`rbioacc`: an R-package to analyse toxicokinetic data

Aude Ratier* Virgile Baudrot* Miléna Kaag* Aurélie Siberchicot*
Christelle Lopes* Sandrine Charles*†

08 September 2021

Summary

The R package `rbioacc` is dedicated to the analysis of experimental data collected from bioaccumulation tests. It provides ready-to-use functions to visualise a data set and to estimate bioaccumulation metrics to be further used in support of environmental risk assessment, in full compliance with regulatory requirements. Such metrics are classically requested by standardised regulatory guidelines on which national agencies base their evaluation of applications for marketing authorisation of chemical active substances.

Package `rbioacc` can be used to get estimates of toxicokinetic (TK) parameters (uptake and elimination rates) and bioaccumulation metrics (e.g., BCF, BSAF, BMF) by fitting a one compartment TK model on exposure-depuration test data. The bioaccumulation metrics estimates as well as the parameters and the predictions of the internal concentrations are given with the quantification of their uncertainty.

This paper illustrates some classical uses of `rbioacc` with internal concentrations collected over time possibly at several exposure concentrations, analysed with a generic TK one-compartment model. These examples can be followed step-by-step to analyse any new data set, as long as the data set format is respected.

Statement of need

Package `rbioacc` (Baudrot et al. 2021) has been tested using R (version 4.1.0 and later) on Linux and Windows machines. Regarding the particular case of TK models, package `rbioacc` was compared with published results considering other TK implementations under different software platforms. Giving very similar results than the other implementations, package `rbioacc` was thus confirmed as fit-for-purpose in fitting TK models on bioaccumulation test data. All functions in package `rbioacc` can be used without a deep knowledge of their underlying probabilistic model or inference methods. Rather, they were designed to behave as well as possible, without requiring the user to provide values for some obscure parameters. Nevertheless, models implemented in `rbioacc` can also be used as a first step to create specially new models for more specific situations. Note that package `rbioacc` benefits from a web interface, $\text{MOSAIC}_{\text{bioacc}}$, from which the same analyses can be reproduced directly on-line without needs to invest in R programming. $\text{MOSAIC}_{\text{bioacc}}$ is freely available on the MOSAIC platform at https://mosaic.univ-lyon1.fr/ (Charles et al. 2021) or directly at https://mosaic.univ-lyon1.fr/bioacc (Ratier et al. 2021).

Availability

Package `rbioacc` is available as an `R` package (with `R >= 4.1.0`) ; it can be directly downloaded from CRAN https://CRAN.R-project.org/package=rbioacc, where package dependencies and system requirements are also documented.

*University of Lyon, University Lyon 1, CNRS UMR5558, Laboratory de Biometry and Evolutionary Biology, 69100 Villeurbanne, France.
†Corresponding author, sandrine.charles@univ-lyon1.fr

Main features

The main functions in package `rbioacc` are `data()`, `modelData()` and `modelData_ode()` to format and visualise raw data as well as to build the corresponding TK model, `fitTK()` which allows to fit a model on data in order to estimate the bioaccumulation metrics and the kinetic parameters. Fitting outputs can be either displayed with `plot()` or synthesized with `quantile_table()`. For the bioaccumulation metrics, both functions `bioacc_metric()` and `plot()` can be run to obtain the plot of bioaccumulation metric density, or `quantile()` to obtain a summary of the distribution. Also, to get the time where 95% of the substance is eliminated, the function `t95()`can be used. The equations can be obtained by the function `equations()`, after the fitting process. Functions are available to check the goodness-of-fit criteria, namely `ppc()`, `plot_PriorPost()`, `corrMatrix()`, `corrPlot()`, `psrf()`, `waic()`, `mcmcTraces()`. At last, functions `predict()` and `predict_manual()` allow to perform predictions with or without previous observed data.

The `rbioacc` package currently handled constant or variable exposure concentrations data. It provides a generic workflow:

1. create and format a data set;
2. explore a data set;
3. plot a data set;
4. fit a TK model on data, get the model equations, the parameter estimates and the bioaccumulation metrics (BCF, BSAF and/or BMF);
5. check the goodness-of-fit criteria (the PPC percentage and the Widely Applicable Information Criterion (WAIC);
6. do predictions and/or validation.

Those steps are described in details in the `Tutorial` available at https://doi.org/10.5281/zenodo.5092316, including the description on how to use all `rbioacc` features. More information on the model and inference process used in this package are given in a document called "User Guide" available at http://lbbe-shiny.univ-lyon1.fr/mosaic-bioacc/data/user_guide.pdf and in the paper of the web interface (Ratier et al. 2021). Please refer to this documentation for further introduction to the use of the `rbioacc` package.

Minimal Working Examples

Loading `rbioacc` and its dependencies

In order to use package `rbioacc`, you need to install it with all its dependencies, including STAN and C++ (see below), as well as other R-packages: mandatory ones (`ggplot2, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), rstantools (>= 2.1.1), testthat, ggmcmc, GGally, loo, stringr, BH (>= 1.66.0), RcppEigen (>= 0.3.3.3.0), StanHeaders (>= 2.18.0)`) and suggested ones (`knitr`, `rmarkdown`). For this purpose, you can use the two classical `R` commands:

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

Stan

The `rbioacc` package is linked to Stan https://mc-stan.org/ that is the Bayesian sampler used to perform inference with all implemented models, which require to install the R packages `rstan` and `rstantools`.

