Ann. Inst. Statist. Math.
Vol. 40, No. 4, 641-663 (1988)

MONOTONICITY OF QUADRATIC-APPROXIMATION ALGORITHMS

DANKMAR BÖHNING* AND BRUCE G. LINDSAY**

Department of Statistics, The Pennsylvania State University, 219 Pond Laboratory, University Park, PA 16802, U.S.A.

(Received June 18, 1987; revised July 28, 1988)

Abstract. It is desirable that a numerical maximization algorithm monotonically increase its objective function for the sake of its stability of convergence. It is here shown how one can adjust the Newton-Raphson procedure to attain monotonicity by the use of simple bounds on the curvature of the objective function. The fundamental tool in the analysis is the geometric insight one gains by interpreting quadratic-approximation algorithms as a form of area approximation. The statistical examples discussed include maximum likelihood estimation in mixture models, logistic regression and Cox's proportional hazards regression.

Key words and phrases: Maximum likelihood estimation, curvature, monotonicity, algorithms, Newton-Raphson algorithm.

1. Introduction and overview

Various statistical questions lead to the following problem:

(1.1) Find $\hat{\pi} \in \Pi$ such that $l(\hat{\pi}) \geq l(\pi)$ for all $\pi \in \Pi$.

The set $\Pi \subseteq \mathbb{R}^p$ is a set of feasible parameter values and

$$l: \Pi \to l(\pi) \in \mathbb{R},$$

describes the objective function to be maximized, in this paper always a log-likelihood function. In the absence of a closed form solution to problem (1.1), we consider *algorithmic* ways to attack the problem at hand.

*The first named author is in the Department of Epidemiology, Freie Universität Berlin, Augustastr. 37, 1000 Berlin 45, Germany.

**The second author's research was partially supported by the National Science Foundation under Grant DMS-8402735.

If the solution to (1.1) lies on the boundary of Π, then the constraints imposed by Π may play an important role in an algorithm. Here we focus on the simpler *unconstrained* optimization problem of finding solutions in the interior.

In specifying the properties an algorithmic solution should have, we frequently find the criterion of *reliable convergence* (to $\hat{\pi}$) in the literature. This issue can be critical: if we take as an example a simulation study in which a certain algorithm is started 100,000 times and converges in 90% of all cases to $\hat{\pi}$, and in 10% to something else, then the study results may be highly biased.

A second major issue is that of *convergence rate*. The convergence rate measures the gain at each step relative to the gain in the step before. An algorithm with a "good" convergence rate will give a "large" improvement at each step. Algorithms with good convergence rates often require a larger number of operations at each step; in other words, the price to pay for the large improvement at each step is a high numerical complexity. A measure which adjusts to both these is the *computational efficiency*: the overall price paid to obtain $\hat{\pi}$.

Sometimes other issues are discussed, such as the *simplicity* and *numerical stability* of an algorithm.

In the present paper we focus on reliable convergence and convergence rates for a class of quadratic-approximation based algorithms related to the Newton-Raphson algorithm. The ideas are related to the EM-concept of Baum and Eagon (1967), Sundberg (1976), Dempster *et al.* (1977) and Wu (1983). In this regard we note that the EM-algorithm—celebrated for the monotonicity property which ensures reliable convergence—converges at a linear rate at best. Despite this apparent disadvantage, we note that the numerical complexity involved at each step of the EM-algorithm is usually low, and, especially in cases where the linear rate is governed by a small rate-factor, the EM-algorithm could be more computationally efficient than Newtonian methods. On the other hand, if the rate-factor is large the convergence can be very slow, and thought should be given to methods to accelerate its speed.

Quadratic methods are known to lead to rapidly convergent algorithms. Moreover, for some statistical problems such as logistic regression, the EM-approach is not applicable. For Newtonian methods, however, problems sometimes arise in terms of non-monotonicity, leading even to non-convergence. This fact is sometimes mentioned in the literature (e.g., Andersen (1980), p. 69). Cox and Oakes ((1984), p. 172) write:

> "Divergence is much more frequent, as a full Newton-Raphson step will not necessarily increase the log-likelihood. As against this, the Newton-Raphson procedure usually converges rapidly when it does converge, in particular when the log-likelihood is

well approximated by a quadratic function, and the inverse of the matrix of second derivatives is often needed for the estimation of standard errors."

The present paper considers the case where the quadratic-approximation to the log-likelihood based on the Taylor series is "flatter" than the objective function, thereby sending the solution too far at the next step. The idea of replacing the flat quadratic by a more curved quadratic leads naturally to our approach of replacing the Hessian by some "bigger" matrix. A key feature of our algorithms will be that they are *monotonic*—that is, every step increases the value of the objective function $l(\pi)$—and so have reliable convergence.

The paper is organized as follows: After presenting some notation, we provide in Section 3 a simple example of a *concave* objective function in which the Newton-Raphson method is nonconvergent. The example leads to a general discussion of convergence properties of the Newton-Raphson (NR) algorithm based on characteristics of the Hessian matrix. The focal point is the introduction of an "area estimation" interpretation of the NR algorithm. Two statistically important cases are discussed: in one, convergence or divergence depends on starting value; in the other, convergence is shown to hold by showing that monotonicity of the algorithm can be violated in at most one step.

Sections 4 and 5 present a lower-bound algorithm for problems in which there exists a single matrix which dominates the Hessian globally. It is shown that this can be used to create a monotonically convergent algorithm, and the linear rate is found. It has the additional advantage over the NR of not requiring repeated calculation and inversion of the Hessian. Two examples are given: logistic regression and Cox's proportional hazards model. For the logistic regression case we also present a simulation study comparing our method with Newton-Raphson in high dimensional problems.

In Section 6 we consider a second class of models in which there does not exist a global lower-bound on the Hessian, but the Hessian does possess a concavity property that enables one to bound the curvature along lines by the curvature at the endpoints. This structure enables one to ensure monotonicity by simple corrections to the NR algorithm.

2. Notations and definitions

- Parameter space of interest: $\Pi \subseteq \mathbb{R}^P$.
- Function to be maximized (*log-likelihood*) $l: \Pi \to \mathbb{R}$.
- Gradient of l at π (*score vector*):

$$\nabla l(\pi) = \left(\frac{\partial l}{\partial \pi_1}(\pi), \ldots, \frac{\partial l}{\partial \pi_p}(\pi) \right)^T .$$

- Hessian matrix of l at π (negative of *sample information*):

$$\nabla^2 l(\pi) = \left(\frac{\partial^2 l}{\partial \pi_j \partial \pi_i}(\pi) \right) .$$

Let A and B be two $p \times p$ matrices. Then we say that "A is greater than B in the Loewner ordering" (and write $A \geq B$) if $A - B$ is nonnegative definite (see also Marshall and Olkin (1979), p. 462).

If $p = 1$, so that π is a scalar, then $\nabla^i l(\pi)$ denotes the i-th derivative of l at π. $\|\pi\|$ denotes the Euclidean norm on $\mathbb{R}^p$, and if A is a p by p matrix, then $\|A\|$ represents the corresponding induced norm, $\sup \{\|A\pi\| / \|\pi\|: \pi \in \mathbb{R}^p\}$.

