Using R to analyse additive designs

Maxwel C Oliveira, Gustavo AM Pereira, Evander A Ferreira, José B Santos, Stevan Z Knezevic, and Rodrigo Werle

January 24, 2018

Supporting Information of manuscript **Additive design: the concept and data analysis**, Weed Research.

Correspondence: Maxwel Coura Oliveira, Department of Agronomy, University of Wisconsin-Madison, Madison, WI, USA 53506. Tel: (+1) 608-262-7130; E-mail: max.oliveira@wisc.edu, maxwelco@gmail.com

Download R and Rstudio

At https://www.r-project.org/, go to download in the left column, click in CRAN and select a location near you. For example, in Brazil, you can select seven different areas. Then download **R** compatible with your system (Linux, Mac or Windows). Once you have downloaded **R**, you should also download **RStudio**. **RStudio** is a friendly interface for programers. Scroll down to **RStudio** and click on Download. Choose the free version of **RStudio**; there will also be **RStudio** version for Linux, Mac, and Windows.

Create an Rstudio file

Open **RStudio** at the toolbar, click in the file, New Project. . . , Existing Directory, and choose the folder that contains your data. Your data file has to be in that selected folder. Also, we recommend your data to be saved as csv (comma delimited) file.

Load the data in Rstudio

Assign the name of your data set (replace "DMT" to a name of your choice). If you use a comma (,) for separating decimals places, use *read.csv2()*. If you use a period (.), use *read.csv()*. In parentheses, write the name of your data set file.

```
DMT=read.csv("dmshoot.csv")
```

The command *head* prints the first six lines of the data set. It is useful for double checking your data.

```
head(DMT)
```

```
##   block treat densitycrop densityweed biomass     yl weed
## 1     1     1           1           0   59.44 -16.38    1
## 2     2     1           1           0   34.39  32.66    1
## 3     3     1           1           0   56.69 -11.00    1
## 4     4     1           1           0   53.77  -5.28    1
## 5     1     2           1           1   12.70  75.13    1
## 6     2     2           1           1   13.95  72.69    1
```

The command *str* prints how **RStudio** is reading the characters vector in a data set.

```
str(DMT)
```

```
## 'data.frame':    40 obs. of  7 variables:
##  $ block      : int  1 2 3 4 1 2 3 4 1 2 ...
##  $ treat      : int  1 1 1 1 2 2 2 2 3 3 ...
##  $ densitycrop: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ densityweed: int  0 0 0 0 1 1 1 1 2 2 ...
##  $ biomass    : num  59.4 34.4 56.7 53.8 12.7 ...
##  $ yl         : num  -16.38 32.66 -11 -5.28 75.13 ...
##  $ weed       : int  1 1 1 1 1 1 1 1 1 1 ...
```

Rectangular hyperbola model

The empirical model:

$$Y = \frac{I * x}{1 + (\frac{I}{A}) * x}$$

is the standard model to describe additive competition studies. I represents the slope of Y (yield loss) when x (weed density) approximate zero. Also, A is the asymptote or maximum expected yield loss (%).

Step 1) Fit a full model, a rectangular hyperbola with 4 parameters

Full is a user-defined name that will contain all information about the fitted model generated by *nls* (nonlinear least squares) function. The *start* is used to estimate values of parameter I and A for the model. Parameters can determine from visual inspection of the data set (plotting data and observing trends). The brackets [weed] for each parameter in the equation tell **R** to estimate a parameter for each weed species (4 parameters).

```
Full = nls(yl ~ (I[weed]*densityweed)/(1+(I[weed]/A[weed])*densityweed), data=DMT,
           start=list(I=c(60,30), A=c(80,60)), trace=T)
```

```
## 33221.22 :  60 30 80 60
## 9405.813 :  165.46347  50.25847  95.53698  80.57663
## 7057.126 :  209.67254  50.19545 108.37955  82.11462
## 7056.696 :  210.23750  50.25690 108.56329  82.06519
## 7056.696 :  210.22928  50.25188 108.56427  82.07029
```

Check estimated parameters

The *summary* command provides the estimated parameters I and A for each weed species, *Commelina benghalensis* (species 1) and *Richardia brasiliensis* (species 2).

```
summary(Full)
```

```
##
## Formula: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A[weed]) * densityweed)
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## I1   210.23      88.55   2.374  0.02304 *
## I2    50.25      22.64   2.220  0.03280 *
## A1   108.56      11.15   9.740 1.25e-11 ***
## A2    82.07      23.06   3.559  0.00107 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14 on 36 degrees of freedom
##
```

```
## Number of iterations to convergence: 4
## Achieved convergence tolerance: 3.438e-06
```

Step 2) Fit a reduced model (Red.1), rectangular hyperbola model with 2 parameters.

