ABSTRACT

Sample Size Determination for Two Sample Binomial and Poisson Data Models Based on Bayesian Decision Theory

Ryan A. Sides, Ph.D.

Chairpersons: James D. Stamey and David J. Kahle

Sample size determination continues to be an important research area in statistical analysis due to the cost and time constraints that often exist in areas such as pharmaceuticals and public health. We begin by outlining the work of a previous article that attempted to find a minimum necessary sample size in order to reach a desired expected power for binomial data under the Bayesian paradigm. We make improvements to their efforts that allow us to specify not only a desired expected Bayesian power, but also a more generic loss function and a desired expected Bayesian significance level, the latter having never been considered previously. We then extend these methodologies to handle Poisson data and discuss challenges in the methodology. We cover a detailed example in both cases and display various results of interest.

We conclude by covering a mixed treatment comparisons meta-analysis problem when analyzing Poisson data. Traditional methods do not allow for the presence of underreporting. Here, we illustrate how a constant underreporting rate for all treatments has no effect on relative risk comparisons; however, when this rate changes per treatment, not accounting for it can lead to serious errors. Our method allows this to be taken into account so that correct analyses can be made.

Sample Size Determination for Two Sample Binomial and Poisson Data Models Based on Bayesian Decision Theory

by

Ryan A. Sides, B.S.E., M.S.

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

David J. Kahle, Ph.D., Co-Chairperson

A. Alexander Beaujean, Ph.D.

Dennis A. Johnston, Ph.D.

Dean M. Young, Ph.D.

Accepted by the Graduate School
December 2013

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

I'd first like to thank my committee for their efforts in this project. Many thanks are due to Dr. Stamey for his guidance, ideas, and willingness to always be available to discuss whatever issue we were having at the time. I'm also grateful for the efforts of Dr. Kahle in making sure that we were clear in our language and that everything we did was mathematically sound.

I'd like to thank the Statistics department at Baylor. To the faculty: I appreciate your hard work not only in the classroom but outside as well. Having a department that always has their door open means a lot. To the students: I enjoyed having you as my classmates – those before me and those after me. I'm sure that I've helped many of you just as you've helped me. Make sure that continues with those behind you.

I'd also like to thank the Statistics department at Sam Houston State. Without your help in getting a Master's degree, none of this would be happening. Thank you for realizing my potential long before I knew about it and pushing me to greater heights.

Lastly, to my family and friends: I wouldn't be here without you. Thanks for always believing in me and sticking with me at all times. Most of you will probably never come across this, but I'm indebted to a few of you more than you will ever know.

CHAPTER ONE

Introduction

Sample size determination continues to be an important research area in statistical analysis due to the cost and time constraints that often exist in areas such as pharmaceuticals and public health. Too small of a sample can lead to under-powered studies and a waste of time and money, while too large of a sample size, while not only being a waste of those resources, can also make any difference in observation seem significant. Because of the numerous advantages to Bayesian sample size determination, we operate under the Bayesian paradigm. The main advantage that we aim to utilize is that we are able to characterize our uncertainty when assessing the parameter of interest. In the frequentist domain, a sample size determination problem requires the researcher to specify what he believes the parameter to be. Because this prior guess is unlikely to be exact, the degree to which the researcher is incorrect can create substantial problems with respect to the power of the test. The ability for the researcher to model his indecision about the parameter through expert knowledge or previous studies can allow the Bayesian approach to have better operating characteristics. However, it should always remain prevalent in the reader's mind that misspecification in the Bayesian context can prevent desirable operating characteristics.

One of the more basic sample size determination problems is when we consider the equivalence of two population proportions. We follow the basic framework outlined by Zhao et al. (2011) when introducing the problem and deriving a general solution; however, we improve their efforts by introducing the option of a loss function without equal losses, something that had not previously been done. This loss function allows us to directly specify the losses associated with Type I and Type

II errors, and eventually, the ability to control for expected Bayesian significance level.[1] As we continue with this problem, we introduce a method to control for both expected Bayesian significance level and expected Bayesian power while letting components of our loss function vary in order to minimize the necessary sample size needed if we wish to meet those operating characteristics. We conclude the chapter by detailing how to fix a loss function and still control for both types of expected Bayesian errors. For each technique we illustrate by using a detailed example and include various results of interest.

Next, we extend these methods to determine the sample size needed to properly assess the equivalence of two Poisson rates. While the construction used in the binomial case can be applied directly to yield a test in the Poisson case, the nature of Poisson data introduces unique features in the implementation stage. For example, whereas in the binomial case we can enumerate the entire possible sample space to produce exact results, this application is not possible in the Poisson case; therefore, only approximate solutions are available. We again follow the same basic framework as the previous chapter, outlining ways to control for various operating characteristics of interest while still covering a basic example and different interesting results.

Lastly, we change gears to consider the problem of handling a mixed treatment comparisons meta-analysis when analyzing underreported Poisson data. In recent years, mixed treatment comparisons meta-analysis has become a popular methodology because of its ability to use separate trials to make comparisons about parameters, even when parameters have not been directly compared. However, traditional methods do not allow for the presence of misspecified data. Here, we introduce a Markov chain Monte Carlo (MCMC) model that accounts for constant underreporting among treatments and illustrate how this form of underreporting has no effect

[1] Previous methods only allowed for controlling expected Bayesian power.

on relative risk comparisons. We then extend this idea to the case where each treatment has a different amount of underreporting present, first showing how neglection of this form of underreporting can lead to serious errors. Next, we introduce a new MCMC model that handles treatments that each have a different amount of underreporting and provides better results. The robustness of this method to changes various components is then illustrated by different simulations. Then we continue by analyzing data with incorrect prior structures in an attempt to show that misspecifying an underreporting probability prior is still preferred to analyzing with no underreporting. We conclude with a discussion about our MCMC assumptions.

CHAPTER TWO

Bayesian Sample Size Determination for Two Sample Binomial Experiments

2.1 Introduction

Sample size determination continues to be an important research area of statistics. Cost and time constraints have made finding the appropriate sample size before conducting a study of the utmost importance. Too small of a sample can lead to under-powered studies and a waste of time and money. Too large of a sample size, while not only being a waste of those resources, can also make any difference in observation seem significant. In this chapter, we aim to find the required sample size to meet a varying array of operating characteristics when testing the equality of two binomial rates from a Bayesian perspective.

The advantages to Bayesian sample size determination are plentiful and have been enumerated by Adcock (1997). Their construction does not depend on asymptotic approximations and allow for the characterization of uncertainty when assessing the parameter of interest. In the frequentist domain, a sample size determination problem requires the researcher to specify what he believes the parameter to be. However, as M'Lan et al. (2008) note, the parameter(s) will most likely never be known with high accuracy at the planning stage, such that the degree to which the researcher is incorrect can create substantial problems with respect to the power of the test and create doubt about the sample size estimated. The ability for the researcher to model his indecision about the parameter through expert knowledge or previous studies can allow the Bayesian approach to have better operating characteristics (such as a smaller required sample size or better Type I and II error rates) as noted by Bayarri and Berger (2004) among others. However, it should always remain prevalent in the reader's mind that misspecification in the Bayesian con-

text can prevent desirable operating characteristics just as it can in the frequentist context.[1]

Various Bayesian sample size determination methods have been previously studied. While there is no way to accurately assess them all, we cover a few specifically here. One of the bigger authorities in Bayesian sample size determination is Lawrence Joseph; of note, Joseph et al. (1995) looked at single binomial proportions before Joseph et al. (1997) adapted the methodology to two sample binomial proportions. However, both of these focus on interval based criteria such as coverage and width. De Santis and Perone Pacifico (2003) and De Santis et al. (2004) extend those ideas to consider both interval-based and test-based criteria; M'Lan et al. (2006) and M'Lan et al. (2008) do likewise. However, all of these efforts are limited in their abilities with respect to test-based criteria; namely, none of them consider the expected power of the test that will be conducted. Katsis and Toman (1999) used more decision theoretic test-based criteria for the two sample binomial case, but only to the the extent that they aim to control the posterior risk with a prespecified bound. Zhao et al. (2011) extend on those ideas by using computational methods now available to consider expected Bayesian power of the test. In this chapter, we extend the results of Zhao et al. (2011) by accounting for both expected Type I and Type II error rates. This sort of sample size procedure for the comparison of two binomial rates has not been considered before in a Bayesian context.

Further, it should be noted that, due to the fact that we are in the Bayesian framework, references to Type I and Type II error rates, significance level, and power are meant in the Bayesian context. By this we mean that they are to be understood in an expected sense, as outlined by O'Hagan et al. (2005). For example, the expected power of a test is the power of the test when averaged over the likelihood

[1] Because Type I error rates are not controlled in the Bayesian paradigm as they are in the frequentist one, misspecification here leads to the possibility of either type of error being quite large.

of some other distribution that assesses the perceived likelihood of observing different possible parameter values.[2] However, when the test itself uses Bayesian methods, we refer to this as the expected Bayesian power (EBP), as is done in Speighelhalter et al. (2004).[3] Thus, we weight the true values of power by the prior distribution associated with the alternative hypothesis. These ideas, though not considered in the literature previously, can also apply to significance level.[4] In this chapter, we consider expected Bayesian significance level (EBSL), such that we weight true values of significance level by the prior distribution associated with the null hypothesis.[5]

This chapter is arranged as follows. In Section 2.2, we introduce the basic framework for the two sample binomial problem. Section 2.3 details how we arrive at the required sample size, introducing some decision theory not previously considered. Section 2.4 outlines the case of controlling for EBP while fixing a loss function; Section 2.5 does the same for controlling EBSL. Section 2.6 describes how we can control for both EBSL and EBP, finding the minimum necessary sample size required if we are not interested in specifying the loss in our loss function; Section 2.7 discusses the case where we are interested in controlling all three of these things.

