ABSTRACT

Logistic Regression with Misclassified Response and Covariate Measurement Error: A Bayesian Approach

Anna E. McGlothlin, Ph.D.

Mentors: James D. Stamey, Ph.D. and John W. Seaman, Jr., Ph.D.

In a variety of regression applications, measurement problems are unavoidable because infallible measurement tools may be expensive or unavailable. When modeling the relationship between a response variable and covariates, we must account for the uncertainty that is inherently introduced when one or both of these variables are measured with error. In this dissertation, we explore the consequences of and remedies for imperfect measurements.

We consider a Bayesian analysis for modeling a binary outcome that is subject to misclassification. We investigate the use of informative conditional means priors for the regression coefficients. Additionally, we incorporate random effects into the model to accommodate correlated responses. Markov chain Monte Carlo methods are utilized to perform the necessary computations. We use the deviance information criterion to aid in model selection.

Next, we consider data where measurements are flawed for both the response and explanatory variables. Our interest is in the case of a misclassified dichotomous response and a continuous covariate that is unobservable, but where measurements are available on

its surrogate. A logistic regression model is developed to incorporate the measurement error in the covariate as well as the misclassification in the response. The methods developed are illustrated through an example. Results from a simulation experiment are provided illustrating advantages of the approach.

Finally, we expand this model to incorporate random effects, resulting in a generalized linear mixed model for a misclassified response and covariate measurement error. We demonstrate the use of this model using a simulated data set.

Logistic Regression with Misclassified Response and Covariate Measurement Error: A Bayesian Approach

by

Anna E. McGlothlin, B.S., M.A.

A Dissertation

Approved by the Department of Statistical Science

_____
Jack D. Tubbs, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of
Doctor of Philosophy

Approved by the Dissertation Committee

_____
James D. Stamey, Ph.D., Co-chairperson

_____
John W. Seaman, Jr., Ph.D., Co-chairperson

_____
Tom L. Bratcher, Ph.D.

_____
Jack D. Tubbs, Ph.D.

_____
Dean M. Young, Ph.D.

_____
Steven L. Green, Ph.D.

Accepted by the Graduate School
May 2007

_____
J. Larry Lyon, Ph.D., Dean

Page bearing signatures is kept on file in the Graduate School.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I must express my immense gratitude to Dr. Seaman and Dr. Stamey for their guidance through this entire process. They have been incredibly generous with their time and offered invaluable advice and encouragement. I have learned so much through their knowledgeable insights, not only about research, but more importantly about life in general. I am truly honored to have worked with such wonderful individuals.

I am especially appreciative of the entire faculty in the Department of Statistical Science. The concern that the professors show for the success of their students makes Baylor a unique place. In particular, I want to thank Dr. Jeanne Hill, who first taught me to enjoy Statistics, and Dr. Jack Tubbs, who persuaded me to embark on this journey. I have also been blessed to work alongside some truly remarkable peers, who offered encouragement and cheered me along each step of the way.

I could never have made it to this point without my parents. They fostered in me a love of learning, and taught me the importance of perseverance. Their unwavering love and support have made so many things possible for me and I will be eternally grateful.

I also wish to thank my good friend Lisa Fahmy for sharing the ups and downs, the laughter and the tears, for always lending an ear, and for pushing me to be my best.

Above all, thanks to the One who plants and waters and causes all things to grow. He has given me blessings too numerous to count and asked for nothing in return except for my whole heart.

CHAPTER ONE

Introduction

1.1 Overview of the Problem

There are many regression applications in which the variables of interest cannot be measured perfectly; that is, without error. Suppose that we wish to discover the relationship between a response variable Y and an explanatory variable X. Complications arise when one or both of these variables are measured imperfectly. Mismeasurement of a continuous variable is typically known as "measurement error," while the term "misclassification" refers to categorical variables. A naïve analysis that ignores measurement error or misclassification will result in biased parameter estimates.

In this chapter, we present examples that demonstrate the consequences of ignoring measurement problems. In Section 1.2, we show how estimates of regression coefficients are biased in simple linear regression with an imperfectly measured covariate. In Section 1.3, we turn our attention to misclassification of a binary response variable. Section 1.4 presents an overview of the dissertation.

1.2 The Impact of Covariate Measurement Error

The consequences of covariate measurement error have received much attention in the literature. Carroll, Ruppert, Stefanski, and Crainiceanu (2006) provide numerous examples of measurement error and discuss methods for appropriately adjusting an analysis. The following example is adapted from Carroll, et al. (2006). Consider the regression of a response Y on a predictor X according to the model $Y = \beta_0 + \beta_1 X + \varepsilon$,

where $\varepsilon \sim \text{Normal}(0, \sigma^2)$. A simulated data set from this model, with independent errors, $\beta_0 = 0$, $\beta_1 = 1$, and $\sigma = 0.25$ is exhibited in Figure 1a, along with the least squares fit for the line. The latter yields estimates of the slope and standard error of 1.002 and 0.062, respectively. Now suppose that we contaminate the data so that $X^* = X + U$, where $U \sim$ Normal(0, 1), independent of Y. This is the classical additive measurement error model; see Section 3.1.1. In Figure 1b, we display a scatterplot of the error-prone data (X^*, Y) with the fitted regression line. Visually, the fitted regression line for the error-prone data is flatter than for the true data. The estimated slope for the contaminated data is 0.4102 with a standard error of 0.156. That is, the effect of ignoring the measurement error is to bias the slope estimate toward zero. This flattening is known as attenuation. We also see that the (X, Y) data are more tightly clustered around the regression line, while the error-prone data have much more variability.

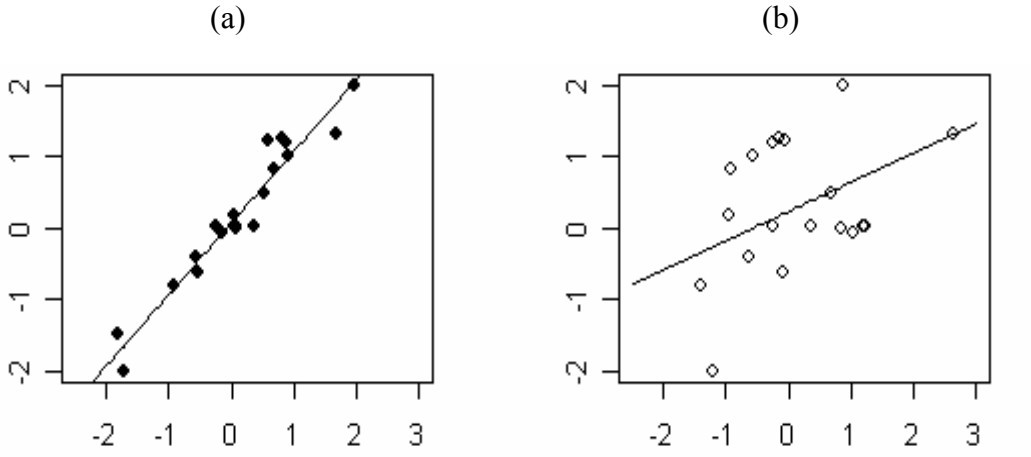

Figure 1. Least Squares Regression Lines for (a) the Error-free Data (X, Y) and (b) the Contaminated Data (X, Y^*).

In Figure 2, we combine the two data sets. The solid circles and solid line are the (X, Y) data and the least squares fit. The open circles and the dotted line are the (X^*, Y)

data and the least squares fit. For this example, we see that disregarding the measurement error will result in underestimating the strength of the relationship between *X* and *Y*. For more examples of measurement error, see Carroll, et al. (2006). For a Bayesian treatment, see Gustafson (2004).

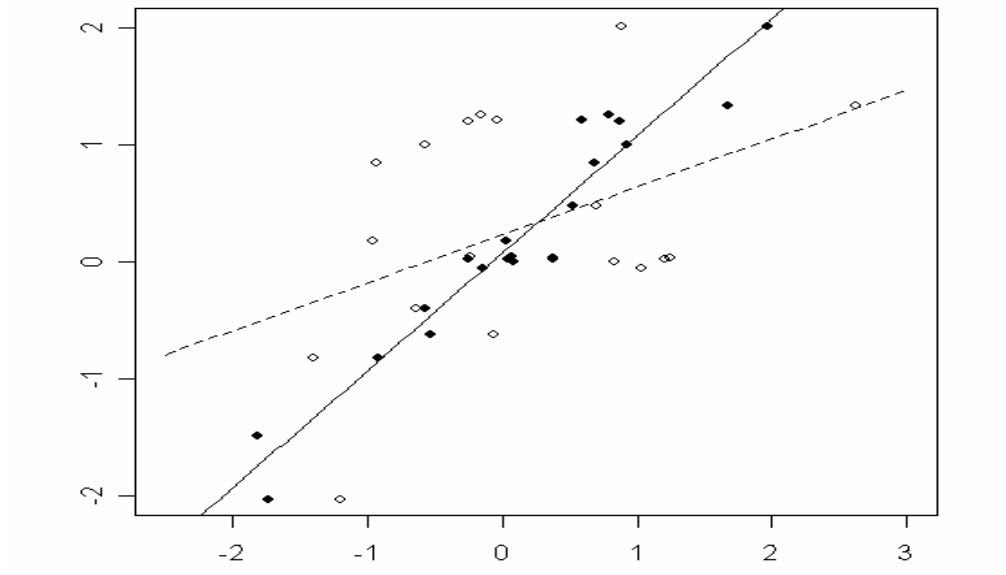

Figure 2. Illustration of Attenuation of the Slope for a Simple Linear Regression. The solid line and solid points are for the true *X* data, while the dotted line and open circles are for the error-prone *X** data.

1.3 The Impact of Response Misclassification

We now consider the scenario in which the response variable *Y* is categorical and is subject to misclassification. The observed response is denoted by *Y**. The following example is adapted from Gustafson (2004). Table 1 displays 500 simulated values for subjects classified by *X* and *Y*. Now suppose that the response is subject to classification error so that we actually observe *Y**. For each subject, there is a 20% chance that *Y** differs from *Y*. Table 2 shows simulated values of *Y**.

Table 1. Simulated (X, Y) Values for 500 Subjects

	$Y = 0$	$Y = 1$
$X = 0$	378	65
$X = 1$	39	18

Table 2. Simulated (X, Y*) Values for 500 Subjects

	$Y^* = 0$	$Y^* = 1$
$X = 0$	331	112
$X = 1$	34	23

We define the odds ratio as

$$OR = \frac{p_1/(1-p_1)}{p_2/(1-p_2)},$$

where p_1 is the proportion of successes for $X = 1$ and p_2 is the proportion of successes for $X = 0$. For the correctly classified data, the odds ratio is estimated to be 2.684, compared to 1.999 for the misclassified data. Thus, as with measurement error, the impact of misclassification is to underestimate the strength of the relationship between the explanatory variable and the response.

As an example of the impact of response misclassification for the case of logistic regression, we consider the following example from Carroll et al. (2006). Let the probability of a success be $\pi \equiv P(Y = 1 \mid X)$. Using the logistic link function, we have $\pi = \exp(\beta_0 + \beta_1 X)/\left[1 + \exp(\beta_0 + \beta_1 X)\right]$. Define the probabilities of correct classification as $\eta = P(Y^* = 1 \mid Y = 1)$ and $\theta = P(Y^* = 0 \mid Y = 0)$. Figure 3 illustrates the impact of response misclassification for $\eta = 0.80$ and $\theta = 0.70$. As in Carroll et al. (2006), we take $\beta_0 = -1.0$ and $\beta_1 = 1.0$. Note that the effect of misclassification is to shift the true line.

As with the previous example, the strength of the relationship between X and Y is underestimated when the misclassification is not accounted for in the model.

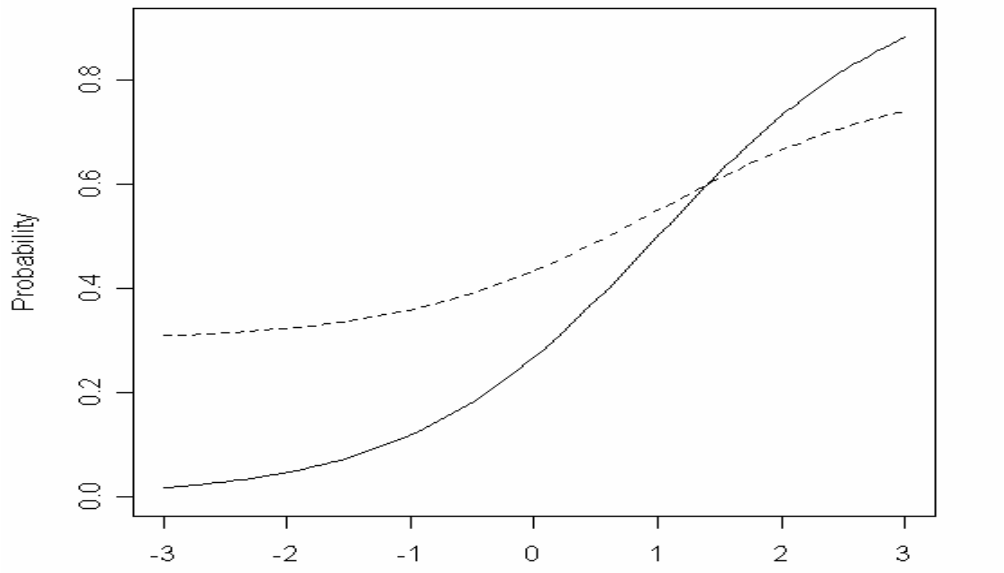

Figure 3. The Effect of Response Misclassification in Logistic Regression. True probability of success (solid line) and observed probability of success (dotted line).

1.4 Plan of the Dissertation

The examples in this chapter illustrate that measurement error and misclassification should not be ignored. In this dissertation, we investigate Bayesian methods to analyze data that is subject to measurement error and misclassification. We also consider a random effects analysis to accommodate extra variation.

The remainder of the dissertation is organized as follows. In Chapter 2, we extend the work of McInturff, Johnson, Cowling, and Gardner (2004) to analyze misclassified binomial data, assuming that all covariates are perfectly measured. A random effect is introduced to incorporate extra variation. In Chapter 3, we consider a model for data with a misclassified response and a continuous covariate which cannot be measured perfectly. The methods developed are illustrated through an example. In

Chapter 4, we develop a mixed effects model for misclassified binary data with covariate measurement error. Concluding remarks are made in Chapter 5. Finally, Chapters Two, Three, and Four are essentially self-contained and each chapter has its own literature review as well as its own statement of conclusions.

CHAPTER TWO

Mixed Model for a Misclassified Binomial Response

The generalized linear model (GLM) (McCullagh and Nelder 1989) is one of the most useful tools in statistics because it provides a unified framework for a handling a broad range of regression models, including continuous and discrete responses. Complications can arise in the analysis of GLM's when the response variable is measured with error. The term "measurement error" typically describes continuous variables that are measured imperfectly. Categorical variables which are measured with error are said to be "misclassified." When misclassification of a dichotomous response variable is ignored, the resulting estimates of regression coefficients will be biased. If the misclassification rates are independent of any covariates in the model (non-differential misclassification), the direction of the bias is always towards zero (Armstrong 1998). Adjustments must be made to account for this bias when misclassification is present.

One of the assumptions for the GLM is that the responses are independent. However, for binomial data, it is reasonable that observations within a cluster will tend to be more alike than observations from different clusters. For example, in longitudinal data sets, repeated observations made on each subject will be correlated. An analysis that assumes independence is inappropriate (Agresti 2002). The dependence structure may be accommodated by including random effects, resulting in a mixed model. The generalized linear mixed model (GLMM) is an extension of the GLM that allows both fixed effects and random effects. Paulino, Silva, and Achcar (2005) develop a mixed model for misclassified binomial counts.

This chapter is an extension of the work by McInturff, Johnson, Cowling, and Gardner (2004). These authors consider a Bayesian fixed effects regression analysis for a binary response that is subject to misclassification. Using expert opinion, the authors construct informative prior distributions for the regression coefficients. Models utilizing three different link functions are compared using Bayes factors. In this chapter, we modify the method of McInturff et al. (2004) by incorporating random effects in order to accommodate correlated binomial responses. The remainder of the chapter is organized as follows. Section 1 presents an overview of the fixed effects model used by McInturff et al. (2004). We also develop the random effects model. In Section 2, we discuss Bayesian model selection criteria that are appropriate for choosing between a fixed effects and a mixed model. Section 3 illustrates our methods for simulated data sets. In Section 4, we make concluding remarks.

2.1 Bayesian Models

We now develop both the fixed effects and mixed models for modeling a binary outcome that is subject to misclassification. The prior structures for the two models are also discussed.

2.1.1 Fixed Effects Model

Suppose that a diagnostic test is available to detect the presence or absence of a certain condition, D. Let Y designate the true status, with $Y = 1$ if D is present and $Y = 0$ otherwise. Let Y^* be the observed result of the diagnostic test. Let $\mathbf{x}$ be the $k \times 1$ vector of covariates. All covariates in the model are assumed to be measured without error. Our

interest lies in examining the relationship between the true response Y and the predictor variables.

To correct for the misclassified response, we define the sensitivity and specificity, which are the probabilities of correct classification. For binary response Y, define the sensitivity as $\eta \equiv P(Y^* = 1|Y = 1)$ and the specificity as $\theta \equiv P(Y^* = 0|Y = 0)$. We assume that sensitivity and specificity are independent of all covariates in the model. The terms "false positive" and "false negative" are also common in the misclassification literature in reference to misclassification rates, with P(false positive) = $1 - \theta$ and P(false negative) = $1 - \eta$.

For an individual with covariate information $\mathbf{x}$, let $\pi_{\mathbf{x}} \equiv P(Y = 1|\mathbf{x})$, be the probability that the individual has condition D. Then by the law of total probability, we obtain the probability that the diagnostic test detects D as

$$\begin{aligned} P(Y^* = 1 \mid \mathbf{x}) &= P(Y^* = 1 \mid Y = 1, \mathbf{x})P(Y = 1 \mid \mathbf{x}) + P(Y^* = 1 \mid Y = 0, \mathbf{x})P(Y = 0 \mid \mathbf{x}) \\ &= P(Y^* = 1 \mid Y = 1, \mathbf{x})P(Y = 1 \mid \mathbf{x}) \\ &\quad + \left[1 - P(Y^* = 0 \mid Y = 0, \mathbf{x})\right]\left[1 - P(Y = 1 \mid \mathbf{x})\right] \\ &= \eta\pi_{\mathbf{x}} + (1 - \theta)(1 - \pi_{\mathbf{x}}). \end{aligned}$$

We assume that $(Y_j^*|\mathbf{x}_j) \sim$ Bernoulli$[\eta\pi_j + (1 - \theta)(1 - \pi_j)]$, where $\pi_j = g^{-1}(\mathbf{x}_j'\boldsymbol{\beta})$ and $g(\cdot)$ is an appropriate link function. The link functions considered by McInturff et al. (2004) are

$$g(p) = \begin{cases} \exp(p)/[1 + \exp(p)] & \text{logit} \\ \Phi(p) & \text{probit} \\ 1 - \exp(-p) & \text{complementary log-log,} \end{cases}$$

where $\Phi(\cdot)$ is the standard normal cdf.

The density function for the observed data is

$$\begin{aligned} f(y_j^* \mid \mathbf{x}_j) &= \left[\pi_j\eta + (1-\pi_j)(1-\theta)\right]^{y_j^*}\left[1-\pi_j\eta-(1-\pi_j)(1-\theta)\right]^{1-y_j^*} \\ &= \left[\pi_j\eta + (1-\pi_j)(1-\theta)\right]^{y_j^*}\left[1-\pi_j\eta-1+\theta+\pi_j-\pi_j\theta\right]^{1-y_j^*} \\ &= \left[\pi_j\eta + (1-\pi_j)(1-\theta)\right]^{y_j^*}\left[\pi_j(1-\eta)+(1-\pi_j)\theta\right]^{1-y_j^*}. \end{aligned}$$

Then by definition, the likelihood function is

$$L(\boldsymbol{\beta},\eta,\theta \mid \mathbf{y}^*) \propto \prod_{j=1}^{n}\left[\pi_j\eta + (1-\pi_j)(1-\theta)\right]^{y_j^*}\left[\pi_j(1-\eta)+(1-\pi_j)\theta\right]^{1-y_j^*}, \qquad (2.1)$$

where $\mathbf{y}^* = (y_1^*,\ldots,y_n^*)$.

2.1.2 Prior Distributions for the Fixed Effects Model

Under the Bayesian framework, prior distributions are required for all unknowns in the model. For the sensitivity and specificity, we assume independent beta priors, $\eta \sim$ Beta(a_η, b_η) and $\theta \sim$ Beta(a_θ, b_θ). For the regression coefficients, we follow McInturff et al. (2004) and construct conditional means priors (CMP) as described in Bedrick, Christensen, and Johnson (1996). The CMP is attractive because it does not require elicitation of prior knowledge directly about the regression coefficients. Rather, the prior information is elicited on probabilities. These priors are constructed as follows.

Assuming k regression coefficients, we select k linearly independent covariate combinations, $\tilde{\mathbf{x}}_i$, $i = 1,\ldots, k$. For each covariate combination, expert opinion is used to specify uncertainty about the probability, $\tilde{\pi}_i \equiv P(Y=1 \mid \tilde{\mathbf{x}}_i)$, that the condition D is present. The uncertainty about these probabilities is modeled with independent Beta(a_i, b_i) distributions. Recall that $g(\tilde{\boldsymbol{\pi}}) = \tilde{\mathbf{X}}\boldsymbol{\beta}$, where $\tilde{\boldsymbol{\pi}} = (\tilde{\pi}_1,\ldots,\tilde{\pi}_k)'$, $\tilde{\mathbf{X}} = (\tilde{\mathbf{x}}_1',\ldots,\tilde{\mathbf{x}}_k')'$, and $g(\cdot)$ is a monotone link function. We construct a prior distribution for the regression

coefficient vector $\boldsymbol{\beta}$ by using the Jacobian transformation technique and the relationship $\boldsymbol{\beta} = \widetilde{\mathbf{X}}^{-1} g(\widetilde{\boldsymbol{\pi}})$. The joint prior for the regression coefficients is

$$p(\boldsymbol{\beta}) = \prod_{i=1}^{k} \frac{\Gamma(a_i + b_i)}{\Gamma(a_i)\Gamma(b_i)} \widetilde{\pi}_i^{a_i - 1} (1 - \pi_i)^{b_i - 1} \left| J \right|_+ ,$$

where $\left| J \right|_+ = \left| \frac{dg^{-1}(\widetilde{\mathbf{X}}\boldsymbol{\beta})}{d\boldsymbol{\beta}'} \right|_+$ is the absolute value of the determinant of the Jacobian of the transformation. For the logit link function, $\left| J \right|_+ = \prod_i \widetilde{\pi}_i (1 - \widetilde{\pi}_i) \left| \widetilde{\mathbf{X}} \right|_+$. For the complementary log-log link function, $\left| J \right|_+ = \prod_i (1 - \widetilde{\pi}_i) \exp(\widetilde{\mathbf{x}}_i' \boldsymbol{\beta}) \left| \widetilde{\mathbf{X}} \right|_+$. Clearly, the priors for the regression coefficients depend on the link function.

Assuming prior independence of all unknown parameters, the joint prior distribution is given by $p(\boldsymbol{\beta}, \eta, \theta) = p(\boldsymbol{\beta}) p(\eta) p(\theta)$. The joint posterior distribution is

$$p(\boldsymbol{\beta}, \eta, \theta \mid \mathbf{y}^*) \propto L(y^* \mid \boldsymbol{\beta}, \eta, \theta)\, p(\boldsymbol{\beta}, \eta, \theta).$$

2.1.3 Mixed Effects Model

In this section, we describe the Bayesian GLMM that accommodates a misclassified response variable. Let y_j^* be the observed number of successes out of n_j Bernoulli trials with success probability $\eta\pi_j + (1 - \theta)(1 - \pi_j)$ for the j^{th} covariate pattern. Let $\mathbf{x}_j$ be a known $k \times 1$ vector of covariates. The total number of covariate combinations is equal to s, and these combinations define the strata, or clusters.

Agresti (2002) describes the GLMM as a two stage model. In the first stage, the observed responses follow a GLM, conditioned on the random effects. That is, $g(\pi_j) = \mathbf{x}_j'\boldsymbol{\beta} + \varepsilon_j$, where the ε_j's are random effects. Thus, conditioned on the random effects,

observations within a cluster may be assumed independent. At the second stage, the random effects are assumed independent, with $\varepsilon_j \sim N(0, \sigma^2)$.

For the mixed model, the likelihood function of the observed data is

$$\begin{aligned} L(\boldsymbol{\beta}, \eta, \theta, \boldsymbol{\varepsilon}, \sigma_\varepsilon^2 \mid \mathbf{y}^*) &= \prod_{j=1}^{s} f(y_j^* \mid \boldsymbol{\beta}, \eta, \theta, \varepsilon_i) f(\varepsilon_j \mid \sigma_\varepsilon^2) \\ &= \prod_{j=1}^{s} \binom{n_i}{y_j^*} \left[\eta\pi_j + (1-\theta)(1-\pi_j)\right]^{y_j^*} \left[\pi_j(1-\eta) + \theta(1-\pi_j)\right]^{n_j - y_j^*} \quad (2.2) \\ &\quad \times \prod_{j=1}^{s} (2\pi\sigma^2)^{-1/2} \exp\left[-\varepsilon_j^2/(2\sigma^2)\right], \end{aligned}$$

where $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_s)$ and $g(\pi_i) = \mathbf{x}_i'\boldsymbol{\beta} + \varepsilon_i$, conditional on the random effects.

To complete the Bayesian model, we must specify prior distributions for all unknowns in the model. As in Section 2.1.2, we assume independent beta priors for the sensitivity and specificity, $\eta \sim \text{Beta}(a_\eta, b_\eta)$ and $\theta \sim \text{Beta}(a_\theta, b_\theta)$. For the regression coefficients, we construct conditional means priors.

Introduction of a random effect is introduced into the model affects the induced prior for the regression coefficients. For the mixed model, we have $g(\tilde{\boldsymbol{\pi}}) = \tilde{\mathbf{X}}\boldsymbol{\beta} + \tilde{\boldsymbol{\varepsilon}}$. Using the transformation technique described in Section 2.1.1, we obtain a prior for $\boldsymbol{\beta}$ in the mixed model by the relationship

$$\begin{aligned} \boldsymbol{\beta} &= \tilde{\mathbf{X}}^{-1}\left[g(\tilde{\boldsymbol{\pi}}) - \tilde{\boldsymbol{\varepsilon}}\right] \\ &= \tilde{\mathbf{X}}^{-1} g(\tilde{\boldsymbol{\pi}}) - \tilde{\mathbf{X}}^{-1}\tilde{\boldsymbol{\varepsilon}}, \end{aligned}$$

assuming $\tilde{\varepsilon}_i \sim N(0, \sigma^2)$. Prior distributions for $\tilde{\pi}_i$ are elicited from expert opinion. We must also specify an appropriate prior distribution for the variance component σ^2.

