POLI 8501
Introduction to Maximum Likelihood Estimation

Maximum Likelihood

Intuition

Consider a model that looks like this:

$$Y_i \sim N(\mu, \sigma^2)$$

So:

$$\begin{aligned} E(Y) = & \quad \mu \\ Var(Y) = & \quad \sigma^2 \end{aligned}$$

Suppose you have some data on Y, and you want to estimate μ and σ^2 from those data...

The whole idea *likelihood* is the find the estimate of the parameter(s) that maximizes the probability of the data.

Example: Suppose Y is income of assistant professors (in thousands of dollars), and we have a random sample of five data points:

$$\begin{aligned} Y = & \quad 54 \\ & \quad 53 \\ & \quad 49 \\ & \quad 61 \\ & \quad 58 \end{aligned}$$

Intuitively, how likely is it that these five data points were drawn from a normal distribution with $\mu = 100$? (Answer: Not very likely)

What about $\mu = 55$ (which happens to be the empirical mean of this sample)? (Hint: More likely)

What maximum likelihood is, is a systematic way of doing exactly this.

Think of the salaries as draws from a normal distribution.

We can write:

$$\Pr(Y_i = y_i) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(Y_i - \mu)^2}{2\sigma^2}\right] \tag{1}$$

This is the density, or probability density function (PDF) of the variable Y.

- The probability that, for any one observation i, Y will take on the particular value y.
- This is a function of μ, the expected value of the distribution, and σ^2, the variability of the distribution around that mean.

We can think of the probability of a single realization being what it is, e.g.:

$$\begin{aligned}\Pr(Y_1 = 54) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(54 - \mu)^2}{2\sigma^2}\right]\\ \Pr(Y_2 = 53) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(53 - \mu)^2}{2\sigma^2}\right]\end{aligned}$$

etc.

Now, we're interested in getting estimates of the parameters μ and σ^2, based on the data...

If we assume that the observations on Y_i are independent (i.e. not related to one another), then we can consider the joint probability of the observations as simply the product of the marginals.

Recall that, for independent events A and B:

$$\Pr(A, B) = \Pr(A) \times \Pr(B)$$

So:

$$\Pr(Y_1 = 54, Y_2 = 53) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(54 - \mu)^2}{2\sigma^2}\right] \times \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(53 - \mu)^2}{2\sigma^2}\right]$$

More generally, for N independent observations, we can write the joint probability of each realization of Y_i as the product of the N marginal probabilities:

$$\Pr(Y_i = y_i \forall i) \equiv L(Y|\mu, \sigma^2) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(y_i - \mu)^2}{2\sigma^2}\right] \tag{2}$$

This product is generally known as the *Likelihood* $[L(Y)]$, and is the probability that each observation is what it is, given the parameters.

Estimation

Of course, we don't know the parameters; in fact, they're what we want to figure out. That is, we want to know the likelihood of some values of μ and σ^2, given Y.

This turns out to be proportional to $L(Y|\mu, \sigma^2)$:

$$L(\hat{\mu}, \hat{\sigma}^2|Y) \propto \Pr(Y|\hat{\mu}, \hat{\sigma}^2)$$

We can get at this by treating the likelihood as a function (which it is). The basic idea is *to find the values of μ and σ^2 that maximize the function; i.e., those which have the greatest likelihood of having generated the data.*

How do we do this?

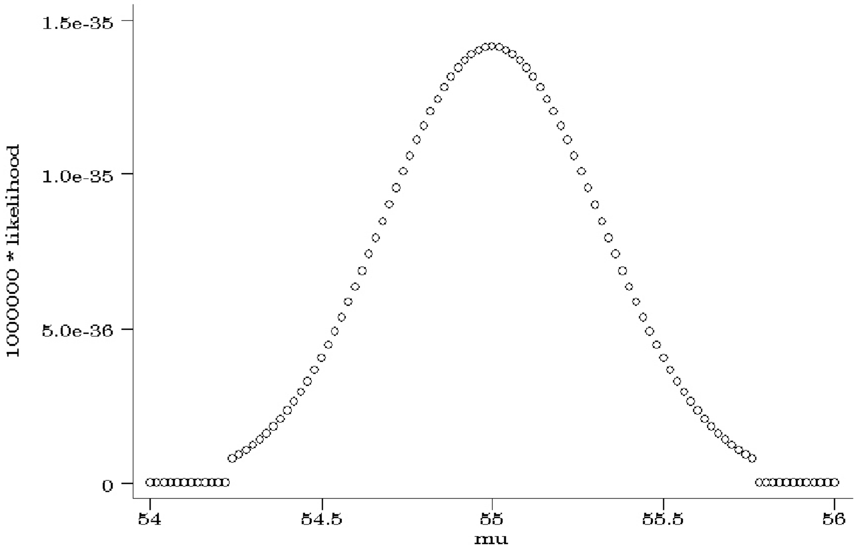
One way would be to start plugging in values for μ and σ^2 and seeing what the corresponding likelihood was...