C++

The `rbioacc` package is also linked to C++. C++ is used for speeding up calculations and for running simulations leading to predictions. In R, you should not have issues with C++ requirements since it is very well integrated.

Toxicokinetic analysis

To illustrate the use of `rbioacc`, we will use two standard bioaccumulation data sets, directly available in the package. The first example is data collected from a laboratory bioaccumulation test on fish *Oncorhynchus promelas* exposed to three different concentrations of a highly hydrophobic compound spiked water during 49 days. There is one replicate per concentration. The internal concentration is monitored at several time points (Crookes and Brooke 2011). The second example is from another laboratory bioaccumulation test where *Chironomus tentans*, a freshwater invertebrate, was exposed to benzo-(a)-pyrene spiked sediment for three days. Only one exposure concentration was tested with two replicates. The internal concentrations for both parent compound and its metabolite are monitored at several time points (Schuler et al. 2003).

For Environmental Risk Assessment (ERA), some OECD test guidelines (*e.g.*, test n°305 (OECD 2012) or 315 (OECD 2008)) delivers the workflow to obtain the bioaccumulation metrics, which are decision criteria to define the capacity of a substance to be bioaccumulated within organism. This framework can also be performed on-line with the MOSAIC web platform, especially the $\text{MOSAIC}_{\text{bioacc}}$ module (https://mosaic.univ-lyon1.fr/bioacc). In order to be in full compliance with these guidelines, the *modus operandi* with package `rbioacc` to be followed step-by-step is given below.

Calibration step

For the first example, three exposure concentrations were tested; thus it is required to select the desired exposure concentration to perform analysis.

Data and inference process

```
# For example 1 ####

### load package `rbioacc`
library(rbioacc)
### load a data set
data("Oncorhynchus_two")
### create a rbioacc object for data analysis with the exposure concentration to test
data <- Oncorhynchus_two[Oncorhynchus_two$expw == 0.00440,]
### build the TK model according to data
modeldata <- modelData(data, time_accumulation = 49) # indicate the time of the end of exposure
### fit a TK model
m1 <- fitTK(modeldata)
```

```
# For example 2 ####

### load the data set of the second example
data("Chironomus_benzoapyrene")
### create a rbioacc object for data analysis
modeldata <- modelData(Chironomus_benzoapyrene, time_accumulation = 3)
### fit a TK model
m2 <- fitTK(modeldata, iter = 10000)
```

Results The major results provided with `rbioacc` are the fitted predictions of internal concentrations compared to observed data against time with the function `plot()`. The probability distribution of the bioaccumulation metric(s), whatever the exposure source(s) and the elimination process(es) is given by the

function `bioacc_metric`. For the two examples, this function is used with the option to ask for the kinetic bioaccumulation metric (provided by default) or the steady-state bioaccumulation metric (if the data have reached the steady-state at the end of the accumulation phase).

It is also possible to obtain a summary of the TK parameter distributions with the function `quantile_table()`. The package also provides the equations of the model used according to input data with the function `equations()`.

```
# For example 1 ####

### plot the fitting result
plot(m1)

### get the bioaccumulation metric
BCFk_all <- bioacc_metric(m1,"k") # for kinetic bioaccumulation metric
### get the summary of the kinetic bioaccumulation metric(s) distribution(s)
for(i in 1:ncol(BCFk_all)){
BCFk <- quantile(BCFk_all[,i], c(0.5, 0.025,0.975))
}
# Here the data have not reached steady-state
BCFk <- t(cbind(BCFk))
BCFk
### Plot the output
plot(BCFk_all)

### get the kinetic parameters estimates of the model
quantile_table(m1)

### get the time for which 95% of the substance is eliminated
t95(m1)

### get the equations of the TK model adapted to data
equations(m1,Oncorhynchus_two)
```

```
# For example 2 ####

### plot the fitting result
plot(m2)

### get the bioaccumulation metric(s)
BSAFk_all <- bioacc_metric(m2,"k") # for kinetic bioaccumulation metric
BSAFss_all <- bioacc_metric(m2,"ss") # for steady-state bioaccumulation metric
### get the summary of the kinetic bioaccumulation metric(s) distribution(s)
for(i in 1:ncol(BSAFk_all)){
BSAFk <- quantile(BSAFk_all[,i], c(0.5, 0.025,0.975))
}
for(i in 1:ncol(BSAFss_all)){
BSAFss <- quantile(BSAFss_all[,i], c(0.5, 0.025,0.975))
}
BSAF <- t(cbind(BSAFk,BSAFss))
BSAF
### Plot the output
plot(BSAFk_all)
plot(BSAFss_all)
```

```
### get the kinetic parameters estimates of the model
quantile_table(m2)

### get the time for which 95% of the substance is eliminated
t95(m2)

### get the equations of the TK model adapted to data
equations(m2,Chironomus_benzoapyrene)
```

The major results provide with `rbioacc` are the fitted predictions of internal concentrations with observed data, as illustrated in Figures 1 and 2.

