Part II

Estimation

8 Numerical Maximization

8.1 Motivation

Most estimation involves maximization of some function, such as the likelihood function, the simulated likelihood function, or squared moment conditions. This chapter describes numerical procedures that are used to maximize a likelihood function. Analogous procedures apply when maximizing other functions.

Knowing and being able to apply these procedures is critical in our new age of discrete choice modeling. In the past, researchers adapted their specifications to the few convenient models that were available. These models were included in commercially available estimation packages, so that the researcher could estimate the models without knowing the details of how the estimation was actually performed from a numerical perspective. The thrust of the wave of discrete choice methods is to free the researcher to specify models that are tailor-made to her situation and issues. Exercising this freedom means that the researcher will often find herself specifying a model that is not exactly the same as any in commercial software. The researcher will need to write special code for her special model.

The purpose of this chapter is to assist in this exercise. Though not usually taught in econometrics courses, the procedures for maximization are fairly straightforward and easy to implement. Once learned, the freedom they allow is invaluable.

8.2 Notation

The log-likelihood function takes the form $\text{LL}(\beta) = \sum_{n=1}^{N} \ln P_n(\beta)/N$, where $P_n(\beta)$ is the probability of the observed outcome for decision maker n, N is the sample size, and β is a $K \times 1$ vector of parameters. In this chapter, we divide the log-likelihood function by N, so that LL is the average log-likelihood in the sample. Doing so does not affect the location of the maximum (since N is fixed for a given sample) and yet

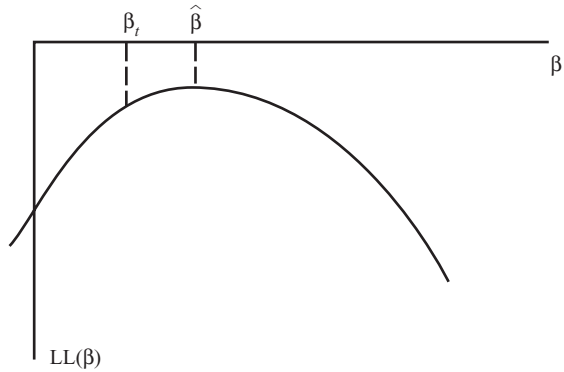

Figure 8.1. Maximum likelihood estimate.

facilitates interpretation of some of the procedures. All the procedures operate the same whether or not the log-likelihood is divided by N. The reader can verify this fact as we go along by observing that N cancels out of the relevant formulas.

The goal is to find the value of β that maximizes $\mathrm{LL}(\beta)$. In terms of Figure 8.1, the goal is to locate $\hat{\beta}$. Note in this figure that LL is always negative, since the likelihood is a probability between 0 and 1 and the log of any number between 0 and 1 is negative. Numerically, the maximum can be found by "walking up" the likelihood function until no further increase can be found. The researcher specifies starting values β_0. Each iteration, or step, moves to a new value of the parameters at which $\mathrm{LL}(\beta)$ is higher than at the previous value. Denote the current value of β as β_t, which is attained after t steps from the starting values. The question is: what is the best step we can take next, that is, what is the best value for β_{t+1}?

The gradient at β_t is the vector of first derivatives of $\mathrm{LL}(\beta)$ evaluated at β_t:

$$g_t = \left(\frac{\partial \mathrm{LL}(\beta)}{\partial \beta}\right)_{\beta_t}.$$

This vector tells us which way to step in order to go up the likelihood function. The Hessian is the matrix of second derivatives:

$$H_t = \left(\frac{\partial g_t}{\partial \beta'}\right)_{\beta_t} = \left(\frac{\partial^2 \mathrm{LL}(\beta)}{\partial \beta\, \partial \beta'}\right)_{\beta_t}.$$

The gradient has dimension $K \times 1$, and the Hessian is $K \times K$. As we will see, the Hessian can help us to know *how far* to step, given that the gradient tells us *in which direction* to step.

8.3 Algorithms

Of the numerous maximization algorithms that have been developed over the years, I next describe only the most prominent, with an emphasis on the pedagogical value of the procedures as well as their practical use. Readers who are induced to explore further will find the treatments by Judge *et al.* (1985, Appendix B) and Ruud (2000) rewarding.

8.3.1. Newton–Raphson

To determine the best value of β_{t+1}, take a second-order Taylor's approximation of $\text{LL}(\beta_{t+1})$ around $\text{LL}(\beta_t)$:

(8.1)

$$\text{LL}(\beta_{t+1}) = \text{LL}(\beta_t) + (\beta_{t+1} - \beta_t)' g_t + \tfrac{1}{2}(\beta_{t+1} - \beta_t)' H_t (\beta_{t+1} - \beta_t).$$

Now find the value of β_{t+1} that maximizes this approximation to $\text{LL}(\beta_{t+1})$:

$$\begin{aligned} \frac{\partial \text{LL}(\beta_{t+1})}{\partial \beta_{t+1}} = g_t + H_t(\beta_{t+1} - \beta_t) &= 0, \\ H_t(\beta_{t+1} - \beta_t) &= -g_t, \\ \beta_{t+1} - \beta_t &= -H_t^{-1} g_t, \\ \beta_{t+1} &= \beta_t + (-H_t^{-1}) g_t. \end{aligned}$$

The Newton–Raphson (NR) procedure uses this formula. The step from the current value of β to the new value is $(-H_t^{-1})g_t$, that is, the gradient vector premultiplied by the negative of the inverse of the Hessian.

