4-22-2013

A Systematic Comparison of MLE and Bayesian Estimation for MPT Models

Quan Tang

A SYSTEMATIC COMPARISON OF MLE AND BAYESIAN ESTIMATION FOR MPT MODELS

by

Quan Tang

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Psychology

The University of Memphis

May 2013

ABSTRACT

Tang, Quan. M.S. The University of Memphis. May, 2013. A Systematic Comparison of MLE and Bayesian Estimation for MPT Models. Major Professor: Xiangen Hu, Ph.D.

As a family of statistical models for categorical data, multinomial processing tree (MPT) models have become popular in cognitive psychology over the course of the past two decades. Classic estimation methods, such as maximum likelihood estimation (MLE) and model fit test (G^2 test), have been applied to MPT models widely. Recent development of Bayesian inference suggests a theoretical alternative for model estimation, though its practical implementation was limited due to the difficulties of computation and sampling capacity of the computers. In this thesis, I apply Bayesian inference to MPT models, develop the programs that implement Bayesian inference for MPT models, and conduct systematic comparisons between the two approaches in terms of their parameter estimation and model evaluation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Multinomial processing tree (MPT) models have been widely used in cognitive psychology, especially in human memory studies (Batchelder and Riefer 1999; Erdfelder et al. 2009) as both a theoretical model and statistical model. In this part of the paper, I will introduce (1) some background information about MPT models and source monitoring research, and (2) statistical methods that can be applied to MPT model analyses, including classic and Bayesian estimations. I will try to introduce these theories and methods through some simple examples. In addition, I will present the reasons for comparing classic estimation to Bayesian estimation.

1.1 Multinomial Processing Tree (MPT) Models and Source Monitoring

Multinomial processing tree (MPT) models are a family of statistical models for serial and discrete data. Formally, MPT models can be regarded as a special family of models in the more general class of parameterized multinomial or product-multinomial models (Stahl and Meiser 2009). MPT models are versatile and may be applied into fields such as cognitive science, medical science, and social science. Though MPT models share basic common features, they (1) are hierarchical and in a tree structure, (2) describe a set of serial processes, and (3) are used to analyze categorical data and may be tailored to different forms according to plausible theories or hypotheses. As a consequence, the development of MPT models has been closely intertwined with the development of paradigms and theories in cognitive psychology.

A typical application of MPT models in cognitive psychology is applying a group of MPT models for source monitoring. Source monitoring research is derived from the interest in human source memories. People remember information from two basic sources: (1) Information perceived from external sources (stimuli), and (2) information generated by internal processes such as reasoning, imagination, and thought. And people may remember, forget, or mix these memories (Johnson and Raye 1981). There is a common phenomenon that most people may have experienced; we heard a story from a friend and forgot who told this story, then we share the story back to this friend with interest. Even worse, we may add something to the story by ourselves unconsciously.

To study different kinds of memories, Johnson and Raye (1981) proposed the concept of “reality monitoring.” Reality monitoring refers to the process of distinguishing the memory of a past perception from the memory of past imagination. As an extension of the reality monitoring, the concept of “source monitoring” was proposed by Johnson and her colleagues (Johnson, Foley, and Leach 1988; Johnson, Hashtroudi, and Lindsay 1993; Johnson and Raye 1981). Compared with reality monitoring which focuses on discriminating memories of internally generated information from memories of externally perceived information, source monitoring refers to discriminating different types of internal or external sources, namely, internal source monitoring or external source monitoring (Johnson, Foley, and Leach 1988). For instance, external source monitoring is interested in discriminating between two externally perceived sources such as statements made by person A or by person B, while internal source monitoring concentrates on discriminating between the memories of what one thought from what one said. Hence source monitoring is derived and generalized from reality monitoring.

After the concepts of reality monitoring and source monitoring were introduced, quite a number of source monitoring experiments were conducted to test different cognitive models or to measure cognitive capacities of different populations. For example, Harvey (1985) studied how different normal and mentally disordered subjects are able to discriminate their own thoughts and information from external sources. Saegert, Hamayan, and Ahmar (1975) tested if source memory for language is dependent on the nature of the memory task itself. And Rose, King, and Perez (1975) examined whether the phenomenon of accurate source memory for language could be found at complex cognitive levels.

In a typical source monitoring experiment, subjects study items from two or more different sources (Johnson, Hashtroudi, and Lindsay 1993). For example, pictures of the items as source A and the names of the items as source B. After these items have been studied, a memory test is given in which the subjects are asked to indicate which source (source A, B or a new source) the test items belong to. Data from a group of subjects can be described by the frequency table as in Table 1, where f_{ij} is the counts of j-type response to i-type source. The row marginal frequency $f_{i.} = \Sigma f_{ij}$ is the total number of i-type source items on the memory test, and $i, j = A, B, C$. In early studies on source monitoring, some ad hoc statistical approaches were adapted for separating the discriminability of the source from the overall detectability of old items (such as Kruskal-Wallis gamma score, identification-of-origin scores, and hit and false-alarm rates for source identification, see Batchelder and Riefer, 1990, for details). The discriminability here means the ability to discriminate the specific old source from other old

Table 1
Data matrix of a typical source monitoring experiment. Rows represent presentation during learning, columns denote the response of the participants, the cells contain raw frequencies

	Participants' response		
Actual source during learning	"Source A"	"Source B"	"New"
Source A	f_{AA}	f_{AB}	f_{AN}
Source B	f_{BA}	f_{BB}	f_{BN}
New	f_{NA}	f_{NB}	f_{NN}

sources after an item has been detected as an old item in the source memory test. And the detectability means the ability to detect an old source item in the test.

The most frequently used method for the data analysis is to compute three measures for each subject as shown in equations 1, 2 and 3: hits (H), indicating the rate at which the subject can detect old items correctly; false alarms (F), indicating the rate at which the subject incorrectly reports a distracter item as an old item; and identification-of-origin scores (I), referring to the rate at which the subject discriminates the exact source from all the responded old sources. The equations of these three rates are shown as follow in terms of the frequencies presented in Table 1.

$$H = \frac{(f_{AA} + f_{AB}) + (f_{BA} + f_{BB})}{f_{A.} + f_{B.}} \tag{1}$$

$$F = \frac{f_{NA} + f_{NB}}{f_{N.}} \tag{2}$$

$$I = \frac{f_{AA} + f_{BB}}{(f_{AA} + f_{AB}) + (f_{BA} + f_{BB})} \tag{3}$$

However, about ten years after the concept of source monitoring had been proposed and a multitude of studies had been done, Batchelder and Riefer (1990) noted that there was not a generally accepted measure of the quantities reported in the source-monitoring experiments. In other words, there was not any substantive model to analyze the data of the contingency table obtained from the source-monitoring experiments (see Table 1). For example, the generally used model depicted in equation 1, 2, and 3 fail to look into the internal cognitive processes such that they cannot distinguish whether the subject really recognizes the exact old source or answers correctly by guessing, when the subject reports an exact old source (e.g., report source A as source A). Therefore, Batchelder and Riefer proposed MPT models for source monitoring experiments as a substantively quantitative measurement tool for the memory retrieving processes during source monitoring experiment tasks.

Because the response frequencies in source monitoring experiments can be considered as multinomially distributed, it is assumed there are finite numbers of observable categories, $C_1, C_2, ..., C_J$, and there are N total observations. Then n_j is defined as the number of observations in C_j, and $D = (n_1, ..., n_j, ..., n_J)$ is defined as the data vector of observations for the model. The joint distribution of the data D can be represented by the general multinomial model

$$P(D;\ p_1, ..., p_J) = n! \prod_{j=1}^{J} \frac{p_j^{n_j}}{n_j!}, \tag{4}$$

where p_j is the probability that an observation falls into C_j if the data observations are mutually independent and identically distributed (i.i.d.), and $n = \sum_{j=1}^{J} n_j$. The general model has the parameter space $G_j = \left\{ p = (p_1, ..., p_J) | 0 \leq p_j \leq 1, \sum_{j=1}^{J} p_j = 1 \right\}$. In addition, a substantive MPT model assigns a parameter to each cognitive event that represents the probability of that event occurring. These events are organized hierarchically according to psychological assumptions or theories, from the very first node to the last, in a tree structure.

Every information source has an MPT model that represents the processing steps (by the parameters) and the categories of the subject's responses. For example, for source A, the first parameter (D_A) in the model is assumed to represent the probability of detecting this source as an old source. Because the detection probabilities for different sources may vary, D_B may be different from D_A. The next step after detection is discrimination with the parameter d_i as its probability if the subject successfully detects old items, or bias with the parameter b otherwise.

If the subject can detect and discriminate an old item successfully, the response is absolutely correct and this response falls in the cell f_{AA} for source A and in the cell f_{BB} for source B in Table 1. If the subject fails in the detecting or discriminating steps, he or she may guess. And if the subject is “lucky” enough, he or she is still be able to report correctly (e.g., first, correctly guess that the item is an old item and, secondly, correctly guess its type).

This set of MPT models is called one high threshold (1HTH) model; because in this set of MPT models, only the trees for “old” source items have detection and discrimination steps, and the tree for “new” source items (distractors) does not have detection and discrimination steps. In contrast, the new items (distractors) are assumed either to be responded to as old items by bias or as new items without bias.

Figure 1 presents the structure of MPT models for source monitoring and the meaning of their parameters. There are 7 parameters in this set of models, with 6 degrees of freedom (3×3 data table with 3 fixed marginal frequencies). Hence, this 7-parameter model is over saturated, and the parameters cannot be uniquely estimated, due to the insufficient degree of freedom in the data, unless we eliminate at least one parameter (e.g., we may equate a parameter with another). Figure 2 shows the 6 sub-models. In 6a, 6b and 6c submodels, two parameters are merged into one, based on the hypothesis that the detection rates, the discrimination rates, or the guessing rates of the two sources are equal, respectively. Likewise, 5-parameter submodels combine another pair of parameters. This paradigm provides 7 submodels corresponding to different psychological hypotheses that allow us to test the fit of each sub-model.

