BSAF) from a toxicokinetic (TK) model fitted

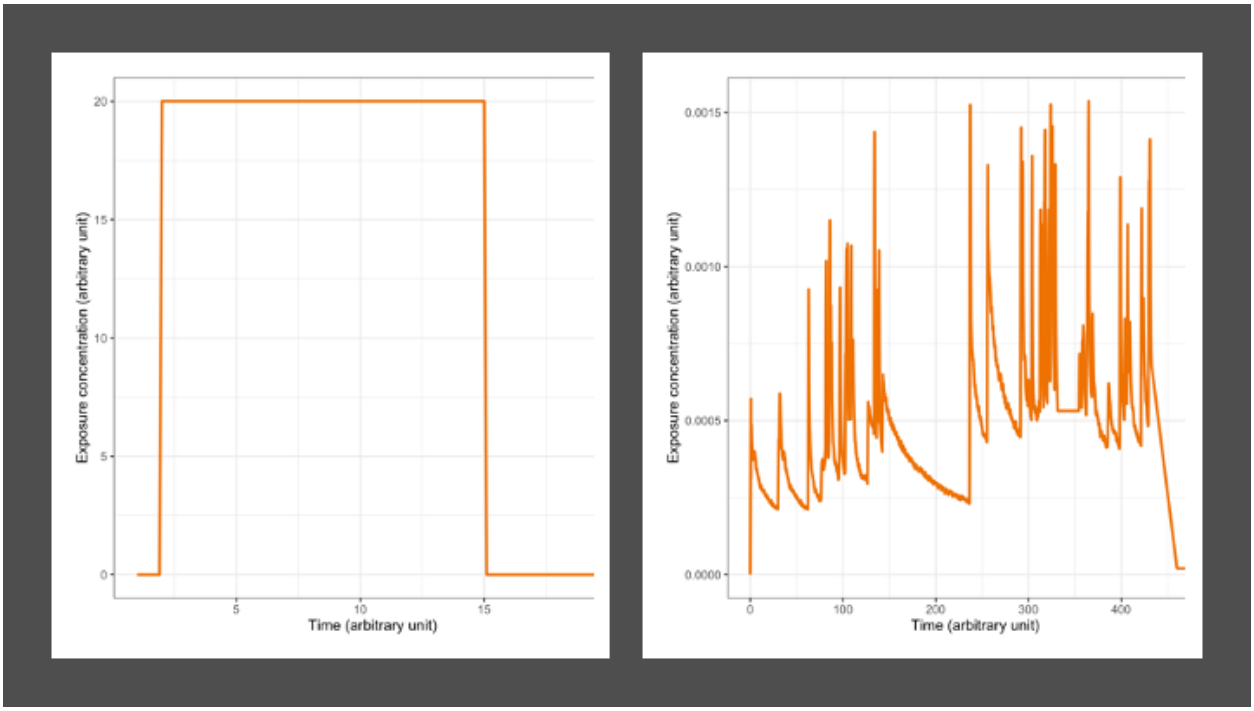

Figure 6: (left) Basic exposure profile; (right) Environmentally realistic exposure profile.

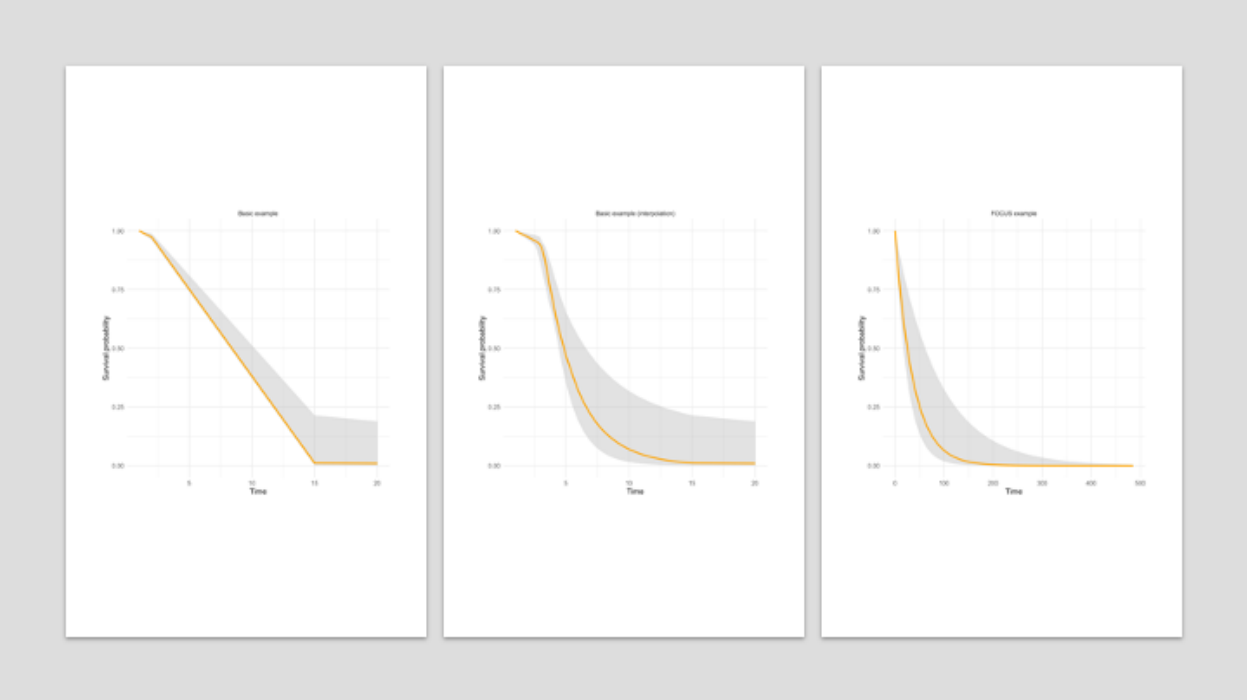

Figure 7: (left) Predicted survival probability over time under the basic exposure profile without interpolation; (middle) Predicted survival probability over time under the basic exposure profile with interpolation; (right) Predicted survival probability over time under an environmentally realistic exposure profile.

