Lab Exercise 4

NRES 746

Fall 2018

As with all lab reports, your answers will either take the form of R functions or short written responses (submitted together in a Word document). The R functions should be stored in an R script file ('.R' extension). To allow me to evaluate your work more efficiently, please name your R script using the following convention: "[your first name]_[your last name]_lab4.R". So my submission would be"kevin_shoemaker_lab4.R". The name of your Word docuement doesn't matter, as long as you submit via WebCampus.

Please submit the R script and the Word document via WebCampus by midnight on the due date (one week after the final lab session allocated for this topic – here, *Nov. 21, 2018*). You can work in groups but please submit the materials individually.

First, take a little time to review the Bayesian lecture and MCMC lecture!

Lab 4 is due before midnight on Tuesday November 21

Bayesian inference (with BUGS!)

NOTE: thanks to Dr. Elizabeth Hunter (formerly a postdoc in the Applied Population Ecology lab) for putting together much of this lab!

If you haven't already installed JAGS on your computer, use this link

In this lab we will be applying a Bayesian approach to model fitting using the same myxomatosis dataset and model that we used in the previous likelihood lab. To do this, weâ€™ll be using the program JAGS (**J**ust **A**nother **G**ibbs **S**ampler).

JAGS runs seamlessly with R, if you download the "R2jags" package or the "jagsUI" package! JAGS is not the only way to do Bayesian analyses, nor is it the only implementation of the BUGS (Bayesian inference Using Gibbs Sampler) language. Others that you may have heard of include WinBUGS and OpenBUGS. STAN is another widely used software for Bayesian inference, that doesn't use the BUGS language...

To familiarize ourselves with the BUGS language, let's look at some BUGS code (for our favorite myxomatosis dataset).

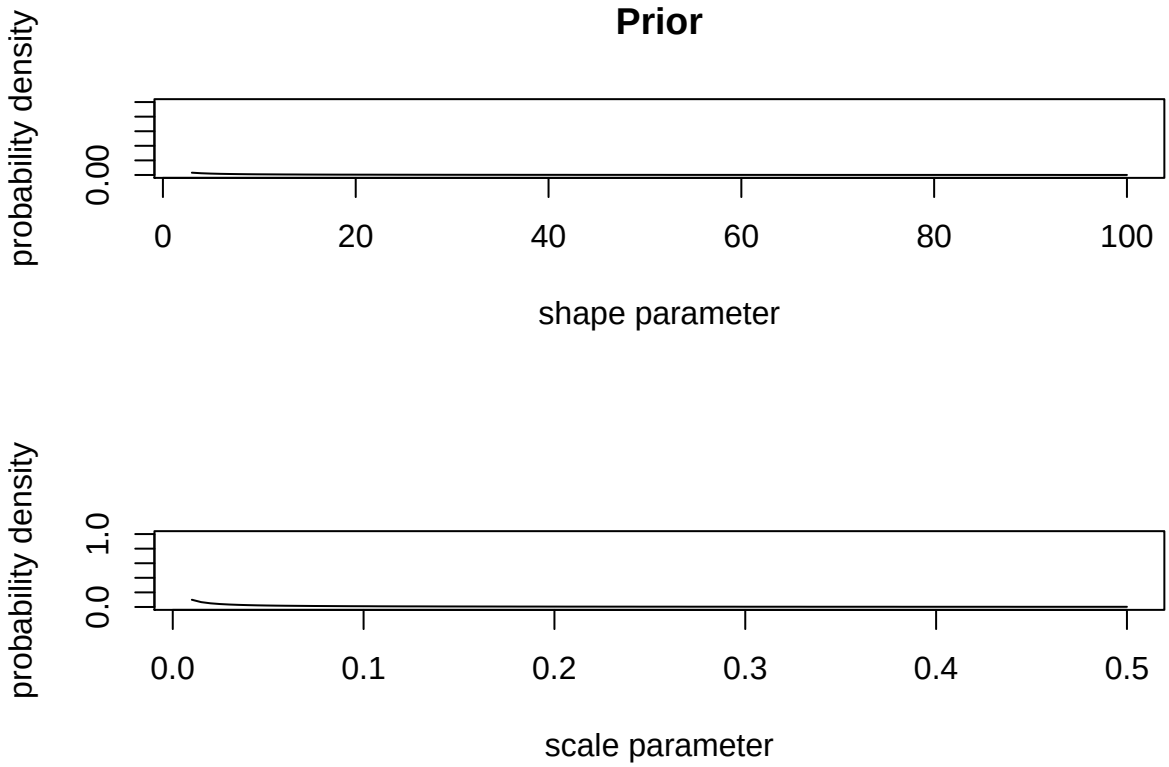
```
#############
# sample BUGS code

model {

      #############
      # LIKELIHOOD
      #############
      for(obs in 1:n.observations){
        titer[obs] ~ dgamma(shape,rate)
      }

      #############
      # PRIORS
      #############
```

```
  shape ~ dgamma(0.01,0.01)
  scale ~ dgamma(0.001,0.001)
  rate <- 1/scale   # in BUGS, gamma distribution needs a "rate" parameter rather than a scale
}
```

The syntax in BUGS/JAGS is very similar to R, but there are a few key differences. Most important, *assignment must always be done by the arrow (<-) instead of an equals sign* for deterministic functions and the *tilde (~) is used to define stochastic processes.*

In the model code, you must also specify your priors. Here we are using relatively vague priors, meaning that the probability is spread out fairly evenly over parameter space. The Gamma distribution is always parameterized as shape and rate in BUGS (rate = 1/scale). If you wanted to see what this distribution looked like, you could plot in R ('dgamma()' allows you to specify a rate or a scale parameter):

```
par(mfrow=c(2,1))
curve(dgamma(x, shape=0.01, rate=0.01),3,100,ylim=c(0,0.1),xlab="shape parameter",ylab="probability dens
curve(dgamma(x, shape=0.001, rate=0.001),0.01,0.5,ylim=c(0,1),xlab="scale parameter",ylab="probability
```



In this case we want a prior distribution that has little to no information about which values of the parameter are most probable. These distributions have no peaks within the range of parameter space of interest, and the probability for any particular x-value is very low, so it does a fairly good job of representing a vague prior for the shape and rate parameters.

Running JAGS through R

Setting up the model!

We can use the R2JAGS package in R to run JAGS through R and have the output returned to R so that we can analyze and manipulate it further.

First, install and load the R2jags package. While you're at it, install and load the coda package, which has some great utilities for visualizing and working with MCMC data.

```
### load key packages for running Bayesian analysis in JAGS

library(R2jags)
library(coda)
```

Writing out the model

We can use the 'cat' function in R to write out a text file- it can be useful to use this functionality to embed the JAGS code within our R script!

```
filename <- "BUGSmodel.txt"
cat("
    model {

      #############
      # LIKELIHOOD
      ############
      for(obs in 1:n.observations){
        titer[obs] ~ dgamma(shape,rate)
      }

      #############
      # PRIORS
      ############
      shape ~ dgamma(0.01,0.01)
      scale ~ dgamma(0.001,0.001)
      rate <- 1/scale
    }
  ",file=filename
)
```

Packaging the data

We will need to tell BUGS/JAGS what the data are. To be read into JAGS (via the R2JAGS package), the data need to be bundled together in a list. The data need to have the same names as specified in the model file:

First we get the data we want into R (we know the drill!):

```
library(emdbook)

MyxDat <- MyxoTiter_sum
Myx <- subset(MyxDat,grade==1)  #Data set from grade 1 of myxo data
head(Myx)
```

Then we package the data as a list, which is what JAGS wants!

```
myx.data.for.bugs <- list(
  titer = Myx$titer,
  n.observations = length(Myx$titer)
)

myx.data.for.bugs
```

```
## $titer
##  [1] 5.207 5.734 6.613 5.997 6.612 6.810 5.930 6.501 7.182 7.292 7.819
## [12] 7.489 6.918 6.808 6.235 6.916 4.196 7.682 8.189 7.707 7.597 7.112
## [23] 7.354 7.158 7.466 7.927 8.499
##
## $n.observations
## [1] 27
```

Setting the initial values for our Markov chain(s)

```
init.vals.for.bugs <- function(){
  list(
    shape=runif(1,20,100),
    scale=runif(1,0.05,0.3)
  )
}
init.vals.for.bugs()
```

```
## $shape
## [1] 22.05974
##
## $scale
## [1] 0.1617185
```

```
init.vals.for.bugs()
```

```
## $shape
## [1] 28.44063
##
## $scale
## [1] 0.1699203
```

Alternatively, we can specify exact initial values for our chains (in this case, we initialize three Markov chains)

```
inits = list(list(shape=90,scale=0.1), list(shape=50,scale=0.2), list(shape=150,scale=0.04))  # a list
```

Note that we need three different sets of starting values here because we are running three different chains. Recall that initial values are required, and you may want to use the same tricks as we have used before (e.g., method of moments) to identify reasonable starting values. JAGS will not work if you specify unreasonable initial parameters (and it wonâ€™t necessarily tell you that is the problem).

Now weâ€™ll run this model through JAGS:

```
params.to.store <- c("shape","scale")    # specify the parameters we want to get the posteriors for

jags.fit <- jags(data=myx.data.for.bugs,inits=init.vals.for.bugs,parameters.to.save=params.to.store,n.i
```

```
## Compiling model graph
```

```
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 2
##    Total graph size: 37
##
## Initializing model
##
##
  |
  |                                                       |   0%
  |
  |++++++++++                                             |  20%
  |
  |++++++++++++++++++++                                   |  40%
  |
  |++++++++++++++++++++++++++++++                         |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                 |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                       |   0%
  |
  |*                                                      |   2%
  |
  |**                                                     |   4%
  |
  |***                                                    |   7%
  |
  |****                                                   |   9%
  |
  |******                                                 |  11%
  |
  |*******                                                |  13%
  |
  |********                                               |  16%
  |
  |*********                                              |  18%
  |
  |**********                                             |  20%
  |
  |***********                                            |  22%
  |
  |************                                           |  24%
  |
  |*************                                          |  27%
  |
  |***************                                        |  29%
  |
  |****************                                       |  31%
  |
```

```
|*****************                                  |  33%
|
|******************                                 |  36%
|
|*******************                                |  38%
|
|*********************                              |  40%
|
|**********************                             |  42%
|
|************************                           |  44%
|
|*************************                          |  47%
|
|**************************                         |  49%
|
|****************************                       |  51%
|
|*****************************                      |  53%
|
|*******************************                    |  56%
|
|********************************                   |  58%
|
|**********************************                 |  60%
|
|***********************************                |  62%
|
|************************************               |  64%
|
|*************************************              |  67%
|
|***************************************            |  69%
|
|****************************************           |  71%
|
|*****************************************          |  73%
|
|*******************************************        |  76%
|
|********************************************       |  78%
|
|*********************************************      |  80%
|
|**********************************************     |  82%
|
|************************************************   |  84%
|
|*************************************************  |  87%
|
|************************************************** |  89%
|
|**************************************************|  91%
|
```

```
  |*********************************************   |  93%
  |
  |**********************************************  |  96%
  |
  |********************************************** |  98%
  |
  |**********************************************| 100%
```

```
jags.fit
```

```
## Inference for Bugs model at "BUGSmodel.txt", fit using jags,
##  3 chains, each with 50000 iterations (first 5000 discarded), n.thin = 20
##  n.sims = 6750 iterations saved
##          mu.vect sd.vect   2.5%    25%    50%    75%  97.5%  Rhat n.eff
## scale      0.161   0.046  0.093  0.128  0.153  0.186  0.269 1.003  1000
## shape     46.482  12.483 25.845 37.334 45.241 54.171 73.946 1.003  1000
## deviance  77.401   2.022 75.385 75.941 76.803 78.226 82.704 1.001  3000
##
## For each parameter, n.eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor (at convergence, Rhat=1).
##
## DIC info (using the rule, pD = var(deviance)/2)
## pD = 2.0 and DIC = 79.4
## DIC is an estimate of expected predictive error (lower deviance is better).
```

For each parameter, n.eff is a crude measure of effective sample size, and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = Dbar-Dhat) pD = 3.0 and DIC = 65.3 DIC is an estimate of expected predictive error (lower deviance is better).