The MPT models for source monitoring (Batchelder and Riefer 1990) use graphical representation to illustrate the plausible cognitive procedure in the source monitoring test and explicitly separate the frequencies (including those in

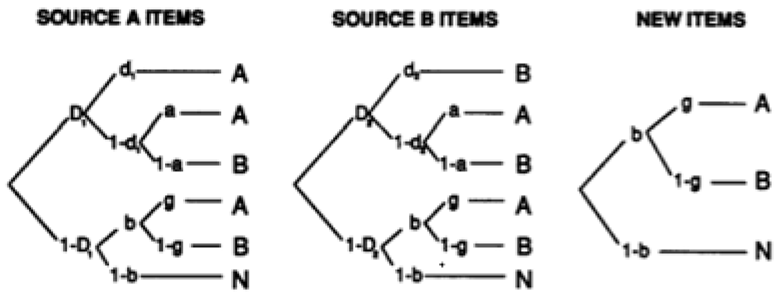


Figure 1
The seven-parameter, joint multinomial model for source monitoring. ($D_1 =$ detectability of the Source A items; $D_2 =$ detectability of the Source B items; $d_1 =$ source discriminability for the Source A items; $d_2 =$ source discriminability for the Source B items; a = guessing that a detected but nondiscriminated item belongs to Source A; b = bias for responding "old" to a nondetected item; g = guessing that a nondetected item belongs to Source A.)

the same cell in the data table) to hierarchically organized origins. For example, as introduced previously, equation 3 cannot separate real discrimination from guessing. When considering the difference between real discrimination and guessing, f_{AA} in equation 3 can be rewritten as: $f_{AA}((D_1d_1) + D_1(1 - d_1)a + (1 - D_1)bg)$. Similarly, f_{BB}, f_{AB} and f_{BA} in equation 3 cannot separate frequencies from plausibly different origins while MPT models separate these origins into different branches. The MPT models provide an approach to measuring the cognitive processes in source monitoring tasks and testing hypotheses of different submodels under various situations, and they have been applied to source monitoring analyses more and more.

1.2 Statistical Theories of MPT Models

In addition to a substantive model for human memory, the MPT model is also a statistical model for categorical observations. I will provide some background about the statistical theories related to MPT model parameter estimation and hypothesis testing.

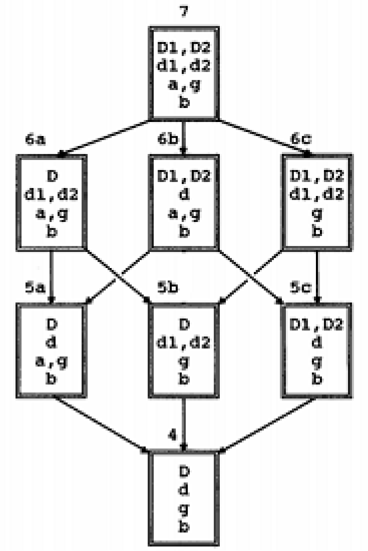


Figure 2
Nested Hierarchy for The Eight Versions of The Multinomial Model Depicted in Figure 1

Let us consider the following case in which two coins are flipped for one trial each and the final result is recorded. There are 4 observed categories: 2 heads (HH), 2 tails (TT) and 1 head followed by 1 tail (HT), or 1 tail followed by 1 head (TH). The category frequencies are represented by $D = (n_1, n_2, n_3, n_4)$, and the probabilities of these outcomes are represented by b_1, b_2, b_3, and b_4 respectively. The parameter vector is denoted by $\Theta = (\Theta_1, \ldots \Theta_s \ldots \Theta_S) \in \Omega$, where Ω is the parameter space, and $\Theta_s = (\theta_{s1}, \ldots, \theta_{sk} \ldots, \theta_{sK_s})$ refers to the K_s parameters in a group (under a same parent node), indicating the probability of the outcomes of each event. In the coin-flipping example, due to the binomial outcomes of each event, there are two parameters (e.g., p and $1-p$) in a group, and only one is independent. Note that from now on the notations above are for all the coin-flipping examples, unless explicitly indicated. Figure 3 illustrates this procedure. The frequencies of the final results follow a multinomial distribution with 4 categories and the probabilities of these outcomes are:

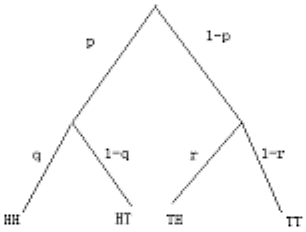


Figure 3
The coin-flipping trials. p = the probability that coin 1 gets a head, q = the probability that coin 2 gets a head if coin 1 gets a head, r = the probability that coin 2 gets a head if coin 1 gets a tail.

$$b_1 = pq, \tag{5}$$

$$b_2 = p(1-q), \tag{6}$$

$$b_3 = (1-p)r, \tag{7}$$

$$b_4 = (1-p)(1-r). \tag{8}$$

To estimate the parameters in the model in Figure 3, we can use the model's likelihood function and plug in the branch probabilities:

$$L(\Theta; D) = \frac{n!}{n_1!n_2!n_3!n_4!} b_1^{n_1} b_2^{n_2} b_3^{n_3} b_4^{n_4}, \tag{9}$$

$$\begin{aligned} L(\Theta; D) &= \frac{n!}{n_1!n_2!n_3!n_4!} (pq)^{n_1} (p(1-q))^{n_2} ((1-p)r)^{n_3} ((1-p)(1-r))^{n_4} \\ &= \frac{n!}{n_1!n_2!n_3!n_4!} p^{(n_1+n_2)} (1-p)^{(n_3+n_4)} q^{n_1} (1-q)^{n_2} r^{n_3} (1-r)^{n_4}. \end{aligned} \tag{10}$$

The likelihood function indicates the likelihood of obtaining the observed data, given the model. Hence the estimates of the parameters that maximize $L(\Theta; D)$ guarantee the maximum likelihood of obtaining the observed data. This estimation method is called maximum likelihood estimation (MLE), which is the most popular approach to parameter estimation. For equation 10, the MLEs of p, q, r are the simultaneous solutions of three equations such that:

$$\frac{\partial(L(\Theta; D))}{\partial p} = \frac{\partial(L(\Theta; D))}{\partial q} = \frac{\partial(L(\Theta; D))}{\partial r} = 0, \tag{11}$$

and

$$\begin{aligned} \frac{\partial^2(L(\Theta; D))}{\partial p^2} < 0 \\ \frac{\partial^2(L(\Theta; D))}{\partial q^2} < 0 \\ \frac{\partial^2(L(\Theta; D))}{\partial r^2} < 0. \end{aligned} \tag{12}$$

In practice, it is often more convenient to work with the logarithm of the likelihood function, called the log-likelihood. For example, in equation 11, we can obtain p by:

$$\begin{aligned} &\frac{\partial(\ln L(\Theta; D))}{\partial p} = 0, \\ &\frac{n_1 + n_2}{p} - \frac{n_3 + n_4}{1 - p} = 0, \\ &p = \frac{n_1 + n_2}{n_1 + n_2 + n_3 + n_4}. \end{aligned} \tag{13}$$

However, MLE may encounter difficulties when the likelihood is not an explicit form in which all the branch probabilities can be separated (e.g., in equation 9 and 10). Let us consider the coin-flipping example again, and suppose that for some reason we only observe the final result without the order of the events. In other words, we do not know a head is first or a tail is first if the result is a head with a tail. Under this circumstance, the frequency of HT (n_2) is combined with that of TH (n_3). This means a complete form of the likelihood function does not exist, such that the parameters p, q, r cannot be uniquely estimated.

An intuitive method to solve this problem is to assign an initial value to each parameter, which can be random, and use these initial parameter values to compute the expected frequency of each branch to “separate” the frequencies of HT from TH. In the coin-flipping example, if we assign $p^{(0)}$ as the initial value of p and $q^{(0)}$ as the initial value of q, then the expected branch frequencies of HT and TH are:

$$n_2^{(0)} = (n_2 + n_3)\frac{b_2}{b_2 + b_3} = (n_2 + n_3)\frac{p^{(0)}(1 - q^{(0)})}{p^{(0)}(1 - q^{(0)}) + r^{(0)}(1 - p^{(0)}),} \tag{14}$$

$$n_3^{(0)} = (n_2 + n_3) - n_2^{(0)}. \tag{15}$$

However, even after assigning initial values to all the parameters and writing the expected frequencies, we can only uniquely estimate at most two independent parameters because the data set only has 2 degrees of freedom (3 observed categories with fixed total frequency). So if we assume the second coin-flipping trial is independent from the first trial, q and r can be set equal, and the equation 14 will be:

$$n_2^{(0)} = (n_2 + n_3)\frac{b_2}{b_2 + b_3} = (n_2 + n_3)\frac{p^{(0)}(1 - q^{(0)})}{p^{(0)}(1 - q^{(0)}) + q^{(0)}(1 - p^{(0)}).} \tag{16}$$

By using this “complete” information we can write the likelihood function as in equation 10 and find the parameter values that maximize this expected likelihood function. Note this estimate is not the final estimate because it is based on the expected frequencies derived from any initial parameter values. Hence, we use the estimate to compute the expected frequency of each result again and find a new estimate that maximizes the new likelihood and do this iteration over and over again until the estimates tend to converge. This method is called the

Expectation-Maximization (EM) algorithm, which is proposed as an intuitive way to recursively find maximum likelihood estimates of the parameters in models when their likelihood cannot be obtained directly due to incomplete data or latent variables. The step that computes the expected frequencies is the E step, and it is followed by the M step in which the expected “complete” likelihood function is maximized (see equation 17 for incomplete likelihood and equation 18) for expected “complete” likelihood).

$$L(\Theta; D) = \frac{n!}{n_1!\,(n_2+n_3)!n_4!}\,(b_1)^{n_1}\,(b_2+b_3)^{(n_2+n_3)}\,(b_4)^{n_4}\,, \tag{17}$$

$$L_C(\Theta^{(0)}; D^{(0)}) = \frac{n!}{n_1!n_2^{(0)}!n_3^{(0)}!n_4!}\left(b_1^{(0)}\right)^{n_1}\left(b_2^{(0)}\right)^{n_2^{(0)}}\left(b_3^{(0)}\right)^{n_3^{(0)}}\left(b_4^{(0)}\right)^{n_4}. \tag{18}$$

EM algorithm was systematically introduced and generalized by Dempster, Laird, and Rubin (1977). This algorithm was originally developed as a general approach to iterative computation of maximum-likelihood estimates in models consisting of incomplete data. The name “EM” comes from its combination of an expectation step (“E” step), followed by a maximization step (“M” step). The EM algorithm requires an initial value for each parameter for the first E step that computes the expectation of the “missing” value and uses these expected values to write the “complete” likelihood (or log-likelihood) function for all the parameters, and then computes the parameter values that maximize this likelihood. After the first iteration, the newly obtained parameter values will be used in the next iteration of the “E” step, followed by the next of the “M” step. This is an iterative computation that will not terminate unless the difference (Euclidean distance) of the values of the parameter vector in two iterations is less than a certain criterion, say 10^{-20} (the convergence criterion). Suppose $L(\Theta; D, z)$ is a likelihood function where Θ is the parameter vector, D is the observed data, and z represents the unobserved latent data or missing values. The MLE of $L(\Theta; D, z)$ is determined by the likelihood of the observed data $L(\Theta; D)$. The MLE can be obtained by applying the EM algorithm as follows:

E step computes the expected value of the "missing" data under the current estimate of the parameter vector $\Theta^{(t)}$ and writes the "complete" log likelihood function $Q(\Theta|\Theta^{(t)})$, based on the observed data and the computed expectations of the "missing" data:

$$Q(\Theta|\Theta^{(t)}; D) = E_{Z|D,\Theta^{(t)}}[log\ L(\Theta; D, Z)]. \tag{19}$$

M step finds the parameter vector $\Theta^{(t+1)}$ maximizing the expected value obtained in the E step:

$$\Theta^{(t+1)} = \arg\max_{\Theta} Q(\Theta|\Theta^{(t)}). \tag{20}$$

After comparing the $\Theta^{(t+1)}$ and $\Theta^{(t)}$, the algorithm will use the $\Theta^{(t+1)}$ in the next E step, if the difference of these two parameter vectors is greater than the error criterion. Otherwise, the algorithm terminates.