2.2 Framework

We follow the general framework of Zhao et al. (2011) in the development of this problem. Suppose observations from two independent populations come from binomial distributions with parameters θ_1 and θ_2, respectively, and common sample size, n. Thus, we have $Y_1 \sim \text{binomial}\,(n, \theta_1)$ and $Y_2 \sim \text{binomial}\,(n, \theta_2)$. Our interest

[2] This has application to the frequentist paradigm, where we could consider this individual value rather than a power curve.

[3] Another common name for this is the probability of a successful test or trial.

[4] While frequentist methods typically report one value for significance level, what they are really doing (in non point null hypotheses) is taking the largest possible significance level; thus, taking an expectation of a significance level curve could be done as well.

[5] Of course this implies that actual Type I and Type II error rates can be greater than or less than what is indicated when considering expected error rates.

is to calculate the necessary sample size required to test the hypotheses

$$H_0 : \theta_1 = \theta_2$$

vs

$$H_1 : \theta_1 \neq \theta_2.$$

As a notational convenience, we will refer to the truth about which hypothesis is correct with φ. That is, when the null hypothesis that $\theta_1 = \theta_2$ is true, we will denote that by saying $\varphi = 0$; when the alternative hypothesis that $\theta_1 \neq \theta_2$ is true, we will denote that by saying $\varphi = 1$. In addition, we place prior probabilities of π_0 and $\pi_1 = 1 - \pi_0$ on H_0 and H_1, respectively; that is, $P(\varphi = 0) = \pi_0$ and $P(\varphi = 1) = \pi_1 = 1 - \pi_0$. This is to say that, in a practical sense when we are unsure about which hypothesis is true, we would set $\pi_0 = \pi_1 = 0.5$.

Also, we assume that the joint prior distribution of θ_1 and θ_2 is of the form

$$f(\theta_1, \theta_2) \;=\; \pi_0 p_0(\theta)\, I[\varphi = 0] + \pi_1 p_1(\theta_1, \theta_2)\, I[\varphi = 1],$$

where $I[\varphi = 0]$ and $I[\varphi = 1]$ are the indicator functions of H_0 and H_1, respectively, $p_0(\theta)$ is the prior distribution of θ under the assumption that $\theta = \theta_1 = \theta_2$, and $p_1(\theta_1, \theta_2)$ is the joint prior distribution of θ_1 and θ_2 when $\theta_1 \neq \theta_2$. Note that we will place prior distributions on θ, θ_1, and θ_2 that summarize our beliefs about them; this information will allow us to compute an estimate for the necessary sample size. For simplicity, we will consider conjugate priors for all three θ's, so that under H_0, $\theta \sim \text{beta}(\alpha, \beta)$ and under H_1, $\theta_i \sim \text{beta}(\alpha_i, \beta_i)$. Thus, we will model our beliefs about these parameters such that we think there is a π_0 probability of the null hypothesis being true, and when that is the case it can be summarized by a $\text{beta}(\alpha, \beta)$ distribution; further, there is a π_1 probability of the alternative hypothesis being true, and when that is the case the two distributions there can be summarized by $\text{beta}(\alpha_i, \beta_i)$ distributions.

2.3 Sample Size Determination

2.3.1 Decision Theory

Consider the loss function

$$L(\varphi, \delta) = \begin{cases} 0, & \text{if } \varphi = \delta, \\ c_1, & \text{if } \varphi = 0 \text{ and } \delta = 1, \\ c_2, & \text{if } \varphi = 1 \text{ and } \delta = 0, \end{cases}$$

where $\delta = 0$ represents choosing H_0, while $\delta = 1$ represents a choice of H_1. Thus, c_1 represents the loss due to a Type I error (rejecting a true null hypothesis) and c_2 represents the loss due to a Type II error (failing to reject a false null hypothesis). This represents an improvement to the methods of Zhao et al. (2011) who only used the $0-1$ loss function such that $c_1 = c_2$.

Thus, our risk can be expressed as

$$\begin{aligned} R(\delta) &= E[L(\varphi, \delta)] \\ &= \begin{cases} c_1 P(\varphi = 0 | \mathbf{Y} = \mathbf{y}), & \text{if } \delta = 1, \\ c_2 P(\varphi = 1 | \mathbf{Y} = \mathbf{y}), & \text{if } \delta = 0. \end{cases} \end{aligned}$$

However, we will define c as the ratio of c_1 to c_2, or rather, $c = c_1/c_2$. In other words, we can think of our loss function constant, c, as how much worse it is to make a Type I error than a Type II error. Thus, because our objective is to minimize risk, we can express this in terms of the optimal decision, δ^*, as

$$\delta^*(\mathbf{y}) = \begin{cases} 0, & \text{if } P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) < cP(\varphi = 0 | \mathbf{Y} = \mathbf{y}), \\ 1, & \text{if } P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) \geq cP(\varphi = 0 | \mathbf{Y} = \mathbf{y}). \end{cases}$$

Because the optimal decision is to reject the null hypothesis when the second inequality holds, this implies that our rejection region, W, for this loss function is

$$W = \{\mathbf{y} : P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) \geq cP(\varphi = 0 | \mathbf{Y} = \mathbf{y})\}. \tag{2.1}$$

Note that this follows the same basic notation used by Zhao et al. (2011) with the addition of the loss function constant, c.

2.3.2 Bayes Factors

The Bayes factor is defined as the ratio of the odds in favor of one hypothesis to the other. While this can be reported in both directions, we follow the form of Kass and Raftery (1995) in defining it as the odds in favor of the alternative hypothesis such that a large Bayes factor is evidence that we should reject the null hypothesis. Thus, this is written analytically as

$$\begin{aligned} B &= \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)/P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)}{\pi_1/\pi_0} \\ &= \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\pi_0}{P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\pi_1}. \end{aligned}$$

This ratio is useful in Bayesian inference because it is often interpretted as partially eliminating the influence of the prior on the posterior and, rather, emphasizing the role of the data. Again following the work of Zhao et al. (2011), it can be shown that our decision rule is a function of a Bayes factor because

$$\begin{aligned} W &= \left\{\mathbf{y} : P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right) \geq cP\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\right\} \\ &= \left\{\mathbf{y} : P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\frac{\pi_0}{\pi_1} \geq cP\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\frac{\pi_0}{\pi_1}\right\} \\ &= \left\{\mathbf{y} : \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\pi_0}{P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\pi_1} \geq c\frac{\pi_0}{\pi_1}\right\} \\ &= \left\{\mathbf{y} : B \geq c\frac{\pi_0}{\pi_1}\right\}. \end{aligned} \tag{2.2}$$

This is particularly useful because the Bayes factor, B, will be the test statistic for this hypothesis test. Further, the right side of Equation 2.2 becomes the decision rule for when this test statistic should be rejected. This allows us to have a specific decision rule rather than rely on traditional methods such as rejecting when this value exceeds some arbitrary number such as three.

2.3.3 Rejection Region

However, we can use Bayes' Theorem to show that

$$\begin{aligned} P(\varphi = 0|\mathbf{Y} = \mathbf{y}) &= \frac{P(\mathbf{Y} = \mathbf{y}, \varphi = 0)}{P(\mathbf{Y} = \mathbf{y})} \\ &= \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) P(\varphi = 0)}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) P(\varphi = 0) + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) P(\varphi = 1)} \\ &= \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0 + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1} \end{aligned} \tag{2.3}$$

and, using similar logic,

$$P(\varphi = 1|\mathbf{Y} = \mathbf{y}) = \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0 + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1}, \tag{2.4}$$

where $\mathbf{y} = (y_1, y_2)$. Further, it is seen that

$$\begin{aligned} P(\mathbf{Y} = \mathbf{y}|\varphi = 0) &= f(\mathbf{y}|\varphi = 0) \\ &= \int_0^1 f(\mathbf{y}, \theta|\varphi = 0) \, d\theta \\ &= \int_0^1 f(\mathbf{y}|\theta, \varphi = 0) \, p_0(\theta|\varphi = 0) \, d\theta \\ &= \int_0^1 \binom{n}{y_1} \theta^{y_1} (1-\theta)^{n-y_1} \binom{n}{y_2} \theta^{y_2} (1-\theta)^{n-y_2} \times \\ &\quad \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1} (1-\theta)^{\beta-1} \, d\theta \\ &= \binom{n}{y_1} \binom{n}{y_2} \frac{1}{B(\alpha, \beta)} \int_0^1 \theta^{y_1+y_2+\alpha-1} (1-\theta)^{2n-y_1-y_2+\beta-1} \, d\theta \\ &= \binom{n}{y_1} \binom{n}{y_2} \frac{1}{B(\alpha, \beta)} B(y_1 + y_2 + \alpha, 2n - y_1 - y_2 + \beta) \times \\ &\quad \int_0^1 \frac{1}{B(y_1 + y_2 + \alpha, 2n - y_1 - y_2 + \beta)} \times \\ &\quad \theta^{y_1+y_2+\alpha-1} (1-\theta)^{2n-y_1-y_2+\beta-1} \, d\theta \\ &= \binom{n}{y_1} \binom{n}{y_2} \frac{B(y_1 + y_2 + \alpha, 2n - y_1 - y_2 + \beta)}{B(\alpha, \beta)}, \end{aligned} \tag{2.5}$$