E.g.: For $\hat{\mu} = 58$ and $\hat{\sigma}^2 = 1$, we would get:

$$\begin{aligned}
L &= \frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(54-58)^2}{2}\right] \times \\
&\quad \frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(53-58)^2}{2}\right] \times \\
&\quad \frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(49-58)^2}{2}\right] \times \\
&\quad \frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(61-58)^2}{2}\right] \times \\
&\quad \frac{1}{\sqrt{2\pi}}\exp\left[-\frac{(58-58)^2}{2}\right] \\
&= 0.0001338 \times 0.0000014 \times \ldots \\
&= \text{some reeeeeally small number...}
\end{aligned}$$

More generally, we can graph the likelihood for these five observations (assuming, for the moment, that $\sigma^2 = 1$)...

Figure 1: Likelihood of $\hat{\mu}$ for five sample data points



Note that the likelihood is maximized at $\mu = 55$, the empirical mean...

But...

- ...likelihood functions often look scary,
- ...products are generally hard to deal with, and
- ...we run into issues with numeric precision when we get into teeeeny probabilities...

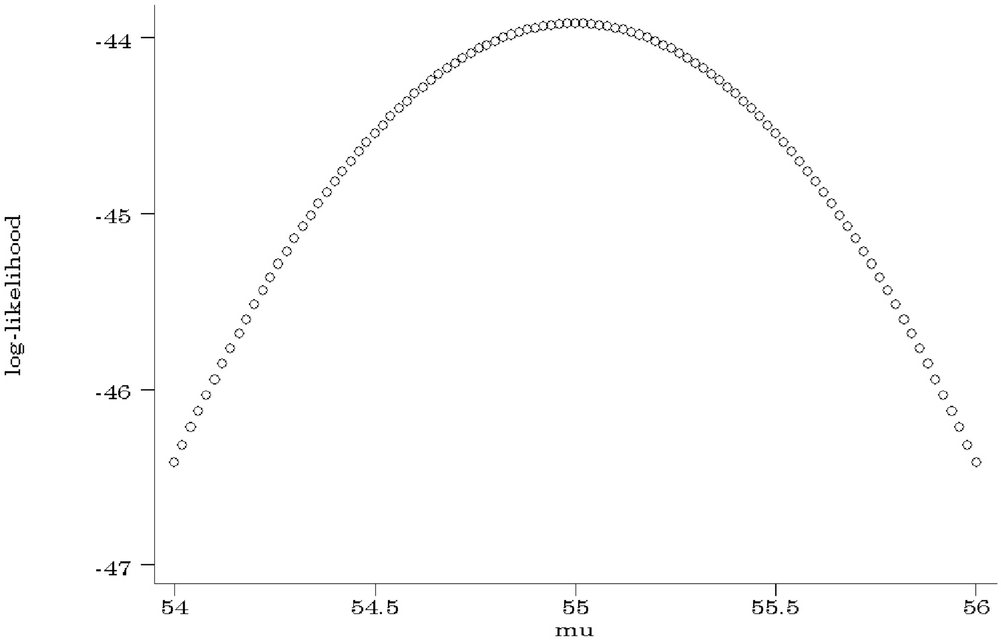
Fortunately, it turns out that if we find the values of the parameters that maximize any monotonic transformation of the likelihood function, those are also the parameter values that maximize the function itself.

