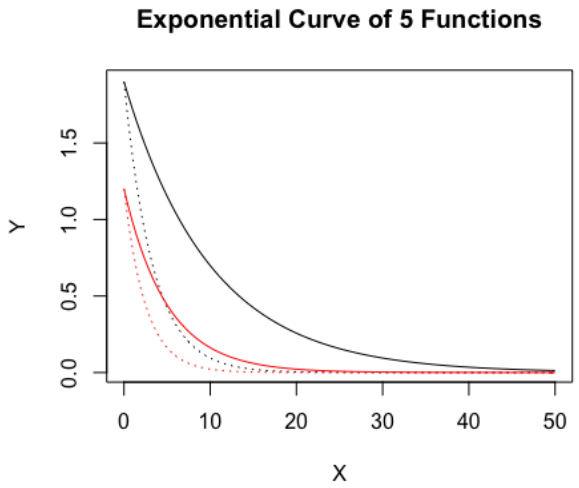
Chloe Lang
Professor Nelson
Lab 5 Report
October 17, 2021

Q1

```
plot(x=dat_dispersal$dist.class, y=dat_dispersal$disp.rate.ftb,
    main="Dispersment vs Distance",
    xlab="Distance",
    ylab="Dispersion Rate")
exp_fun=function(x, a, b)
{
 return (a * exp(-b*x))
}
```

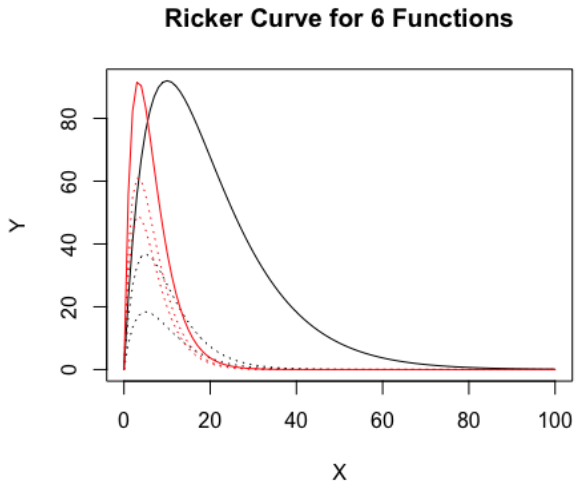
Q2



Q3 The curves are all very similar, but parameter a adjusts the height of the curve. The “a” parameter shows where the top height of the curve is in this function.

Q4 The “b” parameter is the exponent in which x is being multiplied. This shows the steepness/slope of the curve.

Q5

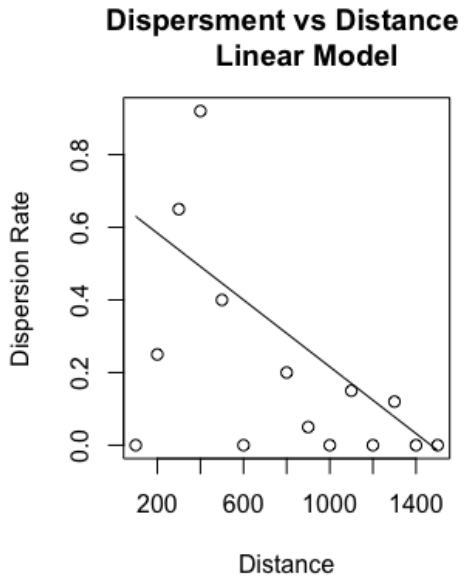


Q6 Parameter a also shows the height of the curve, showing the highest point of the curve.

Q7 Parameter b shows the slope or tightness of the curve. It defines how steep or not steep the actual curve is. It also determines when the curve starts to flatten. For example, a steeper slope occurs when b is larger.

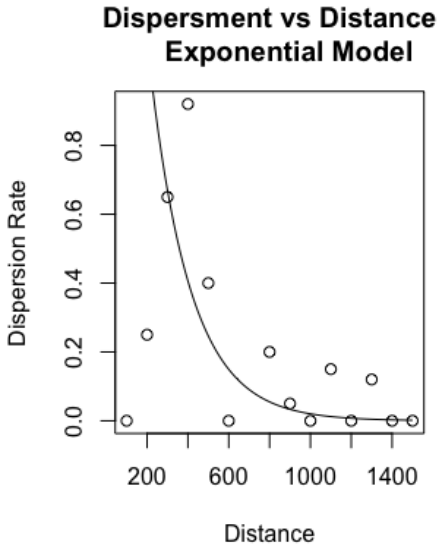
Q8 Slope= -0.00046, x1=600, y1=0.4. I chose these values by first using the locator() function to locate two points on the graph I wanted my linear function to go through. I manually calculated the slope from that. I then used the locator() function again to locate a central point I wanted my line to go through that would fit my data as best as possible.

Q9



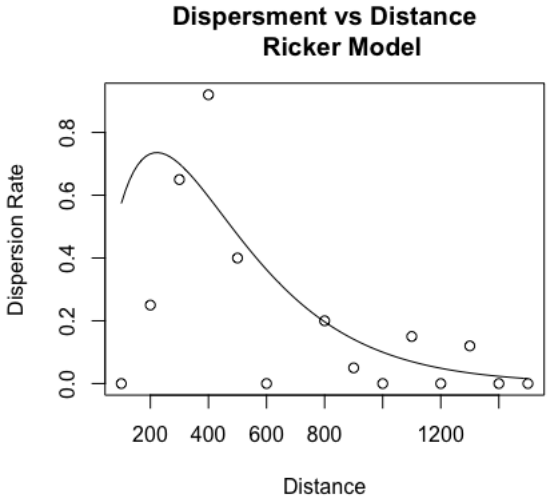
Q10 a=3, b=0.005. These values were chosen because I knew that a needed to be low in value since the scale on the y-axis is just below 1. For b, this value was chosen honestly through a lot of trial and error. I did several algebraic calculations to determine an approximate a and b value and then just adjusted the values slightly until I was able to find an exponential curve, I found to fit the data best.

Q11



Q12. a=0.009, b=0.0045. I chose these two values in a similar way that I did question 10. I was able to use my calculator to figure out approximate values of a and b. I originally had the correct values but was off by a decimal point. I would be interested to see if there is a way through R for it to calculate a fitted Ricker Function.

Q13



Q14

```
dat_dispersal$linear = line_point_slope(dat_dispersal$dist.class, 600,0.4,-0.00046)
dat_dispersal$exp_fun = exp_fun(dat_dispersal$dist.class, 3, 0.005)
dat_dispersal$ricker_fun = ricker_fun(dat_dispersal$dist.class,0.009,0.0045)

dat_dispersal$resids_linear = dat_dispersal$disp.rate.ftb - dat_dispersal$linear
dat_dispersal$resids_exp = dat_dispersal$disp.rate.ftb - dat_dispersal$exp_fun
dat_dispersal$resids_ricker = dat_dispersal$disp.rate.ftb - dat_dispersal$ricker_fun

par(mfrow=c(3,1))
hist(dat_dispersal$resids_linear,
    main = "Histogram of Linear Function Residuals",
    xlab = "Predicted Values",
    col = "darkseagreen2")
hist(dat_dispersal$resids_exp,
    main = "Histogram of
    Exponential Function Residuals",
    xlab = "Predicted Values",
    col = "lightpink")
hist(dat_dispersal$resids_ricker,
    main = "Histogram of Ricker Function Residuals",
    xlab = "Predicted Values",
    col="darkslategray2")
```

Q15