Likelihood

```
library(ggplot2)
library(bbmle)
```

Overview

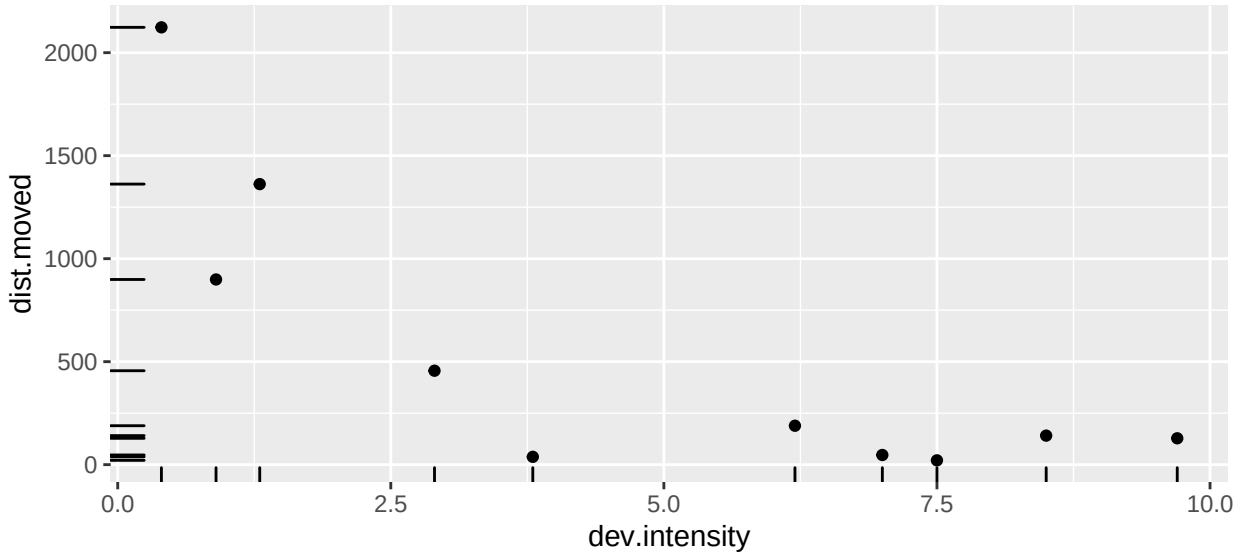
Likelihood is a fundamental idea that is foundational to much of current practice in statistics and data analysis. Although we rarely need to work with it directly, it seems useful to understand how it works.

This lab uses material from a lab developed by Kevin McGarigal at UMass.

Setting

Consider a hypothetical study of moose movement patterns in relation to development intensity. Let's say that you track 10 moose using GPS telemetry and record the geographic location of each moose daily over the course of a season. You are interested in knowing whether the daily movement distance (i.e., Euclidean distance between daily locations) varies among moose in relation to the intensity of human development in the neighborhood. Let's say that you have 100 observations per moose, representing a 100 day period. For our purposes, to keep it simple, let's say that you randomly draw 1 observation per moose to ensure independence among observations. Each observation represents a 24 hour period. The raw data are given here for each moose, including an index of development intensity (`dev.intensity`) in the neighborhood of the moose during the 24 hour observation period and the Euclidean distance (`dist.moved`) during the corresponding 24 hour period. Here are the data.

```
moose <- data.frame(dev.intensity = c(2.9,8.5,7.0,1.3,9.7,7.5,0.4,6.2,0.9,3.8),
                    dist.moved = c(456,141,47,1362,128,21,2123,189,899,38))
ggplot(moose, aes(dev.intensity, dist.moved))+
  geom_point()+
  geom_rug()
```



It seems pretty clear from looking at the plot that moose move around a lot less when there is a lot of development nearby.

A simple hypothesis test

We would like to test the hypothesis that moose movement is dependent on development intensity.

$$H_0 : \text{Moose movement and development intensity are independent.}$$
$$H_A : \text{Moose movement and development intensity are not independent.}$$

Now we have to translate these into something we can actually test, which means that we have to specify the form that we expect this dependence to take and then see if when confronted with our data, this proposed form seems like a plausible explanation. In other words, we need a model.