We will say that a sequence $(\pi_j: j = 1, 2, \ldots)$ converges *linearly* to $\hat{\pi}$ if there exists a constant $c \in (0, 1)$ such that

$$\|\pi_{j+1} - \hat{\pi}\| \leq c \|\pi_j - \hat{\pi}\| .$$

The sequence converges *quadratically* if there exists a positive c such that:

$$\|\pi_{j+1} - \hat{\pi}\| \leq c \|\pi_j - \hat{\pi}\|^2 .$$

The sequence converges *superlinearly* if

$$\limsup_{j \to \infty} \|\pi_{j+1} - \hat{\pi}\| / \|\pi_j - \hat{\pi}\| = 0 .$$

3. Convergence and quadratic-approximation

Suppose one uses a quadratic-approximation to $l(\pi)$ in a neighborhood of a current value π_0 based on the Taylor series:

$$(3.1) \quad l(\pi) \approx Q(\pi) := l(\pi_0) + (\pi - \pi_0)^T \nabla l(\pi_0) + (\pi - \pi_0)^T \nabla^2 l(\pi_0)(\pi - \pi_0)/2 .$$

Solution to the corresponding quadratic maximization problem creates the next value of the *Newton-Raphson algorithm*:

$$(3.2) \qquad \pi_{nr} = \pi_0 - \nabla^2 l(\pi_0)^{-1} \nabla l(\pi_0) .$$

It is somewhat surprising that concavity of the objective function is not sufficient to guarantee convergence of the Newton-Raphson algorithm. In this section we present some results regarding convergence in some impor-

tant statistical problems, based on the structure of the Hessian matrix. We start with a simple example of nonconvergence.

Example A. Suppose that:

$$l(\pi) = \begin{cases} \log(1+\pi) - \pi, & \pi \geq 0, \\ \log(1-\pi) + \pi, & \pi \leq 0. \end{cases}$$

For this objective function the gradient is:

$$\nabla l(\pi) = \begin{cases} \dfrac{1}{1+\pi} - 1, & \pi \geq 0, \\ -\dfrac{1}{1-\pi} + 1, & \pi \leq 0; \end{cases}$$

for which a graph is shown in Fig. 1. Note that l is strictly *concave*, twice continuously differentiable and *uniquely* maximized at $\hat{\pi} = 0$.

Here the Newton-Raphson iteration is given by

$$\pi_{nr} = \begin{cases} -\pi^2, & \pi \geq 0, \\ \pi^2, & \pi \leq 0. \end{cases}$$

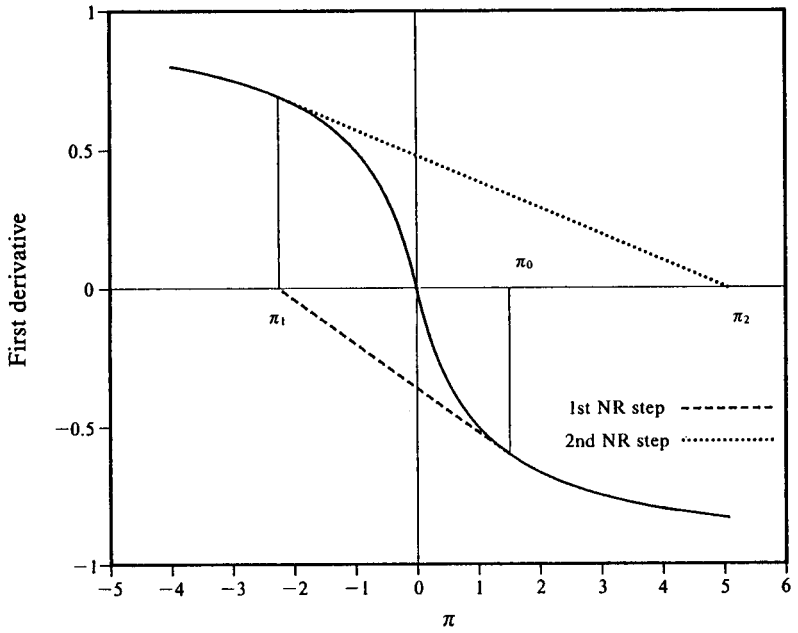

Fig. 1. The spider-web-effect.

Thus the algorithm has the characteristic that, depending on the initial value π, it does one of three things:

Converges	$\lvert\pi\rvert < 1$,
Oscillates between -1 and 1	$\pi \in \{-1, 1\}$,
Diverges	$\lvert\pi\rvert > 1$.

This last case, the *spider-web-effect*, is also pictured in Fig. 1. We seek to eliminate the potential for this behavior.

3.1 *Area-estimation*

Since the concavity of $l(\pi)$ does not ensure the convergence of the NR algorithm, one might ask if there are features of the problem which will guarantee this convergence. In order to explore this issue we offer an interpretation of the Newton-Raphson algorithm as an area-estimation algorithm, restricting attention to the univariate case, $p = 1$. Since $\nabla l(\hat{\pi}) = 0$, we have

$$\begin{aligned} &\nabla l(\pi_0) + \nabla l(\hat{\pi}) - \nabla l(\pi_0) = 0; \quad \text{or} \\ &\nabla l(\pi_0) + \int_{\pi_0}^{\hat{\pi}} \nabla^2 l(\pi) d\pi = 0 . \end{aligned} \tag{3.3}$$

If we think of the integral as the area defined by the Hessian curve, then another way to view equation (3.3) is provided in Fig. 2: namely,

$$\text{AREA} = -\nabla l(\pi_0) .$$

That is, although $\hat{\pi}$ is unknown, we do know the area above $\nabla^2 l(\pi)$ from π_0 to $\hat{\pi}$; it is $-\nabla l(\pi_0)$. Newton-Raphson assumes that AREA can be approximated by a rectangle with height $\nabla^2 l(\pi_0)$ and width $(\pi - \pi_0)$ so that:

$$-\nabla l(\pi_0) = \text{AREA} \approx (\hat{\pi} - \pi_0)\nabla^2 l(\pi_0) ,$$

is satisfied. We note that if the lower edge of the rectangle is entirely below the Hessian curve, then the step must be *directionally monotonic*. That is, the step must land on a point between π_0 and $\hat{\pi}$; if the objective function is concave (negative Hessian), then this is also a point of higher likelihood, and so the step is monotonic.

3.2 *Type I likelihoods*

The Hessian for Example A is pictured in Fig. 3. We might characterize it as follows:

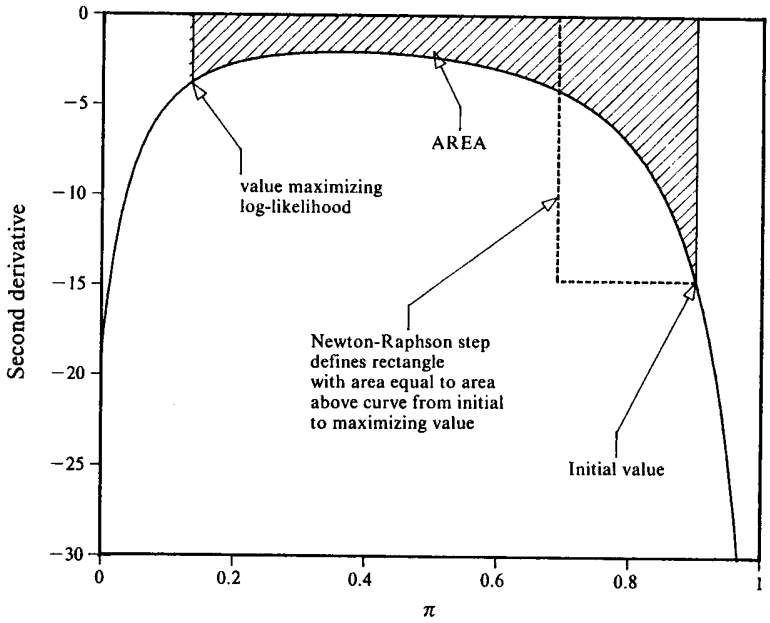

Fig. 2. Geometric interpretation of equation (3.3).

