2005-11-08

Modeling Distributions of Test Scores with Mixtures of Beta Distributions

Jingyu Feng
Brigham Young University - Provo

MODELING DISTRIBUTIONS OF TEST SCORES WITH MIXTURES OF BETA DISTRIBUTIONS

by

Jingyu Feng

A Project submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Statistics

Brigham Young University

August 2005

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a Project submitted by

Jingyu Feng

This Project has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date	G. Bruce Schaalje, Chair
Date	C. Shane Reese
Date	David Whiting

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the Project of Jingyu Feng in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

G. Bruce Schaalje
Chair, Graduate Committee

Accepted for the Department

G. Bruce Schaalje
Graduate Coordinator

Accepted for the College

G. Rex Bryce
Associate Dean, College of Physical and Mathematical Sciences

ABSTRACT

MODELING DISTRIBUTIONS OF TEST SCORES WITH MIXTURES OF BETA DISTRIBUTIONS

Jingyu Feng

Department of Statistics

Master of Science

Test score distributions are used to make important instructional decisions about students. The test scores usually do not follow a normal distribution. In some cases, the scores appear to follow a bimodal distribution that can be modeled with a mixture of beta distributions. This bimodality may be due different levels of students' ability. The purpose of this study was to develop and apply statistical techniques for fitting beta mixtures and detecting bimodality in test score distributions.

Maximum likelihood and Bayesian methods were used to estimate the five parameters of the beta mixture distribution for scores in four quizzes in a cell biology class at Brigham Young University. The mixing proportion was examined to draw conclusions about bimodality. We were successful in fitting the beta mixture to the data, but the methods were only partially successful in detecting bimodality.

Acknowledgements

I would like to thank my advisor, Dr. Schaalje, the members of my graduate committee, and the Department of Statistics for all the continual help and support I received throughout this entire process. Additionally, I want to pay tribute to my family, for all the love, support and patience they have given me these years.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

The interpretation of test scores is a critical component of any educational program. Teachers need to know as much as possible about how their students differ in order to match classroom instruction to the specific needs of each student. School administrators also need to plan for the long-term education of students. By investigating the properties of distributions of test scores, teachers and school administrators can get useful information about the students and teaching strategies.

The mean or the median indicates how well students master the material on average. The variance indicates variability in the levels of mastery of the material. The shape of the distribution also has meaning. For example, if the distribution is left skewed, it may indicate either that students have mastered the material very well or the test is too easy; if the shape is right skewed, it may indicate either that the test is too hard or the way that the teacher delivers the material needs to be improved; if the shape is bimodal, it may indicate that there are two learning styles or backgrounds in the class and the teacher should modify the teaching style to accommodate the different needs.

Researchers can often detect bimodality visually if the peaks are sufficiently

well-defined and far apart. However, bimodality may also be difficult to detect because it may be disguised as a unimodal but complex distribution. Many methods have been developed for detecting bimodality. Most of them can be categorized into two general approaches. The first approach is to calculate an optimized bimodality measure, such as the bimean (Dunn, Janos, and Rosenfeld 1983), Fisher distance (Phillips, Rosenfeld, and Sher 1989) or entropy distance (Gilks, Richardson, and Spiegelhalter 1996), then decide if the measure is extreme enough to suggest bimodality. This approach is usually highly computational and the critical value depends on the nature of the problem (Gilks, Richardson, and Spiegelhalter 1996). The second approach is to fit a parametric mixture model with a mixing parameter π, then draw the inferences about π.

In this paper, we use the second approach to test bimodality. We implement this approach using a likelihood method based on the bootstrap (Davison and Hinkley 1997) and a Bayesian hierarchical method based on Markov Chain Monte Carlo. We also discuss interesting features of both implementations.

We apply these methods to test scores from four quizzes in a cell biology class at BYU (Figure 1.1). The scores for the four quizzes were whole numbers in $[0, 15]$. Because of the properties of the scores, we propose models involving mixtures of two beta distributions to model the test score distributions.

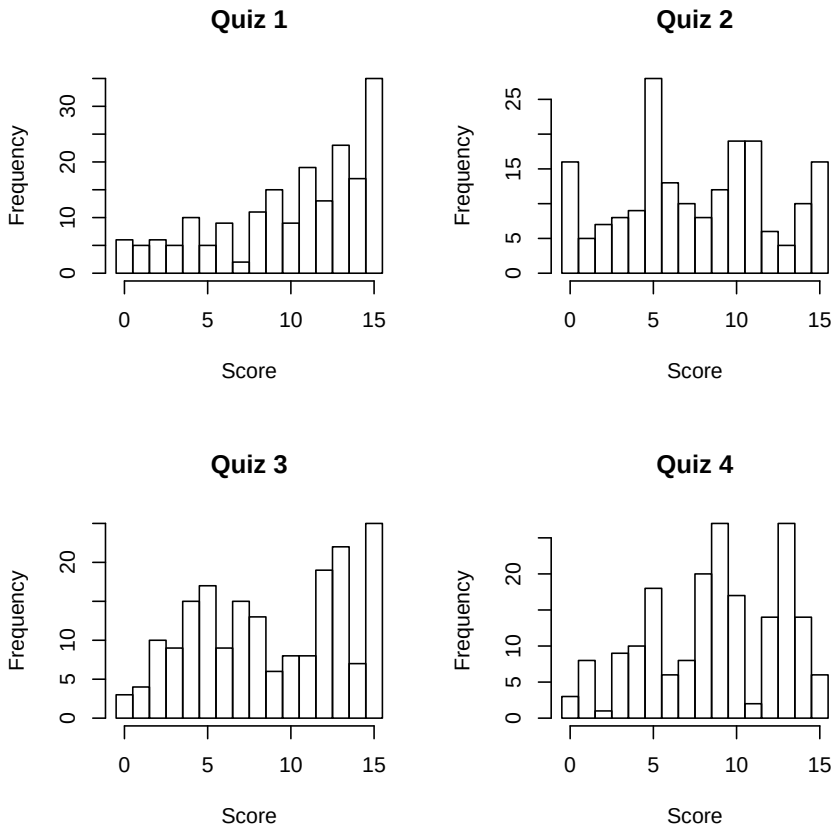

Figure 1.1: Histograms of scores on 4 quizzes in a cell biology class at BYU.

Chapter 2

Literature Review

2.1 Bimodality Measures

For bimodality analysis based on measures of bimodality, there are several choices, such as the bimean, Fisher distance and entropy distance. However, each of them has some undesirable properties.

Methods based on the bimean and the Fisher distance proceed by dividing the range of the data into two subsets (Phillips, Rosenfeld, and Sher 1989). The bimean approach divides the range so as to minimize $\hat{\sigma}_1^2 + \hat{\sigma}_2^2$ where $\hat{\sigma}_1^2$ is the sample variance of the observations in the lower part of the range and $\hat{\sigma}_2^2$ is the sample variance of observations in the upper part of the range. It does not involve either the means or the population sizes. On the other hand, the Fisher distance (FD^2) method divides a population into two subsets so as to maximize

$$FD^2 = n(\hat{\mu}_1 - \hat{\mu}_2)^2/(n_1\hat{\sigma}_1^2 + n_2\hat{\sigma}_2^2)$$

where $\hat{\sigma}_1^2$ and $\hat{\sigma}_2^2$ are as above, $\hat{\mu}_1$ and $\hat{\mu}_2$ are the corresponding sample means, n_1 and n_2 are the corresponding sample sizes, and $n = n_1 + n_2$. The Fisher distance approach is more powerful than the bimean apporach. The literature, does not discuss a sampling

model or a statistical "test" for bimodality based on these measures.

The entropy distance (ED) approach to bimodality analysis is to compare a two-component mixture to "the closest" single-component model where "the closest" means the model that minimizes

$$ED[g,h] = \int log(g(x)/h(x))g(x)dx$$

where $h(x)$ is the single-component model and $g(x)$ is the two-component model. If the entropy distance is large enough, it indicates the presence of bimodality. This approach is difficult to discuss in general because the critical value is dependent on the nature of the problem.

2.2 Mixture Models and Methods

Mixture distributions have been used as models throughout the history of modern statistics. One of the first major analyses involving the use of mixture models was undertaken over 100 years ago by Karl Pearson (Titterington, Smith, and Makov 1985). He used a mixture of two normal probability density functions as a model to describe a set of measurements on the ratio of forehead to body length for 1000 crabs.

A mixture of two distributions will not always have obvious bimodality unless the means are sufficiently different. However, mixtures are often used to investigate bimodality (Reschenhofer 2001).

The formula for a finite mixture density function is

$$p(x) = \pi_1 f_1(x) + \cdots + \pi_k f_k(x)$$

$$\text{where} \quad 1 > \pi_j > 0, \quad j = 1, \cdots, k, \qquad \pi_1 + \cdots + \pi_k = 1$$

$$\text{and} \qquad f_j(.) \geq 0, \qquad \int_x f_j(x)dx = 1, \quad j = 1, \cdots, k,.$$

Because of the flexibility of mixture models, mixtures of many kinds of distributions have been studied, such as Normals (Hosmer 1973), Binomials (Gelfand and Solomon 1975), Compound Poissons (Paull 1978) and Geometrics (Harris 1983).

In line with the popularity of mixture models, a variety of methods have been developed to estimate the parameters of a mixture model, such as the method of moments, maximum likelihood, Bayesian methods, minimum distance estimation and numeric decomposition of mixtures.

The most fundamental 'parameter' in the definition of a mixture model is the number of components. In applications involving one or possibly two components, we can approach the question of the number of component in hypothesis testing terms by testing.

$$\text{H}_0\text{: single component}$$

$$\text{vs} \quad \text{H}_a\text{: two components}$$

Because H_0 can be regarded as a special case of H_a, it is natural to consider applying the generalized likelihood ratio test (Titterington, Smith, and Makov 1985). However, since there is not a unique way of obtaining H_0 from H_a, it is very hard to decide the appropriate degrees of freedom for the chi-squared distribution (Titterington, Smith, and Makov 1985).

Aitkin, Anderson, and Hinde (1981) proposed a mixture to model an extensive body of educational research data on teaching styles and pupil performance. In the ab-

sence of a general procedure to test the hypothesis, they essentially used a bootstrap approach. Aitkin and Rubin (1985) adopted a Bayesian approach which places a prior distribution on the vector of mixing proportions.