Most often we take natural logs, giving something called the *log-likelihood*:

$$
\begin{aligned}
lnL(\hat{\mu}, \hat{\sigma}^2|Y) &= ln \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(Y_i-\mu)^2}{2\sigma^2}\right] \\
&= \sum_{i=1}^{N} ln \left\{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(Y_i-\mu)^2}{2\sigma^2}\right]\right\} \\
&= -\frac{N}{2} ln(2\pi) - \left[\sum_{i=1}^{N} \frac{1}{2} ln\,\sigma^2 - \frac{1}{2\sigma^2}(Y_i-\mu)^2\right] \qquad (3)
\end{aligned}
$$

If we again fix $\sigma^2 = 1$ and consider this *log-likelihood* the same way we did above, we get this figure...

Figure 2: Log-likelihood of $\hat{\mu}$ for five sample data points



Maximization

Question: *Graphs are nice, but how do we normally find a maximum?*

Answer: Good old **differential calculus**...

What happens when we take the first derivatives of (3) with respect to μ and σ^2?

$$
\begin{aligned}
\frac{\partial lnL}{\partial \mu} &= \frac{1}{\sigma^2}\sum_{i=1}^{N}(Y_i - \mu) \\
\frac{\partial lnL}{\partial \sigma^2} &= \frac{-N}{2\sigma^2} + \frac{1}{2}\sigma^4\sum_{i=1}^{N}(Y_i - \mu)^2
\end{aligned}
$$

If we set these equal to zero and solve for the two unknowns, we get:

$$
\begin{aligned}
\hat{\mu} &= \frac{1}{N}\sum_{i=1}^{N} Y_i \\
\hat{\sigma^2} &= \frac{1}{N}\sum_{i=1}^{N}(Y_i - \bar{Y})^2
\end{aligned}
$$

...which are the basic formulas for mean and variance. That is, our standard versions of $\hat{\mu}$ and $\hat{\sigma^2}$ are the *maximum likelihood estimators* for μ and σ^2.

MLE and the Linear Regression Model

Now suppose we want to set the mean of Y to be a function of some other variable X; that is, you suspect that professors' salaries are a function of, e.g., gender, or something. We write:

$$
\begin{aligned}
\mathrm{E}(Y) \equiv \mu &= \beta_0 + \beta_1 X_i \\
\mathrm{Var}(Y) &= \sigma^2
\end{aligned}
$$

We can then just substitute this equation in for the systematic mean part (μ) in the previous equations...

E.g.:

$$
L(\beta_0, \beta_1, \sigma^2 | Y) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(Y_i - \beta_0 - \beta_1 X_i)^2}{2\sigma^2}\right] \tag{4}
$$

and:

$$\begin{aligned} lnL(\beta_0, \beta_1, \sigma^2|Y) &= ln \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \text{exp}\left[-\frac{(Y_i - \beta_0 - \beta_1 X_i)^2}{2\sigma^2}\right] \\ &= -\frac{N}{2} ln(2\pi) - \sum_{i=1}^{N} \left[\frac{1}{2} ln\sigma^2 - \frac{1}{2\sigma^2}(Y_i - \beta_0 - \beta_1 X_i)^2\right] \end{aligned} \tag{5}$$

With respect to the parameters $\{\beta_0, \beta_1, \sigma^2\}$, only the last term is important...

- The first one $(-\frac{N}{2} ln(2\pi))$ is invariant with respect to the parameters of interest, and so can be dropped.
- This is due to something called the *Fisher-Neyman Factorization Lemma.*

Thus, the *kernel* of the log-likelihood is:

$$-\sum_{i=1}^{N} \left[\frac{1}{2} ln\sigma^2 - \frac{1}{2\sigma^2}(Y_i - \beta_0 - \beta_1 X_i)^2\right] \tag{6}$$

...which is the old familiar sums-of-squared-residuals term, scaled by the variance parameter σ^2.

This leads us to several interesting things:

- The least-squares estimator of the OLS βs *is the maximum likelihood estimator as well.*
- MLE is not confined to models with "ugly" dependent variables (though it *is* highly useful for them).

For your homework for this week, you're going to do exactly this: estimate linear regression model using MLE...