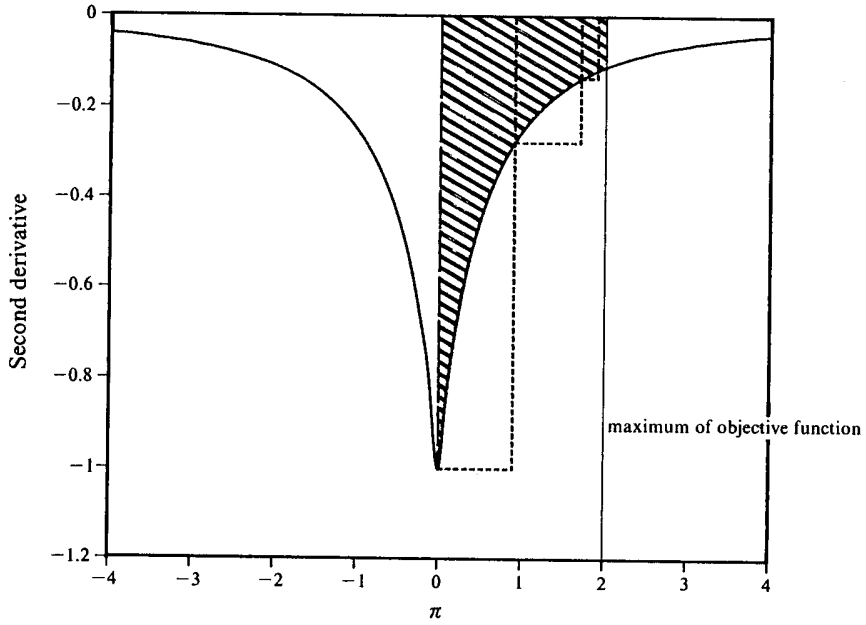

Fig. 3. Second derivative with two-sided monotonicity property.

Type I: • π is scalar;
• $\nabla^2 l(\pi)$ is negative, bounded below, and increasing in both directions from a point of minimality.

It is clear that using the point of minimality as a starting value for the NR algorithm in a Type I likelihood will ensure monotonic convergence, as NR will at every stage use a box that overestimates the corresponding area between the curve and the axis. It will be shown in Section 5 that in the logistic regression model, scalar case, that the likelihood is of Type I, with a point of minimality at $\pi = 0$, and we have therefore shown NR monotonicity when the starting value is $\pi = 0$. This univariate result is not easily extended to $p > 1$, but it does shed some light on the apparent good behavior of the NR algorithm in this model. That this good behavior is related to the starting value will be demonstrated in Section 5.

Example A motivates as well an alternative approach taken in Sections 4 and 5. That is, consider $\nabla^2 l(\pi)$, as shown in Fig. 3. Obviously, $\nabla^2 l$ is bounded below by $B = \nabla^2 l(0) = -1$. If instead of (3.1) we use the quadratic-approximation

$$l(\pi_0) + \nabla l(\pi_0)(\pi - \pi_0) + B(\pi - \pi_0)^2/2\,, \tag{3.4}$$

then, as verified in Theorem 4.1, we achieve a step which necessarily increases l regardless of starting value. That is, we have created a *globally monotonically* convergent procedure described by the mapping

$$\pi_{lb} = \pi + \nabla l(\pi) = \begin{cases} \pi + \left\{ \dfrac{1}{1+\pi} - 1 \right\}, & \pi \geq 0\,, \\ \pi + \left\{ 1 - \dfrac{1}{1-\pi} \right\}, & \pi \leq 0\,. \end{cases}$$

Although the convergence rate for algorithms created in this fashion are—in full generality—only linear, in this example the convergence rate is *superlinear* because the lower-bound is sharp at the solution point $\hat{\pi} = 0$.

3.3 *Type II likelihoods*

Another important type of structure can be recognized in Fig. 2, where the Hessian curve is not bounded below, but is concave. That is, consider the class of models:

Type II: π is scalar; $\nabla^2 l$ is negative; $\nabla^4 l$ is negative.

Here is an important example of a Type II likelihood:

Example B. (Mixture of two known densities) Consider a random sample taken from the density $\pi f(x) + (1 - \pi)g(x)$, where f and g are known densities and π is an unknown parameter in $[0, 1]$. The log-likelihood has the form:

$$l(\pi) = \Sigma k_j \log (f_j + \pi \Delta_j) ,$$

where $f_j = f(x_j)$ and $\Delta_j = g(x_j) - f(x_j)$. We then have:

$$\nabla^m l(\pi) = (m - 1) ! (- 1)^{m-1} \Sigma k_j \Delta_j^m / (f_j + \pi \Delta_j)^m .$$

For $m = 2$, this implies that l is concave. For $m = 4$, this implies that $\nabla^2 l(\pi)$ is concave, which is the property we wish to exploit. We will call this property *double concavity*. Note that this implies that the minimum of $\nabla^2 l(\pi)$ on an interval $[a, b]$ occurs at either a or b.

To demonstrate the behavior of the Newton-Raphson in such a problem, we consider a numerical example: let $g^T = (.6, .3, .05, .05)$, $f^T = (.05, .15, .3, .5)$ and $k^T = (.15, .1, .2, .55)$. The solution is $\hat{\pi} = .13522$. In Table 1 the steps of the Newton-Raphson are shown, given a starting value of .9. Note that Newton-Raphson oversteps the solution at Step 4. The two other algorithms in the table are modifications of the NR algorithm designed for monotonicity; they will be discussed in Section 6.

It is a remarkable property of the Newton-Raphson algorithm that in a doubly concave function the steps can be directionally non-monotonic *at most once*, as we now demonstrate. Let $\hat{\pi}$ denote the value maximizing l and π^* denote the maximizing value of $\nabla^2 l(\pi)$, assumed concave. We can characterize the behavior of the Newton-Raphson algorithm in the doubly concave model—as a function of the initial value π_0—as follows:

Case 1. If $\hat{\pi}$ is between π_0 and π^*, then the Newton-Raphson step will always be monotonic, as illustrated in Fig. 4, and all further steps are in Case 1.

Case 2. If the starting value π_0 lies between $\hat{\pi}$ and π^*, as in Fig. 5, then the step can be an arbitrarily bad overstep—how bad is clearly a

Table 1. A comparison of algorithms: the Newton-Raphson, (6.3) and (6.6).

