



# A simple R-based function to estimate lethal concentrations



Antonio Guilherme Pacheco<sup>a,1</sup>, Mauro de Freitas Rebelo<sup>b,\*,1</sup>

<sup>a</sup> Fundação Oswaldo Cruz, Rio de Janeiro, Brazil

<sup>b</sup> Instituto de Biofísica Carlos Chagas Filho, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil

## ARTICLE INFO

### Article history:

Received 6 February 2013

Received in revised form

1 August 2013

Accepted 2 August 2013

### Keywords:

Toxicity test

New chemicals

Probits

Ecotoxicology

## ABSTRACT

Every year thousands of new substances are synthesized and marketed. Either by regulatory demands or scientific purposes, these substances should all be tested. The determination of lethal concentration (LC) in a given population after toxicity testing in the laboratory is still the best way to establish and compare toxicity. In this technical note, we propose an R function to estimate LC with improvements on the choice of experimental design and calculation parameters, as well as in editing and spreadsheet loading. Although only LC50 values are considered herein, the same statistical considerations apply to EC50 and ED50 from quantal (dichotomous, binomial, binary) data. A running example is also provided.

© 2013 Elsevier Ltd. All rights reserved.

## 1. Introduction

According to Binetti et al. (2008) the number of substances progressively registered by the Chemical Abstract Service has increased from about 212 thousands in 1965 up to 88.7 millions in 2006, with an yearly average increase rate in the order of 15%. Either by regulatory demands or scientific purposes, these substances should all be tested and the determination of Lethal concentration (LC) in a given population after toxicity testing in the laboratory is still the best way to establish and compare toxicity. For Stephan (1977), a classical review on LC literature, data produced by these tests generally consist of organisms percentages that are killed by different concentrations of a toxicant after specified lengths of exposure. Until today, regardless of the advances in statistical software, the TRIMMED SPEARMAN-KARBER (TSK) program v.1.5; created by Hamilton et al. (1977) and further developed by the USEPA (1991) for the analysis of mortality data from acute and chronic toxicity tests, is still the standard to calculate LC50 (lethal concentration to 50% of the tested population).

In this technical note, we propose an R (R Core Team, 2012) function to estimate LC with freedom of choice on the experimental design and calculation parameters. It also addresses some of the user friendly common issues, such as editing and spreadsheet loading.

## 2. Material and methods

The open source statistics-oriented R language and environment is quickly turning into the mainstream tool for the implementation of new and traditional statistical methods. The major advantage of using R is its ability to be openly criticized by the scientific community and users across the world and very efficient network for debugging and solving problems related to the language. As such, it is ideal to give flexibility to users use and adapt functions according to their needs.

Although only LC50 values will be considered herein, the same statistical considerations generally apply to obtaining an estimate of a median effective concentration (EC50) or median effective dose (ED50) from quantal (dichotomous, binomial, binary) data, that is, data consisting of two mutually exclusive categories, such as alive or dead. It can also be used to estimate other ecotoxicologically relevant parameters such as LC10, LC20 and so on.

A running example is provided with a datasets (Table 1) from Calazans et al. (2013), relative to the mortality bivalve mollusk *Limnoperna fortunei* to NaCl.

## 3. Results and discussion

We implemented and described a function in R to calculate lethal concentrations using GLMs to accomplish model fitting. The function allows for users to choose several parameters, as described and the code can be easily modified to add or modify features, according to one's needs.

\* Corresponding author.

E-mail address: [mrebelo@biof.ufrj.br](mailto:mrebelo@biof.ufrj.br) (M.F. Rebelo).