MLE and Statistical Inference

MLE is a very flexible way of getting point *estimates* of parameter(s) (say, θ) under a broad range of conditions... It is also a useful means of *statistical inference*, since, under certain *regularity conditions*, the estimates (and the distribution(s)[1]) of the MLEs will have nice properties...

These nice properties include:

Invariance to Reparameterization

- This means that, rather than estimating a parameter θ, we can instead estimate some function of it $g(\theta)$.
- We can then recover an estimate of θ (that is, $\hat{\theta}$) from $g(\theta)$.
- This can be very useful, as we'll see soon enough...

Invariance to Sampling Plans

This means that ML estimators are the same irrespective of the rule used for determining sample size. This seems like a minor thing, but in fact its very useful, since it means that you can e.g. "pool" data and get "good" estimates without regard for how the sample size was chosen.

Minimum Variance Unbiased Estimates

- If there is a minimum-variance unbiased estimator, MLE will "choose"/"find" it.
- E.g., the CLRM/OLS/MLE example above.

Consistency

This is a **big** one...

- MLEs are generally *not* unbiased (that is, $\mathrm{E}(\hat{\theta}_{MLE}) \neq \theta$). But...
- They *are* consistent ($\mathrm{E}(\hat{\theta}_{MLE}) \underset{a.s.}{\rightarrow} \theta$)
- This means that MLEs are *large-sample estimators*
    - They are "better" with larger Ns
    - How large? Well, that depends...

[1] Remember that estimates are, themselves, random variables; we use our understanding of those variables' distributions to construct measures of uncertainty about our estimates (e.g. standard errors, confidence intervals, etc.).

Asymptotic Normality

- Formally,
$$\sqrt{N}(\hat{\theta}_{MLE} - \theta) \sim N(\mathbf{0}, \mathbf{\Omega})$$
- That is, the asymptotic distribution of the MLEs is standard multivariate normal.
- Is a result of the application of the *central limit theorem.*
- This is true regardless of the distribution assumed in the model itself.
- Allows us to do hypothesis testing, confidence intervals, etc. very easily...

Asymptotic Efficiency

- As we'll see in a few minutes, the variance of the MLE can be estimated by taking the inverse of the "information matrix" (aka, the "Hessian"), which is the matrix of second derivatives of the (log-)likelihood with respect to the parameters:
$$I_\theta = \mathrm{E}\left[\frac{\partial^2 lnL}{\partial^2 \theta}\right]$$
- Among all consistent, asymptotically Normal estimators, the MLE has the smallest asymptotic variance (again, think CLRM here)...
- As a result, "more of the ML estimates that could result across hypothetical experiments are concentrated around the true parameter value than is the case for any other estimator in this class" (King 1998, 80).

All these things make MLE a very attractive means of estimating parameters...

MLE: How do we do it?

Assume we have a $k \times 1$ vector of regression coefficients $\boldsymbol{\beta}$ that is associated with a set of k independent variables $\mathbf{X}$. The goal of MLE is to find the values of the $\boldsymbol{\beta}$s that maximize the (log)likelihood...

How do we do this?

- One way is to "plug in" values for (e.g.) β_0, β_1, etc. until we find the ones that maximize the likelihood...
  - This is very time-consuming, and dull...
  - It is also too complicated for higher-order problems...

- Since the likelihood is a function, we can do what we did for the linear regression model: take the first derivative w.r.t. the parameters, set it to zero, and solve...

But...

- Some functions we're interested in aren't linear...
- E.g., for a logit model, when we take the derivative w.r.t. $\boldsymbol{\beta}$ and set it equal to zero, we get:

$$\frac{\partial lnL}{\partial \beta} = \sum_{i=1}^{N} \mathbf{X}_i \mathbf{Y}_i - \sum_{i=1}^{N} \frac{\exp(\mathbf{X}_i \boldsymbol{\beta})}{1 + \exp(\mathbf{X}_i \boldsymbol{\beta})}$$

- As it turns out, these equations are *nonlinear* in the parameters...
- That is, we can't just take the derivative, set it equal to zero, and solve for the two equations in two unknowns in order to get the maximum.
- Instead, we have to resort to numerical maximization methods...