Several non-informative prior distributions for the variance component have been suggested. The inverse-gamma prior on σ^2 is popular in the literature, but has proven

problematic in certain settings (Gelman 2006). Paulino et al. (2005) use the inverse-gamma prior for their analysis, but we found that this prior resulted in slow convergence. Gelman (2006) recommends the use of a uniform prior on the standard deviation of the random effect. Following this suggestion, we take $\sigma \sim \text{Uniform}(0.1, B)$, where B is an appropriate bound. This model is shown in Figure 4.

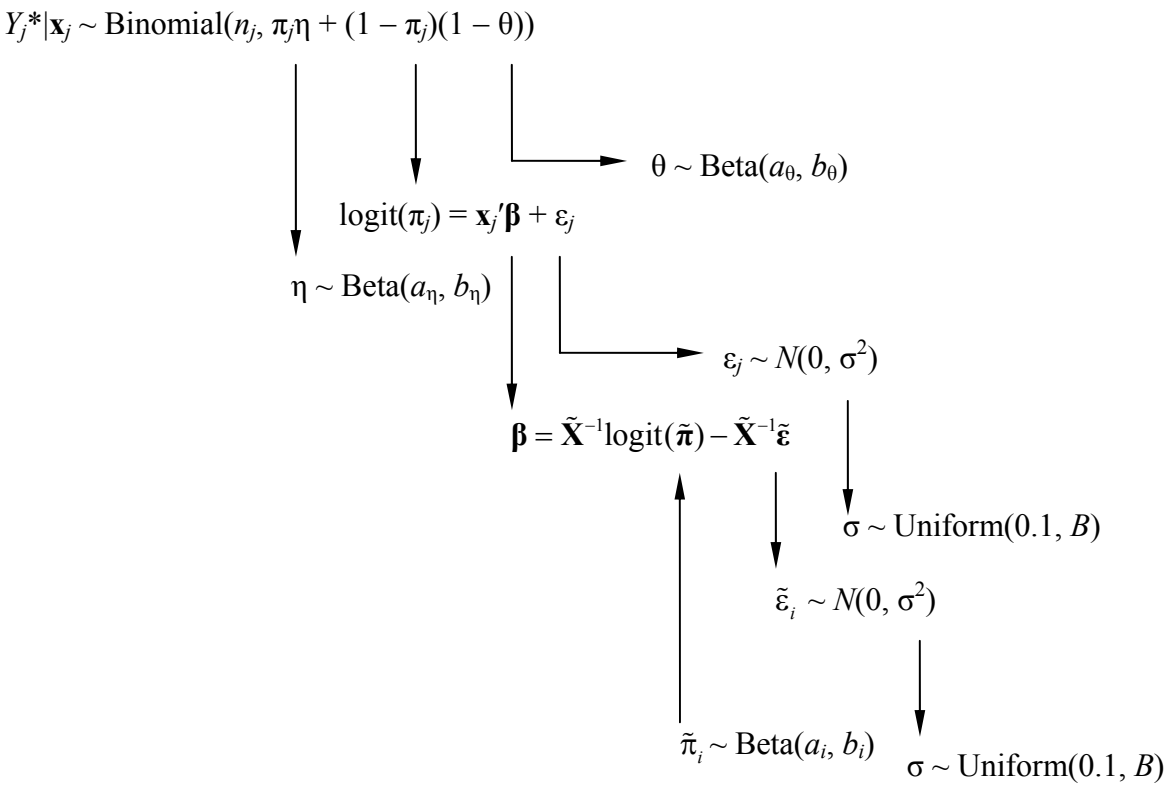

Figure 4. Summary of the Mixed Model for a Misclassified Response. The upward pointing arrow from $\tilde{\pi}_i$ indicates that the conditional means prior for $\boldsymbol{\beta}$ is induced from that elicited for $\tilde{\pi}_i$.

2.1.4 The Influence of the Random Effect Variance on the Induced Prior for $\boldsymbol{\beta}$

To illustrate the effect of the variance component on the induced prior for the regression coefficients, we consider the following simple example. Let

$$\widetilde{\mathbf{X}} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

We take $\widetilde{\pi}_1 \sim$ Beta(8, 10) and $\widetilde{\pi}_2 \sim$ Beta(3, 13) as prior distributions. For the fixed effects model, the prior for **β** is obtained by the transformation

$$\begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{-1} \text{logit} \begin{pmatrix} \widetilde{\pi}_0 \\ \widetilde{\pi}_1 \end{pmatrix}.$$

Under the mixed model, the prior for **β** is a result of the transformation

$$\begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{-1} \text{logit} \begin{pmatrix} \widetilde{\pi}_0 \\ \widetilde{\pi}_1 \end{pmatrix} - \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} \widetilde{\varepsilon}_1 \\ \widetilde{\varepsilon}_2 \end{pmatrix}.$$

We simulated 10,000 values from these priors, using upper bounds of $B = 5$ and $B = 10$ on the uniform prior. Histograms for the resulting prior distributions for the regression coefficients are shown in Figure 5.

The mixed model formulation shows a wider support in the induced prior, and this support becomes wider as the upper bound increases. A slight skewness is also present in the fixed model formulation, but is less apparent in the mixed model formulation. The prior distributions appear to become more symmetric as the upper bound increases.

For further illustration, we examine the CMP priors using $\widetilde{\mathbf{X}}$ and $p(\widetilde{\boldsymbol{\pi}})$ as given in McInturff et al. (2004). We simulate 10,000 values from the induced priors for the regression coefficients, using a sequence of values for the upper bound B with $0 < B \leq 10$. After each simulation, we calculate the interquartile range (IQR) of the resulting prior for each regression coefficient. For illustration, we present the results for only the parameter β_2, but the results were similar for the other coefficients. A plot of the interquartile range versus the corresponding value of B is shown in Figure 6. Not surprisingly, there is an increasing trend in IQR as B increases. Figure 6 also shows the IQR for the prior using

the fixed model (dotted line). This value coincides with the IQR for the mixed model with $B = 0$. Figure 7 shows the interquartile range as a function of the standard deviation, assuming that it is known. As expected, IQR increases with σ.

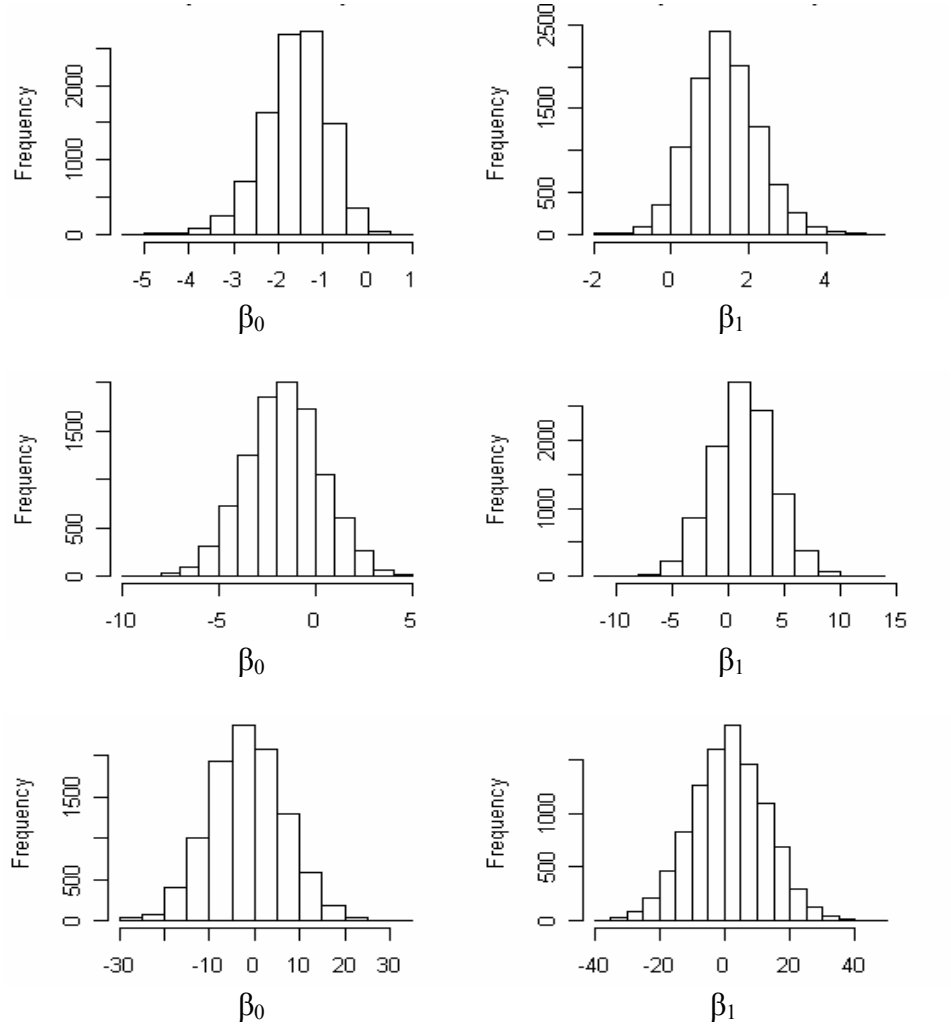

Figure 5. Induced Priors for the Regression Coefficients using the Fixed Model (top), the Mixed Model with $B = 5$ (middle), and the Mixed Model with $B = 10$ (bottom).

We see that adding random effects to the model has an implicit effect on the prior structure for the regression coefficients. This induced prior is completely specified by the priors on $\tilde{\pi}_i$ and σ, and is sensitive to the choice of an upper bound for the prior on the

variance component. A larger upper bound on the prior for σ results in greater variance in the prior for **β**.

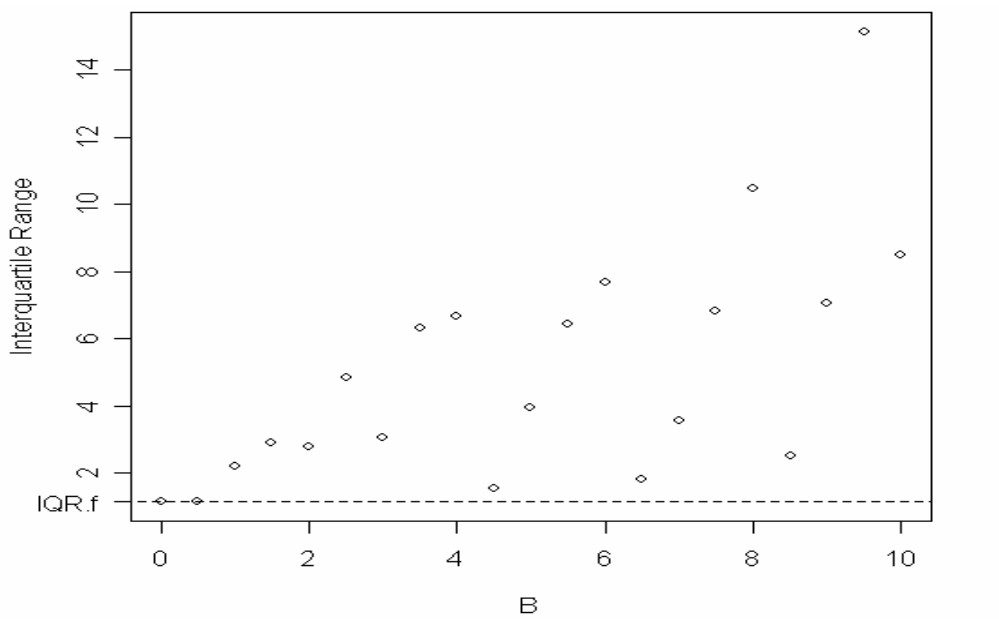

Figure 6. Interquartile Range for the Mixed Model Prior on β_2 for $\sigma \sim$ Uniform(0, *B*).

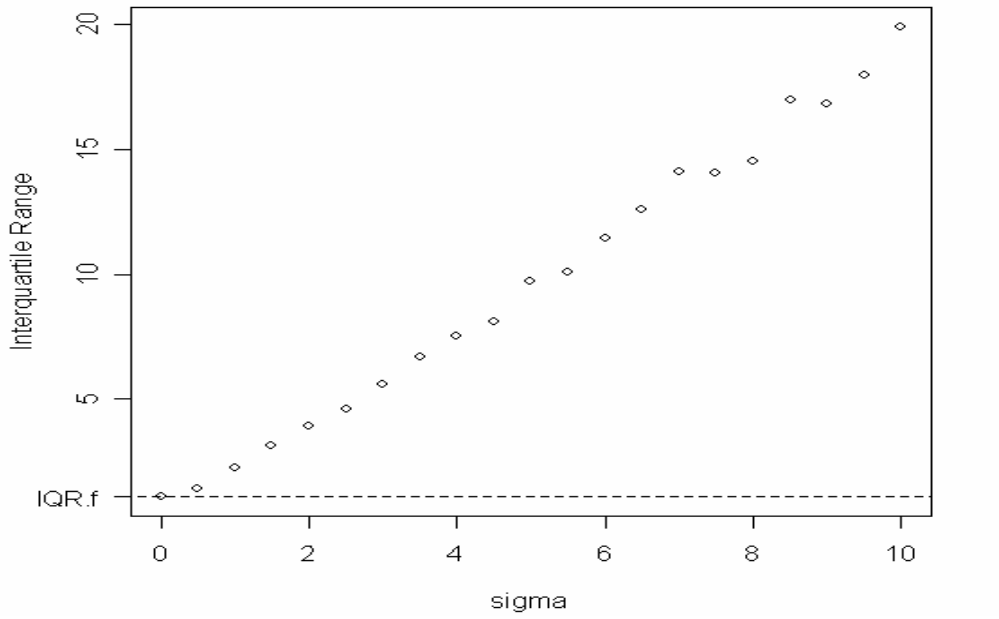

Figure 7. Interquartile Range for the Mixed Model Prior on β_2 for Given σ.

2.2 Model Comparison

Several methods have been developed to compare competing models in the Bayesian setting. McInturff et al. (2004) derive Bayes factors to compare models with different link functions. Other model comparison techniques include the Deviance Information Criterion (DIC), proposed by Spiegelhalter, Best, Carlin, and van der Linde (2002), and the method of Gelfand and Ghosh (1998).

2.2.1 Bayes Factors

Let M_1 and M_2 be two competing fixed effects models for fitting the data (as in Section 2.1.1), with common parameters $\boldsymbol{\gamma} \equiv (\boldsymbol{\beta}, \eta, \theta)$. Here $\boldsymbol{\beta} \in \mathbb{R}^p$, $\eta \in [0, 1]$, and $\theta \in [0, 1]$, so that $\boldsymbol{\gamma} \in G \equiv \mathbb{R}^p \times [0, 1] \times [0, 1]$. Denote by $p(\mathbf{y}^* \mid \gamma, M_i)$ the likelihood function based on M_i and let $p_i(\boldsymbol{\gamma})$ be the joint prior distribution for the model M_i. Recall that the prior for $\boldsymbol{\beta}$ depends on the link function, and hence, on the model. The Bayes factor (BF) is defined as

$$BF = \frac{\int p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_1) p_1(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}{\int p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma}) d\boldsymbol{\gamma}},$$

where all integrals are over G. Multiplying by one, we have

$$BF = \frac{\int p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_1) p_1(\boldsymbol{\gamma}) p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma}) / p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}{\int p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}.$$

Note that $\frac{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma})}{\int p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2) p_2(\boldsymbol{\gamma}) d\boldsymbol{\gamma}}$ is the posterior of $\boldsymbol{\gamma}$ under model M_2. Therefore, we may write the BF as

$$BF = \int \frac{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_1)\, p_1(\boldsymbol{\gamma})}{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2)\, p_2(\boldsymbol{\gamma})} p_2(\boldsymbol{\gamma} \mid \mathbf{y}^*) d\boldsymbol{\gamma} .$$

Recall that the prior for (η, θ) is invariant to the choice of model. Thus, we have

$$BF = \int \frac{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_1)\, p_1(\boldsymbol{\beta})}{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2)\, p_2(\boldsymbol{\beta})} p_2(\boldsymbol{\gamma} \mid \mathbf{y}^*) d\boldsymbol{\gamma} .$$

Now, we can write $BF = \int w(\boldsymbol{\gamma})\, p_2(\boldsymbol{\gamma} \mid \mathbf{y}^*)\, d\boldsymbol{\gamma}$, where

$$w(\boldsymbol{\gamma}) = \frac{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_1)\, p_1(\boldsymbol{\beta})}{p(\mathbf{y}^* \mid \boldsymbol{\gamma}, M_2)\, p_2(\boldsymbol{\beta})} .$$

Thus the BF may be written as the expectation of $w(\boldsymbol{\gamma})$ with respect to the posterior of $\boldsymbol{\gamma}$ under M_2.

As an example, let M_1 represent the model using the complementary log-log link function. Then for covariate combination $\tilde{\mathbf{x}}_i$, $\tilde{\pi}_i{}^* = 1 - \exp(-\exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta}))$, and the joint prior for the $\boldsymbol{\beta}$ under M_1 is

$$\begin{aligned} p_1(\boldsymbol{\beta}) &= c \prod_{i=1}^{k} (\tilde{\pi}_i{}^*)^{a_i - 1} (1 - \tilde{\pi}_i{}^*)^{b_i - 1} (1 - \tilde{\pi}_i{}^*) \exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta}) \left| \tilde{\mathbf{X}} \right|_+ \\ &= c \prod_{i=1}^{k} (\tilde{\pi}_i{}^*)^{a_i - 1} (1 - \tilde{\pi}_i{}^*)^{b_i} \exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta}) \left| \tilde{\mathbf{X}} \right|_+ \end{aligned}$$

where $c = \prod_{i=1}^{k} \frac{\Gamma(a_i + b_i)}{\Gamma(a_i)\Gamma(b_i)}$. Let M_2 represent the model using the logit link function. Then for covariate combination $\tilde{\mathbf{x}}_i$, $\tilde{\pi}_i = \exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta}) / (1 + \exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta}))$, and the prior for $\boldsymbol{\beta}$ under M_2 is

$$\begin{aligned} p_2(\boldsymbol{\beta}) &= c \prod_{i=1}^{k} (\tilde{\pi}_i)^{a_i - 1} (1 - \tilde{\pi}_i)^{b_i - 1} \tilde{\pi}_i (1 - \tilde{\pi}_i) \left| \tilde{\mathbf{X}} \right|_+ \\ &= c \prod_{i=1}^{k} (\tilde{\pi}_i)^{a_i} (1 - \tilde{\pi}_i)^{b_i} \left| \tilde{\mathbf{X}} \right|_+ . \end{aligned}$$

For these two models, we find that

$$w(\boldsymbol{\gamma}) = \frac{\prod_{j=1}^{n} (q_j^*)^{y_j^*} (1-q_j^*)^{1-y_i^*} \prod_{i=1}^{k} (\tilde{\pi}_i^*)^{a_i-1} (1-\tilde{\pi}_i^*)^{b_i} \exp(\tilde{\mathbf{x}}_i'\boldsymbol{\beta})}{\prod_{j=1}^{n} (q_j)^{y_j^*} (1-q_j)^{1-y_j^*} \prod_{i=1}^{k} (\tilde{\pi}_i)^{a_i} (1-\tilde{\pi}_i)^{b_i}}.$$

Using this derivation, computation of Bayes factors can be accomplished using MCMC methods as follows. First compute v_j as

$$\begin{aligned} v_j &= y_j^*[\log q_j^* - \log q_j] + (1-y_i^*)[\log(1-q_j^*) - \log(1-q_j)] \\ &= \log\left(\frac{q_j^*}{q_j}\right)^{y_j^*} + \log\left(\frac{1-q_j^*}{1-q_j}\right)^{1-y_j^*} \\ &= \log\left[\left(\frac{q_j^*}{q_j}\right)^{y_j^*}\left(\frac{1-q_j^*}{1-q_j}\right)^{1-y_j^*}\right]. \end{aligned}$$

Taking the sum, we have

$$\begin{aligned} w_1^* &= \sum_{j=1}^{n} v_j = \sum_{j=1}^{n} \log\left[\left(\frac{q_j^*}{q_j}\right)^{y_j^*}\left(\frac{1-q_j^*}{1-q_j}\right)^{1-y_j^*}\right] \\ &= \log\prod_{j=1}^{n}\left(\frac{q_j^*}{q_j}\right)^{y_j^*}\left(\frac{1-q_j^*}{1-q_j}\right)^{1-y_j^*}. \end{aligned}$$

Exponentiating gives

$$\exp\left(\sum_{j=1}^{n} v_j\right) = \prod_{j=1}^{n}\left(\frac{q_j^*}{q_j}\right)^{y_j^*}\left(\frac{1-q_j^*}{1-q_j}\right)^{1-y_j^*}.$$

Similarly, we compute

$$\begin{aligned} u_i &= (a_i-1)\log\tilde{\pi}_i^* - a_i\log\tilde{\pi}_i + b_i[\log(1-\tilde{\pi}_i^*) - \log(1-\tilde{\pi}_i)] + \tilde{\mathbf{x}}_i'\boldsymbol{\beta} \\ &= \log(\tilde{\pi}_i^*)^{a_i-1} - \log(\tilde{\pi}_i)^{a_i} + \log(1-\tilde{\pi}_i^*)^{b_i} - \log(1-\tilde{\pi}_i)^{b_i} + \tilde{\mathbf{x}}_i'\boldsymbol{\beta} \\ &= \log\left(\frac{(\tilde{\pi}_i^*)^{a_i-1}}{\tilde{\pi}_i^{a_i}}\right) + \log\left(\frac{1-\tilde{\pi}_i^*}{1-\tilde{\pi}_i}\right)^{b_i} + \tilde{\mathbf{x}}_i'\boldsymbol{\beta}. \end{aligned}$$

Taking the sum, we have

$$w_2^* = \sum_{i=1}^{k} u_i = \log \prod_{i=1}^{k} \left(\frac{(\tilde{\pi}_i^*)^{a_i - 1}}{\tilde{\pi}_i^{a_i}} \left(\frac{1 - \tilde{\pi}_i^*}{1 - \tilde{\pi}_i} \right)^{b_i} \right) + \sum_{i=1}^{k} \tilde{\mathbf{x}}_i' \boldsymbol{\beta}.$$

Exponentiating gives

$$\exp\left(\sum_{i=1}^{k} u_i \right) = \prod_{i=1}^{k} \frac{(\tilde{\pi}_i^*)^{a_i - 1}}{\tilde{\pi}_i^{a_i}} \frac{(1 - \tilde{\pi}_i^*)^{b_i}}{(1 - \tilde{\pi}_i)^{b_i}} \exp(\tilde{\mathbf{x}}_i' \boldsymbol{\beta}).$$

Finally, let $w = \exp(w_1^* + w_2^*)$. Then

$$\begin{aligned} w &= \exp\left(\sum_{j=1}^{n} v_j + \sum_{i=1}^{k} u_i \right) \\ &= \exp\left(\sum_{j=1}^{n} v_j \right) \exp\left(\sum_{i=1}^{k} u_i \right) \\ &= \frac{\prod_{j=1}^{n} (q_j^*)^{y_j^*} (1 - q_j^*)^{1 - y_j^*} \prod_{i=1}^{k} (\tilde{\pi}_i^*)^{a_i - 1} (1 - \tilde{\pi}_i^*)^{b_i} \exp(\tilde{\mathbf{x}}_i' \boldsymbol{\beta})}{\prod_{j=1}^{n} q_j^{y_j^*} (1 - q_j)^{1 - y_j^*} \prod_{i=1}^{k} \tilde{\pi}_i^{a_i} (1 - \tilde{\pi}_i)^{b_i}}. \end{aligned}$$

Note that w here is equal to $w(\boldsymbol{\gamma})$, so that $BF = \int_G w(\boldsymbol{\gamma}) p_2(\boldsymbol{\gamma} \mid \mathbf{y}^*) d\boldsymbol{\gamma}$. By sampling from the posterior under model M_2, the BF can be approximated by $\sum w(\boldsymbol{\gamma}) / MC$, where MC is the number of samples from the posterior.

An assumption made in this derivation is that both models under consideration contain the same number of unknown parameters. When the two opposing models contain a different number of parameters, as is the case for fixed effects versus random effects models, the derivation of Bayes factors becomes much more complicated. This difficulty motivates the use of an alternative model selection criterion, as discussed in the next section.

2.2.2 Deviance Information Criterion

Due to the complexity of the models under consideration, we avoid the computation of Bayes factors in favor of the Deviance Information Criterion (DIC), described by Spiegelhalter et al. (2002). The DIC provides a method for comparing the relative fit of several competing models by evaluating their predictive ability while accounting for model dimension. In what follows, we use notation from Congdon (2005).

Let $\boldsymbol{\gamma}$ be the vector of parameters for a model. The deviance is defined as –2 times the log likelihood, or $D(\boldsymbol{\gamma}) \equiv -2 \log p(\mathbf{y}^* \mid \boldsymbol{\gamma})$. Let $\overline{D(\boldsymbol{\gamma})}$ be the posterior mean of the sampled deviances. The larger this value is, the worse the model fits the data. Define $D(\bar{\boldsymbol{\gamma}}) \equiv -2 \log p(\mathbf{y}^* \mid \bar{\boldsymbol{\gamma}})$, where $\bar{\boldsymbol{\gamma}}$ is the posterior mean of $\boldsymbol{\gamma}$. The complexity of a model is measured by the effective total number of parameters, p_D, which is generally less than the nominal number of parameters (Congdon 2005). This total is calculated as the difference between the posterior mean deviance and the deviance evaluated at $\bar{\boldsymbol{\gamma}}$. That is, $p_D = \overline{D(\boldsymbol{\gamma})} - D(\bar{\boldsymbol{\gamma}})$. Then, the DIC is defined as $DIC = \overline{D(\boldsymbol{\gamma})} + p_D$. The model with the smallest DIC should best predict a replicate dataset of the same structure as the one currently observed. Since $\overline{D(\boldsymbol{\gamma})}$ will decrease as the number of parameters increases, the p_D term compensates by penalizing excess complexity.

For our case, we have $Y_j^* \sim$ Binomial(n_j, p_j), where $p_j = \eta\pi_j + (1-\theta)(1-\pi_j)$. The deviance is

$$
\begin{aligned}
D(p_j) &= -2\log f(y_j^* \mid p_j) \\
&= -2\log\left[\binom{n_j}{y_j^*} p_j^{y_j^*}(1-p_j)^{n_j-y_j^*}\right] \\
&= -2\log\binom{n_j}{y_j^*} - 2y_j^*\log p_j - 2(n_j - y_j^*)\log(1-p_j).
\end{aligned}
$$

DIC may be calculated after an MCMC run by taking the sample mean of the simulated values of $D(\boldsymbol{\gamma})$, minus the estimate of deviance obtained by substituting in the sample means of the simulated values of $\boldsymbol{\gamma}$. It is not strictly necessary to use the mean as an estimate of $\boldsymbol{\gamma}$. The median or mode may also be used, and would be preferred if the posterior distribution were highly skewed.

