**FISH 458/558 - Fish Population Dynamics**

**LAB ASSIGNMENT #7 (Stock Recruitment models)**

*Complete and return your assignment (to Canvas by the due date) in the form of a Word document (with any answers and figures requested and with the R script copied in).*

*Guidelines:*

* *Include your Name, course, lab number, and date at the top of the document*
* *Number and label the questions and answers clearly! (We should easily be able to find your answers!)*
* *Include all of the requested output (e.g., values, data tables, and plots), not just the code for them. (We will not copy your code into R to see if it works).*
* ***Include informative captions for figures and tables. See research articles for examples.***
* *Submit a Word document unless directed otherwise (no r files or pdfs please).*
* *Include all your code used for the problems.*
* ***Answer ALL questions using complete sentences that are clear and informative****.*

**Stock-Recruitment (458 & 558 Students)**

1. Download the simulated stock-recruitment data from the “**SR\_data.csv**” file in CANVAS. Assume the units for the spawners is thousands of fish, and tens of thousands for recruits. Fit the density independent, Beverton-Holt, and Ricker stock-recruit (SR) models to the dataset, assuming a multiplicative error structure. (12 pts)
   1. Generate a single graph with the three model fits. Include a legend, figure captions, and appropriate units. (4 pts)
   2. Generate diagnostic plots for each of the three models to evaluate the assumptions of normality and homogeneity of variance. Explain whether the assumptions of normality and homogeneity of variance have been met for each of the models. (6 pts)
   3. Make a table of Akaike’s Information Criterion (AIC) and delta AIC (dAIC) values for each model and use that to determine which of the three models is the best. (2 pts)
2. [**558 GRAD STUDENTS MAY SKIP THIS QUESTION IF THEY CHOOSE, BUT KEEP THE SAME NUMBER FORMATTING AS THIS FILE**]. Using the same data set, fit the Shepherd stock-recruitment model, which is a different SR model that has a more flexible shape. For this you will need to modify the code from lab for this new model, which is a common task (taking existing code and modifying it for your needs). The equation is below. (8 pts)



where a= a productivity parameter, b=a density dependent parameter, and c = a general shape parameter (c<1 gives a density-independent shape; c=1 gives a Beverton-Holt shape; and c>1 gives a Ricker shape).

* 1. Fit the model and state what your best-fit estimates are for the three model parameters: a, b and c. (4 pts)
     1. **HINTS**: (1) make sure you enter the equation with the appropriate parentheses, and operation symbols like a\*S, not aS. (2) make sure to update the names of any objects you create, like mod4 and SEE.mod4. (3) If you get “Error in numericDeriv…” you probably have bad starting values for your parameters. Because the model takes the shape of the Beverton Holt model if c=1, you can use c=1 as the starting value along with whatever starting values you had for the Beverton Holt model in question 1. (4) Your “best-fit estimates” are the estimates you get from the summary of your model after running your code, and they are NOT the *starting* values that you had to plug into your code to get the model to work.
  2. Re-do your plot from question 1a, and include the Shepherd model in your graph. (2 pts)
  3. Is this new model better than the others? Explain why or why not. Use AIC to find out. (2 pts)

1. You are talking with your cranky grand-uncle and you told him with glee that you got your code to work and you just fit some stock-recruitment models in R. Gramps says “What the heck is a ‘stock-recruitment model’? And why are they even important?!” How would you answer him in layman’s terms? Explain this class of model and provide at least 3 reasons for their importance. (3 pts)
2. Answer these questions (2 pt):
   1. How many hours did you spend on this assignment as a whole?
   2. Take a selfie of yourself working with at least one more student from the class.
   3. Were there any particular things you struggled with in this lab and how did you overcome them?

**558 Students – Simulating data.**

1. Simulating data can be very handy and you may find simulated data to be of use for your projects. **Your goal is to create a function to generate random data points and to use this to explore the influence of parameter values on the shape of a stock recruitment function.** For this, you will be simulating data that conform to the Ricker Stock recruitment model. The Ricker model is:

**** with ε~N(0, σ2).

Where R is a measure of recruitment, S is Spawner abundance or biomass, a is the productivity parameter, b is the density dependent parameter, and ε is the random error that is normally distributed with a mean of zero and a variance of σ2.