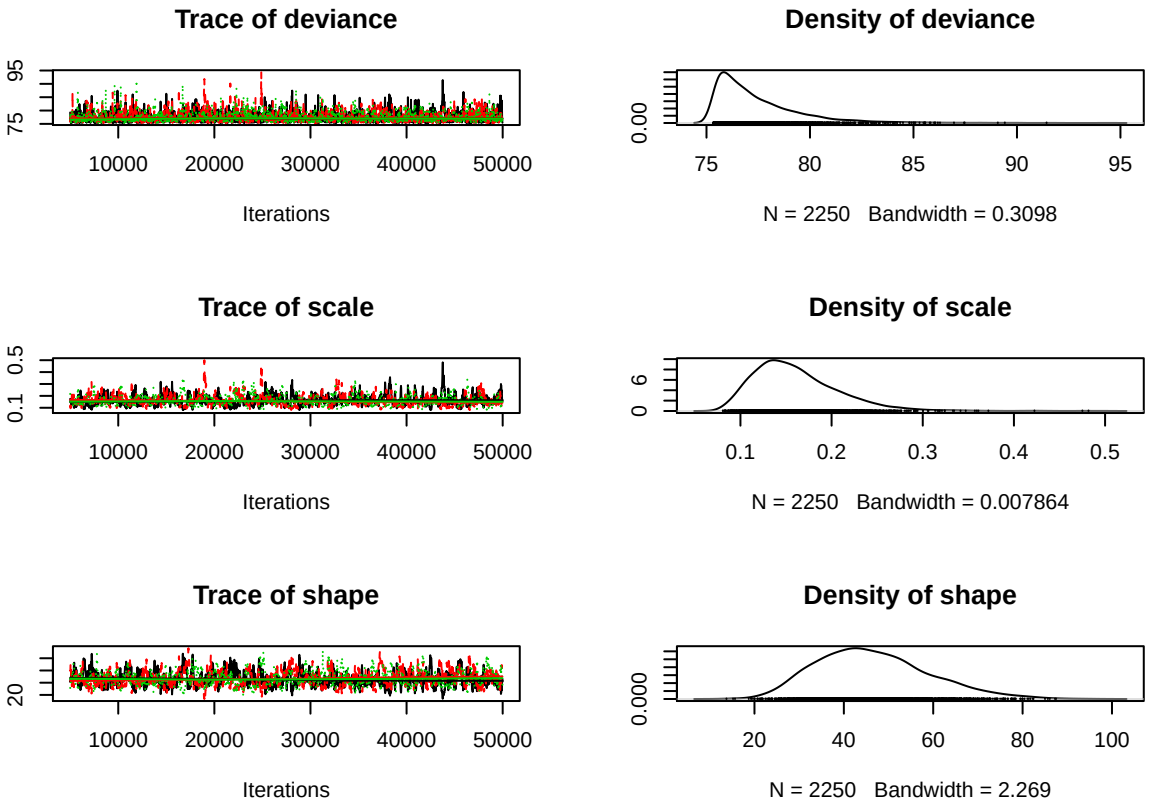
You can see the means and variance for the parameters as well as the DIC for the model. We can summarize these in plots of the posterior distributions. First, we define the MCMC output as a 'coda' object (for storing, visualizing and analyzing MCMC results) that R knows how to work with.

```
jagsfit.mcmc <- as.mcmc(jags.fit)   # convert to "MCMC" object (coda package)

summary(jagsfit.mcmc)
```

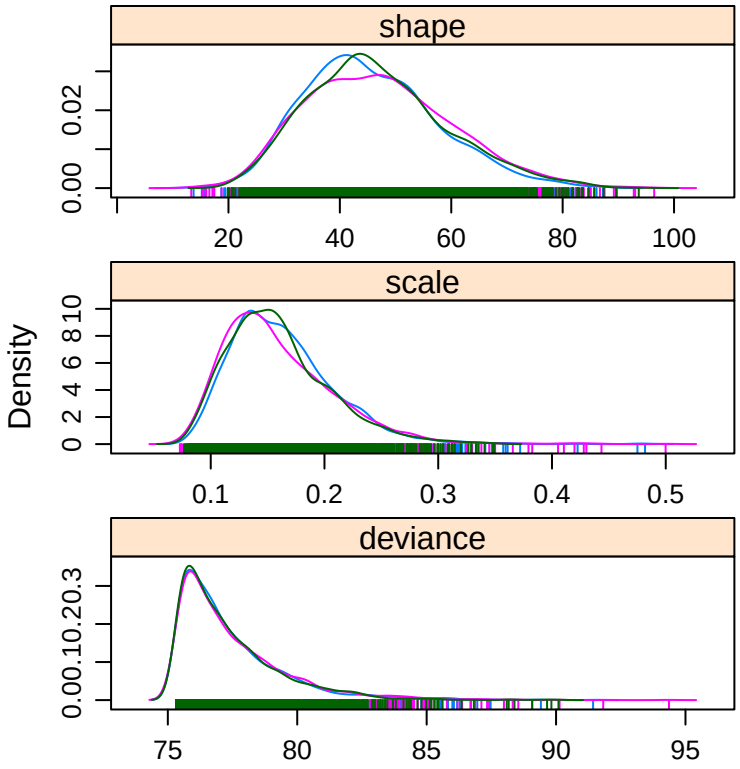
```
##
## Iterations = 5001:49981
## Thinning interval = 20
## Number of chains = 3
## Sample size per chain = 2250
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##             Mean       SD  Naive SE Time-series SE
## deviance 77.4014  2.02150 0.0246049       0.052647
## scale     0.1607  0.04646 0.0005655       0.002149
## shape    46.4821 12.48330 0.1519419       0.579288
##
## 2. Quantiles for each variable:
##
##             2.5%     25%     50%     75%   97.5%
## deviance 75.3850 75.9413 76.8028 78.2256 82.7040
```

```
## scale     0.0932  0.1277  0.1528  0.1857  0.2691
## shape    25.8452 37.3337 45.2415 54.1715 73.9458
```

```
plot(jagsfit.mcmc)
```




```
library(lattice)
densityplot(jagsfit.mcmc)
```



You can visually check for convergence here, using the trace plots â€“ a converged run will look like white noise and the samples will not be hitting any ceilings or floors.

Parameter uncertainty: credible intervals

We can estimate the 95% credible interval by calling directly from the bugs output. The 'sims.list' part of the output has all of the MCMC samples that were created during the run.

```
shape95 = quantile(jags.fit$BUGSoutput$sims.list$shape,c(0.025,0.975))
scale95 = quantile(jags.fit$BUGSoutput$sims.list$scale,c(0.025,0.975))
shape95
```

```
##     2.5%    97.5%
## 25.84524 73.94577
```

```
scale95
```

```
##       2.5%      97.5%
## 0.09319586 0.26911024
```

The probability that the a parameter is between these 2 numbers is actually 95%!

So, whatâ€™s so great about this? Using a vague prior, we get almost the exact same parameter estimates that we did when we just did a straight up likelihood model. This is usually the case with simple models like the run we ran here. However, answers can be quite different with more complicated models, and in fact the Bayesian approach allows us to simultaneously fit complex models that we could not do effectively using likelihood-based approaches.

That said, there are 2 main advantages to the Bayesian approach, both having to do with the fact that the answer comes in the form of a probability distribution instead of a point estimate.

First, the credible interval is much nicer to interpret than a confidence interval. Thereâ€™s none of this "given that the null hypothesis is true, and we were to resample the data 100 timesâ€¦". You can state simply that thereâ€™s a 95% probability that the value of the parameter lies within the credible interval. Period.

Second, and more importantly from a pragmatic standpoint (i.e., for running simulations), you can draw from these parameter probability distributions to create simulations that include the full variability of outcomes instead of just point estimates, which will demonstrate the implications of your parameter estimates (e.g. under different management scenarios, climate change, etc.). Weâ€™ll explore how to do this a little later.

Challenge problem 1: Myxomatosis analysis in JAGS #1: Ricker deterministic function

- **Exercise 1a**. Write a function called "Myx_Ricker_JAGS()" that runs the Myxomatosis/Ricker example from the previous (likelihood) lab in JAGS. Recall that we are modeling titer levels (grade 1 only) as a function of the days since infection. Submit this function as part of your R script.
  - **input**:
    - data = a matrix of 2 columns and one row per observation. The first column should represent the days since infection, and the second column should represent the virus titer.
    - JAGScode = a text string, representing the file name for your JAGS code
    - CImethod = a text string, representing whether the CI should be computed using the quantile method ("quantile") or the HPD method ("HPD"). See the Bayesian lecture for more information.
    - CIlevel = a number from 0 to 1 indicating what confidence level you want for the CI: set the default value to 0.95.
  - **suggested algorithm**:
    - Package the data for JAGS (store as a list object)
    - Write a function that returns a list for initializing the MCMC chains
    - Run the 'jags' function with 3 chains, specifying the parameters for which you want the posterior distribution, and store the results as an object.
    - use the 'as.mcmc()' function to convert the results to a 'coda' object, and store the results as a separate object.
    - Display the trace plots and density plots for the three parameters.
    - Plot the data (days on x axis and titer on y axis), and overlay the Ricker curve, with Bayesian point estimates, on the data to evaluate fit.
    - compute the posterior mean for the a, b and shape params and store as a vector.
    - compute the credible interval (CI) for the three parameters using either the quantile method or the HPD method (as requested by the user), and store as a 2x3 matrix (see below). HINT: use the 'HDIntervalâ€™ package to help compute the HPD.
    - compute the Gelman-Rubin diagnostic ('gelman.diag' function) using the 'coda' version of the JAGS results.
  - **return**:
    - a list of length 3 containing (1) a vector of point estimates (posterior mean) for parameters a, b, and shape, (2) a matrix with 2 rows and 3 columns, containing the 95% credible interval (row 1 is the lower bound, row 2 is the upper bound, columns represent parameters a, b, and shape), and (3) a scalar (floating-point number) representing the Gelman-Rubin diagnostic- a

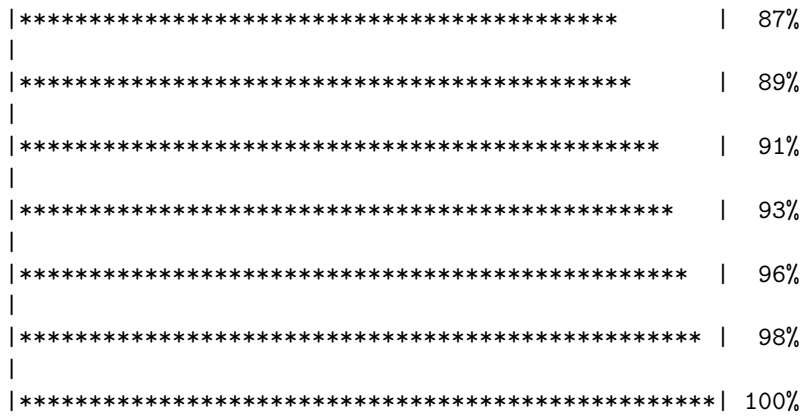
tool for assessing MCMC convergence.