In fact we need two models, one for the null hypothesis and one for the alternate hypothesis. The model for the null hypothesis will have a constant for its deterministic part and the model for the alternate hypothesis will depend on development intensity. Both models will have a stochastic component. The null hypothesis is that this is all that matters.

The stochastic part of the models

From examining the data, it seems that moose always move around some, and sometimes quite a lot. But they never move a negative amount. It seems reasonable to hypothetically allow a moose to move zero. A distribution that has these characteristics is the *exponential* distribution. It has a single parameter λ called the *rate* which is equal to the reciprocal of the mean: $\lambda = 1/\mu$. The exponential distribution has the parameterization

$$P(Y) = \lambda e^{-\lambda Y}$$

and the R commands `dexp`, `rexp` etc. Unfortunately it isn't one of the distributions that can be accessed with `family` in `glm`. So to use this distribution we will have to proceed "by hand."

The deterministic part of the models

As discussed above, the deterministic part of the null model is constant. It is the mean of `dist.moved`.

For the model to be used for the alternate hypothesis, we have infinitely many options, and it is up to us to choose something that is realistic enough to be interesting but not overly complicated. In considering how development intensity influences moose movement, we might decide to try a power law model. A power law model $y = ax^b$ with a negative exponent ($b < 0$) describes a scenario where movement and development intensity are always positive, and the larger the value of one of these two variables, the smaller the value of the other.

Since the rate parameter for the exponential distribution is the reciprocal of the mean, we are proposing that `dist.moved` is distributed as

$$\texttt{dist.moved} \sim \text{Exp}\left(\text{rate} = \frac{1}{a \cdot \texttt{dev.intensity}^b}\right)$$

for some values of a and b.

Phrasing the test in terms of the model

With the above proposed model, our hypothesis test can be expressed in terms of b.

$$H_0 : b = 0, \quad H_A : b \neq 0$$

Building the null model

Even if $b = 0$, we still need to find a. We can use maximum likelihood.

(1) Find a formula in terms of a to compute the negative log-likelihood (nll) of the data `dist.moved`, assuming an exponential distribution with a rate of $1/a$.

(2) Use this formula to compute the nll for 10 values of a approximately in the range 100 to 1000. Make a plot showing these values and their negative log likelihoods. What value of a appears to give the minimum nll, or equivalently the maximum likelihood?

(3) Use the nll function that you have defined and the `mle2` function to find the null model `mod0` with maximum likelihood for your data.

Building the alternative model

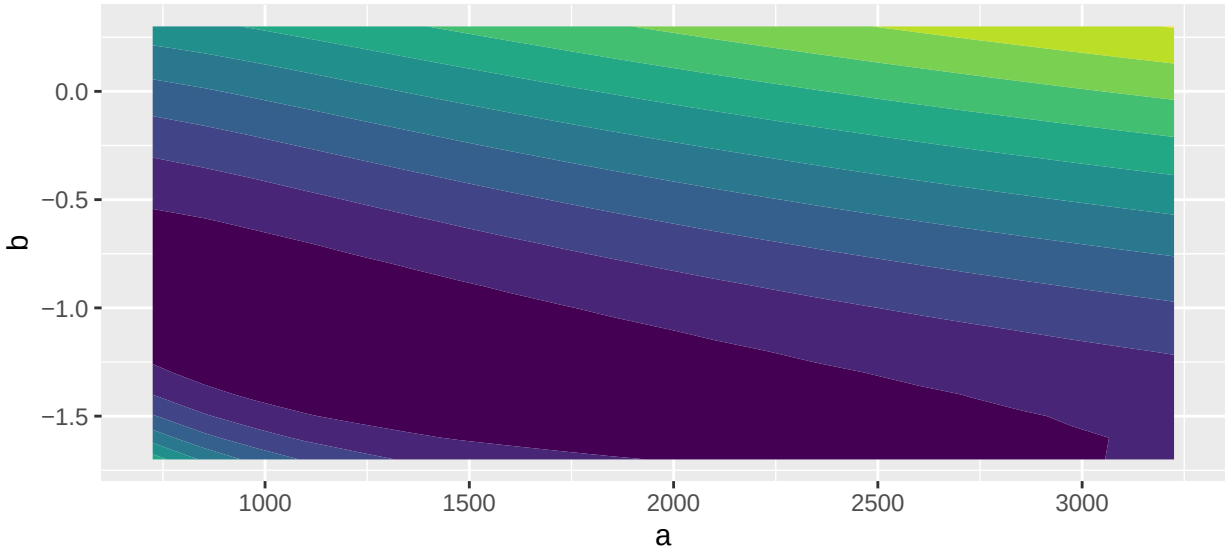
Now we'll find the maximum likelihood estimate for b.