2.3 The Beta Distribution and Mixture of Betas

The beta distribution is related to the gamma distribution. It is a continuous probability distribution with the probability density function (pdf) defined on the interval (0, 1). The pdf of the beta distribution is

$$f(x) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1}$$

where a, b>0 and 0<x<1.

The density function for a mixture of two beta distributions is simply the weighted sum of two beta density functions,

$$f(x) = \pi \frac{\Gamma(a_1+b_1)}{\Gamma(a_1)\Gamma(b_1)} x^{a_1-1}(1-x)^{b_1-1} + (1-\pi)\frac{\Gamma(a_2+b_2)}{\Gamma(a_2)\Gamma(b_2)} x^{a_2-1}(1-x)^{b_2-1}$$

where $0< \pi <1$. The mixture of two beta distributions has rarely been used. A literature search produced only 2 uses of beta mixtures. A two-component beta mixture was used to model gene expression measures and detect periodically expressed genes (Lu, Zhang, and Qin 2004). The maximum likelihood method was used to estimate the five unknown parameters. Also, in the problem of testing whether or not a given parametric model is compatible with the data at hand, researchers used a Bayesian approach to estimate the number of components of a potentially infinite mixture of beta distributions

(Robert and Rousseau 2002).

2.4 The Bootstrap

A problem with inferences about the mixing parameter in a beta mixture is that the likelihood function may not be regular (Pawitan 2001). Such complications suggest investigating nonparametric approaches to inference. Bootstrapping (Efron 1979) is a computer-simulated, nonparametric technique for making inferences about a population parameter based on sample statistics. The basic idea behind bootstrapping is that if the sample is a good approximation of the population, the sampling distribution of interest may be estimated by generating a large number of new samples (called resamples) from the original sample. Bootstrapping is useful when inference is to be based on a complex procedure for which theoretical results are unavailable or not useful for the sample sizes met in practice (Davison and Hinkley 1997). Bootstrapping allows researchers to use an estimator even if the theorical sampling distribution is not available.

The general bootstrap algorithm is:

1. Generate a sample x^* of size n by sampling with replacement from the sample $\{x_1, x_2, \cdots, x_n\}$.

2. Compute the statistic of interest $\hat{\theta}^*$ for this bootstrap sample.

3. Repeat steps 1 and 2 m times, where m is large.

Using this procedure we end up with bootstrap values $\hat{\theta}^* = \{\hat{\theta}_1^*, \hat{\theta}_2^*, \cdots, \hat{\theta}_m^*\}$. These bootstrap values are used for calculating standard errors, test statistics and other quan-

tities of interest in inference. Note that $\hat{\theta}^*$ is treated as a sample from the unknown distribution of $\hat{\theta}$.

2.5 Markov Chain Monte Carlo Computation

Markov Chain Monte Carlo (MCMC) is a numerical integration technique based on simulation. It is primarily used in connection with the Bayesian approach to data analysis for approximating posterior distributions of parameters. MCMC was developed in physics by Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller (1953), and later generalized and put into a statistical framework by Hastings (1970). The principle is to observe a succession of states; once convergence is reached, the consecutive states are assumed to be drawn from the target probability distribution. Using MCMC, it is possible to sample from general joint probability distributions which have been specified using conditional distributions. The conditional specifications allow dependent samples from the joint distribution to be drawn. If the samples constitute an irreducible ergodic Markov chain with stationary distribution, the ergodic theorem implies that the samples can be used to characterize the joint distribution (Gilks, Richardson, and Spiegelhalter 1996).

It is natural to implement MCMC to estimate the parameters of a mixture model because in some situations the maximum likelihood estimators for parameters of a mixture model do not exist whatever the sample size (Lehmann 1983).

The general Metropolis-Hastings algorithm for drawing a sequence $x_i, i = 0, ..., m$

of dependent samples with unnormalized pdf $g(.)$ is:

1. Choose a starting value x_0.

2. Set i=1.

3. Generate a candidate value y from a pdf $p(y|x_i)$.

4. If $u \sim \mu(0,1) < \frac{g(y)p(y|x_i)}{g(x_i)p(x_i|y)}$), then $x_i = y$ else $x_i = x_{i-1}$.

5. Repeat steps 3 and 4 for $i = 2, \cdots, m$.

2.6 Nelder-Mead Simplex (NMS)

The NMS algorithm (Nelder and Mead 1965) has become a widely used method for nonlinear unconstrained optimization because it does not require the objective function $g(x)$ to be differentiable. Compared with other optimization algorithms, such as Newton-Raphson Optimization With Line-Search, Quasi-Newton Optimization, Conjugate Gradient Optimization etc., the NMS method does not use any derivatives. The NMS method is an iterative method that attempts to minimize a scalar-valued nonlinear function of n real variables using only function values.

Four scalar parameters must be specified to define a complete Nelder-Mead method: coefficients of reflection (ρ), expansion (χ), contraction (γ), and shrinkage (σ). The nearly universal choices used in the standard Nelder-Mead algorithm are

$$\rho = 1, \qquad \chi = 2, \qquad \gamma = \tfrac{1}{2}\,, \quad \text{and} \quad \sigma = \tfrac{1}{2}\,.$$

The result of each iteration is either (1) a single new vertex — the accepted point — which replaces x_{n+1} in the set of vertices for the next iteration, or (2) a set of n new

points that, together with x_1, form the simplex at the next iteration.

The likelihood function of a mixture model usually is quite complex and difficult to differentiate. This makes NMS method a natural choice to maximize the likelihood function for a mixture model.

Chapter 3

Methodology

Our objective was to identify whether there were two sub-populations in the cell biology class by detecting bimodality of the distribution of the students' quiz scores. We used two methods to approach this problem.

The first method was an empirical approach. After investigating the distribution of scores, we proposed a mixture of discretized beta distributions. This model,

$$y \sim \pi Beta(\alpha_1, \beta_1) + (1 - \pi)Beta(\alpha_2, \beta_2),$$

has five parameters, $\alpha_1, \alpha_2, \beta_1, \beta_2$ and π. The likelihood is not regular, so inferences could not be based on Fisher Information. To get the maximum likelihood estimates of the parameters we used the NMS algorithms available in the NLP procedure of SAS (SAS 2004). We used the bootstrap procedure with 1000 resamples to compute bootstrap estimates of sampling distributions as well as simple percentile confidence limits for all five parameters. We arbitrarily considered the distribution to be unimodal if the confidence interval for the mixing parameter included 0.025 or 0.975 . Also, with the scores divided into sixteen bins, Chi-Square goodness-of-fit tests with eleven degrees of freedom were used to assess the fit of the mixture model. The SAS code for proc NLP

and the bootstrap resampling are in Appendix A. The R code for generating histograms with superimposed fitted density function is in Appendix B.

The second method used a more theoretical approach, a Bayesian hierarchical model. Inferences were based on the posterior distribution of the mixing parameter. We first assumed that students' ability levels followed a mixture of beta distributions. For each student, the difficulty levels of the questions were then assumed to be draws from some distribution with mean equal to the students ability. The score for each question was then assumed to be a Bernoulli random variable with probability equal to the question difficulty. Using the result on page 103 of McCullagh and Nelder (1989), the complete quiz score then follows the binomial distribution. In symbols,

$$y_i \sim Beta\ mixture(\pi, \alpha_1, \beta_1, \alpha_2, \beta_2)$$

where y_i is the difficulty for student i, and

$$t_i \sim Binom(15, y_i)$$

where t_i is the total score for student i.

The prior distribution for the mixing parameter π was noninformative. For the other four parameters, we assumed there were two groups of students according to their ability levels. The ability level is in $[0, 1]$. We assumed that one group had an average ability level of 0.33 and the other group had an average ability level of 0.66. The standard deviation for the average ability level in both cases was assumed to be 0.13. Thus, we specified there was a 95% chance that the average ability level of low ability group was in $[0.07, 0.59]$, and a 95% chance the average ability level of high ability

group was in $[0.4, 0.92]$. Therefore, the priors that we chose for α_1, β_1, α_2 and β_2 were $Gamma(4, 1)$, $Gamma(8, 1)$, $Gamma(8, 1)$ and $Gamma(4, 1)$ respectively.

Based on the likelihood function for this hierarchical model and prior distributions for the five parameters, we used MCMC based on simple successive substitution to estimate the posterior distributions for the parameters. We used 300,000 iterations as a burn-in. We checked convergence of the chains by examining trace plots. Finally, we used 20,000 iterations to estimate the posterior distributions of the parameters. Also, a goodness-of-fit test for the Bayesian model was done using the predictive distribution of each observation (Robert and Rousseau 2002). The percent of predictive values less than each observation should follow a uniform distribution if the model fits the data well. The R code for MCMC and the goodness-of-fit test are in Appendix C.

Chapter 4

Results

4.1 Analysis of the Quiz 1 Scores Using the Likelihood Approach

The maximum likelihood estimates for the five parameters of the beta-mixture model were: $\hat{\pi} = 0.1604$, $\hat{\alpha}_1 = 0.9451$, $\hat{\beta}_1 = 2.8284$, $\hat{\alpha}_2 = 2.2243$ and $\hat{\beta}_2 = 0.7905$. The empirical beta mixture model visually fits the data well (Figure 4.1). The goodness-of-fit test had a value of 15.6608, which is less than the critical value 19.6751. This indicates that the data are consistent with the beta-mixture model. The bootstrap density for $\hat{\pi}$ is given in Figure 4.2 and the bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ are given in Figure 4.3. The 95% bootstrap interval for π is (0.0206, 0.5342); hence we conclude that quiz 1 scores are unimodal.

4.2 Analysis of the Quiz 1 Scores Using the Bayesian Approach

The posterior means for the five parameters were: $\bar{\pi} = 0.2911$, $\bar{\alpha}_1 = 4.8202$, $\bar{\beta}_1 = 3.4261$, $\bar{\alpha}_2 = 1.3876$ and $\bar{\beta}_2 = 0.5607$. The posterior distribution for π is given in Figure 4.4 and the posterior distributions for α_1, β_1, α_2 and β_2 are given in Figure 4.5. The plot for the goodness-of-fit test is given in Figure 4.6. The plot indicates that the

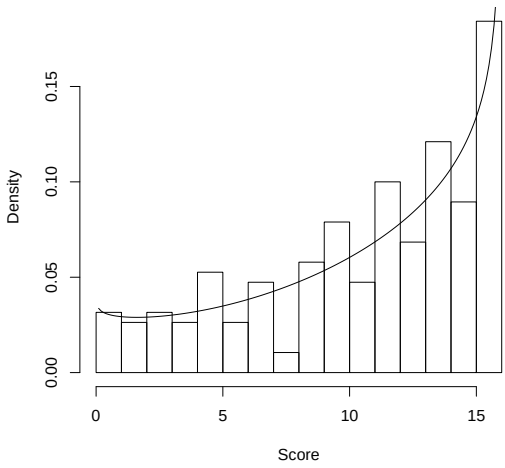

Figure 4.1: Histogram of scores for quiz 1 with fitted beta mixture model.