Red.1 is a user-defined name that will contain information about the first reduced model generated by the *nls* function. Notice that we do not include bracket [weed] after each parameter I and A. In this case, we are combining parameter I and A for both weed species. We hypothesize that a single parameter I and A for both species is enough to describe the crop-weed relationship (e.g., no difference of I and A between species).

```
Red.1 = nls(yl ~ (I*densityweed)/(1+(I/A)*densityweed), data=DMT,
start=list(I=40, A=80), trace=T)
```

```
## 39442.72 :  40 80
## 24501.86 :  96.10443 75.24083
## 19715.79 :  114.37075  92.64128
## 19715.78 :  114.55211  92.61749
## 19715.78 :  114.54577  92.61927
```

Check estimated parameters

This command provides the estimated parameters I and A for both weed species combined.

```
summary(Red.1)
```

```
##
## Formula: yl ~ (I * densityweed)/(1 + (I/A) * densityweed)
##
## Parameters:
##   Estimate Std. Error t value Pr(>|t|)
## I   114.55      55.93   2.048   0.0475 *
## A    92.62      15.93   5.814 1.02e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.78 on 38 degrees of freedom
##
## Number of iterations to convergence: 4
## Achieved convergence tolerance: 7.759e-07
```

Test the first hypothesis

Hypothesis testing using *ANOVA*. We test this hypothesis using the **Full** model (I and A for each species) to compare with **Red.1** (single I and A for both species). If P-value>0.05, models are similar; therefore we should use the **Red.1** model, which means that the simplest model (**Red.1**) is appropriate to describe crop-weed relationship. If not we should proceed to the next hypothesis.

```
anova(Full, Red.1)
```

```
## Analysis of Variance Table
##
## Model 1: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A[weed]) * densityweed)
## Model 2: yl ~ (I * densityweed)/(1 + (I/A) * densityweed)
##   Res.Df Res.Sum Sq Df Sum Sq F value    Pr(>F)
## 1     36     7056.7
## 2     38    19715.8 -2 -12659   32.29 9.293e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F-test showed P<0.05. Therefore the Red.1 model is not appropriate to describe the crop-weed relationship.

Step 3) Fit a reduced model (Red.2), rectangular hyperbola model with 3 parameters

Red.2 is a user-defined name that will contain information about the second reduced model generated by the *nls* function. Notice that the bracket [weed] is after the parameter A only, which means that we are testing a hypothesis of single parameter I, but different A for the species.

```
Red.2 = nls(yl ~ (I*densityweed)/(1+(I/A[weed])*densityweed), data=DMT,
            start=list(I=60, A=c(80,60)),  trace=T)
```

```
## 30094.36 :  60 80 60
## 8675.772 :  122.68194 122.78900  50.51254
## 7952.364 :  161.27367 111.80334  56.19045
## 7864.804 :  162.01662 115.63916  56.19067
## 7864.582 :  163.82772 115.27458  56.09045
## 7864.579 :  163.83130 115.28859  56.07787
## 7864.579 :  163.85015 115.28483  56.07664
```

Check estimated parameters

This command provides the estimated parameters I for both weed species and A for each weed species.

```
summary(Red.2)
```

```
##
## Formula: yl ~ (I * densityweed)/(1 + (I/A[weed]) * densityweed)
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## I   163.850     56.984   2.875  0.00666 **
## A1  115.285     12.779   9.021 7.03e-11 ***
## A2   56.077      5.714   9.813 7.65e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.58 on 37 degrees of freedom
##
## Number of iterations to convergence: 6
## Achieved convergence tolerance: 5.504e-06
```

Test a second hypothesis

Hypothesis testing using F-test. We are using the Full model (separated I and A for each species) to compare with Red.2 (single I and different A for both species). If P-value>0.05, models are similar; therefore, we should use the **Red.2** model, which means that the simplest model (**Red.2**) is appropriate to describe crop-weed relationship. If not we should proceed to the next hypothesis.