Algorithm	Step number							
	1	2	3	4	5	6	7	8
NEWTON	.9	.69191	.34420	.08556	.12609	.13494	.13522	.13522
(6.3)	.9	.69191	.34420	.24900	.17756	.14133	.13534	.13522
(6.6)	.9	.69191	.34420	.17763	.13581	.13522	.13522	.13522

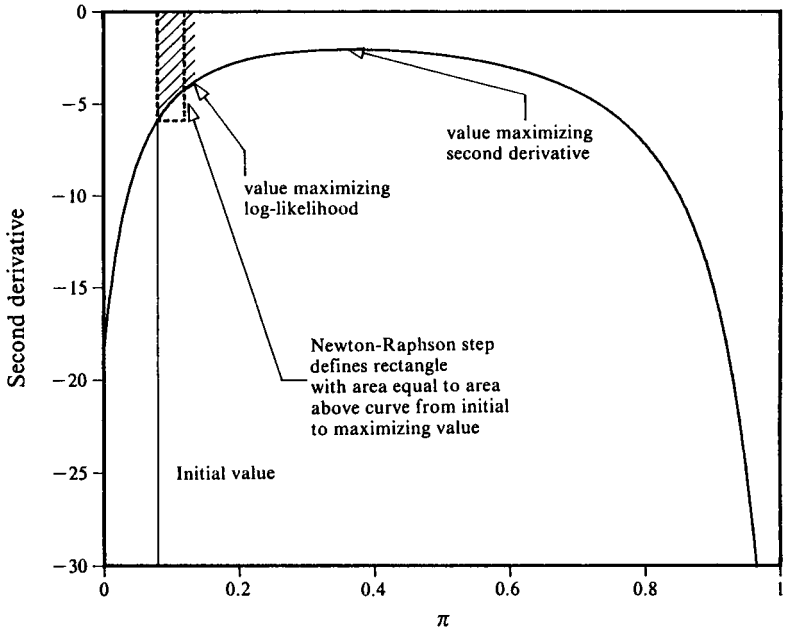

Fig. 4. Case 1 (good case) occurs from Step 4 in Table 1.

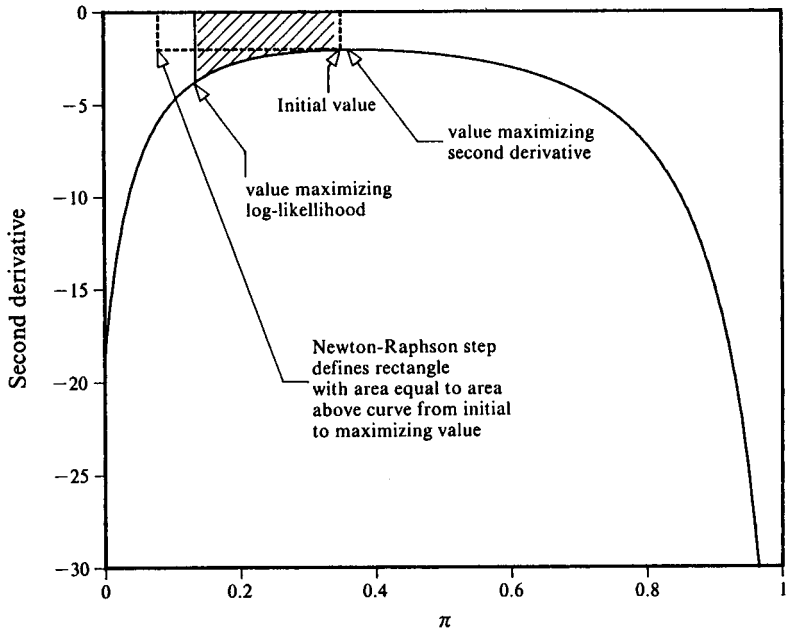

Fig. 5. Case 2 (overstep) occurs in Step 4 in Table 1.

function of the curvature of $\nabla^2 l(\pi)$. One other important property can be deduced from the figure: Once Case 2 occurs, all further steps will be in Case 1, and so a non-monotonic step of this type can occur but once.

Case 3. If π^* is between π_0 and $\hat{\pi}$, then we are in an undecided phase. Note that eventually (in a finite number of steps) the algorithm carries π_0 to some π_j which is in either Case 2 or Case 1.

In conclusion, we have shown that the Newton-Raphson algorithm is guaranteed to converge in the doubly concave situation, but that it can take one arbitrarily bad step.

This should be compared with other results concerning the monotonicity of the Newton-Raphson algorithm (see, e.g., Potra and Rheinboldt (1986)). For a function of one variable the conditions usually require at least concavity (or convexity) of ∇l (Collatz (1961), p. 102 and Horst (1979), p. 152) as the most important condition. For Example B, such a condition does not generally hold.

4. The lower-bound algorithm

We now extend the class of Type I likelihoods to the multivariable case by saying that the likelihood is *Type I** if

$$\nabla^2 l(\pi) \geq B\,, \tag{4.1}$$

is satisfied for all $\pi \in \Pi$, where B is some *symmetric, negative definite* matrix not depending on π. The *Lower Bound* algorithm (*LB*) will have steps defined by

$$\pi_{lb} = \pi_0 - B^{-1}\nabla l(\pi_0)\,. \tag{4.2}$$

This corresponds to a quadratic-approximation to $l(\pi) - l(\pi_0)$ of the following form:

$$Q_B(\pi) := (\pi - \pi_0)^T \nabla l(\pi_0) + (\pi - \pi_0)^T B(\pi - \pi_0)/2\,, \tag{4.3}$$

where π_0, $\pi \in \Pi$ and B is a *symmetric, negative definite* matrix. If $B = \nabla^2 l(\pi_0)$, then (4.1) is the second-order approximation to $l(\pi) - l(\pi_0)$. It is important to notice that, for every B, Q_B matches $l(\pi) - l(\pi_0)$ in slope at π_0 and has value 0 there. The following lemma collects some useful information about this approximation; it is followed by the main theorem concerning the algorithm.

LEMMA 4.1. *For B a symmetric, negative definite matrix:*

(a) *Q_B is maximized at $\hat{\pi} = \pi_0 - B^{-1}\nabla l(\pi_0)$.*

(b) *$Q_B(\hat{\pi}) = -\nabla l(\pi_0)^T B^{-1}\nabla l(\pi_0)/2 \geq 0$, where the inequality is strict if $\nabla l(\pi_0) \neq 0$.*

PROOF. For (a), compute $\nabla Q_B(\pi) = \nabla l(\pi_0) + B(\pi - \pi_0)$ and equate it to zero. For (b) note that

$$\begin{aligned} Q_B(\hat{\pi}) &= -(B^{-1}\nabla l(\pi_0))^T \nabla l(\pi_0) + (B^{-1}\nabla l(\pi_0))^T B(B^{-1}\nabla l(\pi_0))/2 \\ &= -\nabla l(\pi_0)^T B^{-1}\nabla l(\pi_0) + \nabla l(\pi_0)^T B^{-1}\nabla l(\pi_0)/2 \\ &= -\nabla l(\pi_0)^T B^{-1}\nabla l(\pi_0)/2 \geq 0 , \end{aligned}$$

and the inequality is strict if $\nabla l(\pi_0) \neq 0$. □

THEOREM 4.1. *Let $\pi_0 \in \Pi$ and suppose that $(\pi_j : j = 1, 2, \ldots)$ is defined by the LB algorithm. Under assumption* (4.2) *the sequence (π_j) has the following properties:*

(a) *Monotonicity: $l(\pi_{j+1}) \geq l(\pi_j)$, with $>$ if $\pi_{j+1} \neq \pi_j$.*

(b) *Guaranteed convergence: The sequence $\nabla l(\pi_j)$ converges to 0 if l is bounded above.*

(c) *Rate of convergence: The algorithm converges linearly, with rate*

$$\| \mathbb{1} - B^{-1}\nabla^2 l(\hat{\pi}) \| < 1 .$$

PROOF. To prove (a), let $h = -B^{-1}\nabla l(\pi_j)$. Consider the Taylor-expansion about π_j:

$$\begin{aligned} l(\pi_{j+1}) - l(\pi_j) &= h^T \nabla l(\pi_j) + h^T \nabla^2 l(\pi_0 + \alpha^* h)h/2 \\ &\geq h^T \nabla l(\pi_j) + h^T Bh/2 \quad \text{[using (4.2)]} . \end{aligned} \tag{4.4}$$

Now apply part (b) of Lemma 4.1 to verify monotonicity.