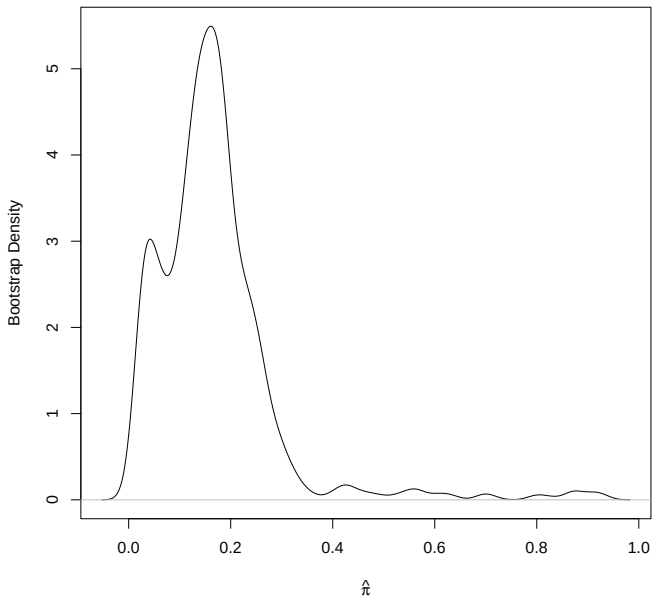

Figure 4.2: Bootstrap density for $\hat{\pi}$ for scores for quiz 1.

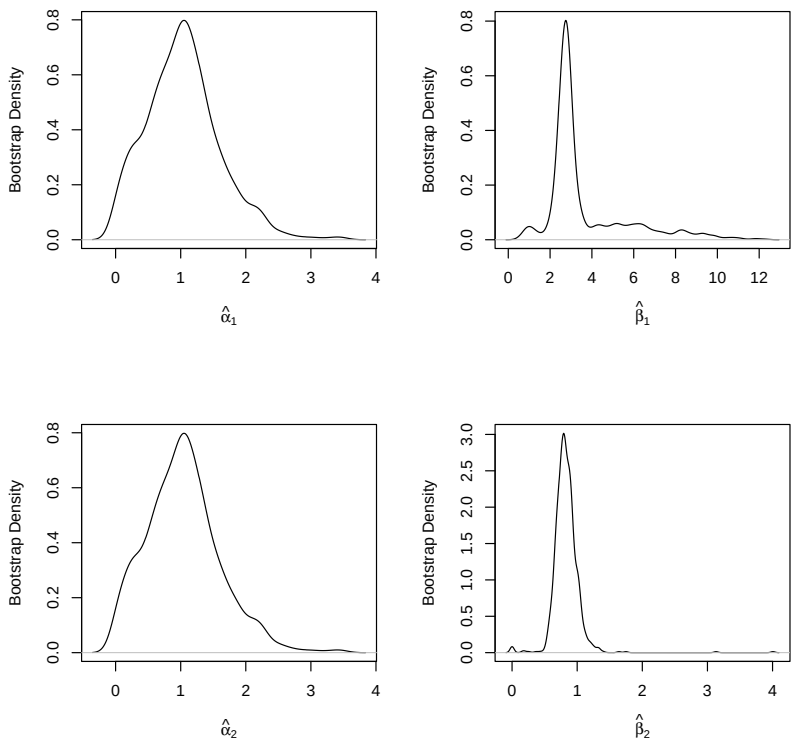

Figure 4.3: Bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ for scores for quiz 1.

Bayesian model fits the data well. We also checked the trace plots to see whether the chains converged. The trace plots show that π, α_1, α_2 and β_2 converged, however the chain for β_1 did not converge. The 95% highest posterior density (hpd) interval for π is (0.0001, 0.7843); hence we conclude that the scores are unimodal.

4.3 Analysis of the Quiz 2 Scores Using the Likelihood Approach

The maximum likelihood estimates for the five parameters of the beta-mixture model were: $\hat{\pi} = 0.95$, $\hat{\alpha}_1 = 1.1221$, $\hat{\beta}_1 = 1.2443$, $\hat{\alpha}_2 = 1.6161$ and $\hat{\beta}_2 = 0.000005$. The empirical beta mixture model does not visually fit the data well (Figure 4.7). The bootstrap density for $\hat{\pi}$ is given in Figure 4.8 and the bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ are given in Figure 4.9. The 95% bootstrap interval for π is (0.0384, 0.9522), which on the surface would suggest bimodality. However, the goodness-of-fit test had a value of 48.317, which is larger than the critical value 19.6751. This indicates that the data are not consistent with the model involving a mixture of two betas.

4.4 Analysis of the Quiz 2 Scores Using the Bayesian Approach

The posterior means for the five parameters are: $\bar{\pi} = 0.6783$, $\bar{\alpha}_1 = 3.3818$, $\bar{\beta}_1 = 3.3976$, $\bar{\alpha}_2 = 0.2713$ and $\bar{\beta}_2 = 0.2431$. The posterior distribution for π is given in Figure 4.10 and the posterior distributions for α_1, β_1, α_2 and β_2 are given in Figure 4.11. The plot for the goodness of fit test is given in Figure 4.12. The plot indicates that the Bayesian model does not fit the data well. We also checked the trace plots to see whether the chains converged. All five chains converged. The 95% hpd interval for π is

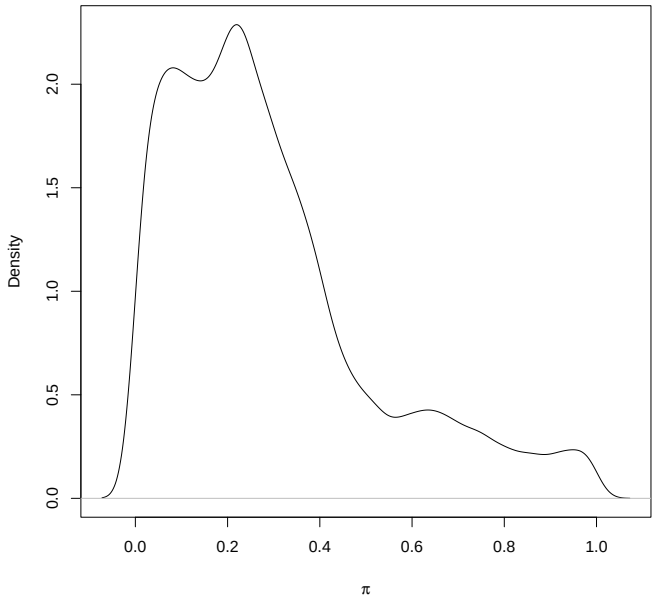

Figure 4.4: The posterior distribution for π for scores for quiz 1.

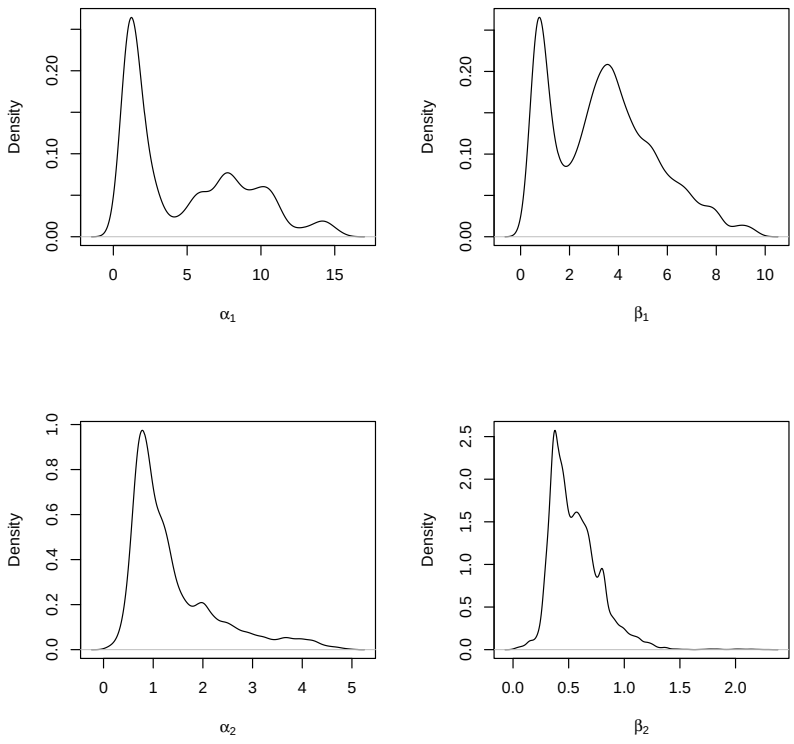

Figure 4.5: The posterior distributions for α_1, β_1, α_2 and β_2 for scores for quiz 1.

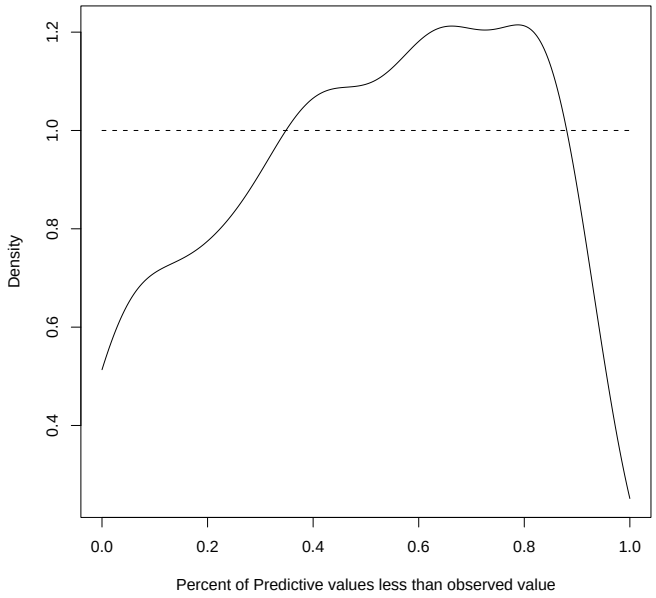

Figure 4.6: The goodness-of-fit test for the Bayesian model for the scores for quiz 1. The model is considered to fit the data well if the kernel density of the posterior predictive quantiles (solid line) approximates the uniform density (dashed line).

