Maximum Likelihood and Bayes Modal Ability Estimation in Two-Parametric IRT Models: Derivations and Implementation

Norman Rose
Institute of Psychology
Friedrich Schiller University Jena
November 2010

In this paper two different estimation procedures, Maximum Likelihood (MLE) and Bayes Modal Estimation (Maximum a posteriori; MAP), are considered. Both methods are well-known and widely used. The aim of this paper is to show how the estimation equations can be derived from the basic model equation of Birnbaum's two-parametric model. We demonstrate the implementation of the derived estimators in the software R using the Newton-Raphson Algorithm.

Introduction

Especially in the field of educational testing, models of Item Response Theory (IRT) have become very popular. The advantage of tests based on IRT is their potential use in computerized adaptive testing (Wainer et al., 2000; van der Linden & Glas, 2000). The focus of this paper is to demonstrate how the person parameter in IRT models can be estimated using maximum likelihood (ML; Enders, 2005; Eliason, 1993; Held, 2008) and Bayesian estimation theory (Gelman, Carlin, Stern, & Rubin, 2003; Held, 2008). There is a considerable body of literature dealing with parameter estimation techniques in IRT (e. g. Baker & Kim, 2004; Embretson & Reise, 2000; de Ayala, 2009; Bock & Aitkin, 1981). Unfortunately, most of the literature provides the final formulas without explicating the associated derivations including the underlying rationale. Only a few papers explain how to implement estimation routines in programming language (Partchev, 2008). In this paper, the ML estimator and the Bayes modal estimator will be derived step by step using Birnbaum's two-parametric logistic model. For didactic reasons the applied mathematics are explained in detail. Subsequently, the estimation equations will be implemented in the software R (R Development Core Team, 2010) applying the Newton-Raphson algorithm.

In IRT models two sets of parameters can be distinguished: (a) item parameters that characterize the items of a test and (b) the person parameter ξ, which can be a latent ability in achievement tests or other personal characteristics. In the present paper the considerations will be confined to the estimation of ξ. This means that the item parameters are assumed to be known. This is not unrealistic as in many computer based test applications the estimation of the latent variable ξ is based on a calibrated set of items. Hence, the item parameters have been estimated in advance based on a sample. Applying CAT, an item bank with a sufficient number of calibrated items, is essential. The previously estimated item parameters are used in the estimation of ξ. Adaptive testing differs from non-adaptive approaches by the administration of items. In a non-adaptive test setting, the number of items is fixed and usually the same for all test takers. Once the test taker has completed the entire test the person parameter is estimated. In contrast, in adaptive testing, after initial responses to starting items, the choice of the subsequent items rests upon tentative estimates $\hat{\xi}$ based on the previous item responses of the examinee. Such items are chosen that are expected to contribute the most information on the person parameter. The concept of information is essential not only in adaptive testing and will be introduced more formally below. The estimators provided in this paper can be used in adaptive and non-adaptive testing settings in order to obtain provisional and final estimates of ξ.

Maximum Likelihood Estimation of a Person's Ability

The maximum likelihood estimation can be used to estimate unknown parameters based on sample data. Let $\boldsymbol{Y} = Y_1, \ldots, Y_k$ be a $1 \times k$ dimensional random variable with the single items Y_i. It is important to distinguish between $\boldsymbol{Y}$ and its realization $\boldsymbol{u} = u_1, \ldots, u_k$ that represents the data. The MLE rests upon the likelihood function $L(\boldsymbol{Y} = \boldsymbol{u}; \boldsymbol{\theta})$. For brevity, we simply write $L(\boldsymbol{u}; \boldsymbol{\theta})$ in the remainder. $\boldsymbol{\theta}$ is the vector of unknown parameters we aim to estimate. If a random sample of N test takers is drawn and the set of k items is presented, the ML function can be defined as

$$L(\boldsymbol{u}; \boldsymbol{\theta}) \quad = \quad P(\boldsymbol{Y}_1 = \boldsymbol{u}_1, \ldots, \boldsymbol{Y}_N = \boldsymbol{u}_N; \boldsymbol{\theta}) \qquad (1)$$

For $N > 1$, $\boldsymbol{u}$ is a $N \times k$ data matrix. Each row represents the realized response vector of an examinee. The right hand side of Equation 1 is also called the joint distribution function (e. g. Eliason, 1993). The ML function as defined here is a probability function describing the probability of the occurrence of the data as a function of the unknown parameters[1]. The

[1] The likelihood function does not necessarily need to be equal to the probability function. It is sufficient that $L(\boldsymbol{u}; \boldsymbol{\theta})$ is proportional to the probability function.

ML estimator $\hat{\boldsymbol{\theta}}$ is defined as

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Omega_{\boldsymbol{\theta}}} L(\boldsymbol{u};\, \boldsymbol{\theta}). \tag{2}$$

Hence, the underlying principle of the ML estimation is to find the value of $\boldsymbol{\theta}$ defined in the parameter space $\Omega_{\boldsymbol{\theta}}$ that maximizes the probability of the occurrence of the data. Note that the data are regarded as events that occur with a particular typically unknown probability.

In the commonly used IRT models it is assumed that the probability to solve a particular item i is conditionally independent from the responses to other items $l \neq i$ given the latent variable ξ. Hence, given person's ability the probability to solve one item does not depend on the performance on other items. Under this so-called local stochastic independence (e. g. Embretson & Reise, 2000) we can write Equation 1 as

$$\begin{aligned} L(\boldsymbol{u};\, \boldsymbol{\theta}) &= P(Y_{11} = u_{11}, \ldots, Y_{Nk} = u_{Nk};\, \boldsymbol{\theta}) \\ &= \prod_{n=1}^{N}\prod_{i=1}^{k} P(Y_{ni} = u_{ni};\, \boldsymbol{\theta}). \end{aligned} \tag{3}$$

Y_{ni} refers to the i-th item and the n-th randomly sampled test taker. u_{ni} is a realization of Y_{ni}. In the most general form, the parameter vector $\boldsymbol{\theta}$ contains all the item and person parameters[2]. The number of item parameters in $\boldsymbol{\theta}$ depends on the specific IRT model chosen to fit the data.

