LAB 2. LIKELIHOOD & MAXIMUM LIKELIHOOD

Goals:
The goals of this lab are to familiarize you with the procedures required to obtain Maximum Likelihood Estimates (MLE) of scientific and statistical models. We will be using built-in R functions. You will need to install the bbmle (Ben Bolker's MLE package) for the second part of the lab. We have uploaded an excellent primer Tom developed for bbmle to the course website.

PART 1. From *Bolker*. The first application of maximum likelihood estimation is when we have a collection of observations that follow a particular distribution and we want to estimate the *true* parameters of the distribution. The method of moment estimates are typically biased and Maximum Likelihood estimates are preferred. To convince you, we are going to simulate a data set and calculate the parameters of the distribution using both the MOM and ML. Start by simulating some negative binomial data:

```
> mu.true<-1
> k.true<-0.4
>x<-rnbinom(50,mu=mu.true,size=k.true)
```

Take a look at the data using some graphics.

```
> plot (table(factor(x,levels=0:max(x))),ylab="Frequency",xlab="x")
```

Now we will build a function that calculates the negative log-likelihood for this distribution given a set a parameters. Arguments for this function are p, the vector of parameters, and dat the vector of data:

```
> NLLfun1 <- function( dat = x, mu, k ) {
-sum(dnbinom(x, mu = mu, size = k, log = TRUE))
 }
```

First we will calculate the negative log-likelihood with the true values of the distribution. We have to combine these values into a vector to be able to pass them to the NLL function:

```
> nll.true <- NLLfun1(mu = mu.true, k = k.true)
> nll.true
```

Let's find the method of moments estimate of the parameters. From last yesterday you know that:

```
> m = mean(x)
> v = var(x)
> mu.mom = m
> k.mom = m/(v/m - 1)
```

and the negative log-likelihood estimate for method of moments parameters:

```
> nll.mom <- NLLfun1(mu = mu.mom, k = k.mom)
> nll.mom
```

What is the difference in the likelihood of the two estimates? The LRT test would say that it has to be greater than a chi-square with two degrees of freedom (0.95)/2. Is it?:

```
>Ldiff<-nll.true-nll.mom
>qchisq(0.95,df=2)/2
```

So there doesn't appear to be a difference in the NLL values of the true and mom parameters. How about the MLE estimates? We use mle2 with the default Nelder-Mead algorithm (more about this later…) and we use the MOM estimates as starting conditions:

```
> sol1=mle2(minuslogl=NLLfun1,start=list(mu=mu.mom, k=k.mom),hessian=TRUE)
> summary(sol1)
```

Is the NLL value better than for the MOM? By how much?

Let's find likelihood surfaces, profiles and confidence intervals.

```
>confint(sol1)
>plot(profile(sol1)
```

Exercise 1: Generate data of your choice using one of the probability functions you learned during the last lab. Calculate the MOM estimates of the parameters. Calculate the MLE and estimate the difference in likelihood and in prediction. Calculate the support intervals for the parameters.

PART 2. Coates and Burton (1999) studied the influence of light on growth rates of conifers in British Columbia. They used the model:

$$\mu_i = \frac{\alpha(L_i - c)}{\left(\frac{\alpha}{\gamma}\right) + (L_i - c)} \qquad \text{[Eqn. 1]}$$

where:

μ=predicted growth rate
α =maximum growth rate
γ=slope of curve at low light
c=light level with growth =0
L=measurement of light availability.

Obtain MLE parameters of the model using Solver in Excel. OK, I know we have invested time in learning R. Why on earth are you asking us to work in Excel? There is a reason for this. When you write code in R, it is easy to fail to understand what is happening "under the hood". The structure of an MLE analysis is much more transparent when you are forced to build a spreadsheet.

Open the spreadsheet named "*Hemlock-light-data.csv*". You want to add columns and cells demonstrating that you know how likelihood works.