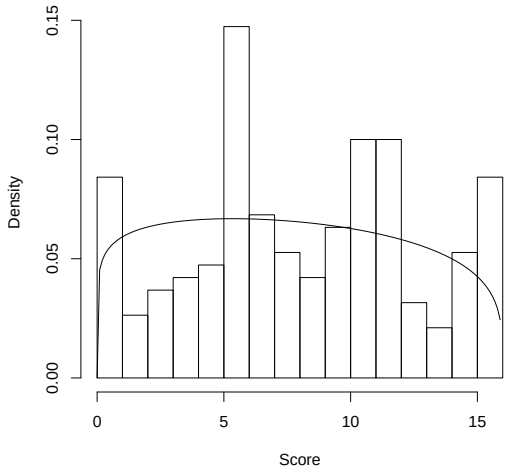

Figure 4.7: Histogram of scores for quiz 2 with fitted beta mixture model.

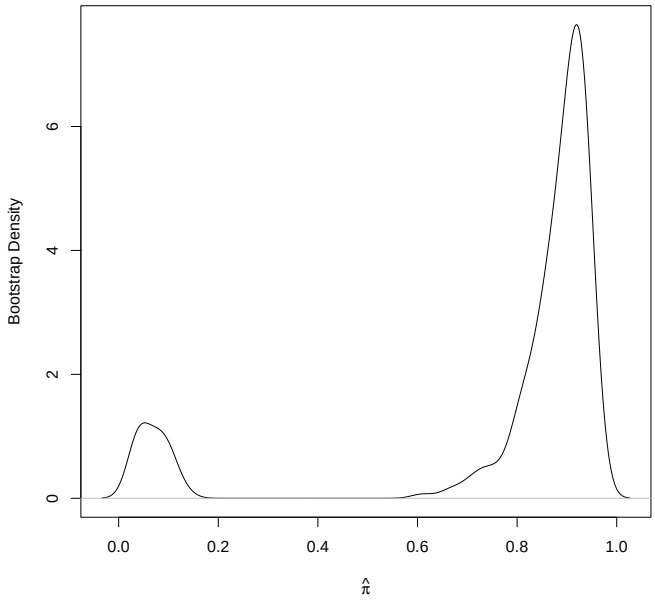

Figure 4.8: Bootstrap density for $\hat{\pi}$ for scores for quiz 2.

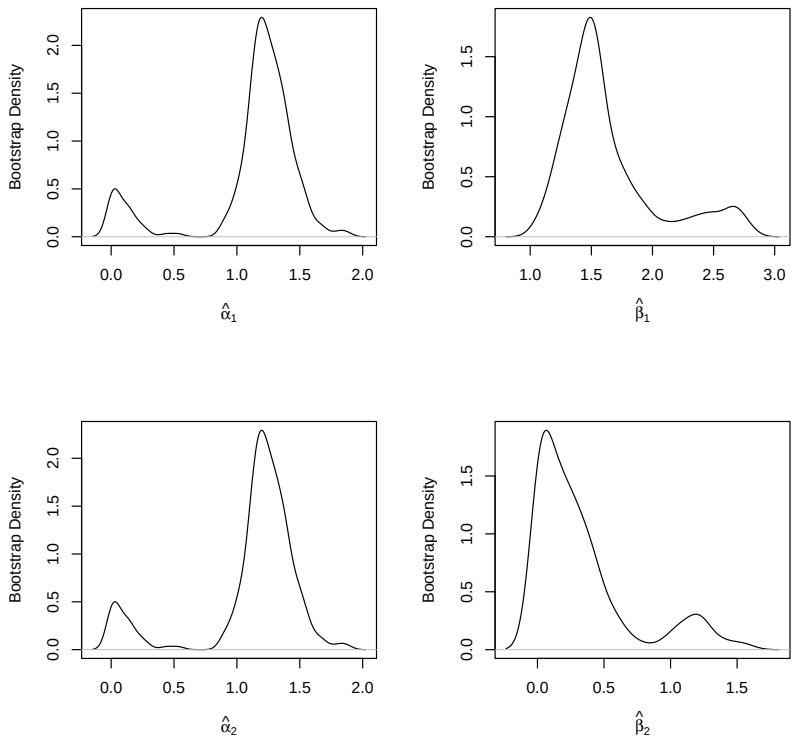

Figure 4.9: Bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ for scores for quiz 2.

(0.5147, 0.8406), which on the surface would suggest bimodality. However, since the data are not consistent with the model involving a mixture of two betas, this prevents us from making any conclusion about bimodality.

4.5 Analysis of the Quiz 3 Scores Using the Likelihood Approach

The maximum likelihood estimates for the five parameters of the beta-mixture model were: $\hat{\pi} = .4058$, $\hat{\alpha}_1 = 2.4325$, $\hat{\beta}_1 = 4.9303$, $\hat{\alpha}_2 = 2.6094$ and $\hat{\beta}_2 = 0.8626$. The empirical beta mixture model visually seems to fit the data well (Figure 4.13). The bootstrap density for $\hat{\pi}$ is given in Figure 4.14 and the bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ are given in Figure 4.15. The 95% bootstrap interval for π is (0.1670, 0.8926), which suggests that the distribution is bimodal. However, the goodness-of-fit test had a value of 23.45504, which is slightly larger than the critical value 19.6751. This indicates that the data are not consistent with the model involving a mixture of two betas.

4.6 Analysis of the Quiz 3 Scores Using the Bayesian Approach

The posterior means for the five parameters are: $\bar{\pi} = 0.544$, $\bar{\alpha}_1 = 3.16$, $\bar{\beta}_1 = 5.488$, $\bar{\alpha}_2 = 8.4856$ and $\bar{\beta}_2 = 1.3595$. The posterior distribution for π is given in Figure 4.16 and the posterior distributions for α_1, β_1, α_2 and β_2 are given in Figure 4.17. The plot for the goodness of fit test is given in Figure 4.18. The plot indicates that the Bayesian model does not fit the data well. We also checked the trace plots to see whether the chains converged. The trace plots shows that chains for π, α_1, β_1 and β_2 converged, however the chain for α_2 did not converge. The 95% hpd interval for π is

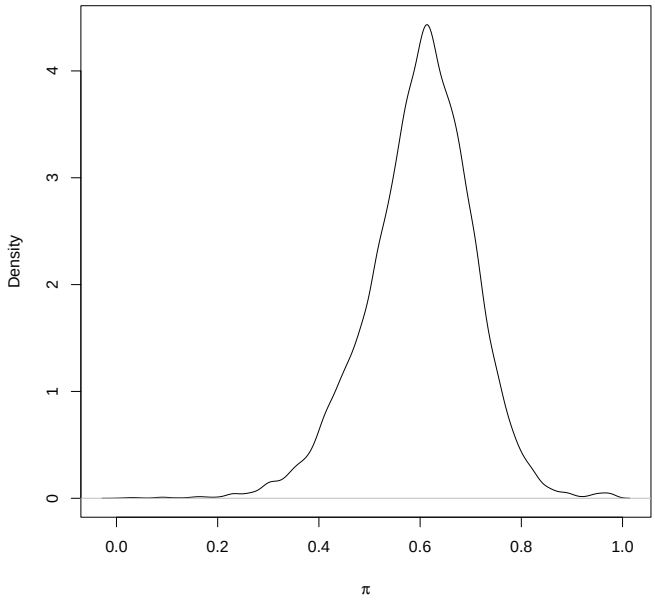

Figure 4.10: The posterior distribution for π for scores for quiz 2.

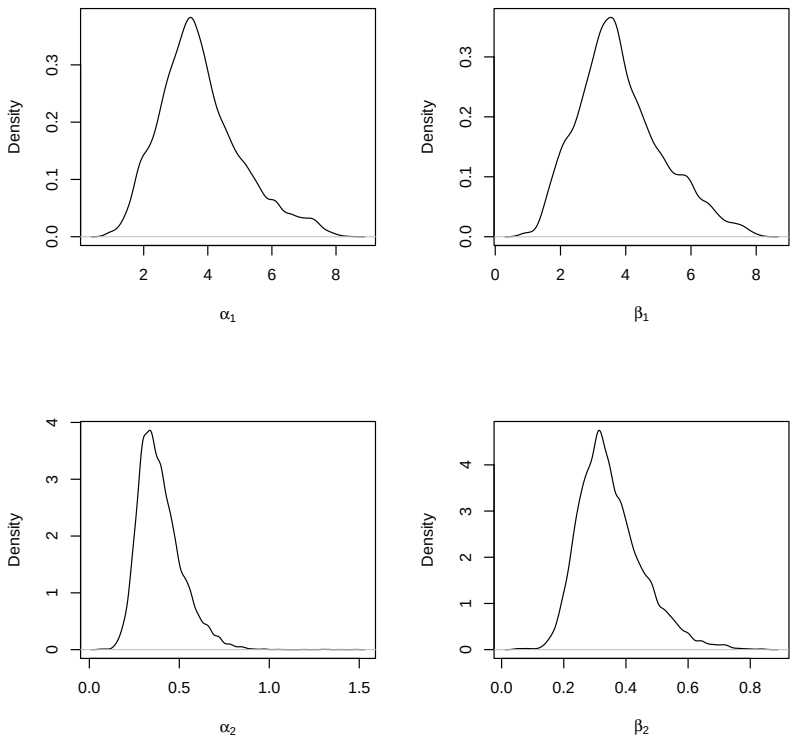

Figure 4.11: The posterior distributions for α_1, β_1, α_2 and β_2 for scores for quiz 2.