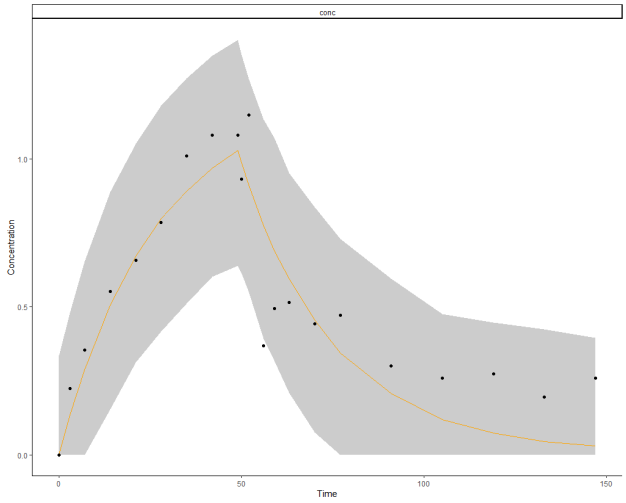

Figure 1: Outputs provided by `rbioacc` from the `plot(fit)` function for the example `Oncorhynchus_two`. Black dots are the observed data. The predicted median is symbolised by the orange curve and the 95% credible interval by the gray area.

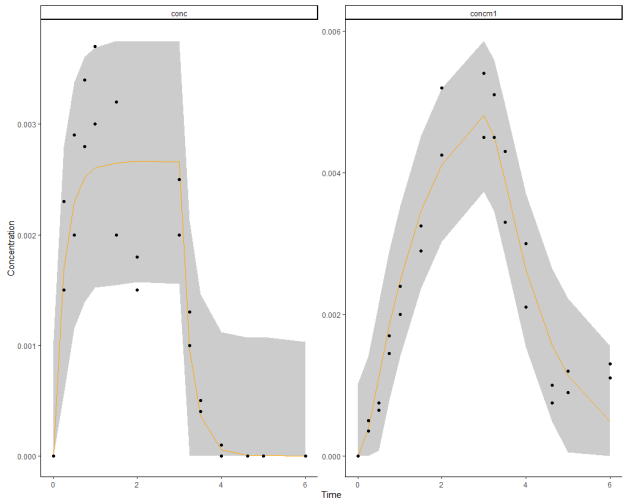

Figure 2: Outputs provided by `rbioacc` from the `plot(fit)` for the example `Chironomus_benzoapyrene` for both (a) parent compound and (b) for the metabolite. Black dots are the observed data. The predicted medians are represented by the orange curves and their uncertainty by the gray areas.

Goodness-of-fit (GOF) criteria Once the predictions of the internal concentrations are visually checked with the function `plot()`, several goodness-of-fit criteria required to be checked. For example, the Posterior Predictive Check (PPC) plot allows to compare each observed value against a prediction from the fitted model at the corresponding concentration and the associated 95% credible interval. If the fit is correct, it is expected to have 95% of the observed values inside the credible intervals. With `rbioacc`, the PPC can be obtained by the function `ppc()`, as illustrated in Figure 3. Other GOF can be checked, such as the comparison of prior and posterior distributions with the function `plot_PriorPost()`, the matrix correlation of parameters (`corrMatrix()` and `corrPlot()`), the potential scale reduction factor (PSRF) for each parameter with the function `psrf()`, the traces of the MCMC for each parameter with `mcmcTraces()`, and the Widely Applicable Information Criterion (WAIC) with the `waic()`function.

```
# For example 1 ####
### get the PPC
ppc(m1)
ppc(m1)$labels$subtitle # to get the %
### get the plot of prior and posterior distributions for each parameter
plot_PriorPost(m1)
### get the correlations between parameters
corrMatrix(m1)
corrPlot(m1)
### get the R_hat value for each parameter
psrf(m1)
### get the traces of MCMC chains
mcmcTraces(m1)
### get the WAIC
waic(m1)
```

```
# For example 2 ####
### get the PPC
ppc(m2)
ppc(m2)$labels$subtitle # to get the %
### get the plot of prior and posterior distributions for each parameter
plot_PriorPost(m2)
### get the correlations between parameters
corrMatrix(m2)
corrPlot(m2)
### get the R_hat value for each parameter
psrf(m2)
### get the traces of MCMC chains
mcmcTraces(m2)
### get the WAIC
waic(m2)
```

Validation step

Validation consists in predicting the internal concentration over time for which observations have also been collected. Predictions are then compared to observations and their adequacy is checked according to several validation criteria, in particular those defined by EFSA (EFSA panel et al. 2018). Then more predictions can be done even if no data exists as it was validated for several exposure concentrations, which is less time-consuming and less expensive.

For example, a data set with three exposure concentrations illustrates the validation step. *Spirostomum* is exposed to the pharmaceutical product fluoxetine at 0.025 $\mu g.mL^{-1}$ by spiked water for six days for the calibration step (Nalecz-Jawecki et al. 2020). The calibration step (i.e., parameters estimation) is performed with the 0.025 $\mu g.mL^{-1}$ exposure concentration. Then the prediction analysis is done for an other exposure