```
anova(Full, Red.2)
```

```
## Analysis of Variance Table
##
## Model 1: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A[weed]) * densityweed)
## Model 2: yl ~ (I * densityweed)/(1 + (I/A[weed]) * densityweed)
##   Res.Df Res.Sum Sq Df  Sum Sq F value  Pr(>F)
## 1     36     7056.7
## 2     37     7864.6 -1 -807.88  4.1214 0.04978 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F-test showed P<0.05. Therefore the Red.2 model is not appropriate to describe the crop-weed relationship.

Step 4) Fit a reduced model (Red.3), rectangular hyperbola model with 3 parameters

Red.3 is a user-defined name that will contain information about the third reduced model generated by the *nls* function. Notice that the bracket [weed] is after the parameter I only, which means that we are testing a hypothesis of different parameter I, but single parameter A for the species.

```
Red.3 = nls(yl ~ (I[weed]*densityweed)/(1+(I[weed]/A)*densityweed), data=DMT,
            start=list(I=c(30,30), A=70), trace=T)
```

```
## 53764.77 :  30 30 70
## 26821.46 :  128.12346  55.66148  65.14150
## 13779.51 :  265.38910  15.43032 100.87497
## 7506.945 :  205.09965  30.83279 107.29119
## 7203.363 :  225.88449  36.30998 106.26798
## 7200.048 :  228.19527  36.97054 106.19148
## 7200.044 :  228.35524  36.99872 106.17039
## 7200.044 :  228.35703  36.99971 106.16982
```

Check estimated parameters

This command provides the estimated parameters I for each weed species and A for both weed species.

```
summary(Red.3)
```

```
##
## Formula: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A) * densityweed)
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## I1  228.357    100.178   2.280   0.0285 *
## I2   37.000      6.196   5.972 6.85e-07 ***
## A   106.170     10.318  10.289 2.10e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.95 on 37 degrees of freedom
```

```
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 1.144e-06
```

Test a third hypothesis

Hypothesis testing using F-test. We are using the Full model (separated I and A for each species) to compare with Red.3 (different I and single A for both species). If P-value>0.05, models are similar; therefore we should use the **Red.3** model, which means that the simplest model (**Red.3**) is appropriate to describe the crop-weed relationship.

```
anova(Full, Red.3)
```

```
## Analysis of Variance Table
##
## Model 1: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A[weed]) * densityweed)
## Model 2: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A) * densityweed)
##   Res.Df Res.Sum Sq Df  Sum Sq F value Pr(>F)
## 1     36     7056.7
## 2     37     7200.0 -1 -143.35  0.7313 0.3981
```

Results showed that P >0.05. Therefore, the **Full** model can be simplified to **Red.3** model.

Plotting the Red.3 model

Rstudio basic figure

The command *par* is used to define the plot size. The command *plot* and *lines* are used to generate the figure, and the averaged points of yield loss at each density (Fig. 5). The command *subset* is adding each weed species separately in the plot (weed 1) and lines (weed 2).

The x is a user-defined name; it will contain the x-axis sequence of the data set. weed1 and weed2 is also a user-defined name, and this is the equation with the previous parameter estimates I and A estimated from Red.3 model using the nls function. Notice that the parameters estimated in Red.3 model were inserted in the rectangular hyperbola model for each weed species (Figure 1).

The command *lines* will insert the previous equation into the plot. Command *lty*, *lwd*, and *col* define the line type, size, and color, respectively.

The command *legend* will add the legend to the plot area.

```
par(mar=c(5,6,2,2), mgp=c(3,1.5,0))
plot(yl~densityweed, data=DMT, subset = weed =="1", pch=16, cex=1, las=1,
     xlab=expression("Weed Density (plants pot"^-1*")"), ylim=c(-10,110),
     ylab = "Yield Loss (%)", cex.axis=1, cex.lab=1)
lines(yl~densityweed, type="p",data=DMT, subset = weed =="2", col=2, cex=1, pch=1)

x=seq(0,4,0.25)
weed1=(228.357*x)/(1+(228.357/106.170)*x)
weed2=(37.000*x)/(1+(37.000/106.170)*x)

lines(x,weed1, lty=1, lwd=1, col=1)
lines(x,weed2, lty=3, lwd=1, col=2)

legend("bottomright", legend=c("C. benghalensis", "R. brasiliensis"), text.font = 3,
       col=c(1,2), pch= c(16,1), lty=c(1,3), lwd= c(1,1), bty="n", cex=1)
```

