

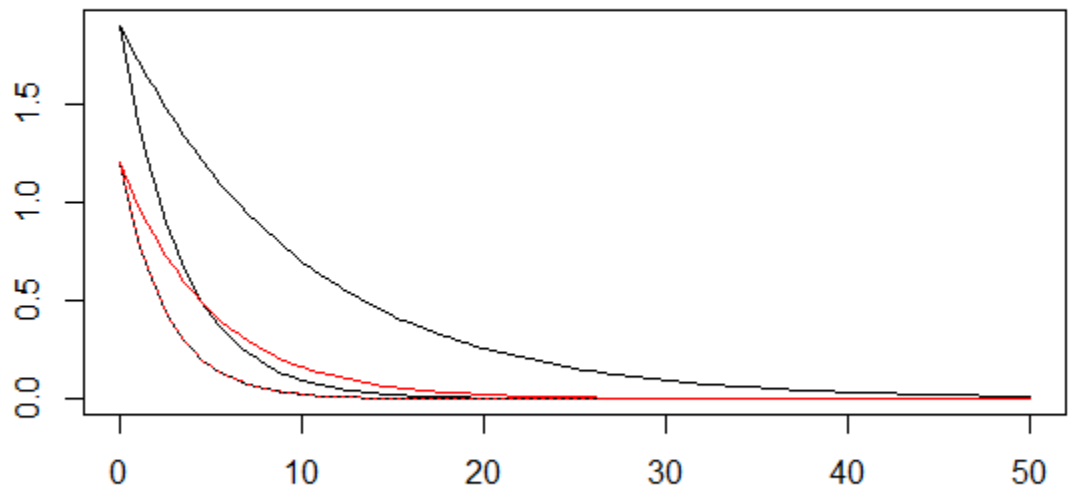
Keegan Moynahan **** worked on with steph

Lab 5

1)

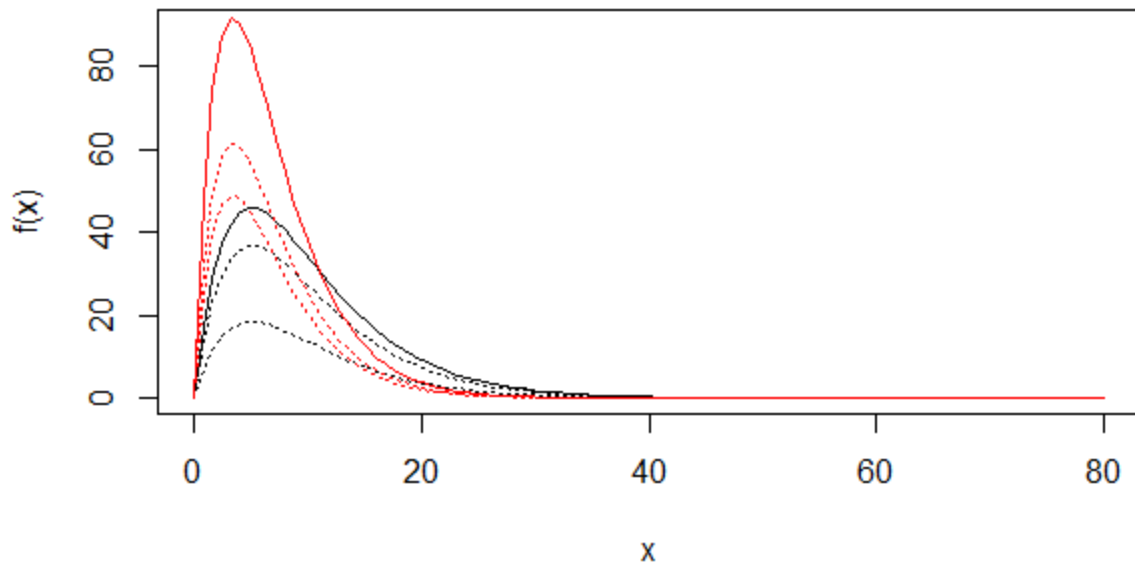
```
exp_fun = function(x, a, b)
{
  return(a* exp(-b * x))
}
```

2)



