Newton-Type Algorithms for the Estimation of Item Response Theory Model

Xinming An[*] Yiu-Fai Yung[†]

Abstract
In the field of item response theory, G-H quadrature based EM algorithm proposed by Bock and Aitkin (1981) has been widely recognized as the gold standard for model estimation because of its several appealing properties. However, these advantages are overshadowed by a number of important issues that has not been resolved successfully. Furthermore, recent developments in item factor analysis, for example, confirmatory analysis, also impair EM's advantages. On the other hand, Newton algorithms do not suffer these problems, but are computationally more expensive than EM. During the last twenty years, statistical researches have been impacted dramatically by the advances in computational sciences. Thus it is worthwhile to apply these computational advances to Newton and EM type algorithms and re-evaluate their relative advantages. To this end, the focus of this research is to (1) introduce some Newton type algorithms for item factor analysis; and (2) investigate the computational properties of these Newton type algorithms as compared with the EM algorithm.

Key Words: Item response theory, Newton algorithm, EM algorithm

1. Introduction

Item response theory (IRT) was first proposed in the field of psychometric for the purpose of educational testing and personality assessment. During the last ten years, it has became increasing popular in other fields, such as health behavior and health policy research. The most widely used estimation method for IRT model is the Gauss-Hermite quadrature based EM algorithm proposed by Bock and Aitkin (1981). Because of its several appealing properties, it has become the gold standard and the most popular method used by all the major IRT packages, such as BILOG and TESTFACT. However, these advantages are overshadowed by a number of important issues, among which slow convergence rate and the lack of standard error estimates and reliable convergence criteria are the most serious. While several attempts, such as the SEM algorithm (Meng and Rubin 1991), have been made, these problems have not been well addressed. The convergence rate of EM algorithm is linear at most and in practice it is often much slower, especially when the fraction of missing information is large. The convergence of EM algorithm is monitored by the biggest parameter change after each iteration which is not reliable, since small parameter change can also be attributed to slow convergence rate instead of convergence. Without a reliable convergence criteria, estimates could be seriously biased because of spurious convergence. On the other hand, standard error plays an important role in testing whether a parameter is significantly different from zero. In comparison, the convergence rates of Newton type algorithms are quadratic or super linear, and gradient based convergence criteria and standard errors are readily available . As a result, Newton type algorithms rather than EM algorithms are often used by major statistical packages, such as SAS and STATA. While a Newton type algorithm was proposed by Bock and Lieberman (1970), it has been ignored since the EM algorithms was introduced, because it is computationally more expensive than the EM algorithm.

[*] SAS Institute, 100 Research Dr , Cary, NC 27513
[†] SAS Institute, 100 Research Dr , Cary, NC 27513

During the last twenty years, statistical researches have been impacted dramatically by the advances in computational sciences. Tasks that used to be difficult, such as big matrix inverse, nowadays may become very easy. Several computationally efficient Newton type algorithms, such as Quasi-Newton, have been proposed and widely used in many different areas, but have not yet been applied for IRT. Furthermore, recent developments in IRT, for example, confirmatory analysis, also impair EM's advantages. Thus it is valuable to re-evaluate their relative advantages. To this end, the focuses of this research are to (1) introduce some Newton type algorithms for the estimation of IRT; and (2) investigate the computational properties of these algorithms. The purpose of this paper is to raise attentions of recent advances in computational tools that are potentially useful for IRT model estimation.

The rest of the paper is organized as follows. First, one dimensional IRT model with binary responses is presented in section 2 for illustration purpose. In section 3, the EM algorithm and several Newton type algorithms are introduced. The computational properties of these algorithms are investigated in section 4. The paper concludes with some remarks on future researches.