A DIC tool is available in WinBUGS, which monitors deviance and computes the DIC. This tool uses the posterior mean as the point estimate of $\boldsymbol{\gamma}$. Again, if the posterior distribution is highly skewed, the median may be a more appropriate point estimate. In this situation, the DIC can be computed externally, such as in the package R. Example code for computing DIC in R is presented in Appendix A.

When comparing models j and k, the model with a smaller value of DIC is preferred. A suggested rule of thumb (Congdon 2005) is that

$$\Delta\text{DIC}_{jk} = \text{DIC}_j - \text{DIC}_k > 4$$

should count as a significant difference.

2.2.3 Gelfand and Ghosh Criterion

We now discuss the decision theoretic approach to model selection of Gelfand and Ghosh (1998). Their method is based on the posterior predictive loss. Let $\mathbf{y}_{obs}^*$ be

the vector of observed data. Define $\mathbf{y}^*_{pred}$ as the predicted data obtained from the posterior predictive distribution,

$$p(\mathbf{y}^*_{pred} \mid \mathbf{y}^*_{obs}) = \int p(\mathbf{y}^*_{pred} \mid \boldsymbol{\gamma})\, p(\boldsymbol{\gamma} \mid \mathbf{y}^*_{obs}) d\boldsymbol{\gamma}\,, \tag{2.3}$$

where $p(\mathbf{y}^*_{pred} \mid \boldsymbol{\gamma})$ is likelihood function of the data, evaluated at $\mathbf{y}^*_{pred}$, and $p(\boldsymbol{\gamma} \mid \mathbf{y}^*_{obs})$ denotes the posterior distribution of $\boldsymbol{\gamma}$ given the observed data. We must define a loss function that measures the discrepancy between $\mathbf{y}^*_{obs}$ and $\mathbf{y}^*_{pred}$. Ghosh and Norris (2005) suggest the Mean Square Predictive Error (MSPE) on the log scale,

$$MSPE = \frac{2}{s(s+1)} \sum_{j=1}^{s} \left[\log(y^*_{i,pred} + 0.5) - \log(y^*_i + 0.5) \right]^2 .$$

The stabilizing factor of 0.5 is to avoid log(0) when the observed counts are zeros. The Gelfand and Ghosh (1998) criteria is based on

$$GGC = E\left[MSPE \mid \mathbf{y}^*_{obs} \right],$$

where the expectation is taken with respect to the predictive distribution function defined in (2.3). Gelfand and Ghosh (1998) show that the GGC may be interpreted as the sum of a goodness-of-fit measure and a penalty term. We select the model with the smallest GGC.

2.3 Examples Using Simulated Data

In this section, we describe an application of the methods in Section 2.2 to simulated data sets. We illustrate the efficacy of a procedure for a known underlying model. Section 2.3.1 describes the simulation for a data set that contains only fixed effects. Section 2.3.2 discusses the application for data generated with both fixed and random effects.

2.3.1 Data with Fixed Effects

Data were generated for six binary explanatory variables, using estimates of the sensitivity, specificity, and regression parameters given in McInturff et al. (2004). Thus, $\boldsymbol{\beta} = (-1.27, -1.75, -1.32, 0.86, 0.37, 1.29)'$, $\eta = 0.992$, and $\theta = 0.901$. The responses y_j^* were generated from Binomial(n_j, p_j), $j = 1, \ldots, 17$, where n_j is the number of individuals in the j^{th} stratum, and $p_j = P(Y_j^* = 1|\mathbf{x}_j) = \eta\pi_j + (1 - \theta)(1 - \pi_j)$. We use the logit link function so that

$$\pi_j = \frac{\exp(\mathbf{x}_j'\boldsymbol{\beta})}{1+\exp(\mathbf{x}_j'\boldsymbol{\beta})}.$$

We refer to this simulated data as Data A.

The prior distributions for all parameters are the same as those used by McInturff et al. (2004). For the mixed model (2.2), the CMP prior for the regression coefficients is modified as discussed in Section 2.2 For the variance component, we use a uniform prior on the standard deviation, $\sigma \sim$ Uniform(0.1, 5). In the support of this prior, we bound the standard deviation away from zero since very small values are unlikely. For comparison of the two competing models, we compute the DIC.

For the fixed model (2.1), we used two independent chains, each of 10,000 iterations after a burn-in of 5,000. For the mixed model, we used the same sample size, and found that the MCMC chains exhibited a large amount of autocorrelation between successive draws from the posterior, especially for the precision of the random effects. Autocorrelation within chains is an indication of slow mixing and usually slow convergence. Several remedies exist for autocorrelation, among them over-relaxation, thinning the chain, and reparameterization of the model.

The process of over-relaxation generates multiple samples at each update of the simulation. The sample that is negatively correlated with the current value is retained. When a chain is thinned, every k^{th} iteration is stored, while all others are discarded. In WinBUGS, thinning can be performed while the simulation is running or after the updates have been completed.

Gelfand, Sahu, and Carlin (2005a) propose hierarchical centering as a reparameterization of the model that can improve mixing and result in better correlation properties. Let $\mu_j = \mathbf{x}'_j\boldsymbol{\beta}$ and let $b_i = \mu_j + \varepsilon_j$. Then the reparameterized mixed model is:

$$\begin{aligned} y_j^* &\sim \text{Binomial}(n_j, \eta\pi_j + (1-\theta)(1-\pi_j)), \\ \text{logit}(\pi_j) &= b_j, \\ b_j &\sim \text{N}(\mu_j, \sigma^2). \end{aligned} \tag{2.4}$$

Note that the centering here is performed on the parameters rather than on the covariates. Figure 8 shows a summary of the hierarchically centered model. Compare this to the model shown in Figure 4.

Reparameterization by sweeping is another solution that has been proposed to improve MCMC convergence. As an illustration of this method, consider the one-way analysis of variance model

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij},\ i = 1, \ldots, m; j = 1, \ldots, n,$$

where the ε_{ij}'sare independent N(0, δ_0), and the priors are $\mu \sim$ N(0, δ_1) and $\alpha_i \sim$ N(0, δ_2). This model is poorly identified in that the parameters μ and α_i are not uniquely determined. This lack of identifiability leads to slow convergence. Vines, Gilks, and Wild (1996) replace α_i by $\tilde{\alpha}_i = \alpha_i - \bar{\alpha}$, with a corresponding change in the prior. This strategy improves mixing, but results in a more complex model structure.

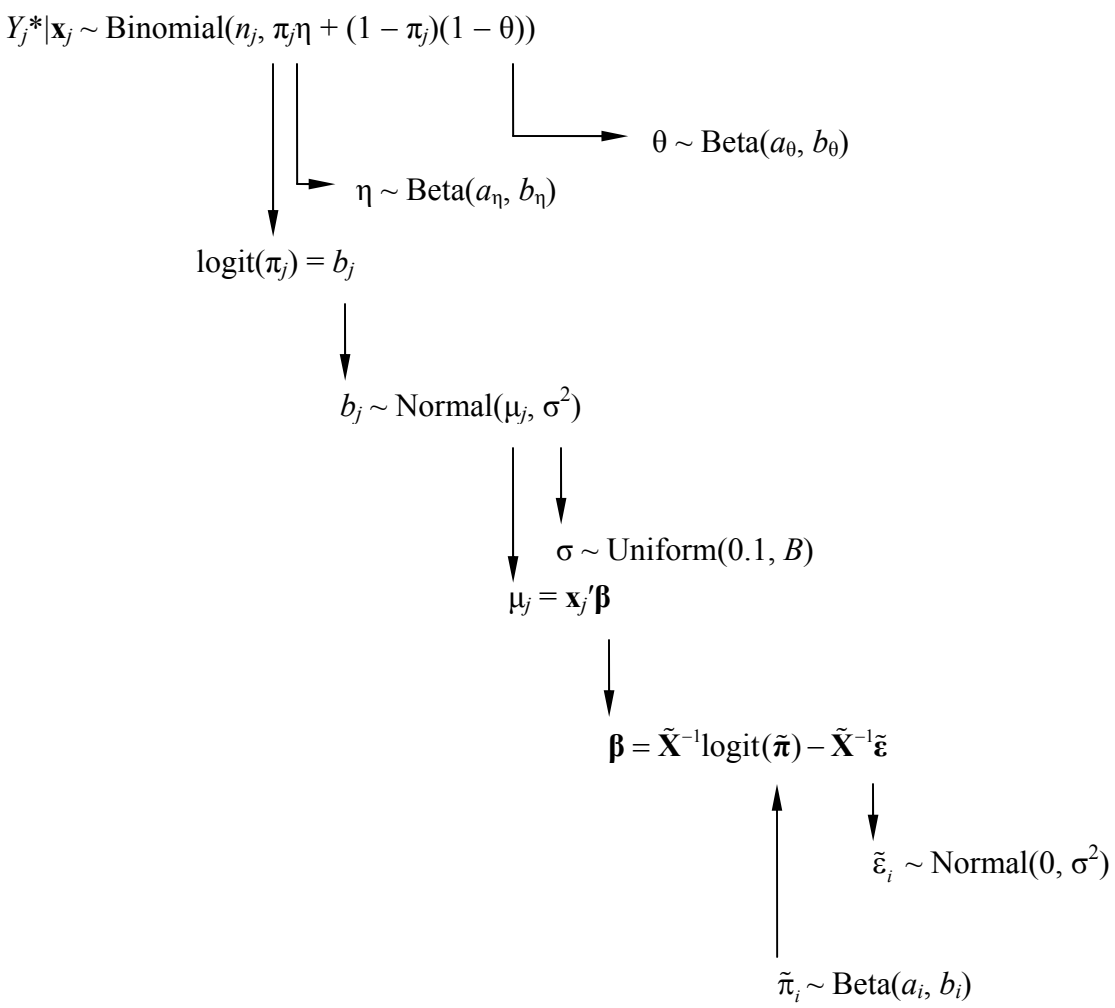

Figure 8. Summary of Hierarchically Centered Mixed Model. The upward pointing arrow indicates that the prior for $\boldsymbol{\beta}$ is induced from that elicited on $\tilde{\pi}_i$

To illustrate the relative merits of thinning and hierarchical centering for our case, we ran the mixed model (2.2) for Data A. We show autocorrelation plots for the parameter β_2, which demonstrated the most dramatic differences between the methods. Figure 9 shows the autocorrelation plot using neither hierarchical centering nor thinning. Figure 10 shows the autocorrelation plot using only hierarchical centering, and Figure 11 shows the autocorrelation plot using only thinning, where we retained every 30^{th} update.

Each of these plots is based on 20,000 usable iterations from two chains after a burn-in of 5,000.

The method of hierarchical centering did not improve the autocorrelation for the parameter β_2, and in fact, appears to have worsened the problem. For our study, over-relaxation slowed down the simulation and resulted in only marginal improvement in the autocorrelation. Hierarchical centering reduced the time per update, but was not effective in alleviating autocorrelation. Thinning the chain yielded the most improvement in the correlation. We therefore decide to run the analysis using only thinning.

The results of our analysis for Data A are shown in Table 3. In this case, the fixed effects model (2.1) has a smaller DIC, indicating a superior fit. After thinning, the parameter with the most noticeable autocorrelation was the random effects precision, ρ. This autocorrelation plot is shown in Figure 12.

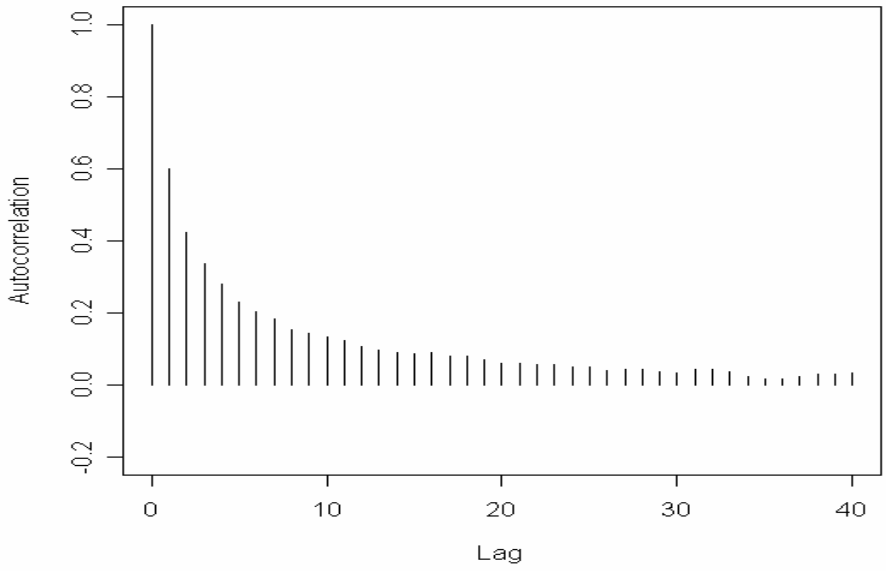

Figure 9. Autocorrelation Plot for β_2 using neither Hierarchical Centering nor Thinning.

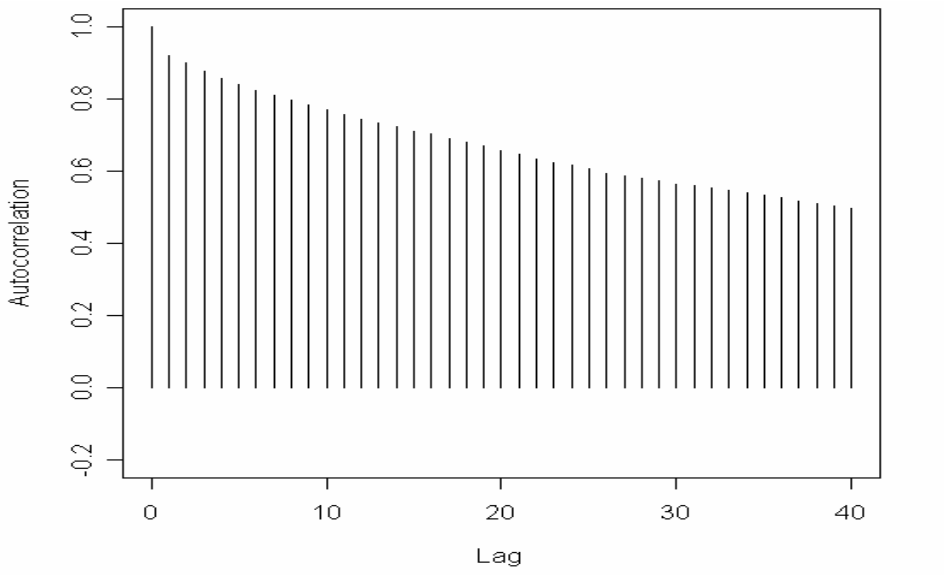

Figure 10. Autocorrelation Plot for β_2 using only Hierarchical Centering.

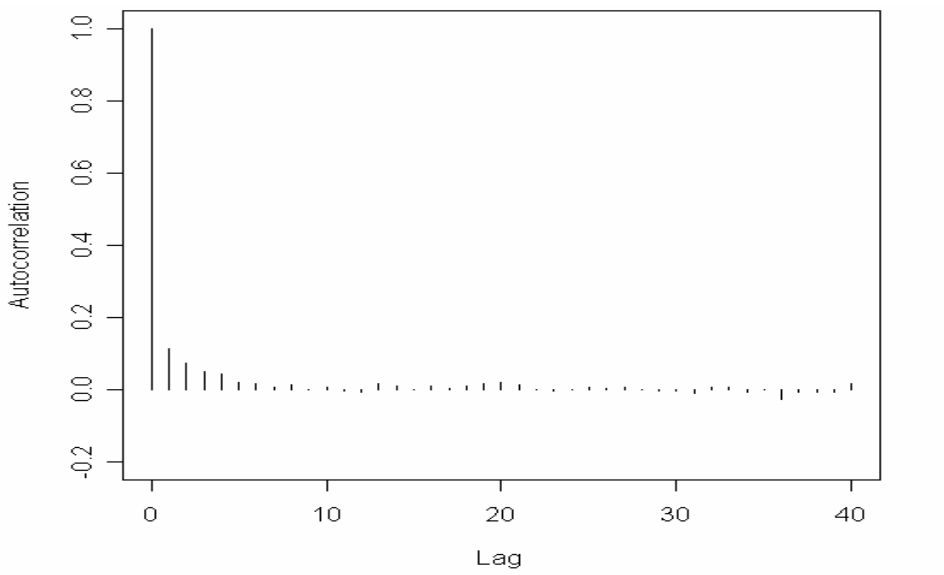

Figure 11. Autocorrelation Plot for β_2 using only Thinning.

2.3.2 Data with Fixed and Random Effects

To compare the fixed to mixed effects models when random effects are actually present, data were generated as in Section 2.4.1, with the exception that

$$\pi_j = \frac{\exp(\mathbf{x}_j'\boldsymbol{\beta} + \varepsilon_j)}{1 + \exp(\mathbf{x}_j'\boldsymbol{\beta} + \varepsilon_j)},$$

where $\varepsilon_j \sim$ i.i.d. N(0, ρ), and ρ is the precision. We refer to this data as Data B.

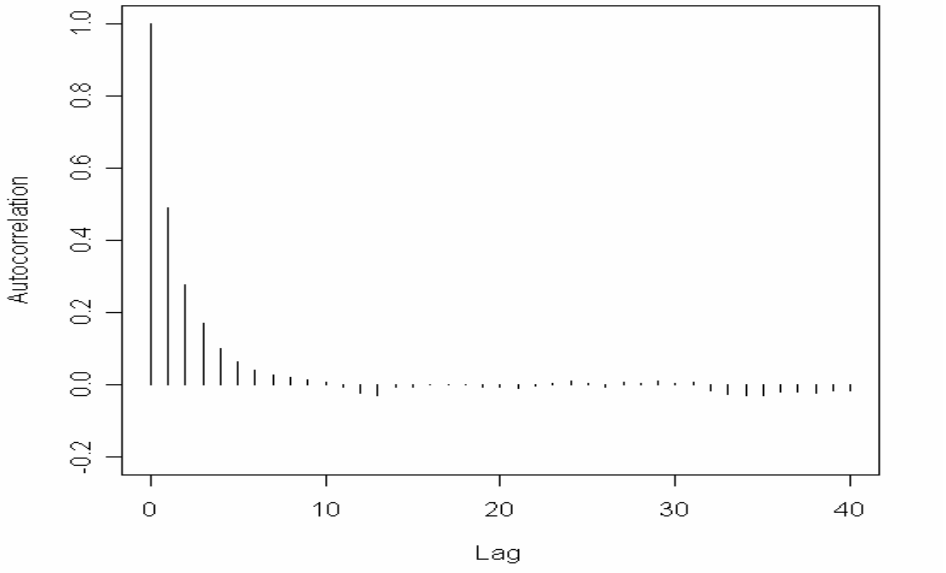

Figure 12. Autocorrelation Plot of ρ for Data A under the Mixed Model (2.2).

Table 3. Posterior Means (Standard Deviations) for Data A.

Parameter	Fixed Effects (2.1)	Mixed Effects (2.2)
β_1	−1.945 (0.527)	−2.236 (0.775)
β_2	−2.452 (0.483)	−2.616 (0.636)
β_3	−1.489 (0.465)	−1.583 (0.616)
β_4	1.359 (0.426)	1.552 (0.598)
β_5	1.061 (0.490)	1.109 (0.634)
β_6	1.714 (0.447)	1.909 (0.619)
η	0.990 (0.010)	0.990 (0.010)
θ	0.896 (0.040)	0.888 (0.043)
ρ	— (—)	21.030 (22.610)
DIC	60.801	62.571

We generated three mixed effects data sets with different values of ρ. These values were chosen by examining the posterior distribution of logit(π_i) under the fixed effects model using the data described in Section 2.3.1. Figure 13 shows the posteriors

for logit(π_2), logit(π_4), logit(π_7) and logit(π_{11}), which are representative of the posteriors with the least dispersion. Based on these plots, the standard deviation for the random effect should be no larger than one third. In other words, the precision of the random effect should be no smaller than 9. We generate data with random effects precisions 9, 12 and 15.

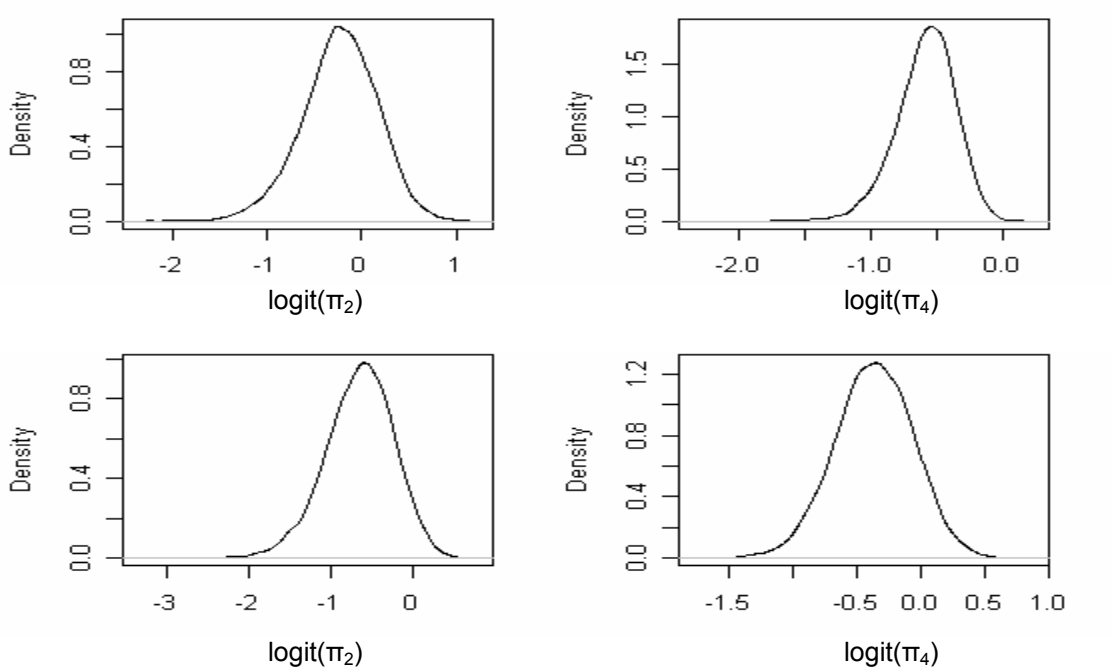

Figure 13. Plots of logit(π) for the Fixed Effects Data.

Results are shown in Table 4. More thinning was required for some of the data sets in order to achieve convergence. Figures 14 – 16 show the autocorrelation plots for the precision, which was typical of the worst in each case.

For each data set, the DIC suggests that the fixed effects model provides a slight gain over the mixed model. For the data with the random effects precision of 9, the average deviance under the mixed model is 55.645 with an effective number of parameters of 6.479. Under the fixed effects model, the average deviance is 57.539 and

the effective number of parameters is 3.821. So, although the average deviance worsens for the fixed effects model, the DIC favors this model because of its reduced complexity.

Table 4. Posterior Means (Standard Deviations) for Data B. "Fixed" refers to the fixed effects model (2.1). "Mixed" refers to the mixed model (2.2)

		$\rho = 9$		$\rho = 12$		$\rho = 15$	
	Truth	Fixed	Mixed	Fixed	Mixed	Fixed	Mixed
β_1	–1.27	–2.287	–2.868	–1.887	–2.475	–1.156	–1.259
		(0.562)	(1.156)	(0.604)	(1.411)	(0.470)	(0.652)
β_2	–1.75	–3.445	–3.842	–2.251	–2.758	–1.816	–2.040
		(0.578)	(1.041)	(0.637)	(1.411)	(0.402)	(0.563)
β_3	–1.32	–1.738	–2.140	–1.213	–1.424	–1.413	–1.326
		(0.512)	(0.970)	(0.474)	(0.930)	(0.459)	(0.596)
β_4	0.86	1.481	1.937	1.153	1.408	0.963	0.926
		(0.473)	(0.939)	(0.446)	(0.864)	(0.378)	(0.526)
β_5	0.37	1.684	1.958	1.174	1.295	0.421	0.435
		(0.541)	(0.943)	(0.522)	(1.011)	(0.439)	(0.598)
β_6	1.29	2.309	2.857	1.210	1.525	1.552	1.627
		(0.523)	(1.087)	(0.473)	(0.947)	(0.388)	(0.526)
η	0.992	0.991	0.991	0.989	0.989	0.990	0.990
		(0.009)	(0.009)	(0.010)	(0.010)	(0.010)	(0.010)
θ	0.901	0.893	0.886	0.830	0.809	0.851	0.846
		(0.031)	(0.033)	(0.052)	(0.056)	(0.058)	(0.059)
ρ	—	—	12.798	—	16.841	—	12.700
		(—)	(19.070)	(—)	(21.313)	(—)	(16.926)
DIC	—	61.361	62.124	66.095	66.219	66.664	68.239

To examine the sensitivity of the results to the upper bound of the uniform prior for the variance component, we repeated the analysis using a sequence of values for *B* and monitored posterior quantiles of the precision. The posterior median, 2.5th percentile, and 97.5th percentile of ρ were monitored and are shown in Figure 17 for Data B ($\rho = 9$). These quantiles appear very stable, but note the large scale on the vertical axis. It may also be useful to check the robustness of the results to the lower bound on the uniform prior, which we fixed at 0.1.

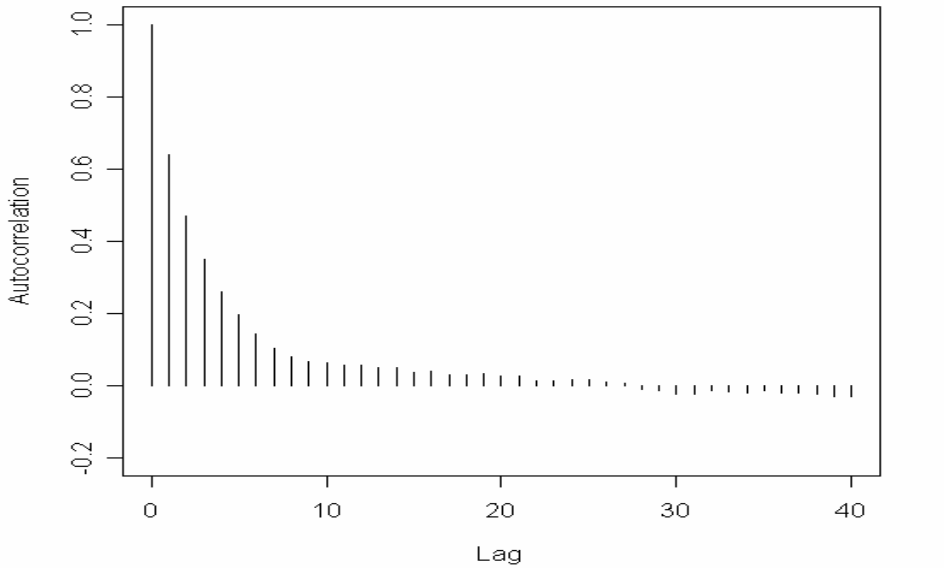

Figure 14. Autocorrelation Plot for the Precision; Data B (ρ = 9) under the Mixed Model.

