A COMPARISON OF THE STEEPEST DESCENT AND NEWTON-RAPHSON ALGORITHMS IN THE LINEAR LOGISTIC TEST MODEL

Fons J.R. van de Vijver

Abstract

In the present Monte Carlo study three methods for the estimation of the basic parameters of the Linear logistic Test Model were compared, viz. Steepest Descent iterations, a Newton-Raphson procedure and a combination of both. It was found that the Newton-Raphson procedure needed the least computer time to reach convergence. As, additionally, this procedure easily allows for the computation of standard errors of the estimates, it was concluded that this method should be preferred.

**Department of Social Sciences
Tilburg University
P.O. Box 90153
5000 LE Tilburg
The Netherlands
Tel.: 013-662528**

In the Rasch model the probability that subject v $(v = 1, \ldots, n)$ with ability θ_v will solve item i $(i = 1, \ldots, k)$ with item difficulty b_i correctly is given by:

$$P_{vi} = \frac{exp(\theta_v - b_i)}{1 + exp(\theta_v - b_i)} \tag{1}$$

The estimation equation of the maximum likelihood estimation procedure is (cf. Fischer, 1974, [14.1.8]):

$$a_{0\alpha} = \sum_{r=1}^{k} \frac{n_r \epsilon_\alpha \gamma_{r-1}^{(\alpha)}}{\gamma_r} \tag{2}$$

in which

- $a_{0\alpha}$ = the number of correct answers at item α $(\alpha = 1, \ldots, k)$;
- n_r = vector representing the number of subjects with score r;
- $\epsilon_\alpha = -ln(b_\alpha)$, i.e., the item easiness of item *alpha*;
- $\gamma_{r-1}^{(\alpha)}$ = first derivative of the elementary symmetric function of order r without item α;
- γ_r = elementary symmetric function of the order r.

From a computational viewpoint the most problematic aspect of (2) concerns the elementary symmetric functions. Two recursive sets of formulas for the computation of the elementary symmetric functions and their derivatives have been described by Fischer (1974, [14.3.3-14.3.4] and [14.3.11]). The first procedure, called the 'Difference Algorithm' by Gustafsson (1980, p. 381) is fast but will become inaccurate when applied to a large number of items, whereas the second procedure, called the 'Summation Algorithm' (Gustafsson, 1980, p. 382) is slower but numerically more stable. These numerical problems in the computations of the elementary symmetric functions also trouble the Linear Logistic Test Model (LLTM).

In this model the item difficulty parameter of equation (1) is linearly decomposed in:

$$b_i = \sum_{j=1}^{m} q_{ij} \eta_j + c \tag{3}$$

in which

- η_j = the basic parameter of the LLTM $(j = 1, \ldots, m)$ and $m < k$ (Fischer, 1974, 1983; cf. also Formann, 1984);
- q_{ij} = an element of the design matrix Q indicating the number of times operation j is presumably invoked in the solution of item i. The matrix Q has to be specified by the investigator prior to the data analysis;
- c = an arbitrary constant.

The estimation equations for the LLTM are given by (cf. Fischer, 1974, [17.2.11a] and Fischer, 1983, the final equation at page 6):

$$\sum_{i=1}^{k} q_{i\alpha}(s_i - \sum_{r} \frac{n_r \epsilon_i \gamma_{r-1}^{(i)}}{\gamma_r}) = 0 \; for \; \alpha = 1, \ldots, m \tag{4}$$

in which

- s_i = the sufficient statistic of the item parameter, i.e., the number of correct responses at item i.