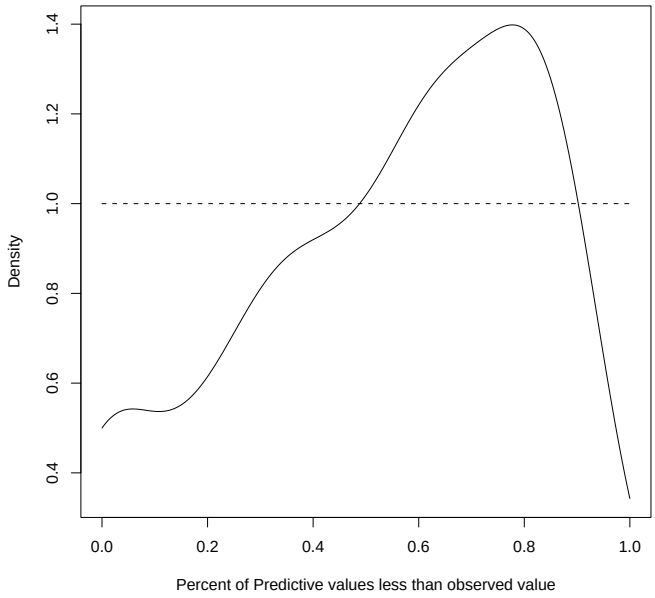

Figure 4.12: The goodness fit test for the Bayesian model for the scores for quiz 2. The model is considered to fit the data well if the kernel density of the posterior predictive quantiles (solid line) approximates the uniform density (dashed line).

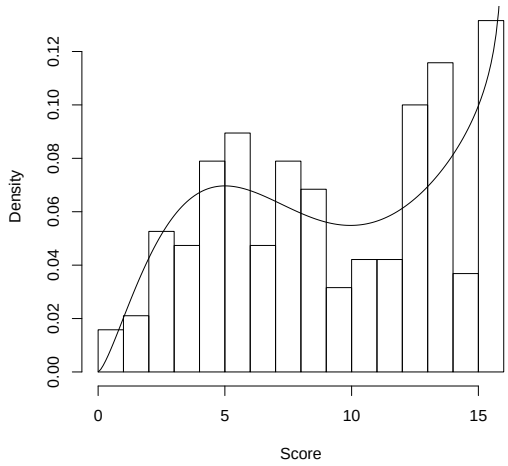

Figure 4.13: Histogram for scores for quiz 3 with fitted beta mixture model.

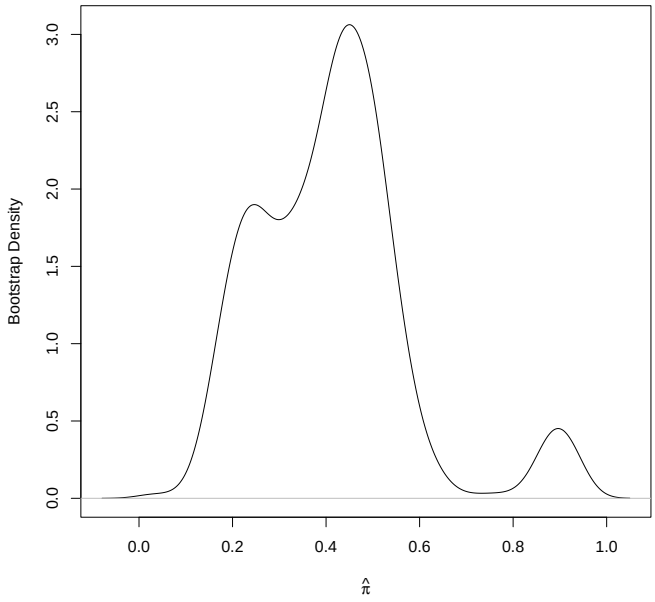

Figure 4.14: Bootstrap density for $\hat{\pi}$ for scores for quiz 3.

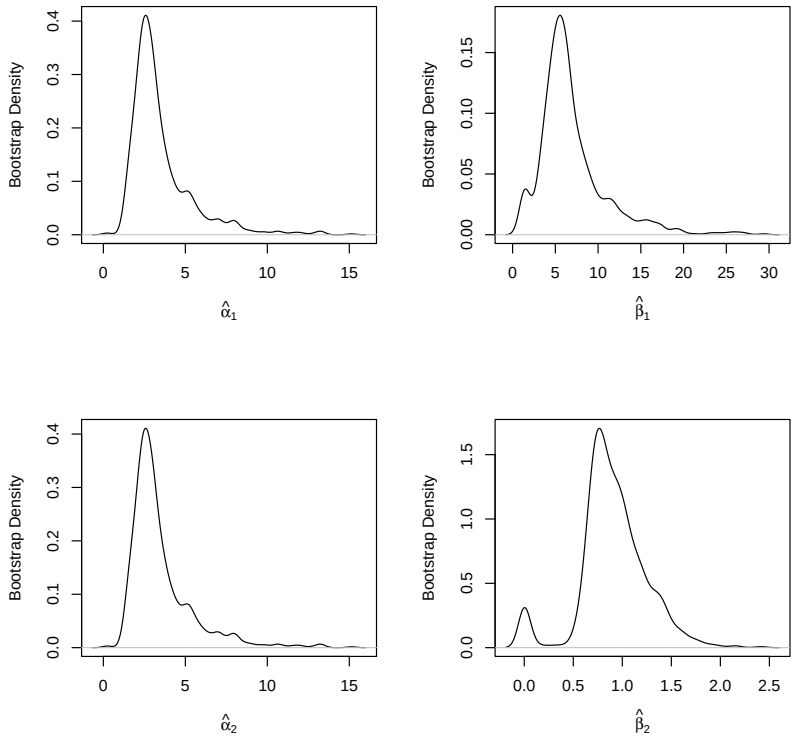

Figure 4.15: Bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ for scores for quiz 3.

(0.2997, 0.8055), which on the surface would suggest bimodality. However, since the data are not consistent with the model involving a mixture of two betas, this prevents us from making any conclusion about bimodality.

4.7 Analysis of the Quiz 4 Scores Using the Likelihood Approach

The maximum likelihood estimates for the five parameters of the beta-mixture model were: $\hat{\pi} = 0.1692$, $\hat{\alpha}_1 = 1.2689$, $\hat{\beta}_1 = 2.8958$, $\hat{\alpha}_2 = 2.6694$ and $\hat{\beta}_2 = 1.5534$. The empirical beta mixture model does not visually fit the data well (Figure 4.19). The goodness-of-fit test had a value of 106.532, which is much larger than the critical value 19.6751. This indicates that the data are not consistent with the model, involving mixture of two betas. The bootstrap density for $\hat{\pi}$ is given in Figure 4.20 and the bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ are given in Figure 4.21. The 95% bootstrap interval for π is (0, 0.8108), which would indicate unimodality if the empirical model were appropriate.

4.8 Analysis of the Quiz 4 Scores Using the Bayesian Approach

The posterior means for the five parameters are: $\bar{\pi} = 0.709$, $\bar{\alpha}_1 = 2.5545$, $\bar{\beta}_1 = 2.1287$, $\bar{\alpha}_2 = 1.1402$ and $\bar{\beta}_2 = 0.6997$. The posterior distribution for π is given in Figure 4.22 and the posterior distributions for α_1, β_1, α_2 and β_2 are given in Figure 4.23. The plot for the goodness-of-fit test is given in Figure 4.24. The plot indicates that the Bayesian model fits the data well. The trace plots show that all chains converged. The 95% hpd for π is (0.0781, 0.9994), which on the surface would suggest unimodality.

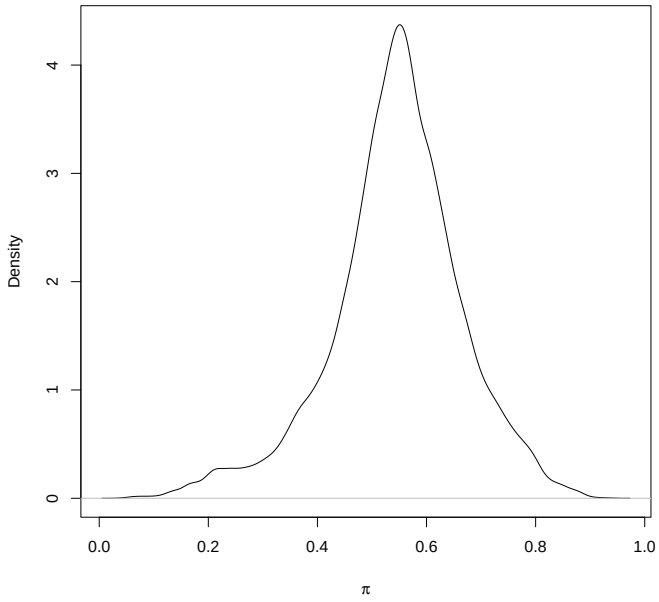

Figure 4.16: The posterior distribution for π for scores for quiz 3.

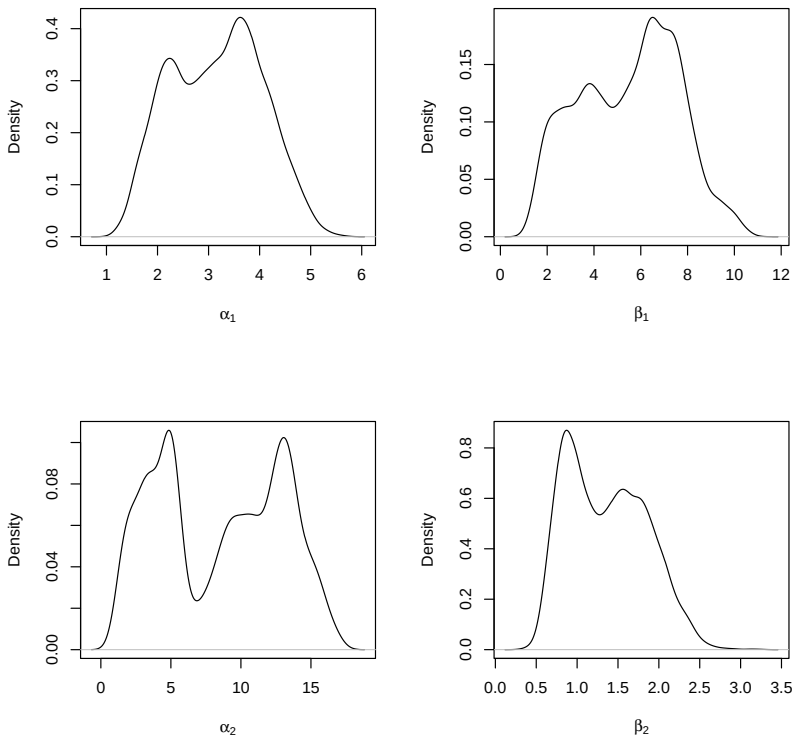

Figure 4.17: The posterior distributions for α_1, β_1, α_2 and β_2 for scores for quiz 3.