2. Model Specification

Investigations of the computational properties of different estimation algorithms in this paper are based on the one dimensional IRT model with binary responses, which can be expressed by the following equations.

$$y_i = \Lambda\eta_i + \epsilon_i \tag{1}$$

$$P(u_{ij} = 1) = P(y_{ij} > \alpha_j) \tag{2}$$

where u_{ij} is the observed binary response from subject i for item j, y_{ij} is a continuous latent response underlying u_{ij}, $\alpha = (\alpha_1, \dots, \alpha_J)$ is a vector of the difficulty (or threshold) parameters, Λ is a matrix of the slop (or discrimination) parameters, η_i and ϵ_i are the latent factor and unique factor for subject i, and $\eta_i \sim N(0, I)$, $\epsilon_i \sim N_p(0, I)$ or $L_p(0, I)$, and $\eta_i \perp \epsilon_i$. Based on the above model specification, we have

$$P_{ij} = P(u_{ij} = 1) = P(y_{ij} > \alpha_j) = \int_{\alpha_j - \lambda_j\eta_i}^{\infty} f(y; 0, 1)dy, \tag{3}$$

where $f(y; 0, 1)$ is the density function of normal or logistic distribution with mean 0 and variance 1. To simplify notations, let $Q_{ij} = 1 - P_{ij}$ and $v_{ij} = 1 - u_{ij}$.

3. Model Estimation

One of the most popular estimation methods for latent variable models with categorical responses is based on the marginal likelihood. Parameter estimates can be obtained by maximizing the marginal likelihood using either EM or Newton type algorithms.

3.1 EM algorithm

The EM algorithm starts from the complete data log likelihood that can be expressed as follows

$$\begin{aligned} logL(\theta|u, \eta) &= \sum_{i=1}^{N}\left[\left(\sum_{j=1}^{J} u_{ij}logP_{ij} + (v_{ij})log(Q_{ij})\right) + log\phi(\eta_i)\right] \\ &\propto \sum_{j=1}^{J}\sum_{i=1}^{N}\left[u_{ij}logP_{ij} + (v_{ij})log(Q_{ij})\right] \end{aligned} \tag{4}$$

where $\phi(\eta_i)$ is the prior distribution for latent factor η_i.

In the E step, we calculate the expectation of the complete data log likelihood with respect to the conditional distribution of η, $f(\eta_i|u_i, \theta^{(t)})$,

$$f(\eta_i|u_i, \theta^{(t)}) = \frac{f(u_i|\eta, \theta^{(t)})\phi(\eta)}{\int f(u_i|\eta, \theta^{(t)})\phi(\eta)d\eta} = \frac{f(u_i|\eta, \theta^{(t)})\phi(\eta)}{f(u_i)}. \tag{5}$$

Let $Q(\theta|\theta^{(t)})$ denote the conditional expectation of the complete data log likelihood, and we have

$$Q(\theta|\theta^{(t)}) = \sum_{j=1}^{J}\sum_{i=1}^{N}\left[u_{ij}E\left[logP_{ij}|u_i, \theta^{(t)}\right] + (v_{ij})E\left[log(Q_{ij})|u_i, \theta^{(t)}\right]\right] = \sum_{j=1}^{J} Q_j. \tag{6}$$

Expectations involved in the above equation are often approximated with either numerical or Monte Carlo integration. Let $\tilde{Q}(\theta|\theta^{(t)})$ denote the approximated conditional expectation of the complete data log likelihood.

In the M step of the EM algorithm, parameters are updated by maximizing $\tilde{Q}(\theta|\theta^{(t)})$. To summarize, the EM algorithm consists the following two steps

E Step: Approximate $Q(\theta|\theta^{(t)})$ with either numerical or Monte Carlo integration;

M Step: Update parameter estimates by maximizing $\tilde{Q}(\theta|\theta^{(t)})$ with one step Newton-Raphson algorithm.

Technical details about the EM algorithm are provided in appendix A.

3.2 Newton Type Algorithms

Compared with the EM algorithms which start from the complete data log likelihood, Newton type algorithms maximize the marginal log likelihood directly. Based on the model specified in the last section, the marginal likelihood is

$$L(\theta|U) = \prod_{i=1}^{N}\int\prod_{j=1}^{J}(P_{ij})^{u_{ij}}(Q_{ij})^{v_{ij}}\phi(\eta)d\eta \tag{7}$$

where $\phi(\eta)$ is the density function for latent factor η. The corresponding log likelihood is

$$LogL(\theta|U) = \sum_{i=1}^{N} logL_i = \sum_{i=1}^{N} log\int\prod_{j=1}^{J}(P_{ij})^{u_{ij}}(1-P_{ij})^{1-u_{ij}}g(\eta)d\eta \tag{8}$$

Similar to the EM algorithm, integrations involved in the above equation are often approximated with either numerical or Monte Carlo integration.