In this paper we confine the estimation problem to estimating ξ of a single randomly drawn person based on the response vector $\boldsymbol{u}$. Additionally, we know the item parameters. This situation is not unrealistic but typical in computerized testings based on calibrated item pools. The parameter vector reduces to a scalar ξ estimated by $\hat{\xi}$. Therefore, Equation 3 reduces to

$$\begin{aligned} L(\boldsymbol{u};\, \xi) &= P(Y_1 = u_1, \ldots, Y_k = u_k;\, \xi) \\ &= \prod_{i=1}^{k} P(Y_i = u_i;\, \xi), \end{aligned} \tag{4}$$

given that local stochastic independence holds. The items Y_i are dichotomous with the values $Y_i = 0$ for answered incorrectly and $Y_i = 1$ when the item i is solved.[3] Using the 2PL-model (Birnbaum, 1968) the probability to solve item Y_i is

$$P(Y_i = 1\,|\,\xi) = \frac{\exp[\alpha_i(\xi - \beta_i)]}{1 + \exp[\alpha_i(\xi - \beta_i)]}. \tag{5}$$

The terms $P(Y_i = u_i;\, \xi)$ in Equation 4 will be replaced by the righthand side of the 2PL-Model equation. In order to estimate the ability ξ given a realized response pattern $\boldsymbol{u}$, a general likelihood function can be written as

$$\begin{aligned} L(\boldsymbol{u};\, \xi) &= \prod_{i} P(Y_i = u_i;\, \xi) \\ &= \prod_{i} P(Y_i = 1\,|\,\xi)^{u_i} \cdot P(Y_i = 0\,|\,\xi)^{1-u_i}. \end{aligned} \tag{6}$$

In real applications this ensures that the likelihood of the actual realized response vector is calculated. As the number of administered items increases the value of the likelihood $L(\boldsymbol{u};\, \xi)$ quickly becomes tiny. To avoid numerical complications because of the small numbers, the natural logarithm $\ln[L(\boldsymbol{u};\, \xi)]$, denoted as $l(\boldsymbol{u};\, \xi)$, is commonly used (see Equation 7). Furthermore, the log-transformation of the likelihood converts a product into a sum which simplifies the subsequent derivations substantially.

$$\begin{aligned} l(\boldsymbol{u};\, \xi) &= \sum_{i} \ln\left[P(Y_i\,|\,\xi)\right] \\ &= \sum_{i} \ln\left[P(Y_i = 1\,|\,\xi)^{u_i} \cdot P(Y_i = 0\,|\,\xi)^{1-u_i}\right] \\ &= \sum_{i}\Big\{u_i \cdot \ln\left[P(Y_i = 1\,|\,\xi)\right] \\ &\qquad +(1 - u_i) \cdot \ln\left[P(Y_i = 0\,|\,\xi)\right]\Big\} \end{aligned} \tag{7}$$

Recall that the ML estimate is that value of the parameter space which maximizes the likelihood or the log-likelihood function. The maximum of any function $f(X)$ of a real-valued variable X can be found by setting the first derivative of this function equal to zero. The first derivative $f'(x)$ of any continuous and differentiable function $f(x)$ is the slope of $f(X)$ at point $X = x$. At any maximum or minimum of $f(X)$ the slope of the function is $f'(X) = 0$. Consequently, a maximum can be found by setting $f'(X) = 0$, and solving for X. So as to estimate ξ the first derivative $l'(\boldsymbol{u};\, \xi)$ of the log-likelihood is needed and a root-finding algorithm needs to be employed.[4] The first derivative $l'(\boldsymbol{u};\, \xi)$ of the log-likelihood (see Equation 7) can be written as

$$\begin{aligned} \frac{d}{d\xi} l(\boldsymbol{u};\, \xi) &= \frac{d}{d\xi}\sum_{i} u_i \cdot \ln\left[P(Y_i = 1\,|\,\xi)\right] \\ &\qquad + (1 - u_i) \cdot \ln\left[P(Y_i = 0\,|\,\xi)\right] \\ &= \sum_{i}\Big\{\frac{d}{d\xi} u_i \cdot \ln\left[P(Y_i = 1\,|\,\xi)\right] \\ &\qquad + \frac{d}{d\xi}(1 - u_i) \cdot \ln\left[P(Y_i = 0\,|\,\xi)\right]\Big\}. \end{aligned} \tag{8}$$

[2] The number of elements of $\boldsymbol{\theta}$ depends on the parametric model and the particular MLE method. Using the Joint Maximum Likelihood (JML) method each person parameter is enclosed. Using Marginal Maximum Likelihood (MML), $\boldsymbol{\theta}$ reduces to the item parameters and the quantities describing the distribution of ξ as the variance and the expected value. For a comparison of different MLE methods see e. g. Baker & Kim (2004) and Embretson & Reise (2000).

[3] The terminology used here is associated with achievement tests in order to keep the language simple. But note, that for items that aim to indicate person characteristics such as neuroticism or openness, it is not appropriate to use terms like *correct* or *incorrect*.

[4] Note that further MLE algorithms exist that avoid the use of the derivative of the likelihood, for instance direct maximization methods (Turner, 2008).

The derivative of a sum of two functions equals the sum of the derivatives of these functions. Therefore, we have to find the first derivatives of two logarithmic functions with respect to each item Y_i. The terms u_i and $1 - u_i$ are constants and can be placed in front of the respective first derivatives. Note that the functions $\ln[P(Y_i = y \,|\, \xi)]$ are chained. The natural logarithm ln() is the outer function and $P(Y_i = y \,|\, \xi)$ is the inner function given by the model equation (Equation 5). Hence the chain rule needs to be applied, which states that the first derivative of a chained function is the product of the first derivative of the outer function and the first derivative of the inner function. For any function $f(X)$ the natural logarithm is $\ln[f(X)] = 1/f(X) \cdot f'(X)$. Applied to Equation 8 this leads to

$$\begin{aligned}\frac{d}{d\xi} l(\boldsymbol{u};\, \xi) = \sum_i \Big\{ & u_i \cdot \frac{1}{P(Y_i = 1 \,|\, \xi)} \cdot \frac{d}{d\xi} P(Y_i = 1 \,|\, \xi) \\ & + (1 - u_i) \cdot \frac{1}{1 - P(Y_i = 1 \,|\, \xi)} \\ & \cdot \frac{d}{d\xi} \left[1 - P(Y_i = 1 \,|\, \xi, \alpha_i, \beta_i)\right] \Big\}.\end{aligned} \tag{9}$$

In this form also the estimation equation contains two derivatives that have to be computed. Both involve the model equation of the respective IRT model, here the 2PL-model given by Equation 5. The model equation is a chained function of the person variable ξ. Hence, the chain rule is required multiple times. Decomposing the model equation gives two functions: (a) the inner function $k(\xi) = 1 + \exp[-\alpha_i(\xi - \beta_i)]$ and (b) $g(\xi) = 1/k(\xi)$ the outer function. Applying the chain rule the first derivative of the model equation is given by:

$$\frac{d}{d\xi} P(Y_i = 1 \,|\, \xi) = g'\left[k(\xi)\right] k'(\xi) \tag{10}$$

Therefore the inner and the outer function have to be derived separately. Using the respective calculus for e-function gives

$$\begin{aligned} k'(\xi) &= \frac{d}{d\xi} \left(1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]\right) \\ &= \frac{d}{d\xi} \left(1 + \exp\left[-\alpha_i \xi + \alpha_i \beta_i\right]\right) \\ &= -\alpha_i \cdot \exp\left[-\alpha_i (\xi - \beta_i)\right]. \end{aligned} \tag{11}$$