(4) Find a formula in terms of a and b to compute the negative log-likelihood (nll) of the data `dist.moved`, assuming an exponential distribution with a rate of $1/a \cdot$ `dev.intensity`^{b}.

(5) Use this formula to compute the nll for the value of a that you found above and 10 values of b in the range -0.5 to -1.5. Make a plot showing these values and their negative log likelihoods. What value of b appears to give the minimum nll, or equivalently the maximum likelihood?

To actually find the maximum likelihood, we have to let both a and b vary. To visualize this is a bit more complicated, but the following defines a function `plotnll` that plots a negative log likelihood for this model.

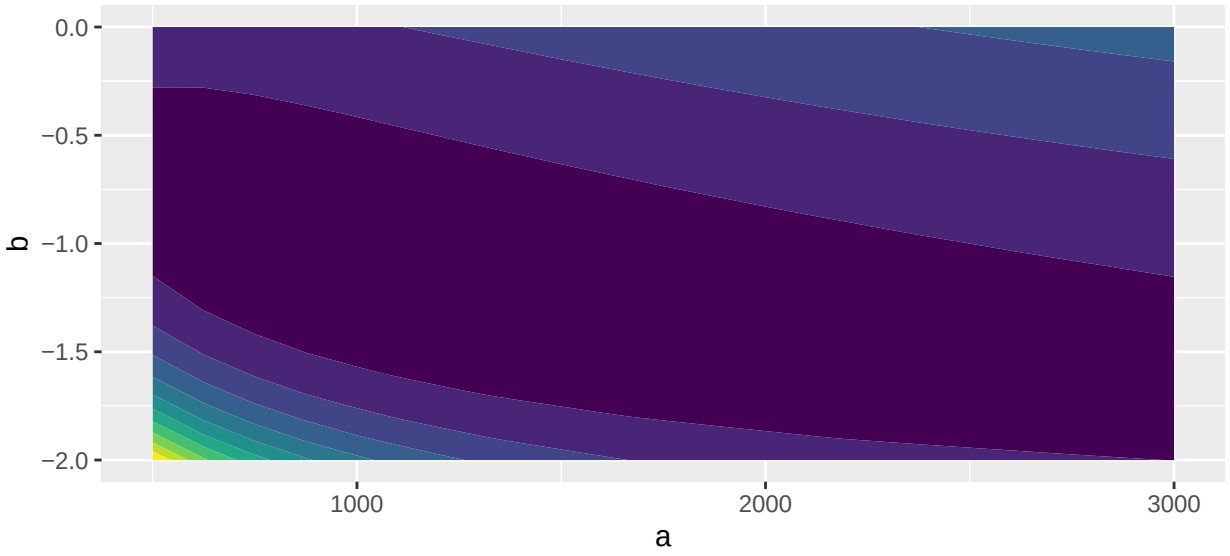
```
nll1 <- function(a,b) -sum(log(dexp(moose$dist.moved,rate = 1/(a*moose$dev.intensity^b) )))
plotnll <- function(nllfunc=nll1,
                    amin=725, bmin= -1.7,
                    astep=125, bstep=0.1,
                    gridsize=20 ){
  params <- expand.grid(a = amin+astep*(0:gridsize),
                        b = bmin+bstep*(0:gridsize))
  params$lik <- mapply(nllfunc,a = params$a, b = params$b)
  ggplot(params, aes(a,b,z=lik))+
    geom_contour_filled()+theme(legend.position="none")
}
plotnll()
```