where $B(\alpha, \beta)$ is the beta function, or rather,

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha, \beta)}.$$

Extending the same logic to the density of $\mathbf{y}$ conditioned on the alternative hypothesis being true,

$$\begin{aligned}
P(\mathbf{Y} = \mathbf{y}|\varphi = 1) &= f(\mathbf{y}|\varphi = 1) \\
&= \int_0^1 f(\mathbf{y}, \boldsymbol{\theta}|\varphi = 1)\, d\boldsymbol{\theta} \\
&= \int_0^1 f(\mathbf{y}|\boldsymbol{\theta}, \varphi = 1)\, p_1(\boldsymbol{\theta}|\varphi = 1)\, d\boldsymbol{\theta} \\
&= \prod_{i=1}^{2} \int_0^1 \binom{n}{y_i} \theta_i^{y_i} (1-\theta_i)^{n-y_i} \frac{1}{B(\alpha_i, \beta_i)} \theta_i^{\alpha_i - 1} (1-\theta_i)^{\beta_i - 1}\, d\theta_i \\
&= \prod_{i=1}^{2} \binom{n}{y_i} \frac{1}{B(\alpha_i, \beta_i)} \int_0^1 \theta_i^{y_i + \alpha_i - 1} (1-\theta_i)^{n - y_i + \beta_i - 1}\, d\theta_i \\
&= \prod_{i=1}^{2} \binom{n}{y_i} \frac{1}{B(\alpha_i, \beta_i)} B(y_i + \alpha_i, n - y_i + \beta_i) \times \\
&\quad \int_0^1 \frac{1}{B(y_i + \alpha_i, n - y_i + \beta_i)} \theta_i^{y_i + \alpha_i - 1} (1-\theta_i)^{n - y_i + \beta_i - 1}\, d\theta_i \\
&= \prod_{i=1}^{2} \binom{n}{y_i} \frac{B(y_i + \alpha_i, n - y_i + \beta_i)}{B(\alpha_i, \beta_i)}, \qquad (2.6)
\end{aligned}$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2)$.

Thus, with this information we can return to our rejection region found in Equation 2.1 and, using Equations 2.3 and 2.4, show that

$$\begin{aligned}
W &= \{\mathbf{y} : P(\varphi = 1|\mathbf{Y} = \mathbf{y}) \geq cP(\varphi = 0|\mathbf{Y} = \mathbf{y})\} \\
&= \{\mathbf{y} : P(\mathbf{Y} = \mathbf{y}|\varphi = 1)\pi_1 \geq cP(\mathbf{Y} = \mathbf{y}|\varphi = 0)\pi_0\},
\end{aligned}$$

which is again similar to the work of Zhao et al. (2011). Further, we can use Equations 2.5 and 2.6 to show that our rejection region for this test consists of all the

points that satisfy Inequality 2.7. Note that the left side of 2.7 is our test statistic and Bayes factor, B. The right side remains our decision rule.

$$\frac{B(\alpha,\beta)}{B(y_1+y_2+\alpha,2n-y_1-y_2+\beta)}\prod_{i=1}^{2}\frac{B(y_i+\alpha_i,n-y_i+\beta_i)}{B(\alpha_i,\beta_i)} \geq c\frac{\pi_0}{\pi_1} \quad (2.7)$$

Next, we use this rejection region to find the required sample size needed to meet certain operating characteristics. In Section 2.4, we consider the case where we want to reach a given EBP for a fixed value of the loss function constant, c. We can then solve for n and find the EBSL. In Section 2.5, we control the EBSL for a fixed value of c, allowing us to solve for n and compute the EBP. Despite both of these cases taken independently seeming impractical in the real world, these applications are crucial for the development of the last two cases of interest. In Section 2.6, we consider the case where we are not concerned with the value of c and would like to reach a given EBP while still controlling the EBSL. We can do this by fixing the two desired operating characteristics and solving iteratively for both c and n. Lastly, in Section 2.7, we consider the case where we know our loss function constant, but still would like to be able to control for both EBSL and EBP while solving for n.

2.4 Finding Sample Size from a Specified EBP and Loss Function

2.4.1 General Algorithm

In this section, we aim to find a solution to the sample size determination problem when we specify our loss function constant and our desired EBP, $1-\beta_0$. We define a candidate solution as any value of n such that Inequality 2.8 holds for n but not $n-1$.

$$\begin{aligned} P(\mathbf{Y}\in W|\varphi=1) &= \sum_{\mathbf{y}\in W} P(\mathbf{Y}=\mathbf{y}|\varphi=1) \\ &= \sum_{\mathbf{y}\in W}\prod_{i=1}^{2}\binom{n}{y_i}\frac{B(y_i+\alpha_i,n-y_i+\beta_i)}{B(\alpha_i,\beta_i)} \\ &\geq 1-\beta_0. \end{aligned} \quad (2.8)$$

However, we are really interested in finding the optimal solution, n^*, defined as the smallest candidate solution such that Inequality 2.8 holds for all values larger than n^*. This is important because we are not guaranteed a nondecreasing EBP as our sample size increases due to the discrete nature of the problem.

Despite the lack of a formal proof, it is a very reasonable conjecture to make that n^* exists; if this is true, it is unique by definition. Our aim is to find this value; however, this has proven to be analytically intractable. Thus, we define n_N^* to be the smallest value of $n \leq N$ such that Inequality 2.8 holds for all values between n_N^* and N. This is clearly a candidate solution that is an approximation of n^*. It should be obvious that $n_N^* = n^*$ for a sufficiently large N, and further, that for increasing N, n_N^* approaches n^*. However, it should also be noted that there is no guarantee that n_N^* exists. Thus, if it does not, we simply report the first candidate solution greater than N that we find.[6] [7]

Note that we condition our data on the alternative hypothesis being true.[8] For a fixed sample size, n, we set up a matrix of all the possible data values that we could observe. Noting that the possible values of $\mathbf{y}$ extend from zero to n, we set up a predictive distribution matrix so that for any values of π_0, c and n, this matrix of all possible values of $\mathbf{y}$ looks like Equation 2.9.

$$P(\mathbf{Y} = \mathbf{y}|\varphi = 1) = \begin{bmatrix} P(\mathbf{Y} = (0,0)|\varphi = 1) & \cdots & P(\mathbf{Y} = (0,n)|\varphi = 1) \\ \vdots & \ddots & \vdots \\ P(\mathbf{Y} = (n,0)|\varphi = 1) & \cdots & P(\mathbf{Y} = (n,n)|\varphi = 1) \end{bmatrix} \quad (2.9)$$

Note that the form of these values is found in Equation 2.6, and that this matrix provides the probabilities of having $\mathbf{y}$ successes; summing all of these values will equal one. Algorithm 2.1 shows how we use this to find a required sample size.

[6] This is not necessarily the smallest candidate solution.

[7] This process is detailed in Algorithm 2.1.

[8] Hence, we use the prior structures specified such that $\theta_i \sim \text{beta}(\alpha_i, \beta_i)$.

Algorithm 2.1: When specifying the desired EBP, $1-\beta_0$, the loss function constant, c, and N, this algorithm finds the approximate solution, n_N^*, if it exists; if it does not, it finds a candidate solution greater than N

1: Fix $n = N$.
2: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 2.9; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 2.7). This is the EBP for the fixed value of n.
3: If the resulting EBP is greater than $1-\beta_0$, $n = n-1$; otherwise, skip to step 5.
4: Repeat steps 2 and 3 until the value of n provides an EBP less than $1-\beta_0$; n_N^* is the last value of n that did not cause the EBP to fall below $1-\beta_0$, and the algorithm ends.
5: n_N^* does not exist; fix $n_0 = N$ and $n_1 = 2N$.
6: Repeat step 2 for n_1.
7: If the resulting EBP of n_1 is less than $1-\beta_1$, $n_0 = n_1$ and $n_1 = 2n_1$; go to step 6.
8: Fix $n = (n_0 + n_1)/2$ and repeat step 2.
9: If the resulting EBP is greater than $1-\beta_0$, $n_1 = n$; else, $n_0 = n$.
10: Repeat steps 8 and 9 until $n_1 - n_0 = 1$; select n_1 as the candidate solution.