The MLE methods interpret data and estimate parameters from the perspective of classic statistics in which the sample data points are considered as independent and identically distributed (i.i.d). However, this may not be true. Again, in the coin-flipping example, if the estimated values of p, q, and r represent our belief in the probability that an event occurs, this belief may be impacted by previous knowledge or prior belief. This indicates that our belief in the probability may not be a constant but may vary as a variable. Based on this assumption, the probability of a parameter should be determined by the prior knowledge and, of course, the data. The statistical inference that takes into account prior information is called the Bayesian inference. The Bayesian inference is derived from the concept of Bayesian probability, the basic idea of which is that any given probability should be a conditional probability (posterior probability), impacted by the prior probability. Therefore information obtained is connected with prior information and will influence the prediction. Bayesian inference can be considered as an alternative perspective for research. The two most important differences between Bayesian and traditional Frequentists' perspectives are (1) whether prior knowledge about the studied objects is involved, and (2) whether the estimate of a parameter is a fixed value or a distribution (Carlin and Louis 2009). In Bayesian probability theory, given observed data and a hypothesis, the posterior probability is proportional to the product of the likelihood function and the prior probability. The likelihood function represents the information from the data and the model, while the prior specifies the hypothesis before the data was observed:

$$P(\Theta|D) = \frac{P(D|\Theta)\ Pr(\Theta)}{Pr(D)} \quad , \tag{21}$$

where Θ is a parameter vector and D is the data. $Pr(\Theta)$ is the prior probability of Θ, and $P(D|\Theta)$ is the conditional probability of observing the data given Θ, namely, $P(D|\Theta)$ is the likelihood. $P(D)$ is the marginal probability of D, and finally $P(\Theta|D)$ is the posterior probability of Θ. The meaning of $P(\Theta|D)$ is the probability that the hypothesis is true, given the data and the previous belief about Θ (the prior). So equation (21) can be rewritten as:

$$P(\Theta|D) = \frac{P(D|\Theta)\ Pr(\Theta)}{\sum Pr(D|\theta_i) Pr(\theta_i)} \quad , \tag{22}$$

where θ_i is every single possible value of Θ if the distribution of Θ is discrete, or

$$P(\Theta|D) = \frac{P(D|\Theta)\ Pr(\Theta)}{\int_{\Omega} Pr(D|\tilde{\Theta}) Pr(\tilde{\Theta}) d\tilde{\Theta}} \quad , \tag{23}$$

where Ω is the parameter space, if the distribution of Θ is continuous (Hoff 2009). Therefore, the most important components of the Bayesian formula are the prior distribution and the likelihood function. Again, consider the coin-flipping example introduced previously. Here, the Bayesian inference for the posterior of the parameter vector is: $P(\Theta|D) = \frac{P(D|\Theta)\ Pr(\Theta)}{\int_{\Omega} Pr(D|\tilde{\Theta}) Pr(\tilde{\Theta}) d\tilde{\Theta}}$, and $P(D|\Theta)$ here is the likelihood function $L(\Theta; D)$ as given in equation (9), and $Pr(\Theta)$ is a prior distribution of the independent parameter vector $\Theta = (p, q, r)$ assigned by the researcher (say, use a beta distribution $B_{\Theta}(\alpha_{\Theta}, \beta_{\Theta})$, as shown in Figure 4).

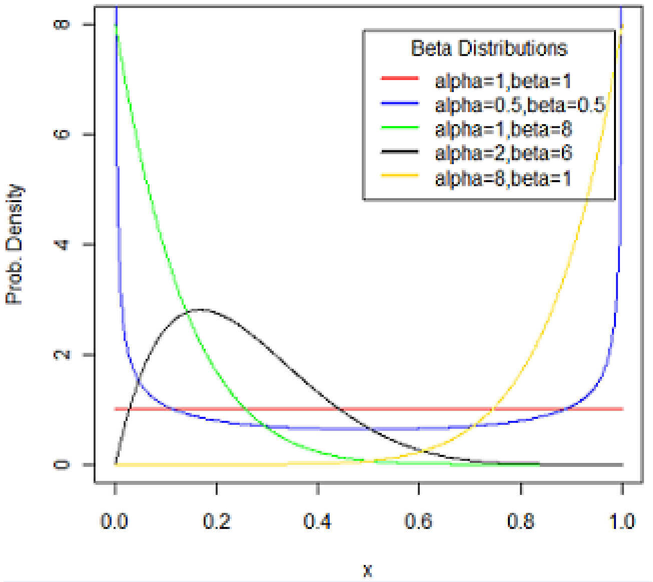


Figure 4
Beta distribution. $B(\alpha, \beta)$ has two parameters α and β.

Finally, $\int_\Omega Pr(D|\tilde{\Theta})Pr(\tilde{\Theta})d\tilde{\Theta}$ is the integration of the probabilities of the observed data given the range of the parameter vector (which, here, is from 0 to 1). Therefore, the Bayesian inference equation for the coin-flipping example is:

$$P(\Theta|D) = \frac{b_1^{n_1} b_2^{n_2} b_3^{n_3} b_4^{n_4} B_\Theta(\alpha_\Theta, \beta_\Theta)}{\int_\Omega b_1^{n_1} b_2^{n_2} b_3^{n_3} b_4^{n_4} B_\Theta(\alpha_\Theta, \beta_\Theta)\ d\Theta}. \tag{24}$$

where $Be(\alpha, \beta; \theta_i)$, $i = 1, 2, 3$ is defined in equation 25:

$$\begin{aligned} Be(\alpha, \beta; \theta_i) &= \frac{1}{B(\alpha, \beta)} \theta_i^{\alpha-1}(1-\theta_i)^{\beta-1} \\ &= \frac{\theta_i^{\alpha-1}(1-\theta_i)^{\beta-1}}{\int_0^1 u^{\alpha-1}(1-u)^{\beta-1} du}. \end{aligned} \tag{25}$$

If we plug in equations (5)–(8),

$$\begin{aligned} P(\Theta|D) &= \frac{p^{(n_1+n_2)}(1-p)^{(n_3+n_4)} q^{n_1}(1-q)^{n_2} r^{n_3}(1-r)^{n_4}}{\int_0^1 \int_0^1 \int_0^1 p^{(n_1+n_2)}(1-p)^{(n_3+n_4)} q^{n_1}(1-q)^{n_2} r^{n_3}(1-r)^{n_4}} \\ & \quad \frac{Be(\alpha_1, \beta_1) Be(\alpha_2, \beta_2) Be(\alpha_3, \beta_3)}{Be(\alpha_1, \beta_1) Be(\alpha_2, \beta_2) Be(\alpha_3, \beta_3) dpdqdr}, \end{aligned} \tag{26}$$

and after simplifying equation (26), we have:

$$\begin{aligned}
P(\Theta|D) &= \frac{1}{B(\alpha_1,\beta_1)}\frac{p^{n_1+n_2+\alpha_1-1}(1-p)^{n_3+n_4+\beta_1-1}}{Be(n_1+n_2+\alpha_1,n_3+n_4+\beta_1)} \\
&\quad \frac{1}{B(\alpha_2,\beta_2)}\frac{q^{n_1+\alpha_2-1}(1-q)^{n_2+\beta_2-1}}{Be(n_1+\alpha_2,n_2+\beta_2)} \\
&\quad \frac{1}{B(\alpha_3,\beta_3)}\frac{r^{n_3+\alpha_3-1}(1-r)^{n_4+\beta_3-1}}{Be(n_3+\alpha_3,n_4+\beta_3)} \qquad (27) \\
&= \frac{B(\alpha_1'-1,\beta_1'-1)Be(\alpha_1'-1,\beta_1'-1)}{B(\alpha_1,\beta_1)Be(\alpha_1',\beta_1')} \\
&\quad \frac{B(\alpha_2'-1,\beta_2'-1)Be(\alpha_2'-1,\beta_2'-1)}{B(\alpha_2,\beta_2)Be(\alpha_2',\beta_2')} \\
&\quad \frac{B(\alpha_3'-1,\beta_3'-1)Be(\alpha_3'-1,\beta_3'-1)}{B(\alpha_3,\beta_3)Be(\alpha_3',\beta_3')}, \qquad (28)
\end{aligned}$$

where $\alpha_1' = n_1 + n_2 + \alpha_1$, $\beta_1' = n_3 + n_4 + \beta_1$, $\alpha_2' = n_1 + \alpha_2$, $\beta_2' = n_2 + \beta_2$, $\alpha_3' = n_3 + \alpha_3$, $\beta_3' = n_4 + \beta_3$. These equations indicate the posterior distribution of the parameters is still in the beta distribution family when the prior distribution is conjugate with the likelihood function, and they also illustrate how prior information impacts the posterior distribution.

Although the equation of Bayesian inference is simple, the real computation may be quite difficult because of the integration in the equation, especially when there are many parameters, or there are latent variables and incomplete data. Therefore, the researchers developed an approximation method named Marchov chain Monte Carlo (MCMC), which is implemented through specific algorithms, such as the Gibbs sampler and the Metropolis algorithm (Hoff 2009) to obtain the approximation of the posterior distribution. (I will introduce the details of how posterior distribution is approximated in section 7.) However, in previous decades, the use of the Bayesian approach was limited due to the insufficient computation power of computers.

Nowadays, computation power has been greatly improved, which makes it possible to estimate parameters and select models using the Bayesian method. Since the Bayesian inference provides a completely different context in which to think about statistics, and we can interpret the data in a quite different way, it is meaningful to compare it with classic statistical approaches. There have been a lot of theoretical papers that focus on the comparison of Bayesian and classic inferences (Carlin and Louis 2009). In this study I propose to conduct the comparison specifically on MPT models, because (1) there has not been a study that applies Bayesian inference to MPT model analyses, and (2) there has not been a study that systematically compares the similarities, differences, and advantages/disadvantages of Bayesian and classic inference for MPT models. To conduct a concrete comparison of these two approaches and illustrate the application of Bayesian methods in psychological research, I introduce a typical memory experiment and related model test.

Chapter 2

Analyses

Following the theoretical introduction of the classic and the Bayesian inferences, I will conduct comparisons between these two inferences. To do that, I will use the specified forms of the MPT models in source monitoring experiments. Then I will introduce how I implement the model estimations on a computer. Finally, I will compare the estimation results from point estimates to their advantages and disadvantages, respectively.