Let $Log\tilde{L}(\theta|U)$ denote the approximated marginal log likelihood. Parameter estimates can be obtained by maximizing $Log\tilde{L}(\theta|U)$ with Newton type algorithms. Two of the most widely used estimation algorithms are Newton-Raphson and Fisher Scoring which rely on the gradient and Hessian of the log likelihood. However, for latent variable models with categorical responses, the Hessian matrix is often expensive to compute. As a result, several Quasi-Newton algorithms only requiring gradients have been proposed. In the field of IRT, Bock and Lieberman (1970) proposed replacing the Hessian with the following information matrix

$$I(\theta) = E\left[\frac{\partial Log\tilde{L}(\theta|U)}{\partial\theta}\left(\frac{\partial Log\tilde{L}(\theta|U)}{\partial\theta}\right)^T\right] = \sum_{h=1}^{2^J}\left[\frac{\partial log\tilde{L}_i}{\partial\theta}\left(\frac{\partial log\tilde{L}_i}{\partial\theta}\right)^T.\right] \tag{9}$$

To calculate the above expectation, we need to sum over not just the observed but all 2^J possible response patterns which will become computationally very intensive when the number of item is large. Fortunately, other Quasi-Newton algorithms that do not suffer this computational difficulty have been proposed but have not been used for IRT. Notable examples include Broyden-Fletcher-Goldfarb-Shanno (BFGS) method, one of the most popular Quasi-Newton algorithms that approximate the Hessian matrix with gradient. It is a general algorithm that does not rely on any statistical properties and its usages are far beyond statistics. The second one is proposed by Berndt et al. (1974) which replaces the expectation in equation (7) with summations runs over only the observed response patterns. The accuracy of this algorithm depends on two statistical properties: the model is correct and the sample size is relatively large. This algorithm has been used for the estimation of generalized linear mixed model (GLMM) and is shown to work well even with bad starting values (Skrondal and Rabe-Hesketh 2004).

4. Comparison of Computational Efficiency

In this section, we will investigate the relative computational properties of the EM and the Quasi-Newton type algorithms. Since the EM and Quasi-Newton algorithms use different convergence criteria and computational efficiency of the algorithm can greatly affected by the implementation, it is very hard to conduct a meaningful comparison with numerical examples. Thus, instead, we will discuss some analytical results that will affect the performance of these algorithms. The purpose of this study is to illustrate the potential advantage of Quasi-Newton algorithms in the filed of IRT. It is not meant to be extensive that will cover all the aspects of the problem.

As shown by equation 6, the Q function is a summation of J functions that involve independent parameters. As a result, maximizing the Q function is equivalent to maximizing J separate functions with 2 parameters each. In contrast, directly maximizing the marginal likelihood of (8) requires handling all $2J$ parameters simultaneously. This is the most important advantage for the EM algorithm. This advantage become more significant as the number of items increases. On the other hand, the advantage of the Quasi-Newton algorithms lies in the calculation of derivatives. To implement the Quasi-Newton algorithm, we only need to calculate the gradient that involve $2J$ elements. In contrast, the M step of the EM algorithm needs to calculate $3J$ elements of the Hessian matrix on top of the gradient. For a multidimensional IRT model with d latent factors, EM algorithms will need to do $2 + \frac{d}{2}$ times more calculations than the Quasi-Newton algorithm. Thus as the number of latent factor, d, increases, this advantage for Quasi-Newton algorithm becomes more obvious.