The first derivative of the outer function is

$$\begin{aligned} g'\left[k(\xi)\right] &= \frac{d}{d\xi} \frac{1}{k(\xi)} \\ &= \frac{d}{d\xi} k(\xi)^{-1} \\ &= -k(\xi)^{-2}. \end{aligned} \tag{12}$$

Combining Equations 11 and 12 yields

$$\begin{aligned} \frac{d}{d\xi} P(Y_i = 1 \,|\, \xi) &= g'\left[k(\xi)\right] k'(\xi) \\ &= -k(\xi)^{-2} \cdot k'(\xi) \\ &= -\left(1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]\right)^{-2} \\ &\quad \cdot (-\alpha_i) \cdot \exp\left[-\alpha_i (\xi - \beta_i)\right] \\ &= \frac{\alpha_i \cdot \exp\left[-\alpha_i (\xi - \beta_i)\right]}{\left(1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]\right)^2}. \end{aligned} \tag{13}$$

Rearranged, the terms reveal the meaning of the first derivative of the model function. It simply is the weighted product of the two conditional category probabilities given the model parameters and the latent variable ξ.

$$\begin{aligned} \frac{d}{d\xi} P(Y_i = 1 \,|\, \xi) &= \alpha_i \cdot \frac{1}{1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]} \\ &\quad \cdot \frac{\exp\left[-\alpha_i (\xi - \beta_i)\right]}{1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]} \\ &= \alpha_i \cdot \frac{1}{1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]} \\ &\quad \cdot \frac{1}{1 + \exp\left[\alpha_i (\xi - \beta_i)\right]} \\ &= \alpha_i \cdot P(Y_i = 1 | \xi) \cdot P(Y_i = 0 \,|\, \xi) \end{aligned} \tag{14}$$

The product $P(Y_i = 1|\xi) \cdot P(Y_i = 0 \,|\, \xi)$ of the two conditional probabilities gives the variance of Y_i given ξ. This is the so-called conditional variance function $Var(Y_i \,|\, \xi)$. Thus, in the case of binary manifest variables the first derivative of the 2PLM is the conditional variance function weighted by the item discrimination α_i. In order to obtain all the building blocks of Equation 9, the first derivative of the conditional probability to fail item Y_i is needed. The computational steps are basically the same as for the derivation of $P(Y_i = 1 \,|\, \xi)$. Attention has to be paid to the fact that the inner function $k(\xi)$ differs between the two equations of $P(Y_i = 1 \,|\, \xi)$ and $P(Y_i = 0 \,|\, \xi)$. In order to avoid confusion, let's denote the inner function of $P(Y_i = 0 \,|\, \xi)$ by $k_0(\xi) = 1 + \exp\left[\alpha_i (\xi - \beta_i)\right]$. The outer function $g(\xi)$ remains the same. Thus, the derivations we did previously are sufficient (see Equation 12). The first derivative of the inner function $k_0(\xi)$ can be obtained utilizing the same mathematical operations as demonstrated in Equation 11. This leads to

$$k_0'(\xi) = \alpha_i \cdot \exp\left[\alpha_i (\xi - \beta_i)\right]. \tag{15}$$

Combined with the first derivative of the outer function we obtain for the first derivative of $P(Y_i = 0 \,|\, \xi)$

$$\begin{aligned} \frac{d}{d\xi} P(Y_i = 0 \,|\, \xi) &= g'\left[k_0(\xi)\right] k_0'(\xi) \\ &= -\alpha_i \cdot P(Y_i = 1 \,|\, \xi) \cdot P(Y_i = 0 \,|\, \xi). \end{aligned} \tag{16}$$

It can be seen that the first derivative of $P(Y_i = 0 \,|\, \xi)$ is the negative first derivation of $P(Y_i = 1 \,|\, \xi)$. Thus again it is the

conditional variance function $Var(Y_i \mid \xi)$ weighted with the negative item discrimination parameter. Now, we can insert Equations 14 and 16 into Equation 9.

$$\begin{aligned}\frac{d}{d\xi} l(\boldsymbol{u};\, \xi) &= \sum_i \Big\{ u_i \cdot \frac{\alpha_i \cdot P(Y_i = 1 \mid \xi) \cdot P(Y_i = 0 \mid \xi)}{P(Y_i = 1 \mid \xi)} \\ &\quad + (1 - u_i) \cdot \frac{-\alpha_i \cdot P(Y_i = 1 \mid \xi) \cdot P(Y_i = 0 \mid \xi)}{P(Y_i = 0 \mid \xi)} \Big\} \\ &= \sum_i \Big\{ u_i \cdot \alpha_i \cdot P(Y_i = 0 \mid \xi) \\ &\quad + (1 - u_i) \cdot (-\alpha_i) \cdot P(Y_i = 1 \mid \xi) \Big\} \\ &= \sum\nolimits_i \alpha_i \left[u_i - P(Y_i = 1 \mid \xi) \right]\end{aligned} \tag{17}$$

Its important to note the difference $u_i - P(Y_i = 1 \mid \xi)$ is nothing else but the residual of the regression $E(Y_i \mid \xi)$. Hence, the person parameter estimate of ξ is that value that minimizes the sum of the weighted residuals over the I items. Here it can directly be seen that the contribution of an item Y_i increases with the item discrimination α_i. In the next section the estimation function for the Bayesian maximum a posteriori (MAP) estimator will be derived and compared to the ML estimator.

Bayes Modal Estimation

As the name implies, the Bayes Modal Estimation (BME) has been developed within the Bayesian statistical framework which faces the frequentist approach. Both of these frameworks have seemed to be incompatible for 250 years. The so-called Bayesian-Frequentist debate (Efron, 2005) will not be continued here. It should only be noted that the MLE method as outlined previously refers to the frequentist approach, whereas BME considered in this section belongs to the Bayesian methods. It is important to note, that both estimation procedures are not compatible. Furthermore, the decision for one of these estimators might be associated with tremendous theoretical and practical consequences. Bayesian point and interval estimators can vary reasonably compared to common ML estimators. Whereas the parameter $\boldsymbol{\theta}$ aimed to be estimated is unknown but considered to be fixed in the frequentist's perspective, it is considered to be a random variable in the Bayesian framework. As a consequence, confidence intervals[5] and p-values have different meanings. A detailed discussion of all these differences is far beyond the scope of this paper. The interested reader is referred to Gelman et al. (2003). Here, we confine the considerations to the mathematical and technical aspects of MLE and BME. The mathematical relations between ML and Bayesian estimators allows us to use several derivations of the previous section. Let us start with the theorem which has given the Bayesian statistics its name: The Bayes' rule, named after the statistician Thomas Bayes (e. g. Everitt, 2005).