It should be noted that it becomes increasingly difficult to find the optimal solution as N gets large because the predictive matrix becomes exponentially large. Further, it should be noted that empirical evidence suggests that the likelihood of a substantial decrease in EBP occurring at larger sample sizes decreases as n increases. This is because EBP is only decreasing because of the discrete nature of the problem; namely, a sample size change of one occasionally affects the dimensions of our rejection region such that we successfully reject a false null hypothesis a smaller percentage of the time than previously. However, as the sample size increases, the changes in the rejection region become less extreme. Thus, it becomes improbable for us to see a large decrease in EBP as n increases. It is for these reasons that we default the algorithm at $N = 100$; however, this option can be changed by the user if desired.

2.4.2 Example

We focus this example on one of the cases in Zhao et al. (2011). The case we consider is the one where we place a beta$(1, 4)$ prior on θ_1, a beta$(3, 7)$ prior on θ_2,

and a beta$(1, 1)$ prior on θ when we assume that the true rates are actually the same. This is another way of saying that we are putting an uninformative uniform prior on θ. These prior distributions can be thought of as θ_1 having a success rate of 20% with a standard deviation of 16%, θ_2 having a success rate of 30% with a standard deviation of 14%, and θ having a success rate of 50% with a standard deviation of 29%. Figure 2.1 shows all of these priors graphically.

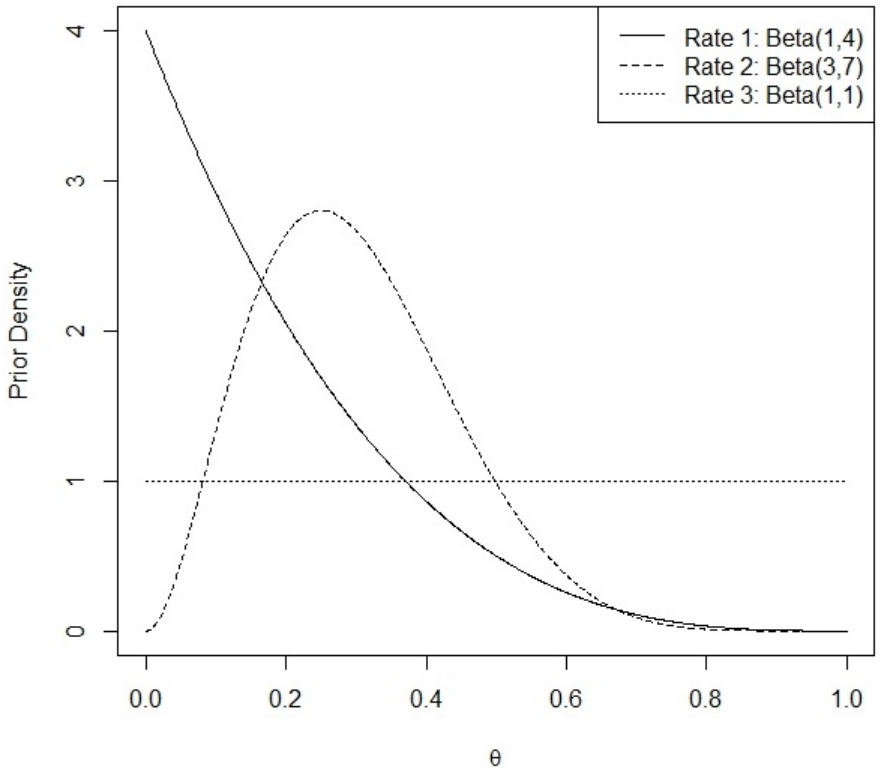


Figure 2.1: Prior structures used in binomial sample size determination example

We will use the same specifications used in the original article; thus, we place a prior probability of 0.6 on $H_0 : \theta_1 = \theta_2$, set $c = 1$, and attempt to find the sample size needed to have an EBP of at least 0.7.

To explain the general method for finding EBP given a fixed sample size, we first consider the case of a sample size of two. We create a predictive density matrix of possible number of successes that only extends from zero to two for both y_1 and

y_2. Each of the nine points in this matrix has a prior predictive density assigned to it, and the sum of this three by three matrix is, of course, one. Then, using the formulation above, we determine which points are part of the rejection region. The sum of the predictive probabilities of the points included in the rejection region is 0.667, which is our EBP for a sample size of two.[9]

Clearly we need to try larger values of n in order to reach our desired EBP of 0.7. The algorithm continues by finding the EBP for our defaulted value of N, 100; if we have not found the correct sample size to meet our desired EBP at this point, it continues by using the bi-sectional approach described in Algorithm 2.1. For this example, a sample size of 64 provides an EBP of 0.750, so we simply have to go backwards until we find the first value that does not exceed 0.7. This gives us $n_N^* = 48$, and it provides an EBP of 0.706 with an EBSL of 0.087.

Figure 2.2 shows the values of EBP for a sample size of up to 98 in our example, and how EBP generally increases as sample size increases. The ridges should provide evidence that candidate solutions initially appearing to be optimal solutions are not guaranteed to be optimal, and that small sample sizes should be checked for validation due to their erratic behavior. Note that Zhao et al. (2011) did not consider this phenomenon and declared 43 to be the optimal sample size, but it should be noted that the EBP for a sample size of 47 actually falls below 0.7.

We can also verify these results via simulation. We generate one million random values of θ_1 and θ_2 from the priors given above. Then using a sample size of 48, we generate a value of y_1 and y_2 for each random value of θ_1 and θ_2. Then, using the rejection region from Equation 2.7, we determine how many of the one million experiments end in correctly rejecting the null hypothesis. Running this simulation produces an approximate EBP of 0.706, which is the same value we find when computing it analytically.

[9] While this seems abnormally large, it is explainable given the constructs of the problem.

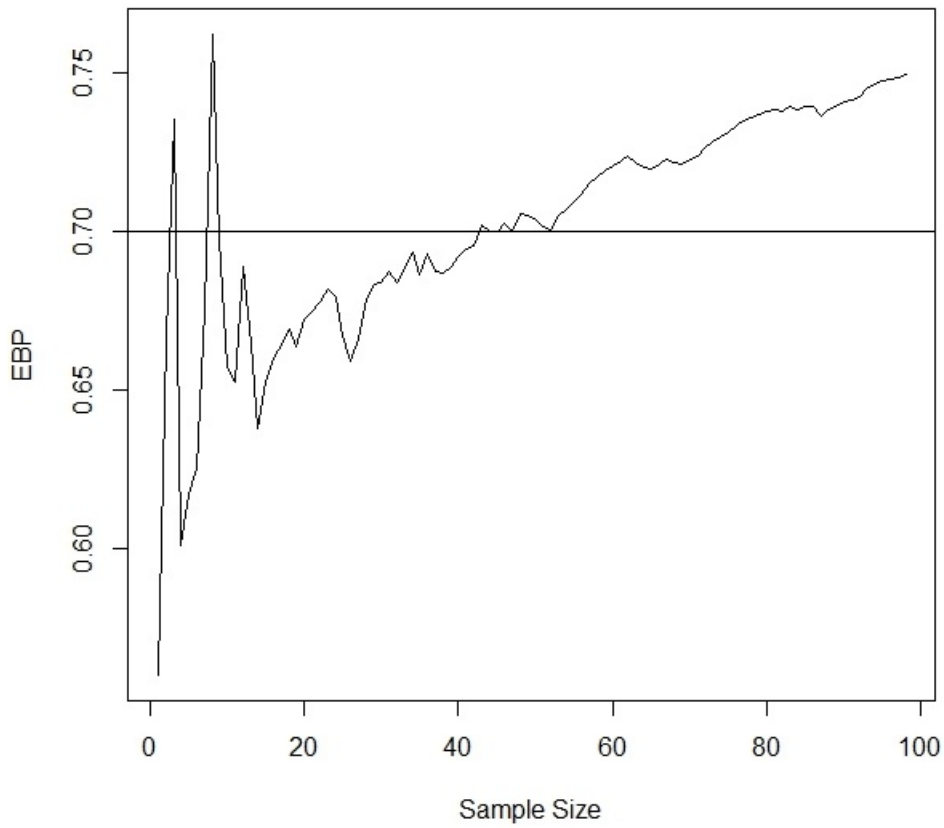


Figure 2.2: EBP curve for binomial sample size determination example

2.4.3 Results Using Alternate Priors

Next, Table 2.1 displays various results of the algorithm for different beta priors where we fix our desired EBP and only consider the $0-1$ loss function such that $c = 1$. We show the results and operating characteristics using a desired EBP of 0.7 with $\pi_0 = 0.6$. Note that the first two cases are replications from Zhao et al. (2011). The last column in the table is the time, in minutes, that the algorithm takes to run.

Note that in the first two cases, the fact that we are specifying a prior distribution on θ independent of our choices for θ_1 and θ_2 means switching these priors does not affect calculations. Also, due to the nature of the beta distribution, switching both parameters within a distribution does not affect results either. This is because flipping both parameters within the beta distributions will only mirror them both.