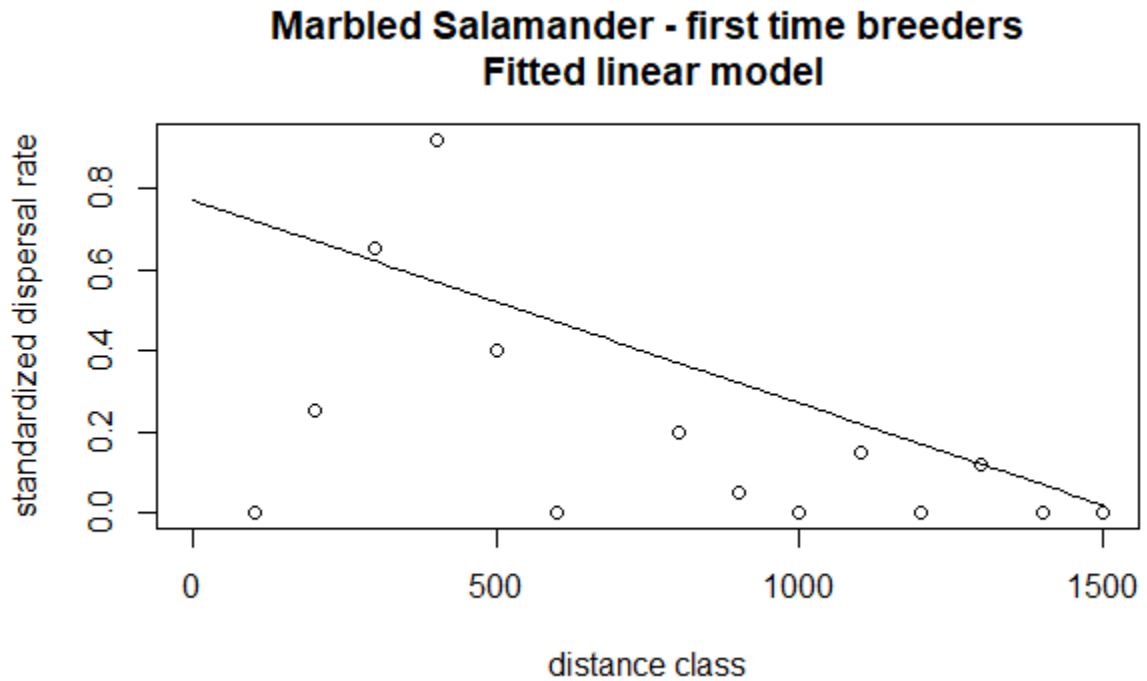
- 3) When changing the value of **a** in the function it changes the line on the y axis. When putting greater values, it created a steeper looking slope and moved the top of the line to the right in the graph. A lower value created a less steep slope and the line was moved to the left of the graph.
- 4) When changing the value of **b** in the function it changes the slope/depth of the line. For example, when putting a greater number, it sagged the line towards the x axis and when a smaller number was put in it moved away from the x axis.
- 5)

Ricker function: $a = 1, b = 1$



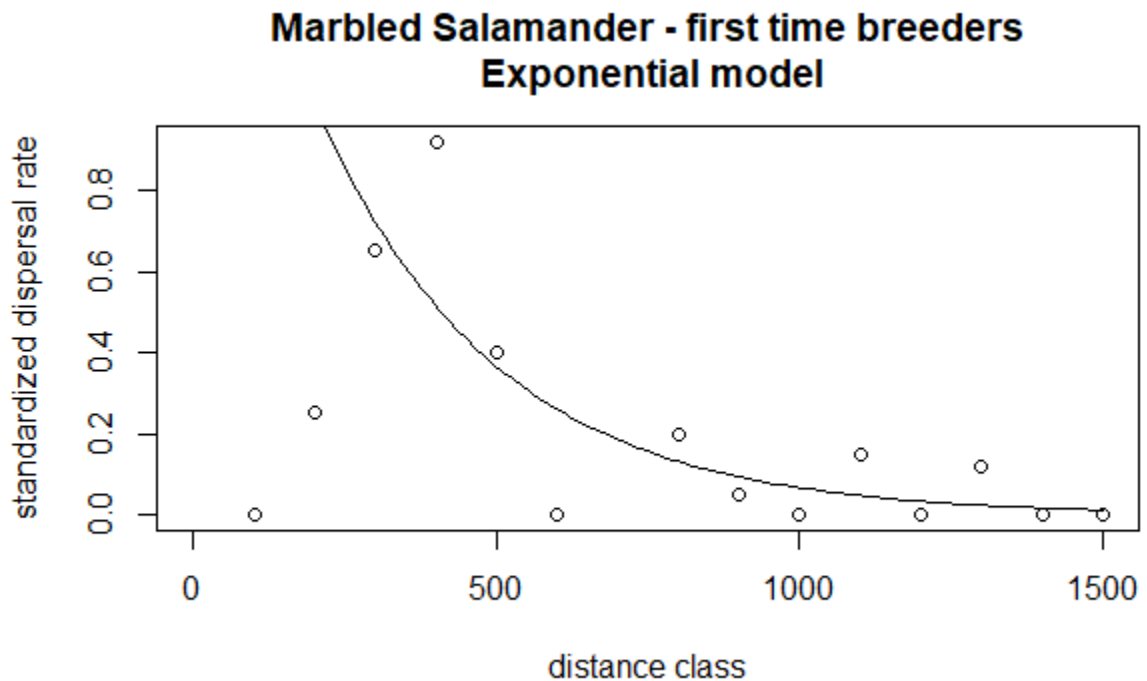
- 6) If you change the **a** value in the function, you are changing the apex of the line. A greater value will make the “hump” higher on the y axis and a smaller number will make the “hump” lower on the y axis.
- 7) If you change the **b** value, the depth of the curve changes. A larger number makes the depth smaller (hump less wide), and a smaller number will create a large depth (wider hump).
- 8) Slope = -0.0005 x1= 1400 y1=0.07 I chose this slope for the fitted linear model because even though there isn't much of a correlation, you can see that it is a negative correlation. I chose the x and y values by playing around with them until I found a combination that fit the area of the plot that had the best correlation.