All Bayesian statistical methodology is based on the Bayes' rule, which relates conditional probabilities of two events, A and B, defined in the same probability space. Thus, A and B have a joint distribution, so that two conditional probabilities $P(A \mid B)$ and $P(B \mid A)$ can be considered. The Bayes' rule describes how these two probabilities are linked mathematically (e. g. Steyer, 2002). Transferred to the context of parameter estimation within the IRT framework, the equivalents for A and B are the response vector $\boldsymbol{Y}$ and the parameter vector $\boldsymbol{\theta}$, which represent all the parameters aimed to be estimated. In the Bayesian framework they are both viewed as random variables with a joint distribution. As shown previously, MLE rests upon maximizing the conditional probability $P(\boldsymbol{Y} \mid \boldsymbol{\theta})$, which is equivalent to the right-hand side of Equation 1. However, the Bayesian inference is based on the so-called posterior distribution $P(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u})$[6]. If $\boldsymbol{\theta}$ is real-valued, the posterior probability function is a density function denoted as $f(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u})$. Due to the Bayes' rule we can write the posterior distribution as

$$f(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u}) = \frac{P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta}) \cdot f(\boldsymbol{\theta})}{\int P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta}) \cdot f(\boldsymbol{\theta}) d\boldsymbol{\theta}}. \tag{18}$$

All Bayesian statistical inference rest upon the posterior distribution. Different statistics that serve as point estimators can be derived from the posterior distribution. The expected a posteriori (EAP) estimator is defined as the expected value of the posterior distribution. An alternative is the Bayes modal estimator, also known as the maximum a posteriori (MAP) estimator, defined as the mode $Mod(\boldsymbol{\theta} \mid \boldsymbol{Y})$ of the posterior distribution.

$$Mod(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u}) = \arg\max_{\boldsymbol{\theta} \in \Omega_\theta} f(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u}) \tag{19}$$

The denominator of Equation 18 refers to the law of total probability with $P(\boldsymbol{Y} = \boldsymbol{u}) = \int P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta}) \cdot f(\boldsymbol{\theta}) d\boldsymbol{\theta}$. The denominator of the posterior distribution is simply the unconditional probability of the response vector. Given any possible response pattern $\boldsymbol{Y}$, this is a constant ensuring that the integral of the posterior distribution is one. In mathematical terms it serves as a normalizing constant. Therefore, the posterior distribution is proportional to the nominator of Equation 18. Consequently, the MAP estimator can equivalently be defined as

$$Mod(\boldsymbol{\theta} \mid \boldsymbol{Y} = \boldsymbol{u}) = \arg\max_{\boldsymbol{\theta} \in \Omega_\theta} P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta}) \cdot f(\boldsymbol{\theta}). \tag{20}$$

For that reason, the denominator of Equation 18 can be omitted for the purpose of parameter estimation. It is sufficient to develop an estimation equation based on $P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta}) \cdot f(\boldsymbol{\theta})$. The first factor $P(\boldsymbol{Y} = \boldsymbol{u} \mid \boldsymbol{\theta})$ is the probability of occurrence of the response vector $\boldsymbol{Y} = \boldsymbol{u}$ given the unknown parameters. This probability function is given by the ML function $L(\boldsymbol{u};\, \boldsymbol{\theta})$ as defined in Equations 1, 2, and 4. The second term $f(\boldsymbol{\theta})$ is the so-called prior distribution or simply the prior. Usually,

[5] The Bayesian analogy of the confidence interval is the so-called credibility interval.

[6] The fact that the model parameters are considered as random variables $\boldsymbol{\theta}$ is highlighted by the cursive notation.

$f(\boldsymbol{\theta})$ is not known previously to the data analysis. The specification of the prior distribution reflects the knowledge or the belief of the researcher with respect to the parameter $\boldsymbol{\theta}$ in advance. Insofar, the choice of the prior is always somewhat arbitrary. How to choose an appropriate prior distribution is one of the fundamental problems of Bayesian statistics and will not be discussed here. The interested reader is referred to Gelman et al. (2003).

Let us go back to the estimation problem of the latent person variable ξ considered here. Assume the distribution of ξ is known to be normally distributed. In fact, this is a common choice of the prior in IRT. As for the ML estimation, let us consider the case where the item parameter α_i and β_i are known. The vector of parameters that have to be estimated reduces to a single value of ξ. The variable $\boldsymbol{Y}$ is the response vector obtained by applying the items $Y_1, \ldots, Y_i, \ldots, Y_k$. Let $L_{MAP}(\xi)$ denote the estimation equation of the MAP estimator, then Equation 20 can be written as

$$L_{MAP} = L(\boldsymbol{u}; \xi) \cdot f(\xi). \tag{21}$$

It can be seen that the MAP estimator is defined as the common ML estimator weighted by the prior distribution. L_{MAP} is a function of the the latent variable ξ. Again we want to find the values defined in the parameter space Ω_ξ that maximize the estimation function. If L_{MAP} is a differentiable function with respect to ξ, we can proceed in the same way as for MLE. The first derivative of L_{MAP} can be set equal to zero and the maxima can be found by a root finding algorithm. Here as well the natural logarithm $l_{MAP} = \ln(L_{MAP})$ is used instead of L_{MAP} because of the mathematical and numerical benefits outlined previously. l_{MAP} is given by

$$l_{MAP} = l(\boldsymbol{u}; \xi) + \ln[f(\xi)]. \tag{22}$$

The first derivative of l_{MAP} is

$$\begin{aligned} l'_{BME} &= \frac{d}{d\xi} l(\boldsymbol{u}; \xi) + \frac{d}{d\xi} \ln[f(\xi)] \\ &= l'(\boldsymbol{u}; \xi) + \frac{d}{d\xi} \ln[f(\xi)]. \end{aligned} \tag{23}$$

Thus the first derivative of l_{MAP} is the sum of the ML estimation equation $l'(\boldsymbol{u}; \xi)$ as derived in the last section and the first derivative of the logarithm of the prior distribution. Let us assume that ξ is known to be normally distributed with $\xi \sim N(\mu_\xi, \sigma_\xi)$. The natural logarithm of the prior distribution is given by

$$\ln[f(\xi)] = \ln\left[\frac{1}{\sqrt{2\pi}\sigma_\xi} \exp\left(\frac{-\left(\xi - \mu_\xi\right)^2}{2\sigma_\xi^2}\right)\right]. \tag{24}$$

Rearranging this equation reveals that the log-transformed prior can be divided into two parts: a function of ξ and a constant that is independent of ξ.

$$\ln[f(\xi)] = \ln\left[\frac{1}{\sqrt{2\pi}\sigma_\xi} \exp\left(\frac{-\left(\xi - \mu_\xi\right)^2}{2\sigma_\xi^2}\right)\right] \tag{25}$$

$$= \ln(1) - \ln\left(\sqrt{2\pi}\right) + \ln\left(\sigma_\xi\right) + \frac{-\left(\xi - \mu_\xi\right)^2}{2\sigma_\xi^2} \tag{26}$$

$$= -0.9189385 + \ln\left(\sigma_\xi\right) - \frac{\left(\xi - \mu_\xi\right)^2}{2\sigma_\xi^2} \tag{27}$$