2.1 Mathematical Representation of MPT Models

Hu and Batchelder (1994) developed the following mathematical expressions to represent the MPT models. Let $C_1, ..., C_j, ..., C_J$ denote the observable categories, and $B_{1j}, ..., B_{ij}, ..., B_{I_j j}$ denote the collection of branches whose ending nodes belong to category C_j. In the MPT models for source monitoring (see Figure 1), C_j represents the probability of a categorical response such as A, B or N; B_{ij} represents the probability of a branch in the model such as the first branch of answering A. Denote the parameters in a group (under a same parent node) by $\Theta_s = (\theta_{s1}, \ldots, \theta_{sk} \ldots, \theta_{sK_s}) \in \Omega_s = \left\{[0,1]^S | \sum_{k=1}^{K_s} \theta_{sk} = 1\right\}$, and there are S groups, namely $\Theta = (\Theta_1, \ldots \Theta_s, \ldots \Theta_S) \in \Omega = \{\prod_{s=1}^{S} \Omega_s\}$, where Ω is the parameter space, K_s is the number of the parameters nested in the s_{th} group, and $0 \leq \theta_{sk} \leq 1$. To estimate the parameters, the first step is to write the mathematical form for the MPT models. In the MPT models, the most basic unit is the link probability $L_{ijl} = (L_{ij1}, ..., L_{ijl}, ..., L_{ijL_{ij}})$, where $l = (1, ..., l_{ij}..., L_{ij})$ is the lth link on the branch B_{ij}. A link in the MPT models represents the transition probability from one cognitive step to the next. The links then form the branch probability B_{ij} that is the probability from the root node to an ending node of the tree. For example, in the MPT models for source monitoring, the first link in the tree A can be represented as $L_{111} = D_1$, and B_{ij} can be written as the product of the links on this branch, such as $B_{11} = D_1 d_1$. To use a generalized form and facilitate computing, we can present any link probability as the product of all the parameters with their powers:

$$L_{ijl} = \prod_{s=1}^{S} \left(\prod_{k=1}^{K_s} \theta_{sk}^{\alpha_{ijlsk}} \right), \tag{29}$$

$$\sum_{k=1}^{K_s} \theta_{sk} = 1, \tag{30}$$

where the α_{ijlsk} is the summation over links of non-negative integer exponents on θ_{sk}. For instance, in the MPT models for source monitoring, the choices of each step are binary, which means there is only one independent parameter under every parent (θ_s and $1 - \theta_s$). Thus, we obtain the link probability $L_{ijl} = \prod_{s=1}^{S} \theta_s (1 - \theta_s)$. Specifically, for example, the first link under the root node of the tree A can be written as $L_{111} = D_1^1(1-D_1)^0 d_1^0(1-d_1)^0 a^0(1-a)^0 b^0(1-b)^0 g^0(1-g)^0 D_2^0(1-D_2)^0 d_2^0(1-d_2)^0 = D_1$. Likewise, we can write a generalized form of the branch probabilities:

$$p_{ij}(\Theta) = Pr(B_{ij};\ \Theta) = c_{ij} \prod_{s=1}^{S} \left(\prod_{k=1}^{K_s} \theta_{sk}^{\alpha_{ijsk}} \right), \tag{31}$$

$$\alpha_{ijsk} = \sum_{l=1}^{L_{ij}} \alpha_{ijlsk}, \tag{32}$$

where $p_{ij}(\Theta)$ is the i_{th} branch probability in the j_{th} category within a tree, and c_{ij} is the product of positive constants on the links in the event that some parameters are set as constants. The use of α_{ijsk} here is to represent the parameters that repeatedly appear on a branch. For example, in the previous coin-flipping example, if the parameters $p = q$, then the power α for p is 2 on B_{11} because $B_{11} = p^2$. Researchers have discussed that the possibility of the constant c_{ij} can arise from the restrictions on some parameters set by the model's hypothesis (Hu and Batchelder 1994; Batchelder and Riefer 1986). In the MPT models, for example, the first branch answering A in the tree A has the probability $B_{11} = D_1^1(1-D_1)^0 d_1^1(1-d_1)^0 a^0(1-a)^0 b^0(1-b)^0 g^0(1-g)^0 D_2^0(1-D_2)^0 d_2^0(1-d_2)^0 = D_1 d_1$.

At last, the category probability is the summation of the probabilities of the branches going to the same observable response category. For instance, the probability of answering source A as A is $D_1 d_1 + D_1(1 - d_1)a + (1 - D_1)bg$. Also, this summation can be written in a generalized form as in equation (31)

$$p_j(\Theta) = Pr(C_j;\ \Theta) = \sum_{i=1}^{I_j} \left[c_{ij} \prod_{s=1}^{S} \left(\prod_{k=1}^{K_s} \theta_{sk}^{\alpha_{ijsk}} \right) \right], \tag{33}$$

where

$$\sum_{j=1}^{J} p_j(\Theta) = 1$$

for all $\Theta \in \Omega$. The equations above depict the probability mass functions (PMF) of the MPT models, and the likelihood functions can be obtained from the PMF.

2.2 Likelihood Functions of MPT Models

The previous chapter introduces two inference approaches (classic and Bayesian) used to estimate parameters and demonstrates the importance of the likelihood function, which is obtained from the observed data and the MPT probability function. As a concrete example, suppose we have a 3×3 data table in which the frequencies are $n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9$, and their summation is N. The likelihood function for this data given the model is:

$$L = N! \frac{p_1^{n_1} p_2^{n_2} p_3^{n_3} p_4^{n_4} p_5^{n_5} p_6^{n_6} p_7^{n_7} p_8^{n_8} p_9^{n_9}}{n_1! n_2! n_3! n_4! n_5! n_6! n_7! n_8! n_9!}.$$

Therefore, given the frequency of observations in a category is n_j, the likelihood function for the MPT models is:

$$L(\Theta;\ < n_j >_{j=1}^{J}) = N! \prod_{j=1}^{J} \frac{[p_j(\Theta)]^{n_j}}{n_j!}, \tag{34}$$

where $p_j(\Theta)$ are given by equation 33, and N is the total number of the observations.

Because of the difficulty of directly obtaining the maximum likelihood estimates (MLEs) when there exist incomplete information (we only know the combined frequency of each category but not each branch), an indirect method such as an iterative algorithm must be recruited. If one had the “missing” branch frequencies m_{ij} (although this is impossible in real experiments),

$$D =<< m_{ij} >_{i=1}^{I_j}>_{j=1}^{J},$$

then the likelihood function with complete data is:

$$L(\Theta;\ D) = N! \prod_{j=1}^{J} \prod_{i=1}^{I_j} \frac{[p_{ij}(\Theta)]^{m_{ij}}}{m_{ij}!}, \tag{35}$$

where $p_{ij}(\Theta)$ is given in equation 31. Moreover, Dempster, Laird, and Rubin (1977) proved that a cycle of the EM does not decrease the likelihood function. This implies that the EM algorithm may be an applicable approach for searching maximum value (at least local maxima) of equation 34.

2.3 The EM Algorithm for MPT Models

As introduced previously, for MPT models, the E step is to get the conditional expected frequency of each branch (m_{ij}) given the value of Θ and the observed category frequency n_j. For example, the expected frequency for the first branch (answering A in tree A) at the first step is

$$m_{11}^{(1)} = n_1 \frac{D_1^{(0)} d_1^{(0)}}{D_1^{(0)} d_1^{(0)} + D_1^{(0)}(1 - d_1^{(0)}) a^{(0)} + (1 - D_1^{(0)}) b^{(0)} g^{(0)}},$$

where $D_1^{(0)}$ denotes the initial value of D_1, and so on. The equation for the **E** step of the MPT models is:

$$m_{ij}(\Theta) = E(M_{ij} | n_j;\ \Theta) = \frac{n_j p_{ij}(\Theta)}{p_j(\Theta)}, \tag{36}$$

where M_{ij} is the random variable denoting the counts in branch B_{ij}, and $p_{ij}(\Theta)$ and $p_j(\Theta)$ are given by equation 31 and equation 33, respectively.

The **M** step, after the E step, obtains the values of Θ that maximize the likelihood function $L(\Theta;\ < n_j >_{j=1}^{J})$ that is based on the expected frequencies computed in the last E step. In MPT models for source monitoring, for example,

$$D_1^{(1)} = \frac{m_{11}^{(1)} + m_{21}^{(1)} + m_{12}^{(1)}}{m_{11}^{(1)} + m_{21}^{(1)} + m_{12}^{(1)} + m_{31}^{(1)} + m_{22}^{(1)} + m_{13}^{(1)}}$$

$m_{ij}^{(1)}$ denotes the first-round expected frequencies of the branches on tree A. The equation for the M step of the MPT models is:

$$\hat{\theta}_{sk} = \frac{\sum_{j=1}^{J} \sum_{i=1}^{I_j} m_{ij} \alpha_{ijsk}}{\sum_{j=1}^{J} \sum_{i=1}^{I_j} \sum_{k=1}^{K_s} m_{ij} \alpha_{ijsk}}. \tag{37}$$

To write this EM algorithm as the form in which $\Theta^{(n+1)}$ is presented as a function of $\Theta^{(n)}$, we have:

$$\theta_{sk}^{(n+1)} = \frac{\sum_{j=1}^{J} \sum_{i=1}^{I_j} m_{ij}^{(n)} \alpha_{ijsk}}{\sum_{j=1}^{J} \sum_{i=1}^{I_j} \sum_{k=1}^{K_s} m_{ij}^{(n)} \alpha_{ijsk}} \tag{38}$$

In the MPT models for source monitoring experiments, the cognitive steps are binary, and each parameter group only has one independent parameter, so we can simplify the indices in equation 38 as:

$$\theta_{s}^{(n+1)} = \frac{\sum_{j=1}^{J} \sum_{i=1}^{I_j} m_{ij}^{(n)} \alpha_{ijs}}{\sum_{j=1}^{J} \sum_{i=1}^{I_j} \sum_{s=1}^{S=2} m_{ij}^{(n)} \alpha_{ijs}} \tag{39}$$

Specifically, the EM algorithm proceeds as follows:

Start with any initial value $\Theta^{(0)} \in \Omega$;

For $(n = 0,\ ...,\ N)$, repeat:

Step 1: Compute the expected frequencies of each branch $m_{ij}^{(n)}$, $n = 0$ in the first iteration;

Step 2: Find the values of $\Theta^{(n+1)}$ that maximize the expected likelihood function (see equation 35);

Step 3: Compare the difference of $\Theta^{(n+1)}$ and $\Theta^{(n)}$, and return to step 1 if the difference is not less than the criterion (e.g., 10^{-10}).

This E and M iteration will be running over and over again, until the difference (Euclidean distance) of two estimates is less than the criterion, which is said to be convergent. To ensure that the EM algorithm for MPT models can finally obtain a unique estimate of Θ, which implies the result will converge, (Hu and Batchelder 1994) proved that provided any initial value $\Theta_s^{(0)}$, the EM algorithm is a convergent procedure for MPT models, although this is not a general character of the EM algorithm.