This formula is intuitively meaningful. Consider $K = 1$, as illustrated in Figure 8.2. The slope of the log-likelihood function is g_t. The second derivative is the Hessian H_t, which is negative for this graph, since the curve is drawn to be concave. The negative of this negative Hessian is positive and represents the degree of curvature. That is, $-H_t$ is the positive curvature. Each step of β is the slope of the log-likelihood function divided by its curvature. If the slope is positive, β is raised as in the first panel, and if the slope if negative, β is lowered as in the second panel. The curvature determines how large a step is made. If the curvature is great, meaning that the slope changes quickly as in the first panel of Figure 8.3, then the maximum is likely to be close, and so a small step is taken.

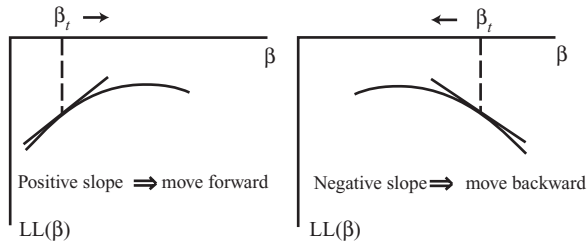

Figure 8.2. Direction of step follows the slope.

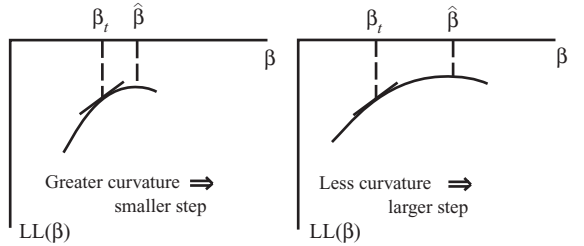

Figure 8.3. Step size is inversely related to curvature.

(Dividing the gradient by a large number gives a small number.) Conversely, if the curvature is small, meaning that the slope is not changing much, then the maximum seems to be further away and so a larger step is taken.

Three issues are relevant to the NR procedure.

Quadratics

If $\mathrm{LL}(\beta)$ were exactly quadratic in β, then the NR procedure would reach the maximum in one step from any starting value. This fact can easily be verified with $K = 1$. If $\mathrm{LL}(\beta)$ is quadratic, then it can be written as

$$\mathrm{LL}(\beta) = a + b\beta + c\beta^2.$$

The maximum is

$$\frac{\partial \mathrm{LL}(\beta)}{\partial \beta} = b + 2c\beta = 0,$$

$$\hat{\beta} = -\frac{b}{2c}.$$

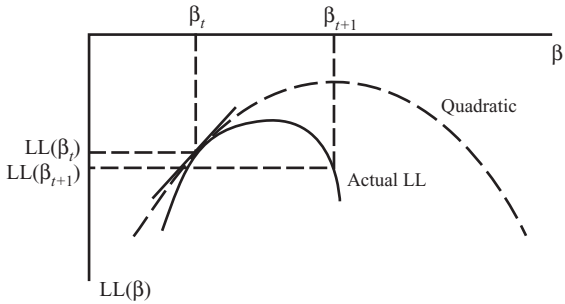

Figure 8.4. Step may go beyond maximum to lower LL.

The gradient and Hessian are $g_t = b + 2c\beta_t$ and $H_t = 2c$, and so NR gives us

$$\begin{aligned}\beta_{t+1} &= \beta_t - H_t^{-1} g_t \\ &= \beta_t - \frac{1}{2c}(b + 2c\beta_t) \\ &= \beta_t - \frac{b}{2c} - \beta_t \\ &= -\frac{b}{2c} = \hat{\beta}.\end{aligned}$$

Most log-likelihood functions are not quadratic, and so the NR procedure takes more than one step to reach the maximum. However, knowing how NR behaves in the quadratic case helps in understanding its behavior with nonquadratic LL, as we will see in the following discussion.

Step Size

It is possible for the NR procedure, as for other procedures discussed later, to step past the maximum and move to a lower LL(β). Figure 8.4 depicts the situation. The actual LL is given by the solid line. The dashed line is a quadratic function that has the slope and curvature that LL has at the point β_t. The NR procedure moves to the top of the quadratic, to β_{t+1}. However, LL(β_{t+1}) is lower than LL(β_t) in this case.

To allow for this possibility, the step is multiplied by a scalar λ in the NR formula:

$$\beta_{t+1} = \beta_t + \lambda(-H_t)^{-1} g_t.$$

The vector $(-H_t)^{-1} g_t$ is called the direction, and λ is called the step size. (This terminology is standard even though $(-H_t)^{-1} g_t$ contains step-size

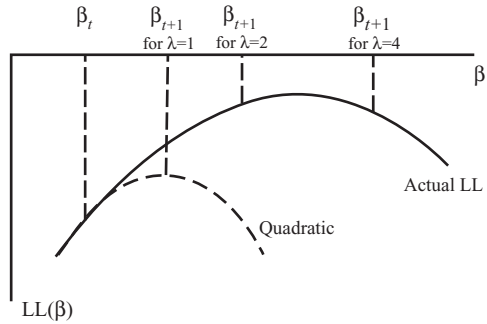

Figure 8.5. Double λ as long as LL rises.

information through H_t, as already explained in relation to Figure 8.3.) The step size λ is reduced to assure that each step of the NR procedure provides an increase in LL(β). The adjustment is performed separately in each iteration, as follows.

Start with $\lambda = 1$. If $\text{LL}(\beta_{t+1}) > \text{LL}(\beta_t)$, move to β_{t+1} and start a new iteration. If $\text{LL}(\beta_{t+1}) < \text{LL}(\beta_t)$, then set $\lambda = \frac{1}{2}$ and try again. If, with $\lambda = \frac{1}{2}$, $\text{LL}(\beta_{t+1})$ is still below $\text{LL}(\beta_t)$, then set $\lambda = \frac{1}{4}$ and try again. Continue this process until a λ is found for which $\text{LL}(\beta_{t+1}) > \text{LL}(\beta_t)$. If this process results in a tiny λ, then little progress is made in finding the maximum. This can be taken as a signal to the researcher that a different iteration procedure may be needed.