Numerical Optimization Methods

First, some truth in advertising: The numerical approaches I discuss here are *only one way of getting parameter estimates that are MLEs.* Others include Bayesaian approaches (e.g., Markov-Chain Monte Carlo methods), derivative-free methods (such as simulated annealing), and many others. A discussion of the relative merits of these approaches is a bit beyond what I care to go into here. To the extent that most, if not all, widely-used software programs use the numerical methods that follow, knowing something about what they are and how they work is highly recommended.

The Intuition

- Start with some guess of $\hat{\boldsymbol{\beta}}$ (call this $\hat{\boldsymbol{\beta}}_{\mathbf{0}}$),
- Then adjust that guess depending on what value of the (log-)likelihood it gives us.
- If we call this adjustment $\mathbf{A}$, we get:

$$\hat{\boldsymbol{\beta}}_{\mathbf{1}} = \hat{\boldsymbol{\beta}}_{\mathbf{0}} + \mathbf{A}_{\mathbf{0}}$$

and, more generally,

$$\hat{\boldsymbol{\beta}}_{\ell} = \hat{\boldsymbol{\beta}}_{\boldsymbol{\ell-1}} + \mathbf{A}_{\boldsymbol{\ell-1}} \qquad (6)$$

We want to move the vector $\hat{\boldsymbol{\beta}}$ to the point at which the value of the likelihood is at its highest...

- This means we need to take into account the slope of the likelihood function at each point...
- Intuitively, we do this by incorporating information from the *gradient matrix* (the matrix of first derivatives of the (log-)likelihood w.r.t. the $\boldsymbol{\beta}$s)
  - ○ If the gradient is positive, then:
    - · the likelihood is increasing in $\hat{\boldsymbol{\beta}}$, and so
    - · we should increase our estimate even more...
  - ○ If the gradient is negative, then:
    - · the likelihood is decreasing in $\hat{\boldsymbol{\beta}}$, and so
    - · we should decrease our estimate of $\hat{\boldsymbol{\beta}}$.

In this way, we "climb to the top" of the likelihood function...

- Once we get near the top, the gradient gets very close to zero (b/c the likelihood is near its maximum).
- At this point, when the changes get small enough from one iteration to the next, we simply stop, and evaluate our estimates.

The Gradient Matrix and "Steepest Ascent"

How do we use the information from the gradient matrix?

- Again, start with some guess of $\hat{\boldsymbol{\beta}}$ (call this $\hat{\boldsymbol{\beta}}_{\mathbf{0}}$).
- Then adjust that guess depending on what value of the (log-)likelihood it gives us.

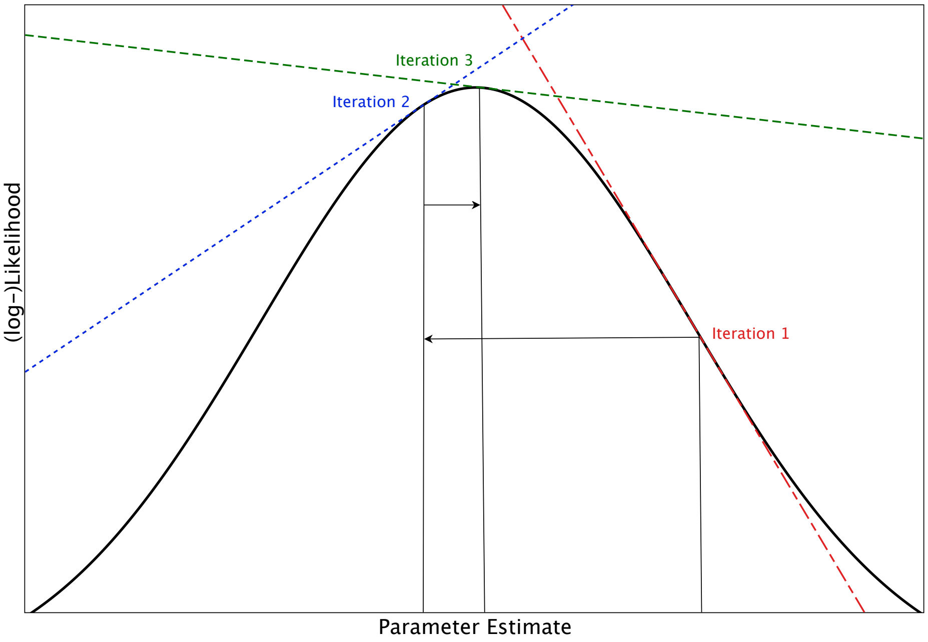
The simplest way to update the parameter estimates is to specify $\mathbf{A_k} = \frac{\partial lnL}{\partial \boldsymbol{\beta}_{\mathbf{k}}}$.