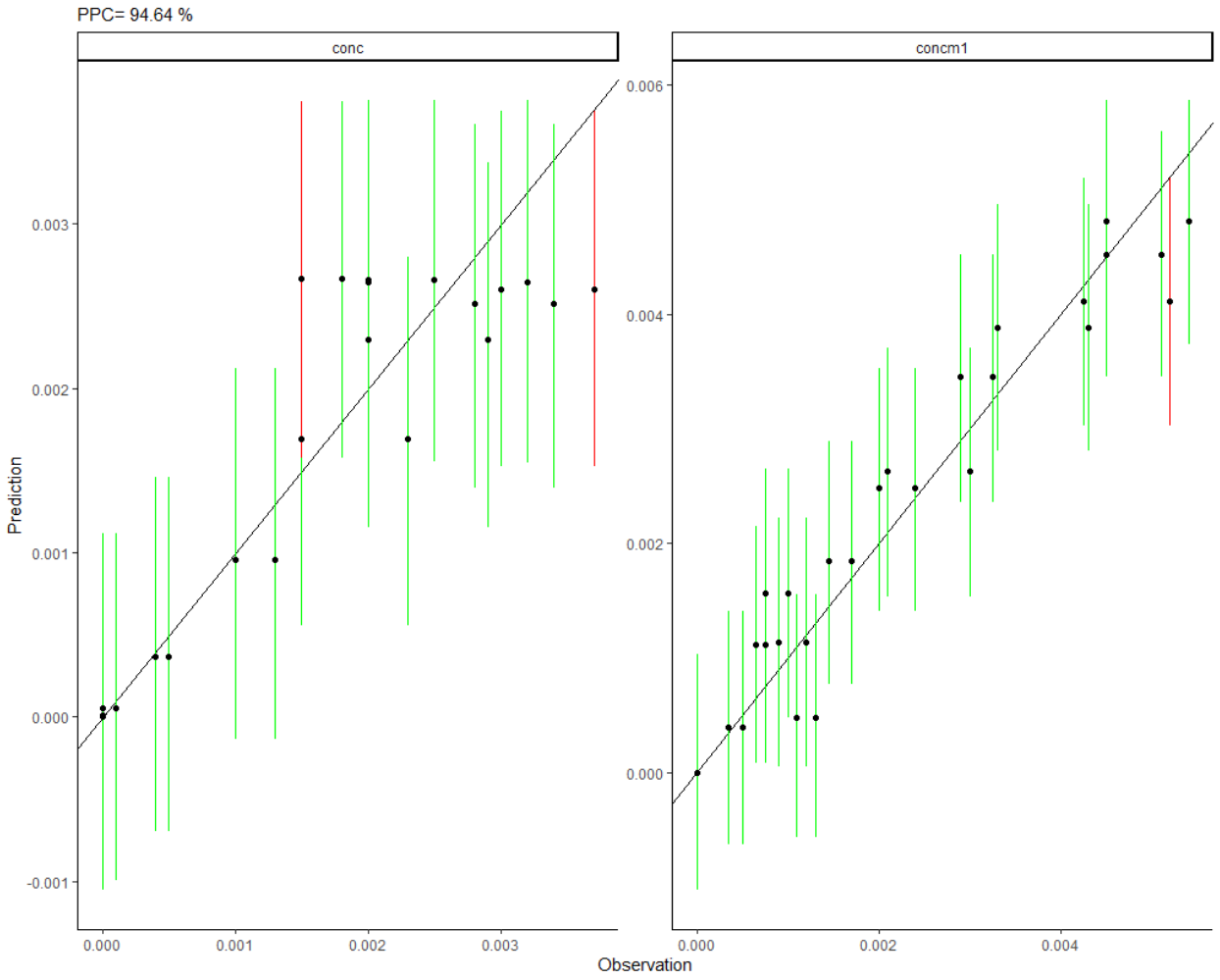

Figure 3: Outputs given by the function `ppc()` for the example `Chironomus_benzoapyrene` for both parent compound (left panel) and for the metabolite (right panel).

concentration for which data are available (e.g., at 0.1 $\mu g.mL^{-1}$). Then for the validation process, the corresponding experimental data for this exposure profile are plotted over predictions, as illustrated on Figure 4.

```
# create the dataframe with collected data at 0.025 µg/mL
df_validation_0.025 <- structure(list(
time = c(0, 0.083, 1, 2, 6, 7, 8, 12, 0, 0.083, 1, 2, 6, 7, 8, 12),
conc = c(0, 1.2, 2.94, 3.57, 4.65, 1.52, 1.73, 1.89, 0, 4.7, 11.04,
    6.65, 10.91, 3.37, 3.13, 2.88),
replicate = c(1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2),
expw = c(0.025, 0.025, 0.025, 0.025, 0.025, 0.025, 0.025, 0.025, 0.025, 0.025,
    0.025, 0.025, 0.025, 0.025, 0.025, 0.025)),
    row.names = c(NA, -16L), class = c("tbl_df", "tbl", "data.frame"))

data <- df_validation_0.025

### build the TK model according to data
modeldata <- modelData(data, time_accumulation = 6) # indicate the time of the
                                                    # end of exposure
### fit the TK model
m3 <- fitTK(modeldata)

# create the dataframe with collected data at 0.1 µg/mL
df_validation_0.1 <- structure(list(
time = c(0, 0.083, 1, 2, 6, 7, 8, 12, 0, 0.083, 1, 2, 6, 7, 8, 12),
conc = c(0, 2.95, 10.02, 15.45, 12.82, 5.75, 5.08, 3.15, 0, 14.66, 29.54,
    30.51, 28.65, 15.46, 11.77, 9.07),
replicate = c(1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2),
expw = c(0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
    0.1, 0.1, 0.1)),
    row.names = c(NA, -16L), class = c("tbl_df", "tbl", "data.frame"))

library(dplyr)
# format data for predictions
data_4pred <- data.frame(unique(df_validation_0.1 %>% select(time,expw)))
# perform the predictions
predict_m3 <- predict(m3, data_4pred)
#plot the predictions
p <- plot(predict_m3)
p

# format data for validation
data_4val <- data.frame(unique(df_validation_0.1 %>% select(time,conc)))
#plot data over predictions
validation <- p + geom_point(data = data_4val, aes(x = time, y = conc))
validation
```

Prediction step

If the validation step is reasonable, then the prediction step can be performed for other simulations with no tested exposure concentrations for example, or a different accumulation time. To do this, both functions `predict()` or `predict_manual()` can be used, for example here with an exposure at 0.05 $\mu g.mL^{-1}$ (Figure 5).