Let's look at these columns and rows. This is the benefit of this exercise so linger on this. Columns A and B are the data. Column C contains the predictions of your model for each level of light, predictions that depend on the values for α, γ, and c. Column D contains the likelihood, one value for each data point. The formula for this cell is:

= NORMDIST(B2,C2,sigma, FALSE)

You could also calculate it yourself using the formula for a normal PDF.

$$f(y_i \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(y_i - \mu)^2}{2\sigma^2})$$

$$Mean = \mu$$

$$Variance = \sigma^2$$

Make a plot of the data and the model predictions. Adjust the model parameters by hand until you get a reasonable fit to the data. When you have your spreadsheet constructed, you will use Solver to find t he MLE of the parameters. You can find Solver in the Data tab. Solver is a sophisticated non-linear numerical optimizer. It searches for values of the parameters that maximize or minimize the output of a function.

You will put the cell containing the sum of the log likelihoods in the **Set Target Cell** field. The cells that contain the parameter values will go in the **By Changing Variables Cells**. Most likely you will be able to do this exercise without putting constraints on the parameter values if you give them reasonable starting values.

What are the MLE values for the parameters?

Exercise 2: Read the hemlock growth file (you can use the read.csv function) we used for the first exercise. You will need to remove the columns with parameters (columns H:I) before reading in the file into R. Check your results using the nls (non-linear least-squares estimates) function in R, which does non-linear estimation of parameters for normally distributed data. Estimate the parameters using nls and compare them to those obtained in the Excel exercise.

For nls, you will need to provide a function that describes your model (i.e., tree growth as a function of light as shown in Eqn. 1) and starting values for the parameters you want to estimate (alpha, gamma and c in Eqn. 1). You can get these values by inspection or from the previous exercise in Excel. Remember: they are starting values only. FOLKS ARE GETTING HUNG UP ON THE SYNTAX OF NLS. ADD SOME DETAILS ON THIS.

Repeat the exercise by writing your own likelihood function instead of nls.

PART 3. Let's look at another dataset. Schmitt *et al.* (1999) explored reef fish recruitment using a dataset from 603 lagoons in Polynesia. I have provided you with a simulated data set that resembles their data (*Reef_fish.txt*). Read the data in and plot it. Looking at the data, does the following scientific model seem reasonable to you?

$$Recruits = \frac{aSettlers}{1 + \frac{a}{b} Settlers}$$

Plot the frequency distribution of the data. What probability distribution(s) would you choose? Why?

Assuming your probability distribution of choice, how would you go about calculating MLE for your scientific model and statistical distribution?

First, we define our scientific model:

```
>adult.recruits<-function(a,b,settlers){a*settlers/(1+(a/b)*settlers)}
```

and our statistical distribution. I will choose a binomial distribution since I am interested in the probability of settlement, that is, an easy way of thinking about this is to say:

$$\Pr ob.\mathrm{Re}cruitment = \frac{a}{1+\frac{a}{b}Settlers}$$

which turns this into a very straight-forward binomial probability. Now our likelihood function would be:

```
>NLLfun<-function(a,b) {
recprob=a/(1+(a/b)*settlers)
-sum(dbinom (recruits, prob=recprob, size=settlers, log=TRUE), na.rm=TRUE)
}
```

Now we need to call an optimization procedure (**NOTE**: R is notoriously fickle about optimization problems and you may end up using a number of different packages to get the procedure to work).

```
>results<-mle2 (minuslogl=NLLfun,start=list(a=0.5,b=10),method="L-BFGS-B",lower=0.003)
```

You will get some error messages from R because it is trying to calculate the likelihood. This is because when you have 0 settlers the probability of having 0 recruits is 1 but according to your function it is *a*. It will still generate results though.

You could now calculate the NLL by plugging the solutions into the likelihood function and plot the fitted curve using the estimated parameters:

```
a<-coef(results)["a"]
b<-coef(results)["b"]
plot(settlers,recruits)
curve(a*x/(1+(a/b)*x), add=TRUE)
```

The negative loglikelihood is:

```
NLL<-NLLfun(a,b)
```

Exercise 3: We chose the binomial likelihood function for convenience. However, there are a number of other possibilities. Choose another likelihood function that fits the data and calculate the NLL. How would you generate starting values for this new distribution? Is it better or worse than the binomial?