Table 2.1: Various results for binomial sample size determination when controlling EBP and setting $c = 1$

θ		θ_1		θ_2		EBP	EBSL	n	time
α	β	α_1	β_1	α_2	β_2				
1	1	1	4	3	7	0.706	0.087	48	<1
1	1	4	1	7	3	0.706	0.087	48	<1
3	1	3	1	1.8	1	0.700	0.038	83	<1
30	10	30	10	18	10	0.700	0.047	288	1
3	1	3	1	1.4	1	0.700	0.039	65	<1
30	10	30	10	14	10	0.703	0.074	70	<1
3	1	3	1	1	1	0.703	0.042	43	<1
30	10	30	10	10	10	0.707	0.116	15	<1

This can be evidenced by the fact that both distributions are centered 0.1 units apart but have the same variability.

The next three sets of cases provide an interesting discussion. In all three cases, we consider the case where our first binomial rate, θ_1, is already established as the industry standard, and a "new" process has a rate of θ_2. Thus, when we assume that the null hypothesis is true, we say that $\theta = \theta_1 = \theta_2$ has the same prior as θ_1 because it is already established as the standard. Here we are interested in comparing a "new" process to an established processes having a success rate of 75%. We then calculate the necessary sample size in order to reach our desired EBP when the "new" process has a success rate of 64%, 58%, and 50%. However, for each of these three comparisons we analyze what happens when we increase or decrease the precision of these priors. When considering the case of comparing to a rate of 64%, we see that our required sample size for a more precise prior is nearly three and a half times as large as the more diffuse prior. However, when we decrease our established success rate to 58%, we see our required sample sizes are nearly the same. But by the time we look at the case of a 50% established success rate, the more diffuse prior is actually the one that requires nearly a three times as large sample size.

This phenomenon can be explained by considering the shape of these distributions. While their centers (noted above) and variances (clearly decreasing on the

more informative priors) tell a large percentage of the story, they do not tell it all. For the more informative priors, we need larger sample sizes as the priors become centered closer together; this is exactly what we would expect. However, for the less informative priors, the shapes of these distributions are not the typical curves that can be approximated by a normal distribution. Rather, these curves increase as our rates increase so that our maximum likelihood occurs at 1 instead of closer to their mean. This, in turn, actually creates less overlap between the priors in line three of Table 2.1 than in line four, which would be counter intuitive if only considering their variances. Further, note that a similar effect can be seen with EBSL values

It should also be pointed out that we did not consider other values of our loss function constant due to space limitations. However, due to the fact that we have defined c to be the ratio of how much worse a Type I error is in relation to a Type II error, increasing this value will decrease our EBSL while increasing the necessary sample size, while decreasing c will have the opposite effect.

2.5 Finding Sample Size from a Specified EBSL and Loss Function

2.5.1 General Algorithm

This algorithm follows similarly to that of the one previously; however, here we aim to find a solution to the sample size determination problem when we specify our loss function constant and our desired EBSL, α_0. We now define a candidate solution as any value of n such that Inequality 2.10 holds for n but not $n-1$.

$$\begin{aligned} P\left(\mathbf{Y} \in W | \varphi = 0\right) &= \sum_{\mathbf{y} \in W} P\left(\mathbf{Y} = \mathbf{y} | \varphi = 0\right) \\ &= \sum_{\mathbf{y} \in W} \binom{n}{y_1} \binom{n}{y_2} \frac{B\left(y_1 + y_2 + \alpha, 2n - y_1 - y_2 + \beta\right)}{B\left(\alpha, \beta\right)} \\ &\leq \alpha_0. \end{aligned} \tag{2.10}$$

However, we are really interested in finding the optimal solution, n^*, defined as the smallest candidate solution such that Inequality 2.10 holds for all values larger than

n^*. This is again important because we are not guaranteed a nonincreasing EBSL as our sample size increases due to the discrete nature of the problem.

Despite the lack of a formal proof, it is again a very reasonable conjecture to make that n^* exists; if this is true, it is unique by definition. The aim of our algorithm is to find this value, but this is again analytically intractable. Thus, we define n_N^* to be the smallest value of $n \leq N$ such that Inequality 2.10 holds for all values between n_N^* and N. This is clearly a candidate solution that is an approximation of n^*. It should again be obvious that $n_N^* = n^*$ for a sufficiently large N, and further, that for increasing N, n_N^* approaches n^*. However, it should still be noted that there is no guarantee that n_N^* exists. Thus, if it does not, we again report the first candidate solution greater than N that we find.[10] [11]

Note that now we condition our data on the null hypothesis being true.[12] For a fixed sample size, n, we again set up a matrix of all the possible data values that we could observe. Because the possible values of $\mathbf{y}$ extend from zero to n, we set up a predictive distribution matrix sot that for any values of π_0, c and n, our predictive distribution matrix of all possible values of $\mathbf{y}$ looks like Equation 2.11.

$$P(\mathbf{Y} = \mathbf{y}|\varphi = 0) = \begin{bmatrix} P(\mathbf{Y} = (0,0)|\varphi = 0) & \cdots & P(\mathbf{Y} = (0,n)|\varphi = 0) \\ \vdots & \ddots & \vdots \\ P(\mathbf{Y} = (n,0)|\varphi = 0) & \cdots & P(\mathbf{Y} = (n,n)|\varphi = 0) \end{bmatrix} \tag{2.11}$$

Note that the form of these values is found in Equation 2.5, and that this matrix provides the probabilities of having $\mathbf{y}$ successes; summing all of these values will equal one. Algorithm 2.2 shows how we use this to find a required sample size.

It should again be noted that it becomes increasingly difficult to find the optimal solution as N gets large because the predictive matrix becomes exponentially

[10] This is not necessarily the smallest candidate solution.

[11] This process is again detailed in Algorithm 2.2.

[12] Hence, we use the prior structure specified such that $\theta \sim \text{beta}(\alpha, \beta)$.

Algorithm 2.2: When specifying the desired EBSL, α_0, the loss function constant, c, and N, this algorithm finds the approximate solution n_N^* it is exists; if it does not, it finds a candidate solution greater than N

1: Fix $n = N$.
2: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 2.11; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 2.7). This is the EBSL for the fixed value of n.
3: If the resulting EBSL is less than α_0, $n = n - 1$; otherwise, skip to step 5.
4: Repeat steps 2 and 3 until the value of n provides an EBSL greater than α_0; n_N^* is the last value of n that did not cause the EBSL to rise above α_0, and the algorithm ends.
5: n_N^* does not exist; fix $n_0 = N$ and $n_1 = 2N$.
6: Repeat step 2 for n_1.
7: If the resulting EBSL of n_1 is greater than α_0, $n_0 = n_1$ and $n_1 = 2n_1$; go to step 6.
8: Fix $n = (n_0 + n_1)/2$ and repeat step 2.
9: If the resulting EBSL is less than α_0, $n_1 = n$; else, $n_0 = n$.
10: Repeat steps 8 and 9 until $n_1 - n_0 = 1$; select n_1 as the candidate solution.

large. Further, it should be noted that empirical evidence suggests that the likelihood of a substantial increase in EBSL occurring at larger sample sizes decreases as n increases. This is because EBSL is only increasing because of the discrete nature of the problem; namely, a sample size change of one occasionally affects the dimensions of our rejection region such that we incorrectly reject a true null hypothesis a larger percentage of the time than previously. However, as the sample size increases, the changes in the rejection region become less extreme. Thus, it becomes improbable for us to see a large increase in EBSL as n increases. It is for these reasons that we again default the algorithm at $N = 100$; however, this option can be changed by the user if desired.

2.5.2 Example

As in the previous example, we will continue with the case that we highlight from Zhao et al. (2011). Recall that this is a beta$(1, 4)$ prior on θ_1, a beta$(3, 7)$ prior on θ_2, and a beta$(1, 1)$ prior on θ when we assume that the true rates are the same under the null hypothesis. These priors can again be seen graphically in Figure 2.1.

As before, we will use the same specifications used in the original article; thus, we place a prior probability of 0.6 on $H_0 : \theta_1 = \theta_2$, set $c = 1$, and attempt to find the sample size needed to have a EBSL no greater than 0.05. We do this by applying a similar approach as before; however, instead of EBP increasing as our sample size increases, we are now dealing with decreasing EBSL as n increases. We again start with our defaulted N of 100, which does not provide a EBSL less than our threshold of 0.05. However, an n of 200 provides an EBSL 0.038. Using the bi-sectional approach described in Algorithm 2.2 suggests that the required sample size is 122, as it provides an EBSL of 0.050 with an EBP of 0.763.

Figure 2.3 shows the values of EBSL for a sample size of up to 172 in our example, and how EBSL generally decreases as sample size increases. The ridges should provide clear evidence that candidate solutions initially appearing to be optimal solutions are not guaranteed to be optimal, and that small sample sizes should be checked for validation due to their erratic behavior.

We can also verify these results via simulation. Because under H_0 we have only one distribution for $\theta_1 = \theta_2 = \theta$, we generate one million values of θ. We then use a sample size of 122 to generate values of y_1 and y_2, and determine how many times we falsely reject a true null hypothesis. Running this simulation produces an EBSL of 0.050, which is the same value we find when computing it analytically.

2.5.3 Results Using Alternate Priors

Next, Table 2.2 displays various results of the algorithm for different beta priors where we fix our desired EBSL and only consider the $0-1$ loss function such that $c = 1$. We show the results and operating characteristics using a desired EBSL of 0.05 with $\pi_0 = 0.6$. Note again that the first two cases are replications from Zhao et al. (2011). The last column in the table is the time, in minutes, that the algorithm takes to run.