Another possibility is that only mean or median value for each parameter is given in the scientific literature,

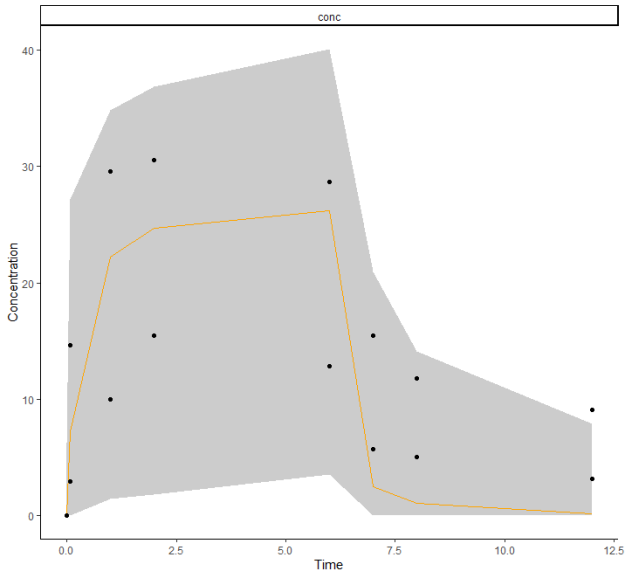

Figure 4: Example of the validation step for *Spirostomum* exposed to fluoxetine at 0.1 $\mu g.mL^{-1}$. Orange curve is the predicted median of internal concentration with the TK parameters estimated from the exposure concentration at 0.025 $\mu g.mL^{-1}$, the gray area is the 95% credible interval of predictions, and black dots are the observed data for the exposure at 0.1 $\mu g.mL^{-1}$.

without raw TK data provided. In this case, the validity step can't be performed, but predictions can be done in order to plan a new experiment according the previous values found in scientific literature or from an older experiment in the laboratory. Thus, the function `predict_manual()` can be used, as illustrated in Figure 6 with only the median value of parameter from the example 1 (Crookes and Brooke 2011).

```
# With predict() function
# prepare data for prediction for an exposure concentration at 0.05 µg/mL with a total
# duration of the experiment of 15 days
data_4pred2 <- data.frame(time = 1:15, expw = 0.05)
plot(predict(m3, data_4pred2))


# With predict_manual() function
parfit_MGSG <- rstan::extract(m3[["stanfit"]]) #extract posterior table for each parameter
mcmc_m3 = data.frame(
  kee = parfit_MGSG$ke[,1],
  kuw = parfit_MGSG$ku[,1],
  sigmaConc = parfit_MGSG$sigmaCGpred[,1]
)

data_4pred2 <- data.frame(time = 1:15, expw = 0.05)
plot(predict_manual(mcmc_m3, data_4pred2))
```

```
# prepare data for prediction for an exposure concentration at 0.01 µg/mL with a total
# duration of the experiment of 75 days
data_4pred <- data.frame(time = 1:75, expw = 0.01)
# perform predictions from a previous fit (distributed parameters)
predict_m1 <- predict(m1, data_4pred)
# perform predictions from median or mean values of parameters (not distributed)
predict_m1_manual <- predict_manual(param=data.frame( # from median parameters
```

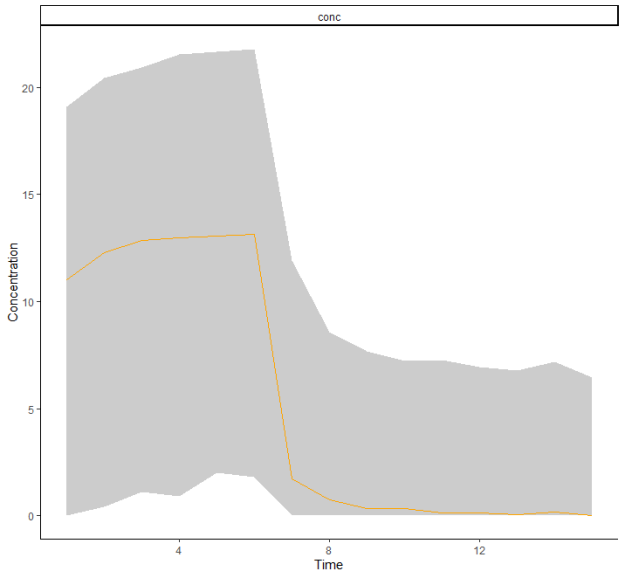

Figure 5: Example of the prediction step for *Spirostomum* exposed to fluoxetine at 0.05 $\mu g.mL^{-1}$. Orange curve is the predicted median of internal concentration with the TK parameters estimated from the exposure concentration at 0.025 $\mu g.mL^{-1}$ and the gray area is the 95% credible interval of predictions.

```
    kee = 0.03834562,
    kuw = 10.56466351
  ), data_4pred, time_accumulation = 49)
### plot predictions
plot(predict_m1)
plot(predict_m1_manual)
```