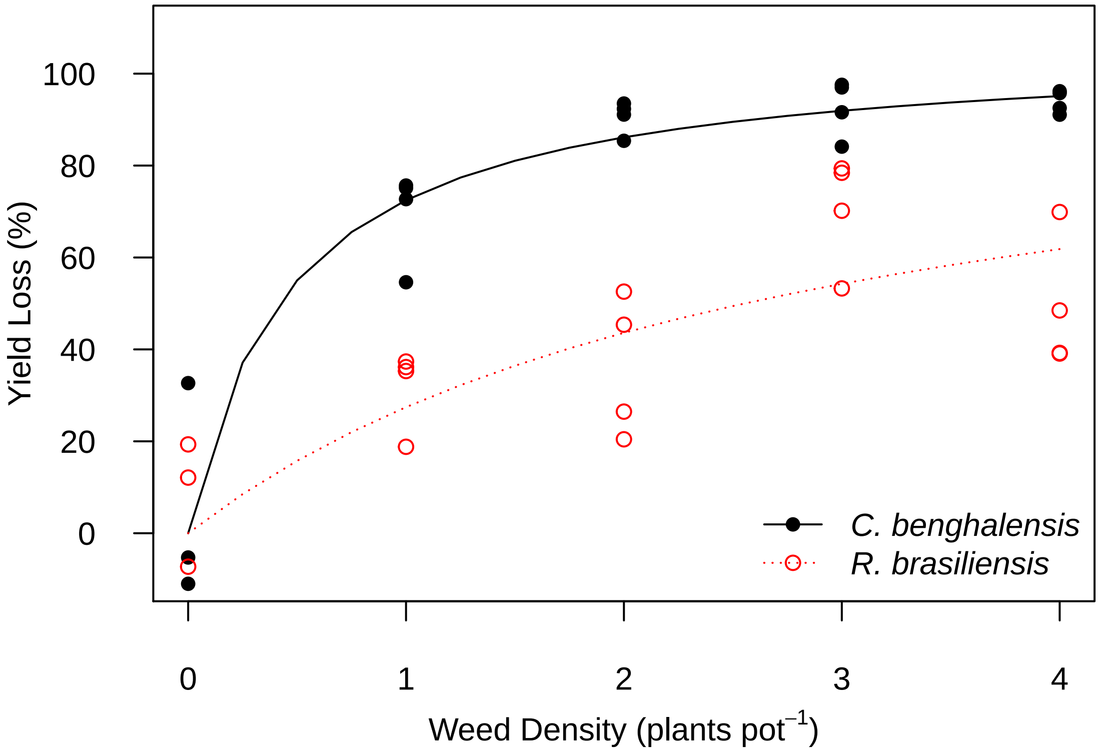

Figure 1: This figure is created with Rstudio basic commands and Red.3 model of the manuscript Additive design: the concept and data analysis.

High-quality figure in Rstudio

The package *ggplot2*, an excellent package for producing high-quality figures in **Rstudio**.

```
#install.packages("ggplot2")
#install.packages("broom")
library(ggplot2)
library(broom)
```

```
DMT$weed<-factor(DMT$weed, levels=c("1", "2"),
                 labels=c("Commelina benghalensis",
                          "Richardia brasiliensis"))


Red.3 = nls(yl ~ (I[weed]*densityweed)/(1+(I[weed]/A)*densityweed), data=DMT,
                      start=list(I=c(30,30), A=70), trace=T)
```

```
## 53764.77 :  30 30 70
## 26821.46 :  128.12346  55.66148  65.14150
## 13779.51 :  265.38910  15.43032 100.87497
## 7506.945 :  205.09965  30.83279 107.29119
## 7203.363 :  225.88449  36.30998 106.26798
## 7200.048 :  228.19527  36.97054 106.19148
## 7200.044 :  228.35524  36.99872 106.17039
## 7200.044 :  228.35703  36.99971 106.16982
```

```
summary(Red.3)
```

```
## 
## Formula: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A) * densityweed)
## 
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)    
## I1  228.357    100.178   2.280   0.0285 *  
## I2   37.000      6.196   5.972 6.85e-07 ***
## A   106.170     10.318  10.289 2.10e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 13.95 on 37 degrees of freedom
## 
## Number of iterations to convergence: 7 
## Achieved convergence tolerance: 1.144e-06
```

```
nd1 = data.frame(densityweed=seq(0, 4, 0.01), weed="Commelina benghalensis")
nd2 = data.frame(densityweed=seq(0, 4, 0.01), weed="Richardia brasiliensis")
nd = rbind(nd1, nd2)

pred<- augment(Red.3, newdata=nd)
```

```
ggplot(DMT, aes(x=densityweed, y=yl, color=weed)) + geom_point(shape=1, size=3) +
  geom_line(data = pred, size=1.3, aes(x=densityweed, linetype=weed, y=.fitted)) +
  labs(fill="", y="Yield loss (%)",
       x=expression(bold(paste("Weed density (plants pot"^"-2",")")))) +
  scale_colour_manual(values = c("red", "black"))+
  scale_y_continuous(limits=c(-25,110), breaks = c(-25,0,25,50,75,100)) +
    theme(axis.text=element_text(size=15, color="black"),
        axis.title=element_text(size=17,face="bold"),
        panel.background = element_rect(fill="white", color = "white"),
        panel.grid.major = element_line(color = "white"),
  panel.grid.minor = element_blank(),
  panel.border = element_rect(fill=NA,color="black", size=0.5,
  linetype="solid"), legend.position=c(0.7,0.15),
  legend.text = element_text(size = 12, colour = "black", face="italic"),
  legend.key = element_rect(fill=NA), legend.key.height  = unit(1.5, "line"),
  legend.key.width = unit(2.2, "line"),
legend.background = element_rect(fill =NA),  legend.title=element_blank())  +
  ggsave("Red.tiff", units="in", width=4, height=4, dpi=300)
```