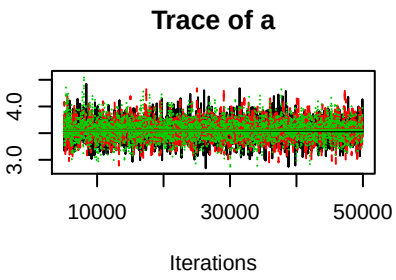
Here is an example of the results you should expect to see when running your new function:

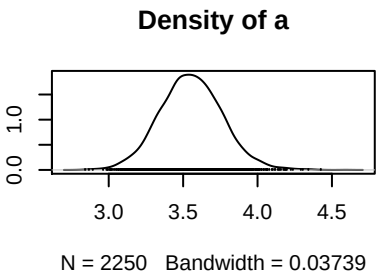
```
Ricker_results <- Myx_Ricker_JAGS(data=Myx[,-1],JAGScode="BUGSmodel_ricker.txt",CImethod="HPD",CIlevel=
```

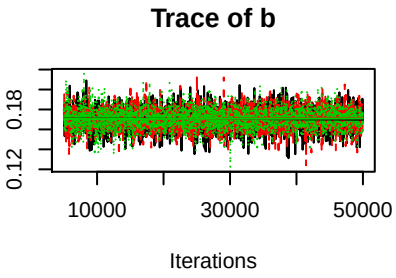
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 124
##
## Initializing model
##
##
  |
  |                                                      |   0%
  |
  |++++++++++                                            |  20%
  |
  |++++++++++++++++++++                                  |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                      |   0%
  |
  |*                                                     |   2%
  |
  |**                                                    |   4%
  |
  |***                                                   |   7%
  |
  |****                                                  |   9%
  |
  |******                                                |  11%
  |
  |*******                                               |  13%
  |
  |********                                              |  16%
  |
  |*********                                             |  18%
  |
  |**********                                            |  20%
  |
  |***********                                           |  22%
  |
  |************                                          |  24%
  |
```

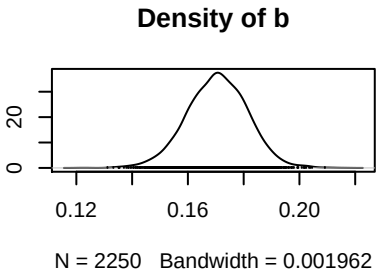
```
|*************                                         |  27%
|
|**************                                        |  29%
|
|****************                                      |  31%
|
|******************                                    |  33%
|
|*******************                                   |  36%
|
|********************                                  |  38%
|
|**********************                                |  40%
|
|***********************                               |  42%
|
|************************                              |  44%
|
|*************************                             |  47%
|
|**************************                            |  49%
|
|****************************                          |  51%
|
|*****************************                         |  53%
|
|******************************                        |  56%
|
|********************************                      |  58%
|
|*********************************                     |  60%
|
|***********************************                   |  62%
|
|************************************                  |  64%
|
|*************************************                 |  67%
|
|***************************************               |  69%
|
|*****************************************             |  71%
|
|******************************************            |  73%
|
|*******************************************           |  76%
|
|*********************************************         |  78%
|
|**********************************************        |  80%
|
|************************************************      |  82%
|
|*************************************************     |  84%
|
```

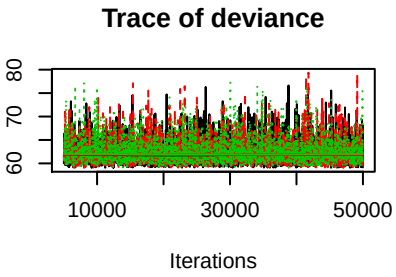

|** | 87%
|
|*** | 89%
|
|** | 91%
|
|*** | 93%
|
|** | 96%
|
|*** | 98%
|
|**| 100%

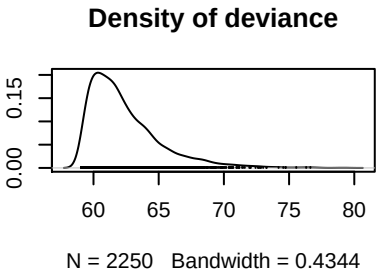

Trace of a
3.0
4.0
10000
30000
50000
Iterations

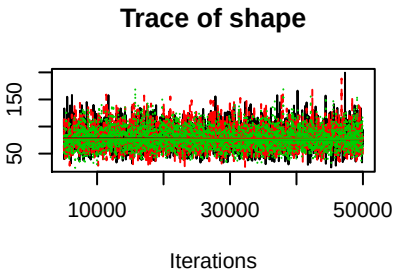

Density of a
0.0
1.0
3.0
3.5
4.0
4.5
N = 2250 Bandwidth = 0.03739

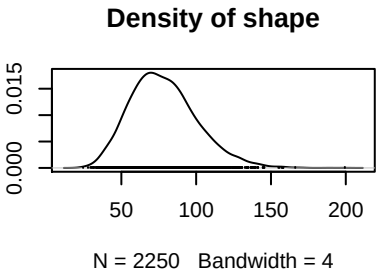

Trace of b
0.12
0.18
10000
30000
50000
Iterations

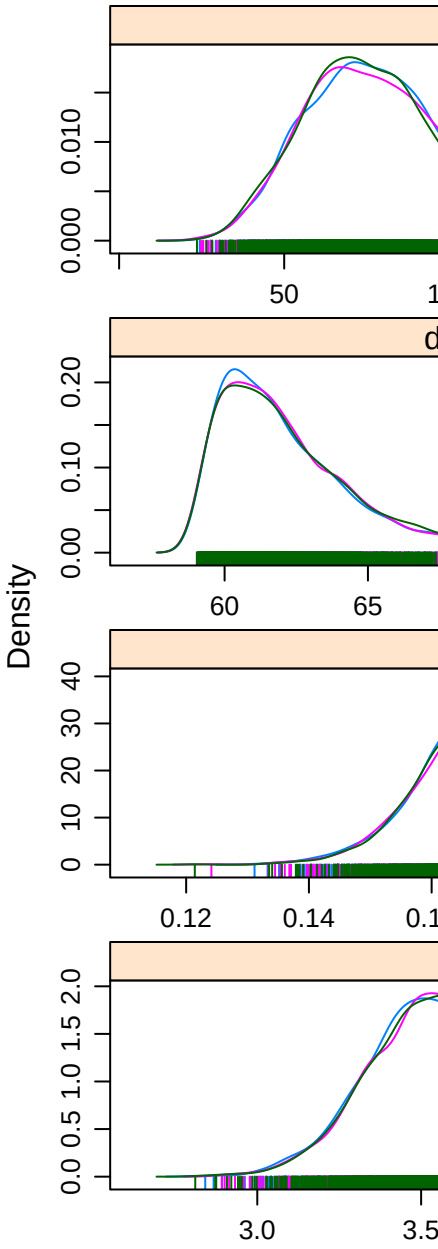

Density of b
0
20
0.12
0.16
0.20
N = 2250 Bandwidth = 0.001962

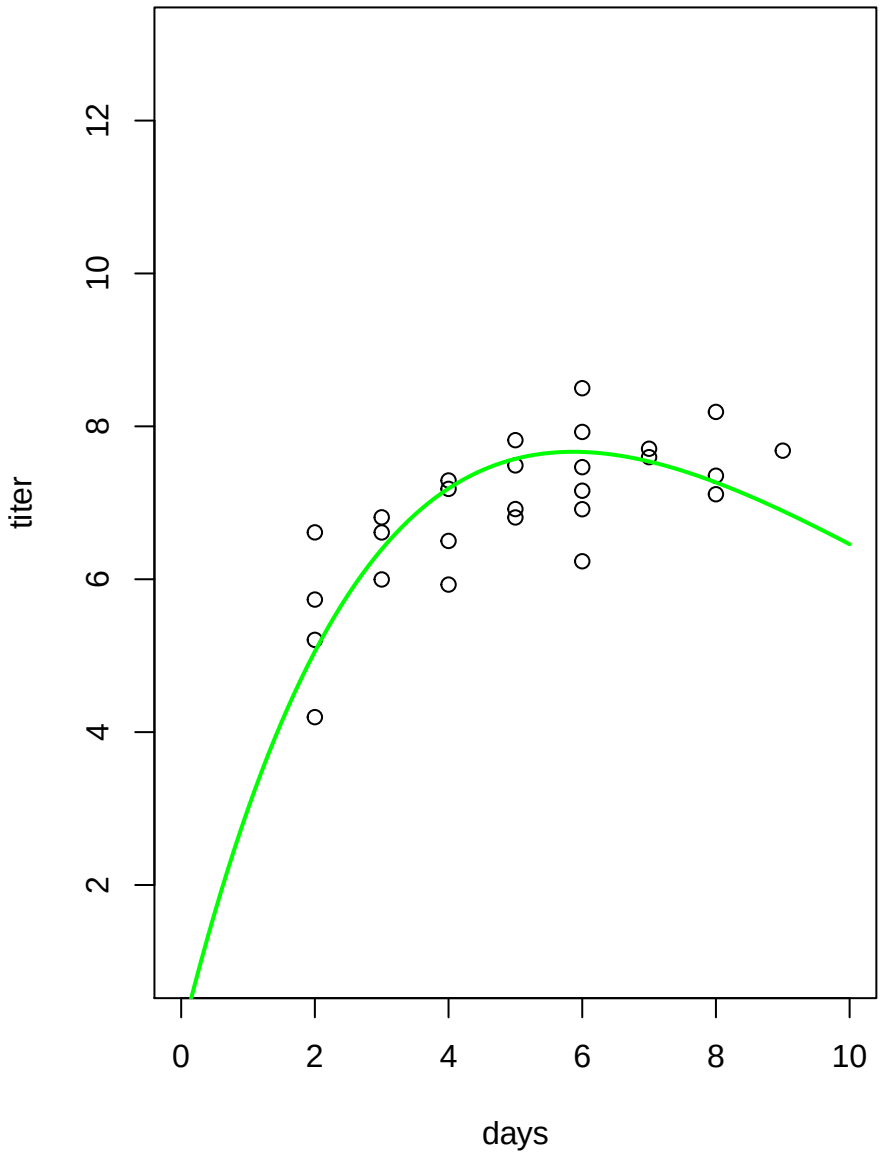

Trace of deviance
60
70
80
10000
30000
50000
Iterations


Density of deviance
0.00
0.15
60
65
70
75
80
N = 2250 Bandwidth = 0.4344


Trace of shape
50
150
10000
30000
50000
Iterations


Density of shape
0.000
0.015
50
100
150
200
N = 2250 Bandwidth = 4


Density
0.000
0.010
50
0.00
0.10
0.20
60
65
0
10
20
30
40
0.12
0.14
0.0
0.5
1.0
1.5
2.0
3.0
3.5




```
Ricker_results
```

```
## [[1]]
## [1]  3.5542497  0.1705374 78.3815059
##
## [[2]]
##             a         b    shape
## [1,] 3.221802 0.1536216  43.6727
## [2,] 3.859886 0.1867468 111.0131
##
## [[3]]
```

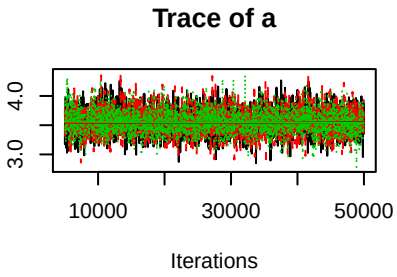
```
## [1] 1.00166
```

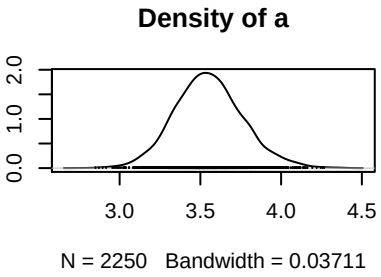
```
Ricker_results <- Myx_Ricker_JAGS(data=Myx[,-1],JAGScode="BUGSmodel_ricker.txt",CImethod="quantile",CIl
```

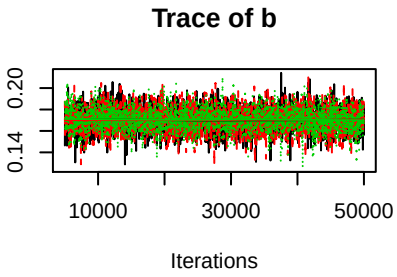
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 124
##
## Initializing model
##
##
  |
  |                                                      |   0%
  |
  |++++++++++                                            |  20%
  |
  |++++++++++++++++++++                                  |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                      |   0%
  |
  |*                                                     |   2%
  |
  |**                                                    |   4%
  |
  |***                                                   |   7%
  |
  |****                                                  |   9%
  |
  |******                                                |  11%
  |
  |*******                                               |  13%
  |
  |********                                              |  16%
  |
  |*********                                             |  18%
  |
  |**********                                            |  20%
  |
  |***********                                           |  22%
  |
  |************                                          |  24%
  |
  |*************                                         |  27%
  |
```