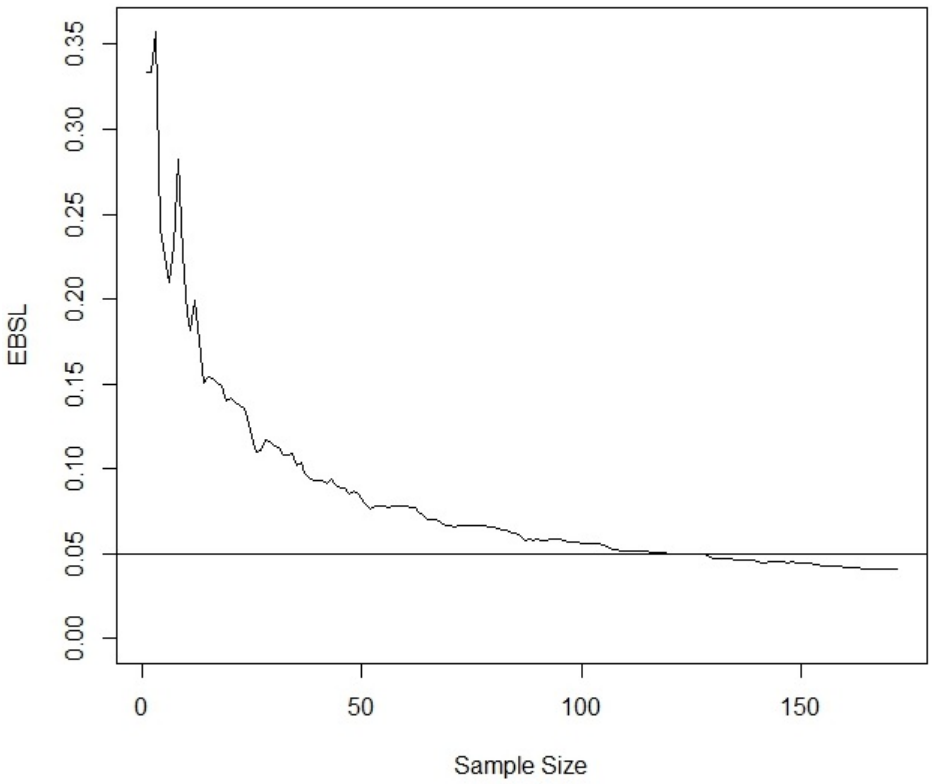


Figure 2.3: EBSL curve for binomial sample size determination example

Note that, as before, the first two cases have the same specifications. Further, when looking at the last six cases, we see a similar phenomenon we noticed earlier when controlling for EBP. Clearly the shapes and precisions of these distributions is creating a similar effect with regard to how much their likelihoods overlap.

As before, even though we did not consider other values of our loss constant, the same discussion about changing c remains. Due to the fact that we have defined c to be the ratio of how much worse a Type I error is in relation to a Type II error, increasing this value will decrease our EBP while increasing the necessary sample size, while decreasing c will have the opposite effect.

Table 2.2: Various results for binomial sample size determination when controlling EBSL and setting $c = 1$

θ		θ_1		θ_2		EBP	EBSL	n	time
α	β	α_1	β_1	α_2	β_2				
1	1	1	4	3	7	0.763	0.050	122	<1
1	1	4	1	7	3	0.763	0.050	122	<1
3	1	3	1	1.8	1	0.641	0.050	49	<1
30	10	30	10	18	10	0.688	0.050	255	1
3	1	3	1	1.4	1	0.639	0.050	38	<1
30	10	30	10	14	10	0.749	0.049	132	<1
3	1	3	1	1	1	0.656	0.047	29	<1
30	10	30	10	10	10	0.823	0.045	61	<1

2.6 Finding Sample Size from a Specified EBSL and EBP

In this section, we aim to find a solution to the sample size determination problem when we specify our desired EBSL and EBP.[13] Here, we define our solution as the first value of n that satisfies Inequalities 2.8 and 2.10. We are not concerning ourselves with an optimal solution because of the effect that c has on these values; any increase in n and subsequent change to c can create a situation where our operating characteristics are not met to satisfaction. Thus, the result from the algorithm is simply the smallest solution out of an infinite number to the situation given, which we notate n^*_{min}.

Further, we force c to be greater than one due to the belief that any practical use of these methods will require the penalty for a Type I error to be greater than that of a Type II error. Thus, a situation that would require c to be less than one is of no consequence to us. Algorithm 2.3 shows how we determine a sample size for this problem.

For the same prior structure that we have been considering throughout this chapter, we run this algorithm in order to find the minimum sample size needed in order to reach an EBP of 0.7 and an EBSL of 0.05. Recall that we are still letting $\pi_0 = 0.6$. For example, letting n be any value less than 48 requires c to be less than

[13] Note that we are not specifying our loss function constant, c, in this case.

Algorithm 2.3: When specifying the desired EBSL, α_0, and EBP, $1-\beta_0$, this algorithm finds n^*_{min}

1: Fix $n = 2$.
2: Fix $c = 1/c = 1$.
3: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 2.9; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 2.7). This is the EBP for the fixed values of n and $1/c$.
4: If the resulting EBP is less than $1-\beta_0$, $n = n+1$; repeat steps 3 and 4.
5: Fix $1/c_0 = 0$ and $1/c_1 = 1$.
6: Fix $1/c = (1/c_0 + 1/c_1)/2$.
7: Repeat step 3 for n and $1/c$.
8: If the resulting EBP is less than $1-\beta_0$, $1/c_0 = 1/c$; else $1/c_1 = 1/c$.
9: Repeat steps 6 and 7 until $|\text{EBP} - (1-\beta_0)|$ is less than some threshold.[14]
10: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 2.11; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 2.7). This is the EBSL for the fixed values of n and $1/c$.
11: If the resulting EBSL is greater than α_0, $n = n+1$ and go to step 5; otherwise $n^*_{min} = n$.

one; because of this restriction, there is no value of c that will bring our EBP above 0.7. However, when $n = 49$, the algorithm suggests that $c = 1.02$ brings down our EBSL 0.002 from its previous value when $c = 1$. The solution to this scenario is to select $c = 1.23$, where we can reach both of our operating characteristics with a sample size of 75.[15] This will provide us with an EBP of 0.701 and an EBSL of 0.048. We can again verify these results via simulation as illustrated previously.

Next, Table 2.3 displays various results of the algorithm for different beta priors where we fix our desired EBSL and EBP while letting our loss function constant, c, vary. We show the results and operating characteristics using a desired EBSL of 0.05, a desired EBP of 0.7, and $\pi_0 = 0.6$. Note again that the first two cases are replications from Zhao et al. (2011). The last column in the table is the time, in minutes, that the algorithm takes to run.

[15] This is an improvement to the required sample size of 122 previously required when controlling for EBSL and using $c = 1$.

Table 2.3: Various results for binomial sample size determination when controlling EBSL and EBP

θ		θ_1		θ_2		EBP	EBSL	c	n	time
α	β	α_1	β_1	α_2	β_2					
1	1	1	4	3	7	0.701	0.048	1.23	75	1
1	1	4	1	7	3	0.701	0.048	1.23	75	1
3	1	3	1	1.8	1	0.700	0.038	1.01	83	<1
30	10	30	10	18	10	0.700	0.047	1.00	288	12
3	1	3	1	1.4	1	0.700	0.039	1.00	65	<1
30	10	30	10	14	10	0.700	0.049	1.25	91	1
3	1	3	1	1	1	0.700	0.040	1.03	43	<1
30	10	30	10	10	10	0.701	0.049	1.74	28	<1

Note that essentially what we have done is what was previously mentioned with regards to our loss function constant, c. Because we know that increasing c will decrease EBSL or EBP while increasing the necessary sample size, we can compare these results to previous results and see that effect occurring. Note how in certain cases we could have decreased c to create the opposite effect (decreasing our necessary sample size to increase EBSL); however, the restriction that c should be at least one prevents this from occurring.

2.7 Finding Sample Size from a Specified EBSL, EBP, and Loss Function

Discussion from industry statisticians prompted the next discussion. Essentially, what if we only desire one unknown instead of two? This section details how to find a minimum necessary sample size in the case where we have a fixed loss function constant and still want to reach a desired EBP and EBSL. Algorithm 2.4 shows how we accomplish this.

Algorithm 2.4: When specifying the desired EBSL, α_0, EBP, $1-\beta_0$, loss function constant, c, and N, this algorithm finds the approximate solution n_N^* it is exists; if it does not, it finds a candidate solution greater than N

1: Run Algorithm 2.1.
2: Run Algorithm 2.2.
3: Choose the larger sample size of the two provided by the two algorithms.

It seems to reason that the smaller of the two sample sizes would only meet one of the two operating characteristics. This can be seen in the example using the same prior structure covered throughout the chapter. If we fix our loss function constant at $c = 1$, the first algorithm to control for EBP gives that a sample size of 48 produces an EBP of 0.706 and an EBSL of 0.087. When running the second algorithm to control for EBSL, we find that a sample size of 122 produces an EBSL of 0.050 with an EBP of 0.763. Thus, we need to take the larger of the two values because it is the one that meets both criteria. This will hold for varying values of c.