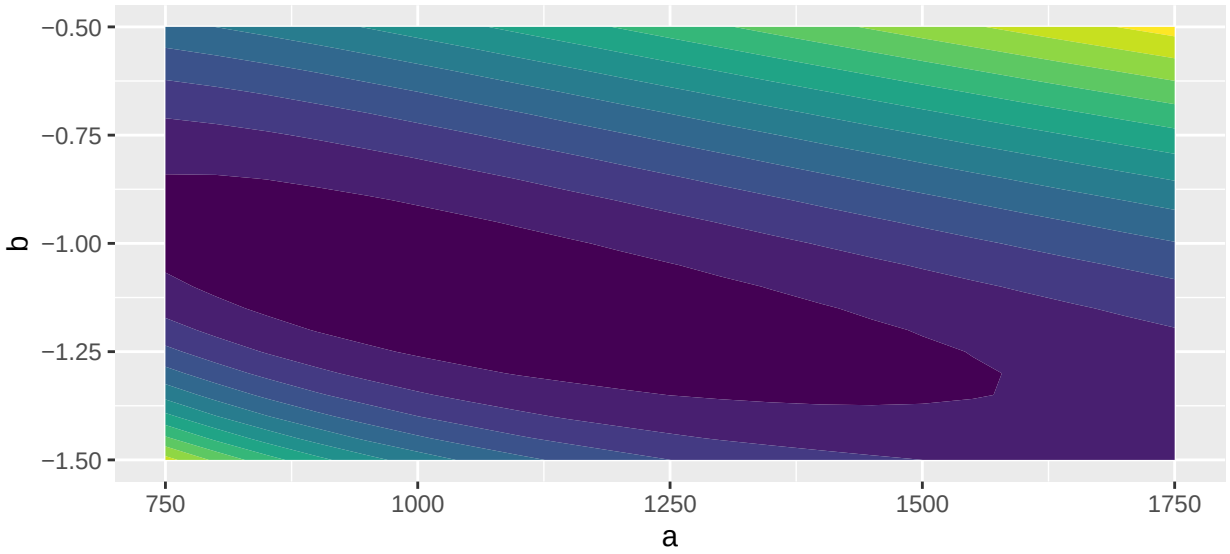
Here the default settings are plotted, which might be a reasonable starting guess about what range of parameters to guess based on the one variable plots.

To demonstrate the use of the function, we can shift and scale our window to make a good starting guess.

```
plotnll(bmin=-2, amin=500)
```




```
plotnll(bmin=-1.5, bstep=0.05, amin=750, astep = 50)
```



(6) Based on plotting the negative log likelihood, make a guess for values of a and b to start searching for the MLE model.

(7) Use the nll function that you have defined with your guess from making plots above and the `mle2` function to find the alternative model `mod1` with maximum likelihood for the data.

Plotting the MLE alternative model.

When we are in a situation like the current one, where there is one predictor variable, it is possible to make a plot showing the data, the deterministic part of a model, and a confidence interval based on the model.

The following code illustrates one way to do this.

```
mod1func <- function(x) coef(mod1)["a"]*x^coef(mod1)["b"]

ggplot(moose, aes(dev.intensity, dist.moved))+
  geom_point()+
  geom_rug()+
  geom_function(fun=mod1func)+
  geom_function(fun=function(x) qexp(0.025, rate = 1/mod1_func(x)), linetype=2)+
  geom_function(fun=function(x) qexp(0.975, rate = 1/mod1_func(x)), linetype=2)+
  lims(y=c(0,2500))
```

(8) Make a plot of `mod1` and the data.

Testing the null and alternative models

(9) The hypothesis test can be carried out using a likelihood ratio test on the two models you have built. The `bbmle` library provides an `anova` method for `mle2` and the default test is a likelihood ratio test: `anova(mod0, mod1)`. If you want to do a likelihood ratio test on models built with other tools, such as `glm`, specify `test="LRT"`.

(10) AIC can also be used to compare the null and alternate model. Recall that $AIC = -2L + 2k$ where L is the log likelihood and k is the number of predictors. Compute the AIC of `mod0` and `mod1` using the formula. This does not give a p-value, but a model with smaller AIC better represents the data.

(11) The `AICtab` command will create a table of AIC values for a list of models: `AICtab(mod0, mod1)`. Use this command to check that you did the calculations above correctly.