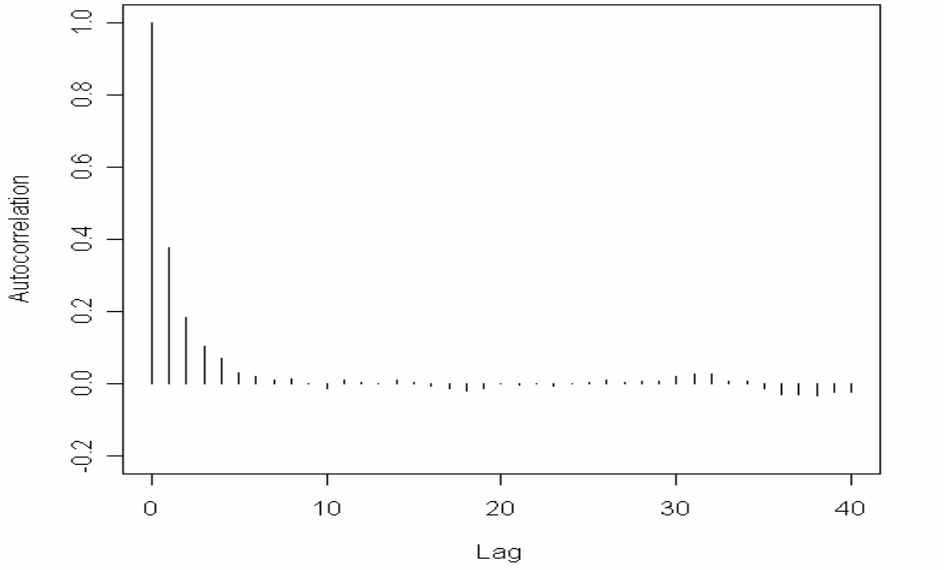

Figure 15. Autocorrelation Plot for the Precision, Data B (ρ = 12) under the Mixed Model.

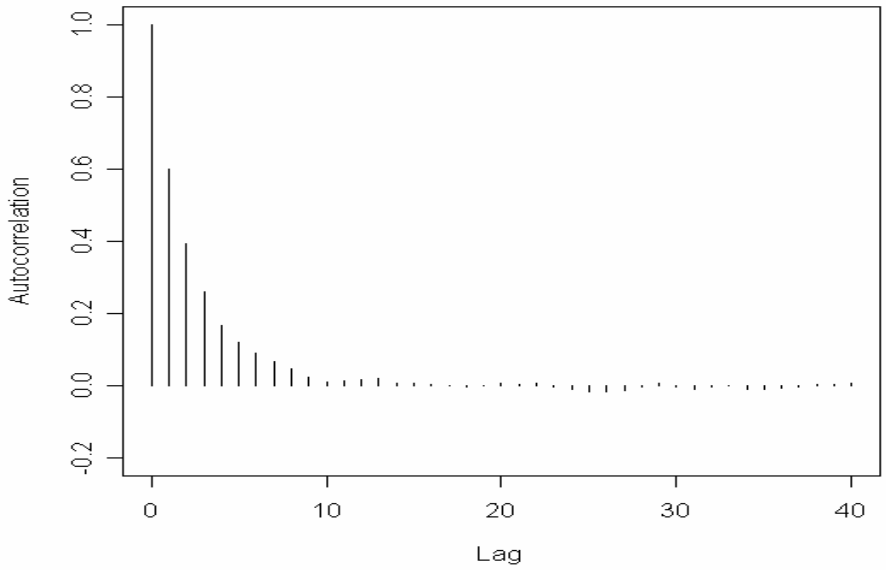

Figure 16. Autocorrelation Plot for the Precision, Data B ($\rho = 15$) under the Mixed Model.

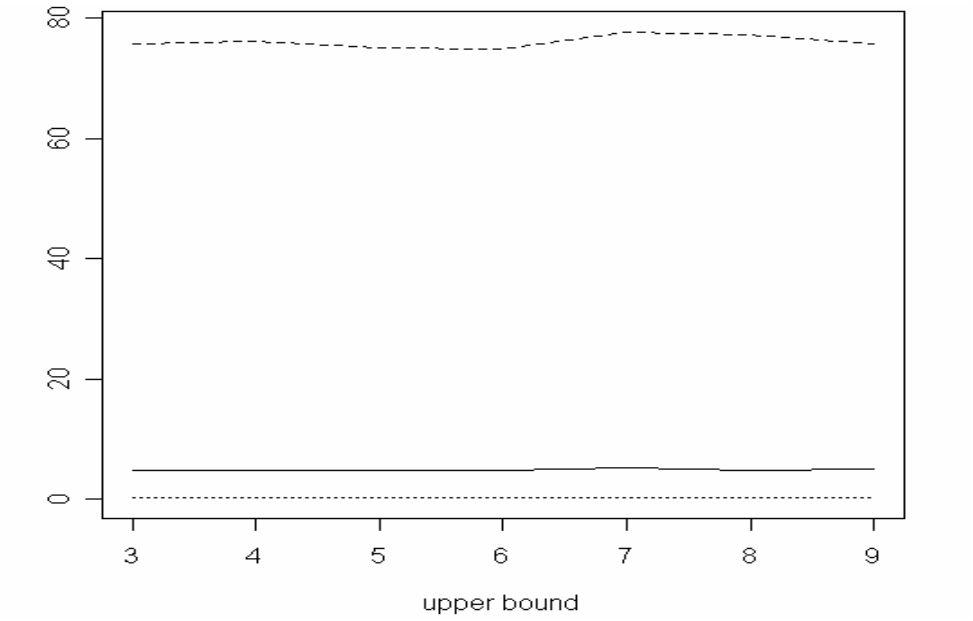

Figure 17. Posterior 2.5th percentile (dotted line), median (solid line), and 97.5th percentile (dashed line) of ρ for various values of B in $\sigma \sim$ Uniform[0.1, B] for Data B ($\rho = 9$).

2.4 Discussion

In this chapter, we have investigated the generalized linear mixed model (GLMM) for misclassified responses. We elaborated on the conditional means priors for regression coefficients and modified this technique for the mixed model setting. We found that these priors can be sensitive to the prior that is specified for the variance component. We demonstrated the adjustment that is made to the induced prior on $\boldsymbol{\beta}$ by changing the value of the upper bound on the prior for σ. Additionally, we should assess the effect of changes to the lower bound, which we fixed at 0.1.

We made use of the technique of thinning the chain in order to reduce autocorrelation and improve convergence for the mixed model. Other methods of reducing autocorrelation such as over-relaxation and hierarchical centering were not effective for our case.

Our methods were demonstrated through simulated data, and the DIC was calculated as a measure of relative fit. The DIC favored the less complex fixed effects model to the mixed model. Other model selection criteria such as the method of Gelfand and Ghosh (1998) may prove useful in this situation.

CHAPTER THREE

Bayesian Model for Misclassified Response with Covariate Measurement Error

Suppose that interest lies in examining the relationship between a binary response variable and various explanatory variables. In many regression applications, one or more explanatory variables may be measured imperfectly. When the response variable is also measured with error additional complications arise. If one proceeds as usual, making no adjustments to correct for these sources of error, biased parameter estimates and potentially misleading inference result. In this chapter, we consider a fully Bayesian analysis that affords such adjustments, accounting for the two sources of error and correcting estimates of the regression parameters.

The consequences of imperfect measurements have received much attention in the literature. The term "misclassification" typically refers to categorical variables that are measured with error, while "measurement error" refers to continuous variables. The problem of an imperfectly measured covariate is commonly known as the "error in variables" problem. Ignoring measurement error in a covariate leads to biased parameter estimates and obscures the relationship between the response and the covariates.

Response misclassification can also lead to large biases in parameter estimates, and adjustments must be made. For a dichotomous response in which the misclassification rates do not depend on the covariates, Neuhaus (1999) shows that estimates of the regression coefficient are biased toward zero. McInturff, Johnson, Cowling, and Gardner (2004) present a Bayesian analysis for a misclassified response. They use expert opinion to construct informative prior distributions for the regression

coefficients. Priors for the misclassification parameters are also taken to be relatively informative.

Sposto, Preston, Shimizu, and Mabuchi (1992) analyze data from the Life Span Study, which is conducted by the Radiation Effects Research Foundation (RERF). The study consists of a large cohort of individuals who survived the atomic bombings of Hiroshima and Nagasaki. One purpose of the study was to gauge the effect of radiation dose on cancer mortality. The cause of death was recorded from death certificates, which have been shown to be fallible. For a large subset of the subjects, autopsy information was also available to determine the true cause of death. This autopsy data was then used to estimate the probabilities of misclassification.

Roy, Banerjee, and Maiti (2005) extend the research of an analysis involving misclassified response by including the scenario in which one or more covariates are measured imperfectly. They develop a maximum likelihood approach to find corrected estimates of the regression coefficients. Roy et al. (2005) demonstrate their method using the Life Span Study. In addition to the misclassification for cause of death, radiation dose is taken to be measured with error. Radiation exposure for each individual was estimated by DS86 dosimetry, in which interviews and other efforts were made to determine the location and available shielding when the bombs detonated. This information was then used to estimate individual radiation exposure. Two people exposed to the same amount of radiation, however, may absorb different amounts due to biological differences. This difference is taken to be the source of measurement error.

In this chapter, we expand the work of Roy, et al. (2005) by proposing a Bayesian analysis that accounts for a misclassified binary response and a mismeasured continuous

covariate. The Bayesian approach has the advantage of not having to assume any parameters are known and allows for an operational way to combine results from previous studies and expert opinions.

The remainder of the chapter is organized as follows. In Section 1, we give an overview of methods that are available in the literature for handling measurement error. In Section 2, we develop a Bayesian regression model that corrects the bias in the regression parameter estimates by accounting for both misclassification in the response and measurement error in the covariate. In Section 3, we illustrate our method using the Life Span Study as an example. In Section 4, we discuss a simulation study. Finally, we make some concluding remarks in Section 5.

3.1 Overview of Measurement Error

3.1.1 Measurement Error Models

Suppose Y is a response variable of interest. Let X be a continuous explanatory variable that is subject to measurement error, and let X^* be a surrogate for X. For example, Y might be cause of death, with $Y = 1$ if the cause of death is cancer, and $Y = 0$ otherwise. X might be a dose of radiation, for which only an estimate, X^* is available. We may have additional covariates, Z, that are considered to be measured without error. Our interest lies in examining the relationship between the response Y and predictor variables X and Z.

The measurement error model is a model that relates the unobserved true variable X to its surrogate measurement, X^*. Two broad classes of measurement error models are available: the classical model and the Berkson model. The classical model is appropriate

when repeated measurements of a covariate vary, but the average of many such measurements approaches the true value of the covariate. For classical error, the additive model is

$$X^* = X + U,$$

where U is independent of X, with $E(U|X, Z) = 0$. The error structure of U may be either homoscedastic or heteroscedastic. A multiplicative model may also be considered, in which $X^* = XU$, where $E(U|X, Z) = 1$. For the classical measurement error model, one defines the conditional distribution of the surrogate given the unobserved covariate.

The Berkson model (Berkson 1950) defines the conditional distribution of X given the surrogate. Berkson errors arise, for example, when individual values of a covariate are replaced by the average value for several subjects (Armstrong 1998). For the additive Berkson model,

$$X = X^* + U,$$

where U is independent of X^* and $E(U|X^*, Z) = 0$. Again, U may have homoscedastic or heteroscedastic variance. In the Life Span Study that we consider, dose exposure is averaged over subjects at each level, making the Berkson model the most appropriate one.

Carroll, Ruppert, Stefanski, and Crainiceanu (2006) show that, for linear regression, the effect of classical measurement error in a continuous explanatory variable is to bias the estimate of the associated regression coefficient toward zero, a result known as attenuation. Berkson errors, however, result in no bias for the linear regression case. For the case of logistic regression, attenuation is present even for Berkson errors (Reeves, Cox, Darby, and Whitley 1998; Kim, Yasui, and Burstyn 2006).

Several methods for correcting this bias have been proposed in the literature. An overview of some of these methods is presented below. For a more in depth treatment, see Carroll et al. (2006).

3.1.2 Regression Calibration

In regression calibration the unknown X is replaced by the regression of X on (X*, Z). Then the desired analysis can be carried out as if X were observed. The regression of X on (X*, Z) is called the "calibration function." The difficulty in regression calibration is that the calibration function may be difficult to estimate. It is necessary to have either (1) validation data, in which the true value of X is available in addition to (X*, Y, Z) for some study subjects, (2) replicate data, in which many measurements, X*, of X are available for some subjects, or (3) an "instrumental variable," T, which is unbiased for X, and is measured in addition to X*.

Several methods have been suggested for estimating the calibration function. When validation data is available, we simply regress X on the other covariates, using standard regression techniques. Linear regression is typical, but not required (Carroll et al. 2006). When an instrumental variable is available, Rosner, Spiegelman, and Willett (1990) estimate the calibration function by regressing T on (X*, Z). If replicates are available, Carroll et al. (2006) recommend a "best linear approximation" to the calibration function.

Once the calibration function has been estimated, we replace the unobserved X by its estimate from the calibration model, and the desired analysis is run. Finally, the resulting standard errors must be adjusted to account for the estimation. Methods such as the bootstrap or sandwich methods may be used for this purpose. While regression

calibration is useful for generalized linear models it performs poorly for highly nonlinear models (Carroll et al. 2006).

3.1.3 Simulation Extrapolation (SIMEX)

Simulation extrapolation (SIMEX) was proposed by Cook and Stefanski (1994). The basic idea is to simulate data with increasing amounts of measurement error, establishing a trend of the bias that is introduced as a result of measurement error. Then this trend is used to extrapolate back to the case of no measurement error. As an example, consider the simple linear regression $Y = \beta_0 + \beta_1 X + \varepsilon$, with classical additive measurement error $X^* = X + U$, where U is independent of (Y, X) with mean zero and variance σ_u^2. The ordinary least squares analysis that ignores measurement error produces an estimate not of β_1 but instead of $\beta_1 \sigma_x^2 / \left(\sigma_x^2 + \sigma_u^2\right)$. For the SIMEX method, we generate several "contaminated" data sets, each with successively larger measurement error variances. For each of these contaminated data sets, we obtain regression estimates. We repeat this simulation a large number of times, and calculate the average value of the estimates for each level of contamination. These averages are regressed against the amount of measurement error. Finally, we use this regression model to extrapolate back to the case of no measurement error.

3.1.4 Maximum Likelihood Methods

For the maximum likelihood technique, stronger distributional assumptions will be made. We must specify a parametric model for each component of the data. This typically includes a response model, a measurement error model, and possibly an exposure model. The response model relates the response to the actual explanatory

variables, and is the model that would be used if X were measured perfectly. This model may only be known up to a parameter vector, γ_R, so that we specify $f(y \mid x, z, \gamma_R)$. The measurement error model, as discussed in Section 3.1.1, may also be dependent upon a parameter vector, γ_M, so that for classical measurement error, we specify $f(x^* \mid x, \gamma_M)$. For Berkson errors we take $f(x \mid x^*, \gamma_M)$. If the measurement error model contains classical components, it is necessary to specify an exposure model, which is the distribution of the unobserved X given the observed Z. This model may be written as $f(x|z, \gamma_E)$, where γ_E is a vector of parameters.

Once each of these models has been specified, the likelihood function is formed, where the unobserved variable X is integrated out. Finally, the likelihood function is maximized.

3.1.5 Bayesian Methods

Bayesian methods that correct for measurement error are similar to maximum likelihood methods, in that one must specify a response model, measurement model, and possibly an exposure model. Additionally, it is necessary to select a prior distribution for each unknown parameter. These models can then be used to compute the full conditional distributions needed to implement Markov chain Monte Carlo (MCMC) methods. Gustafson (2004) provides extensive coverage of Bayesian methods for classical measurement error.

3.2 Bayesian Model

Suppose that a diagnostic test is available to detect the presence or absence of a certain condition, D. Let Y designate the true status, with $Y = 1$ if D is present and $Y = 0$

otherwise. Let Y^* be the observed result of the diagnostic test. Furthermore, let X be a continuous explanatory variable that is subject to measurement error, and let X^* be a surrogate for X. Additional covariates, Z, that are measured perfectly may be available. Our interest lies in examining the relationship between the true response Y and predictor variables X and Z.

To correct for misclassification in the response, we first define the sensitivity and specificity, which are the probabilities of correct classification. For binary response Y, define the sensitivity as $\eta \equiv P(Y^* = 1|Y = 1)$ and the specificity as $\theta \equiv P(Y^* = 0|Y = 0)$. As in Roy et al. (2005), we assume that sensitivity and specificity are independent of all covariates in the model. For an individual with covariate information ($X = x$, $Z = z$), we let $\pi_{x\,z} \equiv P(Y = 1|X = x, Z = z)$, be the probability that the individual has condition D. Then by the law of total probability, we obtain the probability that the diagnostic test detects D as

$$P(Y^* = 1|X = x, Z = z) = \eta\pi_{x,\,z} + (1 - \theta)(1 - \pi_{x,\,z}).$$

To correct for measurement error in a continuous covariate, we must specify a distribution for each component of the data. For our response model, we use $(Y^*|X = x, Z = z, \boldsymbol{\gamma}_R) \sim \text{Binomial}(n_i, \eta\pi_{x,\,z} + (1 - \theta)(1 - \pi_{x,\,z}))$ where $\pi_{x\,z} = g^{-1}(\beta_0 + \beta_1 x + \beta_2 z)$, and $g(\cdot)$ is an appropriate link function. Here, $\boldsymbol{\gamma}_R = (\eta, \theta, \beta_0, \beta_1, \beta_2)$. We use the logit link function, defined as $g(p) = \log(p/(1 - p))$.

For the measurement error process, we use the Berkson model

$$(X \mid \boldsymbol{\gamma}_M, X^* = x^*) \sim N(h(x^*), \sigma^2(x^*)), \tag{3.1}$$

where $h(x^*)$ and $\sigma^2(x^*)$ are known functions that may depend on $\boldsymbol{\gamma}_M$. This particular model was utilized by Roy, et al. (2005) in their analysis of the Life Span Study, assuming that $\boldsymbol{\gamma}_M$ was fully known. Our model will not require complete knowledge of

$\boldsymbol{\gamma}_M$; only moderately informative priors are required. See, for example, Gustafson (2004). We assume that the measurement error process is non-differential, so that X^* contains no information about the response other than what is available in X. That is, $f(y|x, x^*, z) = f(y|x, z)$. Since our measurement error model is Berkson, an exposure model is not required.

Bayesian inference requires that we specify a prior distribution for each unknown parameter. The family of beta distributions is rich in shapes, affording flexible modelling of uncertainty about probabilities. Thus, for the sensitivity and specificity we assume independent beta priors, so that $\eta \sim \text{Beta}(a, b)$ and $\theta \sim \text{Beta}(c, d)$. The regression coefficients are each given diffuse, locally uniform priors. Specifically, we take the joint prior of the regression coefficients to be $\boldsymbol{\beta} \sim N(0, \mathbf{B})$, where $\mathbf{B}$ is the diagonal matrix $\mathbf{B} = \text{diag}(100, \ldots, 100)$. This prior is non-informative relative to the likelihood because it is essentially uniform over the range of interest. The specification of prior distributions for the measurement error parameters will be dependent upon the particular choice of the measurement error model. We follow Roy et al. (2005) and use the measurement model given in (3.1) with $h(x_i^*) = x_i^*$ and $\sigma^2(x_i^*) = c(x_i^*)^2$, where c is an unknown constant. In this case, γ_M is a scalar, i.e. $\gamma_M = c$. For illustration here, we take the prior for c to be $\text{Exp}(\lambda)$. Other priors for c are discussed in Section 3.3. We denote the joint prior distribution for all parameters by $p(\boldsymbol{\gamma}_R, \gamma_M)$.

Having specified the response and measurement models and the prior distributions, the joint distribution of all relevant quantities, conditioned on z, is

$$f(x^*, y^*, x, \boldsymbol{\gamma}_R, \gamma_M \mid z) = f(y^* \mid x, z, \boldsymbol{\gamma}_R)\, f(x \mid x^*, \gamma_M)\, p(\boldsymbol{\gamma}_R, \gamma_M).$$

An important feature of this model is that we only observe (X^*, Y^*, Z). We must therefore integrate over the unobserved quantity X, so that the joint distribution is

$$\begin{aligned} f(x^*, y^*, \boldsymbol{\gamma}_R, \gamma_M \mid z) &= \int (x^*, y^*, x, \boldsymbol{\gamma}_R, \gamma_M \mid z) dx \\ &= \left\{ \int f(y^* \mid x, z, \boldsymbol{\gamma}_R) f(x \mid x^*, \gamma_M) dx \right\} p(\boldsymbol{\gamma}_R, \gamma_M). \end{aligned}$$

WinBUGS performs this integration implicitly.

For our proposed response and measurement error models, the likelihood for the observed data is

$$\begin{aligned} L(x^*, y^* \mid x, z, \boldsymbol{\gamma}_R, \gamma_M) \propto & \prod_{i=1}^{n} \left[\pi_i \eta + (1-\pi_i)(1-\theta) \right]^{y_i^*} \left[\pi_i (1-\eta) + (1-\pi_i)\theta \right]^{n_i - y_i^*} \\ & \times \prod_{i=1}^{n} \frac{1}{\sqrt{c} x_i^*} \exp\left[-\frac{1}{2} \frac{(x_i - x_i^*)^2}{c (x_i^*)^2} \right]. \end{aligned} \tag{3.2}$$

Now, the joint posterior for $\boldsymbol{\gamma}_R$, γ_M, and x can be expressed as

$$p(x, \boldsymbol{\gamma}_R, \gamma_M \mid x^*, y^*, z) \propto \left\{ \prod_{i=1}^{n} f(y_i^* \mid x_i, z_i, \boldsymbol{\gamma}_R) f(x_i \mid x_i^*, \gamma_M) \right\} p(\boldsymbol{\gamma}_R, \gamma_M).$$

In order to perform the MCMC analysis, we augment the observable data with the latent data. For the ith category, let w_{0i} denote the number of observations that are correctly classified as not having the condition. Similarly, let w_{1i} be the number of truly observations that are correctly classified as diseased. The likelihood based on the augmented data is

$$\begin{aligned} L(x^*, y^* \mid x, z, w_0, w_1, \eta, \theta, c) \propto & \, \theta^{\sum_{i=1}^{s} w_{0i}} (1-\theta)^{\sum_{i=1}^{s} (y_i^* - w_{1i})} \eta^{\sum_{i=1}^{s} w_{1i}} (1-\eta)^{\sum_{i=1}^{s} (n_i - y_i^* - w_{0i})} \\ & \times \prod_{i=1}^{s} \pi_i^{n_i - y_i^* - w_{0i} + w_{1i}} (1-\pi_i)^{y_i^* - w_{1i} + w_{0i}} \\ & \times \prod_{i=1}^{s} \frac{1}{\sqrt{c} x_i^*} \exp\left[-\frac{1}{2} \frac{(x_i - x_i^*)^2}{c (x_i^*)^2} \right]. \end{aligned} \tag{3.3}$$

Then the joint posterior, given the augmented data, is

$$p(\boldsymbol{\beta},\eta,\theta,c \mid x^*,y^*,w_0,w_1) \propto \prod_{i=1}^{s} \exp\left[(n_i - y_i^* - w_{0i} + w_{1i})(\mathbf{x}_i'\boldsymbol{\beta})\right]\left[1+\exp(\mathbf{x}_i'\boldsymbol{\beta})\right]^{-n_i}$$
$$\times \theta^{c-1+\sum_{i=1}^{s} w_{0i}} (1-\theta)^{d-1+\sum_{i=1}^{s}(y_i^* - w_{1i})} \eta^{a-1+\sum_{i=1}^{s} w_{1i}} (1-\eta)^{b-1+\sum_{i=1}^{s}(n_i - y_i^* - w_{0i})}$$
$$\times \exp\left[-\boldsymbol{\beta}'\mathbf{B}^{-1}\boldsymbol{\beta}/2\right]$$
$$\times \lambda \exp(-\lambda c)$$
$$\times \prod_{i=1}^{s} \frac{1}{\sqrt{c}x_i^*} \exp\left[\frac{(x_i - x_i^*)^2}{2c(x_i^*)^2}\right].$$

The full conditional distributions are given by

$$\eta \mid \boldsymbol{\beta},\theta,c,w_0,w_1,y^*,x^* \sim Beta\left(a-1+\sum_{i=1}^{s} w_{1i}, b-1+\sum_{i=1}^{s}(n_i - y_i^* - w_{0i})\right);$$

$$\theta \mid \boldsymbol{\beta},\eta,c,w_0,w_1,y^*,x^* \sim Beta\left(c-1+\sum_{i=1}^{s} w_{0i}, d-1+\sum_{i=1}^{s}(y_i^* - w_{1i})\right);$$

$$p(\boldsymbol{\beta} \mid \theta,\eta,c,w_0,w_1,y^*,x^*) \propto \prod_{i=1}^{s} \exp\left[(n_i - y_i^* - w_{0i} + w_{1i})(\mathbf{x}_i'\boldsymbol{\beta})\right]\left[1+\exp(\mathbf{x}_i'\boldsymbol{\beta})\right]^{-n_i}$$
$$\times \exp\left[\boldsymbol{\beta}'\mathbf{B}^{-1}\boldsymbol{\beta}/2\right];$$

and

$$p(c \mid \boldsymbol{\beta},\theta,\eta,w,y^*,x^*) \propto \lambda \exp(-\lambda c) \times \prod_{i=1}^{s} \frac{1}{\sqrt{c}x_i^*} \exp\left[\frac{(x_i - x_i^*)^2}{2c(x_i^*)^2}\right].$$

The full conditional distributions for $\boldsymbol{\beta}$ and for c do not have recognizable forms.

The WinBUGS software can perform the analysis using either the observed data model or the augmented latent data model, and in fact, runs faster using the observed data model provided in (3.2). However, model (3.3) might be of use if interest lies in quantities such as the total number of false positives or false negatives, which can be calculated directly from the MCMC output. Diagnostics such as the Brooks-Gelman-Rubin statistic, trace plots, and autocorrelation plots are available in WinBUGS and may be used to monitor convergence.