Notice that the figure is created with Red.3 model (rectangular hyperbola model) using *ggplot2* package (Figure 2). This is the Fig. 5 published in the manuscript **Additive design: the concept and data analysis**.

AICc model selection and Goodness of fit

According to the AICc criterion, the top model has the lowest AICc value. The AICc calculation can be simplified using **R**, the first step is loading the package *AICcmodavg*.

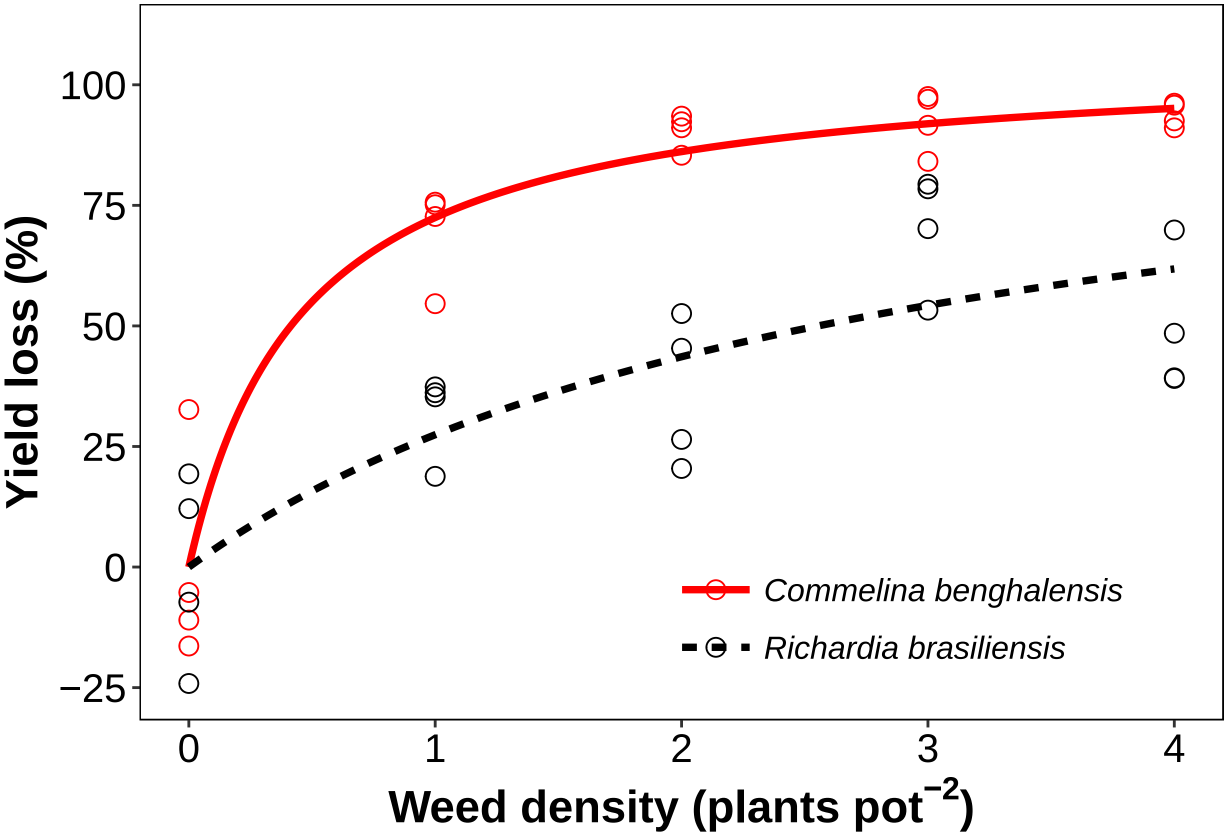

Figure 2: This is the Fig. 5 of the manuscript Additive design: the concept and data analysis. This Figure is created with package ggplot2 in Rstudio

```
library(AICcmodavg)
```

The four candidate models using the rectangular hyperbola are compared using AICc.

```
cand.mods<- list(Full, Red.1, Red.2, Red.3)

Modnames<- c('Full',' Red.1',' Red.2',' Red.3')

aictab(cand.set = cand.mods,modnames = Modnames, sort = TRUE)
```

```
##
## Model selection based on AICc:
##
##        K   AICc Delta_AICc AICcWt Cum.Wt      LL
##  Red.3 4 330.38       0.00   0.64   0.64 -160.62
## Full   5 332.19       1.82   0.26   0.89 -160.21
##  Red.2 4 333.91       3.53   0.11   1.00 -162.38
##  Red.1 3 368.19      37.82   0.00   1.00 -180.76
```

Root mean square error (RMSE) for goodness of fit of the top model (**Red.3**) selected.

```
mse <- mean(residuals(Red.3)^2/df.residual(Red.3))
rmse <- sqrt(mse)
rmse
```

```
## [1] 2.205651
```

Obtaining the Confidence Internals for the Top model (Red.3)

It is needed the package *nlstools* and the command *confint2* to obtain the 95% confidence intervals for parameters *I* and *A* for the **Red.3**.

```
#install.packages("nlstools")
library(nlstools)
```

```
##
## 'nlstools' has been loaded.

## IMPORTANT NOTICE: Most nonlinear regression models and data set examples

## related to predictive microbiolgy have been moved to the package 'nlsMicrobio'
```