For the solution of this set of equations the first derivative of logarithm of the likelihood L of the data matrix with respect to the basic parameter η_α $(\alpha = 1, \ldots, m)$ is needed in some procedures. The first derivative is given by (cf. Fischer, 1983, p. 6):

$$\frac{d \, lnL}{d \, \eta_\alpha} = s_\alpha - \sum_r n_r \gamma_r^{-1} \sum_i \gamma_{r-1}^{(i)} \epsilon_i q_{i\alpha} \tag{5}$$

The second derivative of the logarithms of L with respect to η_α $(\alpha = 1, \ldots, m)$ and η_β $(\beta = 1, \ldots, m)$ is:

$$\frac{d^2 \, lnL}{d \, \eta_\alpha \eta_\beta} = -\sum_i q_{i\alpha} q_{i\beta} \sum_r \frac{\epsilon_i \gamma_{r-1}^{(i)}}{\gamma_r} + \sum_i \sum_l q_{i\alpha} q_{l\beta} \sum_v \epsilon_i \epsilon_l \frac{\gamma_{r-1}^{(i)} \gamma_{r-1}^{(l)} - \gamma_r \gamma_{r-2}^{(i,l)}}{\gamma_r^2} \tag{6}$$

in which

- $\gamma_{r-2}^{(i,l)}$ = the second derivative of the elementary symmetric functions without the items i and l.

Two solution procedures are typically used to solve (4). The first is a *Steepest Descent Method*, described by Fischer (1974, p. 255), who also gives a computer program. Essentially the method consists of the following steps:

1. Choose a vector of initial estimates of the basic parameters $\eta_1, \eta_2, \ldots, \eta_m$ and a step size Ω.
2. Compute the derivatives of the log-likelihood of the data matrix for the estimated values of the basic parameters according to (4) for $\alpha = 1, \ldots, m$. Store the result in vector f.
3. Normalize f and compute the descent for f.
4. Compute the descent for the step size.
5. Compute (by means of linear interpolation) a new step size Ω_p (the index p refers to the number of the iteration):

$$\Omega_{p+1} = \frac{\Omega_{p-1} f(\Omega_p) - \Omega_p f(\Omega_{p-1})}{f(\Omega_p) - f(\Omega_{p-1})} \tag{7}$$

This is repeated until the change in step size becomes smaller than some arbitrary value. The step size obtained is designated Ω_∞.

6. A new η-vector is computed from

$$\eta_p = \eta_{p-1} + \Omega_\infty f(\eta_{p-1}) \tag{8}$$

7. The procedure is repeated from step 1 onwards until the absolute values of the derivatives of all parameter estimates are smaller than some arbitrary value.

An assured convergence, even with poor initial estimates, and a low computational load per iteration can be mentioned as the advantages of this procedure. During the first iterations large improvements of the estimates are typically found. However, when the estimates come nearer to the final maximum likelihood estimates, the rate of convergence often becomes painstakingly slow.

The second method is the *Newton-Raphson procedure* (cf. Fischer, 1974, [14.4.12]). The general structure of this procedure is rather simple:

> For a set of estimates of the basic parameters the first and second drivatives are computed according to (5) and (6). A vector $\Delta\eta$ is then computed by means of:
>
> $$\begin{pmatrix} \Delta\eta_1 \\ \vdots \\ \Delta\eta_m \end{pmatrix} = - \begin{pmatrix} \frac{d^2\, lnL}{d\,\eta_1^2} & \dots & \frac{d^2\, lnL}{d\,\eta_1\eta_m} \\ \vdots & \ddots & \vdots \\ \frac{d^2\, lnL}{d\,\eta_m\eta_1} & \dots & \frac{d^2\, lnL}{d\,\eta_m^2} \end{pmatrix}^{-1} \begin{pmatrix} \frac{d\, lnL}{d\,\eta_1} \\ \vdots \\ \frac{d\, lnL}{d\,\eta_m} \end{pmatrix} \tag{9}$$
>
> The values of $\Delta\eta$ are added to the parameter estimates of the previous iteration, thereby constituting the parameter estimates of the following iteration. This procedure is repeated until the absolute value of the changes in all estimates from one iteration to the next are smaller than some arbitrary value.

The iterative procedure is rather involved from a computational viewpoint; it entails the computation of the second derivatives of the elementary symmetric functions and of the inversion of this matrix. Per iteration the computational load is much larger than in the Steepest Descent Method. An advantage of this procedure is the small number of iterations needed to reach convergence. Also, on the basis of the inverted matrix of the second derivates confidence intervals of the parameter estimates can be easily computed.