Technical detail provided in the appendix suggest that both algorithms involve similar calculations for each iteration which can be divided into two steps. The first step calculates the gradient and(or) hessian, which is accomplished by a nested loop that involve $N \times G$ summations in total. The key components for each summation are the calculations of exponential function and(or) the CDF of standard normal distribution. The total number of summations ranges from thousands to millions. Table 1 lists the computation time used to calculate different number of exponential functions and CDFs

Then in the second step, parameters are updated with the Newton type equation as follows

$$\theta_{t+1} = \theta_t - \frac{f'(\theta_t)}{f''(\theta_t)}, \tag{10}$$

which is equivalent to solving a system of linear equations. With current computational techniques and resources, solving a system of linear equations is easy and fast. In Table

	Number of summations		
	2000	20000	400000
Exp	0.01	0.09	1.7
CDF	0.01	0.12	2.3

Table 1: Computation time for the calculation of different number of exponential functions and standard normal CDFs. Computations are conducted in SAS IML

	Number of Parameters					
	2	10	100	500	1000	2000
Computation Time	9e-05	1e-4	6e-4	0.04	0.29	2.2

Table 2: Computation time used for solving system of linear equations with different number of parameters. Computations are conducted in SAS IML

2 we list the computation time used for solving system of linear equations with different number of parameters.

Note that these calculations are conducted using SAS IML. While the absolute computational time might be different if different programming tools are used, we assume the relative computational time between the calculation of exponential function, CDF and solving system of linear equations are the same. Comparing Table 1 and 2, we can observe that computations for the first step often dominate the computation time for each iteration unless the number of parameters is very large, for example above 1000. Thus the advantage for EM algorithm has a less significant impact on the total computation time for each iteration, and as a result, we can expect that the Quasi-Newton will be as fast as, if not faster than, the EM for each iteration. Since EM algorithm's convergence rate is linear at most and Quasi-Newton is supper linear, the Quasi-Newton algorithm is expected to use less iterations to reach the same convergence criteria. Furthermore, as the development of IRT models, especially multidimensional cases, confirmatory analysis becomes increasing useful and desire. When parameter restrictions are applied across different items, the Q function in (6) can not be decomposed into the summation of independent functions and consequently the above advantage for EM algorithm will be impaired.

5. Discussion

In this paper, we try to demonstrate that, under most cases, Quasi-Newton algorithms for IRT model are computationally as efficient as, if not more than, EM algorithms and meanwhile can avoid the problems associated the EM algorithms. We do not claim that Newton type algorithms are always better than EM algorithms. EM algorithms, especially Monte Carlo EM, are usually easier to implement. That makes EM very popular among methodology researchers who need to implement estimation algorithms for newly developed modeling techniques. However, these disadvantages make EM type algorithms not a good candidate for commercial softwares for whom estimation accuracy and reliability are invaluable. A hybrid algorithm that starts with EM and then switches to Newton type algorithms has also been proposed. It could be a better option than EM and Newton under certain conditions, but more explorations are needed to identify these situations for IRT.

A. Technical details for EM

The conditional distribution $f(\eta|u_i, \theta^{(t)})$ is

$$f(\eta|u_i, \theta^{(t)}) = \frac{f(u_i|\eta, \theta^{(t)})\phi(\eta)}{\int f(u_i|\eta, \theta^{(t)})\phi(\eta)d\eta} = \frac{f(u_i|\eta, \theta^{(t)})\phi(\eta)}{f(u_i)} \tag{11}$$

Then these conditional expectations involved in the Q function can be expressed as follows

$$E[logP_{ij}|u_i, \theta^{(t)}] = \int logP_{ij} f(\eta|u_i, \theta^{(t)})d\eta \tag{12}$$

$$E[log(1-P_{ij})|u_i, \theta^{(t)}] = \int log(1-P_{ij}) f(\eta|u_i, \theta^{(t)})d\eta \tag{13}$$

$$E[log\phi(\eta)|u_i, \theta^{(t)}] = \int log\phi(\eta) f(\eta|u_i, \theta^{(t)})d\eta \tag{14}$$

then we have

$$\begin{aligned} Q_{1j} &= \int \sum_{i=1}^{N} \left[u_{ij} logP_{ij} f(\eta|u_i, \theta^{(t)}) + (1-u_{ij}) log(1-P_{ij}) f(\eta|u_i, \theta^{(t)})\right] d\eta \\ &= \int \left[logP_{ij} \left[\sum_{i=1}^{N} u_{ij} f(\eta|u_i, \theta^{(t)})\right] + log(1-P_{ij}) \left[\sum_{i=1}^{N} (1-u_{ij}) f(\eta|u_i, \theta^{(t)})\right]\right] d\eta \\ &= \int \left[logP_{ij} r_j(\theta^{(t)}) + log(1-P_{ij})[n(\theta^{(t)}) - r_j(\theta^{(t)})]\right] \phi(\eta|\theta^{(t)}))d\eta \end{aligned} \tag{15}$$

where $r_j(\theta^{(t)}) = \sum_{i=1}^{N} u_{ij} \frac{f(u_i|\eta, \theta^{(t)})}{f(u_i)}$, and $n(\theta^{(t)}) = \sum_{i=1}^{N} \frac{f(u_i|\eta, \theta^{(t)})}{f(u_i)}$.