```
confint2(Red.3, level=0.95)
```

```
##       2.5 %    97.5 %
## I1 25.37778 431.33628
## I2 24.44635  49.55307
## A  85.26278 127.07685
```

Extra - Setting a limit to the rectangular hyperbola parameters

Here we demonstrate how to set an upper limit to parameter *A* of **Red.3** model. Notice that we have to add *alg="port"* and *upper* command to the function. The *upper* command has three numbers, the first two set a limit of 10000 to parameter *I* of *R. brasiliensis* and *C. benghalensis*. The last *upper* number set a limit *A*=100%, which will lock the upper limit to a biologically meaningful value.

```r
Red.3_lim = nls(yl ~ (I[weed]*densityweed)/(1+(I[weed]/A)*densityweed), data=DMT,
                start=list(I=c(30,30), A=70), alg="port",
                upper=c(10000, 10000, 100), trace=T)
```

```
##   0:     26882.386:  30.0000  30.0000  70.0000
##   1:     18178.029:  42.8369  32.3788  84.6203
##   2:     6263.6597:  102.843  37.2539  100.000
##   3:     4270.9222:  165.478  39.1160  100.000
##   4:     3745.8945:  228.173  39.3120  100.000
##   5:     3641.7888:  285.567  39.3210  100.000
##   6:     3640.0157:  296.138  39.3212  100.000
##   7:     3640.0156:  296.037  39.3212  100.000
##   8:     3640.0156:  296.041  39.3212  100.000
```

```r
summary(Red.3_lim)
```

```
##
## Formula: yl ~ (I[weed] * densityweed)/(1 + (I[weed]/A) * densityweed)
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## I1  296.041    163.899   1.806    0.079 .
## I2   39.321      6.983   5.631 1.98e-06 ***
## A   100.000      9.337  10.710 6.85e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.03 on 37 degrees of freedom
##
## Algorithm "port", convergence message: relative convergence (4)
```

Acknowledgements

The statistical procedures are presented here with **Rmarkdown** and **RStudio**.