- This yields:

$$\hat{\boldsymbol{\beta}}_{\ell} = \hat{\boldsymbol{\beta}}_{\boldsymbol{\ell-1}} + \frac{\partial \mathbf{lnL}}{\partial \boldsymbol{\beta}_{\boldsymbol{\ell-1}}} \tag{7}$$

- That is, we adjust the parameter by a factor equal to the first derivative of the function w.r.t. the parameters.
- Review Q: What does a first derivative do? (A: Tells us the slope of the function at that point.)

Figure 3: MLE: Graphical Intuition



- So:
  - If the slope of the function is positive, then the likelihood is increasing in $\hat{\boldsymbol{\beta}}$, and we should increase $\hat{\boldsymbol{\beta}}$ even more the next time around...
  - If the derivative is negative, then the likelihood is decreasing in $\hat{\boldsymbol{\beta}}$, and we should "back up" (make $\hat{\boldsymbol{\beta}}$ smaller) to increase the likelihood.
  - This general approach is called the *method of steepest ascent* (sometimes also called steepest descent).

Using the Second Derivatives

Steepest ascent seems attractive, but has a problem... the method doesn't consider *how fast the slope is changing.* You can think of the matrix $\frac{\partial lnL}{\partial \boldsymbol{\beta}_k}$ as the *direction* matrix – it tells us which direction to go in to reach the maximum. We could generalize (6), so that we changed our estimates not only in terms of their "direction", but also by a factor determining how far we changed the estimate (Greene calls this the "step size"):

$$\hat{\boldsymbol{\beta}}_{\mathbf{k}} = \hat{\boldsymbol{\beta}}_{\mathbf{k-1}} + \lambda_{\mathbf{k-1}} \boldsymbol{\Delta}_{\mathbf{k-1}} \tag{8}$$

Here, we've decomposed $\mathbf{A}$ into two parts:

- $\boldsymbol{\Delta}$ tells us the *direction* we want to take a step in, while

- λ tells us the *amount* by which we want to change our estimates (the "length" of the step size).

To get at this, we need to figure out a way of determining *how fast the slope of the likelihood is changing at that value of* $\hat{\boldsymbol{\beta}}$*...*

First, let's consider the usual approach to maximization...

- Once we've solved for the maximum (first derivative), we use the second derivative to determine if the value we get is a minimum or maximum...
- We do this, because the 2nd derivative tells us the "slope of the line tangent to the first derivative;" i.e., the *direction in which the slope of the function is changing.*
- So, if the second derivative is negative, then the slope of the first derivative is becoming less positive in the variable, indicating a maximum.
- It also tells us the *rate* at which that slope is changing:
    - E.g., if its BIG, then the slope is increasing or decreasing quickly.
    - This, in turn, tells us that the slope of the function at that point is very steep.

Now think about our likelihood example of the mean again, but compare it to another function with the same maximum...