To prove part (b), suppose for purposes of contradiction that $\| \nabla l(\pi_j) \|$ is bounded away from 0. From part (b) of Lemma 4.1, it can be seen that the increments in (4.3) are bounded below, contradicting the boundedness of l.

For part (c), note that because of (4.2) we have $h^T \nabla^2 l(\pi)h \geq h^T Bh$ and thus the Rayleigh-quotient of $\nabla^2 l(\pi)$, namely $h^T \nabla^2 l(\pi)h / h^T h$, is larger than the Rayleigh-quotient of B, $h^T Bh / h^T h$, for every h. Therefore:

$$\lambda = \| B^{-1}\nabla^2 l(\hat{\pi}) \| \leq \| B^{-1} \| \, \| \nabla^2 l(\hat{\pi}) \| = \| \nabla^2 l(\hat{\pi}) \| / \| B \| \leq 1 .$$

Note that λ is the maximal absolute eigenvalue of $B^{-1}\nabla^2 l(\hat{\pi})$. The proof

concludes by noting that $\|\mathbb{I} - B^{-1}\nabla^2 l(\hat{\pi})\| = 1 - \lambda < 1$. □

5. Applications

We look now at two important models of Type I*.

5.1 *Logistic regression*

For each i from 1 to n, let Y_i be a 0-1 variate whose distribution depends on a vector x_i of predictors. Let $p = p(x) \in (0, 1)$, depending on x, represent the probability of obtaining $Y = 1$ from an individual with predictors x. Logistic regression modelling (Pregibon (1981), McCullagh and Nelder (1983), p. 75 and Moolgavkar *et al.* (1985)) assumes that conditionally upon the observed x's, the Y's are independent Bernoulli random variables with respective success probabilities:

$$p_i = p(x_i) = \frac{\exp(x_i^T\pi)}{1 + \exp(x_i^T\pi)}. \tag{5.1}$$

Thus the log-likelihood is

$$\begin{aligned} l(\pi) &= \sum_{i=1}^{n} y_i \log(p_i) + (1 - y_i)\log(1 - p_i) \\ &= \Sigma y_i x_i^T\pi - \Sigma \log(1 + \exp(x_i^T\pi)), \end{aligned}$$

and the score is

$$\begin{aligned} \nabla l(\pi) &= \sum_{i=1}^{n} x_i \left\{ y_i - \frac{\exp(x_i^T\pi)}{1 + \exp(x_i^T\pi)} \right\} \\ &= X^T(Y - \hat{Y}_\pi), \end{aligned}$$

where X is the $n \times p$ design matrix, Y is the vector of observations and $\hat{Y}_\pi$ is the corresponding vector of expected values given x's and the current value of π. Note that the design matrix X is independent of π. The Hessian is given by

$$\nabla^2 l(\pi) = -\sum_{i=1}^{n} x_i x_i^T p_i(1 - p_i) = -X^T \Lambda_\pi X,$$

where Λ_π is diagonal, with (i, i)-th entry $p_i(1 - p_i)$. The Newton-Raphson iteration with initial value π is therefore

$$\pi - \nabla^2 l(\pi)^{-1}\nabla l(\pi) = \pi + (X^T\Lambda_\pi X)^{-1}X^T(Y - \hat{Y}_\pi). \tag{5.2}$$

An algorithm whose steps have the given form are sometimes called *iteratively weighted* (or *reweighted*) *least squares*. Note also that (5.2) coincides with Fisher-scoring since the expected and the sample information matrix are equal.

Since we have $p_j(1-p_j) \leq .25$ for $j = 1,\ldots,n$, a lower-bound for the Hessian is

$$\nabla^2 l(\pi) \geq -X^T X/4 \quad \text{for all} \quad \pi\,. \tag{5.3}$$

Note that this lower-bound is *sharp* since it is attained for $\pi = 0$. We summarize this application of the *LB* algorithm with:

THEOREM 5.1. *In the logistic model the sequence* (π_j) *created by arbitrary* π_0 *and*

$$\pi_{j+1} = \pi_j + 4(X^T X)^{-1} X^T (Y - \hat{Y}_{\pi_j})\,, \tag{5.4}$$

converges monotonically. If Newton-Raphson is started at $\pi_0 = 0$ *the first Newton-Raphson coincides with the lower-bound step.*

To illustrate the algorithm, we consider a data set found in Neter *et al.* ((1985), p. 365). Here x represents the amount of price reduction on a given product, Y indicates the selected household response. The following table compares the Newton-Raphson (5.2) and lower-bound algorithm (5.4) for a logistic model which includes a constant and linear term in x, with corresponding parameters π_1 and π_2. The lower-bound algorithm reaches 4 digits of accuracy in the parameter estimates in the 7th step (3 steps for Newton-Raphson). It is rather interesting to note that 4 digits of accuracy in the likelihood are already reached in the 4th step (2 steps for Newton-Raphson).

We note that if one uses *nonzero* start values for Newton-Raphson in the logistic regression model, one can obtain the spider-web-effect. In particular, in the example of Table 2, if we start Newton-Raphson at $\pi_0^T = (.15, .15)$ instead of $(0,0)$, then the first two iterations yield:

$$\pi_1 = \begin{Bmatrix} 5.067 \\ -.812 \end{Bmatrix}, \quad \pi_2 = \begin{Bmatrix} -.7\text{E}+09 \\ .6\text{E}+08 \end{Bmatrix}.$$

From the same starting value the *LB* algorithm needed 10 steps to reach 4 digits of accuracy.

Which algorithm is better (NR or lower-bound) depends on the overall computational efficiency, which will depend on the size of the matrix to be inverted, the location of the solution and other factors. To gain some

Table 2. A comparison of algorithms (5.2) and (5.4).

Algorithm		Step number					
		0	1	2	3	4	5
(5.2)	π_{1j}	0	−1.9027	−2.1730	−2.1854	−2.1855	—
	π_{2j}	0	.0949	.0970	.1087	.1087	—
	$l(\pi_j)$	−693.15	−585.90	−584.38	−584.38	−584.38	—
(5.4)	π_{1j}	0	−1.9027	−2.0977	−2.1563	−2.1756	−2.1821
	π_{2j}	0	.0949	.1044	.1073	.1082	.1085
	$l(\pi_j)$	−693.15	−585.90	−584.52	−584.39	−584.38	−584.38

information on this issue, a simulation study was conducted.

The design of the study was based on the opinion that the greatest potential value for this procedure is in large scale data snooping projects, where one is hunting for potentially important covariates among a very large set of candidates, possibly over a large number of data sets. In such a setting, most of the "true" values of the π-components would be nearly zero. Consistent with this consideration, for each trial, $n = 300$ observations (Y, x) were generated. Each x-vector consisted of independent uniform $(0, 1)$ variates. The corresponding value of y was Bernoulli, with success probability .5, as determined by the logistic model (5.1) with $\pi = 0$. The number of predictors, p, was varied from 2 to 15, and 10 trials were conducted at each value of p.