The term $-0.9189385 + ln(\sigma_\xi)$ does not depend on ξ. Hence, it can be dropped from the estimation equation. In graphical terms, an additive constant shifts the log-likelihood function vertically but neither changes the form of the function nor the location of the maxima or minima. Hence, the equation that has to be derived is further simplified. Only the first derivation of the reduced log-prior has to be found.

$$\begin{aligned} \frac{d}{d\xi} \frac{-\left(\xi - \mu_\xi\right)^2}{2\sigma_\xi^2} &= \frac{d}{d\xi} \frac{-\left(\xi^2 - 2\xi\mu_\xi + {\mu_\xi}^2\right)}{2\sigma_\xi^2} \\ &= -\frac{d}{d\xi}\frac{1}{2\sigma_\xi^2}\xi + \frac{d}{d\xi}\frac{\mu_\xi}{\sigma_\xi^2}\xi - \frac{d}{d\xi}\mu_\xi^2 \\ &= -\frac{\xi - \mu_\xi}{\sigma_\xi^2} \end{aligned} \tag{28}$$

The last line could also be written as $(\mu_\xi + \xi)/\sigma_\xi^2$. Inserting the term of Equation 28 into Equation 23 and including the results from Equation 17 yields

$$l'_{MAP} = \sum\nolimits_i \alpha_i \left[u_i - P(Y_i = 1 \,|\, \xi)\right] - \frac{\xi - \mu_\xi}{\sigma_\xi^2}. \tag{29}$$

Written in this way, Equation uncovers a general characteristic associated with Bayesian estimation procedures. The further more the values of ξ deviate negatively from the mean of the prior distribution, the more positive the first derivative of the log-prior becomes. In turn, the more ξ deviates from the mean of the prior in the positive direction the more negative does the prior become. Hence, the Bayesian estimator tends to pull the estimator toward the mean of the prior. In other words, the influence of the prior increases with the absolute value of the difference $\xi - \mu_\xi$ resulting in a shrinkage of Bayesian estimators towards the mean of the prior distribution. The extent of shrinkage depends on the variance of the prior. The smaller the σ_ξ^2 is, the greater the shrinkage effect, and the more influential the prior becomes in the estimation stage. Recall that all inference in Bayesian statistics rests on the posterior distribution. This poses a combination of the previous knowledge expressed by the prior distribution and the data given a particular model expressed by the likelihood. How strong the previous knowledge is weighted is determined by the variance of the prior. The larger σ_ξ^2 is, the

less influential the prior knowledge is.[7]

With the two derived estimation equations for MLE and MAP the parameter ξ can be estimated for a given response pattern $\boldsymbol{Y}$. It is important to note that the ML estimation method is only applicable if the sum $\sum_i Y_i$ is not zero or equal to the number of answered items. In other words, if all items have been answered correctly or none of the items have been answered correctly the ML estimation will fail. More formally, the ML estimates in these cases are in fact $-\infty$ and $+\infty$. Such values are of no diagnostic value. In contrast, even under these circumstances, the Bayesian procedures yield an estimator for ξ. Unfortunately, neither MLE nor MAP equations can be solved analytically. This means that the extremes of these functions need to be found by using numerical methods. A well known algorithm applicable in these cases is the Newton-Raphson algorithm, which will be introduced in the next section.

Newton-Raphson Algorithm

The Newton-Raphson algorithm (NRA) is a numerical method that allows to find the roots of real-valued functions, which must be continuous and twice differentiable. Since the method is iterative, in a number of identical cycles, the iterations, the true root is approximated. The application of NRA needs an initial guess of the root to serve as a starting value. Let $\hat{\xi}_0$ be denoted as the starting value. The algorithm then finds the value $\hat{\xi}_1$ in the first iteration, $\hat{\xi}_2$ in the second iteration and so forth. In general the estimator $\hat{\xi}_{j+1}$ of the $j+1$th iteration is found by

$$\hat{\xi}_{j+1} = \hat{\xi}_j - \frac{l'\left(\hat{\xi}\right)}{l''\left(\hat{\xi}\right)}. \tag{30}$$

$l''(\hat{\xi})$ is the second derivative of the log-likelihood. The first derivative is also involved, so we can use the derivations done previously. Equation 30 implies that the difference $\hat{\xi}_{j+1} - \hat{\xi}_j$ between the point estimators of two successive iterations is the ratio of the first and second derivative of the log-likelihood. This difference is large if $l'(\hat{\xi})$ is large. This is the case when the log-likelihood is steep and therefore the maximum is expected to be far away. But the distance of the maxima does not only depend on the first derivation but also on the curvature of a function that is expressed by the second derivative. The higher the value of the second derivative, the more rapidly the slope of the function changes, implying that the maximum can be close to the provisional estimate.

The accuracy of the final estimate depends on the arbitrarily chosen convergence criteria that has to be specified prior to the analysis. If no convergence criteria is provided, the algorithm would theoretically never stop to iterate. Different convergence criteria are in use. A freuquently used criterion is the modulus $|l'(\hat{\xi}_j) - l'(\hat{\xi}_j)|$ of the difference between the values of the first derivatives of two successive iterations. The idea is that the allgorithm has converged when the likelihood ceases to change substantially. Typically a value very close to zero is chosen (e. g. < 0.00005). Hence, the result of NRA is an approximate value of the true root approximated by any desired degree of accuracy determined by the convergence criteria. The second derivative of the log-likelihood $l''(\boldsymbol{u}; \xi)$ is not only required in order to employ NRA. The negative second derivative of $l(\boldsymbol{u}; \xi)$ is also called the observed Fisher information and is beyond IRT directly related with standard errors of ML estimates. This will be shown below for the case of estimating the person parameter ξ. At first I provide the technical details for the derivation of $l''(\boldsymbol{u}; \xi)$.

In general, the computation of the second derivative $f''(X)$ of a function $f(X)$ rests on the same mathematical rules as used for the first derivative. This is because $f''(X) = \frac{d}{dX} f'(X)$. So, we can use all the results achieved previously. Let us start with

$$\begin{aligned} \frac{d^2}{d\xi^2} l(\xi) &= \frac{d}{d\xi} l'(\xi) \\ &= \frac{d}{d\xi} \sum\nolimits_i \alpha_i \left[u_i - P(Y_i = 1 \,|\, \xi)\right]. \end{aligned} \tag{31}$$

Solving the brackets and inserting the model equation yields

$$\begin{aligned} \frac{d^2}{d\xi^2} l(\xi) &= \frac{d}{d\xi} \sum\nolimits_i \alpha_i u_i - \alpha_i P(Y_i = 1 \,|\, \xi) \\ &= \frac{d}{d\xi} \sum\nolimits_i \alpha_i u_i - \frac{d}{d\xi} \sum\nolimits_i \alpha_i P(Y_i = 1 \,|\, \xi) \\ &= -\frac{d}{d\xi} \sum\nolimits_i \alpha_i P(Y_i = 1 \,|\, \xi) \\ &= -\sum\nolimits_i \frac{d}{d\xi} \frac{\alpha_i}{1 + \exp\left[-\alpha_i (\xi - \beta_i)\right]}. \end{aligned} \tag{32}$$