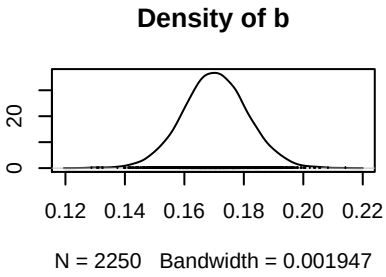
```
|**************                                    |  29%
|
|****************                                  |  31%
|
|******************                                |  33%
|
|********************                              |  36%
|
|**********************                            |  38%
|
|************************                          |  40%
|
|**************************                        |  42%
|
|****************************                      |  44%
|
|******************************                    |  47%
|
|********************************                  |  49%
|
|**********************************                |  51%
|
|************************************              |  53%
|
|**************************************            |  56%
|
|****************************************          |  58%
|
|******************************************        |  60%
|
|********************************************      |  62%
|
|**********************************************    |  64%
|
|************************************************  |  67%
|
|**************************************************|  69%
|
|**************************************************|  71%
|
|**************************************************|  73%
|
|**************************************************|  76%
|
|**************************************************|  78%
|
|**************************************************|  80%
|
|**************************************************|  82%
|
|**************************************************|  84%
|
|**************************************************|  87%
|
```

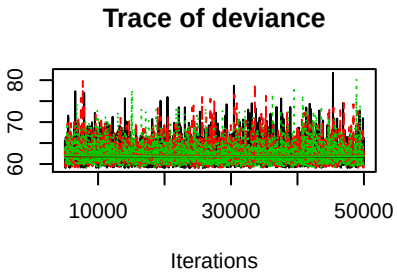
```
|*****************************************       |  89%
|
|******************************************      |  91%
|
|*******************************************     |  93%
|
|********************************************    |  96%
|
|*********************************************** |  98%
|
|************************************************|  100%
```

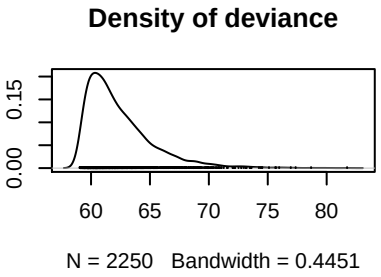


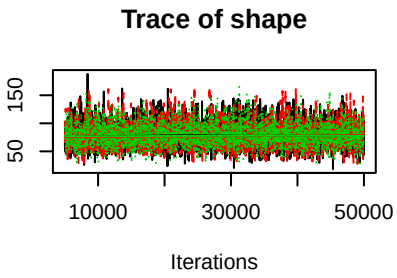


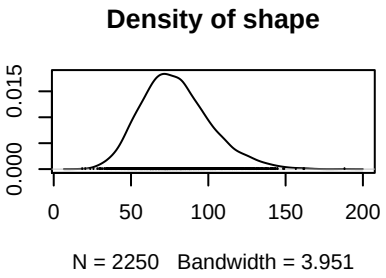


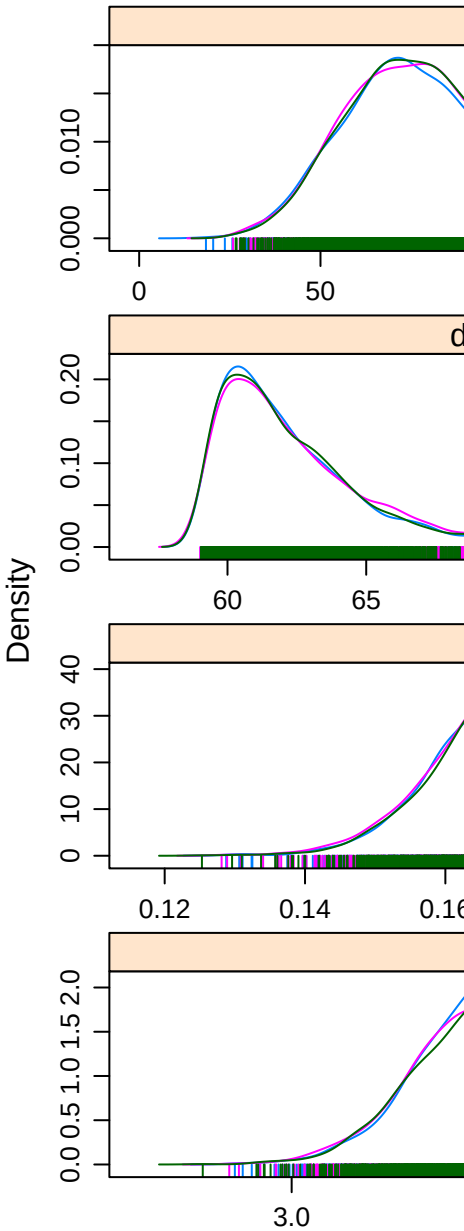


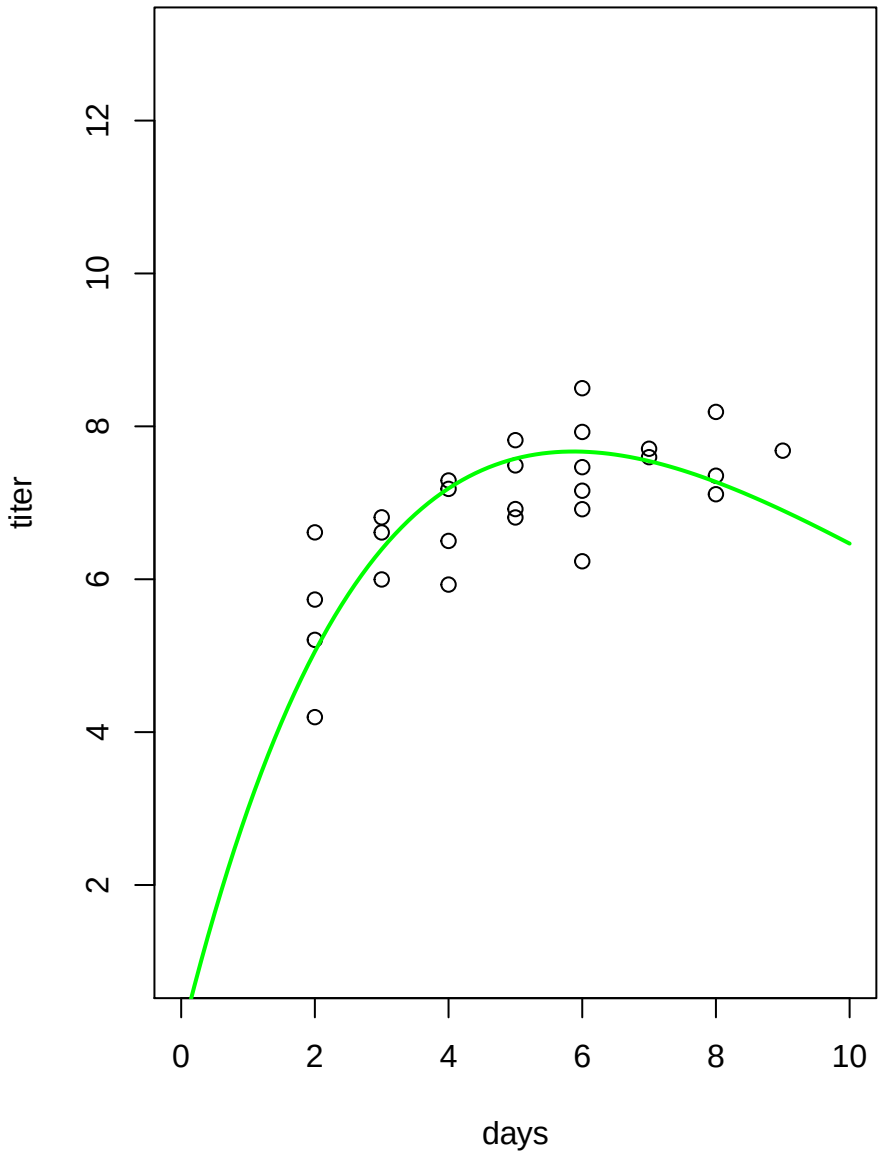













```
Ricker_results
```

```
## [[1]]
## [1]  3.5521705  0.1703568 78.9988302
##
## [[2]]
##             a         b     shape
## [1,] 3.238454 0.1535503  47.77953
## [2,] 3.890410 0.1872480 116.96952
##
## [[3]]
```

```
## [1] 1.000562
```

- **Exercise 1b**. Can you identify any differences between the parameter estimates from the analysis in a likelihood vs Bayesian perspective? Compare your results from Lab 3, and briefly describe your findings in your Word document.
- **Exercise 1c**. Did your MCMC chains converge on the joint posterior distribution? Briefly explain your reasoning in your Word document! Use several lines of reasoning!

Challenge problem 2: Myxomatosis in JAGS with Michaelis-Menten function

Recalling our plot (from the last lab) of the Ricker function with the maximum-likelihood parameter estimates drawn over the scatterplot of titers by day, there really isnâ€™t much evidence in the data for that downward turn in the function after day 6. We chose a model that indicated decline in titer levels following a peak based on the behavior of other myxomytosis grades, but given the virulence of this particular grade most animals die once the titer levels reach their maximum. Might it be more appropriate to fit a model that levels off at some asymptote instead of declining following the peak? Let's find out, using a little Bayesian model selection!

- **Exercise 2a**. Repeat the myxomatosis example in BUGS/JAGS, but this time use the **Michaelis-Menten function**, which has the same number of parameters as the Ricker function, but increases to an asymptote (see below). Continue to use a Gamma distribution to describe the error! Please name your new function "Myx_MM_JAGS()". The inputs and outputs should be the same as for "Myx_Ricker_JAGS()". Please save this new function in your R script.