In Fig. 6 the two algorithms are compared by showing the CPU time required to attain 4-digit accuracy on a Prime 2250 minicomputer. Although the number of steps required by Newton-Raphson was fairly constant across values of p, the plot shows the greatly increasing cost of the matrix operations. The plot for the *LB* method shows a nonlinear trend corresponding to the single matrix inversion operation required. In *every case* the *LB* method was faster, and the gap increased in p.

Since an important limitation of the study was its focus on $\pi = 0$, which gives $\hat{\pi}$ a tendency to be near zero, an attempt was made to see if the comparison between algorithms depended on the distance of the solution $\hat{\pi}$ from the initial value 0. The tradeoff here is not clear; Newton-Raphson will need more inversions to get there, but the lower-bound will require more iterations as well. Figures 7 and 8 show how the difference (CPU(Newton-Raphson)-CPU(lower-bound)) depended on the standardized distance $\Sigma\hat{\pi}_i^2/p$ of the solution from the initial value 0 for the ten problems generated at each level of p. Least squares lines are plotted to indicate separate trends for each value of p. While there is some sign of downward trend, it appears that for moderate distances beyond the generated range the lower-bound method will remain superior. Moreover, the higher the dimensions, the longer the advantage would stay with the *LB* method.

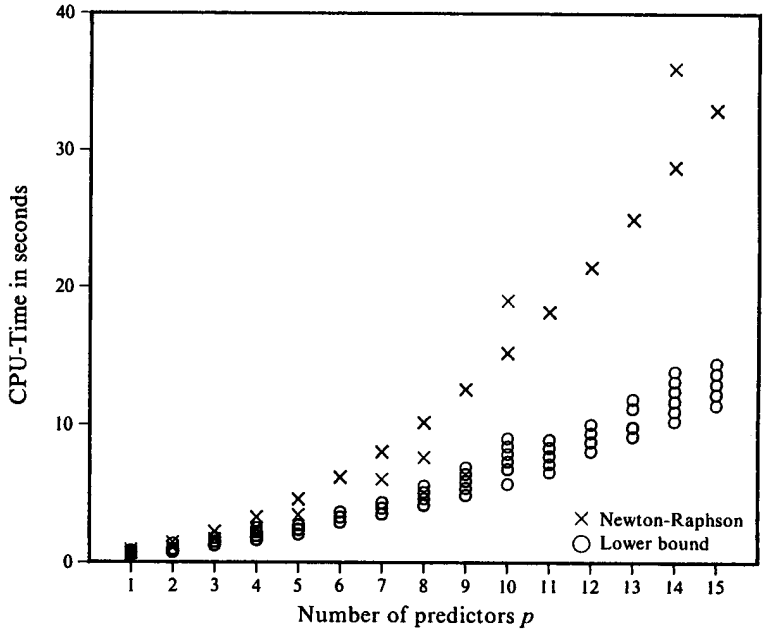

Fig. 6. A comparison of Newton-Raphson and lower-bound methods by Monte Carlo.

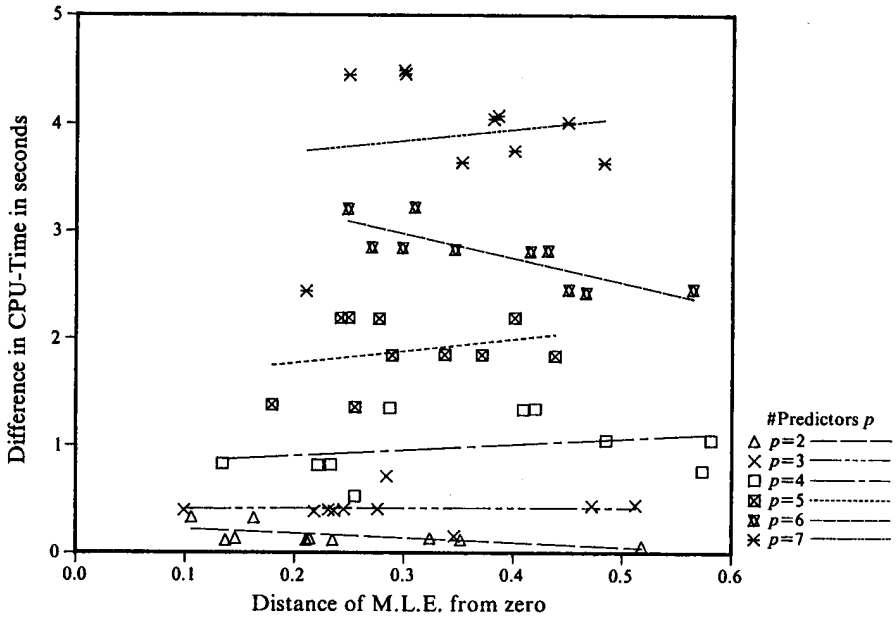

Fig. 7. A comparison of Newton-Raphson and lower-bound methods by Monte Carlo.

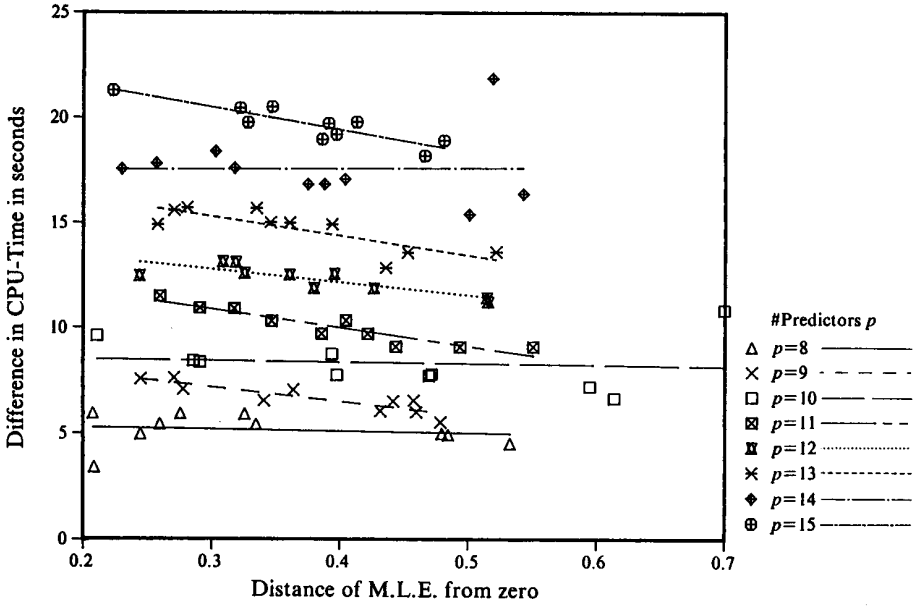

Fig. 8. A comparison of Newton-Raphson and lower-bound methods by Monte Carlo.