Bioaccumulation under any variable exposure profile

Finally, it may be useful to predict the internal concentration under any exposure profile (time-variable or not), as in the environment exposure concentrations are more fluctuating than constant (e.g., for field data collected). First, it requires to load two types of data instead of one for constant exposure: one for variable exposure concentrations (at least two columns, `time` and `Cwater`) and one for internal concentration (at least two columns, `time` and `Cinternal`). Secondly, it requires to call the function `modelData_ode()` which will perform calculations of ordinary differential equations for the variable exposure profile. Then the other functions can be used as previously to obtain all the results and the GOF criteria. The following code performs inference on a data set of *Sialis lutaria* exposed to a time-variable exposure profile of chlorpyrifos spiked water for two days (Rubach et al. 2010).

```
data("Exposure_Sialis_lutaria")
data("Internal_Sialis_lutaria")
  Exposure_Sialis_lutaria$value = Exposure_Sialis_lutaria$Cwater
  Internal_Sialis_lutaria$value = Internal_Sialis_lutaria$Cinternal

  modeldata_SL <- modelData_ode(Exposure_Sialis_lutaria, Internal_Sialis_lutaria,
  time_accumulation = 2.170)
  fit_SL <- fitTK(modeldata_SL, iter = 100)
  quantile_table(fit_SL)
```

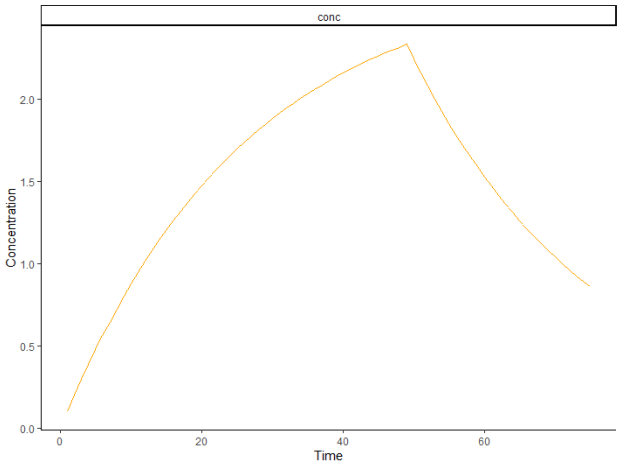

Figure 6: Example of the prediction step for fish *Oncorhynchus promelas* exposed to a very hydrophobic substance for 49 days. Orange curve is the predicted median of internal concentration with the TK parameters provided by the study (Crookes and Brooke 2011) at an exposure concentration of 0.01 $\mu g.mL^{-1}$ (which is chosen by the user according the experimental design desired).

A web application using `rbioacc`: MOSAIC$_{\text{bioacc}}$

Short presentation

In 2020, the MOSAIC$_{\text{bioacc}}$ application (Ratier et al. 2021) has been designed to allow for an easier toxicokinetics modelling of complex situations in the field of ecotoxicology. The MOSAIC$_{\text{bioacc}}$ application exploits a one compartment model, meaning that the organism is considered as a whole. It explores a more complex TK part than what is used nowadays, where multiple exposure routes are considered (up to 4, among water, sediment, pore water and food) as well as biotransformation of the chemical in several metabolites and growth of the organisms. The application strives to offer an easy access to TK modelling, allowing the user to work with complex situations. It also provides several criteria for the user to easily assess the quality of the obtained fit. MOSAIC$_{\text{bioacc}}$ is a tool thought for the regulatory domain, but also for research, as it provides a database of accumulation-depuration data fully referenced and associated with their fitting results and reports from MOSAIC$_{\text{bioacc}}$. The web application was developed using the R package `shiny` (Chang et al. 2021), that is an interactive interface to the package `rbioacc`.

Last updates

The present section showcases the last update of the MOSAIC$_{\text{bioacc}}$ project since the last papers (Ratier et al. 2021; Charles et al. 2021). The MOSAIC$_{\text{bioacc}}$ application shifted from an all embarked web application to being a user interface of a newly developed R package: `rbioacc`, that allows all users to bring MOSAIC$_{\text{bioacc}}$ and its self-explanatory functions to their own device for an easy integration of its method to every workflow. Furthermore, new features were added: a prediction and a validation tools were added to the web application thanks to the functions implemented in `rbioacc`.

Aesthetic

Whereas the previous version of MOSAIC$_{\text{bioacc}}$ was based on a scroll of the page, this new version works with several tabs unlocked once the user performed the actions needed in the previous tab (except for the prediction (and validation) tool that is always available). Globally, the same visual elements were kept, but they got reorganised in a tabs structure with 5 different levels (Figure 7) : Data upload, Model and parameters, Results, Downloads, Prediction tool. The results of the fitting process were also compacted in a tabs and columns structure to better fit the new no-scroll policy. This tremendously reduced the length

of the section, as it can be seen in Figure 8. This revamp in a tabs based interface, as well as other minor changes, were partly decided upon by analysing the behavior of the users of the application.