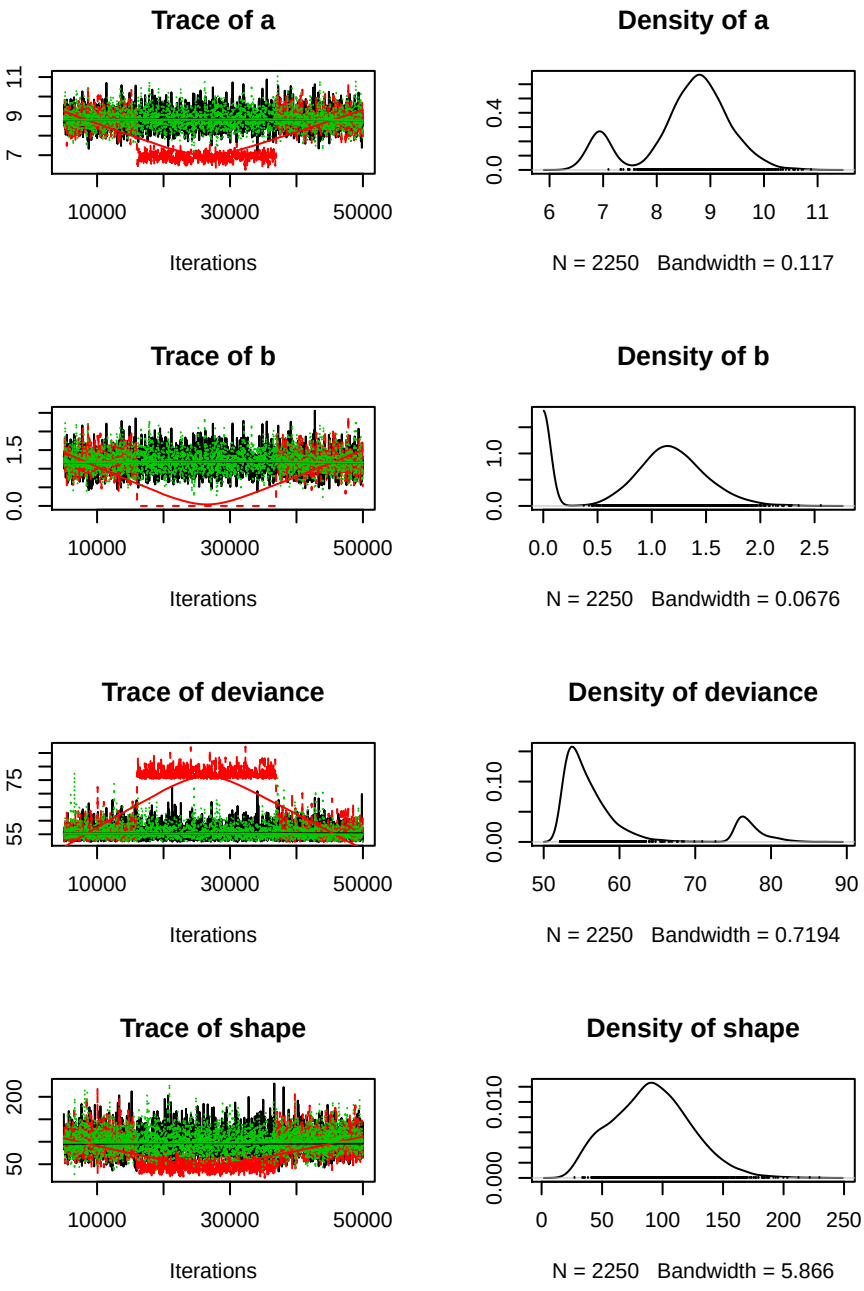
And here is some test commands with output:

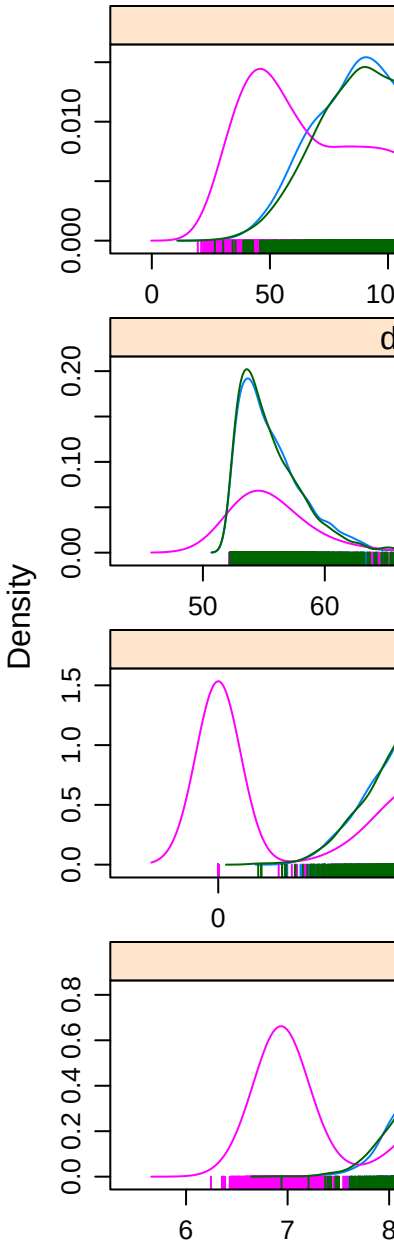
```
MM_results <- Myx_MM_JAGS(data=Myx[,-1],JAGScode="BUGSmodel_mm.txt",CImethod="HPD",CIlevel=0.88)
```

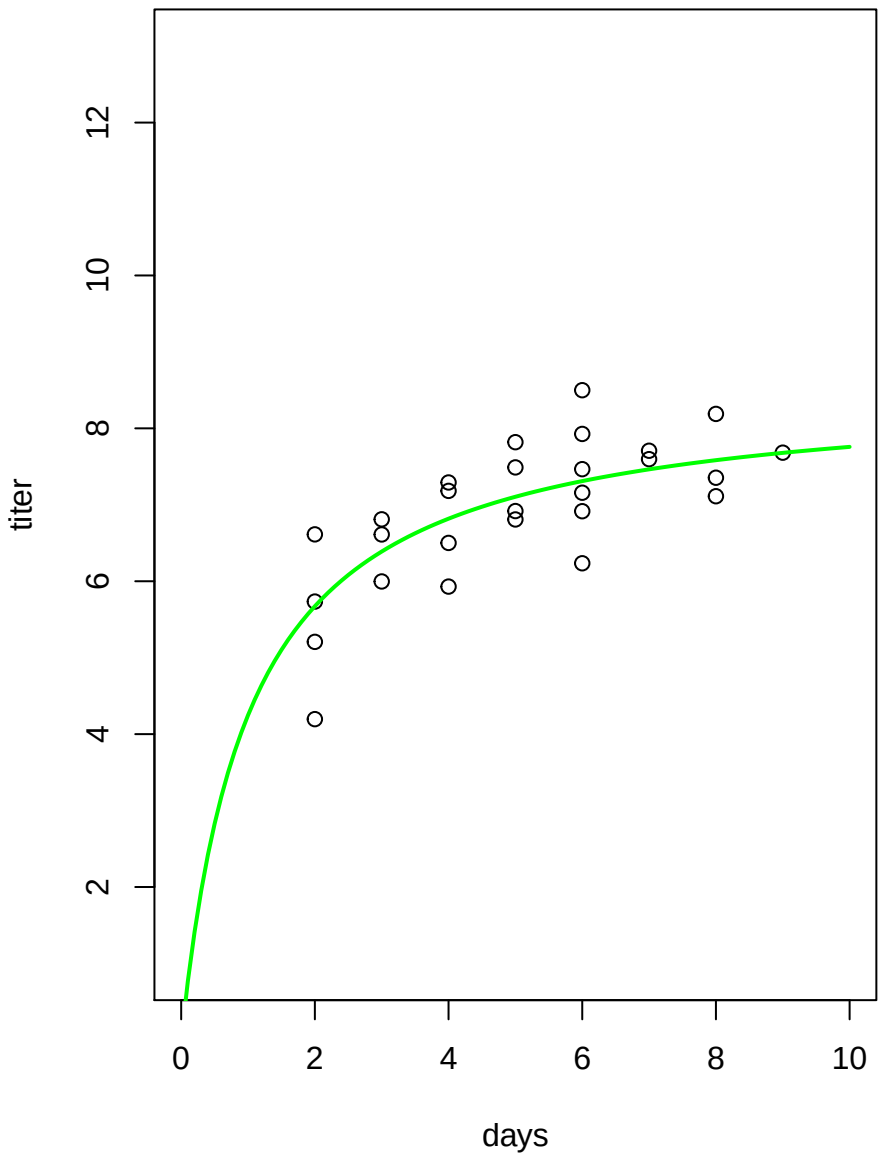
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 96
##
## Initializing model
##
##
  |
  |                                                      |   0%
  |
  |++++++++++                                            |  20%
  |
  |++++++++++++++++++++                                  |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                      |   0%
  |
```

```
|*                                                    |   2%
|
|**                                                   |   4%
|
|***                                                  |   7%
|
|****                                                 |   9%
|
|******                                               |  11%
|
|*******                                              |  13%
|
|********                                             |  16%
|
|*********                                            |  18%
|
|**********                                           |  20%
|
|***********                                          |  22%
|
|************                                         |  24%
|
|*************                                        |  27%
|
|**************                                       |  29%
|
|****************                                     |  31%
|
|*****************                                    |  33%
|
|******************                                   |  36%
|
|*******************                                  |  38%
|
|********************                                 |  40%
|
|*********************                                |  42%
|
|**********************                               |  44%
|
|************************                             |  47%
|
|*************************                            |  49%
|
|**************************                           |  51%
|
|***************************                          |  53%
|
|*****************************                        |  56%
|
|******************************                       |  58%
|
|*******************************                      |  60%
|
```

```
|*******************************                  |  62%
|
|********************************                 |  64%
|
|*********************************                |  67%
|
|**********************************               |  69%
|
|***********************************              |  71%
|
|************************************             |  73%
|
|*************************************            |  76%
|
|**************************************           |  78%
|
|***************************************          |  80%
|
|****************************************         |  82%
|
|*****************************************        |  84%
|
|******************************************       |  87%
|
|*******************************************      |  89%
|
|********************************************     |  91%
|
|*********************************************    |  93%
|
|**********************************************   |  96%
|
|*********************************************** |  98%
|
|************************************************|  100%
```


Trace of a
Density of a
Iterations
N = 2250 Bandwidth = 0.117
Trace of b
Density of b
Iterations
N = 2250 Bandwidth = 0.0676
Trace of deviance
Density of deviance
Iterations
N = 2250 Bandwidth = 0.7194
Trace of shape
Density of shape
Iterations
N = 2250 Bandwidth = 5.866


Density




```
MM_results
```

```
## [[1]]
## [1]  8.542342  1.011493 91.046732
##
## [[2]]
##             a             b    shape
## [1,] 6.817522 1.372590e-129  36.3580
## [2,] 9.548808  1.521837e+00 135.6405
##
## [[3]]
```

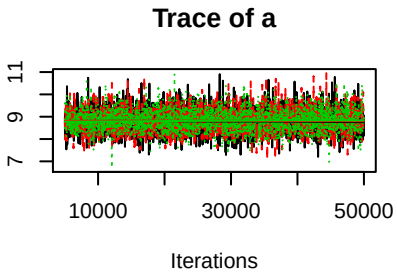
```
## [1] 1.415901
```

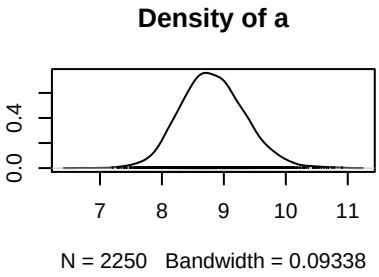
```
MM_results <- Myx_MM_JAGS(data=Myx[,-1],JAGScode="BUGSmodel_mm.txt",CImethod="quantile",CIlevel=0.88)
```

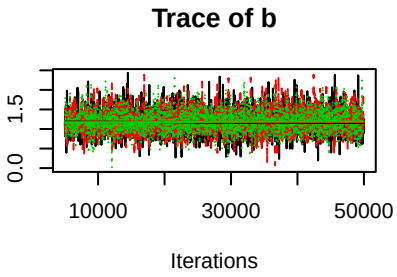
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 96
##
## Initializing model
##
##
  |
  |                                                      |   0%
  |
  |++++++++++                                            |  20%
  |
  |++++++++++++++++++++                                  |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                      |   0%
  |
  |*                                                     |   2%
  |
  |**                                                    |   4%
  |
  |***                                                   |   7%
  |
  |****                                                  |   9%
  |
  |******                                                |  11%
  |
  |*******                                               |  13%
  |
  |********                                              |  16%
  |
  |*********                                             |  18%
  |
  |**********                                            |  20%
  |
  |***********                                           |  22%
  |
  |************                                          |  24%
  |
  |*************                                         |  27%
  |
```