Next, Table 2.4 displays various results using this approach for different beta priors where we fix our desired EBSL, EBP, and loss function constant such that $c = 1$. We show the results and operating characteristics using a desired EBSL of 0.05, a desired EBP of 0.7, and $\pi_0 = 0.6$. Note again that the first two cases are replications from Zhao et al. (2011). The last column in the table is the time, in minutes, that the algorithm takes to run.

Table 2.4: Various results for binomial sample size determination when controlling EBSL, EBP, and setting $c = 1$

θ		θ_1		θ_2		EBP	EBSL	n	time
α	β	α_1	β_1	α_2	β_2				
1	1	1	4	3	7	0.763	0.050	122	<1
1	1	4	1	7	3	0.763	0.050	122	<1
3	1	3	1	1.8	1	0.700	0.038	83	<1
30	10	30	10	18	10	0.700	0.047	288	1
3	1	3	1	1.4	1	0.700	0.039	65	<1
30	10	30	10	14	10	0.749	0.049	132	<1
3	1	3	1	1	1	0.703	0.042	43	<1
30	10	30	10	10	10	0.823	0.045	61	<1

It can be clearly seen that some of the operating characteristics are not only met, but well exceeded. This is a product of only having one unknown with two equations; because we have fixed the other three values, there might not be an intersection where we barely eclipse both operating characteristics.

2.8 Conclusion

To recap, we have used conjugate prior structures in order to asses our beliefs about a rate parameter in a two sample binomial trial a priori in order in to find the minimal sample size needed to reach certain expected operating characteristics. By the use of a loss function constant, we are able to control for at least two properties between how much worse a Type I error is in relation to a Type II error, desired expected Bayesian significance level, and desired expected Bayesian power. This type of analysis had never been considered previously.

It is of note that we were not able to consider the use of analysis priors in this research. Ideally, we would be able to adapt this research to account for the fact that researchers often times use one set of priors when conducting sample size analyses, but a more vague or non-informative set of priors when actually analyzing the experiment. This adaptation would provide better sample size and operating characteristic estimates, though for now we are limited to the case where we use the same prior structure throughout. Further, this process could be expanded to consider non-conjugate priors as well. However, the analytical tractability of conjugate priors made it an ideal use, and modeling prior beliefs of a binomial rate with a beta distribution is not an unreasonable thing to do.

It also should be noted that time considerations, while already improved throughout the process, can always continue to improve. One improvement to current methods involve replacing the bi-sectional approaches described in Algorithms 2.2 and 2.2 with one that approximates the EBP curve with some logarithmic function; this improvement should get us in the ballpark of a candidate solution much quicker. Future work also includes a more in depth look at how expected Bayesian error rates compare to typical frequentist ones, and potentially an in-depth look at the different sample size determination and testing methods in order to determine the relative advantages and disadvantages of each. Further, we could generalize the

algorithm such that we are not looking at a common sample size $n = n_1 = n_2$, but rather two different sample sizes n_1 and n_2 such that they do not need to be equal.

Lastly, while the general code used for sample size determination can be found in the Appendices, the entire package will be made available for download soon in the software program R. It will be able to not only handle sample size determination for both the binomial and Poisson cases, but also provide various graphics along with the actual test that would be conducted after data collection.

CHAPTER THREE

Bayesian Sample Size Determination for Two Sample Poisson Experiments

3.1 Introduction

Sample size determination continues to be an important research area of statistics. Cost and time constraints have made finding the appropriate sample size before conducting a study of the utmost importance. Too small of a sample can lead to under-powered studies and a waste of time and money. Too large of a sample size, while not only being a waste of those resources, can also make any difference in observation seem significant. In this chapter, we aim to find the required sample size to meet a varying array of operating characteristics when testing the equality of two Poisson rates from a Bayesian perspective.

The advantages to Bayesian sample size determination are plentiful and have been enumerated by Adcock (1997). Their construction does not depend on asymptotic approximations and allow for the characterization of uncertainty when assessing the parameter of interest. In the frequentist domain, a sample size determination problem requires the researcher to specify what he believes the parameter to be. However, as M'Lan et al. (2008) note, the parameter(s) will most likely never be known with high accuracy at the planning stage, such that the degree to which the researcher is incorrect can create substantial problems with respect to the power of the test and create doubt about the sample size estimated. The ability for the researcher to model his indecision about the parameter through expert knowledge or previous studies can allow the Bayesian approach to have better operating characteristics (such as a smaller required sample size or better Type I and II error rates) as noted by Bayarri and Berger (2004) among others. However, it should always remain prevalent in the reader's mind that misspecification in the Bayesian con-

text can prevent desirable operating characteristics just as it can in the frequentist context.[1]

Various Bayesian sample size determination methods have been previously studied. While there is no way to accurately assess them all, we cover a few specifically here. Stamey et al. (2006) considered one and two sample Poisson rates from the perspective of interval based criteria such as coverage and width. Hand et al. (2011) extend those ideas to consider both interval-based and test-based criteria. However, that effort is limited in its abilities with respect to test-based criteria; namely, it does not consider the expected power of the test that will be conducted. Katsis and Toman (1999) used more decision theoretic test-based criteria for the two sample binomial case, but only to the the extent that they aim to control the posterior risk with a prespecified bound. Zhao et al. (2011) extend on those ideas by using computational methods now available to consider expected Bayesian power of the test. In this chapter, we extend the results of Zhao et al. (2011) by accounting for both expected Type I and Type II error rates while adapting from the binomial data model they considered to the Poisson data model. This sort of sample size procedure for the comparison of two Poisson rates has not been considered before in a Bayesian context.

Further, it should be noted that, due to the fact that we are in the Bayesian framework, references to Type I and Type II error rates, significance level, and power are meant in the Bayesian context. By this we mean that they are to be understood in an expected sense, as outlined by O'Hagan et al. (2005). For example, the expected power of a test is the power of the test when averaged over the likelihood of some other distribution that assesses the perceived likelihood of observing different

[1] Because Type I error rates are not controlled in the Bayesian paradigm as they are in the frequentist one, misspecification here leads to the possibility of either type of error being quite large.

possible parameter values.[2] However, when the test itself uses Bayesian methods, we refer to this as the expected Bayesian power (EBP), as is done in Speighelhalter et al. (2004).[3] Thus, we weight the true values of power by the prior distribution associated with the alternative hypothesis. These ideas, though not considered in the literature previously, can also apply to significance level.[4] In this chapter, we consider expected Bayesian significance level (EBSL), such that we weight true values of significance level by the prior distribution associated with the null hypothesis.[5]

This chapter is arranged as follows. In Section 3.2, we introduce the basic framework for the two sample Poissonl problem. Section 3.3 details how we arrive at the required sample size, introducing some decision theory not previously considered. Section 3.4 outlines the case of controlling for EBP while fixing a loss function; Section 3.5 does the same for controlling EBSL. Section 3.6 describes how we can control for both EBSL and EBP, finding the minimum necessary sample size required if we are not interested in specifying the loss in our loss function; Section 3.7 discusses the case where we are interested in controlling all three of these things.

3.2 Framework

We follow the general framework of Zhao et al. (2011) in the development of this problem, adapting the binomial case to fit the Poisson data model. Suppose observations from two independent populations come from Poisson distributions with rate parameters λ_1 and λ_2, respectively, and common sample size, t, which is often person years. Thus, we have $Y_1 \sim \text{Poisson}(t\lambda_1)$ and $Y_2 \sim \text{Poisson}(t\lambda_2)$. Our interest

[2] This has application to the frequentist paradigm, where we could consider this individual value rather than a power curve.

[3] Another common name for this is the probability of a successful test or trial.

[4] While frequentist methods typically report one value for significance level, what they are really doing (in non point null hypotheses) is taking the largest possible significance level; thus, taking an expectation of a significance level curve could be done as well.

[5] Of course this implies that actual Type I and Type II error rates can be greater than or less than what is indicated when considering expected error rates.

is to calculate the necessary sample size required to test the hypotheses

$$H_0 : \lambda_1 = \lambda_2$$

vs

$$H_1 : \lambda_1 \neq \lambda_2.$$

As a notational convenience, we will refer to the truth about which hypothesis is correct with φ. That is, when the null hypothesis that $\lambda_1 = \lambda_2$ is true, we will denote that by saying $\varphi = 0$; when the alternative hypothesis that $\lambda_1 \neq \lambda_2$ is true, we will denote that by saying $\varphi = 1$. In addition, we place prior probabilities of π_0 and $\pi_1 = 1 - \pi_0$ on H_0 and H_1, respectively; that is, $P(\varphi = 0) = \pi_0$ and $P(\varphi = 1) = \pi_1 = 1 - \pi_0$. This is to say that, in a practical sense when we are unsure about which hypothesis is true, we would set $\pi_0 = \pi_1 = 0.5$.