In the present Monte Carlo investigation the behaviour of both the gradient method and the Newton-Raphson method will be studied. Additionally, a combined procedure is developed, which will be called the *Combination Method.* In the first part of this procedure the Steepest Descent METHOD is used until the absolute value of the gradient in all parameters to be estimated becomes smaller than some arbitrary, relatively large value. These parameters are then the starting values of a Newton-Raphson procedure, which is continued until convergence is reached. By doing so, an attempt is made to put the substantial improvements during the first iterations of the gradient method to an advantage, i.c. to speed up the convergence of the Newton-Raphson procedure.

Method

The computations were carried out on a VAX 11/785 (Digital Equipment Corporation). The structure of the program written in FORTRAN-77, to generate and analyze the data was as follows:

1. **Generating the data.** A vector of standard normally distributed item difficulties and person abilities were generated. A matrix $P\ (n,k)$ was computed containing the probabilities at a correct answer under the Rasch model. Then a matrix R with the same dimensions was generated containing uniformly distributed random numbers at the interval (0,1). Finally, a dichotomous data matrix D, again with the same dimensions, was composed with elements d_{ij}; d_{ij} was equal to zero if p_{ij} was smaller than r_{ij} and d_{ij} was equal to one, otherwise. If the matrix D contained any row or vector with only zeros or only ones, the data generating restarted from the beginning.

2. **Generating the design matrix.** First, a design matrix (k,m) was filled with uniformly distributed random numbers on the interval (0,1). The entries of the matrix were then dichotomized; if the entry was greater than a constant (0.40 throughout the study), it was set equal to 1; otherwise, a value of 0 was assigned. The generated matrix was checked for allowance of unique maximum likelihood estimates (cf. Fischer, 1983). If the conditions for uniqueness were not met, step 2 was repeated.

3. **The analysis according to the LLTM.** A program has been developed based on the computer program described by Fischer (1974, pp. 538-548). For the computation of the basic symmetric functions and their first derivatives the algorithms mentioned previously were used. For small item numbers (up to 10) the 'Difference Algorithm' (Gustafsson, o.c., p. 381) was used, while for larger numbers of items the 'Summation Algorithm' was invoked. The three estimation procedures, viz. the Steepest Descent, the Newton-Raphson and the Combination Method, were then used to estimate the parameters of the LLTM.
   In the Combination method the Steepest Descent Method was maintained until the absolute values of the derivatives of all parameter estimates were smaller than 0.5, after which Newton-Raphson iterations were started. The number of iterations and the CPU-time needed by each of these were recorded. The CPU-time was determined by means of Run Time Library Routines, which are accurate in two decimals.
   The initial estimates were derived from the sufficent statistics of the basic parameters; the 'p-value' of each operation, the classical difficulty index, was the starting value of each iterative procedure.

In the present Monte Carlo study three different test lengths were used, namely 10, 20 and 40 items. In each of these structure matrices the number of basic parameters to be estimated was systematically varied from $2,4,\ldots,k-2$; that is, for each test length $k/2-1$ different number basic parameters were estimated. A sample size of 500 subjects was maintained throughout. For each combination of the number of

	Test Length								
	10 items			20 items			40 items		
Number of parameters	SD [1]	CM	NR	SD	CM	NR	SD	CM	NR
2	0.8	0.5	0.2	4.9	3.4	1.7	11.8	17.8	18.3
4	2.6	1.5	0.3	8.7	5.3	2.4	25.0	21.2	18.6
6	21.6	9.6	0.5	11.2	6.2	2.6	42.3	34.7	25.6
8	41.5	17.5	0.8	25.7	13.5	3.1	52.6	36.2	30.1
10				50.0	22.1	3.6	77.0	45.2	27.8
12				77.3	35.4	4.3	97.3	57.1	29.6
14				201.4	78.0	6.6	175.9	89.7	35.4
16				215.9	111.6	9.4	292.3	155.5	37.9
18				268.2	164.9	24.1	468.5	228.0	76.5
20							243.1	118.1	44.4
22							377.9	181.5	48.6
24							800.3	369.2	52.9
26							1186.0	526.0	78.2
28							2232.0	1145.6	146.0
30							1811.8	794.6	105.9
32							2782.1	2117.4	173.3
34							2252.7	1606.0	179.4
36							2996.5	2167.3	211.0
38							1520.9	1139.0	387.1

Table 1: CPU Time Needed to Reach Convergence (in sec.)

basic parameters and items new data and design matrices were generated. Three independent replications were carried out for each combination.