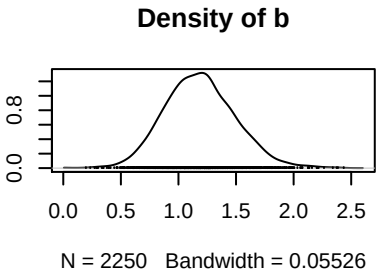
```
|**************                                    |  29%
|
|****************                                  |  31%
|
|******************                                |  33%
|
|********************                              |  36%
|
|**********************                            |  38%
|
|************************                          |  40%
|
|**************************                        |  42%
|
|****************************                      |  44%
|
|******************************                    |  47%
|
|********************************                  |  49%
|
|**********************************                |  51%
|
|************************************              |  53%
|
|**************************************            |  56%
|
|****************************************          |  58%
|
|******************************************        |  60%
|
|********************************************      |  62%
|
|**********************************************    |  64%
|
|************************************************  |  67%
|
|**************************************************|  69%
|
|**************************************************|  71%
|
|**************************************************|  73%
|
|**************************************************|  76%
|
|**************************************************|  78%
|
|**************************************************|  80%
|
|**************************************************|  82%
|
|**************************************************|  84%
|
|**************************************************|  87%
|
```

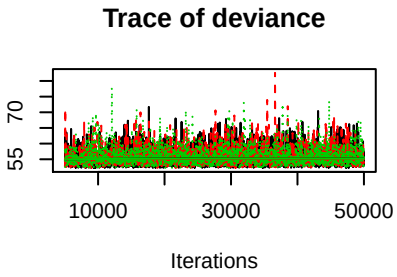
```
|*****************************************       |  89%
|
|******************************************      |  91%
|
|*******************************************     |  93%
|
|********************************************    |  96%
|
|*********************************************** |  98%
|
|************************************************|  100%
```

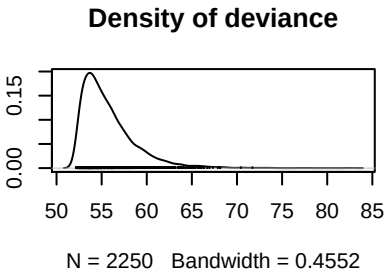


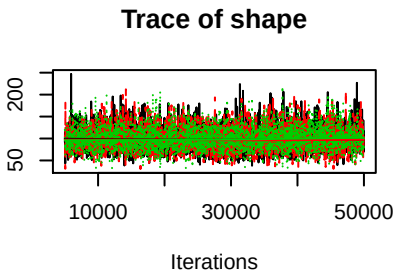


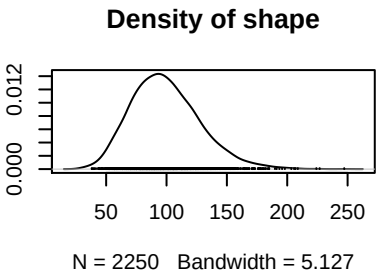


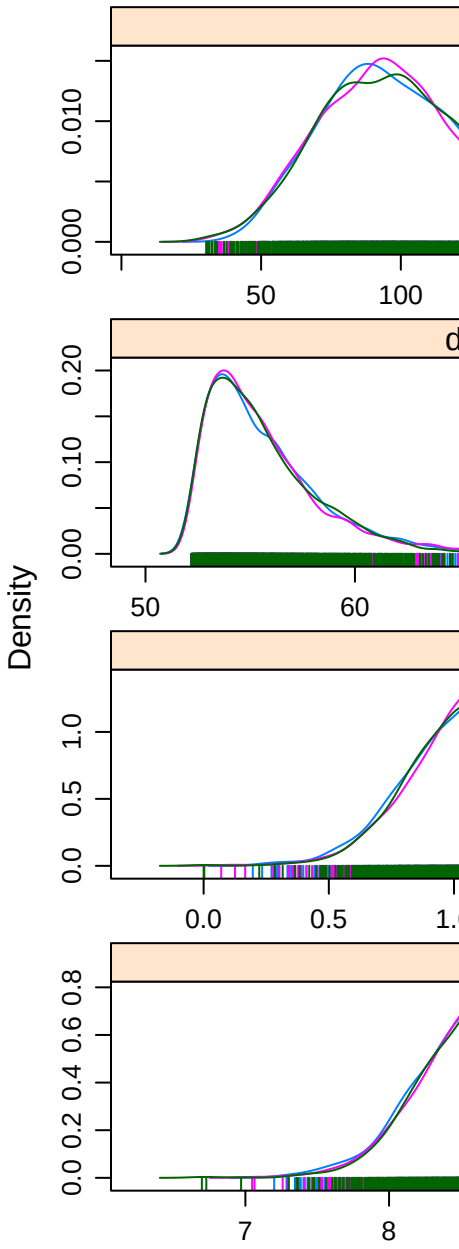


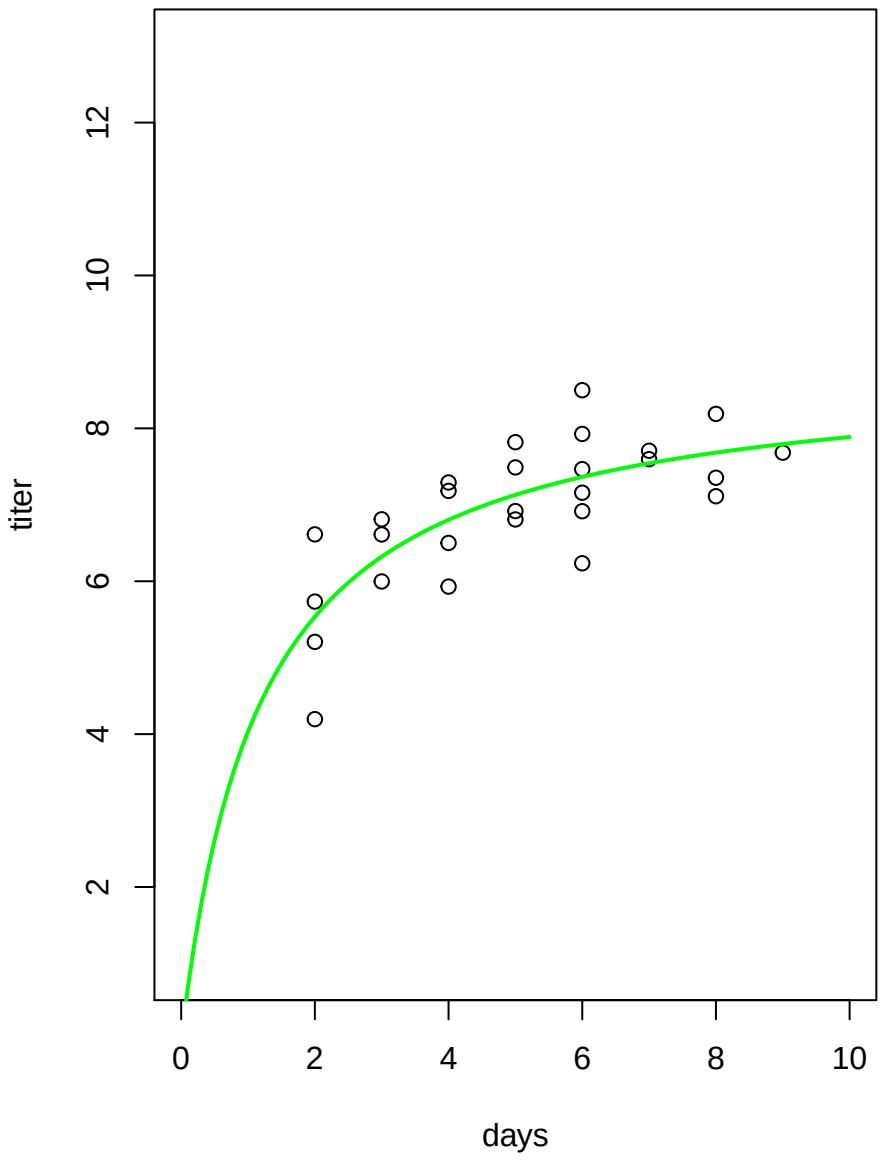













```
MM_results
```

```
## [[1]]
## [1]  8.822635  1.187170 99.400218
##
## [[2]]
##             a        b     shape
## [1,] 8.059169 0.731623  59.21099
## [2,] 9.658326 1.680144 146.83025
##
## [[3]]
```

```
## [1] 1.00188
```

- **Exercise 2b**. In your Word document, please answer the following question: overlay both the best-fit M-M and the best-fit Ricker curves on a plot of the data. On the basis of simple visual cues, does the M-M function seem to fit better than the Ricker function to describe this relationship? Explain your reasoning.

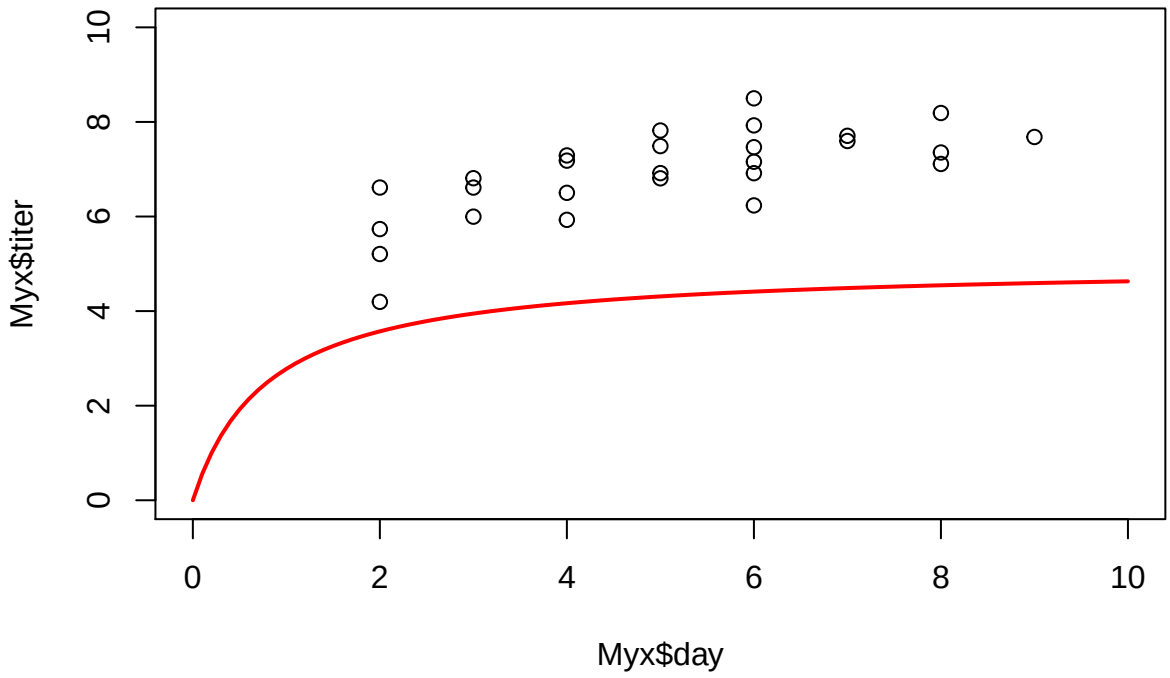
The M-M function looks like this:

$\frac{a \cdot x}{b+x}$

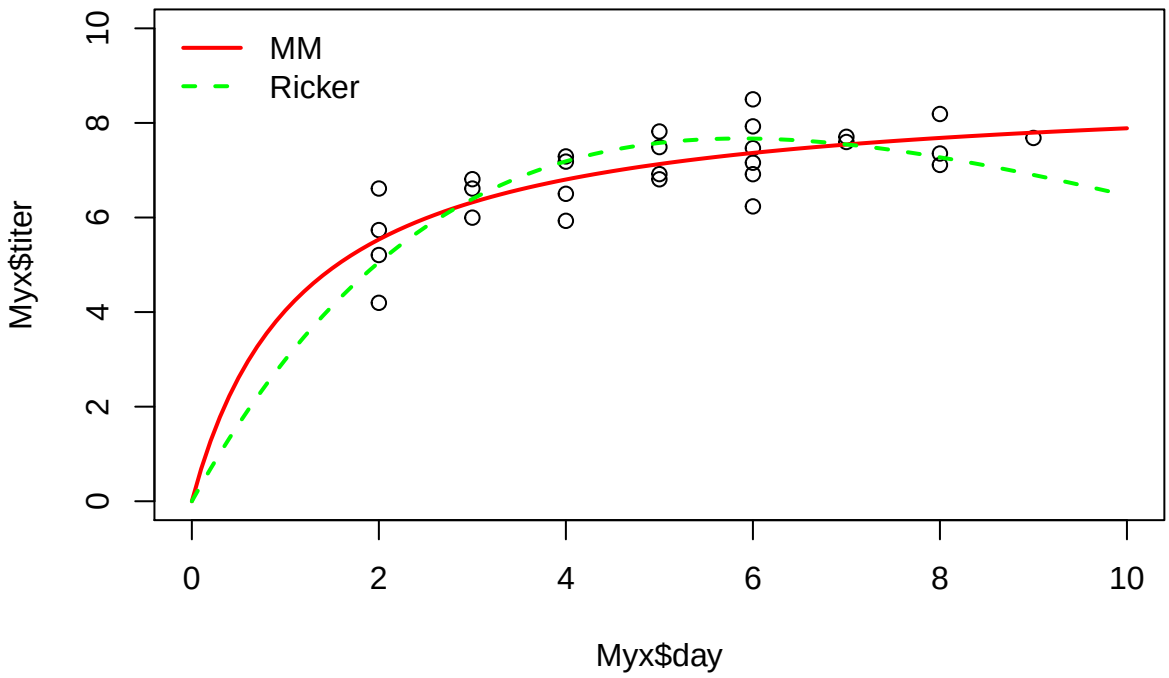
```
mm = function(x,a,b) { a*x / (b+x) }
```

You can plot an M-M function over the points to get some initial parameter estimates.

```
plot(Myx$titer~Myx$day,xlim=c(0,10),ylim=c(0,10))
curve(mm(x,a=5,b=0.8),from=0,to=10,add=T,col="red",lwd=2)
```



And here is an example of what your figure should look like with the best-fit M-M and Ricker functions overlaid on the same plot:



Challenge problem 3. Goodness-of-fit and model comparison

Let's try a different approach for visualizing the goodness-of-fit of these two models. For each of these models, use the pseudocode provided below to evaluate goodness-of-fit. On the basis of a *posterior predictive check* (see below), how well would you say each model fits the observed data? Do you see any red flags that would indicate poor goodness of fit? Explain your reasoning.

- **Exercise 3a**. Write a function called "Myx_PostPredCheck()" that compares the goodness-of-fit for the two models (ricker vs M-M) using a posterior predictive check.
  - **input**:
    - MCMC1 = the object generated from running the 'jags()' function with the Ricker function.
    - MCMC2 = the object generated from running the 'jags()' function with the Michaelis-Menten function.
  - **suggested algorithm**:
    - For each model (Ricker, M-M), generate new data under the fitted model. For each of an arbitrarily large number (e.g., 1000) of replicates, sample from the joint posterior distribution:
      - pick a random integer between 1 and the number of MCMC samples (e.g., call that number "random_index")
      - Then, use that random index to select a single random sample from the joint posterior distribution. This will look something like this: `new.draw <- jags.fit$BUGSoutput$sims.list$a[random_inc`
      - Use those parameters (drawn from the joint posterior distribution) to simulate a data set (For each observation, generate a single prediction under the data generating model)

- For each simulated data point, compute the squared residual error- that is, the squared difference between the simulated data point and the expected value from the Ricker or M-M model.
- For each observed data point, compute the squared residual error- that is, the squared difference between the observed data point and the expected value from the Ricker or M-M model.
- Compute the sum of squared errors (across all observations) for both the observed data and the simulated data corresponding to each sample from the joint posterior distribution.

* For each model (Ricker, M-M), use the 'boxplot()' function to display the range of plausible data that could be generated under the fitted model (using the simulated data). Overlay the observed data. This provides a visual goodness-of-fit evaluation. Based on a visual inspection, how well does the model fit the data?
* Plot the sum of squared error for the simulated data sets (Y axis) against the sum of squared error for the actual data set. Overlay a 1:1 line (that is, the line corresponding to y=x). This is called a *Posterior Predictive Check*.
* Compute the percent of the time that the discrepancy metric (SSE) for the simulated data exceeds the discrepancy metric for the observed data. This quantity is often called a *Bayesian p-value*.

– **return**:

* a list of length 3 containing (1) a vector of length 2 storing the Bayesian p-values for the two models (Ricker first, then M-M), computed above, (2) a data frame with number of rows equal to the number of simulation replicates and columns indicating, respectively, the samples from the joint posterior, SSE for the simulated data, and SSE for the real observations (the data used to compute the Bayesian p-value) for the Ricker model, and (3) a data frame with number of rows equal to the number of simulation replicates and columns indicating, respectively, the samples from the joint posterior, SSE for the simulated data, and SSE for the real observations (the data used to compute the Bayesian p-value) for the M-M model.

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 124
##
## Initializing model
##
##
  |
  |                                                      |   0%
  |
  |+++++++++++                                           |  20%
  |
  |+++++++++++++++++++++                                 |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
```