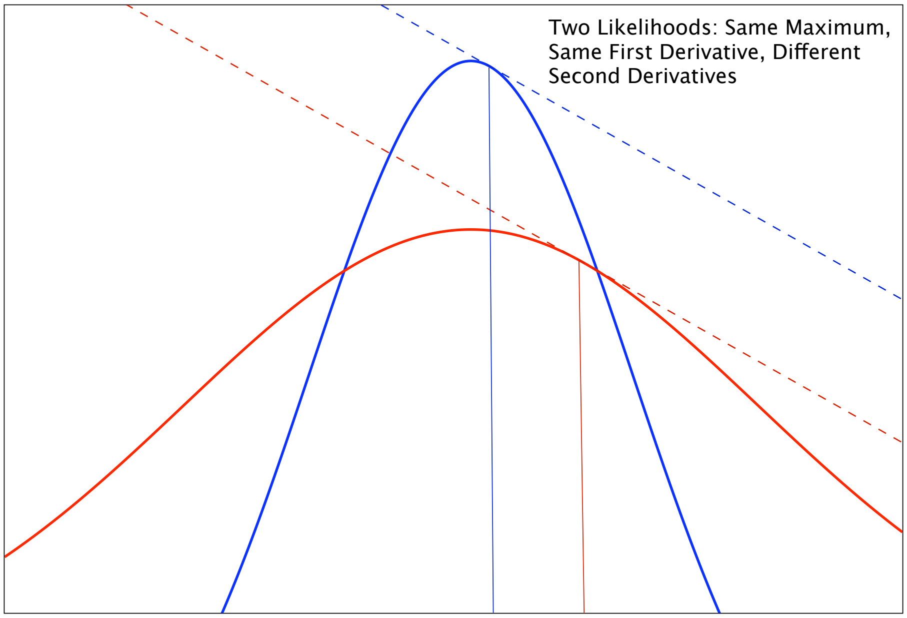
- For red line, the function is flatter; the first derivative is negative, while the second will be quite small.
- For the blue line, the second derivative will be quite a bit larger, because the value of the likelihood is changing rapidly depending on the values of $\hat{\boldsymbol{\beta}}$.

This shows that the second derivative illustrates the speed with which the slope of the function is changing.

In likelihood terms, we want to somehow incorporate this second derivative into the maximization routine:

- If the function is relatively "steep", we don't want to change the parameters very much from one iteration to the next.
- OTOH, if it is flat, we can adjust the coefficients more from one step to the next.

Figure 4: Two Likelihood Functions, with Differing Rates of Change



To incorporate this, we can use three possibilities:

- The *Hessian*:

$$\frac{\partial^2 lnL}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'} \tag{9}$$

  - This is the second derivative of the likelihood function with respect to the parameters.
  - This $k \times k$ matrix contains the second derivatives along the main diagonal, and the cross-partials of the elements of $\hat{\boldsymbol{\beta}}$ in the off-diagonal elements.

In some cases, figuring out this second derivative can be really hard, to the point of being impracticable. Two alternatives to this are:

- The *information matrix*:

$$-\operatorname{E}\left[\frac{\partial^2 lnL}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'}\right] \tag{10}$$

  - This is the negative of the expected value of the Hessian.

  - It can be easier to compute than the Hessian itself, in some cases.
  - Yields a similar kind of result (tells how quickly the slope is changing...)

- If even the information matrix is too hard to compute, we can also use the *outer product approximation* to the information matrix:

$$\sum_{i=1}^{N} \frac{\partial lnL_i}{\partial \boldsymbol{\beta}} \frac{\partial lnL_i{}'}{\partial \boldsymbol{\beta}} \tag{11}$$

  - That is, we sum over the "squares" (outer products) of the first derivatives of the log-likelihoods.
  - This is nice because we don't have to deal with the second derivatives at all; we have only to evaluate the gradient.
  - This can be really useful when we have very complex likelihoods

Optimization Methods

Each of these options for incorporating the "rate of change" has an associated maximization algorithm associated with it...

1. **Newton-Raphson** uses the (inverse of the) Hessian:

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-1} - \left[\left(\frac{\partial^2 lnL}{\partial \hat{\boldsymbol{\beta}}_{\ell-1} \partial \hat{\boldsymbol{\beta}}'_{\ell-1}} \right)^{-1} \frac{\partial lnL}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \right] \tag{12}$$

   - I.e., the new parameter estimates are equal to the old ones, adjusted in the *direction* of the first derivative (a la steepest descent), BUT
   - The *amount* of change is inversely related to the size of the second derivative.
     - If the second derivative is big, the function is steep, and we don't want to change very much.
     - The opposite is true if the second derivative is small and the slope is relatively flat.