3.3 Example: The Life Span Study

We will illustrate our model using the data from the Life Span Study (LSS), which was described in the introduction. Recall that cause of death is subject to misclassification and that radiation dose is subject to measurement error. For this data, Y represents the number of actual cancer deaths, while Y^* is the number of cancer deaths observed from death certificates. Let X be the true radiation dose, and X^* the dose as estimated by DS86 dosimetry. Subjects were separated into 10 dose categories, with the mean of each category representing the surrogate, X^*. This is an example of the Berkson model. As another illustration, see Armstrong (1998), especially Design 2 on page 652. For the LSS data, there are no perfectly measured covariates. The data can be found in both Sposto et al. (1992) and Roy et al. (2005).

For a large subset of the subjects in the Life Span Study, autopsy data was available to determine true cause of death. Using this validation data, Sposto et al. (1992) estimate the sensitivity and specificity to be 0.78 and 0.965, respectively. Roy et al. (2005) take these values to be fixed for their analysis. Our Bayesian model will use these estimates in the construction of prior distributions for the misclassification parameters.

For the measurement error model, we first consider the homoscedastic Berkson model $(X|X^* = x^*) \sim N(x^*, \sigma_u^2)$. Thus the unknown parameter for the measurement model is $\gamma_M = \sigma_u^2$. Sensitivity and specificity are given independent beta priors, with $\eta \sim$ Beta(78, 22) and $\theta \sim$ Beta(96.5, 3.5). The parameters of beta priors are often thought of in terms of the "equivalent sample size" of a binomial experiment. For the purpose of illustrating our model, we have artificially set the effective sample size to be 100, though the autopsy data had a sample size of several hundred. Thus the prior distributions for

sensitivity and specificity are not as precise as the validation data would have allowed. The regression coefficients were given diffuse, locally uniform priors, as in Section 3.2. For the measurement model parameter, we first examine $\sigma_u^2 \sim \text{Exp}(1)$. We assume prior independence of all unknown parameters so that the joint prior distribution is given by $p(\eta, \theta, \beta_0, \beta_1, \beta_2, \sigma_u^2) = p(\eta)\, p(\theta)\, p(\beta_0)\, p(\beta_1)\, p(\beta_2)\, p(\sigma_u^2)$. The WinBUGS software is used to carry out the MCMC simulations from the posterior distribution. Table 5 shows the results for two simultaneous chains, each of 100,000 iterations (after a 5,000 burn-in), starting from different initial values. To reduce autocorrelation within chains, we only retain every 10^{th} iteration, a process known as thinning. Thus, the results are based on 20,000 usable iterations. Trace plots and the Gelman-Rubin diagnostic (Gelman and Rubin 1992) provide reasonable evidence that convergence has obtained. The deviance information criterion (Spiegelhalter, Best, Carlin, and van der Linde 2002) for this model was computed to be 90.925.

Table 5. Posterior Estimates for the LSS Data Using the Homoscedastic Berkson Error Model.

Parameter	Mean	Standard Deviation
β_0	−1.017	0.1530
β_1	0.279	0.0701
η	0.775	0.0421
θ	0.963	0.0193
σ_u^2	0.664	0.6276

Next, we consider the prior $\sigma_u^2 \sim \text{Gamma}(0.01, 0.01)$. This prior specification resulted in poor convergence, especially for σ_u^2, with the Gelman-Rubin diagnostic estimated to be $\hat{R} = 1.19$. The autocorrelation plots also show high autocorrelation within

chains. This choice for the measurement error variance prior is evidently problematic, as it has proven to be in other contexts. See, for example, Gelman (2006).

Next, we assume the heteroscedastic measurement error model $(X|X^* = x^*) \sim N(x^*, c(x^*)^2)$, where c represents constant coefficient of variation. Thus the unknown parameter for the measurement model is $\gamma_M = c$. This model was employed by Roy, et al. (2005), taking c to be a fixed constant. In our Bayesian approach, we construct a prior distribution to model our uncertainty about the value of c.

For the prior distributions, we again take $\eta \sim$ Beta(78, 22) and $\theta \sim$ Beta(96.5, 3.5). The regression coefficients were given diffuse, locally uniform priors as in Section 3.2. For the measurement model parameter, c must be positive, and we expect that small values are more likely. We therefore began our analysis using $c \sim$ Uniform[0.005, 1]. This distribution is bounded away from zero, since very small values for c are unlikely. For five independent chains, this prior specification resulted in poor convergence, as seen in the trace plot for β_1 in Figure 18. The poor convergence is also reflected in the Gelman-Rubin diagnostic, which is estimated to be $\hat{R} = 4.28$ for β_1.

The uniform distribution may not be a reasonable prior for c, however, because this distribution is truncated at the value one. Therefore, we consider $c \sim$ Exp(1), which allows larger values for c. A visual summary of the model is shown in Figure 19.

Table 6 shows the results for two simultaneous chains, each of 100,000 iterations, after discarding the first 5,000 as burn-in. We retained every 10^{th} update in order to get a final sample of 20,000 usable iterations. The DIC for this model is estimated to be 85.917, which is smaller than for the homoscedastic model. The simulation was also run using five parallel chains with overdispersed starting values, with similar results.

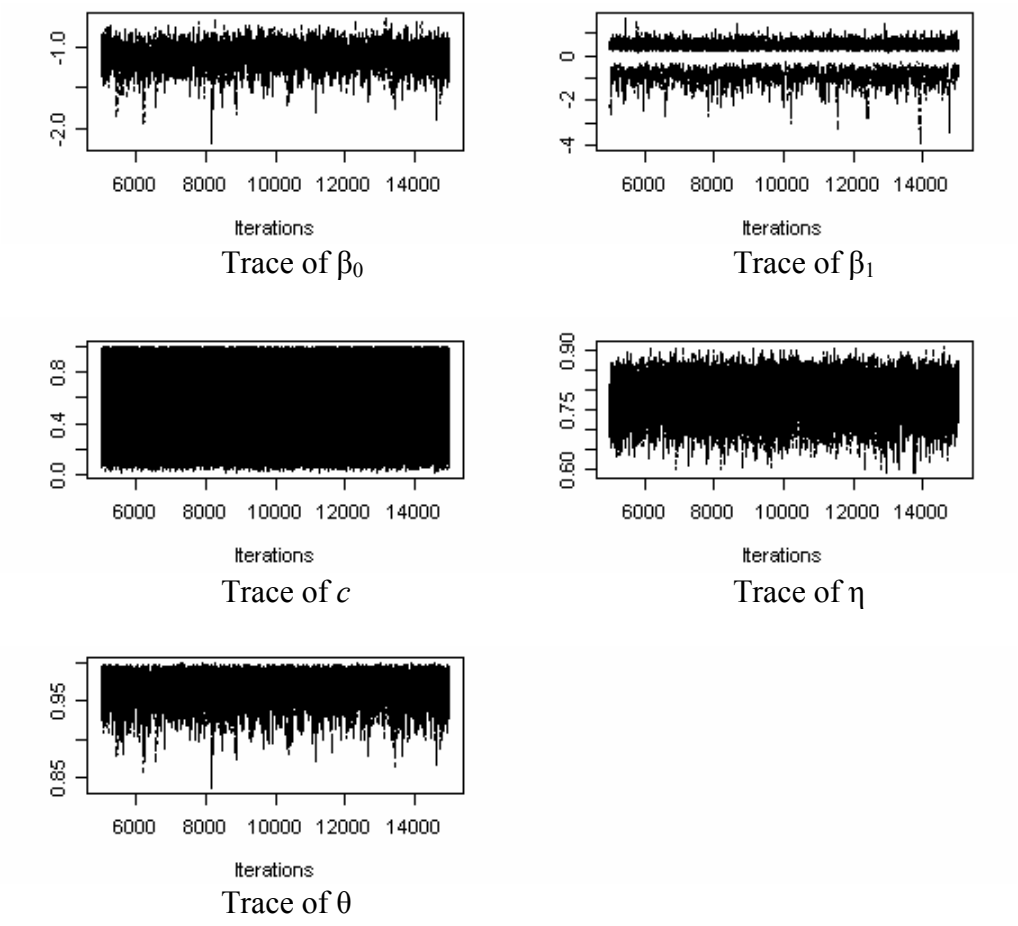

Figure 18. Trace Plots of Five Chains for the LSS Data with $c \sim$ Uniform[0.005, 1].

The parameter of particular interest is β_1, which is the regression coefficient associated with radiation dose. Its value represents the change in log odds of cancer mortality for a one-unit increase in dose. The estimate of the posterior mean of β_1 is 0.4667, which has been updated from the prior mean of zero. The approximate posterior distribution for β_1 is shown in Figure 20. Recall that the prior means for sensitivity and specificity were 0.78 and 0.965, respectively. Very little updating has occurred for these parameters, which is typical of these types of problems. Validation data or the use of more informative priors on the regression coefficients would have resulted in more updating for the misclassification parameters. Also of interest for the Life Span Study

data is π_x, which is the probability that the true cause of death was cancer for an individual with radiation dose $X = x$. Figure 21 shows the approximate posterior density of π_x, for two different values of estimated radiation dose.

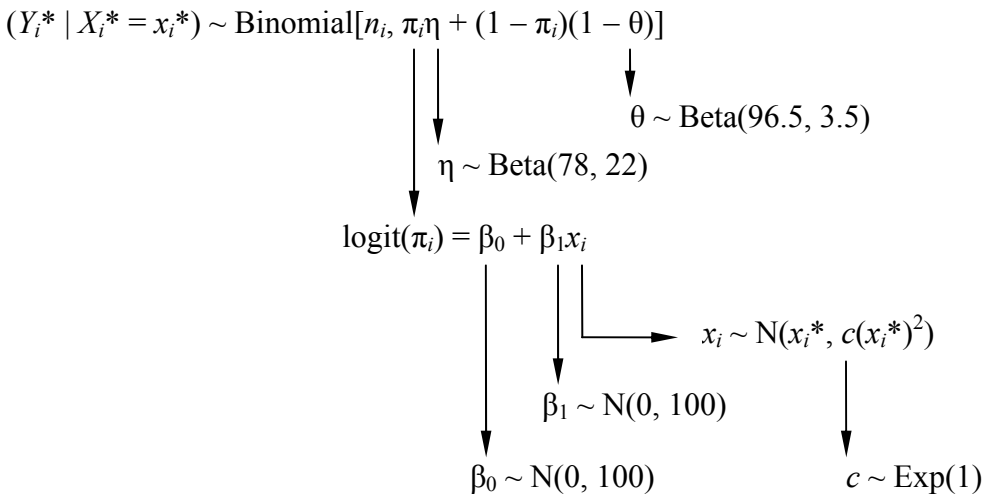

Figure 19. Summary of the Model for the Life Span Study.

Table 6. Posterior Estimates for the LSS Data Using the Heteroscedastic Error Model.

Parameter	Mean	Standard Deviation
β_0	−1.114	0.1373
β_1	0.467	0.1295
η	0.773	0.0424
θ	0.965	0.0189
c	0.455	0.4508

Trace plots for all parameters are shown in Figure 22. No observable patterns are apparent, and thus it appears that the MCMC algorithm has converged. The Gelman-Rubin statistic was close to one for all parameters in the model.

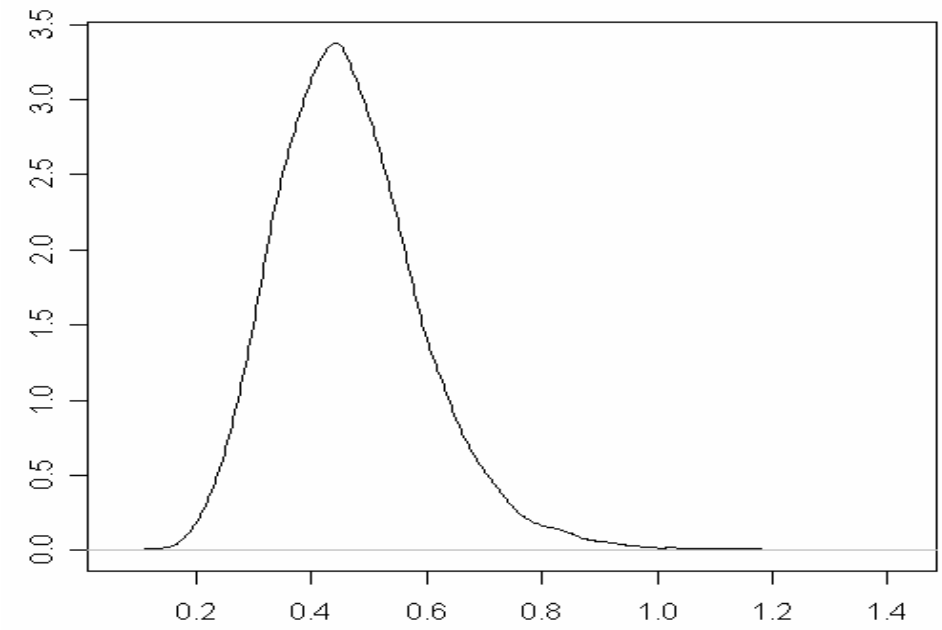

Figure 20. Approximate Posterior Distribution of β_1 for the LSS Data.

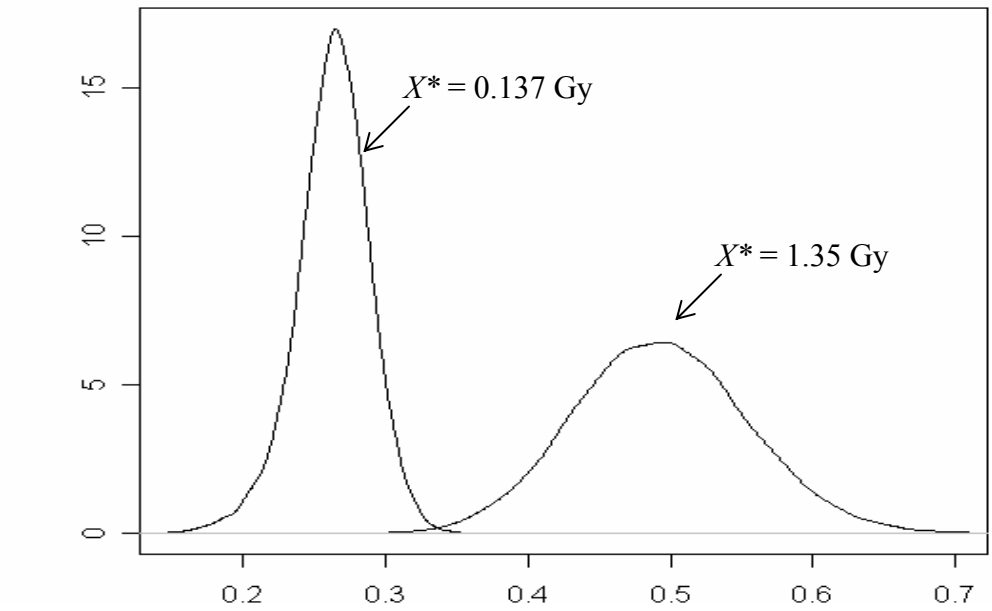

Figure 21. Approximate Posterior Distribution of π_x for Two Values of Estimated Radiation Dose (in Grays).

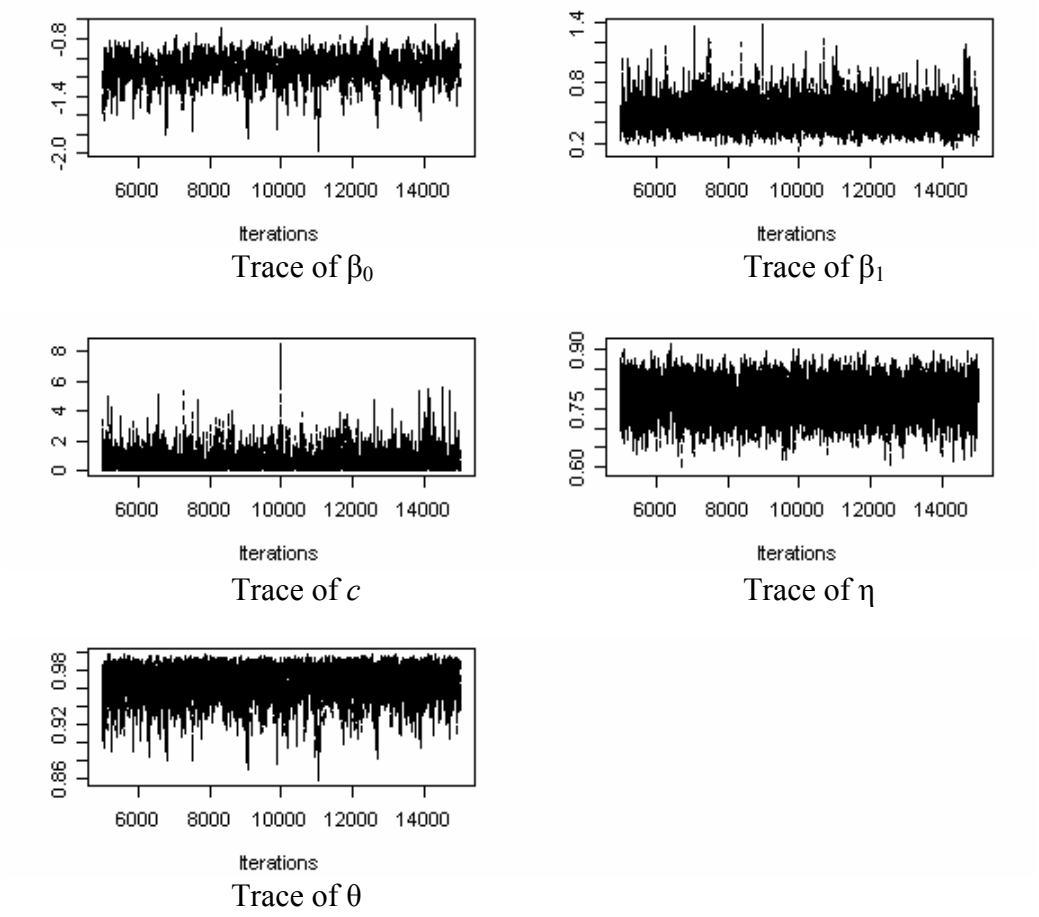

Figure 22. Trace Plots of Two Chains for the LSS Data Using $c \sim \text{Exp}(1)$.

We ran the model again using less informative priors for the sensitivity and specificity, particularly $\eta \sim \text{Beta}(39, 11)$ and $\theta \sim \text{Beta}(48.25, 1.75)$, which have an effective sample size of 50. The results were very similar to those using an effective sample size of 100.

3.3.1 Comparison of Four Models

For further comparison, we consider the following four models:

Model M1: The naïve analysis, which makes adjustments for neither measurement error nor misclassification;

Model M2: Only the misclassification is accounted for by the model;

Model M3: Only the measurement error is accounted for by the model;

Model M4: Both measurement error and misclassification are accounted for in the model.

The results for these four models are given in Table 7. Two parallel chains were used for each model, with the priors described in Figure 19 for the LSS data. These results are based on 20,000 usable iterations.

Table 7. Posterior Means (Standard Deviations) for the Four Models.

Parameter	M1	M2	M3	M4
β_0	−1.265 (0.014)	−1.005 (0.122)	−1.271 (0.016)	−1.114 (0.137)
β_1	0.331 (0.034)	0.455 (0.066)	−3.629 (0.179)	0.467 (0.130)
η	—	0.743 (0.049)	—	0.773 (0.042)
θ	—	0.972 (0.015)	—	0.965 (0.019)
c	—	—	3.298 (1.470)	0.455 (0.451)

Compared to our analysis, the naïve estimate for β_1 is biased toward zero. This is an example of attenuation as discussed in Section 3.1.1. The posterior standard deviation is also artificially small, since the naïve model does not account for the uncertainty that is introduced by the measurement error and misclassification. Figure 23 shows a comparison of the posterior distribution of β_1 using the naïve method versus our method. For *Model M3*, the estimate of β_1 is drastically different from the other models. A possible explanation for this discrepancy is the large estimated value of *c* for this model. Roy et al. (2005) found that estimates of the regression coefficients did not change much for values of *c* between 0.1 and 0.8. The estimate of *c* for *Model M3* is well outside of these bounds, but is reasonable for *Mode M4*.

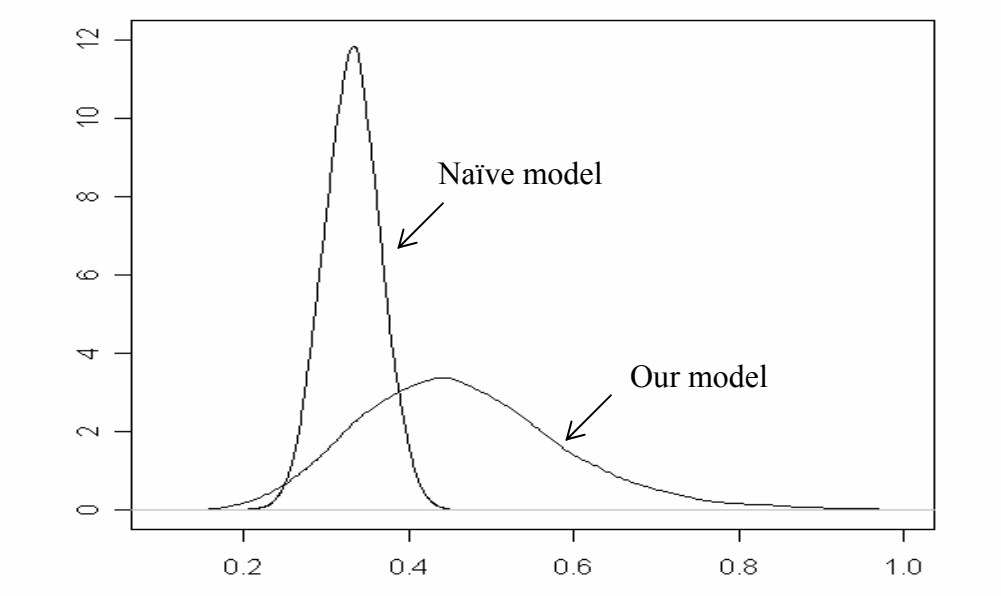

Figure 23. Approximate Posterior Distribution of β_1 for the Naïve Model and for our Model.

3.3.1 Sensitivity Analysis

To examine the sensitivity of the results to the choice of prior distributions, other choices of priors were considered. Table 8 shows the priors that were considered, with the first row representing the configuration that was used in our analysis of the data.

For the sensitivity parameter, the Beta(78, 22) prior has a standard deviation of 0.04. The other priors were obtained by shifting the mean by 0.04 in either direction, keeping the effective sample sized fixed at 100. For the measurement error parameter, an Exp(1) prior was utilized in our analysis. This prior has a mean of one and a mode of zero. We also consider an Exp(2) prior and a Gamma(2, 4) prior, both of which have a mean of 0.5. The Gamma(2, 4) prior, however, has mode 0.25. Figures 24 and 25 show the prior distributions for η and c, respectively. There are a total of nine configurations for these priors, and these configurations are shown in Table 9.

Table 8. Prior Distributions for the Sensitivity Analysis.

c	η	θ
Exp(1)	Beta(78, 22)	Beta(96.5, 3.5)
Exp(2)	Beta(74, 26)	
Gamma(2, 4)	Beta(82, 18)	

Table 9. Prior Configurations for the Sensitivity Analysis

Case	*c*	η	θ
1	Exp(1)	Beta(78,22)	Beta(96.5, 3.5)
2	Exp(2)	Beta(78,22)	Beta(96.5, 3.5)
3	Gamma(2, 4)	Beta(78,22)	Beta(96.5, 3.5)
4	Exp(1)	Beta(74, 26)	Beta(96.5, 3.5)
5	Exp(2)	Beta(74, 26)	Beta(96.5, 3.5)
6	Gamma(2, 4)	Beta(74, 26)	Beta(96.5, 3.5)
7	Exp(1)	Beta(82, 18)	Beta(96.5, 3.5)
8	Exp(2)	Beta(82, 18)	Beta(96.5, 3.5)
9	Gamma(2, 4)	Beta(82, 18)	Beta(96.5, 3.5)

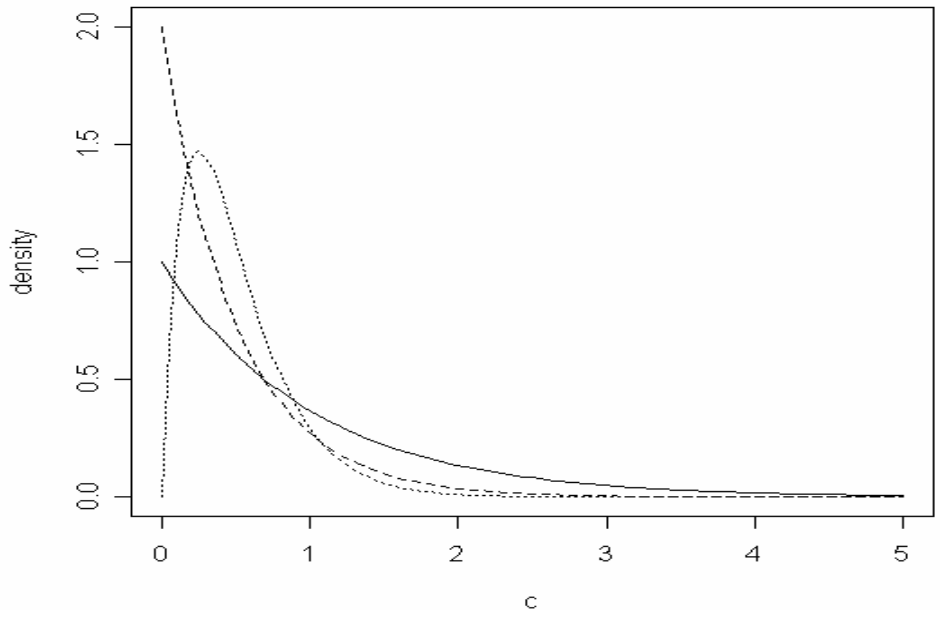

Figure 24. Three Prior Distributions for the Measurement Model Parameter, *c*, as used for the Sensitivity Analysis: Exp(1) (solid curve), Exp(2) (dashed curve), and Gamma(2, 4) (dotted curve).