An analogous step-size adjustment can be made in the other direction, that is, by increasing λ when appropriate. A case is shown in Figure 8.5. The top of the quadratic is obtained with a step size of $\lambda = 1$. However, the LL(β) is not quadratic, and its maximum is further away. The step size can be adjusted upward as long as LL(β) continues to rise. That is, calculate β_{t+1} with $\lambda = 1$ at β_{t+1}. If $\text{LL}(\beta_{t+1}) > \text{LL}(\beta_t)$, then try $\lambda = 2$. If the β_{t+1} based on $\lambda = 2$ gives a higher value of the log-likelihood function than with $\lambda = 1$, then try $\lambda = 4$, and so on, doubling λ as long as doing so further raises the likelihood function. Each time, $\text{LL}(\beta_{t+1})$ with a doubled λ is compared with its value at the previously tried λ, rather than with $\lambda = 1$, in order to assure that each doubling raises the likelihood function further than it had previously been raised with smaller λ's. In Figure 8.5, a final step size of 2 is used, since the likelihood function with $\lambda = 4$ is lower than when $\lambda = 2$, even though it is higher than with $\lambda = 1$.

The advantage of this approach of raising λ is that it usually reduces the number of iterations that are needed to reach the maximum. New values of λ can be tried without recalculating g_t and H_t, while each new

iteration requires calculation of these terms. Adjusting λ can therefore quicken the search for the maximum.

Concavity

If the log-likelihood function is globally concave, then the NR procedure is guaranteed to provide an increase in the likelihood function at each iteration. This fact is demonstrated as follows. $\text{LL}(\beta)$ being concave means that its Hessian is negative definite at all values of β. (In one dimension, the slope of $\text{LL}(\beta)$ is declining, so that the second derivative is negative.) If H is negative definite, then H^{-1} is also negative definite, and $-H^{-1}$ is positive definite. By definition, a symmetric matrix M is positive definite if $x'Mx > 0$ for any $x \neq 0$. Consider a first-order Taylor's approximation of $\text{LL}(\beta_{t+1})$ around $\text{LL}(\beta_t)$:

$$\text{LL}(\beta_{t+1}) = \text{LL}(\beta_t) + (\beta_{t+1} - \beta_t)' g_t.$$

Under the NR procedure, $\beta_{t+1} - \beta_t = \lambda(-H_t^{-1})g_t$. Substituting gives

$$\begin{aligned}\text{LL}(\beta_{t+1}) &= \text{LL}(\beta_t) + \big(\lambda\big(-H_t^{-1}\big)g_t\big)' g_t \\ &= \text{LL}(\beta_t) + \lambda g_t'\big(-H_t^{-1}\big)g_t.\end{aligned}$$

Since $-H^{-1}$ is positive definite, we have $g_t'(-H_t^{-1})g_t > 0$ and $\text{LL}(\beta_{t+1}) > \text{LL}(\beta_t)$. Note that since this comparison is based on a first-order approximation, an increase in $\text{LL}(\beta)$ may only be obtained in a small neighborhood of β_t. That is, the value of λ that provides an increase might be small. However, an increase is indeed guaranteed at each iteration if $\text{LL}(\beta)$ is globally concave.

Suppose the log-likelihood function has regions that are not concave. In these areas, the NR procedure can fail to find an increase. If the function is convex at β_t, then the NR procedure moves in the opposite direction to the slope of the log-likelihood function. The situation is illustrated in Figure 8.6 for $K = 1$. The NR step with one parameter is $\text{LL}'(\beta)/(-\text{LL}''(\beta))$, where the prime denotes derivatives. The second derivative is positive at β_t, since the slope is rising. Therefore, $-\text{LL}''(\beta)$ is negative, and the step is in the opposite direction to the slope. With $K > 1$, if the Hessian is positive definite at β_t, then $-H_t^{-1}$ is negative definite, and NR steps in the opposite direction to g_t.

The sign of the Hessian can be reversed in these situations. However, there is no reason for using the Hessian where the function is not concave, since the Hessian in convex regions does not provide any useful information on where the maximum might be. There are easier ways to find

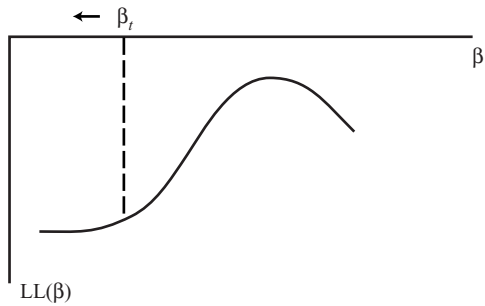

Figure 8.6. NR in the convex portion of LL.

an increase in these situations than calculating the Hessian and reversing its sign. This issue is part of the motivation for other procedures.

The NR procedure has two drawbacks. First, calculation of the Hessian is usually computation-intensive. Procedures that avoid calculating the Hessian at every iteration can be much faster. Second, as we have just shown, the NR procedure does not guarantee an increase in each step if the log-likelihood function is not globally concave. When $-H_t^{-1}$ is not positive definite, an increase is not guaranteed.