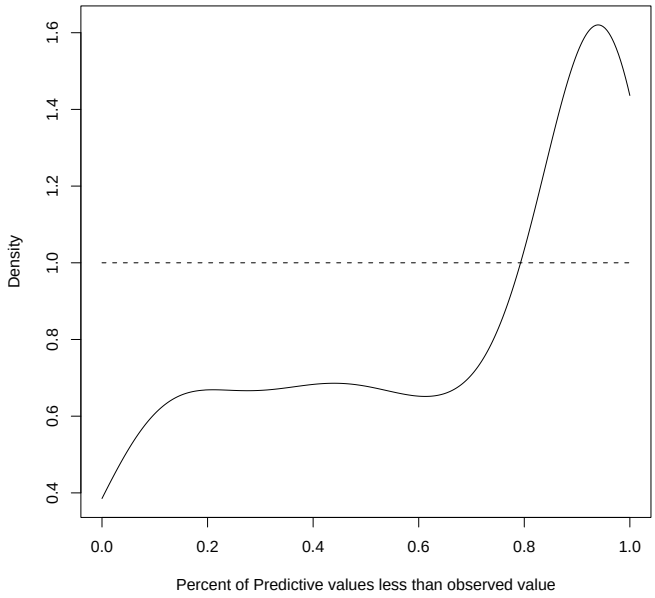

Figure 4.18: The goodness fit test for the Bayesian model for the scores for quiz 3. The model is considered to fit the data well if the kernel density of the posterior predictive quantiles (solid line) approximates the uniform density (dashed line).

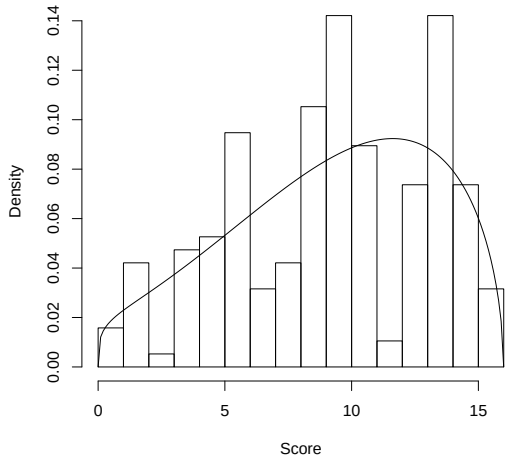

Figure 4.19: Histogram for scores for quiz 4 with fitted beta mixture model.

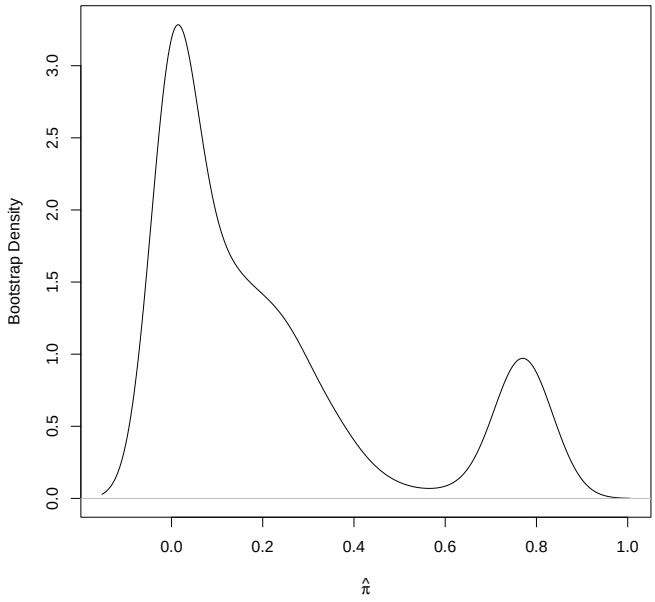

Figure 4.20: Bootstrap density for $\hat{\pi}$ for scores for quiz 4.

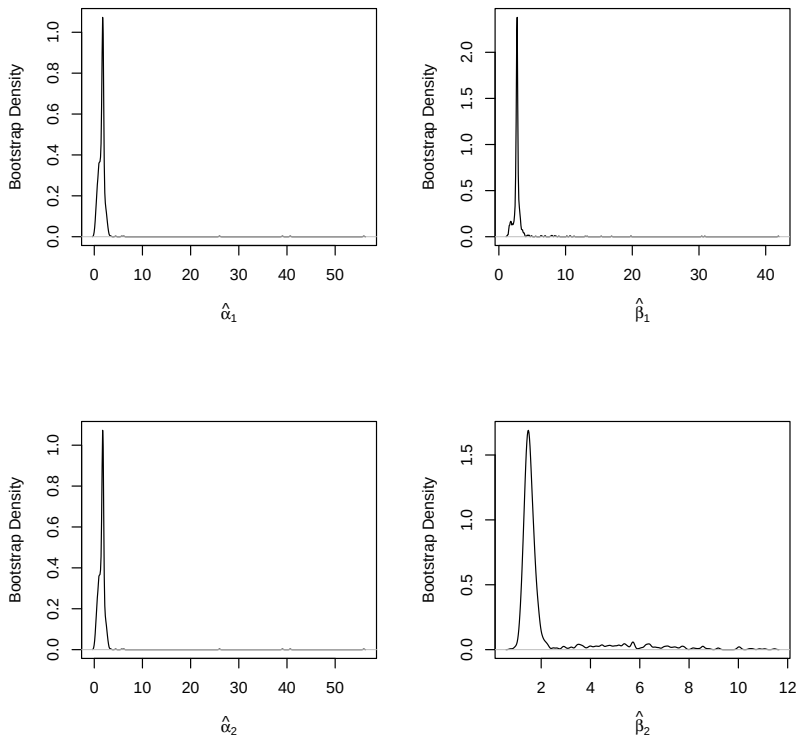

Figure 4.21: Bootstrap densities for $\hat{\alpha}_1$, $\hat{\beta}_1$, $\hat{\alpha}_2$ and $\hat{\beta}_2$ for scores for quiz 4.

However, since the data are not consistent with the model involving a mixture of two betas, this prevents us from making any conclusion about unimodality.

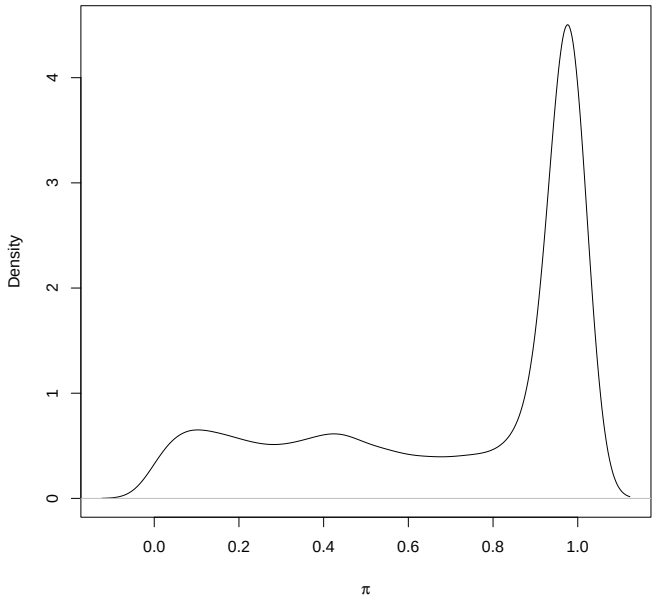

Figure 4.22: The posterior distribution for π for scores for quiz 4.

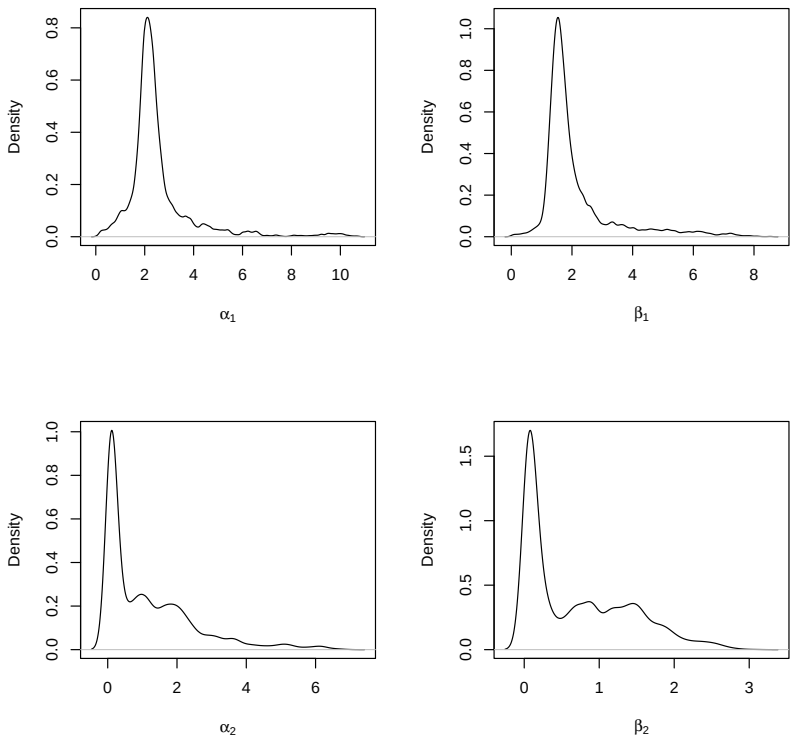

Figure 4.23: The posterior distributions for α_1, β_1, α_2 and β_2 for scores for quiz 4.

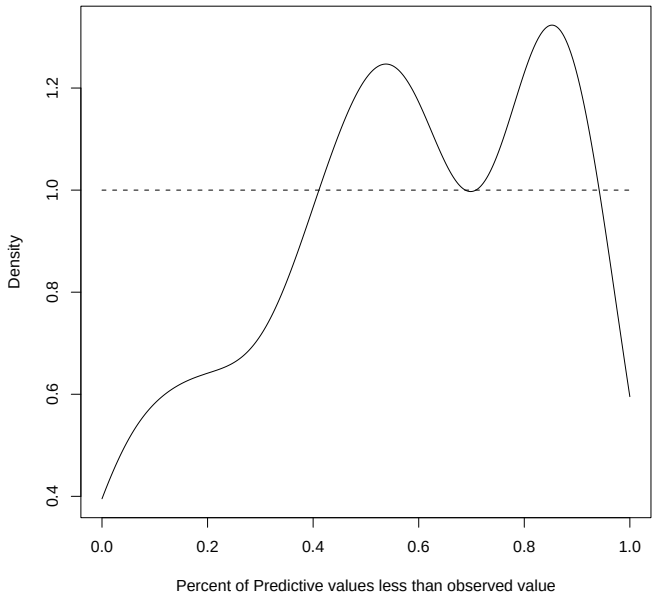

Figure 4.24: The goodness fit test for the Bayesian model for the scores for quiz 4. The model is considered to fit the data well if the kernel density of the posterior predictive quantiles (solid line) approximates the uniform density (dashed line).