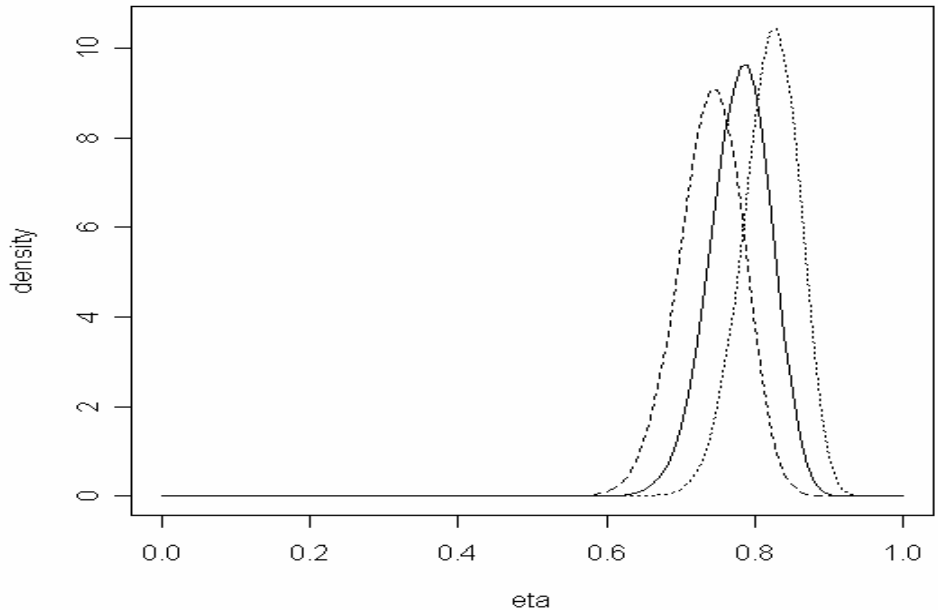

Figure 25. Three Prior Distributions for η, as used for the Sensitivity Analysis: Beta(78, 22) (solid curve), Beta(74, 26) (dashed curve), and Beta(82, 18) (dotted curve).

Table 10 shows the results of the sensitivity analysis. Results are given only for the parameter β_1. Some of these cases required more thinning than others to achieve convergence. The first row again represents the configuration that was used in our analysis of the data. The 95% credible sets were of similar width regardless of the choice of priors. There was also very little variation in the point estimate for β_1. The posterior mean ranged from 0.4464 (Case 7) to 0.4958 (Case 5). Case 7 had $\eta \sim$ Beta(74, 26) and $c \sim$ Exp(1), and Case 5 had $\eta \sim$ Beta(82, 18) and $c \sim$ Exp(1). Figure 26 compares the posterior distribution of β_1 for these two "extreme" cases, and for Case 1, which was the configuration used in our analysis. The posteriors were similar for all other parameters, illustrating nice robustness to moderate changes in the priors used.

Table 10. Results of the Sensitivity Analysis.

Case	Mean	2.5%	97.5%
1	0.4667	0.2534	0.7565
2	0.4704	0.2726	0.7381
3	0.4708	0.2794	0.7485
4	0.4791	0.2651	0.7758
5	0.4958	0.2913	0.7879
6	0.4883	0.2853	0.7783
7	0.4464	0.2459	0.7171
8	0.4663	0.2713	0.7344
9	0.4061	0.2724	0.7425

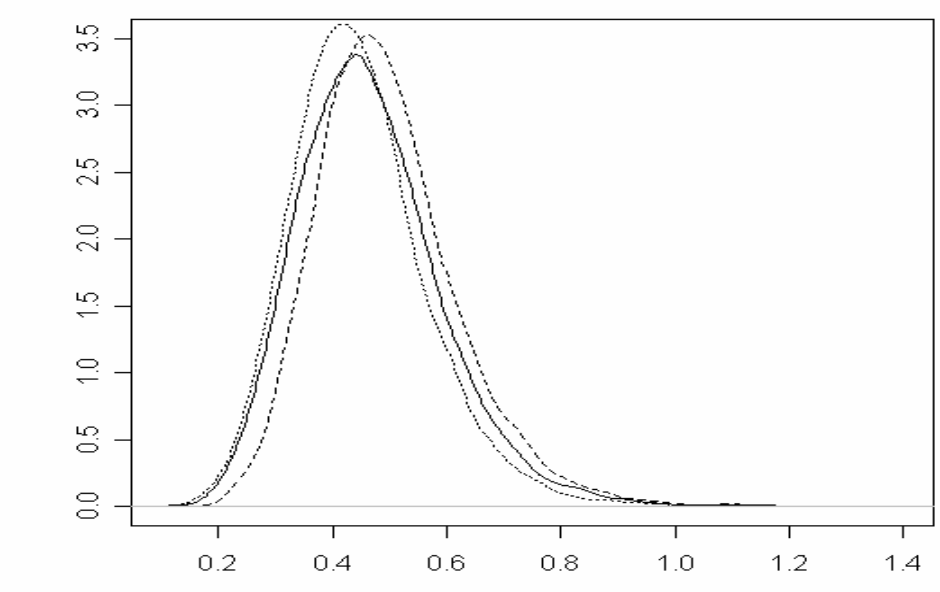

Figure 26. Approximate Posterior Distribution of β_1 for the Sensitivity Analysis: Case 1 (solid curve), Case 5 (dashed curve), and Case 7 (dotted curve).

3.4 Simulation Study

To investigate the performance of our model, the following simulation study was conducted. First, we generated the surrogate x_i^* from Uniform(−3, 3), $i = 1, \ldots 25$. Next, we generated x_i from $N(x_i^*, \sigma^2)$ for $\sigma^2 = 0.3$. We computed the true success probability as $\pi_i = \exp(\beta_0 + \beta_1 x_i)/[1 + \exp(\beta_0 + \beta_1 x_i)]$, with $\beta_0 = 0$ and $\beta_1 = 1$. The observed outcomes, y_i^* were generated from Binomial(n, $\eta\pi_i + (1 - \theta)(1 - \pi_i)$) for fixed values of η and θ.

Given the data (x_i*, y_i*), we fit the model accounting for both measurement error and misclassification.

The prior distributions were $\beta_0 \sim N(0, 100)$, $\beta_1 \sim N(0, 100)$, $\eta \sim \text{Beta}(20, 5)$, $\theta \sim \text{Beta}(22.5, 2.5)$, and $\tau \sim \text{Gamma}(33, 9)$, where τ is the precision for the measurement error model, and is defined as $\tau = 1/\sigma^2$. Note this prior for τ is equivalent to an inverse gamma prior with parameters 33 and 9 which has mean of 0.28 and standard deviation 0.051. Thus we use non-informative priors for the regression parameters, a moderately informative prior centered slightly away from the truth for the measurement error variance, and informative priors for the misclassification parameters whose true values change for each of the nine configurations of the simulation. These values are provided in Table 11.

Table 11. Fixed Values of Sensitivity and Specificity for the Simulation Study.

Case	η	θ
1	0.76	0.94
2	0.80	0.94
3	0.84	0.94
4	0.76	0.90
5	0.80	0.90
6	0.84	0.90
7	0.76	0.86
8	0.80	0.86
9	0.84	0.86

The above process was repeated 300 times. We recorded the posterior mean and 95% credible set. We also considered sample sizes of $n = 100$ and $n = 200$. Tables 12 and 13 show the average posterior mean (across the 300 replications) and the coverage for each configuration when $n = 100$ and $n = 200$, respectively.

Table 12. Average Posterior Means (Coverage) for $n = 100$.

Case	β_0	β_1	η	θ	σ
1	–0.072	1.225	0.762	0.922	0.529
	(0.960)	(0.970)	(0.987)	(0.977)	(1.00)
2	–0.011	1.237	0.788	0.922	0.529
	(0.970)	(0.953)	(0.977)	(0.973)	(1.00)
3	0.033	1.304	0.816	0.916	0.529
	(0.967)	(0.917)	(0.923)	(0.963)	(1.00)
4	–0.010	1.252	0.767	0.897	0.529
	(0.967)	(0.967)	(0.963)	(0.990)	(1.00)
5	0.012	1.216	0.794	0.894	0.530
	(0.983)	(0.970)	(0.980)	(0.980)	(1.00)
6	0.058	1.264	0.821	0.893	0.529
	(0.983)	(0.953)	(0.960)	(0.970)	(1.00)
7	0.025	1.176	0.774	0.875	0.527
	(0.990)	(0.983)	(0.983)	(0.990)	(1.00)
8	0.052	1.181	0.802	0.871	0.528
	(0.967)	(0.973)	(0.983)	(0.970)	(1.00)
9	0.079	1.207	0.829	0.864	0.528
	(0.970)	(0.983)	(0.980)	(0.990)	(1.00)

Table 13. Average Posterior Mean (Coverage) for $n = 200$.

Case	β_0	β_1	η	θ	σ
1	–0.017	1.153	0.761	0.929	0.530
	(0.960)	(0.963)	(0.967)	(0.977)	(1.00)
2	-0.023	1.195	0.789	0.925	0.529
	(0.963)	(0.933)	(0.963)	(0.957)	(1.00)
3	-0.010	1.196	0.825	0.923	0.531
	(0.977)	(0.943)	(0.960)	(0.963)	(1.00)
4	-0.023	1.081	0.774	0.902	0.530
	(0.960)	(0.993)	(0.970)	(0.983)	(1.00)
5	-0.016	1.135	0.800	0.896	0.528
	(0.967)	(0.963)	(0.977)	(0.977)	(1.00)
6	0.054	1.170	0.828	0.894	0.530
	(0.937)	(0.957)	(0.933)	(0.970)	(1.00)
7	.001	1.021	0.780	0.880	0.529
	(0.987)	(0.990)	(0.970)	(0.960)	(1.00)
8	0.031	1.092	0.807	0.871	0.529
	(0.953)	(0.963)	(0.990)	(0.963)	(1.00)
9	0.075	1.122	0.0936	0.869	0.527
	(0.933)	(0.983)	(0.987)	(0.973)	(1.00)

The average width of the 95% credible set for each parameter did not vary much for the different values of sensitivity and specificity. For the regression coefficients, the average widths were 1.34 and 1.46, respectively across simulations for the case of n = 100. For the sensitivity and specificity, the average widths were 0.17 and 0.14, respectively. For the measurement error standard deviation, the average width was 0.173. For n = 200, posterior intervals were slightly narrower. The average widths were 1.09 and 0.986 for the regression coefficients. For the sensitivity and specificity, the average widths were 0.12 and 0.11. Finally, the measurement error standard deviation interval width was 0.16 on average.

For the case n = 100, we also ran the naïve analysis which makes no corrections for measurement error and misclassification. In each case, the average posterior mean for β_1 was biased toward zero, with the bias being more than 0.3. The 95% posterior intervals for the regression coefficients were extremely narrow (0.18 and 0.12 on average for n = 100 and n = 200, respectively), leading to poor coverage (near zero percent).

This simulation study illustrates the tradeoff that exists between bias and variance. Our method, which corrected for bias, resulted in much larger interval estimates than the naïve method. However, the intervals for our method were mostly above zero, accurately indicating a relationship between the response variable and the explanatory variable. The naïve analysis also results in intervals that are entirely above the null. Thus, if interest lies only in detecting a relationship between the response and the explanatory variable, then it may not be necessary to correct for the measurement error and misclassification (Luan, Wei, Gerberich, and Carlin 2005).

3.5 Discussion

In this chapter, we have developed a Bayesian approach to modelling a misclassified binary response in the presence of measurement error in a continuous covariate. This model extends the work of Roy, et al. (2005) by requiring fewer assumptions regarding the parameters for misclassification and measurement error. We use prior distributions to model our uncertainty about the values of the parameters in both the response model and the measurement error model. In this way, the Bayesian approach provides an attractive method for adjusting inferences to account for measurement error and misclassification.

The measurement error model proposed here is only one of many possibilities. Other measurement error models should be investigated. An extension of our model could be the incorporation of random effects to accommodate additional variation. A Bayesian model selection criterion could then be used to choose between the fixed effects and random effects models.

CHAPTER FOUR

Generalized Linear Mixed Model for Misclassified Response with Covariate Measurement Error

4.1 Introduction

Misclassification in a response variable occurs in many regression applications. For instance, consider a medical test used to diagnose a certain condition. The test may be fallible and lead to an incorrect diagnosis. An analysis that ignores the misclassification in a response variable will result in biased parameter estimates. Frequentist methods for analyzing misclassified data can be found for example in Roy, Banerjee, and Maiti (2005). A Bayesian approach is presented in McInturff, Johnson, Cowling, and Gardner (2004).

An additional difficulty may arise if one or more explanatory variables are also measured imperfectly. Carroll, Ruppert, Stefanski, and Crainiceanu (2006) describe numerous methods to adjust for covariate measurement error. For one application of these methods, see Roy et al. (2005). Gustafson (2004) gives special attention to measurement error in the Bayesian setting.

An assumption that is typically made in regression problems is that the responses are independent. In many situations, however, this assumption may be violated. For example, in a longitudinal study in which multiple measurements are taken on the same individual, correlation between the responses is likely. Such a correlation structure can be accommodated by including random effects, resulting in a mixed model. This technique is illustrated in Paulino, Silva, and Achcar (2005).

In this chapter, we formulate a Bayesian model that corrects for a misclassified response and a mismeasured covariate while making adjustments for correlation between responses. The remainder of the chapter is organized as follows. In Section 4.2, we develop the Bayesian model. In Section 4.3, we describe the data that was generated to illustrate the model. The analysis of the simulated data is presented in Section 4.4, and we conclude with a discussion in Section 4.5.

4.2 Bayesian Model

Let Y_i be a binomial random variable with success probability π_i, $i = 1, \ldots, s$, where s represents the number of strata. Let the sensitivity, η, be the probability of correctly classifying a "success" for each Bernoulli trial, and let the specificity, θ, be the probability of correctly classifying a "failure." Let Y_i^* be a random variable counting the number of observations that are classified as successes. Then Y_i^* is a binomial random variable with success probability given by $p_i = \eta\pi_i + (1 - \theta)(1 - \pi_i)$.

Our interest lies in determining the relationship between the true response Y and a continuous explanatory variable X that is measured with error. That is, instead of observing X, we only observe its surrogate, X^*. As in Carroll et al. (2006), we must specify three components to modify our analysis in the presence of measurement error. These components are the response model, the measurement error model, and the exposure model. The response model relates the dependent variable to the explanatory variables. We take $\text{logit}(\pi_i) = \beta_0 + \beta_1 x_i + \varepsilon_i$, where the ε_i's are independent random effects to account for possible correlation among the responses. The random effects are assumed to be normally distributed with mean zero and constant variance, σ_ε^2.

The measurement error model relates the true covariate to its surrogate. For our measurement error process, we assume the homoscedastic Berkson model given by $X|X^* = x^* \sim N(x^*, \sigma_B^2)$. This choice was motivated by Roy et al. (2005). The Berkson model assumes that group averages for a covariate are used in place of individual measurements; see Section 3.1.1 and Armstrong (1998). This is in contrast to classical measurement error, in which the average of multiple measurements of a covariate will equal the true value. Since our measurement model contains no classical components, we do not require an exposure model. We assume non-differential measurement error so that $f(y|x, x^*) = f(y|x)$.

Let $\mathbf{y}^* = (y_1^*, \ldots, y_s^*)$, $\mathbf{x}^* = (x_1^*, \ldots, x_s^*)$, and $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_s)$. Given our response and measurement models, the likelihood of the observed data is given by

$$L(\mathbf{y}^*, \mathbf{x}^* \mid \mathbf{x}, \beta_0, \beta_1, \eta, \theta, \boldsymbol{\varepsilon}, \sigma_B^2, \sigma_\varepsilon^2) \propto \prod_{i=1}^{s} p_i^{y_i^*} (1 - p_i)^{n_i - y_i^*} \times \exp\left[-\frac{1}{2\sigma_B^2}(x_i - x_i^*)^2 - \frac{\varepsilon_i^2}{2\sigma_\varepsilon^2} \right],$$

where $\mathbf{x} \equiv (x_1, \ldots, x_s)$ are not observed. Consequently, the likelihood must be integrated over X. WinBUGS performs this integration implicitly.

To complete the Bayesian model, we must specify prior distributions for all unknown parameters. The family of beta distributions has a great variety of shapes that are useful for describing prior information for probabilities. Accordingly, sensitivity and specificity are given independent beta priors. For the regression coefficients, we use independent N(0, 100) priors. These diffuse normal priors are relatively noninformative compared to the likelihood because they are essentially flat where the likelihood is peaked. For the variance component, we place a uniform prior on the standard deviation, σ_ε, as recommended by Gelman (2006). In the support of this prior, we constrain the

standard deviation away from zero since very small values are unlikely. An appropriate upper bound, *UB*, must be chosen for this uniform prior. For the standard deviation of the measurement error, σ_B, we use a Gamma(α, δ) prior distribution. Appropriate hyper-parameters must be specified. A graphical summary of this model is displayed in Figure 27.

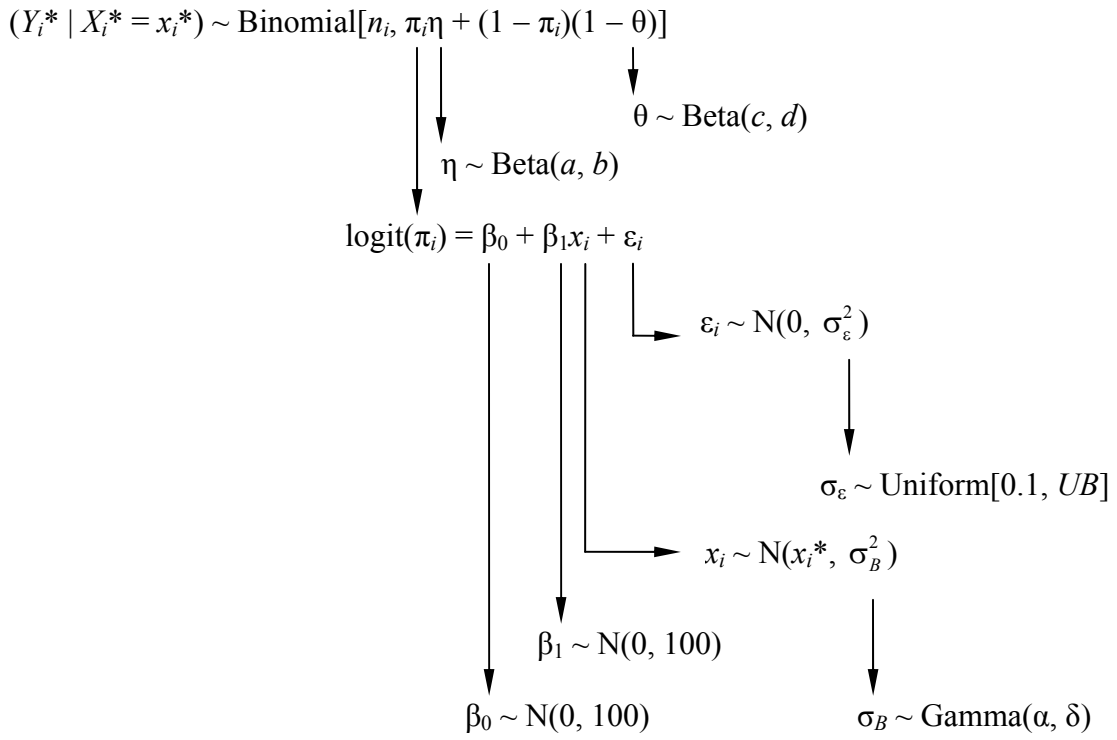

Figure 27. Summary of the Mixed Model for Misclassified Response with Covariate Measurement Error.

The determination of the prior parameters for θ and η proceeds as follows. We shall use the notion of equivalent sample size and set the mean of each prior to the prior hypothesized values of sensitivity and specificity. Thus, if an equivalent sample size, n^*, is considered, and it is believed $\theta = \theta^*$ and $\eta = \eta^*$ are the likely values, we have the equations

$$a + b = n^*, \quad c + d = n^*, \quad \frac{a}{a+b} = \theta^*, \text{ and } \frac{c}{c+d} = \eta^*.$$

From this system we obtain, for specified θ^*, η^*, and n^*,

$$b = n^*(1 - \theta^*), \; d = n^*(1 - \eta^*), \; a = n^* - b, \text{ and } c = n^* - d. \tag{4.1}$$

Once appropriate prior distributions have been specified, the joint posterior distribution is proportional to the product of the likelihood and the joint prior distribution. The MCMC analysis is carried out using the WinBUGS software.

In order to perform the MCMC analysis, we augment the observable data with the latent data. For the i^{th} category, let w_{0i} denote the number of observations correctly classified as not having the condition. Similarly, let w_{1i} be the number of observations that are correctly classified as diseased. Then the likelihood based on the augmented data is

$$\begin{aligned} L(\mathbf{x}^*, \mathbf{y}^* \mid \mathbf{x}, w_0, w_1, \eta, \theta, c) \propto\; & \theta^{\sum_{i=1}^{s} w_{0i}} (1-\theta)^{\sum_{i=1}^{s} (y_i^* - w_{1i})} \eta^{\sum_{i=1}^{s} w_{1i}} (1-\eta)^{\sum_{i=1}^{s} (n_i - y_i^* - w_{0i})} \\ & \times \prod_{i=1}^{s} \pi_i^{n_i - y_i^* - w_{0i} + w_{1i}} (1-\pi_i)^{y_i^* - w_{1i} + w_{0i}} \\ & \times \prod_{i=1}^{s} \frac{1}{\sigma_B \sigma_\varepsilon} \exp\left[-\frac{(x_i - x_i^*)^2}{2\sigma_B^2} - \frac{\varepsilon_i^2}{2\sigma_\varepsilon^2} \right]. \end{aligned}$$

Then the posterior distribution, given the augmented data is

$$\begin{aligned} p(\boldsymbol{\beta}, \eta, \theta \mid \mathbf{x}^*, \mathbf{y}^*, w_0, w_1) \propto\; & \prod_{i=1}^{s} \exp\left[(n_i - y_i^* - w_{0i} + w_{1i})(\mathbf{x}_i' \boldsymbol{\beta}) \right] \left[1 + \exp(\mathbf{x}_i' \boldsymbol{\beta}) \right]^{-n_i} \\ & \times \theta^{c-1+\sum_{i=1}^{s} w_{0i}} (1-\theta)^{d-1+\sum_{i=1}^{s} (y_i^* - w_{1i})} \eta^{a-1+\sum_{i=1}^{s} w_{1i}} (1-\eta)^{b-1+\sum_{i=1}^{s} (n_i - y_i^* - w_{0i})} \\ & \times \exp\left[-\boldsymbol{\beta}' \mathbf{B}^{-1} \boldsymbol{\beta} / 2 \right] \\ & \times (\sigma_B)^{\alpha - s - 1} \sigma_\varepsilon^{-s} \exp\left[-\delta \sigma_B - \frac{1}{2} \sum_{i=1}^{s} \left(\frac{(x_i - x_i^*)^2}{\sigma_B^2} + \frac{\varepsilon_i^2}{\sigma_\varepsilon^2} \right) \right]. \end{aligned}$$

4.3 Simulated Data

To demonstrate our model, we generate data with known properties. This section gives the details for this simulated data. Parameter values for the regression coefficients, the sensitivity and specificity were motivated by the analysis of the Life Span Study data as discussed in Chapter 3.

A grid of 25 surrogate x^* values was constructed on the interval (0, 5), with a distance of 0.01 between values. We generated values of the covariate x from $N(x^*, \sigma_B^2)$, with $\sigma_B = 0.5$. Random effects were generated from $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, with $\sigma_\varepsilon = 0.5$. The probability of true success, $P(Y = 1)$, was $\pi_i = \exp(\beta_0 + \beta_1 x_i + \varepsilon_i) / \left[1 + \exp(\beta_0 + \beta_1 x_i + \varepsilon_i)\right]$. For the regression coefficients, we set $\beta_0 = -1.1140$ and $\beta_1 = 0.4667$. The sensitivity and specificity were taken to be $\eta = 0.78$ and $\theta = 0.965$, respectively. Then the probability of an observed success, $P(Y^* = 1)$, was calculated from $p_i = \eta\pi_i + (1 - \theta)(1 - \pi_i)$. Finally, we generated the observed responses y_i^* from Binomial(50, p_i).

4.4 Analysis of Simulated Data

In this section, we describe the analysis of the simulated data (Section 4.3) using the model described in Section 4.2. We assume independent beta priors for the sensitivity and specificity, with "likely" values $\theta^* = 0.965$ and $\eta^* = 0.78$, and equivalent sample size $n^* = 50$. Using (4.1) we obtain the priors $\eta \sim$ Beta(39, 11) and $\theta \sim$ Beta(48.25, 1.75). Noninformative priors are used for the regression coefficients, as discussed in Section 4.2. We place a Uniform(0.1, 4) prior on the standard deviation of the random effect. For the standard deviation of the measurement error model, we use Gamma(2, 4) which has a mean of 0.5 and a mode of 0.25. To alleviate autocorrelation

within chains, we thinned each chain, retaining every 50th update. We used two independent chains, each with 10,000 usable iterations after thinning and a 5,000 burn-in. The results are presented in Table 14.

Table 14. Posterior Summaries for the Simulated Data.

Parameter	Mean	SD	2.5%	Median	97.5%
β_0	–1.112	0.4171	–2.009	–1.079	–0.395
β_1	0.571	0.2005	0.303	0.539	1.039
η	0.745	0.0651	0.609	0.749	0.862
θ	0.958	0.0308	0.878	0.965	0.996
σ_B	0.506	0.3086	0.072	0.452	1.220
σ_ε	0.576	0.2567	0.155	0.552	1.151

Gelman (2006) describes miscalibration as the Bayesian analog to bias. Thus, if τ^* is the "true" value of a parameter then $E(\tau) - \tau^*$ is the miscalibration for τ under the model yielding posterior expectation $E(\tau)$. Table 14 shows that the miscalibration is minimal for our model with these parameters. The parameter of particular interest is β_1, the coefficient associated with the explanatory variable. There is a slight positive miscalibration for this parameter. The credible interval is very wide, but is still entirely above zero.

The autocorrelation plots in Figure 28 show that the correlation within chains drops off after about 10 lags. The trace plots in Figure 29 do not suggest any noticeable patterns, indicating good mixing. The Gelman-Rubin statistic is approximately one, as desired, for all parameters.

The deviance information criterion (DIC) was proposed by Spiegelhalter, Best, Carlin, and van der Linde (2002) as a method to compare the relative fit of competing models. See Section 2.2.2 for a formal definition of DIC. The model with the smallest

DIC should best predict a new data set with similar structure. For our model the estimated DIC is 148.994. We also analyze the data using a fixed effects model that does not accommodate correlation between responses. The estimated DIC for the fixed effects analysis is 150.345. The DIC indicates a preference for the random effects model.