5.2 *Cox proportional hazards model*

We adopt here the notation of Miller ((1981), p. 123). The object to be maximized is the *partial likelihood*:

$$\exp[l(\pi)] = \prod_{i=1}^{n} \frac{\exp(x_{(i)}^T \pi)}{\sum_{j \in R_i} \exp(\pi^T x_j)}.$$

Here the product is taken over all uncensored observations and R_i denotes the risk set at the ordered failure time $y_{(i)}$. The Hessian of the log-likelihood can be computed to be

$$\nabla^2 l(\pi) = -\sum_i \left[\sum_{j \in R_i} x_j x_j^T p_j^i - \left[\sum_{j \in R_i} x_j p_j^i \right] \cdot \left[\sum_{j \in R_i} x_j^T p_j^i \right] \right], \tag{5.5}$$

where:

$$p_j^i = \frac{\exp(\pi^T x_j)}{\sum_{j \in R_i} \exp(\pi^T x_j)}.$$

Note first that, for given i, $\{p_j^i\}$ is a probability distribution on $\{x_j | j \in R_i\}$. This implies that $-\nabla^2 l(\pi)$ can be represented as a sum of covariance

matrices and as such is non-negative definite. This implies the concavity of the log-likelihood l.

The question remains to find a lower-bound for $\nabla^2 l(\pi)$ in (5.5). As a simple and quick answer in the spirit of (5.3), we find that replacing all the p_j^i by $1/2$ and deleting the second term in the Hessian will do the job.

THEOREM 5.2.

$$\nabla^2 l(\pi) \geq -\sum_i \sum_{j \in R_i} x_j x_j^T / 2 .$$

PROOF. Consider for arbitrary $h \in \mathbb{R}^P$ the quadratic form:

$$h^T \left[\sum_{j \in R_i} x_j x_j^T p_j^i - \sum_{j \in R_i} x_j p_j^i \sum_{j \in R_i} x_j^T p_j^i \right] h \tag{5.6}$$

$$= \sum_{j \in R_i} (h^T x_j)^2 p_j^i - \left(\sum_{j \in R_i} (h^T x_j) p_j^i \right)^2 .$$

Regarding $\{p_j^i | j \in R_i\}$ as a probability distribution with support points in the set $\{h^T x_j | j \in R_i\}$, then (5.6) is the variance of a random variable U_i which takes on the value $h^T x_j$ with probability p_j^i. Var (U_i) becomes maximal if p_j^i puts mass $1/2$ at the smallest and largest of the $h^T x_j$. Identifying these values of $\{h^T x_j : i \in R_i\}$ by m_i and M_i, we have that (5.6) is dominated by:

$$B_i(h) = \{(m_i)^2 + (M_i)^2 - (m_i + M_i)^2/2\}/2 \tag{5.7}$$

$$\leq \sum_{j \in R_i} h^T x_j x_j^T h / 2 . \tag{5.8}$$

The last step comes from noticing that m_i^2 and M_i^2 are summands in (5.8). The argument is valid for all i and all h. □

However, we note that inequality between (5.7) and (5.8) can be rather crude. In particular, the bound lacks the location invariance in the x's that the Hessian has. The following theorem eliminates this difficulty:

THEOREM 5.3. *Let* $n_i = \#\mathbb{R}_i$. *Then*:

$$\nabla^2 l(\pi) \geq -\sum_i \left\{ \sum_{j \in R_i} x_j x_j^T - \left[\sum_{j \in R_i} x_j \right] \left[\sum_{j \in R_i} x_j^T \right] \Big/ n_i \right\} \Big/ 2 . \tag{5.9}$$

PROOF. Consider a $N \times N$ matrix of the form $M(p) = D(p) - pp^T$, where p^T is a vector of probabilities (p_i) and $D(p)$ is a diagonal matrix with the value p_i in location (i, i). This is the covariance matrix for the multinomial with cell probabilities p_i. In the Bernoulli case $(N = 2)$,

$M(p) \leq M(p^*)$, where $p^{*T} = (.5, .5)$. We seek to generalize this inequality for arbitrary N by choosing $c(N)$ such that $M(p) \leq c(N)M(p^*)$ *for every* p, where p^* represents the uniform multinomial distribution.

This can be accomplished by choosing c to be the supremum of $v^T M(p)v / v^T M(p^*)v$ over all choices of vector v and probability vector p. Maximizing over p first, the argument of Theorem 5.2 indicates the maximum occurs when mass .5 is put at the maximum and minimum of the v_i. Leaving the maximum and minimum coordinates of v fixed, we now maximize over the remaining coordinates of v, which corresponds to minimizing the denominator. This is equivalent to minimizing the variance of a random variable which has fixed mass $1/N$ at specified maximum and minimum—the solution is clearly to put mass $(N-2)/N$ at the midrange. Evaluating this solution yields the optimal $c(N)$: $N/2$ (a second proof could be constructed from the results of Baksalary and Pukelsheim (1985)).

Finally, note that (5.6) has the form:

$$h^T \nabla^2 l(\pi) h = -\sum_i \{v^{i^T} [M(p^i)] v^i\},$$

where $v^i = (h^T x_j : j \in R_i)^T$. The bound of the theorem consists of substituting $n_i M(p^*)/2$ for the term in square brackets of (5.6). □

Note that the bound (5.9) becomes the bound in Theorem 5.2 if n_i becomes large. Moreover, the bound, although the best of its type, is still crude as n gets large: this can be seen by considering the starting value $\pi = 0$, which generates the uniform distribution on the p_j^i, and comparing the Hessian there with our lower-bound. In this problem there is an inherent loss of sharpness in using a single bound because the curvature can vary sharply as a function of the initial value π_0 and the direction h. Because of this, we now turn to simple algorithmic improvement which utilizes the sharper bound in (5.7). Assuming π_0 and direction h are given, it provides a curvature bound along the line $\pi_0 + \alpha h$, $\alpha \in \mathbb{R}$.

Step 1. Compute the lower-bound step $h = -B^{-1} \nabla l(\pi)$, where B is the lower-bound found in (5.9). (One could choose any direction of increase h for this step. This one has the feature that the inversion of B need be carried out only in the initial pass through the algorithm, and that it is guaranteed to be a direction of increase.)

Step 2. Consider $\pi + \alpha h$ with scalar α and note that

$$l(\pi + \alpha h) - l(\pi) \geq \alpha h^T \nabla l(\pi) + \alpha^2 b_{new}/2,$$

when the lower-bound $b_{new} = \Sigma B_i(h)$ is determined as in (5.7) from the minimum and maximum of $\{h^T x_j\}$. Solving for α gives an extended step in the h-direction:

$$\hat{a} = -h^T \nabla l(\pi)/b_{new} .$$

(This step is necessarily monotonic.)

6. Monotonic algorithms in Type II models

Returning now to Type II models, with double concavity, we note that even though the Newton-Raphson convergence holds, there are some good arguments for considering simple fixups to the NR procedure which will guarantee monotonicity. The most important aspect is speed and reliability of convergence in high dimensional problems, where each step is costly. We will discuss the multivariate extension of the Type II model after we present two modified NR algorithms which are strictly monotonic in the univariate Type II model; their behavior in a particular data set was demonstrated in Table 1.

6.1 *The adaptive lower-bound algorithm*

Step 1. Given current value π_0, construct Newton-Raphson step π_{nr}.

Step 2. Let $b = b(\pi_0, \pi_{nr})$ be a lower-bound for $\nabla^2 l(\pi)$ for π between π_0 and π_{nr}. The adjusted step is:

$$\pi_{alb} = \pi_0 - \nabla l(\pi_0)/b . \tag{6.1}$$

The applicability of the ALB algorithm depends on one's ability to construct the bound b. In the doubly concave case, one only needs to compare the Hessians at each endpoint of the interval and use the smaller one. If the initial Hessian is the smallest, then the step is the Newton-Raphson step; if not, then one makes a backstep. This algorithm is easily shown to be quadratically convergent.