The number of basic parameters was not set equal to the number of items - i.e., when the design matrix of the LLTM is an identity matrix and the LLTM is the Rasch model - as the efficiency of algorithms in the Rasch model is dealt with elsewhere (e.g., Gustafsson, 1980; Wainer, Morgan & Gustafsson, 1980).

Results and Discussion

In Table 1 the CPU time is given which was needed to reach convergence in each of the experimental conditions. The most remarkable and clear-cut result from this Table is the overall superiority of the Newton-Raphson procedure; the Steepest Descent Method nearly always consumes the most CPU time. The difference in CPU time is most pronounced in small data sets, in which the Newton-Raphson procedure can be ten times faster than the Steepest Descent Method. The difference in performance of the iterative procedures is also affected by the number of parameters to be estimated. The superiority of the Newton-Raphson method is most pronounced with a large number of parameters to be estimated. The results of the Combination Method always fell

[1] SD = Steepest Descent Method; CM = Combination Method; NR = Newton Raphson Method

between the results of the two other procedures.

It could be argued that the generalizability of the present study is limited for two reasons. First, in the Combination Method the switch from Steepest Descent to Newton-Raphson iterations was always made when the absolute values of the derivatives of all parameter estimates were less than 0.5. It is obvious that other values of the 'switch parameter' would lead to different results. In fact, provided the present outcomes, a substantially larger value of this parameter should be preferable. Even for these values, however, it remains doubtful whether the Combination Method would be able to outperform the Newton-Raphson procedure. From the present results it is clear that in a combined algorithm the Steepest Descent Method should only be used for a few iterations, before the computations should switch to Newton-Raphson iterations.

Second, in the Steepest Descent Method the step size was always fixed in the first iteration. A value of 3, such as used here, can be a less than optimal choice. If the step size is taken too large, the parameter estimates will oscillate around their true values; if the step size is taken too small the improvement of the estimates from one iteration to the other will be marginal. In both cases too many iterations will be needed. There is no rationale for the choice of the initial step size, which substantially hampers the suitability of the method.

The major reason for the weak performance of the Steepest Descent algorithm is the poor estimation of the derivatives of the parameters. Although the likelihood of the data matrix increases during subsequent iterations, the accuracy of the separate parameter estimates does not always increase.

It has been found in the Rasch model, which is an LLTM with an identity matrix as the structure matrix, that the convergence can be speeded up considerably by Aitken extrapolations (cf. Fischer, 1974; Gustafsson, 1980). Although the extrapolation method can also be used in the LLTM, its feasability will be limited. On the one hand, in the Steepest Descent Method the estimates often behave irregularly which will prohibit an adequate extrapolation; in fact, any extrapolation will involve a real danger of divergence of the estimates. On the other hand, in the Newton-Raphson procedure the number of iterations is usually small, thereby rendering the feasability of extrapolations.

It can be concluded form the present study that there is little reason to use the Steepest Descent Method. In the estimation of the basic parameters, the Newton-Raphson method appears to offer two advantages; first, the method reaches convergence relatively quickly and second, on the basis of its results confidence intervals of the estimated parameters can be easily computed.

References

Fischer, G.H. (1974). *Einführung in die Theorie psychologischer Tests.* Bern: Huber.

Fischer, G.H. (1983). Logistic latent trait models with linear constraints. *Psychometrika, 48*, 3-26.

Formann, A.K. (1984). *Die latent-Class Analysis.* Weinheim: Beltz.

Gustafsson, J. (1980). A solution of the conditional estimation problems for long tests in the Rasch model for dichotomous items. *Educational and Psychological Measurement*, *40*, 377-385.

Wainer, H., Morgan, A., & Gustafsson, J. (1980). A review of estimation procedures for the Rasch model with an eye toward longish tests. *Journal of Educational Statistics*, *5*, 35-64.

Ontvangen: 22-05-1988
Geaccepteerd: 12-09-1988