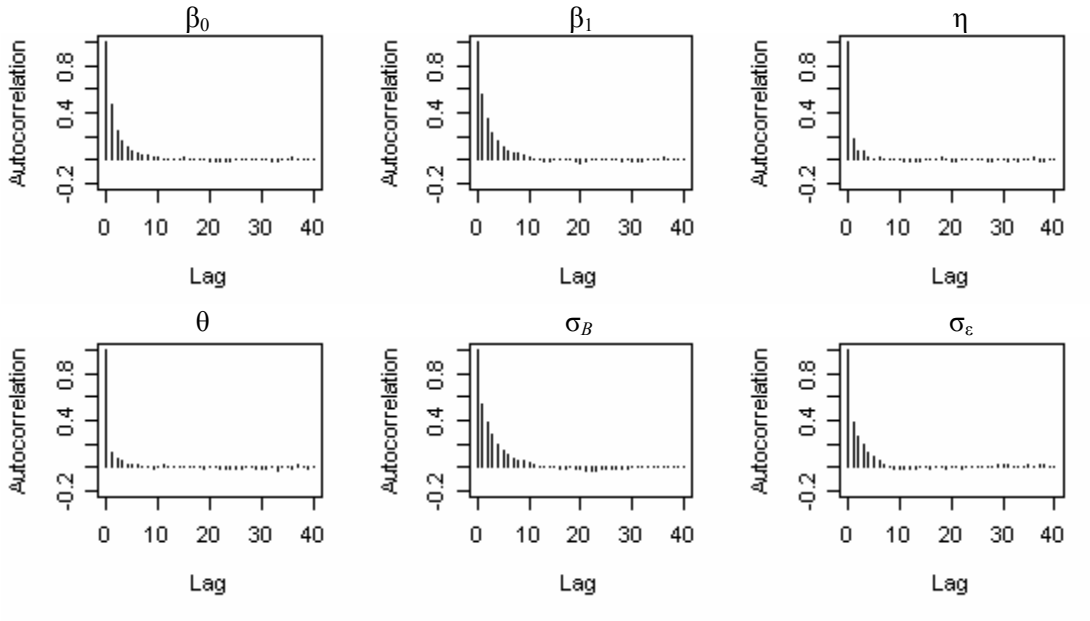

Figure 28. Autocorrelation Plots for the Simulated Data.

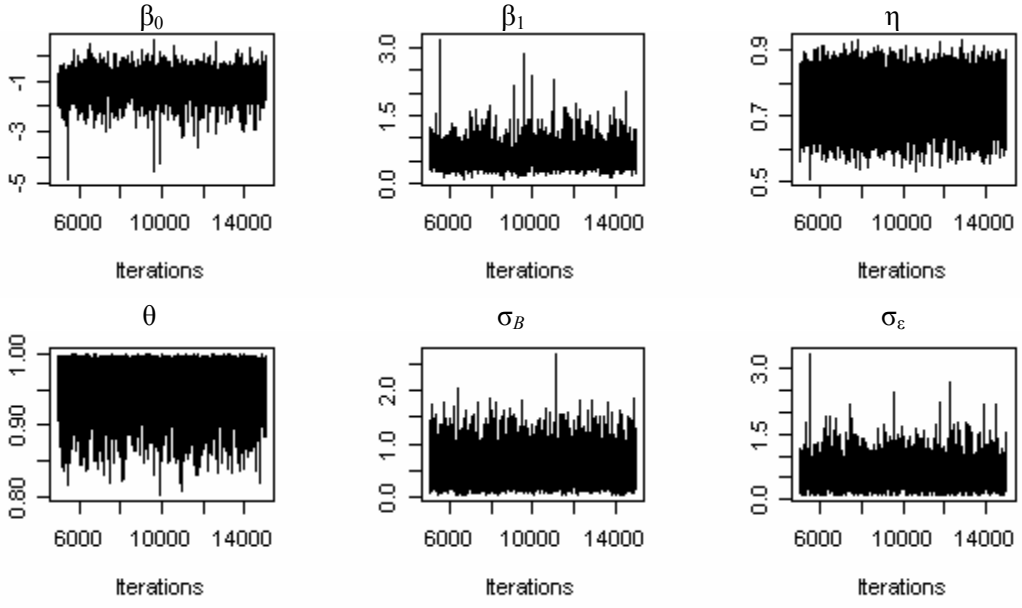

Figure 29. Trace Plots for the Simulated Data.

We checked the robustness of the posterior to changes in the upper bound on the uniform prior on σ_ε. To do so, we considered a sequence of values for *UB*, and ran the model for each value in the sequence, each time monitoring the posterior summaries for σ_ε. In Figure 30, we plot the posterior 2.5th percentile, median, and 97.5th percentile against the upper bound *UB*. Note that these quantities become very stable after *UB* = 5.

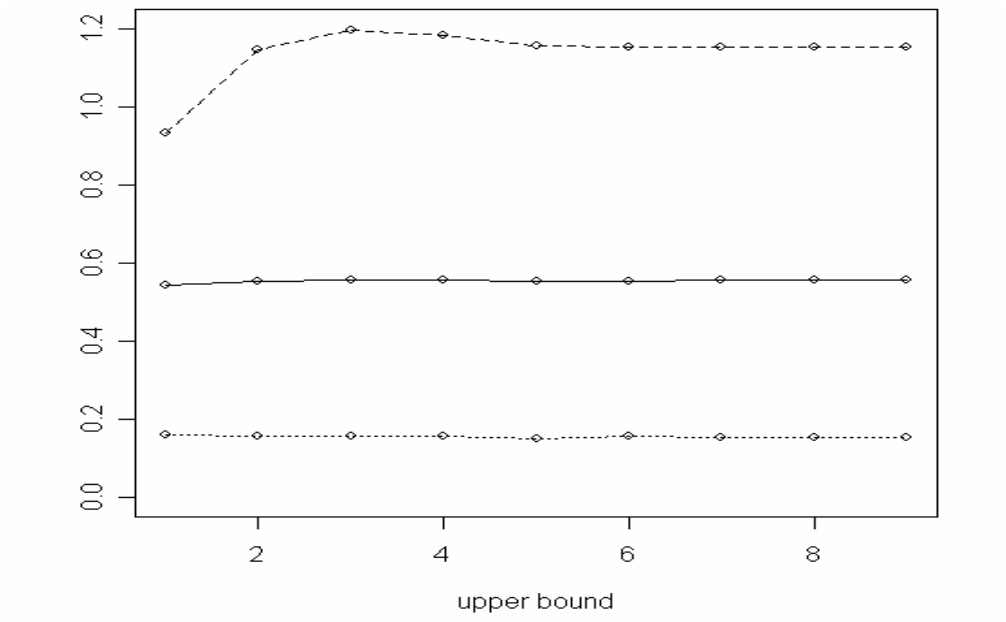

Figure 30. Posterior 2.5th Percentile (dotted line), Median (solid line), and 97.5th Percentile (dashed line) for Various Values of *UB* in $\sigma_\varepsilon \sim$ Uniform[0.1, *UB*].

We now compare our results to the model in which one naïvely ignores the measurement error and misclassification. In Figure 31, we present a comparison of the posterior distribution for β_1 under the naïve model and our model. The vertical line indicates the actual value of β_1. We see that that the posterior under the naïve model is shifted toward zero and displays smaller variability than our model. This attenuated posterior variance is artificial, and could easily yield inappropriate inferences about β_1. Table 15 gives the posterior estimates under the naïve model.

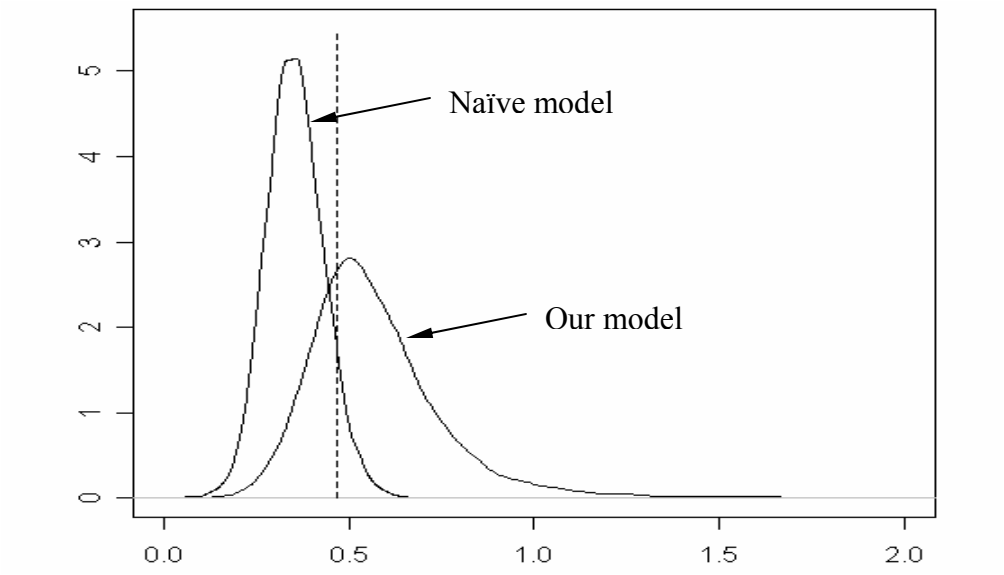

Figure 31. Posterior Distribution of β_1 using the Naïve Model and the Mixed Model for a Misclassified Response and Covariate Measrement Error. The vertical line indicates the true value of β_1.

Table 15. Posterior Summaries for the Simulated Data using the Naïve Model

Parameter	Mean	SD	2.5%	Median	97.5%
β_0	−1.196	0.2195	−1.642	−1.187	−0.786
β_1	0.355	0.0780	0.210	0.352	0.516
σ_ε	0.436	0.1025	0.260	0.428	0.664

4.5 Discussion

In this chapter, we have developed a Bayesian model to analyze correlated data in which the response is misclassified and a covariate is measured with error. The proposed model performed well for a simulated data set with known characteristics. By modifying the analysis to incorporate the measurement error and misclassification, we were able to adjust the parameter estimates to reduce bias. We also found that the DIC indicated the superior efficacy of the random effects model over the fixed effects model.

To further verify the robustness of our model, we should generate data with different properties. It would be interesting to investigate the performance of the model

for data with various amounts of measurement error and misclassification. In this chapter, we have only considered Berkson measurement error with constant variance. A similar model could be developed for data with classical measurement error.

CHAPTER FIVE

Conclusions

Measurement error and misclassification are inevitable in a variety of regression applications. Fallible measurement tools must often be used because infallible methods are either expensive or not available. Ignoring measurement problems will result in biased estimates for the associated regression parameters. The models presented in this dissertation are designed to correct this bias and adjust the variability of the estimates to appropriately reflect the uncertainty that is introduced by flawed measurements.

In Chapter 2, we present a generalized linear mixed model (GLMM) for a misclassified response. This model accommodates correlation among responses within a cluster. Informative priors are utilized for the regression coefficients. We discuss a variety of methods to improve the convergence properties of the model. Several techniques for model selection are presented, and the deviance information criterion (DIC) is used to compare the mixed model to the fixed effect model. For our simulated data, the DIC preferred the less complex fixed effects model.

In Chapter 3, we develop techniques for the case where measurement problems exist in both the response and explanatory variables. Specifically, we assume a dichotomous response and a continuous covariate. We illustrate the model using the Life Span Study, and show that our model is superior to an analysis that ignores the two sources of error. For our analysis we chose to examine only the logistic link function. Roy et al. (2005) use the probit link in their analysis of the LSS data. For a direct comparison to their results, we should modify our analysis accordingly.

In Chapter 4, we combine the techniques of the previous two chapters and develop a GLMM for a correlated misclassified response and covariate measurement error. This model is illustrated using a simulated data set.

There are many opportunities for further research regarding the techniques described in this dissertation. We have focused our attention on Berkson errors and essentially ignored classical measurement error. This choice was motivated by our analysis of the Life Span Study data. Furthermore, we have been concerned only in methods to correct for measurement error, and have not discussed the important task of choosing an appropriate measurement error model. This choice can have a considerable impact on the analysis.

Finally, we have assumed nondifferential misclassification and measurement error for all models. An interesting avenue for further research could be techniques for data in which either misclassification or measurement error is differential.

APPENDICES

APPENDIX A

Chapter Two Programs

A.1 Code to Illustrate CMP Priors for the Mixed Model

This program simulates values from the conditional means priors (See Sections 2.1.2 and 2.1.4) on the regression coefficients. For the mixed model, we use a uniform prior on the standard deviation of the random effects. That is, $\sigma \sim \text{Uniform}(0, B)$. We consider the CMP under the fixed effects model, the mixed model with $B = 5$, and the mixed model with $B = 10$. This program produces the figures in Section 2.1.4.

First, we set up the design matrix $\tilde{\mathbf{X}}$. *Then we simulate n = 10,000 observations from the priors on* $\tilde{\pi}_i$. *For this example, we take* $\tilde{\pi}_1 \sim$ *Beta(8, 10) and* $\tilde{\pi}_i \sim$ *Beta(3, 13).*

```
X <- cbind(c(1, 1), c(1, 0)); n <- 10000
pi1 <- rbeta(n, 8, 10); pi2 <- rbeta(n, 3, 13)
PI <- rbind(pi1, pi2)
```

Here, we use a transformation to induce a prior for $\boldsymbol{\beta}$ *under the three different models. The "solve" command computes the inverse of the matrix* $\tilde{\mathbf{X}}$.

```
# fixed model:
logit.pi <- log(PI/(1 - PI))
beta.f <- solve(X)%*%logit.pi

# mixed model with B = 5;
UB <- 5; sig <- runif(1, 0, UB); eps <- matrix(NA, 2, n)
for(i in 1:2){
 eps[i,] <- rnorm(n, 0, sig)
}
U <- solve(X)%*%logit.pi; V <- solve(X)%*%eps
beta.m <- U - V

# mixed model with B = 10
UB <- 10; sig <- runif(1,0,UB); eps <- matrix(NA,2,n)
for(i in 1:2){
 eps[i,] <- rnorm(n,0,sig)
}
U <- solve(X)%*%logit.pi; V <- solve(X)%*%eps
beta.m2 <- U – V
```

In this section of code, we plot histograms of the simulated values from the priors on $\boldsymbol{\beta}$.

```
# histograms of beta:
par(mfrow=c(3,2))
hist(beta.f[1,],xlab="",main="beta0
(fixed model)")
hist(beta.f[2,],xlab="",main="beta1
(fixed model)")
hist(beta.m[1,],xlab="",main="beta0
(mixed model, B=5)")
hist(beta.m[2,],xlab="",main="beta1
(mixed model, B=5)")
hist(beta.m2[1,],xlab="",main="beta0
(mixed model, B=10)")
hist(beta.m2[2,],xlab="",main="beta1
(mixed model, B=10)")
```

Now we examine the CMP priors using $\tilde{\mathbf{X}}$ *and* $p(\tilde{\boldsymbol{\pi}})$ *as given in McInturff et al. (2004). We consider a grid of values for B, and monitor the interquartile range of the induced prior on* $\boldsymbol{\beta}$ *after each simulation. We are concerned only with the results for* β_2.

```
n <- 10000
Xtili <- cbind(rep(1,6),c(1,0,1,1,1,0),c(0,1,0,0,0,0),c(1,1,1,0,0,1),
 c(0,0,0,1,0,0),c(1,1,0,1,1,0))
pi1 <- rbeta(n, 8, 15); pi2 <- rbeta(n, 10, 15); pi3 <- rbeta(n, 3, 13)
pi4 <- rbeta(n, 8, 10); pi5 <- rbeta(n, 4, 15); pi6 <- rbeta(n, 6, 15)

PI <-rbind(pi1, pi2, pi3, pi4, pi5, pi6)
logit.pi <- log(PI/(1-PI))

# fixed model:
beta.f <- solve(Xtili)%*%logit.pi

upperBound <- seq(0,10,by=.5)      # the upper bound for the uniform prior on sigma
IQR.m <- vector(mode="numeric",length=length(upperBound))
for (j in 1:length(upperBound)){
 UB <- upperBound[j]
 sig <- runif(1, 0, UB);  eps <- matrix(NA, 6, n)
 for(i in 1:6){
  eps[i,] <- rnorm(n, 0, sig)
 }
 U <- solve(Xtili)%*%logit.pi;  V <- solve(Xtili)%*%eps
 beta.m <- U - V
 IQR.m[j] <- IQR(beta.m[2,])
}
```

```
# for beta2:
IQR.f <- IQR(beta.f[2,])
```

The following code produces a plot of IQR by B. The output is shown in Figure 6. The "lines" function draws a line on the existing plot, showing the IQR for the fixed effects model. The "lty" command specifies that the line should be a dashed line. The axis function places an axis on the vertical axis (indicated by "2"). The "las" command indicates that the axis label should be placed perpendicular to the axis.

```
par(mfrow=c(1,1))
plot(upperBound, IQR.m, xlab="B", ylab="Interquartile Range")
lines(c(-.5, 10), rep(IQR.f, 2), lty=2)
axis(2, at=IQR.f, labels="IQR.f", las=1)
```

A.2 Code to Analyze Simulated Data Similar to McInturff et al. (2004)

This program simulates data using parameter estimates given in McInturff et al. (2004). We simulate data with only fixed effects (Data A) and data with both fixed and random effects (Data B). These data sets are then analyzed using both a fixed effects and a mixed model. This program was used to produce the output in Section 2.3

First, we set the working directory in R. The WinBUGS model should be saved as a text file in this folder under the name "McBugs.txt." All MCMC output from WinBUGS will be stored in this folder. We also load the necessary libraries in R. The package R2WinBUGS contains functions for linking between R and WinBUGS. The coda package has functions for analyzing MCMC output.

```
setwd("E:/Research/R dir")

# load the required packages in R:
library(R2WinBUGS); library(coda)
```

Now we set up the data structure. The vector n gives the sample size for each stratum defined by the covariates. There are 17 strata. The matrix X is the design matrix, with the first column consisting of ones.

```
N <- 17
n <- c(9, 27, 1, 6, 8, 6, 18, 6, 9, 10, 15, 23, 24, 12, 49, 32, 106)
x2 <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1)
x3 <- c(0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0)
x4 <- c(0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1)
x5 <- c(0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0)
x6 <- c(0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1)
X <-cbind(rep(1, 17), x2, x3, x4, x5, x6)
```

The matrix xtili is the inverse of $\tilde{\mathbf{X}}$ *(see McInturff et al. 2004), and is used in the computation of the conditional means prior (CMP) for the regression coefficients. We note that WinBUGS and R structure matrices differently, so that we actually create the transpose of xtili.*

```
txtili <- structure(.Data=c(
       -1.0,  1.943E-16,  3.331E-16,  2.567E-16,        1.0,        1.0,
  -7.772E-16,  2.498E-16,        1.0, -4.857E-17,  3.331E-16,        1.0,
       -1.0,        1.0,        1.0, -3.747E-16, -3.331E-16,       -1.0,
        1.0, -5.829E-16, -5.551E-16, -9.021E-17,       -1.0,  8.882E-16,
   4.163E-16, -6.106E-16, -3.886E-16,        1.0,       -1.0,  5.412E-16,
        1.0,  1.665E-16,       -1.0,  3.331E-16,  5.551E-17,  7.772E-16),
 .Dim=c(6,6))
xtili<-t(txtili)
```

Here we give the true parameter values, which were taken from McInturff et al. (2004).

```
eta <- 0.992; theta <- 0.901;  beta <- c(-1.27, -1.75, -1.32, 0.86, 0.37, 1.29)
```

Here we simulated Data A, which has only fixed effects. The responses y[i] are simulated from a binomial distribution with parameters n[i] and p[i], where p[i] is a function of the sensitivity (eta), specificity (theta), and the probability of true success (pie).

```
# Data with NO random effects:
yA <- vector(mode="numeric", length=17)
logit.pie <- X%*%beta
pie <- exp(logit.pie)/(1 + exp(logit.pie))
p <- pie*eta + (1 - pie)*(1 - theta)
for(i in 1:17){
 yA[i] <- rbinom(1, n[i], p[i])
}
```

Here we simulated Data B, which has both fixed and random effects. The random effects are simulated from a normal distribution with mean zero and common precision (rho). We considered three different values for the true precision: 9, 12, and 15.

```
# Data WITH random effects:
yB <- vector(mode="numeric", length=17)
rho <- 9
e <- rnorm(17, 0, 1/sqrt(rho))
logit.pie <- X%*%beta + e
pie <- exp(logit.pie)/(1 + exp(logit.pie))
p <- pie*eta + (1 - pie)*(1 - theta)
for(i in 1:17){
 yB[i] <- rbinom(1, n[i], p[i])
```

```
}
```

The following lines identify the data set to be analyzed. Change yA to yB in order to analyze Data B. The value UB is the upper bound on the uniform prior for the standard deviation of the random effects. The "data" statement formats the data for use by WinBUGS, and the "parameters" statement lists all parameters of interest. For the fixed effects analysis, we would delete UB from the data list and remove rho from the parameter list.

```
y <- yA
UB <- 5
data <- list("y","n","x2","x3","x4","x5","x6","xtili","N","UB")
parameters <- c("beta","eta","theta","rho")
```

We give initial starting values for all unknown parameters in the model. Different values are given for each chain. For the fixed effects analysis, remove from this list the following: eps 361 through eps366, sigma, and eps.

```
# Initial values:
inits1<-list(pi361=0.33,pi362=0.39,pi363=0.14,pi364=0.44,pi365=0.18,pi366=0.26,
  eta=0.99,theta=0.90,sigma=.2,eps=rep(0,N),
  eps361=-.27,eps362=.72,eps363=1.19,eps364=1.62,eps365=-1.79,eps366=.45)
inits2<-list(pi361=0.5,pi362=0.5,pi363=0.3,pi364=0.2,pi365=0.4,pi366=0.4,
  eta=0.9,theta=0.75,sigma=.95,eps=rep(0,N),
  eps361=.61,eps362=-.45,eps363=.08,eps364=.31,eps365=2.68,eps366=.96)
inits<-list(inits1, inits2)
```

The bugs command below is a function in the package R2WinBUGS. The function calls the WinBUGS program to perform the MCMC analysis. The "date" commands keep track of the starting and ending time of the simulation. We can thus calculate the total time required for the analysis. We have specified a burn-in of 5,000 followed by 300,000 additional updates, where we save every 30th iteration. Thus the results are based on 10,000 usable iterations from each chain.

```
date()
fit <- bugs(data, inits, parameters, model.file = "McBugs.txt",
       n.chains = 2, n.iter = 450000, n.burnin = 150000,
       n.thin = 30,debug = FALSE, DIC = TRUE, digits = 5, codaPkg = FALSE)
date()
fit
```

The sampled values for each chain are now stored in a text file in the working directory. The file "codaIndex.txt" provides an index for the coda associated with each parameter. The MCMC output may now be read into R for further analysis. To monitor convergence, we compute the Gleman-Rubin statistic, and look at trace plots and autocorrelation plots.

```
# Read coda into R:
results.coda1 <- read.coda("coda1.txt","codaIndex.txt")
results.coda2 <- read.coda("coda2.txt","codaIndex.txt")
coda.all <- mcmc.list(results.coda1,results.coda2)

gelman.diag(coda.all)
traceplot(coda.all)
```

The following function has been modified from the auto.plot function in R. Changes are made to the plotting area.

```
auto.plot.new <- function (x, lag.max, auto.layout = TRUE, ask = dev.interactive(),
    ...)
{
    oldpar <- NULL
    on.exit(par(oldpar))
    if (!is.mcmc.list(x))
        x <- mcmc.list(x)
    for (i in 1:nchain(x)) {
        xacf <- if (missing(lag.max))
            acf(as.ts.mcmc(x[[i]]), plot = FALSE)
        else acf(as.ts.mcmc(x[[i]]), lag.max = lag.max, plot = FALSE)
        for (j in 1:nvar(x)) {
            plot(xacf$lag[, j, j], xacf$acf[, j, j], type = "h",
                ylab = "Autocorrelation", xlab = "Lag", ylim = c(-.2,
                  1), ...)
            title(paste(varnames(x)[j], ifelse(is.null(chanames(x)),
                "", ":"), chanames(x)[i], sep = ""))
            if (i == 1 && j == 1)
                oldpar <- c(oldpar, par(ask = ask))
        }
    }
    invisible(x)
}
auto.plot.new(results.coda1[, 2])       # autocorrelation plot for beta1.
```

The following section gives code for a sensitivity analysis on the upper bound of the uniform distribution on the standard deviation of the random effects. We create a sequence of values for UB and run the analysis for each value. After each run, we save the posterior 2.5th percentile, median, and 97.5th percentile of the parameter rho. These values are saved in the matrix "rho" and plotted against the corresponding values of UB.

```
upperBound <- seq(3, 8, by=1)      # the upper bound for the uniform prior on sigma

rho <- matrix(NA, length(upperBound), 4)
names <- list("B","2.5th","mdn","97.5th"); dimnames(rho)[[2]] <- names
```

```
date()
for(k in 1:length(upperBound)){
UB <- upperBound[k]

fit <- bugs(data, inits, parameters, model.file = "McBugs.txt",
        n.chains = 2, n.iter = 750000, n.burnin = 250000,
        n.thin = 50, debug = FALSE, DIC = TRUE, digits = 5, codaPkg = FALSE)
rho[k,1] <- UB
rho[k,2] <- fit$summary[9,3]
rho[k,3] <- fit$summary[9,5]
rho[k,4] <- fit$summary[9,7]
}
date()
```

The following lines of code plot the posterior values as a function of the upper bound. The three plots are overlaid.

```
par(mfrow=c(1,1))
plot(rho[,1],rho[,4],type="l",xlab="upper bound",ylim=c(0,3.8),ylab="",lty=2)
par(new=T)
plot(rho[,1],rho[,3],type="l",xlab="upper bound",ylim=c(0,3.8),ylab="",lty=1)
par(new=T)
plot(rho[,1],rho[,2],type="l",xlab="upper bound",ylim=c(0,3.8),ylab="",lty=3)
par(new=T)
plot(rho[,1],rho[,4],type="p",xlab="upper bound",ylim=c(0,3.8),ylab="")
par(new=T)
plot(rho[,1],rho[,3],type="p",xlab="upper bound",ylim=c(0,3.8),ylab="")
par(new=T)
plot(rho[,1],rho[,2],type="p",xlab="upper bound",ylim=c(0,3.8),ylab="")
```

A.3 WinBUGS Code for the Mixed Model

The following program contains the WinBUGS model and should be saved as a text file in the working directory under the name "McBugs.txt." The model here is model (2.1).