Other approaches use approximations to the Hessian that address these two issues. The methods differ in the form of the approximation. Each procedure defines a step as

$$\beta_{t+1} = \beta_t + \lambda M_t g_t,$$

where M_t is a $K \times K$ matrix. For NR, $M_t = -H^{-1}$. Other procedures use M_t's that are easier to calculate than the Hessian and are necessarily positive definite, so as to guarantee an increase at each iteration even in convex regions of the log-likelihood function.

8.3.2. BHHH

The NR procedure does not utilize the fact that the function being maximized is actually the sum of log likelihoods over a sample of observations. The gradient and Hessian are calculated just as one would do in maximizing any function. This characteristic of NR provides generality, in that the NR procedure can be used to maximize any function, not just a log likelihood. However, as we will see, maximization can be faster if we utilize the fact that the function being maximized is a sum of terms in a sample.

We need some additional notation to reflect the fact that the log-likelihood function is a sum over observations. The *score* of an

observation is the derivative of that observation's log likelihood with respect to the parameters: $s_n(\beta_t) = \partial \ln P_n(\beta)/\partial\beta$ evaluated at β_t. The gradient, which we defined earlier and used for the NR procedure, is the average score: $g_t = \sum_n s_n(\beta_t)/N$. The outer product of observation n's score is the $K \times K$ matrix

$$s_n(\beta_t)s_n(\beta_t)' = \begin{pmatrix} s_n^1 s_n^1 & s_n^1 s_n^2 & \cdots & s_n^1 s_n^K \\ s_n^1 s_n^2 & s_n^2 s_n^2 & \cdots & s_n^2 s_n^K \\ \vdots & \vdots & & \vdots \\ s_n^1 s_n^K & s_n^2 s_n^K & \cdots & s_n^K s_n^K \end{pmatrix},$$

where s_n^k is the kth element of $s_n(\beta_t)$ with the dependence on β_t omitted for convenience. The average outer product in the sample is $B_t = \sum_n s_n(\beta_t)s_n(\beta_t)'/N$. This average is related to the covariance matrix: if the average score were zero, then B would be the covariance matrix of scores in the sample. Often B_t is called the "outer product of the gradient." This term can be confusing, since B_t is not the outer product of g_t. However, it does reflect the fact that the score is an observation-specific gradient and B_t is the average outer product of these observation-specific gradients.

At the parameters that maximize the likelihood function, the average score is indeed zero. The maximum occurs where the slope is zero, which means that the gradient, that is, the average score, is zero. Since the average score is zero, the outer product of the scores, B_t, becomes the variance of the scores. That is, at the maximizing values of the parameters, B_t is the variance of scores in the sample.

The variance of the scores provides important information for locating the maximum of the likelihood function. In particular, this variance provides a measure of the curvature of the log-likelihood function, similar to the Hessian. Suppose that all the people in the sample have similar scores. Then the sample contains very little information. The log-likelihood function is fairly flat in this situation, reflecting the fact that different values of the parameters fit the data about the same. The first panel of Figure 8.7 depicts this situation: with a fairly flat log likelihood, different values of β give similar values of $\text{LL}(\beta)$. The curvature is small when the variance of the scores is small. Conversely, scores differing greatly over observations mean that the observations are quite different and the sample provides a considerable amount of information. The log-likelihood function is highly peaked, reflecting the fact that the sample provides good information on the values of β. Moving away from the maximizing values of β causes a large loss of fit. The second panel

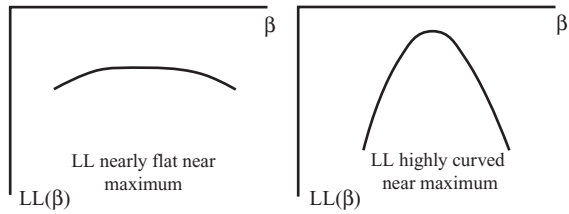

Figure 8.7. Shape of log-likelihood function near maximum.

of Figure 8.7 illustrates this situation. The curvature is great when the variance of the scores is high.

These ideas about the variance of the scores and their relation to the curvature of the log-likelihood function are formalized in the famous *information identity*. This identity states that the covariance of the scores at the true parameters is equal to the negative of the expected Hessian. We demonstrate this identity in the last section of this chapter; Theil (1971) and Ruud (2000) also provide useful and heuristic proofs. However, even without proof, it makes intuitive sense that the variance of the scores provides information on the curvature of the log-likelihood function.

Berndt, Hall, Hall, and Hausman (1974), hereafter referred to as BHHH (and commonly pronounced B-triple H), proposed using this relationship in the numerical search for the maximum of the log-likelihood function. In particular, the BHHH procedure uses B_t in the optimization routine in place of $-H_t$. Each iteration is defined by

$$\beta_{t+1} = \beta_t + \lambda B_t^{-1} g_t.$$

This step is the same as for NR except that B_t is used in place of $-H_t$. Given the preceding discussion about the variance of the scores indicating the curvature of the log-likelihood function, replacing $-H_t$ with B_t makes sense.

There are two advantages to the BHHH procedure over NR:

1. B_t is far faster to calculate that H_t. The scores must be calculated to obtain the gradient for the NR procedure anyway, and so calculating B_t as the average outer product of the scores takes hardly any extra computer time. In contrast, calculating H_t requires calculating the second derivatives of the log-likelihood function.
2. B_t is necessarily positive definite. The BHHH procedure is therefore guaranteed to provide an increase in LL(β) in each iteration, even in convex portions of the function. Using the proof given previously for NR when $-H_t$ is positive definite, the BHHH step $\lambda B_t^{-1} g_t$ raises LL(β) for a small enough λ.