2. **Method of Scoring** uses the (inverse of the) information matrix:

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-1} - \left[\mathrm{E} \left(\frac{\partial^2 lnL}{\partial \hat{\boldsymbol{\beta}}_{\ell-1} \partial \hat{\boldsymbol{\beta}}'_{\ell-1}} \right) \right]^{-1} \frac{\partial lnL}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \tag{13}$$

3. **Berndt, Hall, Hall and Hausman (BHHH)** uses the inverse of the outer product approximation to the information matrix:

$$\hat{\boldsymbol{\beta}}_\ell = \hat{\boldsymbol{\beta}}_{\ell-1} - \left(\sum_{i=1}^{N} \frac{\partial lnL_i}{\partial \boldsymbol{\beta}_{\ell-1}} \frac{\partial lnL_i}{\partial \boldsymbol{\beta}_{\ell-1}}' \right)^{-1} \frac{\partial lnL}{\partial \hat{\boldsymbol{\beta}}_{\ell-1}} \tag{14}$$

There are others...

- E.g., the "Davidson-Fletcher-Powell" (DFP) algorithm, in which the "step length" is chosen in such a way as to always make the updated matrix of parameter estimates positive definite.
- See Judge et al. (Appendix B) or Greene (2003, Appendix E.6) for more discussion...

MLE, Uncertainty and Inference

Since the matrix of second derivatives tells us the rate at which the slope of the function is changing (i.e., its "steepness" or "flatness" at a given point), it makes sense that we can use this information to determine the variance ("dispersion") of our estimate...

- If the likelihood is very steep,
  - We can be quite sure that the maximum we've reached is the "true" maximum.
  - I.e., the variance of our estimate of $\hat{\boldsymbol{\beta}}$ will be small.
- If the likelihood is relatively "flat," then
  - We can't be as sure of our "maximum," and...
  - so the variance around our estimate will be larger.

Maximum likelihood theory tells us that, asymptotically, the variance-covariance matrix of our estimated parameters is equal to the inverse of the negative of the information matrix:

$$\text{Var}(\hat{\boldsymbol{\beta}}) = -\left[\text{E}\left(\frac{\partial^2 lnL}{\partial \hat{\boldsymbol{\beta}} \partial \hat{\boldsymbol{\beta}}'} \right) \right]^{-1} \tag{15}$$

As above, we can, if need be, substitute the outer product approximation for this...

Typically, whatever "slope" (i.e., second derivative) matrix a method uses is also used for estimating the variance of the estimated parameters:

Method	"Step size" (∂^2) matrix	Variance Estimate
Newton	Inverse of the observed second derivative (Hessian)	Inverse of the negative Hessian
Scoring	Inverse of the expected value of the Hessian (information matrix)	Inverse of the negative information matrix
BHHH	Outer product approximation of the information matrix	Inverse of the outer product approximation

There are a few other alternatives as well (e.g., "robust" / "Huber-White" standard errors); we'll talk more about those a bit later on.

General comments on Numerical Optimization

- Newton works very well and quickly for simple functions with global maxima.
- Scoring and BHHH can be better alternatives when the likelihood is complex or the data are ill-conditioned (e.g. lots of collinearity).

Software issues

- Stata uses a modified version of Newton ("Quasi-Newton") for the `-ml-` routine that makes up most of its "canned" routines.
  - Requires that it have the gradient and Hessian at each iteration...
  - Uses numeric derivatives when analytic ones aren't available.
  - This can be slow, since at actually calculates the inverse Hessian at each step, BUT
  - ...it also tends to be very reliable (more so than some other packages).
  - Stata *doesn't* allow for scoring or BHHH options.
- S-Plus / R have `-glm-` routines which default to the method of scoring (or IRLS – more on that later).
- In `LIMDEP` (and a few others, like `GAUSS`) you can specify the maximization algorithm.
- Some programs (e.g. `LIMDEP`) also allow you to start out with steepest ascent, to get "close" to a maximum, then switch to Newton or BHHH for the final iterations.

Practical Tips

- One can occasionally converge to a local minimum or maximum, or to a *saddle point*.
- Estimates may not converge at all...
- If convergence is difficult, try the following:
  - Make sure that the model is properly- (or, at least, well-)*specified*.
  - Delete all *missing data* explicitly (esp. in `LIMDEP`).
  - *Rescale* the variables so that they're all more-or-less on the same scale.
  - If possible, try another optimization *algorithm*.