2.4 The Bayesian Inference and Its Algorithm for MPT Models

According to the Bayesian rule (see equation 23), we start with an initial probability distribution for the parameter vector Θ, which is called prior distribution $Pr(\Theta)$ and use the prior and the likelihood to obtain their univariate posteriors or joint multivariate posterior.

Assuming we have no prior information about the Θ, a non-informative prior can be employed. Here I apply a B(1, 1) prior to each of the parameters, which is subjective but based on a commonly used rule when the parameter is a probability (Karabatsos 2006).

Therefore, we can rewrite equation 23 as:

$$P(\Theta|D) = P(D|\Theta) \times \frac{Pr(\Theta)}{Pr(D)} = \frac{P(D|\Theta)\ Pr(\Theta)}{\int_{\Omega} P(D|\tilde{\Theta})Pr(\tilde{\Theta})d\tilde{\Theta}}, \tag{40}$$

where $P(D|\Theta)$ is the likelihood function of the MPT models, and Pr(D) is the marginal likelihood. In other words, Pr(D) is the probability distribution of obtaining data set D regardless of the value of Θ. Hence equation 40 can be rewritten as:

$$P(\Theta|D) \propto P(D|\Theta)Pr(\Theta), \tag{41}$$

which means the posterior is proportional to the likelihood times the prior. Specifically, in MPT models for source monitoring, if a non-informative prior beta(1,1) is assigned, equation 41 can be written as $P(\Theta|D) \propto L(D;\Theta)B(1,1)$, where $L(D;\Theta)$ is the likelihood function given in equation 34.

However, when the combination of the prior and the likelihood is complex due to their forms or the numbers of parameters, the calculation of the multiple univariate posteriors and multivariate posterior may be difficult or impossible. To overcome the computation difficulty, we can generate random sample values of the parameters from their (candidate) posterior distributions; all of these posterior statistics of interest can be approximated to an arbitrary degree of precision using the Monte Carlo approximation method (Hoff 2009). Because the process of generating the random samples is a Marchov chain process, this approximation is called Marchov chain Monte Carlo (MCMC) (Hoff 2009).

To implement the MCMC computing, there are two main algorithms, the Metroplis (or Metropolis-Hasting) algorithm and the Gibbs sampler. The Metropolis algorithm is a rejection algorithm that can be applied to arbitrary prior distributions. The original Metropolis algorithm originated because, although the joint posterior is too complicated to sample from (because we probably cannot find a familiar distribution that exactly fits the posterior), it is possible to sample from a candidate-generating distribution $q(\Theta^*|\Theta^{(t-1)})$ that has the same parameter space for Θ and can satisfy $q(\Theta^*|\Theta^{(t-1)}) = q(\Theta^{(t-1)}|\Theta^*)$ (which denotes the transition probability from $\Theta^{(t-1)}$ to Θ^* equals to its reverse transition probability). Whether we accept a sample point Θ^* depends on whether Θ^* increases the density of the joint posterior distribution when compared to the previous density. If it does, we accept this point for the posterior. Otherwise, we accept this point by the ratio of its density divided by that of the previous point or keep $\Theta^{(t-1)}$.

Suppose our goal is to draw samples from some distribution $p(\Theta)$, where $p(\Theta) = f(\Theta)/K$, $f(\Theta)$ is the posterior, and K is the normalizing constant that may not be known or be very difficult to compute. The Metropolis algorithm proceeds as follows:

Start with any initial value Θ_0 satisfying $f(\Theta_0) > 0$;

For $(t = 1,\ ...,\ T)$, repeat:

Step 1: Using current Θ value, sample a candidate point Θ^* from a candidate-generating (or say proposal) distribution $q(\Theta)$ that satisfies $q(\Theta_1, \Theta_2) = q(\Theta_2, \Theta_1)$;

Step 2: Given the candidate point Θ^*, calculate the ratio of the density of $p(\Theta)$ at Θ^* and current Θ_{t-1},

$$\alpha = \frac{p(\Theta^*)}{p(\Theta_{t-1})} = \frac{f(\Theta^*)}{f(\Theta_{t-1})}$$

(because we are considering the ratio of $p(\Theta)$ under two different values, the normalizing constant K cancels out);

Step 3: If the ratio is $\alpha \geq 1$, accept the candidate point (set $\Theta_t = \Theta^*$). Otherwise, either accept Θ^* with the probability α or reject with the probability $1 - \alpha$;

The first few hundred iterations are the burn-in period, which find the stationary distribution of the posterior. After enough burn-in period (say, k steps), the chain approaches its stationary distribution and samples from the vector $(\Theta_{k+1}, \ldots, \Theta_{k+n})$, which are samples from $p(x)$.

The Metropolis-Hasting algorithm generalized the original Metropolis algorithm and does not require the condition $q(\Theta_1, \Theta_2) = q(\Theta_2, \Theta_1)$ of the proposal distribution. Therefore, the ratio α in step 2 becomes $\alpha = \frac{f(\Theta^*)q(\Theta^*, \Theta_{t-1})}{f(\Theta_{t-1})q(\Theta_{t-1}, \Theta^*)}$, where $q(\Theta_i, \Theta_j) = Pr(\Theta_i \rightarrow \Theta_j)$.

The Gibbs sampler is typically used for conjugate priors or other priors in which the marginal (conditional) distribution can be easily computed. The Gibbs sampler is a technique for generating random variables from the marginal distribution directly, in situations where the conditional distributions of each parameter can be acquired when all the others are fixed. This algorithm does not have to calculate the density, which is difficult to compute in complex cases. Rather than compute or approximate a (marginal) distribution directly, the Gibbs sampler allows us to effectively generate a sample sequence from this distribution without requiring its density. The Gibbs sampler can be considered as a special case of the Metropolis-Hasting algorithm, and the acceptance rate α is always 1 because we are always sampling from the real conditional posteriors instead of a proposal distribution, so it is much easier and more efficient. If the likelihood has only one parameter, which means the posterior has only one parameter, we can draw the samples directly to depict the posterior distribution given the data. In more complicated cases, such as the coin-flipping case, for example, the parameters in the posterior distribution (see equation 26) cannot be drawn simultaneously. In this case, we can start from a random set of initial values for the parameters, except the first parameter (such as $q^{(0)}, r^{(0)}$). This step is to fix all the other parameters in the posterior except the first one, and then the computer may sample a value from the posterior, indicated by $p^{(1)}$, and then use this newly sampled $p^{(1)}$ as the value of p to generate the values for q and r. Note the newest drawn parameter values will be used in the posterior for the next sampling right away. Additionally, to avoid the influence of the initial values, the first several hundred rounds are usually ignored (as burn-in period), and the subsequent samples will form the posterior distributions of the parameters. Though the Gibbs sampler is efficient and simple, it may encounter difficulties when the prior is not conjugate with the likelihood, which leads to the situation where the Gibbs sampler cannot find a proper distribution from which to draw samples.

Since our models are binary for every parameter, and we recruit a beta distribution, which is conjugate to binomial distribution, as the prior, we can use the Gibbs sampler algorithm. Suppose we have k parameters, $\Theta = (\theta_1,\ ...,\ \theta_k)'$, in our model. Like with the EM algorithm, we assume that the samples are generated from each of the complete conditional distributions $\{p(\theta_i|\theta_{j\neq i},\ y),\ i = 1,\ ...,\ k\}$ in the model, and the samples might be available directly or indirectly. In either case, the collection of full conditional distributions can uniquely determine the joint posterior distributions $\{p(\theta_i|y),\ i=1,\ ...,\ k\}$. So, given an arbitrary set of the initial values $\{\theta_2^{(0)},\ ...,\ \theta_k^{(0)}\}$, the Gibbs sampler algorithm proceeds as follows:

For $(t=1,\ ...,\ T)$, repeat:

Step 1: Draw $\theta_1^{(t)}$ from $p\left(\theta_1|\theta_2^{(t-1)},\ \theta_3^{(t-1)},\ ...,\ \theta_k^{(t-1)},\ y\right)$

Step 2: Draw $\theta_2^{(t)}$ from $p\left(\theta_2|\theta_1^{(t)},\ \theta_3^{(t-1)},\ ...,\ \theta_k^{(t-1)},\ y\right)$

$\vdots$

Step k: Draw $\theta_k^{(t)}$ from $p\left(\theta_k|\theta_1^{(t)},\ \theta_2^{(t)},\ ...,\ \theta_{k-1}^{(t)},\ y\right)$.

The k-tuple obtained at iteration t, $\{\theta_1^{(t)},\ ...,\ \theta_k^{(t)}\}$ will converge to the true joint posterior distribution $p\left(\theta_1,\ ...,\ \theta_k|y\right)$.

2.5 Programs for Implementation of The Two Algorithms

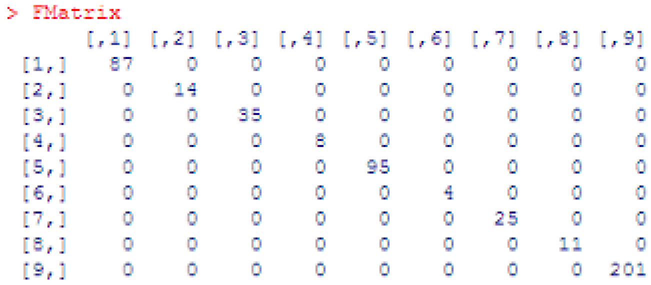
To implement the EM and Bayesian algorithms on the computer, an operative program is needed. After MPT models for source monitoring were proposed, useful implementing software packages such as GPT.exe (Hu and Philips 1999), AppleTree for Mac (Rothkegel 1999), HMMTree (Stahl and Klauer 2007), and multiTree (Moshagen 2010) were developed.

However, a new program for MPT model analyses is needed, because (1) none of the former programs can implement Bayesian analysis for MPT models due to limitations of the algorithms (such as algorithms for random sampling) and of the computer hardware, and (2) because there is a trend towards more and more statisticians collaborating and sharing the statistical tools developed by themselves in an open source developing environment. Therefore, I chose R (http://www.r-project.org/) as the platform for implementing Bayesian inference in the proposed projects in my thesis.

2.5.1 Model Representation and Implementation in R Environment

The model information, including the tree structure, parameter definition, and data sets, is restored in an Extensible Markup Language (XML) file. Also we developed a software package GPT-R to parse all the information in the XML file to R. The appendix details how the model information is stored and the functionality of the GPT-R package.