Our discussion about the relation of the variance of the scores to the curvature of the log-likelihood function can be stated a bit more precisely. For a correctly specified model at the true parameters, $B \to -H$ as $N \to \infty$. This relation between the two matrices is an implication of the information identity, discussed at greater length in the last section. This convergence suggests that B_t can be considered an approximation to $-H_t$. The approximation is expected to be better as the sample size rises. And the approximation can be expected to be better close to the true parameters, where the expected score is zero and the information identity holds, than for values of β that are farther from the true values. That is, B_t can be expected to be a better approximation close to the maximum of the LL(β) than farther from the maximum.

There are some drawbacks of BHHH. The procedure can give small steps that raise LL(β) very little, especially when the iterative process is far from the maximum. This behavior can arise because B_t is not a good approximation to $-H_t$ far from the true value, or because LL(β) is highly nonquadratic in the area where the problem is occurring. If the function is highly nonquadratic, NR does not perform well, as explained earlier; since BHHH is an approximation to NR, BHHH would not perform well even if B_t were a good approximation to $-H_t$.

8.3.3. BHHH-2

The BHHH procedure relies on the matrix B_t, which, as we have described, captures the covariance of the scores when the average score is zero (i.e., at the maximizing value of β). When the iterative process is not at the maximum, the average score is not zero and B_t does not represent the covariance of the scores.

A variant on the BHHH procedure is obtained by subtracting out the mean score before taking the outer product. For any level of the average score, the covariance of the scores over the sampled decision makers is

$$W_t = \sum_n \frac{(s_n(\beta_t) - g_t)(s_n(\beta_t) - g_t)'}{N},$$

where the gradient g_t is the average score. W_t is the covariance of the scores around their mean, and B_t is the average outer product of the scores. W_t and B_t are the same when the mean gradient is zero (i.e., at the maximizing value of β), but differ otherwise.

The maximization procedure can use W_t instead of B_t:

$$\beta_{t+1} = \beta_t + \lambda W_t^{-1} g_t.$$

This procedure, which I call BHHH-2, has the same advantages as BHHH. W_t is necessarily positive definite, since it is a covariance matrix, and so the procedure is guaranteed to provide an increase in $\mathrm{LL}(\beta)$ at every iteration. Also, for a correctly specified model at the true parameters, $W \to -H$ as $N \to \infty$, so that W_t can be considered an approximation to $-H_t$. The information identity establishes this equivalence, as it does for B.

For β's that are close to the maximizing value, BHHH and BHHH-2 give nearly the same results. They can differ greatly at values far from the maximum. Experience indicates, however, that the two methods are fairly similar in that either both of them work effectively for a given likelihood function, or neither of them does. The main value of BHHH-2 is pedagogical, to elucidate the relation between the covariance of the scores and the average outer product of the scores. This relation is critical in the analysis of the information identity in Section 8.7.

8.3.4. Steepest Ascent

This procedure is defined by the iteration formula

$$\beta_{t+1} = \beta_t + \lambda g_t.$$

The defining matrix for this procedure is the identity matrix I. Since I is positive definite, the method guarantees an increase in each iteration. It is called "steepest ascent" because it provides the greatest possible increase in $\mathrm{LL}(\beta)$ for the distance between β_t and β_{t+1}, at least for small enough distance. Any other step of the same distance provides less increase. This fact is demonstrated as follows. Take a first-order Taylor's expansion of $\mathrm{LL}(\beta_{t+1})$ around $\mathrm{LL}(\beta_t)$: $\mathrm{LL}(\beta_{t+1}) = \mathrm{LL}(\beta_t) + (\beta_{t+1} - \beta_t)g_t$. Maximize this expression for $\mathrm{LL}(\beta_{t+1})$ subject to the Euclidian distance from β_t to β_{t+1} being $\sqrt{k}$. That is, maximize subject to $(\beta_{t+1} - \beta_t)'(\beta_{t+1} - \beta_t) = k$. The Lagrangian is

$$\mathrm{L} = \mathrm{LL}(\beta_t) + (\beta_{t+1} - \beta_t)g_t - \frac{1}{2\lambda}[(\beta_{t+1} - \beta_t)'(\beta_{t+1} - \beta_t) - k],$$

and we have

$$\begin{aligned} \frac{\partial \mathrm{L}}{\partial \beta_{t+1}} &= g_t - \frac{1}{\lambda}(\beta_{t+1} - \beta_t) = 0, \\ \beta_{t+1} - \beta_t &= \lambda g_t, \\ \beta_{t+1} &= \beta_t + \lambda g_t, \end{aligned}$$

which is the formula for steepest ascent.

At first encounter, one might think that the method of steepest ascent is the best possible procedure, since it gives the greatest possible increase in the log-likelihood function at each step. However, the method's property is actually less grand than this statement implies. Note that the derivation relies on a first-order approximation that is only accurate in a neighborhood of β_t. The correct statement of the result is that there is some sufficiently small distance for which the method of steepest ascent gives the greatest increase for that distance. This distinction is critical. Experience indicates that the step sizes are often very small with this method. The fact that the ascent is greater than for any other step of the same distance is not particularly comforting when the steps are so small. Usually, BHHH and BHHH-2 converge more quickly than the method of steepest ascent.

8.3.5. DFP and BFGS

The Davidon–Fletcher–Powell (DFP) and Broyden–Fletcher–Goldfarb–Shanno (BFGS) methods calculate the approximate Hessian in a way that uses information at more than one point on the likelihood function. Recall that NR uses the actual Hessian at β_t to determine the step to β_{t+1}, and BHHH and BHHH-2 use the scores at β_t to approximate the Hessian. Only information at β_t is being used to determine the step in these procedures. If the function is quadratic, then information at one point on the function provides all the information that is needed about the shape of the function. These methods work well, therefore, when the log-likelihood function is close to quadratic. In contrast, the DFP and BFGS procedures use information at several points to obtain a sense of the curvature of the log-likelihood function.