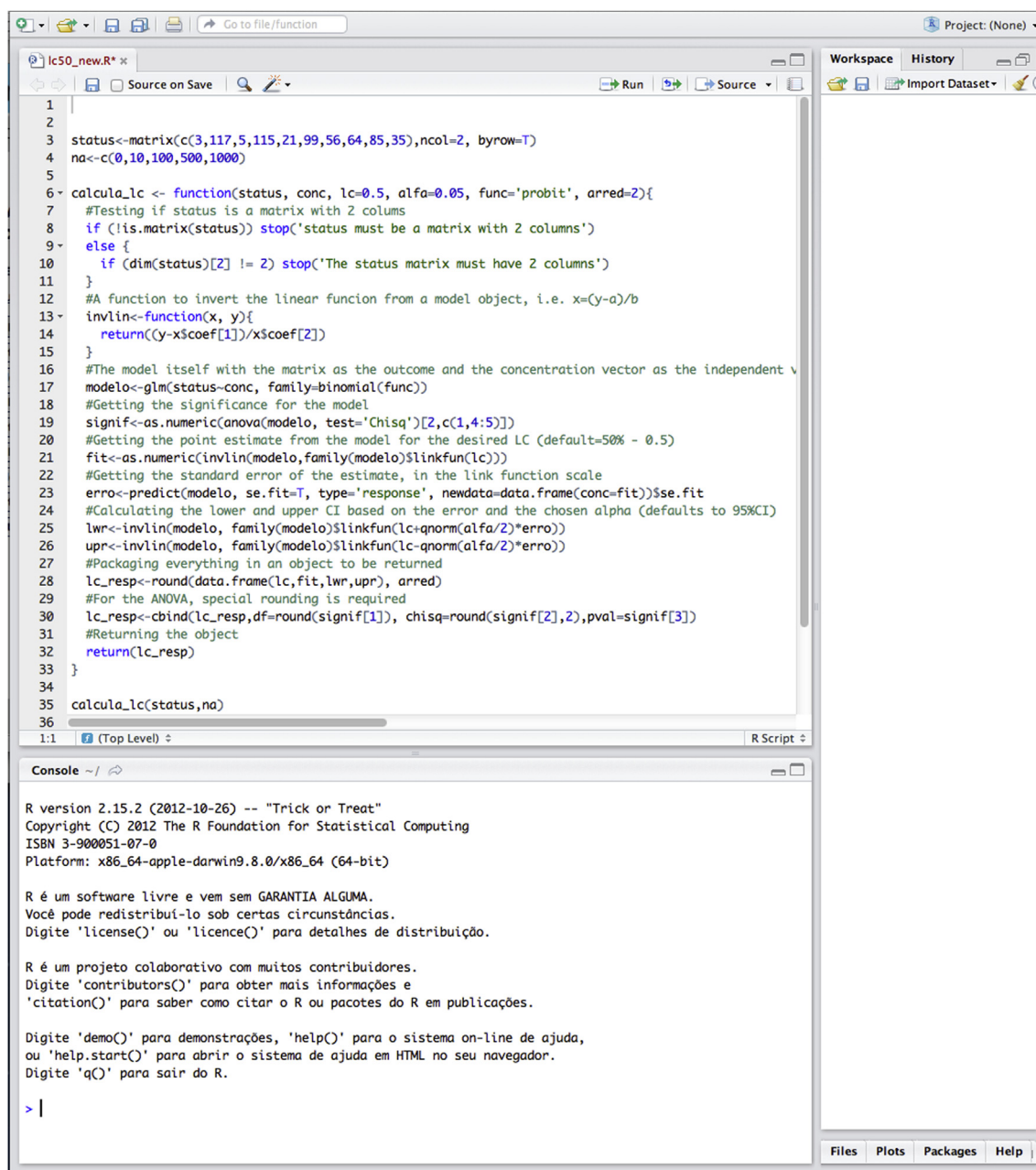
<sup>1</sup> Both authors equally contributed to this article.

**Table 1**  
Mortality of bivalve mollusk *Limnoperna fortunei* exposed to NaCl for 48 h in semi-static laboratory conditions (Calazans et al., 2013).

NaCl mg L <sup>-1</sup>	Dead	Alive	NaCl mg L <sup>-1</sup>	Dead	Alive	NaCl mg L <sup>-1</sup>	Dead	Alive
0	0	10	3000	2	8	10,000	9	1
0	0	10	3000	6	4	10,000	4	6
0	0	10	3000	2	8	10,000	6	4
1000	0	10	4000	7	3	15,000	7	3
1000	0	10	4000	1	9	15,000	7	3
1000	0	10	4000	6	4	15,000	6	4
2000	0	10	5000	6	4	20,000	10	0
2000	0	10	5000	4	6	20,000	10	0
2000	0	10	5000	5	5	20,000	10	0

The function [calcula.lc()] is provided in the appendix and it will output the desired lethal concentration (defaults to LC50) using generalized linear models (McCullagh and Nelder, 1989) with the binomial family and a link function to be chosen by the user

(defaults to probit). In this case the model would be written as:  $\eta_i = \beta_0 + \beta_1 x_i$ , where  $\eta_i$  is the transformed response (proportion dead,  $p_i$ ) by the link function – probit by default, i.e.  $\Phi^{-1}(p_i)$ ,  $\beta_0$  is the intercept,  $\beta_1$  is the coefficient of the dose and  $x_i$  is the observed



**Fig. 1.** Screenshots of open source RStudio (one of many available environments to run R functions) with loaded LC50 R function (a) and DOS Trimmed executable file (B). Even though basic knowledge of R language is required to run the function, the interface is far more amenable and friendly, allowing upload of spreadsheets, value editing and undo.

dose for experiment  $i$ .  $\Phi^{-1}$  is the inverse normal cumulative distribution function.

The function is amenable for the choice of the experimental design and calculation of parameters, as well as in editing and spreadsheet loading, as shown in the snapshot (Fig. 1). Specially if compared to the non-editing command line interface of TSK.

The output includes the best fit, along with a confidence interval (default 95%), calculated by normal approximation, based on the calculated standard deviation calculated at the desired proportion, i.e.  $p_d \pm 1.96 \times SE(p_d)$  and the overall model significance, including degrees of freedom, chi-square and  $p$  values for the deviance test. In our case the deviance test is just the chi-square approximation of the difference between the deviance of the null model (with the intercept only)  $D_{M_0}$ , the model deviance  $D_M$  over the scale parameter, taken to be 1 in these models, thus  $D_{M_0} - D_M \sim \chi_1^2$ . In our function that will always be a chi-square with one degree of freedom corresponding to the linear dose variable.