The third line of Equation 32 results from the fact that the sum $\sum_i \alpha_i u_i$ is not a function of ξ, so the first derivative of this term is zero. The remaining derivative is rather similar to the computation of the first derivative of the model equation for the 2PLM (see Equation 14). Thus, the chain rule needs to be applied again. The inner functions of ratio in the last term of Equation 32 is equivalent to the inner function of the model equation of the 2PLM. We can use $k'(\xi) = -\alpha_i \cdot \exp\left[-\alpha_i (\xi - \beta_i)\right]$ from Equation 11. It remains the computation of the first derivative $g'(\xi)$ of the outer function that is here $g'(\xi) = \frac{\alpha_i}{k(\xi)}$.

$$\begin{aligned} g'\left[k(\xi)\right] &= \frac{d}{d\xi} \frac{\alpha_i}{k(\xi)} \\ &= \frac{d}{d\xi} \alpha_i \cdot k(\xi)^{-1} \\ &= -\alpha_i \cdot k(\xi)^{-2} \end{aligned} \tag{33}$$

[7] In Bayesian literature a prior with a large variance σ^2_ξ is also called an informative prior. Whereas prior distributions with small values of σ^2_ξ are named non-informative prior, the term informative refers to the weight of the prior knowledge.

Due to the chain rule, we obtain for an item Y_i

$$\begin{aligned}\frac{d}{d\xi}\frac{\alpha_i}{1+\exp\left[-\alpha_i(\xi-\beta_i)\right]} &= g'\left[k\left(\xi\right)\right]k'\left(\xi\right) \\ &= -\alpha_i \cdot k(\xi)^{-2} \cdot k'\left(\xi\right) \\ &= \alpha_i^2 \cdot \frac{\exp\left[-\alpha_i\left(\xi-\beta_i\right)\right]}{\left(1+\exp\left[-\alpha_i\left(\xi-\beta_i\right)\right]\right)^2} \\ &= \alpha_i^2 \cdot P\left(Y_i=1\,|\,\xi\right)\cdot P\left(Y_i=0\,|\,\xi\right).\end{aligned} \tag{34}$$

The second derivative is again the conditional variance function $Var(Y_i\,|\,\xi)$, weighted by the squared item discrimination parameter. In the 2PL-model for dichotomous data this is the item information function $I(Y\,|\,\xi)$. The sum $\sum_i I(Y\,|\,\xi)$ is the test information function $T(\xi)$. Inserting the results of Equation 32 yields

$$\frac{d^2}{d\xi^2}l(\boldsymbol{u};\,\xi) = -\sum_i \alpha_i^2 \cdot P(Y_i=1\,|\,\xi)\cdot P(Y_i=0\,|\,\xi) \tag{35}$$

As stated previously, $-l''(\boldsymbol{u};\,\xi)$ is the observed Fisher information. Additionally, from Equation 35 follows that $-l''(\boldsymbol{u};\,\xi) = T(\xi)$. Thus, in IRT models the test information function is the observed Fisher information. On the basis of $T(\xi)$ the standard errors of $\hat{\xi}$ can be computed by

$$SE(\hat{\xi}) = \frac{1}{\sqrt{T(\xi)}}. \tag{36}$$

So far, we developed the NRA for MLE. As a by-product, we derived the standard errors from the second derivative of the log-likelihood. Let us now consider the NRA for the MAP estimator. Similarly to MLE, the first and second derivative of l_{MAP} is required. Starting with l_{MAP} (Equation), we need to find

$$\begin{aligned}\frac{d^2}{d\xi^2}l_{MAP}\left(\xi\right) &= \frac{d}{d\xi}\sum\nolimits_i \alpha_i\left[u_i - P\left(Y_i=1\,|\,\xi\right)\right] - \frac{\xi-\mu_\xi}{\sigma_\xi^2} \\ &= l''\left(\xi\right) - \frac{d}{d\xi}\frac{\xi-\mu_\xi}{\sigma_\xi^2}.\end{aligned} \tag{37}$$

Again, we benefit from the additive composition of l_{MAP}, which consists of the log-likelihood and the log-transformed prior. So we can use $l''\left(\xi\right)$ from Equation 35 and only need to find the derivative of $(\xi-\mu_\xi)/\sigma_\xi^2$. Rearranging this term reveals that it is a linear function of ξ which can easily be derived.

$$\begin{aligned}\frac{d}{d\xi}\frac{\xi-\mu_\xi}{\sigma_\xi^2} &= \frac{d}{d\xi}\frac{1}{\sigma_\xi^2}\cdot\xi - \frac{\mu_\xi}{\sigma_\xi^2} \\ &= \frac{1}{\sigma_\xi^2}\end{aligned} \tag{38}$$

Inserting this expression into Equation 37 finally yields

$$l''_{BME}\left(\xi\right) = -\sum_i \alpha_i^2 \cdot P(Y_i=1\,|\,\xi)\cdot P(Y_i=0\,|\,\xi) - \frac{1}{\sigma_\xi^2} \tag{39}$$

Similarly the MLE, the standard error of the Bayes modal estimator rests also on the second derivative, here l''_{BME}, which is given by

$$\begin{aligned}SE_{BME}(\hat{\xi}) &= \frac{1}{\sqrt{-l''_{BME}}} \\ &= \frac{1}{\sqrt{T(\xi)+\frac{1}{\sigma_\xi^2}}}.\end{aligned} \tag{40}$$

In Bayesian terminology $-l''_{BME}$ is called the observed posterior information. Comparing the standard errors of MLE and BME reveals an additional difference that is generally associated with Bayesian estimators. The standard errors of Bayesian estimators are smaller compared to those of MLE. The extend to which $SE_{BME}(\hat{\xi})$ is smaller than $SE(\hat{\xi})$ is driven by the variance σ_ξ^2. The smaller σ_ξ^2, the smaller the standard error. Since standard errors are an inverse measure of accuracy of a parameter estimator, smaller standard errors are preferable. But simply choosing a prior with a small variance is not a sufficient way in order to increase the reliability. Due to the considerations made previously, we already know that smaller values of σ_ξ^2 are also associated with a stronger weighting of prior knowledge or belief. Hence, the standard error $SE_{BME}(\hat{\xi})$ reflects the variability of the estimate given the data and the more or less weighted previous knowledge or belief. The latter is more or less arguable.