9)



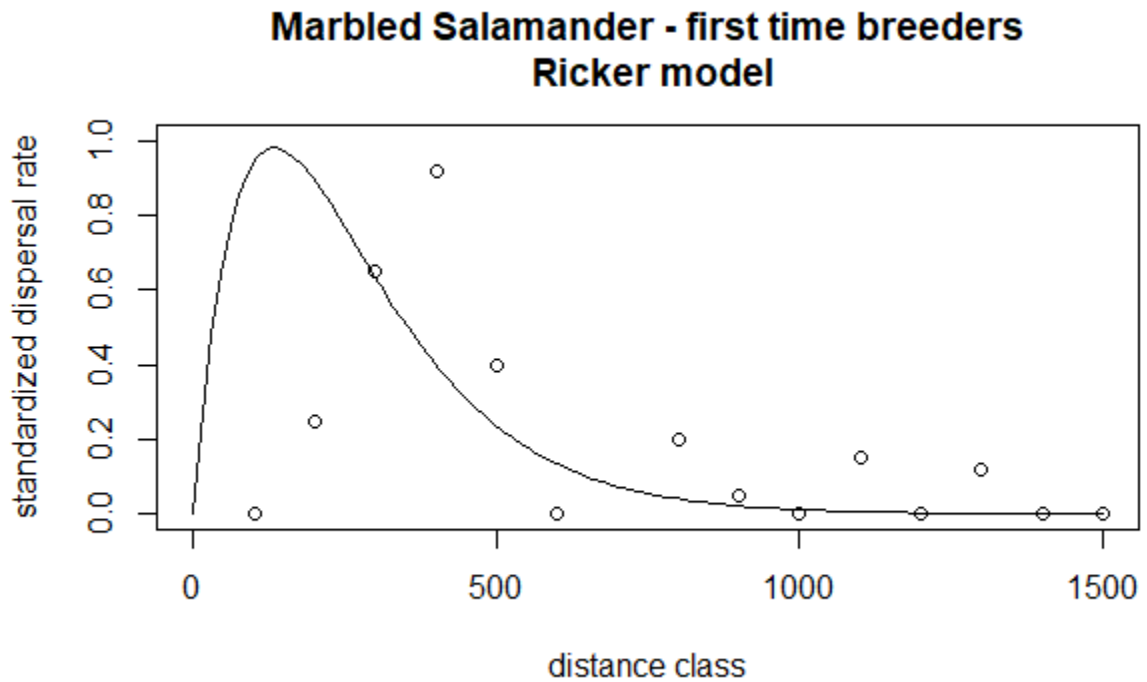
10) $A=2$ $b=0.0034$ These values were chosen because they best fit the area of the model that shows some correlation. These values produced the line that shows the negative correlation, and the points close to and away from the center of the data.

11)



12) $A = 0.02$ $B = 0.0075$ These numbers show how the beginning of the plot shoots up and then gradually slopes down for a negative correlation. I think this model is the best fit for this data and plot.

13)



14)

```
dat_sal$predicted_linear = line_point_slope(dat_sal$dist.class, 1400, 0.07, -0.0005)
```

```
dat_sal$residuals_linear = dat_sal$disp.rate.ftb - dat_sal$predicted
```

```
dat_sal$predicted_exponential = exp_fun(dat_sal$dist.class, 2, 0.0034)
```

```
dat_sal$residuals_exponential = dat_sal$disp.rate.ftb - dat_sal$predicted
```

```
dat_sal$predicted_ricker = ricker_fun(dat_sal$dist.class, 0.02, 0.0075)
```

```
dat_sal$residuals_ricker = dat_sal$disp.rate.ftb - dat_sal$predicted
```

15)