For very little additional cost, one can construct a monotonic algorithm which is superior in convergence. In the above we have used the concavity of $\nabla^2 l(\pi)$ in a crude way: its minimum value on an interval $[a, b]$ occurs at the endpoints. However, given its values at the endpoints, we can in fact construct the stronger inequality:

$$\nabla^2 l(\bar{\alpha}a + \alpha b) \geq \bar{\alpha}\nabla^2 l(a) + \alpha \nabla^2 l(b) \quad \text{for} \quad \alpha = 1 - \bar{\alpha} \in [0, 1] . \tag{6.2}$$

Consider the following cubic approximation to $l(\pi) - l(\pi_0)$ for π in the interval between π_0 and π_{nr}:

$$C(\pi) = (\pi - \pi_0)\nabla l(\pi_0) + (\pi - \pi_0)^2 \nabla^2 l(\pi_0)/2 + (\pi - \pi_0)^3 A/3! . \tag{6.3}$$

We set A equal to $[\nabla^2 l(\pi_{nr}) - \nabla^2 l(\pi_0)]/(\pi_{nr} - \pi_0)$, a secant approximation to the third derivative. Then C has the same gradient at π_0 as $l(\pi) - l(\pi_0)$ but,

by differentiating twice and applying (6.4), one can show that everywhere on the interval from π_0 to $\pi_1^* C$ has greater curvature and so lies beneath this objective function. Hence at its maximum value *on that interval* we are guaranteed that the objective function has increased. Incorporating this refined approximation gives the following.

6.2 *The cubic adaptive lower-bound algorithm*

Compute π_{nr}. If the Hessian has increased, the step is saved. Otherwise π_{nr} is replaced by the value maximizing (6.5):

$$\pi_{calb} = \pi_0 + \left[-\nabla^2 l(\pi_0) - \sqrt{(\nabla^2 l(\pi_0))^2 - 2\nabla l(\pi_0) A} \right] / A \,. \tag{6.4}$$

Note that the enhanced cubic algorithm (6.4), once it enters a region of declining curvature (the region of overstep), estimates area by trapezoidal regions, replacing the curve $\nabla^2 l(\pi)$ with a line segment between endpoints.

There are a number of important statistical models with multidimensional π which have the property that when the likelihood is viewed along a line $l^*(\alpha) := l(\pi_0 + \alpha h)$, it is doubly concave in α. Although in such a *Type II** likelihood it is therefore true that there can be at most one bad step along any line, this is no guarantee of convergence, as the Newton-Raphson procedure continually changes direction.

It is, however, a simple matter to fix up the Newton-Raphson algorithm so that it is monotonic.

6.3 *Multivariate adaptive lower-bound algorithm*

For initial value π_0, define

$$\varphi(\alpha) := l(\pi_0 + \alpha h) \,,$$

where h represents a selected vector known to be a direction of increase to the objective function, such as the gradient, conjugate gradient, or Newton-Raphson direction. Take one step along the line $\pi_0 + \alpha h$, based on $\varphi(\alpha)$, using one of the two adaptive lower-bound algorithms.

We conclude by describing two examples with Type II* structure.

Example B. (Mixture of densities) We consider the log-likelihood generated by a finite mixture of known densities:

$$l(\pi) = \sum_{j=1}^{n} \log \left(\sum_{i=1}^{p} \pi_i f_i(x_j) \right) . \tag{6.5}$$

For an arbitrary vector h we have

$$\varphi(\alpha) = \sum_{j=1}^{n} \log (g_j + \alpha \Delta_j) ,$$

where $g_j = \Sigma \pi_i f_i(x_j)$ and $\Delta_j = \Sigma h_i f_i(x_j)$, which is readily shown to be doubly concave.

The adaptive lower-bound may be of particular interest in the case of mixtures since frequently the flat log-likelihood leads to convergence problems. Titterington *et al.* ((1985), p. 89) report a simulation study where the Newton-Raphson algorithm failed in about 50% of the cases even though the algorithm was started from the true parameter values. The EM-algorithm converged monotonically, but was dreadfully slow. Similar problems are mentioned and discussed in Everitt and Hand ((1981), p. 38).

Example C. (Log-linear model) Suppose that the dependent variable Y, conditionally upon a vector of covariates x, follows a Poisson distribution with mean value $E(Y) = \exp (x^T \pi)$. That is, in the terminology of McCullagh and Nelder (1983), the model is linear in the natural link function $\log (\mu) = \eta$. Here the log-likelihood is proportional to:

$$l(\pi) = \sum_{j=1}^{n} [y_j x_j^T \pi - \exp (x_j^T \pi)] . \tag{6.6}$$

And so we have

$$\nabla^2 \varphi(\alpha) = - \Sigma (x_j^T h)^2 \exp (x_j^T (\pi + \alpha h)) ,$$

and

$$\nabla^4 \varphi(\alpha) = - \Sigma (x_j^T h)^4 \exp (x_j^T (\pi + \alpha h)) ,$$

as required.

Acknowledgements

The first author wishes to thank Professor Bill Harkness and the Pennsylvania State Department of Statistics for their support during his visit there.

REFERENCES

Andersen, E. B. (1980). *Discrete Statistical Models with Social Science Applications*, North-Holland, Amsterdam.

Baksalary, J. K. and Pukelsheim, F. (1985). A note on the matrix ordering of special C-matrices, *Linear Algebra Appl.*, **70**, 263–267.

Baum, L. E. and Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology, *Bull. Amer. Math. Soc.*, **73**, 360–363.

Collatz, L. (1961). Monotonie und Extremal Prinzipien beim Newtonschen Verfahren, *Numer. Math.*, **3**, 99–106.

Cox, D. R. and Oakes, D. (1984). *Analysis of Survival Data*, Chapman & Hall, London-New York.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood estimation from incomplete data via the EM algorithm (with discussion), *J. Roy. Statist. Soc. Ser. B*, **39**, 1–38.

Everitt, B. S. and Hand, D. B. (1981). *Finite Mixture Distributions*, Chapman & Hall, London-New York.

Horst, R. (1979). *Nichtlineare Optimierung*, Carl Hanser, Munchen-Wien.

Marshall, A. W. and Olkin, I. (1979). *Inequalities: Theory of Majorization and Its Applications*, Academic Press, New York.

McCullagh, P. and Nelder, J. A. (1983). *Generalized Linear Models*, Chapman & Hall, London-New York.

Miller, R. G. (1981). *Survival Analysis*, Wiley, New York.

Moolgavkar, S. H., Lustbader, E. D. and Venson, D. J. (1985). Assessing the adequacy of the logistic regression model for matched case-control studies, *Statistics in Medicine*, **4**, 425–435.

Neter, J., Wasserman, W. and Kutner, M. (1985). *Applied Linear Statistical Models*, 2nd ed., Homewood, Irwin.

Potra, F. A. and Rheinboldt, W. C. (1986). On the monotone convergence of Newton's method, *Computing*, **36**, 81–90.

Pregibon, D. (1981). Logistic regression diagnostics, *Ann. Statist.*, **9**, 705–724.

Sundberg, R. (1976). An iterative method for solution of the likelihood equations for incomplete data from exponential families, *Comm. Statist. B—Simulation Comput.*, **5**, 55–64.

Titterington, D. M., Smith, A. F. M. and Makov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions*, Wiley, Chichester-New York.

Wu, C. F. (1983). On the convergence properties of the EM algorithm, *Ann. Statist.*, **11**, 95–103.