

Juliana Berube

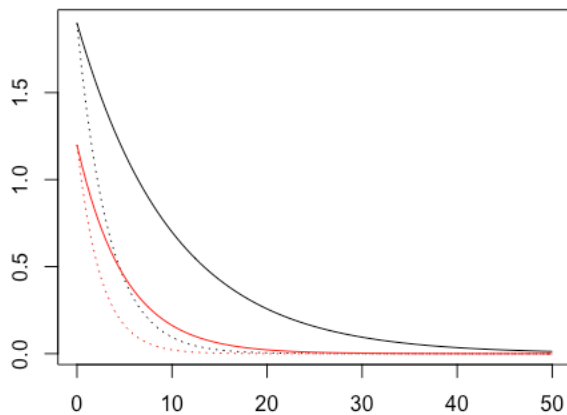
Lab05

*Worked with Jessica Bonin

1.

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

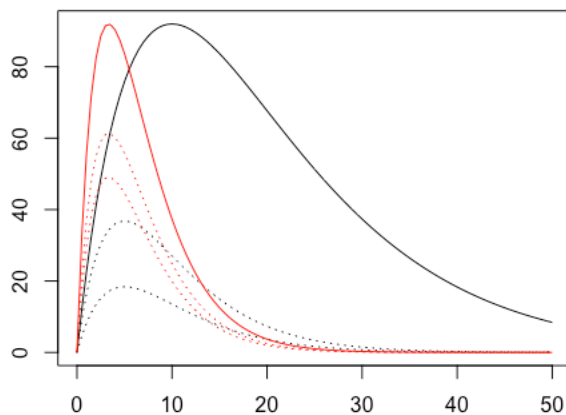
2.



3. A is where the x and y axis intersect, and it alters the values of the y axis.

4. Varying parameter b changes the depth of the curve. A lower b values (ex. 0.25) creates a less steep curve, and a higher b value (ex. 0.75) will create a steeper curve.

5.



6. A controls the height of the curve, and changing a also changed the values of the y axis according to that value.

7. Changing the b value changed the tightness of the function. The higher the value, the more quickly the curve will peak and descend. A lower b value widens out the curve.

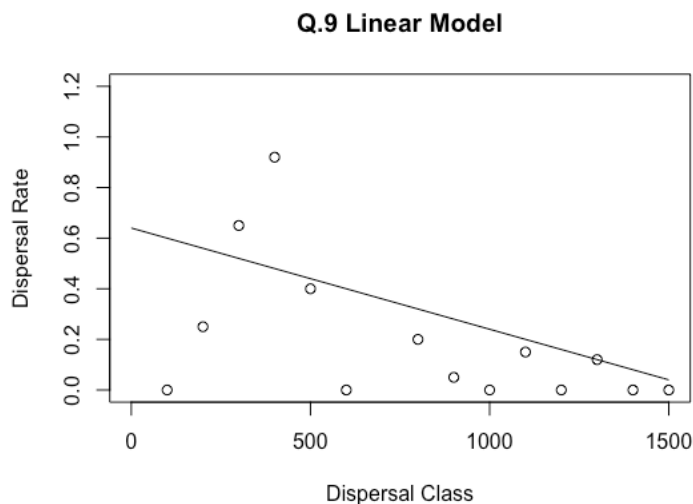
8. X: 600

Y: 0.4

Slope: -.0004

I chose the x and y values based on the x and y limits of the data and where I wanted the function to begin and sit in the graph, and I chose the slope based on the difference in values on the x and y axes. Since the difference is so large, the slope needed to be a small value to create the appropriate fit. Based on a visual inspection, these values seemed to create the best fit.

9.



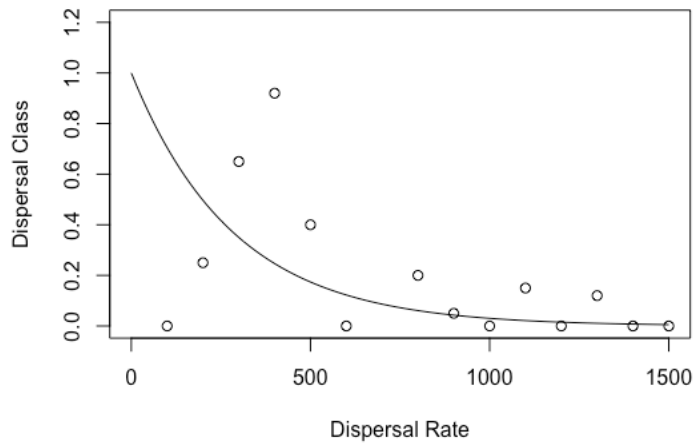
10. A: 1

B: 0.0035

I chose value a because a is where x and y intersect on the graph, so, I wanted the model to start at the beginning of the data. I chose value b because that is the decay factor, which needed to be very small, similar to above. Because the x and y values have such a big difference, b again needed to be very small to fit.

11.

Q.11 Exponential Model



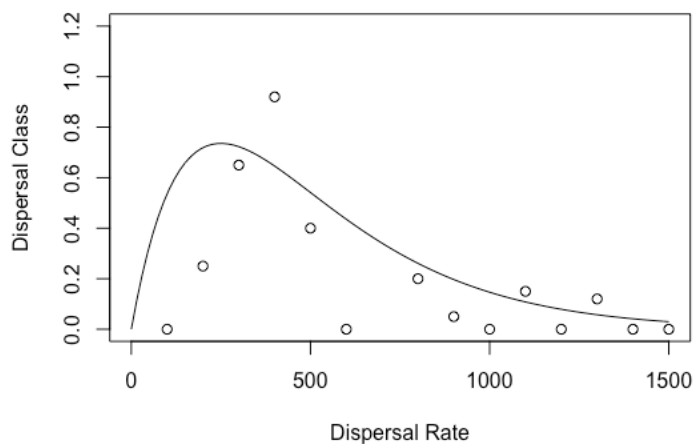
12. A: 0.008

B: 1/250

Value a controls the height of the curve, so I set value a to a small number because the slope will rise and fall quickly on the graph, especially due to the difference in x and y units. Value b I chose because that controls the x maximum, so I inspected the graph and approximately where I wanted it to peak.

13.

Q.13 Ricker Model



14.

```
#lin
```

```
y_predicted1<-line_point_slope(dispersal$dist.class, guess_x, guess_y, guess_slope)
```

```
resids_linear<-y_predicted1-dispersal$disp.rate.ftb
```

```
dispersal=cbind(dispersal, resids_linear)
```

```
#Exp
```

```
y_predicted2<-exp_fun(dispersal$dist.class, 1, .0035)
```

```
resids_exp=y_predicted2-dispersal$disp.rate.ftb
dispersal=cbind(dispersal, resids_exp)
```

```
#Ricker
```

```
y_predicted3=ricker_fun(dispersal$dist.class,.008, 1/250)
```

```
resids_ricker=y_predicted3-dispersal$disp.rate.ftb
```

```
dispersal=cbind(dispersal, resids_ricker)
```

15.