Integrations in above equations can be approximated as follows using G-H quadrature. Note that these quadrature points, x_g, and weights, w_g, are corresponding to $\phi(\eta|\theta^{(t)})$ which is the density function of $N(0, \Phi^{(t)})$.

$$\tilde{Q}_{1j} = \sum_{g=1}^{G} \left[logP_{ij}(x_g) r_j(x_g, \theta^{(t)}) + log(1-P_{ij}(x_g))(n(x_g, \theta^{(t)}) - r_j(x_g, \theta^{(t)}))\right] w_g \tag{16}$$

We take the derivatives of Q_{1j} with respect to model parameters

$$\frac{\partial \tilde{Q}_{1j}}{\partial \alpha_j} = \sum_{g=1}^{G} \left[\frac{r_j(x_g, \theta^{(t)})}{P_{ij}(x_g)} - \frac{n(x_g, \theta^{(t)}) - r_j(x_g, \theta^{(t)})}{1-P_{ij}(x_g)}\right] \frac{\partial P_{ij}(x_g)}{\partial \alpha_j} w_g \tag{17}$$

$$\frac{\partial \tilde{Q}_{1j}}{\partial \lambda_j} = \sum_{g=1}^{G} \left[\frac{r_j(x_g, \theta^{(t)})}{P_{ij}(x_g)} - \frac{n(x_g, \theta^{(t)}) - r_j(x_g, \theta^{(t)})}{1-P_{ij}(x_g)}\right] \frac{\partial P_{ij}(x_g)}{\partial \lambda_j} w_g \tag{18}$$

$$\frac{\partial^2 \tilde{Q}_{1j}}{\partial \alpha_j^2} = \sum_{g=1}^{G} \left[\left[\frac{-r_j}{P_{ij}^2} - \frac{n-r_j}{(1-P_{ij})^2}\right] \left[\frac{\partial P_{ij}(x_g)}{\partial \alpha_j}\right]^2 + \left[\frac{r_j}{P_{ij}} - \frac{n-r_j}{1-P_{ij}}\right] \frac{\partial^2 P_{ij}(x_g)}{\partial \alpha_j^2}\right] w_g \tag{19}$$

$$\frac{\partial^2 \tilde{Q}_{1j}}{\partial \lambda_j^2} = \sum_{g=1}^{G} \left[\left[\frac{-r_j}{P_{ij}^2} - \frac{n-r_j}{(1-P_{ij})^2}\right] \left[\frac{\partial P_{ij}(x_g)}{\partial \lambda_j}\right]^2 + \left[\frac{r_j}{P_{ij}} - \frac{n-r_j}{1-P_{ij}}\right] \frac{\partial^2 P_{ij}(x_g)}{\partial \lambda_j^2}\right] w_g \tag{20}$$

$$\frac{\partial^2 \tilde{Q}_{1j}}{\partial\alpha_j\partial\lambda_j} = \sum_{g=1}^{G^d}\left[\left[\frac{-r_j}{P_{ij}^2} - \frac{n-r_j}{(1-P_{ij})^2}\right]\left[\frac{\partial P_{ij}(x_g)}{\partial\alpha_j}\frac{\partial P_{ij}(x_g)}{\partial\lambda_j}\right] + \left[\frac{r_j}{P_{ij}} - \frac{n-r_j}{1-P_{ij}}\right]\frac{\partial^2 P_{ij}(x_g)}{\partial\alpha_j\partial\lambda_j}\right] w_g \tag{21}$$