After parsed into R, the original tree is transformed to a power table. In addition, the observed frequencies are also transformed into a frequency table. Figure 5 shows the frequency table in which the frequency of each category and tree is shown on the diagonal. In this table, the first three numbers on the diagonal are the frequencies of response category A, B and N in the first source tree, and the second three numbers and last three numbers represent the observed frequencies in the source tree B and N (new items), respectively.



```
> FMatrix
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
 [1,]   87    0    0    0    0    0    0    0    0
 [2,]    0   14    0    0    0    0    0    0    0
 [3,]    0    0   35    0    0    0    0    0    0
 [4,]    0    0    0    8    0    0    0    0    0
 [5,]    0    0    0    0   95    0    0    0    0
 [6,]    0    0    0    0    0    4    0    0    0
 [7,]    0    0    0    0    0    0   25    0    0
 [8,]    0    0    0    0    0    0    0   11    0
 [9,]    0    0    0    0    0    0    0    0  201
```

Figure 5
Observed Frequency Table

For MLE estimation, we implement EM parameter estimation, equating-parameter hypothesis tests, and model goodness-of-fit tests. For Bayesian estimation, we implement parameter estimation via MCMC approximation (Metropolis algorithm), hypothesis tests, and model evaluation via

Bayesian information criterion (BIC), which I will introduce in the subsequent section.

2.5.2 Bayesian Parameter Estimation in WinBUGS

Although the Metroplis and Metropolis-Hasting algorithm implemented in R can cope with arbitrary prior, it is not easy to get a full description of the posterior sampling. So I also wrote a code for the WinBUGs (http://www.mrc-bsu.cam.ac.uk/bugs/) version, to obtain the full description of the posterior distribution including its mean, standard deviation, and median, as well as density and history trace plots, etc. The BUGS (Bayesian inference Using Gibbs Sampling) project is concerned with flexible software for the Bayesian analysis of complex statistical models using Markov chain Monte Carlo (MCMC) methods, and the WinBUGS is its Windows version. WinBUGS can be called from R with the R2WinBUGS package. Although this version can provide comprehensive description on the parameter posterior distributions, the Gibbs sampler algorithm recruited here may be incapable when coping with non-conjugate priors, as introduced previously.

2.6 Comparison Using Empirical Data

In this thesis, I use the empirical data sets in a collection of published research papers that recruit MPT models for source monitoring. The comparisons include (1) point estimates, (2) model evaluation methods, 3) estimates based on cumulative data, and 4) inference approaches.

2.6.1 Empirical Data Sets

Batchelder and Riefer (1990) classic paper of MPT models contains twelve 3 × 3 data sets from four experiments. Three are from Johnson, Foley, and Leach (1988) experiments (Table 2), five are from Harvey (1985) experiments (Table 3), two are from Saegert, Hamayan, and Ahmar (1975) experiments (Table 4), and the final two are from Rose, King, and Perez (1975) experiments (Table 5), for a total of twelve 3 × 3 tables. I use these data sets to implement MLE and Bayesian point estimates and make a basic comparison.

Table 2
Empirical 3× 3 Data Tables

	L(a)-I(s)			L(a)-I(b)			L(a)-I(a)		
	response			response			response		
Source item	L	I	N	L	I	N	L	I	N
Listen	87	8	25	74	16	45	63	13	29
Imagine	14	95	11	23	76	36	46	36	23
New	35	4	201	28	17	225	19	13	178

Note. Data are from Johnson, Foley, and Leach (1988) Experiments. Experimental conditions are as follows: L(a)-I(s) = listen to A, imagine in subject's voice; L(a)-I(b) = listen to A, imagine in B's voice; and L(a)-I(a) = listen to A, imagine in A's voice. L= listen; I = imagine; N = new.

Table 3
Empirical 3× 3 Data Tables

	Manic subjects						Schizophrenic subjects						Normal		
	NTD			TD			NTD			TD			subjects		
Source	S	T	N	S	T	N	S	T	N	S	T	N	S	T	N
Say	22	27	31	43	6	31	13	21	46	44	10	26	23	22	35
Think	7	54	19	20	15	45	4	42	34	32	8	40	9	45	26
New	4	26	50	5	9	66	6	20	54	24	7	49	7	10	63

Note. Data are from Harvey (1985) experiments. NTD = non-thought disordered; TD = thought disordered; responses are as follows: S = say; T = think; N = new.

2.6.2 Point Estimation Results and Comparison

Batchelder and Riefer (1990) tested the sub models with 6 parameters (6C), 5 parameters (5C) and 4 parameters (Figure 2) using the EM algorithm. Correspondingly, I test these sub models using Bayesian approach and report the results in the tables with their EM counterparts. The first sub model is 6C, in which there are 6 parameters: D_1, D_2, d_1, d_2, b, g. In this model, the probability of guessing that a detected but nondiscriminated item belongs to Source A is assumed to be equal to the probability of guessing that a nondetected item belongs to Source A. This hypothesis assumes that subjects guess that an item belongs to Source A at the same rate whether the item is detected as an old item or not. In this case, the parameters a and g in the 7-parameter saturated model are set as equal and represented as g. Because the estimates of Bayesian approach for the parameters are their distributions rather than single probabilities, I use the posterior mean (Karabatsos 2006) as the estimator of the posterior. I sampled 20,000 times, in which the first 500 are set as burn-in, ensuring that the

Table 4
Empirical 3× 3 Data Tables

	Sentence group response			Word group response		
Source Item	E	F	N	E	F	N
English	184	75	173	152	19	45
French	77	187	168	21	143	52
New	58	75	155	26	19	99

Note. Data are from Saegert, Hamayan, and Ahmar (1975) experiments. E = English; F = French; N = new.

Table 5
Empirical 3× 3 Data Tables

	Related sentences response			Unrelated sentences response		
Source Item	E	S	N	E	S	N
English	164	46	30	181	39	20
Spanish	46	158	36	47	173	20
New	111	107	262	102	85	293

Note. Data are from Rose, King, and Perez (1975) experiments. E = English; S = Spanish; N = new.

final estimates are not influenced by them. The parameter estimates of the EM algorithm and Bayesian methods for MPT models are shown in Table 6.

Table 6
Parameter Estimates

	EM/(Bayesian) estimation					
Condition	D_1	D_2	d_1	d_2	b	g
L(a)-I(s)	.75 (.74)	.89 (.88)	.19 (.36)	.87 (.86)	.16 (.17)	.90 (.87)
L(a)-I(b)	.60 (.59)	.68 (.67)	.59 (.56)	.68 (.66)	.17 (.17)	.62 (.62)
L(a)-I(a)	.67 (.67)	.74 (.73)	.62 (.53)	.06 (.15)	.15 (.16)	.59 (.63)

Note. D_1 = detectability of the listen items; D_2 = detectability of the imagine items; d_1 = source discriminability for the listen items; d_2 = source discriminability for the imagine items; b = bias for responding "old"; g = guessing that the item was a listen item; L(a)-I(s) = listen to A, imagine in subject's voice; L(a)-I(b) = listen to A, imagine in B's voice; L(a)-I(a) = listen to A, imagine in A's voice.

From Table 6 we derive some intuitive sense about the estimates of the two methods. Most estimates of the corresponding parameters are similar, but most of the Bayesian estimates are more centralized than that of EM. For instance, the Bayesian estimates in this table do not have extremely large (greater than .90) or small values (less than .10); they are closer to .5 when compared with those in EM estimates (see the red-colored pairs in Table 6). This may result from the fact that the estimator recruited in the Bayesian estimation is the posterior mean, which is the average of all the estimates, whereas the estimator used in the EM is the peak point of the likelihood function that is more likely to generate extreme values. The estimator used in the Bayesian approach may have an advantage in some unusual situations. For example, the frequencies on the diagonal of the data table (e.g., response of source A as "A") are generally higher than off-diagonal frequencies (e.g., response of source A as “B”). However, if the frequencies on the diagonal are lower than their off-diagonal counterparts, the Bayesian estimation can give a much more reasonable estimation than the EM does. In Table 7, data are modified according to Table 2 (condition L(a)-I(s)) by switching some frequencies on-diagonal or off-diagonal such that the frequencies on the table diagonal are lower than they usually should be. Given this mimic data (which is unusual but theoretically possible, e.g., the subjects did not follow the instructions, which may result in the failure of MPT model assumptions), the EM algorithm will push some estimates to the boundary (which is 0) to satisfy the MLE, compared with Bayesian estimation. Table 8 shows the estimates of the EM and the Bayesian approach.

Table 7
A Mimic 3$\times$ 3 Data Table with Low Diagonal Frequencies

	Response		
Source Item	L	I	N
Listen	8	87	25
Imagine	95	14	11
New	201	35	4

Note. L = Listen; I = Imagine; N = new.

Moreover, the model with the parameters estimated using the EM algorithm in Table 6 cannot be tested by the goodness-of-fit (G^2) test when the model is saturated. However, Bayesian information criterion (BIC) (Carlin and Louis 2009) can be used as the criterion to test the model fit in this case. The BIC is a criterion for model selection across a class of parametric models with different numbers of parameters. It is similar to Akaike’s information criterion (AIC) (Akaike 1973; Karabatsos 2006) in equation (42), but the penalty for additional parameters is stronger than that of the AIC.

Table 8

Parameter Estimates for The Mimic Data in Table 7

	EM/(Bayesian) estimation					
Condition	D1	D2	d1	d2	b	g
L(a)-I(s)	0	0	0	.87	.91	.69
	(.03)	(.08)	(.36)	(.26)	(.91)	(.69)

Note. D_1 = detectability of the listen items; D_2 = detectability of the imagine items; d_1 = source discriminability for the listen items; d_2 = source discriminability for the imagine items; b = bias for responding "old"; g = guessing that the item was a listen item; L(a)-I(s) = listen to A, imagine in subject's voice.

$$AIC \equiv -2\ ln\ L_{max} + 2k\ , \tag{42}$$

and

$$BIC \equiv -2\ ln\ L_{max} + k\ ln\ N\ , \tag{43}$$

where L_{max} is the maximum likelihood achievable by the model, k is the number of parameters in the model, and N is the number of data points in the experiment. In many cases, informal likelihood or penalized likelihood criteria may be feasible. Log-likelihood summaries are easy to estimate using posterior samples $\{\theta^{(g)}, g = 1, ..., G\}$, since we may think of $l \equiv logL(\theta)$ as a parametric function of interest, and subsequently compute

$$\hat{l} \equiv E[ln\ L(\theta)|y] \approx \frac{1}{G}\sum_{g=1}^{G} ln\ L(\theta^{(g)}). \tag{44}$$

BIC is used here as an overall measure of model fit to be compared across models. Unlike G^2 for goodness-of-fit, the BIC is not a criterion that can have a generic standard for determining whether or not a model can be accepted, but a relative criterion for comparing across models with different parameters to determine which model better fits the given data. In general, the model fit is better when the BIC is smaller because the BIC value is conversely related to both the likelihood and the penalty term.

Next, Table 9 shows the EM and Bayesian parameter estimates and goodness-of-fit test of Harvey (1985) experiment data, given sub model 5C.