Summary

In section 2 and 3, we developed the idea of how the unknown value ξ can be estimated based on an observed response pattern. Using MLE, the estimator $\hat{\xi}$ is defined as the value of the parameter space that maximizes the likelihood. Under the assumption of local stochastic independence, the likelihood function is simply the product of the k conditional probabilities to solve the item expressed by the model equations of the 2PLM. Given that the item parameters are known, the likelihood is simply a function of the sought variable ξ. Thus, the ML estimation problem is a maximization problem that can be solved by a root finding algorithm applied to the first derivative of the likelihood or the log-likelihood function respectively. As an alternative for the ML estimator the Bayesian MAP estimator was introduced. All Bayesian inference rests upon the posterior distribution. We showed how the likelihood is connected to the posterior distribution and how a MAP estimation equation can be derived. Since the first derivatives of the likelihood as well as for the MAP estimator cannot be solved analytically we developed the root finding Newton-Raphson algorithm, a very general iterative method applicable to both MLE and MAP estimation. The NRA additionally requires the second derivatives of the log-likelihood and the logarithm of the MAP estimation equation. It was shown that the second derivatives of the estimation functions can also be used to compute the standard errors of the parameter estimates. Although the maximum likelihood estimation and the Bayesian estimation methods are fundamentally different with respect to the underlying

scientific perspective, many mathematical similarities exist. Along with the derivation of the ML and MAP estimator some basic properties of both were compared. We stated that the standard error of the MAP is consistently smaller compared to the standard error of the ML estimate. However, Bayesian estimation requires the specification of a prior distribution that reflects prior knowledge or prior beliefs. However, the prior information might not be appropriate. In these cases biased estimates result. In general, Bayesian estimators are consistently biased at the individual level given the latent ability respectively. Due to the shrinkage, the bias increases the more the latent ability deviates from the mean of the prior distribution.

So far, all considerations were purely theoretical. In the next section it will be shown that all derivations from above are sufficient in order to implement the estimation equations in a computer software. It will be demonstrated how the MLE and the MAP estimation functions can be specified in the freeware R. For didactic reasons we will not use optimization routines readily available in R but our own built functions. These functions are sufficient for adaptive and non-adaptive testing settings when the item parameters are known. The only difference is that in non-adaptive testings only one estimation process runs after the test has been completed. In adaptive testings ability estimation is repeated again and again after each item response until an accuracy criteria or a maximum number of items is reached.

Implementation in R

In this section we demonstrate how the previously derived methods can easily be implemented in the programming language R. This software is optimized for statistical applications and already provides different functions such as `optim()` or `optimize()` allowing for numerical optimization. For didactic reasons, we will not use these functions here. Instead, we will use the previously derived equations and write our own estimation functions. Therefore, it will be easy to understand how the theoretically developed formulas work in an implemented algorithm. In a very small hypothetical example, the use of MLE and the MAP estimator will be demonstrated.

Imagine a test taker was faced with five items of a mathematic test. He has completed the items with solving three of the five items. The response vector $\boldsymbol{u} = (1, 1, 0, 0, 1)$ of realized responses to five dichotomous items. The item parameters are given by the items difficulties $\boldsymbol{\beta} = (-1, -0.5, 0, 0.5, 1)$ and the item discriminations $\boldsymbol{\alpha} = (1, 2, 0.5, 1, 2)$.

We start with the implementation of the `MLE.estimator` function in R. The code is given in Figure 1. The aim is to obtain the ML point estimate, its standard error and a short table that gives information about the estimation process. These three elements are denoted by `xi`, `SE.xi` and `it.log` in the R function (see line 34-36 in Figure 1). Five arguments need to be specified in the `MLE.estimator` function: (a) The response vector $\boldsymbol{u}$ denoted by `resp.vect`. (b) `alpha` denotes the vector $\boldsymbol{\alpha}$ of item discriminations (c) `beta` is the vector $\boldsymbol{\beta}$ of item difficulties, (d) `xi.start` is the starting value ξ_0 to initiate the Newton-Raphson algorithm, and (e) the convergence criteria denoted by `tol`. The NRA is an iterative procedure. Hence, the values of the first and second derivative of the likelihood will be calculated again and again until the convergence criteria given by `tol` is reached. The point estimate `xi` will be updated in each iteration (see line 20). This is realized in R by a while loop (see lines 7-27), which means that the calculations specified within the loop will be carried out as long as the condition `d1.logLikDiff < tol` is not met. The expression `d1.logLikDiff` stands for the modulus of the difference of the derivatives of the log-likelihood between two successive iterations and is the convergence criteria. In order to start the iteration process, the value of `d1.logLikDiff` is arbitrarily set to 999 before starting the first iteration (see line 3). Additionally, the point estimator `xi` is set to the staring value `xi.start` before the while loop begins (see line 2). Each iteration starts with the computation of the vector `probs.1` of conditional expected values $P(Y_1 = 1|\hat{\xi}_j), \ldots, P(Y_I = 1|\hat{\xi}_j)$ (see line 9). The vector $P(Y_1 = 0|\hat{\xi}_j), \ldots, P(Y_I = 0|\hat{\xi}_j)$ with the corresponding counter probabilities is denoted by `probs.0` (see line 10). The first and the second derivative (`d1.logLik` and `d2.logLik`) of the log-likelihood are calculated in lines 13 and 16. The iteration protocol `it.log` is initialized as an empty matrix in lines 4 and 5. In each iteration a row with the updated point estimator and with the first and the second derivative will be added to `it.log` (see line 17). The computation of the convergence criteria `d1.logLikDiff` is specified in line 25. The `if`-condition ensures that `d1.logLikDiff` is only computed from the second iteration on (see line 23-26). If the algorithm has converged, the while loop stops and the standard error `SE.xi` can be computed for the final point estimate `xi` based on the negative second derivative (see line 30). Finally, the three elements `xi`, `SE.xi` and `it.log`, which the function `MLE.estimator` should return, are specified as a list with the respective names (see line 34 - 36). The R-output using `MLE.estimator` to our hypothetical data example is shown in Figure 2. It can be seen that five iterations were needed to reach the convergence criteria. The final point estimator is $\hat{\xi} = 1.183539$ with $SE(\hat{\xi}) = 0.8255662$. Column two of the „iteration log“ reveals that the first derivative of the log-likelihood $l'(\boldsymbol{u}; \xi)$ approaches zero during the iteration procedure. Due to the mathematical similarity of the MLE and MAP estimator, the implementation is akin as well. The R function `BME.estimator` is given in Figure 3. Here I will only mention the differences to the `MLE.funtion`. Since a prior distribution needs to be specified, the list of arguments of the `BME.estimator` function is seven instead of five. New arguments are the expected value of the prior distribution `prior.mean` and the standard deviation of the prior denoted by `prior.std`. The remaining arguments are equal to the ML estimator function. A careful comparison between the R codes for the two estimation functions `MLE.funtion` and `BME.estimator` shows that the only difference occurs in the estimation of the first and the second derivatives in lines 15 and 19, where the prior comes into play. Figure 4 shows the R-output when the `BME.estimator` function is

```
MLE.estimator <- function(resp.vect, alpha, beta, xi.start, tol) {
   xi               <- xi.start
   d1.logLikDiff    <- 999
   it.log           <- matrix(ncol=3,nrow=0)
   colnames(it.log) <- c("estimator", "d1.logLik", "d2.logLik")

   while(d1.logLikDiff > tol)
     {
     probs.1 <-  1/(1+ exp(-alpha*(xi - beta)))
     probs.0 <-  1 - probs.1

     # first derivative
     d1.logLik <- sum(alpha*(resp.vect - probs.1))

     # second derivative
     d2.logLik <- -1* sum(alpha^2  * probs.1 * probs.0)
     it.log <- rbind(it.log, c(xi,d1.logLik,d2.logLik))

     # Newton-Raphson
     xi <- xi - (d1.logLik / d2.logLik)

     # convergence criteria
     if(nrow(it.log) > 1)
        {
        d1.logLikDiff <- abs(d1.logLik - it.log[(nrow(it.log)-1), 2])
        }
     }

   # Standard error
   SE.xi <-  1/sqrt(-1 * d2.logLik)

   # Output of the MLE.estimator function
   Results <- list(xi,SE.xi,it.log)
   names(Results) <- c("point estimator", "standard error", "iteration log")
   return(Results)
}
```