In the above equations, we have

$$\frac{\partial P_{ij}(x_g)}{\partial\alpha_j} = -\phi(\alpha_j - \lambda_j x_g) = -\frac{\partial Q_{ij}(x_g)}{\partial\alpha_j} \tag{22}$$

$$\frac{\partial P_{ij}(x_g)}{\partial\lambda_j} = \phi(\alpha_j - \lambda_j x_g)x_g = -\frac{\partial Q_{ij}(x_g)}{\partial\lambda_j} \tag{23}$$

$$\frac{\partial^2 P_{ij}(x_g)}{\partial\alpha_j^2} = -\frac{\partial\phi(\alpha_j - \lambda_j x_g)}{\partial\alpha_j} = \phi(\alpha_j - \lambda_j x_g)(\alpha_j - \lambda_j x_g) = -\frac{\partial^2 Q_{ij}(x_g)}{\partial\alpha_j^2} \tag{24}$$

$$\frac{\partial^2 P_{ij}(x_g)}{\partial\alpha_j\partial\lambda_j} = -\frac{\partial\phi(\alpha_j - \lambda_j x_g)}{\partial\lambda_j} = -\phi(\alpha_j - \lambda_j x_g)(\alpha_j - \lambda_j x_g)x_g = -\frac{\partial^2 Q_{ij}(x_g)}{\partial\alpha_j\partial\lambda_j} \tag{25}$$

$$\frac{\partial^2 P_{ij}(x_g)}{\partial\lambda_j^2} = \frac{\partial\phi(\alpha_j - \lambda_j x_g)x_g}{\partial\lambda_j} = \phi(\alpha_j - \lambda_j x_g)(\alpha_j - \lambda_j x_g)x_g^2 = -\frac{\partial^2 Q_{ij}(x_g)}{\partial\lambda_j^2} \tag{26}$$

B. Technical details for Quasi-Newton

For our objective function, $Log\tilde{L}(\theta)$, the first derivatives with respect to θ_j, parameter for the jth item, is

$$\frac{\partial log\tilde{L}(\theta|U)}{\partial\theta_j} = \sum_{i=1}^{N}\left[(\tilde{L}_i)^{-1}\frac{\partial\tilde{L}_i}{\partial\theta_j}\right] = \sum_{i=1}^{N}\left[(\tilde{L}_i)^{-1}\sum_{g=1}^{G}\left[\frac{\partial f_i(x_g)}{\partial\theta_j}w_g\right]\right], \tag{27}$$

where

$$\tilde{L}_i = \sum_{g=1}^{G}\left[\prod_{j=1}^{J}(P_{ij}(x_g))^{u_{ij}}(Q_{ij}(x_g))^{1-u_{ij}}\right] w_g = \sum_{g=1}^{G} f_i(x_g)w_g, \tag{28}$$

$$\frac{\partial f_i(x_g)}{\partial\theta_j} = \frac{\partial[P_{ij}(x_g)^{u_{ij}}Q_{ij}(x_g)^{1-u_{ij}}]}{\partial\theta_j}\frac{f_i(x_g)}{P_{ij}(x_g)^{u_{ij}}Q_{ij}(x_g)^{1-u_{ij}}}. \tag{29}$$

where for the probit link

$$\frac{\partial P_{ij}(x_g)}{\partial\alpha_j} = -\phi(\alpha_j - \lambda_j x_g) = -\frac{\partial Q_{ij}(x_g)}{\partial\alpha_j} \tag{30}$$

$$\frac{\partial P_{ij}(x_g)}{\partial\lambda_j} = \phi(\alpha_j - \lambda_j x_g)x_g = -\frac{\partial Q_{ij}(x_g)}{\partial\lambda_j} \tag{31}$$

References

E. Berndt, B. Hall, R.E.Hall, and J. Hausman. Estimation and inference in nonlinear structural models. *Annals of Economic and Social Measurement*, 3:653–666, 1974.

R. D. Bock and M. Aitkin. Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, 46(4):443–459, 1981.

R. D. Bock and M. Lieberman. Fitting a response model for n dichotomously scored items. *Psychometrika*, 35:179–197, 1970.

X. L. Meng and D. B. Rubin. Using EM to Obtain asymptotic variance - covariance matrices - the SEM algorithm. *Journal of the American Statistical Association*, 86(416): 899–909, 1991.

A. Skrondal and S. Rabe-Hesketh. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Chapman & Hall/CRC, 2004.