Table 9
Parameter Estimates

	EM/(Bayesian) estimation					
Group	D_1	D_2	d	b	g	$G^2(1)$/(BIC)
Manic NTD	.39 (.36)	.62 (.59)	.51 (.55)	.37 (.39)	.17 (.18)	0.50 (58.07)
Manic TD	.53 (.51)	.29 (.25)	.43 (.51)	.18 (.21)	.69 (.67)	9.94* (66.39)
Schizophrenic NTD	.11 (.16)	.36 (.37)	.87 (.67)	.34 (.32)	.21 (.21)	0.25 (57.05)
Schizophrenic TD	.47 (.44)	.18 (.15)	.03 (.23)	.39 (.40)	.80 (.79)	0.18 (57.71)
Normal	.44 (.42)	.59 (.57)	.42 (.44)	.21 (.23)	.30 (.31)	1.20 (59.01)

Note. D_1 = detectability of the listen items; D_2 = detectability of the imagine items; d = source discriminability; b = bias for responding “old”; g = guessing that the item was a listen item; L(a)-I(s) = listen to A, imagine in subject’s voice; L(a)-I(b) = listen to A, imagine in B’s voice; L(a)-I(a) = listen to A, imagine in A’s voice.
* $p < .01$.

In addition, the parameter estimates for Saegert, Hamayan, and Ahmar (1975) experiments are shown in Table 10.

Batchelder and Riefer finally tested the experiment offered by Rose, King, and Perez (1975), and the parameter estimates of EM algorithm and Bayesian approach are presented in Table 11.

2.6.3 Comparison Based on Cumulative Data

Another feature that differentiates Bayesian inference from MLE is the use of cumulative data. Classic methods usually pool the data of similar experiments across different times, stimuli, and subjects to obtain a larger sample size and greater statistical power. However, this approach may cause some potential problems if the populations vary significantly (e.g., the variances are heterogeneous), which violates the basic assumption that variables are distributed independently and identically. This potential problem also occurs when the stimuli in different experiments are not really identical. Therefore, classic methods of data combination are based on the assumptions that the stimuli,

Table 10
Parameter Estimates

	EM/(Bayesian) estimation				
Condition	D	d	b	g	$G^2(2)$/(BIC)
Sentence group	.27 (.29)	.95 (.86)	.46 (.45)	.48 (.48)	1.55 (69.53)
Word group	.67 (.67)	.88 (.87)	.31 (.32)	.56 (.56)	.82 (60.03)

Note. D = item detectability; d = source discriminability; b = bias for responding “old”; g = guessing that the item was in English.

Table 11
Parameter Estimates

	EM/(Bayesian) estimation				
Condition	D	d	b	g	$G^2(2)$/(BIC)
Related sentences	.75 (.74)	.64 (.64)	.45 (.46)	.51 (.51)	0.64 (65.75)
Unrelated sentences	.86 (.86)	.65 (.65)	.39 (.39)	.55 (.54)	0.0003 (63.99)

Note. D = item detectability; d = source discriminability; b = bias for responding “old”; g = guessing that the item was in English.

subjects, and other experimental conditions are similar, such that the parameters in the models used in different experiments are theoretically equal. Meanwhile, Bayesian inference uses the posterior obtained in similar data to update the estimates of the parameters in subsequent experiments.

In practice, one can hardly find identical experiments if we strictly consider the equivalence of the subjects, stimuli, and other factors such as instructions in the experiments. Therefore, I use cumulative data from different experimental conditions with broader assumptions.

Sahakyan and Delaney (2005) studied how directed forgetting affects subsequent learning. In a typical directed forgetting study, participants are presented with two word lists to study. Between administration of the two lists, the experimenter instructs half of the participants to forget the first list and the remaining half of the participants to keep remembering the words. After studying the second list, participants are asked to recall all the items, including any items they were earlier instructed to forget (if applicable). Table 12 shows the data collected in the experiment, and Table 13 shows the MLE and Bayesian estimates for these data sets.

Table 12
Group 3×3 Data Tables

	Short lists (16 - 22)			Long lists (30 - 36)		
Group	List 1	List 2	New	List 1	List 2	New
Forget						
List 1	151	95	58	178	197	153
List 2	19	224	61	54	379	95
New	28	35	545	38	66	952
Remember						
List 1	170	60	74	240	154	134
List 2	36	197	71	99	254	175
New	46	29	533	106	104	846

Note. Forget group is instructed to forget list 1 after learning. Remember group is instructed to remember list 1 after learning.

Table 13
Parameter Estimates for The Data in Table 12

	EM/(Bayesian) estimation					
Condition	D_1	D_2	d_1	d_2	a=g	b
Forget & Short	.79	.78	.31	.85	.44	.10
	(.78)	(.77)	(.30)	(.84)	(.45)	(.10)
Remember & Short	.72	.73	.34	.78	.61	.12
	(.72)	(.73)	(.33)	(.77)	(.61)	(.13)
Forget & Long	.68	.80	.18	.67	.37	.10
	(.68)	(.80)	(.18)	(.66)	(.36)	(.10)
Remember & Long	.68	.59	.23	.51	.50	.20
	(.68)	(.58)	(.23)	(.50)	(.50)	(.20)

Note. D_1 = detectability of list 1 items; D_2 = detectability of list 2 items; d_1 = source discriminability for list 1 items; d_2 = source discriminability for list 2 items; b = bias for responding "old"; g = guessing that the item was a list 1 item.

The estimates of MLE and Bayesian approaches in Table 13 are quite similar in four conditions. Now, the pooled data (i.e., the forget and remember conditions are combined) is used for the MLE estimation and the cumulative data (i.e., the forget condition followed by the remember condition) for the Bayesian estimation. The pooled data set is shown in Table 14.

Table 14
Combined Forget Condition and Remember Condition Data from Table 12

Group	List 1	List 2	New
Short			
List 1	321	155	132
List 2	55	421	132
New	74	64	1078
Long			
List 1	418	351	287
List 2	153	633	270
New	144	170	1798

Note. Short lists contain 16 - 22 words. Long lists contain 30 - 36 words.

Given that the Bayesian inference uses the posterior of the previous inference as the prior of the latter inference, it is necessary to find the explicit mathematical form of the posterior of the previous inference. The MCMC approximation gives the approximated mean and standard deviation of the beta posterior. According to the properties of beta distributions, we have:

$$m = E[\theta] = \frac{\alpha}{\alpha + \beta}, \tag{45}$$

and

$$V = S^2[\theta] = Var[\theta] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \tag{46}$$

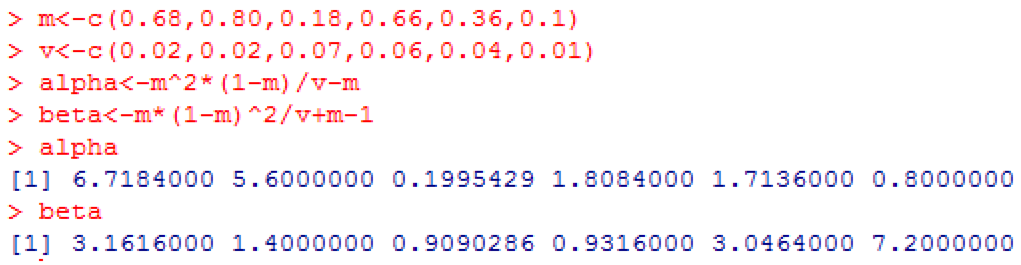
Hence we have

$$\alpha = \frac{m^2(1-m)}{V} - m, \tag{47}$$

and

$$\beta = \frac{m(1-m)^2}{V} + m - 1. \tag{48}$$

Therefore, the beta posterior can be fully specified from equation (45) and equation (46). As an example, the computation of α and β of the posteriors for the “remember-short” condition is shown in Figure 6.



```
> m<-c(0.68,0.80,0.18,0.66,0.36,0.1)
> v<-c(0.02,0.02,0.07,0.06,0.04,0.01)
> alpha<-m^2*(1-m)/v-m
> beta<-m*(1-m)^2/v+m-1
> alpha
[1] 6.7184000 5.6000000 0.1995429 1.8084000 1.7136000 0.8000000
> beta
[1] 3.1616000 1.4000000 0.9090286 0.9316000 3.0464000 7.2000000
```

Figure 6
Computation of α And β of The Posterior for The“Forget-long” Condition. This Posterior Acts as The Prior of The “Remember-long” Condition.

Table 15

Parameter Estimates for The Data in Table 14

	EM/(Bayesian) estimation					
Condition	D_1	D_2	d_1	d_2	$a = g$	b
Short	.76 (.76)	.76 (.75)	.31 (.38)	.81 (.78)	.54 (.56)	.11 (.11)
Long	.68 (.68)	.70 (.59)	.17 (.18)	.61 (.52)	.46 (.52)	.15 (.20)

Note. D_1 = detectability of list 1 items; D_2 = detectability of list 2 items; d_1 = source discriminability for list 1 items; d_2 = source discriminability for list 2 items; b = bias for responding "old"; g = guessing that the item was a list 1 item.

The MLE and Bayesian estimates for the data sets that combine (pooled or cumulative) forget and remember conditions are shown in Table 15. It should be noted that most of the estimates are still quite close, and the Bayesian estimates tend to be more centralized. This result shows that Bayesian and MLE estimations are not significantly different when they are based on the same information.

However, it is possible that the informative prior misleads the estimation when it is specified incorrectly. Sahakyan and Delaney (2005) find that learning short and long word lists may yield different discrimination rates (d_1). Therefore, it is inappropriate to use the posterior of d_1 in the short word list as the prior of the long word list. To test the effect of the inappropriate prior, the pooled data (i.e., the short and long lists are combined) was used for the MLE estimation and the cumulative data (i.e., the short list followed by the long list) for the Bayesian estimation. The pooled data set is shown in Table 16. The estimates are shown in Table 17. As discussed in the previous section, the Bayesian estimates are usually more centralized than MLE estimates due to different estimators they recruit. In this example, however, the Bayesian estimates of d_1 tend to be closer to 0. Apparently, the cumulative estimate should fall into the range of the estimates of the two separate data sets. However, the Bayesian estimates in Table 17 are not greater than the smaller estimate of the separate estimates. This fact indicates that the inference based on the informative prior may not be appropriate.

Table 16
Combined Short List and Long List Data from Table 12

Group	List 1	List 2	New
Forget			
List 1	329	73	66
List 2	292	603	101
New	211	156	1497
Remember			
List 1	410	135	152
List 2	214	451	133
New	208	246	1379

Note. Forget group is instructed to forget list 1 after learning. Remember group is instructed to remember list 1 after learning.

2.6.4 Comparison of Inferences

In the previous section, the comparison of the EM and Bayesian estimates is presented in terms of equating parameters and testing the model fit. Additionally, the Bayesian approach has an alternative feature for testing hypotheses.