Chapter 5

Conclusion

A summary of results from the two approaches is given in Table 5.1 and Table 5.2. For the first quiz, both methods produced confidence intervals for π which included the value 0.025. Also, both goodness-of-fit tests indicated that the data are consistent with the beta-mixture model. Thus we can conclude that there is no bimodality. For the second quiz, neither confidence interval included 0.025. However, both goodness-of-fit tests indicated that the data are not consistent with the model. This lack-of-fit prevents us from making any decision about bimodality. For the third quiz, neither confidence interval included 0.025. Again, both goodness-of-fit tests showed that the data are not consistent with the model, although the fit was visually good. So we cannot make a decision about bimodality. For the fourth quiz, one interval included 0.025 while the

95% Bootstrap Interval for π	**Goodness-of-fit Test**	**Conclusion**
(0.02, 0.53)	Yes	Unimodal
(0.04, 0.95)	No	Not Available
(0.17, 0.89)	No	Not Available
(0, 0.81)	No	Not Available

Table 5.1: Summary Table of the Results from Likelihood Approach

95% Posterior Interval for π	**Goodness-of-fit Test**	**Conclusion**
(0, 0.78)	Yes	Unimodal
(0.51, 0.84)	No	Not Available
(0.30, 0.81)	No	Not Available
(0.08, 1)	No	Not Available

Table 5.2: Summary Table of the Results from Bayesian Approach

other did not. Once again, both goodness of fit tests showed that the data are not consistent with the model. So we cannot make any conclusion about the fourth quiz. Upon more careful examination, the reason for lack-of-fit for the quiz 4 data is that a three-component model may be more appropriate.

To compare the two approaches, both methods required a great deal of computing time. The likelihood approach took much longer and Proc NLP has limitations. It cannot handle large data sets while using nms option. Also, it often failed to converge. However, the goodness-of-fit test associated with the likelihood approach was much easier than that for Bayesian approach. Also, the Bayesian goodness-of-fit test does not have a strict decision rule.

Initially, when we approached this problem, we assumed that there were 15 questions, all of which were independent. In fact, there are three questions for each test, each worth five points. This partially explains why the models did not fit well, especially the Bayesian model. However, we hope the methods that we developed to test bimodality can be used in similar situations and generalized.

The methods used in this study were only partially successful in detecting bimodality. The beta-mixture model only fit well for one data set, so conclusions about

bimodality were hard to draw. Also, the cutoff value of 0.025 for drawing the conclusion of bimodality was arbitrary. Instead of only looking at the mixing parameter with an arbitrary cutoff, we should perhaps consider all parameters in assessing bimodality.

Appendix A

```
data scores; infile 'g:/project/final/final3.txt'; input score;
run; data scores; set scores; where score<>.; proc print; run;

proc nlp data=scores tech=nms maxfunc=300000 maxiter=100000 ; max
logf; lincon
       p>0,
       p<1,
       a2>0,
       b2>0,
       a1>0,
       b1>0;
nlincon c2-c1>0; c1=a1*b2; c2=a2*b1; parms  p=.5, a1=1,
b1=2.75,a2=2, b2=.5;
logf=log(p*(probbeta((score+1)/16,a1,b1)-probbeta((score)/16,a1,b1))+
     (1-p)*(probbeta((score+1)/16,a2,b2)-probbeta((score)/16,a2,b2)));
run;
%macro profile (n=);
%do i=0 %to &n-2 %by 1;
ods listing close; proc surveyselect data=scores out=testing
sampsize=190 method=urs outhits; run; proc nlp data=testing
tech=nms maxfunc=300000 maxiter=100000 outvar=new; max logf;
lincon p>0,
       p<1,
       a2>0,
       b2>0,
       a1>0,
       b1>0;
nlincon c1-c2<0; c1=a1*b2; c2=a2*b1; parms  p=.5, a1=1, b1=2.75,
a2=2, b2=.5;
logf=log(p*(probbeta((score+1)/16,a1,b1)-probbeta((score)/16,a1,b1))+
     (1-p)*(probbeta((score+1)/16,a2,b2)-probbeta((score)/16,a2,b2)));
run; data new; set new; where _TYPE_="PARMS"; run; data new5 ; if
&i=0 then set new(); else set new new5; run;
%end;
ods listing;
%mend profile;
```

Appendix B

```
v=seq(0,16, by=1)
s<-read.table("G:/project/final/final1.txt",col.names="Q")
hist(s\$Q, breaks=v,xlab='Score', include.lowest = TRUE,
right=FALSE,main="") x<-seq(0,15,by=.1) pi<-.1604 a1<-0.9451
b1<-2.8284 a2<-2.2243 b2<-0.7905
prob<-(pi*(pbeta((x+1)/16,a1,b1)-pbeta((x)/16,a1,b1))+
     (1-pi)*(pbeta((x+1)/16,a2,b2)-pbeta((x)/16,a2,b2)))
lines(x,190*prob) dev.copy2eps(file="G:\\project\\final\\sg1.eps")
```