In the example, the function was successfully used to calculate the LC50 values for bivalve mollusk *L. fortunei* exposed to NaCl for 48 h: LC50 was 8336.7 mg L<sup>-1</sup> with upper and lower 95%

confidence intervals of 7075.1 and 9598.3 mg L<sup>-1</sup>. Chi-square goodness of fit was calculated with 25 degrees of freedom with chi-square statistic of 68.96 and  $p$ -value of  $3.32 \times 10^{-27}$ .

The implementation of this method is smooth and all the datasets used to test it [including Calazans et al., 2013] yielded reasonable results if compared to visual observation of the toxicity curves and TSK calculated data (see Calazans et al., 2013). The LC50 R function does not limit test concentrations to 10, like TSK. However, important TSK limitations, like consideration of technical replicates and parameter overdispersion when using the binomial family, are not covered by this implemented function. Nonetheless, they should be addressed with the use of models that account for those problems such as generalized linear mixed effects models with real or simulated data to study its impact.

Finally, this function is implemented and made available for the R environment, which is free, open source, and widely used by the scientific community in several areas.

#### Appendix. The calcula.lc() function

```
calcula.lc <- function(status, conc, lc=0.5, alfa=0.05,
func='probit', arred=2){
  #Testing if status is a matrix with 2 colums
  if (!is.matrix(status)) stop('status must be a matrix with 2
columns')
  else {
    if (dim(status)[2] != 2) stop('The status matrix must have 2
columns')
  }
  #A function to invert the linear function from a model object, i.e.
x=(y-a)/b
  invlin<-function(x, y){
    return((y-x$coef[1])/x$coef[2])
  }
  #The model itself with the matrix as the outcome and the
concentration vector as the independent variable - the link function
is probit by default, but other links of the binomia family can be
used, e.g. logit
  modelo<-glm(status~conc, family=binomial(func))
  #Getting the significance for the model
  signif<-as.numeric(anova(modelo, test='Chisq')[2,c(1,4:5)])
  #Getting the point estimate from the model for the desired LC
(default=50% - 0.5)
  fit<-as.numeric(invlin(modelo,family(modelo)$linkfun(lc)))
  #Getting the standard error of the estimate, in the link function
scale
  erro<-predict(modelo, se.fit=T, type='response',
newdata=data.frame(conc=fit))$se.fit
  #Calculating the lower and upper CI based on the error and the
chosen alpha (defaults to 95%CI)
  lwr<-invlin(modelo, family(modelo)$linkfun(lc+qnorm(alfa/2)*erro))
  upr<-invlin(modelo, family(modelo)$linkfun(lc-qnorm(alfa/2)*erro))
  #Packaging everything in an object to be returned
  lc_resp<-round(data.frame(lc,fit,lwr,upr), arred)
  #For the ANOVA, special rounding is required
  lc_resp<-cbind(lc_resp,df=round(signif[1]),
chisq=round(signif[2],2),pval=signif[3])
  #Returning the object
  return(lc_resp)
}
```

## An example

```
status<-
matrix(c(rep(0,9),2,6,2,7,1,6,6,4,5,9,4,6,7,7,6,10,10,10,rep(10,9),8,
4,8,3,9,4,4,6,5,1,6,4,3,3,4,0,0,0),ncol=2, byrow=F)
na<-rep(c(0,1000,2000,3000,4000,5000,10000,15000,20000), each=3)
calcula.lc(status,na)
```

## References

- Binetti, R., Costamagna, F.M., Marcello, I., 2008. Exponential growth of new chemicals and evolution of information relevant to risk control. *Annali dell'Istituto Superiore di Sanità* 44, 13–15.
- Calazans, S.H.C., Americo, J.A., Fernandes, F.C., Aldridge, D., Rebelo, M.F., 2013. Assessment of toxicity of dissolved and microencapsulated biocides for control of the Golden Mussel *Limnoperna fortunei*. *Marine Environmental Research* 91, 104–108.
- Hamilton, M.A., Russo, R.C., Thurston, R.V., 1977. Trimmed Spearman–Kärber method for estimating median lethal concentrations in toxicity bioassays. *Environmental Science and Technology* 11, 714.
- McCullagh, P., Nelder, J.A., 1989. *Generalized Linear Models*. Chapman and Hall, London.
- R Core Team, 2012. R: a Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0. URL <http://www.R-project.org/>.
- Stephan, C.E., 1977. Methods for calculating an LC50. In: Mayer, F.L., Hamelink, J.L. (Eds.), *Aquatic Toxicology and Hazardous Evaluation*. American Society for Testing and Materials STP 634, pp. 65–84.
- USEPA, 1991. *Methods for Measuring the Acute Toxicity of Effluents and Receiving Waters to Freshwater and Marine Organisms*. Environmental Monitoring Systems Laboratory, U. S. Environmental Protection Agency, Cincinnati, Ohio.