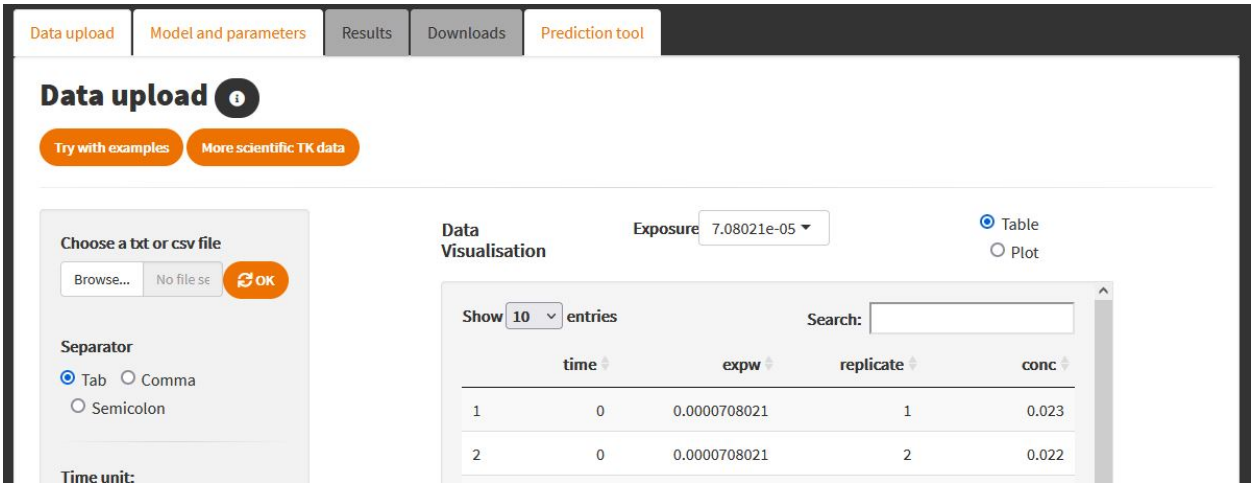

Figure 7: Visual of the first tab once a table was uploaded: the second tab is unlocked.

Architecture

The old version of the application suffered from a mix of classical R coding and reactive R coding, the biggest obstacle being the use of global variables. These are generally not meant to be used in a reactive context, but can be forced as they can be useful in very punctual cases. Using them allowed for a more classical way of coding, but it generated a series of scope related bugs, notably for the downloads, that needed to be fixed with lengthy patch ups. Furthermore, using global variables promote excessive code reevaluation, making the application less effective. Due to the long, iterative development process, other issues appeared such as code duplication, repetitive operations on variables, creation of unused variables and non explicit, temporary variable names that ended up being kept. All in all, the 'server' file required in Shiny development of the old version was approximately 4.5k lines long. For all these reasons, the file was entirely rewritten, which allows an easier code to maintain, to understand and to upgrade, by sectioning the code in multiple files, by introducing documentation with roxygen and more extensive comments through the code, and by strictly managing variable dependencies in an explicit way. With both the refactoring of the code and the integration of the `rbioacc` package, the server file became approximately 1k lines long, and 1.5k lines long by taking into account the other files containing code used by the server file. The major benefit to this restructuring was to the include the `rbioacc` package in the downloadable R code from the application, gaining clarity and lines long, easier for a user to reuse the R code.

New features

A prediction and a validation tools were added to $\text{MOSAIC}_{\text{bioacc}}$. The prediction tool allows the user to propagate the parameter values of a previous fit and their uncertainties to calculate a prediction for a different exposure profile. The users can also use this tool on its own, with their own parameter estimates. The validation tool takes things a step further by allowing the visual comparison of a predicted accumulation-depuration curve to observed validation data.

EFSA workflow with `rbioaccor` $\text{MOSAIC}_{\text{bioacc}}$

The environmental risk assessment workflow proposed by the EFSA (European Food Safety Agency) consists of three steps: the calibration, the validation and the prediction steps. $\text{MOSAIC}_{\text{bioacc}}$ can facilitate the first two steps of this workflow.

Figure 8: Same scale size comparison of the results section in the old version (left) and this new version (right)

The first step, the calibration of the model, consists of the fitting of a TK model on bioaccumulation test data to get parameter estimates, from which we can get the bioaccumulation metrics that are used as decision criteria. The application takes as input the data of bioaccumulation tests, during which organisms are exposed to a substance in the accumulation phase, and are then transferred to a clean medium (depuration phase). The internal concentration of the studied substance in the organisms is measured throughout the test duration, resulting in the test data. MOSAIC$_{\text{bioacc}}$ allows the modeling of complex TK scenarios and easily provides parameter estimates and bioaccumulation metrics, namely the bioaconcentration factor (BCF), the bio-sediment accumulation factor (BSAF) or the biomagnification factor (BMF). This method allows the simultaneous estimation of all kinetic parameters, taking into account both their correlations and uncertainties, giving better results than the other methods classically used.

The second step of the EFSA workflow is the validation step, where the internal concentration over time induced by an exposure profile is simulated and then compared to observed data from a corresponding bioaccumulation test. For this step, MOSAIC$_{\text{bioacc}}$ allows the user to visually compare the predicted TK model to observed data, which is the first part of validation. The second part would be to calculate the three validation criteria that are recommended by the EFSA (EFSA panel et al. 2018).

Finally the third step, the prediction, consists of making simulation under realistic exposure scenarios to assess the risk in real life. It is also important to define how real life scenarios are "far" from causing a defined effect, by calculating indicators like the multiplication factor (x% Lethal Profile). For this step, it is of the utmost importance for the model to have been thoroughly calibrated and validated beforehand.