Appendix C

```
s<-read.table("c:/project/final1.txt",col.names="Q1")
t1<-as.matrix(s$Q1)

#remove missing value t1<-t1[!is.na(t1)] k<-length(t1) m<-15

length <-20000 burn <- 3000000

candsig.y<-.09 candsig.a1 <- .15 candsig.b1 <- .15 candsig.a2 <-.1
candsig.b2 <- .12 candsig.p<- .06

y<-matrix(.4,ncol=k,nrow=(length+burn)) a1 <- numeric(length+burn)
b1 <- numeric(length+burn) a2 <- numeric(length+burn)
b2<-numeric(length+burn) p <- numeric(length+burn)

a1[1]<-23/27 b1[1]<-46/27 a2[1]<-46/27 b2[1]<-23/27 p[1]<-.4

a1.theta<-4 a1.kapa<-1 b1.theta<-8 b1.kapa<-1 a2.theta<-8 a2.kapa<-1
b2.theta<-4 b2.kapa<-1

for(i in 2:(length+burn)){

# update yi (j = 1,...,k)
    for (j in 1:k){
                y[i,j]<-y[i-1,j]
                old<-y[i-1,j]
                new<-rnorm(1,old,candsig.y)
                if ((new>=0) & (new<=1))
                {
                   llo<-log(dbinom(t1[j],m,old))
                    + log(p[i-1]*dbeta(old,a1[i-1],b1[i-1])
                    +(1-p[i-1])*dbeta(old,a2[i-1],b2[i-1]))
                   lln<-log(dbinom(t1[j],m,new))
                    + log(p[i-1]*dbeta(new,a1[i-1],b1[i-1])
                    +(1-p[i-1])*dbeta(new,a2[i-1],b2[i-1]))
        uu<-runif(1,0,1)
            if(log(uu)<(lln-llo)){y[i,j]<-new}
                }
```

```
       }


# update for a1
    a1[i] <- a1[i-1]
    old <- a1[i-1]
    new <- rnorm(1,old,candsig.a1)
        if (new>0)
        {
    llo <-sum(log(p[i-1]*dbeta(y[i,],old,b1[i-1])
    +(1-p[i-1])*dbeta(y[i,],a2[i-1],b2[i-1])))
     + log(dgamma(old,shape=a1.kapa,scale=a1.theta))
    lln <-sum(log(p[i-1]*dbeta(y[i,],new,b1[i-1])
    +(1-p[i-1])*dbeta(y[i,],a2[i-1],b2[i-1])))
    + log(dgamma(new,shape=a1.kapa,scale=a1.theta))
        uu<-runif(1,0,1)
    if(log(uu)<(lln-llo)){a1[i]<-new}
        }

# update for b1
    b1[i] <- b1[i-1]
    old <- b1[i-1]
    new <- rnorm(1,old,candsig.b1)
        if (new>0)
        {
    llo <-sum(log(p[i-1]*dbeta(y[i,],a1[i],old)
    +(1-p[i-1])*dbeta(y[i,],a2[i-1],b2[i-1])))
    + log(dgamma(old,shape=b1.kapa,scale=b1.theta))
    lln <-sum(log(p[i-1]*dbeta(y[i,],a1[i],new)
    +(1-p[i-1])*dbeta(y[i,],a2[i-1],b2[i-1])))
     + log(dgamma(new,shape=b1.kapa,scale=a1.theta))
        uu<-runif(1,0,1)
    if(log(uu)<(lln-llo)){b1[i]<-new}
        }


# update for a2
    a2[i] <- a2[i-1]
    old <- a2[i-1]
    new <- rnorm(1,old,candsig.a2)
        if (new>0)
        {
    llo <-sum(log(p[i-1]*dbeta(y[i,],a1[i],b1[i])
    +(1-p[i-1])*dbeta(y[i,],old,b2[i-1])))
     + log(dgamma(old,shape=a2.kapa,scale=a2.theta))
    lln <-sum(log(p[i-1]*dbeta(y[i,],a1[i],b1[i])
    +(1-p[i-1])*dbeta(y[i,],new,b2[i-1])))
     + log(dgamma(new,shape=a2.kapa,scale=a2.theta))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){a2[i]<-new}
        }
```

```
# update for b2
    b2[i] <- b2[i-1]
    old <- b2[i-1]
    new <- rnorm(1,old,candsig.b2)
        if (new>0& new<(a2[i]*b1[i]/a1[i]))
        {
    llo <-sum(log(p[i-1]*dbeta(y[i,],a1[i],b1[i])
    +(1-p[i-1])*dbeta(y[i,],a2[i],old)))
     + log(dgamma(old,shape=b2.kapa,scale=b2.theta))
    lln <-sum(log(p[i-1]*dbeta(y[i,],a1[i],b1[i])
    +(1-p[i-1])*dbeta(y[i,],a2[i],new)))
     + log(dgamma(new,shape=b2.kapa,scale=b2.theta))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){b2[i]<-new}
        }


# update for p
    p[i] <- p[i-1]
    old <- p[i-1]
    new <- rnorm(1,old,candsig.p)
        if (new>=0 & new<=1)
        {
    llo <-sum(log(old*dbeta(y[i,],a1[i],b1[i])
    +(1-old)*dbeta(y[i,],a2[i],b2[i])))
    lln <-sum(log(new*dbeta(y[i,],a1[i],b1[i])
    +(1-new)*dbeta(y[i,],a2[i],b2[i])))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){p[i]<-new}
        }

}

# get mean and variance of five parameters

mean(p[burn+1:length+burn])
mean(a1[burn+1:length+burn])
mean(b1[burn+1:length+burn])
mean(a2[burn+1:length+burn])
mean(b2[burn+1:length+burn])
var(p[burn+1:length+burn])
var(a1[burn+1:length+burn])
var(b1[burn+1:length+burn])
var(a2[burn+1:length+burn])
var(b2[burn+1:length+burn])


# get 95% hpd

hp<-sort(p[burn+1:length+burn])
ha1<-sort(a1[burn+1:length+burn])
```

```
hb1<-sort(b1[burn+1:length+burn])
ha2<-sort(a2[burn+1:length+burn])
hb2<-sort(b2[burn+1:length+burn])

hpd<-matrix(NA,500,3)

for (i in 1:500) {
 hpd[i,1]<-hp[i]
 hpd[i,2]<-hp[19499+i]
 hpd[i,3]<-hpd[i,2]-hpd[i,1]
}
hpd[hpd[,3]==min(hpd[,3]),c(1,2)]

n<-5000 t1<-s\$Q1 k<-length(t1) py<-matrix(0,ncol=k-1, nrow=n)
ps<-matrix(0,ncol=k,nrow=n) pc<-numeric(k) pp<-numeric(n) pa1
<-numeric(n) pb1 <- numeric(n) pa2 <- numeric(n) pb2 <- numeric(n)
i<-150000 pa1[1]<-a1[i] pb1[1]<-b1[i] pa2[1]<-a2[i] pb2[1]<-b2[i]
pp[1]<-p[i]

pa1.theta<-4 pa1.kapa<-1 pb1.theta<-8 pb1.kapa<-1 pa2.theta<-8
pa2.kapa<-1 pb2.theta<-4 pb2.kapa<-1


candsig.py<-.09 candsig.pa1 <- .15 candsig.pb1 <- .15 candsig.pa2<-
.1 candsig.pb2 <- .12 candsig.pp<- .06

for (x in 1: k) { t1<-s$Q1[-x] py[1,]<-y[i,-x]

for(i in 2:n){

# update pyi (j = 1,...,k-1)
    for (j in 1:(k-1)){
                py[i,j]<-py[i-1,j]
                old<-py[i-1,j]
                new<-rnorm(1,old,candsig.py)
                if ((new>=0) & (new<=1))
                {
                   llo<-log(dbinom(t1[j],m,old))
                   + log(pp[i-1]*dbeta(old,pa1[i-1],pb1[i-1])
                   +(1-pp[i-1])*dbeta(old,pa1[i-1],pb2[i-1]))
                   lln<-log(dbinom(t1[j],m,new))
                    + log(pp[i-1]*dbeta(new,pa1[i-1],pb1[i-1])
                    +(1-pp[i-1])*dbeta(new,pa1[i-1],pb2[i-1]))
        uu<-runif(1,0,1)
            if(log(uu)<(lln-llo)){py[i,j]<-new}
                }
       }

# update for pa1
    pa1[i] <- pa1[i-1]
    old <- pa1[i-1]
    new <- rnorm(1,old,candsig.pa1)
```

```
        if (new>0)
        {
    llo <-sum(log(pp[i-1]*dbeta(py[i,],old,pb1[i-1])
    +(1-pp[i-1])*dbeta(py[i,],pa2[i-1],pb2[i-1])))
    + log(dgamma(old,shape=pa1.kapa,scale=pa1.theta))
    lln <-sum(log(pp[i-1]*dbeta(py[i,],new,pb1[i-1])
    +(1-pp[i-1])*dbeta(py[i,],pa2[i-1],pb2[i-1])))
    + log(dgamma(new,shape=pa1.kapa,scale=pa1.theta))
        uu<-runif(1,0,1)
    if(log(uu)<(lln-llo)){pa1[i]<-new}
        }

# update for pb1
    pb1[i] <- pb1[i-1]
    old <- pb1[i-1]
    new <- rnorm(1,old,candsig.pb1)
        if (new>0)
        {
    llo <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],old)
    +(1-pp[i-1])*dbeta(py[i,],pa2[i-1],pb2[i-1])))
     + log(dgamma(old,shape=pb1.kapa,scale=pb1.theta))
    lln <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],new)
    +(1-pp[i-1])*dbeta(py[i,],pa2[i-1],pb2[i-1])))
     + log(dgamma(new,shape=pb1.kapa,scale=pa1.theta))
        uu<-runif(1,0,1)
    if(log(uu)<(lln-llo)){pb1[i]<-new}
        }

# update for pa2
    pa2[i] <- pa2[i-1]
    old <- pa2[i-1]
    new <- rnorm(1,old,candsig.pa1)
        if (new>0)
        {
    llo <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],pb1[i])
    +(1-pp[i-1])*dbeta(py[i,],old,pb2[i-1])))
     + log(dgamma(old,shape=pa2.kapa,scale=pa2.theta))
    lln <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],pb1[i])
    +(1-pp[i-1])*dbeta(py[i,],new,pb2[i-1])))
     + log(dgamma(new,shape=pa2.kapa,scale=pa2.theta))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){pa2[i]<-new}
        }

# update for pb2
    pb2[i] <- pb2[i-1]
    old <- pb2[i-1]
    new <- rnorm(1,old,candsig.pb2)
        if (new>0&new<(a2[i]*b1[i]/a1[i]))
        {
    llo <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],pb1[i])
    +(1-pp[i-1])*dbeta(py[i,],pa2[i],old)))
```

```
     + log(dgamma(old,shape=pb2.kapa,scale=pb2.theta))
    lln <-sum(log(pp[i-1]*dbeta(py[i,],pa1[i],pb1[i])
    +(1-pp[i-1])*dbeta(py[i,],pa2[i],new)))
    + log(dgamma(new,shape=pb2.kapa,scale=pb2.theta))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){pb2[i]<-new}
        }

# update for pp
    pp[i] <- pp[i-1]
    old <- pp[i-1]
    new <- rnorm(1,old,candsig.p)
        if (new>=0 & new<=1)
        {
    llo <-sum(log(old*dbeta(py[i,],pa1[i],pb1[i])
    +(1-old)*dbeta(py[i,],pa2[i-1],pb2[i])))
    lln <-sum(log(new*dbeta(py[i,],pa1[i],pb1[i])
    +(1-new)*dbeta(py[i,],pa2[i-1],pb2[i])))
    uu<-runif(1,0,1)
        if(log(uu)<(lln-llo)){pp[i]<-new}
        }
# generate predictive distribution
 ps[i,x]<-rbinom(1,15,rbeta(1,pa2[i],pb2[i]))
uu<-runif(1,0,1)
 if (pp[i]>uu){ps[i,x]<-rbinom(1,15,rbeta(1,pa1[i],pa2[i]))}

} pc[x]<-length(ps[ps[,x]<=s$Q1[x],x])/n } for( x in 1:190) {
 pc[x]<-(length(ps[ps[-1,x]<s$Q1[x],x])-1)/(n-1)
} plot(density(pc),xlab="Percent of predictive values less than
observed value",main="")
dev.copy2eps(file="G:\\project\\final\\bg1.eps")
```

Bibliography

Aitkin, M., Anderson, D., and Hinde, J. (1981), "Statistical modeling of data on teaching styles", *Journal of the Royal Statistical Society*, 144, 419–461.

Aitkin, M. and Rubin, D. (1985), "Estimation and hypothesis testing in finite mixture models", *Journal of the Royal Statistical Society*, 47, 67–75.

Davison, A. and Hinkley, D. (1997), *Bootstrap Methods and Their Application*, Cambridge: Cambridge University Press.

Dunn, S., Janos, L., and Rosenfeld, A. (1983), "Bimean Clustering", *Pattern Recognition*, 1, 169–173.

Efron, B. (1979), "Bootstrap methods: another look at the jackknife", *Annals of Statistics*, 7, 1–26.

Gelfand, A. E. and Solomon, N. (1975), "Analyzing the decision making process of the American jury", *Journal of the American Statistical Association*, 70, 305–310.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996), *Markov Chain Monte Carlo in Practice*, London: Chapman & Hall.

Harris, C. M. (1983), "On finite mixtures of geometric and negative binomial distributions", *Communications in Statistics*, 12, 987–1007.

Hastings, W. K. (1970), "Monte Carlo sampling methods using Markov chains and its applications", *Biometrika*, 57, 97–109.

Hosmer, D. W. (1973), "A comparison of iterative maximum likelihood estimates of the parameters of a mixture of two normal distributions under three different types of sample", *Biometrics*, 29, 761–770.

Lehmann, E. L. (1983), *Theory of Point Estimation*, New York: John Wiley & Sons Ltd.

Lu, X., Zhang, W., and Qin, Z. S. (2004), "Statistical resynchronization and Bayesian detection of periodically expressed genes", *Nucleic Acids Research*, 32, 447–455.

McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, London: Chapman & Hall.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equations of state calculations by fast computing machines", *Journal of Chemical Physics*, 21, 1087–1091.

Nelder, J. A. and Mead, Y. R. (1965), "A Simplex Method for Function Minimization", *Computer Journal*, 7, 308–311.

Paull, A. E. (1978), "A generalized compound Poisson model for consumer purchase panel data analysis", *Journal of the American Statistical Association*, 73, 706–713.

Pawitan, Y. (2001), *In All Likelihood: Statistical Modelling and Inference Using Likelihood*, Oxford: Oxford University Press.

Phillips, T. Y., Rosenfeld, A., and Sher, A. C. (1989), "O (log n) Bimodality Analysis", *Pattern Recognition*, 22, 741–746.

Reschenhofer, E. (2001), "The Bimodality Principle", *Journal of Statistics Education*, 9.

Robert, P. R. and Rousseau, J. (2002), "A Mixture Approach to Bayesian Goodness of Fit", *Pre Publication*.

SAS (2004), *SAS/OR User's Guide: Mathematical Programming*, Cary, NC: SAS Institute Inc.

Titterington, D. M., Smith, A. F. M., and Makov, U. E. (1985), *Statistical Analysis of Finite Mixture Distributions*, New York: John Wiley & Sons Ltd.