The Hessian is the matrix of second derivatives. As such, it gives the amount by which the slope of the curve changes as one moves along the curve. The Hessian is defined for infinitesimally small movements. Since we are interested in making large steps, understanding how the slope changes for noninfinitesimal movements is useful. An *arc* Hessian can be defined on the basis of how the gradient changes from one point to another. For example, for function $f(x)$, suppose the slope at $x = 3$ is 25 and at $x = 4$ the slope is 19. The change in slope for a one unit change in x is -6. In this case, the arc Hessian is -6, representing the change in the slope as a step is taken from $x = 3$ to $x = 4$.

The DFP and BFGS procedures use these concepts to approximate the Hessian. The gradient is calculated at each step in the iteration process. The difference in the gradient between the various points that have been reached is used to calculate an arc Hessian over these points. This

arc Hessian reflects the change in gradient that occurs for actual movement on the curve, as opposed to the Hessian, which simply reflects the change in slope for infinitesimally small steps around that point. When the log-likelihood function is nonquadratic, the Hessian at any point provides little information about the shape of the function. The arc Hessian provides better information.

At each iteration, the DFP and BFGS procedures update the arc Hessian using information that is obtained at the new point, that is, using the new gradient. The two procedures differ in how the updating is performed; see Greene (2000) for details. Both methods are extremely effective – usually far more efficient that NR, BHHH, BHHH-2, or steepest ascent. BFGS refines DFP, and my experience indicates that it nearly always works better. BFGS is the default algorithm in the optimization routines of many commercial software packages.

8.4 Convergence Criterion

In theory the maximum of LL(β) occurs when the gradient vector is zero. In practice, the calculated gradient vector is never exactly zero: it can be very close, but a series of calculations on a computer cannot produce a result of exactly zero (unless, of course, the result is set to zero through a Boolean operator or by multiplication by zero, neither of which arises in calculation of the gradient). The question arises: when are we sufficiently close to the maximum to justify stopping the iterative process?

The statistic $m_t = g_t'(-H_t^{-1})g_t$ is often used to evaluate convergence. The researcher specifies a small value for m, such as $\breve{m} = 0.0001$, and determines in each iteration whether $g_t'(-H_t^{-1})g_t < \breve{m}$. If this inequality is satisfied, the iterative process stops and the parameters at that iteration are considered the converged values, that is, the estimates. For procedures other than NR that use an approximate Hessian in the iterative process, the approximation is used in the convergence statistic to avoid calculating the actual Hessian. Close to the maximum, where the criterion becomes relevant, each form of approximate Hessian that we have discussed is expected to be similar to the actual Hessian.

The statistic m_t is the test statistic for the hypothesis that all elements of the gradient vector are zero. The statistic is distributed chi-squared with K degrees of freedom. However, the convergence criterion $\breve{m}$ is usually set far more stringently (that is, lower) than the critical value of a chi-squared at standard levels of significance, so as to assure that the estimated parameters are very close to the maximizing values. Usually, the hypothesis that the gradient elements are zero cannot be rejected for a

fairly wide area around the maximum. The distinction can be illustrated for an estimated coefficient that has a t-statistic of 1.96. The hypothesis cannot be rejected if this coefficient has any value between zero and twice its estimated value. However, we would not want convergence to be defined as having reached any parameter value within this range.

It is tempting to view small changes in β_t from one iteration to the next, and correspondingly small increases in $\text{LL}(\beta_t)$, as evidence that convergence has been achieved. However, as stated earlier, the iterative procedures may produce small steps because the likelihood function is not close to a quadratic rather than because of nearing the maximum. Small changes in β_t and $\text{LL}(\beta_t)$ accompanied by a gradient vector that is not close to zero indicate that the numerical routine is not effective at finding the maximum.

Convergence is sometimes assessed on the basis of the gradient vector itself rather than through the test statistic m_t. There are two procedures: (1) determine whether each element of the gradient vector is smaller in magnitude than some value that the researcher specifies, and (2) divide each element of the gradient vector by the corresponding element of β, and determine whether each of these quotients is smaller in magnitude than some value specified by the researcher. The second approach normalizes for the units of the parameters, which are determined by the units of the variables that enter the model.

8.5 Local versus Global Maximum

All of the methods that we have discussed are susceptible to converging at a local maximum that is not the global maximum, as shown in Figure 8.8. When the log-likelihood function is globally concave, as for logit with linear-in-parameters utility, then there is only one maximum and the issue doesn't arise. However, most discrete choice models are not globally concave.

A way to investigate the issue is to use a variety of starting values and observe whether convergence occurs at the same parameter values. For example, in Figure 8.8, starting at β_0 will lead to convergence at β_1. Unless other starting values were tried, the researcher would mistakenly believe that the maximum of $\text{LL}(\beta)$ had been achieved. Starting at β_2, convergence is achieved at $\hat{\beta}$. By comparing $\text{LL}(\hat{\beta})$ with $\text{LL}(\beta_1)$, the researcher finds that β_1 is not the maximizing value. Liu and Mahmassani (2000) propose a way to select starting values that involves the researcher setting upper and lower bounds on each parameter and randomly choosing values within those bounds.

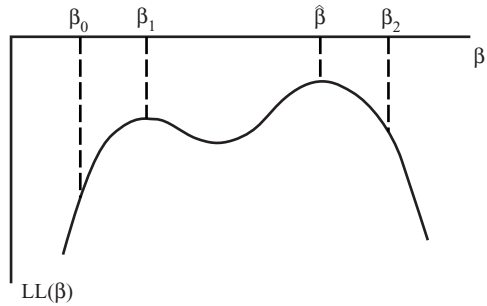

Figure 8.8. Local versus global maximum.