Figure 1. R code for the MLE function.

```
mle  <- MLE.estimator(resp.vect = c(1,1,0,0,1),
                      alpha = c(1,2,0.5,1,2), beta = c(-1,-0.5,0,0.5,1),
                      xi.start = 0, tol = 0.00005)
# R-Output
> mle
$`point estimator`
[1] 1.183539

$`standard error`
[1] 0.8255663

$`iteration log`
     estimator    d1.logLik d2.logLik
[1,]  0.000000 1.940878e+00 -1.700538
[2,]  1.141332 6.257442e-02 -1.497460
[3,]  1.183119 6.166447e-04 -1.467536
[4,]  1.183539 6.581029e-08 -1.467223
[5,]  1.183539 3.885781e-16 -1.467223
```

Figure 2. Example for ML estimation using the MLE function.

```
BME.estimator <- function(resp.vect, alpha, beta, xi, xi.start, tol,
                          prior.mean, prior.std)
{
   xi <- xi.start
   it.log <- matrix(ncol=3,nrow=0)
   colnames(it.log) <- c("estimator", "d1.logLik", "d2.logLik")
   d1.logLikDiff <- 999

  while(d1.logLikDiff > tol)
    {
    probs.1 <-  1/(1+ exp(-alpha*(xi - beta)))
    probs.0 <-  1 - probs.1

    # first derivative
    d1.logLik <- sum(alpha  * (resp.vect - probs.1)) - (xi-prior.mean)/prior.std^2

    # second derivative
    d2.logLik <- -1* sum(alpha^2  * probs.1 * probs.0) - 1/prior.std^2
    it.log <- rbind(it.log,c(xi,d1.logLik,d2.logLik))

    # Newton-Raphson
    xi <- xi - (d1.logLik / d2.logLik)

    # convergence criteria
       if(nrow(it.log) > 1)
          {
          d1.logLikDiff <- abs(d1.logLik - it.log[(nrow(it.log)-1), 2])
          }
    }

  # Standard error
  SE.xi <-  1/sqrt(-1 * d2.logLik)

  # Output of the BME.estimator function
  Results <- list(xi,SE.xi,it.log)
  names(Results) <- c("estimator","standard error","iteration log")
  return(Results)
}
```

Figure 3. R code for the BME function.

applied to our data example. Two results are remarkable. First, the value $\hat{\xi} = 1.183539$ of the point estimator is closer to zero as the ML estimator. This is the effect of the shrinkage toward the mean of the prior. Secondly, the standard error $SE(\hat{\xi}) = 0.8255662$ is smaller compared to those of the ML estimator. As outlined above, this is mainly driven by the standard deviation of the prior, that weights the knowledge or belief prior to the estimation. The smaller the standard deviation of the prior the more is the prior information weighted and the smaller is the standard error of the estimate.

References

Baker, F. B., & Kim, S.-H. (2004). *Item response theory: Parameter estimation techniques* (2nd ed.). Boca Raton, FL: CRC. Hardcover.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. In Lord & M. Novick (Eds.), *Statistical theories of mental test scores.* MA: Addison-Wesley.

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

de Ayala, R. J. (2009). *The theory and practice of item response theory.* New York, NJ: Guilford Press.

Efron, B. (2005). Bayesians, frequentists, and scientists. *Journal of the American Statistical Association*, *100*(469), 1-5.

Eliason, S. R. (1993). *Maximum likelihood estimation: Logic and practice*. Sage University Papers Series on Quantitative Applications in the Social Sience, 07-096. Thousend Oaks, CA: Sage.

Embretson, S. E., & Reise, S. (2000). *Item Response Theory for Psychologists*. Mahwah, NJ: Lawrence Erlbaum Associates.

Enders, C. K. (2005). Maximum likelihood estimation. In B. S. Everitt & D. C. Howell (Eds.), *Encyclopedia of statistics in behavioral science* (p. 1164 - 1170). New York, NJ: Wiley.

Everitt, B. S. (2005). Bayes, Thomas. In B. S. Everitt & D. C. Howell (Eds.), *Encyclopedia of statistics in behavioral*

```
bayes <- BME.estimator(resp.vect = c(1,1,0,0,1),
                       alpha     = c(1,2,0.5,1,2), beta = c(-1,-0.5,0,0.5,1),
                       xi.start  = 0, tol = 0.00005,
                       prior.mean = 0, prior.std  = 1)

# R-Output
> bayes
$estimator
[1] 0.7259562

$`standard error`
[1] 0.6135844

$`iteration log`
     estimator    d1.logLik d2.logLik
[1,] 0.0000000 1.940878e+00 -2.700538
[2,] 0.7187005 1.927532e-02 -2.656972
[3,] 0.7259551 2.982087e-06 -2.656143
[4,] 0.7259562 7.382983e-14 -2.656142
```

Figure 4. Example for MAP estimation using the BME function.

science (p. 129-130). New York, NJ: Wiley.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2003). *Bayesian data analysis*. London: Chapman & Hall.

Held, L. (2008). *Methoden der statistischen Inferenz*. Berlin: Spektrum Akademischer Verlag.

Partchev, I. (2008). *Maximum likelihood and Bayes modal estimation*. Unpublished Manuscript, Department of Methodology and Evaluation Research, Friedrich-Schiller-University, Jena, Germany.

R Development Core Team. (2010). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Available from `http://www.R-project.org` (ISBN 3-900051-07-0)

Steyer, R. (2002). *Wahrscheinlichkeit und Regression*. Berlin: Springer.

Turner, R. (2008). Direct maximization of the likelihood of a hidden markov model. *Computational Statistics & Data Analysis*, *52*(9), 4147-4160.

van der Linden, W., & Glas, A. W. C. (2000). *Computerized adaptive testing: Theory and practice*. Boston: Kluwer Academic Publishers.

Wainer, H., Dorans, N. J., Flaugher, R., Green, B. F., Mislevy, R. J., Steinberg, L., et al. (2000). *Computerized adaptive testing: A primer* (2nd ed.). Mahwah, NJ: Lawrence Erlbaum. Hardcover.