to accumulation-depuration data. The `rbioacc` package also supports the MOSAIC$_{bioacc}$ web platform https://mosaic.univ-lyon1.fr/bioacc. Last but not least, the 'rDEBtktd' package is currently under development https://gitlab.in2p3.fr/sandrine.charles/rDEBtktd to complement the `morse` package in order to fit, validate, and predict with TKTD models simultaneously describing growth and reproduction dynamics of living organisms under chemical pressure.

Data availability

A collection of eight datasets is made available directly in the `morse` package (use function `data()`). These datasets can also be downloaded online from the MOSAIC web platform by visiting the different modules: https://mosaic.univ-lyon1.fr.

Author contributions

V.B. (main developer of `morse`): conceptualization, methodology, formal analysis, data curation, visualization, writing manuscript. S.C.: supervision, funding acquisition, project administration, formal analysis, data curation, writing manuscript.

Acknowledgments

We fully thank all contributors to the `morse` package: Marie Laure Delignette-Muller, Wandrille Duchemin, Benoît Goussen, Nils Kehrein, Guillaume Kon-Kam-King, Christelle Lopes, Philippe Ruiz, Alexander Singer and Philippe Veber. We also thank the French National Agency for Water and Aquatic Environments (ONEMA, now the French Agency for Biodiversity) for its financial support to the development of the `morse` package. Finally, we thank a lot our reviewers, Tom Purucker and Peter Vermeiren, for their helpful comments leading to improvements in the manuscript and directly in the `morse` package, as well as Kristina Riemer for the management of the review process and for being so patient in finalizing this manuscript.

References

Baudrot, Virgile, and Sandrine Charles. 2019. "Recommendations to address uncertainties in environmental risk assessment using toxicokinetics-toxicodynamics models." *Scientific Reports, Natureresearch* 9: 11432. https://doi.org/10.1038/s41598-019-47698-0.

Baudrot, Virgile, Sandrine Charles, Marie Laure Delignette-Muller, Wandrille Duchemin, Benoit Goussen, Nils Kehrein, Guillaume Kon-Kam-King, et al. 2021. *Morse: Modelling Tools for Reproduction and Survival Data in Ecotoxicology.* https://CRAN.R-project.org/package=morse.

Baudrot, Virgile, Sara Preux, Virginie Ducrot, Alain Pavé, and Sandrine Charles. 2018. "New insights to compare and choose TKTD models for survival based on an inter-laboratory study for Lymnaea stagnalis exposed to Cd." *Environmental Science & Technology* 52 (3): 1582–90. https://doi.org/10.1021/acs.est.7b05464.

Baudrot, Virgile, Philippe Veber, Guillaume Gence, and Sandrine Charles. 2018. "Fit Reduced GUTS Models Online: From Theory to Practice." *Integrated Environmental Assessment and Management* 14 (5): 625–30. https://doi.org/10.1002/ieam.4061.

Billoir, Elise, Marie Laure Delignette-Muller, Alexandre R R Pery, and Sandrine Charles. 2008. "A Bayesian Approach to Analyzing Ecotoxicological Data." *Environmental Science & Technology* 42 (23): 8978–84. https://doi.org/10.1021/es801418x.

Charles, Sandrine, Marie Laure Delignette-Muller, Philippe Veber, and Marie Laure Delignette-Muller. 2018. "MOSAIC: a web-interface for statistical analyses in ecotoxicology." *Environmental Science and Pollution Research* 25: 11295–302. https://doi.org/10.1007/s11356-017-9809-4.

Delignette-Muller, Marie Laure, Christelle Lopes, Philippe Veber, and Sandrine Charles. 2014. "Statistical handling of reproduction data for exposure-response modeling." *Environmental Science & Technology* 48 (13): 7544–51. https://doi.org/10.1021/es502009r.

EFSA PPR Panel. 2018. "Scientific Opinion on the state of the art of Toxicokinetic/Toxicodynamic (TKTD) effect models for regulatory risk assessment of pesticides for aquatic organisms." *EFSA Journal* 16 (8): 5377. https://doi.org/10.2903/j.efsa.2018.5377.

Jager, Tjalling, and Roman Ashauer. 2018. *Modelling survival under chemical stress. A comprehensive guide to the GUTS framework*. Leanpub. https://leanpub.com/guts_book.

Nyman, A.-M., Kristin Schirmer, and Roman Ashauer. 2012. "Toxicokinetic-toxicodynamic modelling of survival of Gammarus pulex in multiple pulse exposures to propiconazole: model assumptions, calibration data requirements and predictive power." *Ecotoxicology* 21: 1828–40. https://doi.org/10.1007/s10646-012-0917-0.

OECD. 2006. *Current Approaches in the Statistical Analysis of Ecotoxicity Data*. https://doi.org/10.1787/9789264085275-en.

———. 2012. *Daphnia Magna Reproduction Test (OECD TG 211)*. https://doi.org/10.1787/9789264304741-12-en.

———. 2016. *Test No. 243: Lymnaea Stagnalis Reproduction Test*. https://doi.org/10.1787/9789264264335-en.