The first section of code sets up the likelihood. The responses are from a binomial distribution with parameters p[i] and n[i], where p[i] is the probability of an observed success. The probability of true success, pi[i], is modeled with logistic regression. To run the fixed effects analysis, remove eps[i] from the regression equation.

```
model {
 for (i in 1:N) {
  p[i] <- pi[i]*eta + (1 - pi[i])*(1 - theta)
  y[i] ~ dbin(p[i], n[i])
  eps[i] ~ dnorm(0, rho)
```

```
  logit(pi[i]) <- beta[1] + beta[2]*x2[i] + beta[3]*x3[i]
                + beta[4]*x4[i] + beta[5]*x5[i] + beta[6]*x6[i] + eps[i]
}
```

The precision, rho is defined as the inverse of the variance.

```
rho<-1/(sigma*sigma)
```

Now we specify prior distributions for all parameters. These priors were given in McInturff et al. (2004). The priors for pi361 through pi366 and eps361 through eps366 are used to induce the priors for the regression coefficients. These are the CMP priors described in Section 2.1.1. To run the fixed effects model, remove the priors for sigma and eps361 through eps366.

```
sigma ~ dunif(.1,UB)
eta ~ dbeta(99, 1)
theta ~ dbeta(14, 2)

pi361 ~ dbeta(8, 15)
pi362 ~ dbeta(10, 15)
pi363 ~ dbeta(3, 13)
pi364 ~ dbeta(8, 10)
pi365 ~ dbeta(4, 15)
pi366 ~ dbeta(6, 15)
eps361 ~ dnorm(0, rho)
eps362 ~ dnorm(0, rho)
eps363 ~ dnorm(0, rho)
eps364 ~ dnorm(0, rho)
eps365 ~ dnorm(0, rho)
eps366 ~ dnorm(0, rho)
```

The following section of code defines the CMP for the regression coefficients under the mixed model. For the fixed effects model, remove eps361 through eps366.

```
beta[1] <- xtili[1,1]*(logit(pi361)-eps361) + xtili[1,2]*(logit(pi362)-eps362)
          + xtili[1,3]*(logit(pi363)-eps363) + xtili[1,4]*(logit(pi364)-eps364)
          + xtili[1,5]*(logit(pi365)-eps365) + xtili[1,6]*(logit(pi366)-eps366)
beta[2] <- xtili[2,1]*(logit(pi361)-eps361) + xtili[2,2]*(logit(pi362)-eps362)
          + xtili[2,3]*(logit(pi363)-eps363) + xtili[2,4]*(logit(pi364)-eps364)
          + xtili[2,5]*(logit(pi365)-eps365) + xtili[2,6]*(logit(pi366)-eps366)
beta[3] <- xtili[3,1]*(logit(pi361)-eps361) + xtili[3,2]*(logit(pi362)-eps362)
          + xtili[3,3]*(logit(pi363)-eps363) + xtili[3,4]*(logit(pi364)-eps364)
          + xtili[3,5]*(logit(pi365)-eps365) + xtili[3,6]*(logit(pi366)-eps366)
beta[4] <- xtili[4,1]*(logit(pi361)-eps361) + xtili[4,2]*(logit(pi362)-eps362)
          + xtili[4,3]*(logit(pi363)-eps363) + xtili[4,4]*(logit(pi364)-eps364)
          + xtili[4,5]*(logit(pi365)-eps365) + xtili[4,6]*(logit(pi366)-eps366)
beta[5] <- xtili[5,1]*(logit(pi361)-eps361) + xtili[5,2]*(logit(pi362)-eps362)
```

```
          + xtili[5,3]*(logit(pi363)-eps363) + xtili[5,4]*(logit(pi364)-eps364)
          + xtili[5,5]*(logit(pi365)-eps365) + xtili[5,6]*(logit(pi366)-eps366)
beta[6] <- xtili[6,1]*(logit(pi361)-eps361) + xtili[6,2]*(logit(pi362)-eps362)
          + xtili[6,3]*(logit(pi363)-eps363) + xtili[6,4]*(logit(pi364)-eps364)
          + xtili[6,5]*(logit(pi365)-eps365) + xtili[6,6]*(logit(pi366)-eps366)

}
```

A.4 WinBUGS Code for the Hierarchically Centered Model

This program gives the WinBUGS model for the hierarchically centered model (2.4).

First, we define b[i] = μ[i] + ε[i]. Then the b[i] are normally distributed with mean mu[i] and precision rho. All prior distributions are the same as those given in the WinBUGS model in Section A.2.

```
model {
 for (i in 1:N) {
  p[i] <- pi[i]*eta + (1 - pi[i])*(1 - theta)
  y[i] ~ dbin(p[i], n[i])
  b[i] ~ dnorm(mu[i], rho)
  mu[i] <- beta[1] + beta[2]*x2[i] + beta[3]*x3[i]
          + beta[4]*x4[i] + beta[5]*x5[i] + beta[6]*x6[i]
  logit(pi[i]) <- b[i]
}
```

A.5 Program to compute DIC in R

This program can be used to calculate the Deviance Information Criterion (DIC) outside of WinBUGS. This is particularly useful when it is desirable to thin the chain after all updates have been completed. This code could be modified to calculate DIC using the median rather than the mean.

First, we set the working directory to the folder where the MCMC output is stored. We must load the coda package to analyze the output.

```
setwd("E:/Research/R dir")
library(coda)
```

Here we read the MCMC output from each chain into R. The file "codaIndex.txt" gives the index for the sampled values for each parameter. The thin statement is used to specify the desired amount of thinning.

```
results.coda1 <- read.coda("coda1.txt","codaIndex.txt", thin=10)
```

```
results.coda2 <- read.coda("coda2.txt","codaIndex.txt", thin=10)
```

Now the coda is stored in two large matrices: one for each chain. In the following lines, we merge the sampled values from the two chains and create a separate vector or matrix (as appropriate) for each parameter.

```
logit.pi <- rbind(results.coda1[,1:17],results.coda2[,1:17])
pi.hat <- exp(logit.pi)/(1+exp(logit.pi))
eta.vec <- c(results.coda1[,25],results.coda2[,25])
ones <- matrix(1,length(eta.vec),N)
eta.mat <- eta.vec*ones
theta.vec <- c(results.coda1[,27],results.coda2[,27])
theta.mat <- theta.vec*ones
p.hat <- pi.hat*eta.mat + (1-pi.hat)*(1-theta.mat)
```

Here, we define the deviance for our problem and calculate the mean posterior deviance.

```
log.nCy <- rep(sum(log(choose(n,y))),length(eta.vec))
d.vec <- (-2)*(log(p.hat)%*%y + log(1 - p.hat)%*%(n - y) + log.nCy)
Dbar <- mean(d.vec)
```

Now we calculate the posterior mean for all parameters.

```
theta.bar <- mean(theta.vec)
eta.bar <- mean(eta.vec)
pi.bar.logit <- colMeans(logit.pi)
pi.bar <- exp(pi.bar.logit)/(1+exp(pi.bar.logit))
p.bar <- pi.bar*eta.bar + (1-pi.bar)*(1-theta.bar)
```

Dhat calculates the deviance using the posterior means computed above. The effective number of parameters is given by pD. Finally, DIC is the sum of Dbar and pD.

```
Dhat <- (-2)*(log(p.bar)%*%y + log(1-p.bar)%*%(n-y) + sum(log(choose(n, y))))
Dhat <- drop(Dhat)
pD <- Dbar - Dhat
DIC <- Dbar + pD

Dbar;Dhat;pD;DIC
```

APPENDIX B

Chapter Three Programs

B.1 R Code to Analyze the Life Span Study (LSS) Data

The following R program is used to analyze the Life Span Study data discussed in Section 3.3. The program calls WinBUGS to perform the MCMC analysis. The output from this program is shown in Section 3.3.

We first set the working directory in R. The WinBUGS model should be saved as a text file in this folder under the name "AbombBugs.txt."

```
setwd("E:/Research/R dir")

library(R2WinBUGS); library(coda)
```

The following commands input the Life Span Study data and format the data for use by WinBUGS.

```
ystar <- c(2784, 2105, 439, 523, 586, 339, 204, 57, 21, 13)
xstar <- c(.0001, .018, .072, .137, .324, .693, 1.35, 2.35, 3.52, 4.43)
n <- c(12985, 9556, 1948, 2224, 2371, 1165, 573, 143, 72, 36)
J <- 10

# Data for WinBUGS:
data<-list("y","z","n","J")
```

Next we give initial values for each unknown in the model. Different initial values are specified for each chain then compiled into a single list.

```
inits.X <- rep(0,10)
inits1<-list(beta=c(-.6,.1), x=inits.X, eta=.65, theta=.9,c=.5)
inits2<-list(beta=c(-1.5,1), x=inits.X, eta=.9, theta=.97,c=.7)
inits<-list(inits1, inits2)
```

We specify the parameters of interest:

```
parameters <- list("beta","eta","theta","c")
```

We use two chains, with a burn-in of 5,000 updates and an additional 10,000 updates per chain. We retain only every 10^{th} update. The date commands (before and after the bugs statement) allow us to keep track of the total running time for the analysis.

```
date()
fit <- bugs(data, inits, parameters, model.file = "AbombBugs.txt",
      n.chains = 2, n.iter = 150000, n.burnin = 50000,
      n.thin = 10,debug = FALSE, DIC = TRUE, digits = 5, codaPkg = FALSE)
date()
fit
```

B.2 WinBUGS Code for the Life Span Study Data

The following program is the WinBUGS model and should be saved in the working directory under the name "AbombBugs.txt."

```
model{
```

The following loop sets up the likelihood structure. The observed responses have binomial distributions with success probabilities given by p[j].

```
 for(j in 1:J){
  # outcome model:
   p[j] <- pi[j]*eta + (1 - pi[j])*(1 - theta)
   ystar[j] ~ dbin(p[j], n[j])
   logit(pi[j]) <- beta[1] + beta[2]*x[j]
  # Berkson measurement model:
   x[j] ~ dnorm(xstar[j], tau[j])
```

WinBUGS parameterizes the normal distribution with the precision rather than variance. The precision of the Berkson error model is given by tau[j].

```
   tau[j] <- 1/(c*xstar[j]*xstar[j])
  }
```

Finally, we specify the prior structure for all unknowns in the model. The normal priors for the regression coefficients are "uninformative" because they are essentially flat. The sensitivity and specificity have priors with an effective sample size of 100.

```
 # priors:
 beta[1] ~ dnorm(0, .01)
 beta[2] ~ dnorm(0, .01)
 eta ~ dbeta(78, 22)
 theta ~dbeta(96.5, 3.5)
 c ~ dexp(1)
}
```

B.3 R Code for the Simulation Study

The following R code was used to create the simulation study in Section 3.4. The results of the simulation study are shown in Tables 12 and 13.

First we set the working directory and load the required packages for linking WinBUGS.

```
setwd("E:/Research/R dir")

library(R2WinBUGS); library(coda)
```

For each combination of the sensitivity and specificity (given in Section 3.4), we create a total of 300 data sets, as specified by n.reps. There are n = 25 values of the covariate.

```
n.reps <- 300
n <- 25
```

Next, we specify the true values of the parameters. The values of the sensitivity and specificity varied for each run of the simulation. We considered a total of nine combinations of eta and theta.

```
# Actual parameter values:
sigma <- sqrt(.3)
beta <- c(0, 1)
eta <- 0.8
theta <- 0.9
```

The following section of code was used to generate the data. The true covariate, x, is simulated from a Berkson model using values of the surrogate, xstar. The surrogate is simulated from Uniform(–3, 3). The probability of observing a success is given by p. The responses y are simulated from a binomial distribution with parameters 100 and p. The data is stored externally in the text file "SimData.txt."

```
# Create storage matrices:
xstar <- matrix(NA, n.reps, n)
x <- matrix(NA, n.reps, n)
pie.logit <- matrix(NA, n.reps, n)
ystar <- matrix(NA, n.reps, n)

# Generate the data:
for(i in 1:n.reps){
  xstar[i,] <- runif(n,min=-3, max=3)                  # xstar is surrogate
  for(h in 1:n){
   x[i,h] <- rnorm(1, mean=xstar[i,h], sd=sigma)       # x is true dose
   pie.logit[i,] <- beta[1] + beta[2]*x[i,]
  }
```

```
}
pie <- exp(pie.logit) / (1+exp(pie.logit))          # pie = P(Y = 1 | X)
p <- eta*pie + (1 - theta)*(1 - pie)                # p = P(Y* = 1 | X)
for(i in 1:n.reps){
 for(h in 1:n){
  ystar[i,h] <- rbinom(1, 100, p[i,h])
}}
simdata <- cbind(xstar, ystar)

# Write the data to an external file:
write.table(simdata,file="SimData.txt", row.names=F, col.names=F, append=F)
```

The following section of code analyzes the data using the model that accounts for both misclassification and measurement error. Slight modifications to this code could be made to analyze the data under the naïve analysis.

```
# Read in the data:
seq <- 1:n
simdata <- read.table(file="SimData1.txt", header=F)
xstar.mat <- as.matrix(simdata[,seq]); ystar.mat <- as.matrix(simdata[,-seq])
```

Now we put the data in the correct format for WinBUGS. We identify the parameters of interest and give starting values for each parameter in the two Markov Chains.

```
# Data for WinBUGS:
data<-list("ystar","xstar","n")

# Parameters to monitor:
parameters<-list("beta","eta","theta","sigma")

# Initial values:
inits.X <- rep(0, n)
inits1<-list(beta=c(0, 1),x=inits.X, eta=.8, theta=.9, tau=2)
inits2<-list(beta=c(0, 1),x=inits.X, eta=.8, theta=.9, tau=2)
inits<-list(inits1, inits2)
```

Before beginning the analysis, we must create matrices to store the results and give appropriate labels for reference.

```
# Create storage matrices for the results:
post.means <- matrix(NA,n.reps,5)
post.2.5 <- matrix(NA,n.reps,5)
post.97.5 <- matrix(NA,n.reps,5)
names <- list("beta1","beta2","eta","theta","sigma")
dimnames(post.means)[[2]] <- names
dimnames(post.2.5)[[2]] <- list("b1.2.5","b2.2.5","eta2.5","theta2.5","sig2.5")
```

```
dimnames(post.97.5)[[2]] <- list("b1.97.5","b2.97.5","eta97.5","theta97.5","sig97.5")
```

The analysis of the simulated data is carried out using a loop. The data has been stored in a matrix, and we use the start and end commands to step through the data sets. These commands may be used for computational efficiency, so that we may analyze the data in pieces or on different machines.

```
# ------------------------------ START ------------------------------
# Specify the data sets to use:
start <- 1; end <- 300

date()
for(k in start:end){

# Step through the data:
xstar <- xstar.mat[k,]; ystar <- ystar.mat[k,]
```

The bugs command sends the data to WinBUGS for the MCMC analysis. We specify and burn-in period of 5,000 followed by 100,000 additional updates, where every 10^{th} update is stored. The amount of thinning is given by n.thin.

```
# Call WinBUGS:
fit<- bugs(data, inits, parameters, model.file = "SimBugs.txt",
       n.chains = 2, n.iter = 130000, n.burnin = 50000,
       n.thin = 10,debug = FALSE, DIC = TRUE, digits = 5, codaPkg = FALSE)
```

For each data set, we record the posterior means, 2.5^{th} percentiles, and 97.5^{th} percentiles.

```
# Record the results:
post.means[k,] <- fit$summary[1:5,1]
post.2.5[k,] <- fit$summary[1:5,3]
post.97.5[k,] <- fit$summary[1:5,7]

}
date()
# ------------------------------ END ------------------------------
```

The results of the simulation study are saved in an external file, and may be read back into R for further analysis.

```
# Write results to an external file:
results <- cbind(post.means, post.2.5, post.97.5)
write.table(results, file="results D5.txt", row.names=F, append=F)

# Read results into R:
results <- read.table(file="results D5.txt", header=T)
post.means <- results[,1:5]; post.2.5 <- results[,6:10]; post.97.5 <- results[,11:15]
```

```
names <- list("beta1","beta2","eta","theta","sigma")
```

To assess the performance of the naïve model, we compute the average posterior mean for each case and compare to the truth. We also compute the coverage as the proportion of times that the actual value of the parameter fell into the 95% credible set. The width of the credible set is also calculated.

```
# Average estimates:
colMeans(post.means)

# Compute coverage:
cover <- matrix(NA,n.reps,5)
dimnames(cover)[[2]] <- names
truth <- c(beta,eta,theta,sigma)
for(k in 1:n.reps){
 for(i in 1:5){
  cover[k,i] <- ifelse(post.2.5[k,i] < truth[i] &
    truth[i] < post.97.5[k,i],1,0)
 }
}
coverage <- colMeans(cover)
coverage

# Width of credible sets:
width <-post.97.5 - post.2.5
dimnames(width)[[2]] <- names
avg.width <- colMeans(width)
avg.width
```

B.4 WinBUGS code for the Simulation Study

The following code should be saved as "SimBugs.txt" in the R working directory. This file gives the WinBUGS model.

```
model{
  for(j in 1:n){
    # response model:
    # p[j] <- pi[j]*eta + (1 - pi[j])*(1 - theta)
     ystar[j] ~ dbin(p[j], 100)
     logit(p[j]) <- beta[1] + beta[2]*xstar[j]
    # logit(pi[j]) <- beta[1] + beta[2]*x[j]
    # Berkson measurement model:
    # x[j] ~ dnorm(xstar[j], tau)
    }
```

Here we specify prior distributions for all unknowns. When analyzing the data under the naïve model, we do not need to give prior distributions for eta, theta, or tau.

```
  # priors:
  beta[1] ~ dnorm(0, .01)
  beta[2] ~ dnorm(0, .01)
  eta ~ dbeta(20,5)
  theta ~dbeta(22.5,2.5)
  tau ~ dgamma(33, 9)
```

Finally, we give the relationship between the precision and the standard deviation.

```
 sigma <- 1/sqrt(tau)
}
```

APPENDIX C

Chapter Four Programs

C.1 R Code to Simulate and Analyze the Chapter Four Data

The following program generates data that has a misclassified response, covariate measurement error, and random effects. The data is then analyzed using the model presented in Section 4.2. Output from this program is given in Section 4.4

First, we set the working directory in R and load the appropriate packages.

```
setwd("E:/Research/R dir")

library(R2WinBUGS); library(coda)
```

Next, we specify the sample size. We simulate 25 values for the covariate, and observe 50 responses for each value of x.

```
J<-25                                   # J=number of x values
n <- 50                                 # n=sample size for each x
```

Next, we give true values for the parameters. The values for the regression coefficients were motivated by the Life Span Study data.

```
# Actual parameter values:
sigB <- .5
beta <- c(-1.114, 0.4667)
eta <- 0.78
theta <- 0.965
tau <- 4
sigEps <- 1/sqrt(tau)

truth <- c(beta, eta, theta, c, sigma)
```

In the following section of code, we simulate the data. First, we create matrices to store the values. We create a grid of values for the surrogate xstar. These values are evenly spaced on the interval from 0 to 5. The true values of the covariate are simulated from a normal Berkson model with mean xstar and standard deviation sigB. Random effects are simulated from N(0, sigEps). The observed response y is simulated from a binomial distribution with parameters 50 and p.

```
# Create storage matrices:
x <- vector(mode="numeric", length=J)
pie.logit <- vector(mode="numeric", length=J)
ystar <- vector(mode="numeric", length=J)

# Generate the data:
xstar <- seq(.01, 5, .2)                    # xstar is surrogate
eps <- rnorm(J,mean=0, sd=sigEps)           # eps are random effects
for(i in 1:J){
  x[i]<-rnorm(1,mean=xstar[i],sd=sigB)      # x is true dose
  pie.logit[i] <- beta[1] + beta[2]*x[i] + eps[i]
}
pie <- exp(pie.logit)/(1+exp(pie.logit))    # pie = P(Y = 1 | X)
p <- eta*pie + (1-theta)*(1-pie)            # p = P(Y* = 1 | X)
for(i in 1:J){
  ystar[i] <- rbinom(1,n,p[i])
}
```

Now we put the data in the correct formal for WinBUGS. The upper bound on the uniform prior for sigma is given by B. We identify the parameters to be monitored and give appropriate starting values for the two chains.

```
# Data for WinBUGS:
B <- 4
data<-list("ystar","xstar","J","n","B")

# Parameters to monitor:
parameters<-list("beta","eta","theta","sigB","sigEps")

# Initial values:
inits.X <- rep(0,J); inits.eps <- rep(0,J)
inits1<-list(beta=c(-.6,.1),x=inits.X,eps=inits.eps,eta=.78,theta=.9,sigB=.5,sigEps=.15)
inits2<-list(beta=c(-1.5,1),x=inits.X,eps=inits.eps,eta=.8,theta=.96,sigB=.5,sigEps=3)
inits<-list(inits1,inits2)
```

Here we use the bugs command to call WinBUGS and perform the MCMC analysis. We use a burn-in of 5,000 followed by 500,000 updates, where we retained every 50^{th} update. Thus the results are based on 10,000 samples from each chain.

```
# Call WinBUGS:
date()
fit<- bugs(data, inits, parameters, model.file = "Ch4Bugs.txt",
       n.chains = 2, n.iter = 750000, n.burnin = 250000,
       n.thin = 50,debug = FALSE, DIC = TRUE, digits = 5, codaPkg = FALSE)
date()
```

C.2 WinBUGS Code for the Simulated Data

The following code should be saved in the working directory as "Ch4Bugs.txt."

The model statement sets up the likelihood in WinBUGS. Here we give the response model and the measurement error model. The probability of success is modeled using logistic regression.

```
model{
  for(j in 1:J){
   # response model:
     p[j] <- pi[j]*eta + (1-pi[j])*(1-theta)
     ystar[j] ~ dbin(p[j],n)
     eps[j] ~ dnorm(0,rho)
     logit(pi[j]) <- beta[1] + beta[2]*x[j] + eps[j]
   # Berkson measurement model:
     x[j] ~ dnorm(xstar[j],tau)
    }
  rho <- 1/(sigEps*sigEps)
  tau <- 1/(sigB*sigB)
```

Finally, we specify prior distributions for all parameters in the model. Informative priors are specified for the sensitivity and specificity, while diffuse normal priors are used for the regression coefficients.

```
 # priors:
  beta[1] ~ dnorm(0,.01)
  beta[2] ~ dnorm(0,.01)
  eta ~ dbeta(39,11)
  theta ~dbeta(48.25,1.75)
  sigB ~ dgamma(2,4)
  sigEps ~ dunif(.1,B)
}
```

REFERENCES

Agresti, A. (2002), *Categorical Data Analysis* (2nd ed.), New York: John Wiley.

Armstrong, B. G. (1998), "Effect of Measurement Error on Epidemiological and Occupational Exposures," *Occupational and Environmental Medicine*, 55, 651-656.

Bedrick, E. J., Christensen, R., and Johnson, W. O. (1996), "A New Perspective on Priors for Generalized Linear Models," *Journal of the American Statistical Association*, 91, 1450-1460.

Berkson, J. (1950), "Are There Two Regressions?" *Journal of the American Statistical Association*, 45, 164-180.

Carroll, R. J., Ruppert, D., Stefanski, L. A., and Crainiceanu, C. M. (2006), *Measurement Error in Nonlinear Models: A Modern Perspective* (2nd ed.), New York: Chapman and Hall.

Congdon, P. (2003), *Applied Bayesian Modelling*, Chichester: John Wiley & Sons.

——— (2005), *Bayesian Models for Categorical Data*, Chichester: John Wiley & Sons.

Cook, J. R. and Stefanski, L. A. (1994), "Simulation-Extrapolation Estimation in Parametric Measurement Error Models," *Journal of the American Statistical Association*, 83, 596-610.

Gelfand, A. E. and Ghosh, S. K. (1998), "Model Choice: A Minimum Posterior Predictive Loss Approach," *Biometrika*, 85, 1-11.

Gelfand, A. E., Sahu, S. K., and Carlin, B. P. (1995a), "Efficient Parameterizations for Generalized Linear Mixed Models," in *Bayesian Statistics*, eds. J. M. Bernardo, J. Berger, A. P. Dawid, and A. F. M. Smith, Oxford: Oxford University.

——— (1995b), "Efficient Parameterizations for Normal Linear Mixed Models," *Biometrika,* 82, 479-488.

Gelman, A. (2006), "Prior Distributions for Variance Parameters in Hierarchical Models," *Bayesian Analysis*, 1, 515-533.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003), *Bayesian Data Analysis* (2nd ed.), London: Chapman and Hall.

Gelman, A. and Rubin, D. B. (1992), "Inference From Iterative Simulation Using Multiple Sequences" (with discussion), *Statistical Science*, 7, 457-511.

Ghosh, S.K. and Norris, J.L. (2005), "Bayesian Capture-Recapture Analysis and Model Selection Allowing for Heterogeneity and Behavioral Effects," *Journal of Agricultural, Biological, and Environmental Statistics*, 10, 35-49.

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

Gustafson, P. (2003). *Measurement Error and Misclassification in Statistics and Epidemiology: Impacts and Bayesian Adjustments*, Boca Raton: Chapman and Hall/CRC.

Kim, H.-Y., Yasui, Y., and Burstyn, I. (2006), "Attenuation in Risk Estimates in Logistic and Cox Proportional-Hazards due to Group-based Exposure Assessment Strategy," *Annals of Occupational Hygiene*, 50, 623-635.

Luan, X., Pan, W., Gerberich, S. G., and Carlin, B. P. (2005), "Does it Always Help to Adjust for Misclassification of a Binary Outcome in Logistic Regression?" *Statistics in Medicine*, 24, 2221-2234.

McCullagh, P. and Nelder, J. A. (1989), Generalized Linear Models (2^{nd} ed.), London: Chapman and Hall.

McInturff, P., Johnson, W. O., Cowling, D., and Gardner, I. A. (2003), "Modelling Risk When Binary Outcomes are Subject to Error," *Statistics in Medicine*, 23, 1095-1109.

Neuhaus, J. M. (1999), "Bias and Efficiency Loss due to Misclassified Responses in Binary Regression," *Biometrika*, 86, 843-855.

Paulino, C. D., Silva, G., and Achcar, J. A. (2005), "Bayesian Analysis of Correlated Misclassified Binary Data," *Computational Statistics and Data Analysis*, 49, 1120-131.

Reeves, G. K., Cox, D. R., Darby, C., and Whitley, E. (1998), "Some Aspects of Measurement Error in Explanatory Variables for Continuous and Binary Regression Models," *Statistics in Medicine*, 17, 2157-2177.

Rosner, B., Spiegelman, D., and Willett, W. C. (1990), "Correction of Logistic Regression Relative Risk Estimates and Confidence Intervals for Measurement Error: The Case of Multiple Covariates Measured With Error," *American Journal of Epidemiology*, 132, 734-745.

Roy, S. Banerjee, T., and Maiti, T. (2005), "Measurement Error Model for Misclassified Binary Responses," *Statistics in Medicine*, 24, 269-283.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and van der Linde, A. (2002), "Bayesian Measures of Model Complexity and Fit" (with discussion and rejoinder), *Journal of the Royal Statistical Society*, Ser. B, 64, 583-639.

Spiegelhalter, D. J., Thomas, A., Best, N. G., and Lunn, D. (2001), *WinBUGS User Manual, Version 1.4*, available at *http://www.mrc-bsu.cam.ac.uk/bugs*.

Sposto, R., Preson, D. L., Shimizu, Y., and Mabuchi, K. (1992), "The Effect of Diagnostic Misclassification on Non-cancer and Cancer Mortality Dose Response in A-bomb Survivors," *Biometrics*, 48, 605-617.

Tuley, M. R. (1990), "Double-Sampling Approach for Logistic Regression With Misclassification," Ph.D. Dissertation, Baylor University, Department of Statistical Science.

Vines, S. K., Gilks, W. R., and Wild, P. (1996), "Fitting Bayesian Multiple Random Effects Models," *Statistics and Computing*, 6, 337-346.

Zhao, Y., Staudenmayer, J., Coull, B. A., and Wand, M. P. (2006), "General Design Bayesian Generalized Linear Mixed Models," *Statistical Science*, 21, 35-51.