8.6 Variance of the Estimates

In standard econometric courses, it is shown that, for a correctly specified model,

$$\sqrt{N}(\hat{\beta} - \beta^*) \xrightarrow{d} N(0, (-\mathbf{H})^{-1})$$

as $N \to \infty$, where β^* is the true parameter vector, $\hat{\beta}$ is the maximum likelihood estimator, and $\mathbf{H}$ is the expected Hessian in the population. The negative of the expected Hessian, $-\mathbf{H}$, is often called the information matrix. Stated in words, the sampling distribution of the difference between the estimator and the true value, normalized for sample size, converges asymptotically to a normal distribution centered on zero and with covariance equal to the inverse of the information matrix, $-\mathbf{H}^{-1}$. Since the asymptotic covariance of $\sqrt{N}(\hat{\beta} - \beta^*)$ is $-\mathbf{H}^{-1}$, the asymptotic covariance of $\hat{\beta}$ itself is $-\mathbf{H}^{-1}/N$.

The boldface type in these expressions indicates that $\mathbf{H}$ is the average in the population, as opposed to H, which is the average Hessian in the sample. The researcher calculates the asymptotic covariance by using H as an estimate of $\mathbf{H}$. That is, the asymptotic covariance of $\hat{\beta}$ is calculated as $-H^{-1}/N$, where H is evaluated at $\hat{\beta}$.

Recall that W is the covariance of the scores in the sample. At the maximizing values of β, B is also the covariance of the scores. By the information identity just discussed and explained in the last section, $-H$, which is the (negative of the) average Hessian in the sample, converges to the covariance of the scores for a correctly specified model at the true parameters. In calculating the asymptotic covariance of the estimates $\hat{\beta}$, any of these three matrices can be used as an estimate of $-\mathbf{H}$. The asymptotic variance of $\hat{\beta}$ is calculated as W^{-1}/N, B^{-1}/N, or $-H^{-1}/N$, where each of these matrices is evaluated at $\hat{\beta}$.

If the model is not correctly specified, then the asymptotic covariance of $\hat{\beta}$ is more complicated. In particular, for any model for which the expected score is zero at the true parameters,

$$\sqrt{N}(\hat{\beta} - \beta^*) \xrightarrow{d} N(0, \mathbf{H}^{-1}\mathbf{V}\mathbf{H}^{-1}),$$

where $\mathbf{V}$ is the variance of the scores in the population. When the model is correctly specified, the matrix $-\mathbf{H} = \mathbf{V}$ by the information identity, such that $\mathbf{H}^{-1}\mathbf{V}\mathbf{H}^{-1} = -\mathbf{H}^{-1}$ and we get the formula for a correctly specified model. However, if the model is not correctly specified, this simplification does not occur. The asymptotic variance of $\hat{\beta}$ is $\mathbf{H}^{-1}\mathbf{V}\mathbf{H}^{-1}/N$. This matrix is called the *robust covariance matrix*, since it is valid whether or not the model is correctly specified.

To estimate the robust covariance matrix, the researcher must calculate the Hessian H. If a procedure other than NR is being used to reach convergence, the Hessian need not be calculated at each iteration; however, it must be calculated at the final iteration. Then the asymptotic covariance is calculated as $H^{-1}WH^{-1}$, or with B instead of W. This formula is sometimes called the "sandwich" estimator of the covariance, since the Hessian inverse appears on both sides.

An alternative way to estimate the covariance matrix is through bootstrapping, as suggested by Efron (1979). Under this procedure, the model is re-estimated numerous times on different samples taken from the original sample. Let the original sample be labeled A, which consists of the decision-makers that we have been indexing by $n = 1, \ldots, N$. That is, the original sample consists of N observations. The estimate that is obtained on this sample is $\hat{\beta}$. Bootstapping consists of the following steps:

1. Randomly sample *with replacement* N observations from the original sample A. Since the sampling is with replacement, some decision-makers might be represented more than once in the new sample and others might not be included at all. This new sample is the same size as the original, but looks different from the original because some decision-makers are repeated and others are not included.
2. Re-estimate the model on this new sample, and label the estimate β_r with $r = 1$ for this first new sample.
3. Repeated steps 1 and 2 numerous times, obtaining estimates β_r for $r = 1, \ldots, R$ where R is the number of times the estimation is repeated on a new sample.
4. Calculate the covariance of the resulting estimates around the original estimate: $V = \frac{1}{R}\sum_r (\beta_r - \hat{\beta})(\beta_r - \hat{\beta})'$.

This V is an estimate of the asymptotic covariance matrix. The sampling variance for any statistics that is based on the parameters is calculated similarly. For scalar statistic $t(\beta)$, the sampling variance is $\sum_r [t(\beta_r) - t(\hat{\beta})]^2 / R$.

The logic of the procedure is the following. The sampling covariance of an estimator is, by definition, a measure of the amount by which the estimates change when different samples are taken from the population. Our original sample is one sample from the population. However, if this sample is large enough, then it is probably similar to the population, such that drawing from it is similar to drawing from the population itself. The bootstrap does just that: draws from the original sample, with replacement, as a proxy for drawing from the population itself. The estimates obtained on the bootstrapped samples provide information on the distribution of estimates that would be obtained if alternative samples had actually been drawn from the population.

The advantage of the bootstrap is that it is conceptually straightforward and does not rely on formulas that hold asymptotically but might not be particularly accurate for a given sample size. Its disadvantage is that it is computer-intensive since it entails estimating the model numerous times. Efron and Tibshirant (1993) and Vinod (1993) provide useful discussions and applications.