```
|                                                    |   0%
|
|*                                                   |   2%
|
|**                                                  |   4%
|
|***                                                 |   7%
|
|****                                                |   9%
|
|******                                              |  11%
|
|*******                                             |  13%
|
|********                                            |  16%
|
|*********                                           |  18%
|
|**********                                          |  20%
|
|***********                                         |  22%
|
|************                                        |  24%
|
|*************                                       |  27%
|
|**************                                      |  29%
|
|****************                                    |  31%
|
|*****************                                   |  33%
|
|******************                                  |  36%
|
|*******************                                 |  38%
|
|********************                                |  40%
|
|**********************                              |  42%
|
|***********************                             |  44%
|
|************************                            |  47%
|
|*************************                           |  49%
|
|***************************                         |  51%
|
|****************************                        |  53%
|
|*****************************                       |  56%
|
|******************************                      |  58%
|
```

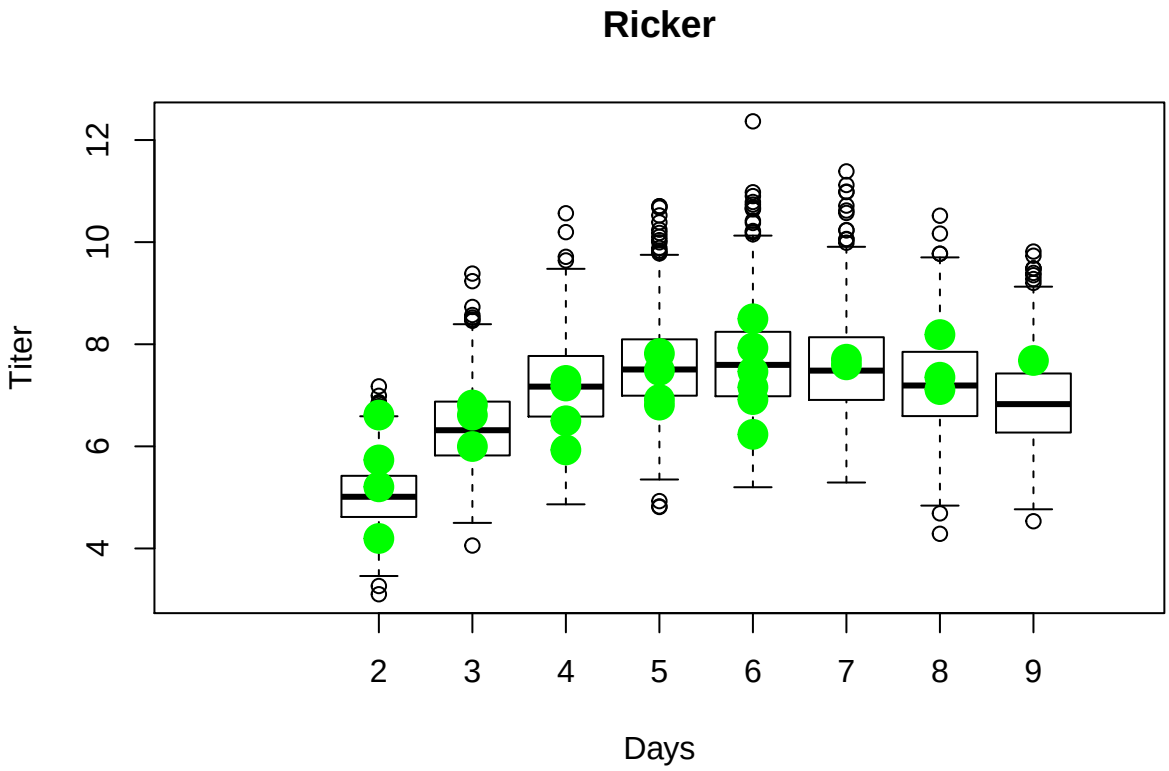
```
  |*****************************                    |  60%
  |
  |******************************                   |  62%
  |
  |*******************************                  |  64%
  |
  |********************************                 |  67%
  |
  |*********************************                |  69%
  |
  |**********************************               |  71%
  |
  |***********************************              |  73%
  |
  |************************************             |  76%
  |
  |*************************************            |  78%
  |
  |**************************************           |  80%
  |
  |***************************************          |  82%
  |
  |****************************************         |  84%
  |
  |*****************************************        |  87%
  |
  |******************************************       |  89%
  |
  |*******************************************      |  91%
  |
  |********************************************     |  93%
  |
  |*********************************************    |  96%
  |
  |************************************************ |  98%
  |
  |************************************************|  100%
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 27
##    Unobserved stochastic nodes: 3
##    Total graph size: 96
##
## Initializing model
##
##
  |
  |                                                  |   0%
  |
  |++++++++++                                        |  20%
  |
```

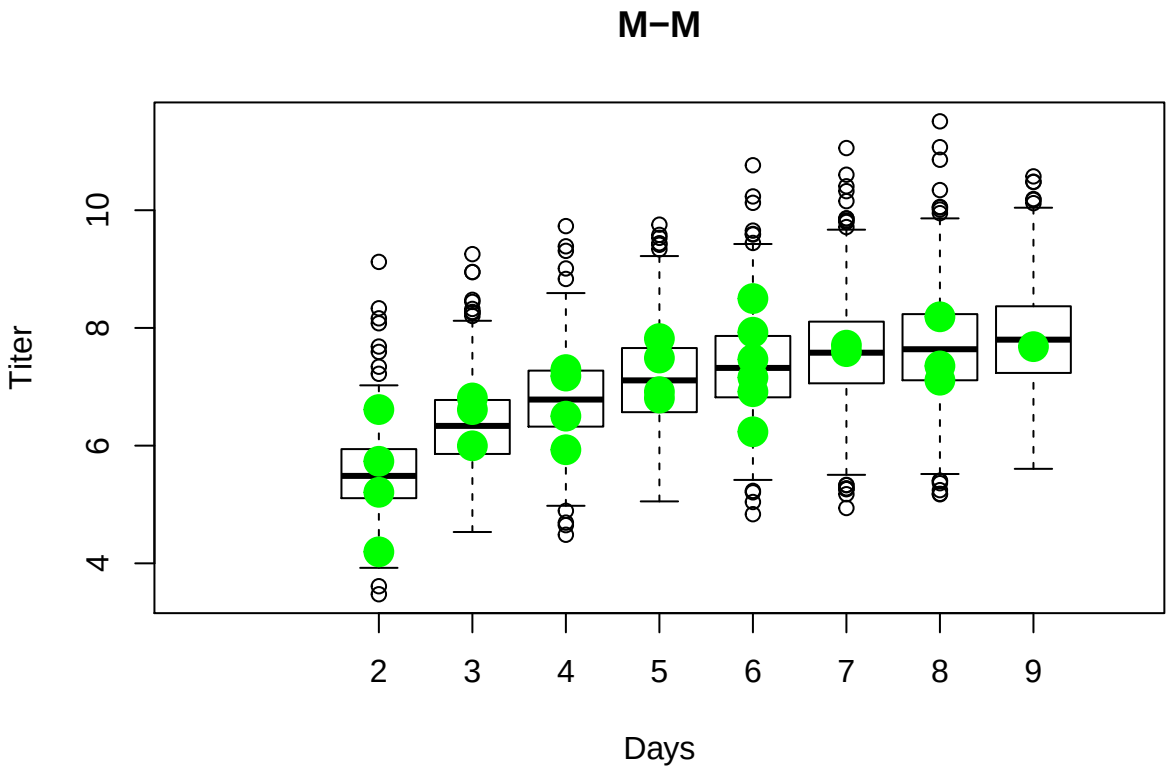
```
  |+++++++++++++++++++                                   |  40%
  |
  |++++++++++++++++++++++++++++++                        |  60%
  |
  |++++++++++++++++++++++++++++++++++++++                |  80%
  |
  |++++++++++++++++++++++++++++++++++++++++++++++++++++++| 100%
##
  |
  |                                                      |   0%
  |
  |*                                                     |   2%
  |
  |**                                                    |   4%
  |
  |***                                                   |   7%
  |
  |****                                                  |   9%
  |
  |******                                                |  11%
  |
  |*******                                               |  13%
  |
  |********                                              |  16%
  |
  |*********                                             |  18%
  |
  |**********                                            |  20%
  |
  |***********                                           |  22%
  |
  |************                                          |  24%
  |
  |*************                                         |  27%
  |
  |**************                                        |  29%
  |
  |****************                                      |  31%
  |
  |*****************                                     |  33%
  |
  |******************                                    |  36%
  |
  |*******************                                   |  38%
  |
  |********************                                  |  40%
  |
  |**********************                                |  42%
  |
  |***********************                               |  44%
  |
  |************************                              |  47%
  |
  |*************************                             |  49%
```

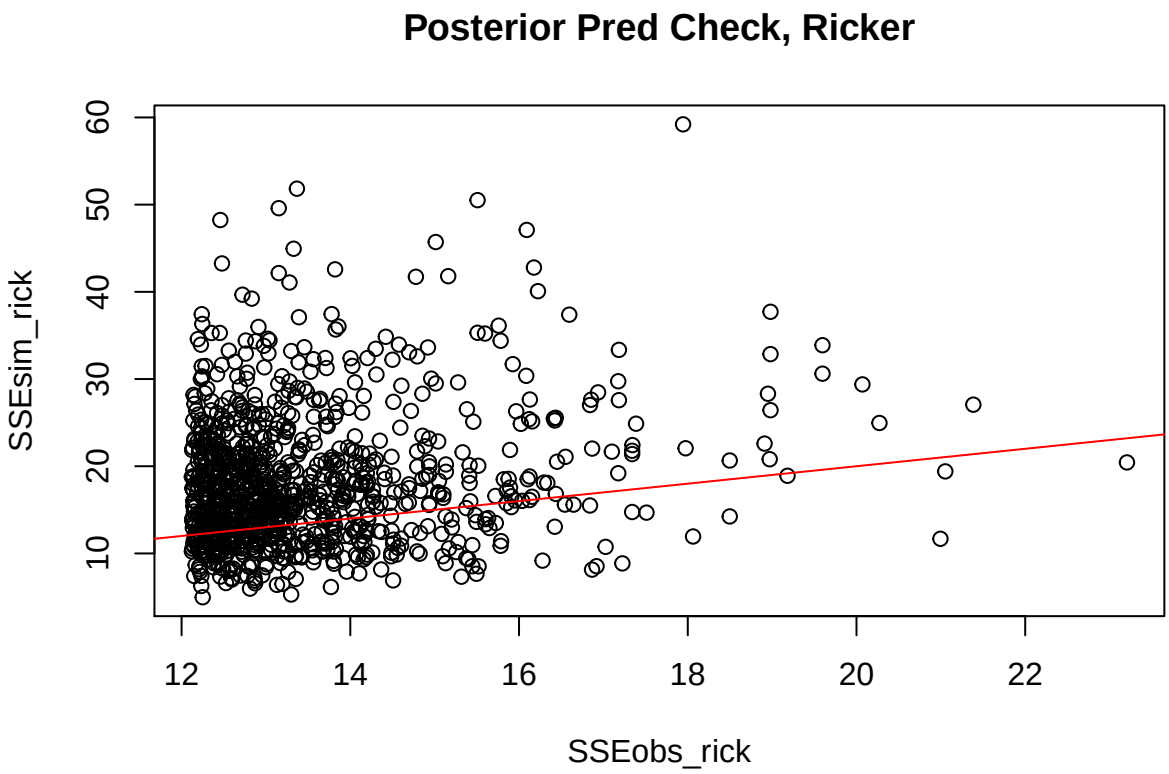
```
  |
  |*************************                             |  51%
  |
  |*************************                             |  53%
  |
  |**************************                            |  56%
  |
  |**************************                            |  58%
  |
  |***************************                           |  60%
  |
  |***************************                           |  62%
  |
  |****************************                          |  64%
  |
  |****************************                          |  67%
  |
  |*****************************                         |  69%
  |
  |*****************************                         |  71%
  |
  |******************************                        |  73%
  |
  |******************************                        |  76%
  |
  |*******************************                       |  78%
  |
  |*******************************                       |  80%
  |
  |********************************                      |  82%
  |
  |********************************                      |  84%
  |
  |*********************************                     |  87%
  |
  |*********************************                     |  89%
  |
  |**********************************                    |  91%
  |
  |**********************************                    |  93%
  |
  |***********************************                   |  96%
  |
  |************************************************************ |  98%
  |
  |************************************************************| 100%
```

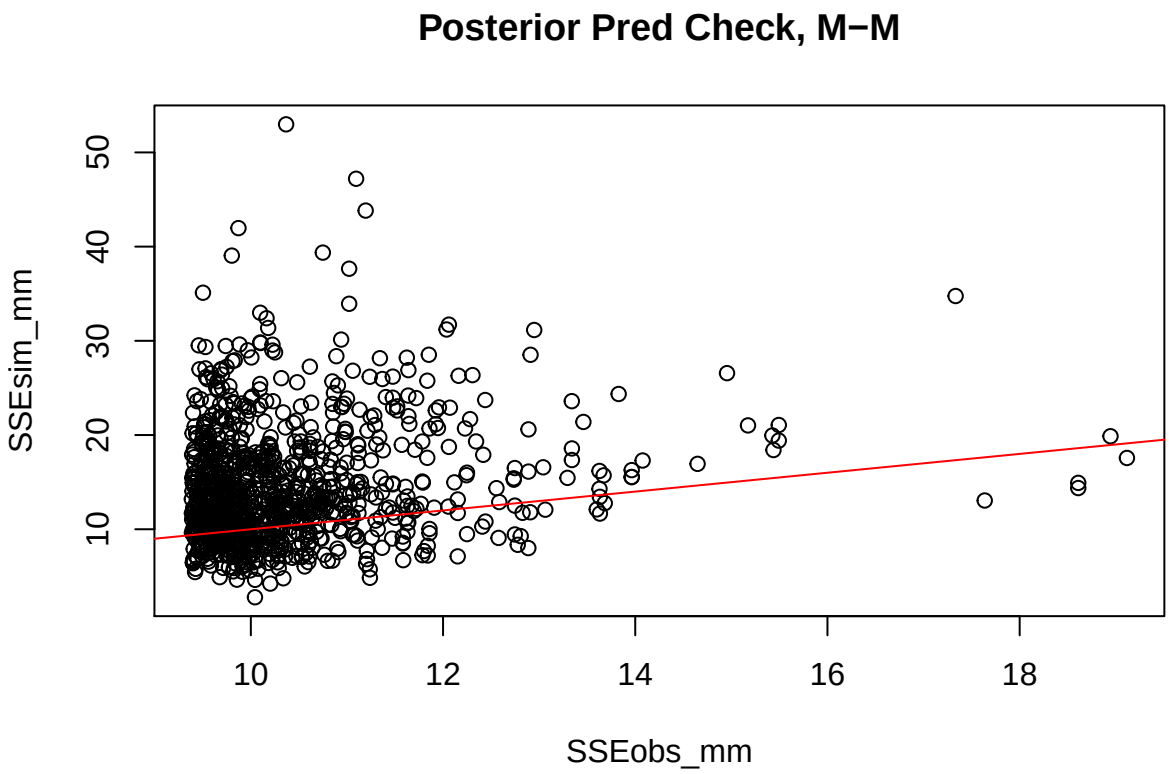
And here is what the results of your function should look like!

```
####
# Test the function

test <- Myx_PostPredCheck(MCMC1=jags.fit_ricker,MCMC2=jags.fit_mm)
```


Ricker
Titer
4
6
8
10
12
2
3
4
5
6
7
8
9
Days


M−M
Titer
4
6
8
10
2
3
4
5
6
7
8
9
Days


Posterior Pred Check, Ricker
SSEsim_rick
10
20
30
40
50
60
12
14
16
18
20
22
SSEobs_rick




```
test[[1]]   # p-vals
```

```
## [1] 0.719 0.749
```

```
head(test[[2]])    # ppc for ricker

head(test[[3]])    # ppc for M-M
```

- **Exercise 3b**. In your Word document, please explain how do you interpret the Bayesian p-value. What Bayesian p-value would you expect if the model fit was perfect? What would you expect if the model fit was poor? What does the Bayesian p-value tell us about how well the Ricker and M-M models fit the data? Does one model fit better than the other?

Challenge Problem 4 (extra credit)

DIC (analogous to AIC) is a model selection criterion for Baysian models fitted with MCMC. Just like for AIC, models with smaller values of DIC are favored. Which model (M-M or Ricker) is the better model based on DIC? What does the difference in DIC mean in terms of the strength of evidence in support of one of these models as the "best model"? [You should be aware that DIC is not a perfect solution for Bayesian model selection. For one, it is only valid if the posterior is approximately multivariate normal. Also, DIC tends to select overfitted models!]

WAIC (also analogous to AIC) is a newer (and better) model selection criterion for Baysian models fitted with MCMC. Just like for AIC, models with smaller values of WAIC are favored. Which model (M-M or Ricker) is the better model based on WAIC? What does the difference in WAIC mean in terms of the strength of evidence in support of one of these models as the "best model"?

```
## Warning: 1 (3.7%) p_waic estimates greater than 0.4. We recommend trying
## loo instead.
```

```
## Warning: 2 (7.4%) p_waic estimates greater than 0.4. We recommend trying
## loo instead.
```

Here are the results I got:

```
DIC_ricker
```

```
## [1] 65.93683
```

```
DIC_MM
```

```
## [1] 59.61532
```

```
WAIC_ricker$estimates["waic",]
```

```
##  Estimate        SE
## 65.967948  8.510563
```

```
WAIC_mm$estimates["waic",]
```

```
##  Estimate        SE
## 60.023988  9.081519
```