Data availability

A collection of five data sets is made available directly in package `rbioacc` (use function `data()`). These data sets can also be downloaded on-line from the MOSAIC web platform by visiting the 'bioacc' module: https://mosaic.univ-lyon1.fr or directly at https://mosaic.univ-lyon1.fr/bioacc.

Research applications

`rbioacc` is designed to help stakeholders or researchers to analyse toxicokinetic data obtained from a bioaccumulation test. It is a turn-key package providing bioaccumulation metrics (BCF/BMF/BSAF) and it is designed to fulfill the requirements of regulators when examining applications for market authorization of active substances for example.

Author contributions

A.R.: supervision, formal analysis, data curation, writing manuscript. V.B. (main developer of `rbioacc`): conceptualisation, methodology, formal analysis, data curation, visualisation, writing manuscript. M.K.: data curation, conceptualisation, methodology, visualisation, writing manuscript. C.L.: supervision, formal analysis, data curation, reviewing manuscript. A.S.: conceptualisation, maintainer, formal analysis, data curation, reviewing manuscript. S.C.: supervision, funding acquisition, project administration, formal analysis, data curation, reviewing manuscript.

Acknowledgments

The authors are thankful to ANSES for providing the financial support for the development of the MOSAIC$_{\text{bioacc}}$ web tool (CNRS contract number 208483). This work is part of the ANR project APPROve (ANR-18-CE34-0013) for an integrated approach to propose proteomics for biomonitoring: accumulation, fate and multi-markers (https://anr.fr/Projet-ANR-18-CE34-0013). This work was also made with the financial support of the Graduate School H2O'Lyon (ANR-17-EURE-0018) and "Université de Lyon" (UdL), as part of the program "Investissements d'Avenir" run by "Agence Nationale de la Recherche" (ANR). Authors fully

thank all contributors to the `rbioacc` package: Théo Ciccia, Ophélia Gestin, Gauthier Multari and Alain Pavé.

References

Baudrot, Virgile, Sandrine Charles, Miléna Kaag, Aude Ratier, and Aurélie Siberchicot. 2021. *Rbioacc: Inference and Prediction of ToxicoKinetic (TK) Models*.

Chang, Winston, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert, and Barbara Borges. 2021. *Shiny: Web Application Framework for r*. https://CRAN.R-project.org/package=shiny.

Charles, Sandrine, Aude Ratier, Virgile Baudrot, Gauthier Multari, Aurélie Siberchicot, Dan Wu, and Christelle Lopes. 2021. "Taking full advantage of modelling to better assess environmental risk due to xenobiotics." *Environmental Science and Pollution Research* https://ww.

Crookes, Michael J, and David N Brooke. 2011. "Estimation of fish bioconcentration factor (BCF) from depuration data." Environmen. Bristol, UK. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment%7B/_%7Ddata/file/291527/scho0811buce-e-e.pdf.

EFSA panel, Colin, Ockleford, Paulien Adriaanse, Philippe Berny, Theodorus Brock, Sabine Duquesne, Sandro Grilli, Antonio F. Hernandez-Jerez, et al. 2018. "Scientific Opinion on the state of the art of Toxicokinetic/Toxicodynamic (TKTD) effect models for regulatory risk assessment of pesticides for aquatic organisms." https://doi.org/10.2903/j.efsa.2018.5377.

Nalecz-Jawecki, Grzegorz, Milena Wawryniuk, Joanna Giebułtowicz, Adam Olkowski, and Agata Drobniewska. 2020. "Influence of Selected Antidepressants on the Ciliated Protozoan Spirostomum Ambiguum: Toxicity, Bioaccumulation, and Biotransformation Products." *Molecules* 25 (7). https://doi.org/10.3390/molecules25071476.

OECD. 2008. "Test No. 315: Bioaccumulation in Sediment-Dwelling Benthic Oligochaetes." Paris. https://doi.org/https://doi.org/https://doi.org/10.1787/9789264067516-en.

———. 2012. "Test No. 305: Bioaccumulation in Fish: Aqueous and Dietary Exposure." Paris. https://doi.org/https://doi.org/10.1787/9789264185296-en.

Ratier, Aude, Christelle Lopes, Gauthier Multari, Vanessa Mazerolles, Patrice Carpentier, and Sandrine Charles. 2021. "New Perspectives on the Calculation of Bioaccumulation Metrics for Active Substances in Living Organisms." *Integrated Environmental Assessment and Management* n/a (n/a). https://doi.org/https://doi.org/10.1002/ieam.4439.

Rubach, Mascha N., Roman Ashauer, Stephen J. Maund, Donald J. Baird, and Paul J. Van den Brink. 2010. "Toxicokinetic Variation in 15 Freshwater Arthropod Species Exposed to the Insecticide Chlorpyrifos." *Environmental Toxicology and Chemistry* 29 (10): 2225–34. https://doi.org/https://doi.org/10.1002/etc.273.

Schuler, Lance J., Matthew Wheeler, A. John Bailer, and Michael J. Lydy. 2003. "Toxicokinetics of Sediment-Sorbed Benzo[a]pyrene and Hexachlorobiphenyl Using the Freshwater Invertebrates Hyalella Azteca, Chironomus Tentans, and Lumbriculus Variegatus." *Environmental Toxicology and Chemistry* 22 (2): 439–49. https://doi.org/https://doi.org/10.1002/etc.5620220227.