8.7 Information Identity

The information identity states that, for a correctly specified model at the true parameters, $\mathbf{V} = -\mathbf{H}$, where $\mathbf{V}$ is the covariance matrix of the scores in the population and $\mathbf{H}$ is the average Hessian in the population. The score for a person is the vector of first derivatives of that person's $\ln P(\beta)$ with respect to the parameters, and the Hessian is the matrix of second derivatives. The information identity states that, in the population, the covariance matrix of the first derivatives equals the average matrix of second derivatives (actually, the negative of that matrix). This is a startling fact, not something that would be expected or even believed if there were not proof. It has implications throughout econometrics. The implications that we have used in the previous sections of this chapter are easily derivable from the identity. In particular:

(1) *At the maximizing value of β, $W \to -H$ as $N \to \infty$, where W is the sample covariance of the scores and H is the sample average of each observation's Hessian.* As sample size rises, the sample covariance approaches the population covariance: $W \to \mathbf{V}$. Similarly, the sample average of the Hessian approaches the population average: $H \to \mathbf{H}$.

Since $\mathbf{V} = -\mathbf{H}$ by the information identity, W approaches the same matrix that $-H$ approaches, and so they approach each other.

(2) *At the maximizing value of β, $B \to -H$ as $N \to \infty$, where B is the sample average of the outer product of the scores.* At $\hat{\beta}$, the average score in the sample is zero, so that B is the same as W. The result for W applies for B.

We now demonstrate the information identity. We need to expand our notation to encompass the population instead of simply the sample. Let $P_i(x, \beta)$ be the probability that a person who faces explanatory variables x chooses alternative i given the parameters β. Of the people in the population who face variables x, the share who choose alternative i is this probability calculated at the true parameters: $S_i(x) = P_i(x, \beta^*)$ where β^* are the true parameters. Consider now the gradient of $\ln P_i(x, \beta)$ with respect to β. The average gradient in the population is

$$\mathbf{g} = \int \sum_i \frac{\partial \ln P_i(x, \beta)}{\partial \beta} S_i(x) f(x)\, dx, \tag{8.2}$$

where $f(x)$ is the density of explanatory variables in the population. This expression can be explained as follows. The gradient for people who face x and choose i is $\partial \ln P_{ni}(\beta)/\partial \beta$. The average gradient is the average of this term over all values of x and all alternatives i. The share of people who face a given value of x is given by $f(x)$, and the share of people who face this x that choose i is $S_i(x)$. So $S_i(x) f(x)$ is the share of the population who face x and choose i and therefore have gradient $\partial \ln P_i(x, \beta)/\partial \beta$. Summing this term over all values of i and integrating over all values of x (assuming the x's are continuous) gives the average gradient, as expressed in (8.2).

The average gradient in the population is equal to zero at the true parameters. This fact can be considered either the definition of the true parameters or the result of a correctly specified model. Also, we know that $S_i(x) = P_i(x, \beta^*)$. Substituting these facts into (8.2), we have

$$0 = \int \sum_i \frac{\partial \ln P_i(x, \beta)}{\partial \beta} P_i(x, \beta) f(x)\, dx,$$

where all functions are evaluated at β^*. We now take the derivative of this equation with respect to the parameters:

$$0 = \int \sum_i \left(\frac{\partial^2 \ln P_i(x, \beta)}{\partial \beta\, \partial \beta'} P_i(x, \beta) + \frac{\partial \ln P_i(x, \beta)}{\partial \beta} \frac{\partial P_i(x, \beta)}{\partial \beta'} \right) f(x)\, dx.$$

Since $\partial \ln P/\partial\beta = (1/P)\,\partial P/\partial\beta$ by the rules of derivatives, we can substitute $[\partial \ln P_i(x,\beta)/\partial\beta']P_i(x,\beta)$ for $\partial P_i(x,\beta)/\partial\beta'$ in the last term in parentheses:

$$0 = \int \sum_i \left(\frac{\partial^2 \ln P_i(x,\beta)}{\partial\beta\,\partial\beta'} P_i(x,\beta) + \frac{\partial \ln P_i(x,\beta)}{\partial\beta} \frac{\partial \ln P_i(x,\beta)}{\partial\beta'} P_i(x,\beta) \right) f(x)\,dx.$$

Rearranging,

$$-\int \sum_i \frac{\partial^2 \ln P_i(x,\beta)}{\partial\beta\,\partial\beta'} P_i(x,\beta) f(x)\,dx = \int \sum_i \frac{\partial \ln P_i(x,\beta)}{\partial\beta} \frac{\partial \ln P_i(x,\beta)}{\partial\beta'} P_i(x,\beta) f(x)\,dx.$$

Since all terms are evaluated at the true parameters, we can replace $P_i(x,\beta)$ with $S_i(x)$ to obtain

$$-\int \sum_i \frac{\partial^2 \ln P_i(x,\beta)}{\partial\beta\,\partial\beta'} S_i(x) f(x)\,dx = \int \sum_i \frac{\partial \ln P_i(x,\beta)}{\partial\beta} \frac{\partial \ln P_i(x,\beta)}{\partial\beta'} S_i(x) f(x)\,dx.$$

The left-hand side is the negative of the average Hessian in the population, $-\mathbf{H}$. The right-hand side is the average outer product of the gradient, which is the covariance of the gradient, $\mathbf{V}$, since the average gradient is zero. Therefore, $-\mathbf{H} = \mathbf{V}$, the information identity. As stated, the matrix $-\mathbf{H}$ is often called the information matrix.