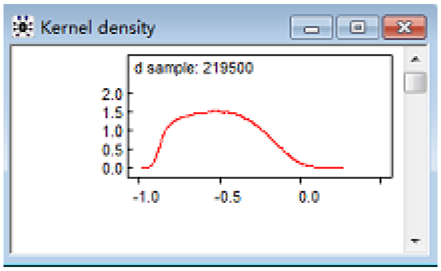
Table 17
Parameter Estimates for The Data in Table 16

	EM/(Bayesian) estimation					
Condition	D_1	D_2	d_1	d_2	$a = g$	b
Forget	.72	.79	.23	.75	.40	.10
	(.72)	(.79)	(.18)	(.74)	(.39)	(.10)
Remember	.70	.64	.28	.62	.53	.17
	(.68)	(.59)	(.18)	(.53)	(.52)	(.20)

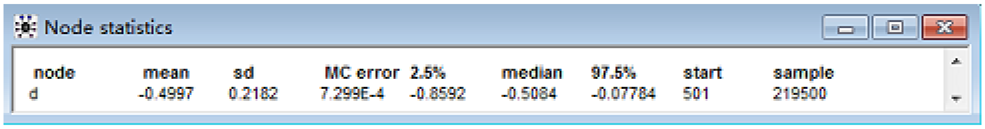
Note. D_1 = detectability of list 1 items; D_2 = detectability of list 2 items; d_1 = source discriminability for list 1 items; d_2 = source discriminability for list 2 items; b = bias for responding “old”; g = guessing that the item was a list 1 item.

In Bayesian hypotheses testing, one can set an additional parameter, which is the difference between the two parameters that we intend to test. For example, in the 6C model case, one can set a new parameter as $d = d_1 - d_2$ to test the difference between d_1 and d_2, instead of set $d = d_1 = d_2$ to test 5C sub model. From Figure 7(a) and 7(b), it is evident that the difference between d_1 and d_2 satisfies the 95% Bayesian confidence interval (two-tailed). Hence, the 5C sub model that equates d_1 and d_2 will not fit the data as well.

This method of testing sub model hypotheses provides an approach not only to testing model fit, but also to distinguishing the difference between two parameters. Hence, this method is better than classic hypothesis testing, because in classic model hypothesis testing, the parameters are equated and may lose potential information by merging parameters. Depicting the distribution of the parameter difference provides a full description of the relation between parameters.



(a) Posterior of $d = d_1 - d_2$



node	mean	sd	MC error	2.5%	median	97.5%	start	sample
d	-0.4997	0.2182	7.299E-4	-0.8592	-0.5084	-0.07784	501	219500

(b) Statistics of $d = d_1 - d_2$

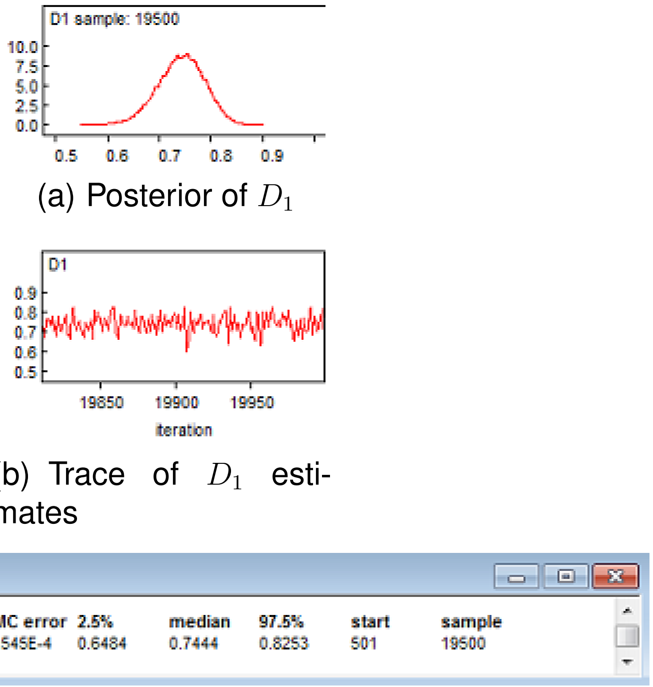
Figure 7
Bayesian Estimates of The Parameters

2.7 Discussion

In the previous section, I have compared the EM and the Bayesian approaches by estimating MPT models in source monitoring experiments as examples.

As two different theoretical contexts, the most important differences between classic frequentism and Bayesianism are (1) whether or not the “subjective” prior information should be involved in the data analysis, and (2) the logic of reasoning, namely whether we assume that we have given (or fixed) parameters which are to be estimated using the data in the experiments (or the data), or the parameters follow distributions that need to be continuously updated via the new data (Carlin and Louis 2009). This basic difference leads to other distinctions. For instance, all frequentist assumptions and tests are based on the precondition that the observations are independently and identically distributed (i.i.d) following a certain distribution (with fixed parameters). However, the Bayesian approach does not rely on this condition. Instead, the Bayesian approach believes that every subject has his or her own item response distributions in terms of different items. This approach allows us to have various beliefs (priors) from person to person. Furthermore, the Bayesian approach believes that not only can individuals have different priors, but that the population is dynamic due to the variability of its members, and it varies as a certain distribution with higher dimensional parameters. Moreover, rather than an absolute conclusion given in frequentism (accept H_0 or reject) with an α, the Bayesian approach offers us the more reasonable conclusion that we have some probability to accept or reject a hypothesis. Last but not least, the logic of the Bayesian inference is to maximize $P(\theta|D)$ while MLE is to maximize $P(D|\theta)$.

Resulting from the theoretical differences of classic statistical methods and the Bayesian inference, the two estimations have essential differences that (1) the Bayesian estimation is cumulative, which means the beta priors I use for further similar analyses may be the posteriors I obtained from current estimation, rather than the non-informative prior. Therefore, the estimates of the Bayesian approach change more dramatically than the estimates of the EM approach, and (2) the Bayesian estimates are actually distributions instead of point estimates in the EM. Bayesian estimates may also change when different estimators are chosen. Figure 8 shows the posterior distribution of parameter D_1 (Figure 8(a)) and its trace of the estimates (Figure 8(b)), as well as other statistical descriptions of the posterior of D_1 (Figure 8(c)). Furthermore, the Bayesian approach can detect



(a) Posterior of D_1

(b) Trace of D_1 estimates

```
node    mean    sd       MC error  2.5%    median  97.5%   start  sample
D1      0.7425  0.04542  3.545E-4  0.6484  0.7444  0.8253  501    19500
```

(c) Statistics of D_1

Figure 8
Detailed Bayesian Estimates of The Parameters

individual differences with respect to subjects and source items, which is neglected in classic frequentism due to its basic preassumption of i.i.d. This analysis cannot be performed at present due to the fact that only group data were available in the original articles. However, this issue can be addressed by conducting Monte Carlo simulations of the models. Meanwhile, this issue indicates the importance of keeping original data and response items.

Although these two approaches are contradictory in terms of theoretical basis, there are some remarkable similarities with respect to their reasoning and scenarios. First, the information used in two estimations is all from the likelihood function (because I used a non-informative prior). Therefore, the point estimates for most parameters are very close, even though some are not (probably because the EM is trying to find the mode of the likelihood while the estimator recruited in the Bayesian approach is the posterior mean). In addition, as a typical method used in classic statistics coping with incomplete data, the iterative process of the EM algorithm can be considered as a Markov process as well, because it satisfies the definition of a Markov chain that the status next step is only determined by its previous step. This fact indicates that when the estimate becomes convergent, it will be completely independent from the parameters' initial values. This scenario is quite similar with that used in the Metropolis algorithm when approximating the Bayesian posterior. Lastly, the EM algorithm is also applied to the Bayesian inference to find the maximum posteriori (MAP) estimate, which is the mode of the posterior distribution (Carlin and Louis 2009).

References

Akaike, H. 1973. Information theory and the an extension of the maximum likelihood principle. In B. Petrov and T. Meiser, editors, *Second International Symposium on Information Theory*, pages 267–281, Academiai Kiado.

Batchelder, W. H., and D. M. Riefer. 1986. The statistical analysis of a model for storage and retrieval processes in human memory. *British Journal of Mathematical and Statistical Psychology*, 39:129–149.

Batchelder, W. H., and D. M. Riefer. 1990. Multinomial processing models of source monitoring. *Psychological Review*, 97(4):548–564.

Batchelder, W. H., and D. M. Riefer. 1999. Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin and Review*, 6:57–86.

Carlin, B. P., and T. A. Louis. 2009. *Bayesian Methods for Data Analysis*. CRC Press, Boca Raton, FL, 3rd edition.

Dempster, A. P., N. M. Laird, and D. B Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B*, 34:1–38.

Erdfelder, E., T. S. Auer, B. E. Hilbig, A. Abfalg, M. Moshagen, and L. Nadarevic. 2009. Multinomial processing tree models: A review of the literature. *Zeitschrift fur Psychology / Journal of Psychology*, 217:108–124.

Harvey, P. D. 1985. Reality monitoring in mania and schizophrenia. *The Journal of Nervous and Mental Disease*, 173:67–72.

Hoff, P. D. 2009. *A First Course in Bayesian Statistical Methods*. Springer, New York, NY.

Hu, X., and W. H. Batchelder. 1994. The statistical analysis of general processing tree models with the EM algorithm. *Psychometrika*, 59(1):21–47.

Hu, X., and G. A. Philips. 1999. GPT.EXE: A powerful tool for the visualization and analysis of general processing tree models. *Behavior research methods, instruments, and computers*, 31(2):220–234.

Johnson, M. K., M. A. Foley, and K. Leach. 1988. The consequences for memory of imagining in another person's voice. *Memory and Cognition*, 16(4):337–342.

Johnson, M. K., S. Hashtroudi, and D. S. Lindsay. 1993. Source monitoring. *Psychological Bulletin*, 144(1):3–28.

Johnson, M. K., and C. L. Raye. 1981. Reality monitoring. *Psychological Review*, 88:67–85.

Karabatsos, G. 2006. Bayesian nonparametric model selection and model testing. *Journal of Mathematical Psychology*, 50(2):123–148.

Moshagen, M. 2010. multiTree: A computer program for the analysis of multinomial processing tree models. *Behavior research methods*, 42(1):42–54.

Rose, R. G, N. King, and A. Perez. 1975. Bilingual memory for related and unrelated sentences. *Journal of Experimental Psychology: Human learning and Memory*, 1:599–606.

Rothkegel, R. 1999. AppleTree: A multinomial processing tree modeling program for macintosh computers. *Behavior research methods, instruments, and computers*, 31(4):696–700.

Saegert, J., E. Hamayan, and H. Ahmar. 1975. Memory for language of input in polyglots. *Journal of Experimental Psychology: Human Learning and Memory*, 5:607–613.

Sahakyan, L., and P. F. Delaney. 2005. Directed forgetting in incidental learning and recognition testing: Support for a two-factor account. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 31(4):789–801.

Stahl, C., and K. C. Klauer. 2007. HMMTree: A computer program for latent-class hierarchical multinomial processing tree models. *Behavior research methods*, 39(2):267–273.

Stahl, C., and T. Meiser. 2009. New directions in multinomial modeling. *Zeitschrift fur Psychology / Journal of Psychology*, 217(3):105–107.