Although this may be one of your first functions you create, I provide some guidance and a code skeleton below. Also, I have posted a script with an example function (**“Lab 01 HW\_answers – Function Example.r”)** that will likely be of help. Here are some general considerations for you as you embark on your task:

1. First, get your script working for a given set of parameters without worrying about having it be a function. Once you have the script working, then you can wrap it in the function and test it.
2. To simulate your spawner data (**S.sim**), we can use something simple like this:

**S.sim = seq(0, 1500, length=50)**

1. Calculate new recruitment data (**R.sim**) based on the S values from part (ii) using the Ricker model as specified above. To start, use *a=3* and *b=0.001* as initial, default values. You will need to build in multiplicative random errors (also known as lognormal deviates) by multiplying your predictions by exp(ε) where ε is a random, normally distributed number with a mean of zero and a standard deviation, σ (or **e.sd**, for the “error’s standard deviation” in code below). Random numbers can be generated using function **rnorm()**.
2. Once you’ve generated new recruitment values, plot them (R vs. S). Set up your plot with appropriate axis limits (where the scale captures the density dependent effects), and have your title indicate the values of your parameters. The sample script below has some code for automating the title using the **paste()** function.
3. Add in a line to your plot that represents the Ricker model with no random error, based on your a and b parameters. *Note: at NO point for this exercise will you need to fit a model to the fake data you are generating using the nls() function*.
4. Once you can generate a plot (e.g., Figure 1), you can work on wrapping your code up in a function (which means your code goes inside of curly brackets {} in the code below).

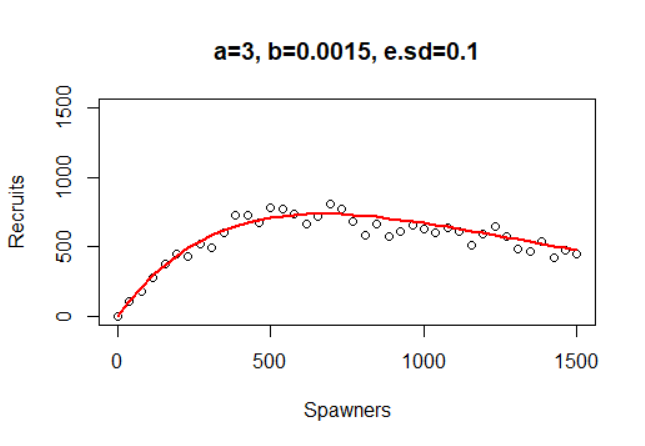


Figure 1. Example plot of the type of spawner-recruit plot your function should generate.

**Questions:**

* 1. **Using your function and holding the other parameters constant, explore the effect of changes to a, b, and e.sd individually (i.e., hold 2 parameters constant and change the third to see how the plots are affected). Describe the effect that each parameter has on the simulated data and provide graphs to depict the effect. (6 pts)**
  2. **How might a fisheries scientist use a stock-recruitment data simulation like this? What could be some potential applications? (1 pts)**
  3. **Was this the first function you’ve coded? How did it go? How could you envision using functions for any aspects of your research or futuer work? (1 pts)**

*#SAMPLE CODE FOR YOUR FUNCTION (Your function should look something like this):*

myfunc=function(a=3, b=0.0015, e.sd=0.1){ #This line is the start of the function you are creating

#(Note that the arguments within the parentheses above specify default values for the parameters)

#Script for calculating Recruits based on the parameters a, b, and e.sd goes here:

…

#Script for generating plot of R vs. S

#You can make the title indicate the parameter values automatically

#using: plot(…, main=paste("a=", a,", b=",b,", e.sd=",e.sd, sep="") )

#You may also want to specify specific xlim and ylim values so they are constants.

…

} #This is the end of the function

#Then you can run your function for different parameter values and examine your plot:

myfunc(a=5, b=6e-5, e.sd=0.3)

##NOTE: See Lab 01 and its handouts (or Google) for basics on functions. Also, I’ve posted code for a similar function I created to answer the 558 question from Lab 01 homework. See: **“Lab 01 HW\_answers – Function Example.r”**. That script will likely be helpful!