Also, we assume that the joint prior distribution of λ_1 and λ_2 is of the form

$$f(\lambda_1, \lambda_2) \;=\; \pi_0 p_0(\lambda) I[\varphi = 0] + \pi_1 p_1(\lambda_1, \lambda_2) I[\varphi = 1],$$

where $I[\varphi = 0]$ and $I[\varphi = 1]$ are the indicator functions of H_0 and H_1, respectively, $p_0(\lambda)$ is the prior distribution of λ under the assumption that $\lambda = \lambda_1 = \lambda_2$, and $p_1(\lambda_1, \lambda_2)$ is the joint prior distribution of λ_1 and λ_2 when $\lambda_1 \neq \lambda_2$. Note that we will place prior distributions on λ, λ_1, and λ_2 that summarize our beliefs about them; this information will allow us to compute an estimate for the necessary sample size. For simplicity, we will consider conjugate priors for all three λ's, so that under H_0, $\lambda \sim \text{gamma}(\alpha, \beta)$ and under H_1, $\lambda_i \sim \text{gamma}(\alpha_i, \beta_i)$. Thus, we will model our beliefs about these parameters such that we think there is a π_0 probability of the null hypothesis being true, and when that is the case it can be summarized by a $\text{gamma}(\alpha, \beta)$ distribution; further, there is a π_1 probability of the alternative hypothesis being true, and when that is the case the two distributions there can be summarized by $\text{gamma}(\alpha_i, \beta_i)$ distributions.

3.3 Sample Size Determination

3.3.1 Decision Theory

Consider the loss function

$$L(\varphi, \delta) = \begin{cases} 0, & \text{if } \varphi = \delta, \\ c_1, & \text{if } \varphi = 0 \text{ and } \delta = 1, \\ c_2, & \text{if } \varphi = 1 \text{ and } \delta = 0, \end{cases}$$

where $\delta = 0$ represents choosing H_0, while $\delta = 1$ represents a choice of H_1. Thus, c_1 represents the loss due to a Type I error (rejecting a true null hypothesis) and c_2 represents the loss due to a Type II error (failing to reject a false null hypothesis). This represents an improvement to the methods of Zhao et al. (2011) who only used the $0-1$ loss function such that $c_1 = c_2$.

Thus, our risk can be expressed as

$$\begin{aligned} R(\delta) &= E[L(\varphi, \delta)] \\ &= \begin{cases} c_1 P(\varphi = 0 | \mathbf{Y} = \mathbf{y}), & \text{if } \delta = 1, \\ c_2 P(\varphi = 1 | \mathbf{Y} = \mathbf{y}), & \text{if } \delta = 0. \end{cases} \end{aligned}$$

However, we will define c as the ratio of c_1 to c_2, or rather, $c = c_1/c_2$. In other words, we can think of our loss function constant, c, as how much worse it is to make a Type I error than a Type II error. Thus, because our objective is to minimize risk, we can express this in terms of the optimal decision, δ^*, as

$$\delta^*(\mathbf{y}) = \begin{cases} 0, & \text{if } P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) < cP(\varphi = 0 | \mathbf{Y} = \mathbf{y}), \\ 1, & \text{if } P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) \geq cP(\varphi = 0 | \mathbf{Y} = \mathbf{y}). \end{cases}$$

Because the optimal decision is to reject the null hypothesis when the second inequality holds, this implies that our rejection region, W, for this loss function is

$$W = \{\mathbf{y} : P(\varphi = 1 | \mathbf{Y} = \mathbf{y}) \geq cP(\varphi = 0 | \mathbf{Y} = \mathbf{y})\}. \tag{3.1}$$

Note that this follows the same basic notation used by Zhao et al. (2011) with the addition of the loss function constant, c.

3.3.2 Bayes Factors

The Bayes factor is defined as the ratio of the odds in favor of one hypothesis to the other. While this can be reported in both directions, we follow the form of Kass and Raftery (1995) in defining it as the odds in favor of the alternative hypothesis such that a large Bayes factor is evidence that we should reject the null hypothesis. Thus, this is written analytically as

$$\begin{aligned} B &= \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)/P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)}{\pi_1/\pi_0} \\ &= \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\pi_0}{P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\pi_1}. \end{aligned}$$

This ratio is useful in Bayesian inference because it is often interpretted as partially eliminating the influence of the prior on the posterior and, rather, emphasizing the role of the data. Again following the work of Zhao et al. (2011), it can be shown that our decision rule is a function of a Bayes factor because

$$\begin{aligned} W &= \left\{\mathbf{y} : P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right) \geq cP\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\right\} \\ &= \left\{\mathbf{y} : P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\frac{\pi_0}{\pi_1} \geq cP\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\frac{\pi_0}{\pi_1}\right\} \\ &= \left\{\mathbf{y} : \frac{P\left(\varphi = 1|\mathbf{Y} = \mathbf{y}\right)\pi_0}{P\left(\varphi = 0|\mathbf{Y} = \mathbf{y}\right)\pi_1} \geq c\frac{\pi_0}{\pi_1}\right\} \\ &= \left\{\mathbf{y} : B \geq c\frac{\pi_0}{\pi_1}\right\}. \end{aligned} \tag{3.2}$$

This is particularly useful because the Bayes factor, B, will be the test statistic for this hypothesis test. Further, the right side of Equation 3.2 becomes the decision rule for when this test statistic should be rejected. This allows us to have a specific decision rule rather than rely on traditional methods such as rejecting when this value exceeds some arbitrary number such as three.

3.3.3 Rejection Region

However, we can use Bayes' Theorem to show that

$$\begin{aligned} P(\varphi = 0|\mathbf{Y} = \mathbf{y}) &= \frac{P(\mathbf{Y} = \mathbf{y}, \varphi = 0)}{P(\mathbf{Y} = \mathbf{y})} \\ &= \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) P(\varphi = 0)}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) P(\varphi = 0) + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) P(\varphi = 1)} \\ &= \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0 + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1} \end{aligned} \tag{3.3}$$

and, using similar logic,

$$P(\varphi = 1|\mathbf{Y} = \mathbf{y}) = \frac{P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1}{P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \pi_0 + P(\mathbf{Y} = \mathbf{y}|\varphi = 1) \pi_1}, \tag{3.4}$$

where $\mathbf{y} = (y_1, y_2)$. Further, it is seen that

$$\begin{aligned} P(\mathbf{Y} = \mathbf{y}|\varphi = 0) &= f(\mathbf{y}|\varphi = 0) \\ &= \int_0^\infty f(\mathbf{y}, \lambda|\varphi = 0) d\lambda \\ &= \int_0^\infty f(\mathbf{y}|\lambda, \varphi = 0) p_0(\lambda|\varphi = 0) d\lambda \\ &= \int_0^\infty \frac{(\lambda t)^{y_1} e^{-\lambda t}}{y_1!} \frac{(\lambda t)^{y_2} e^{-\lambda t}}{y_2!} \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda} d\lambda \\ &= \int_0^\infty \frac{(\lambda t)^{y_1+y_2} e^{-2\lambda t}}{y_1! y_2!} \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda} d\lambda \\ &= \frac{t^{y_1+y_2} \beta^\alpha}{y_1! y_2! \Gamma(\alpha)} \int_0^\infty \lambda^{y_1+y_2+\alpha-1} e^{-\lambda(2t+\beta)} d\lambda \\ &= \frac{t^{y_1+y_2} \beta^\alpha}{y_1! y_2! \Gamma(\alpha)} \frac{\Gamma(y_1 + y_2 + \alpha)}{(2t+\beta)^{y_1+y_2+\alpha}} * \\ &\quad \int_0^\infty \frac{(2t+\beta)^{y_1+y_2+\alpha}}{\Gamma(y_1 + y_2 + \alpha)} \lambda^{y_1+y_2+\alpha-1} e^{-\lambda(2t+\beta)} d\lambda \\ &= \frac{t^{y_1+y_2} \beta^\alpha \Gamma(y_1 + y_2 + \alpha)}{y_1! y_2! \Gamma(\alpha) (2t+\beta)^{y_1+y_2+\alpha}}, \end{aligned} \tag{3.5}$$

and, extending the same logic to the density of $\mathbf{y}$ conditioned on the alternative hypothesis being true,

$$\begin{aligned}
P\left(\mathbf{Y}=\mathbf{y}|\varphi=1\right) &= f\left(\mathbf{y}|\varphi=1\right) \\
&= \int_0^\infty f\left(\mathbf{y},\boldsymbol{\lambda}|\varphi=1\right)d\boldsymbol{\lambda} \\
&= \int_0^\infty f\left(\mathbf{y}|\boldsymbol{\lambda},\varphi=1\right)p_1\left(\boldsymbol{\lambda}|\varphi=1\right)d\boldsymbol{\lambda} \\
&= \prod_{i=1}^{2}\int_0^\infty \frac{\left(\lambda_i t\right)^{y_i}e^{-\lambda_i t}}{y_i!}\frac{\beta_i^{\alpha_i}}{\Gamma\left(\alpha_i\right)}\lambda_i^{\alpha_i-1}e^{-\beta_i\lambda_i}d\lambda_i \\
&= \prod_{i=1}^{2}\frac{t^{y_i}\beta_i^{\alpha_i}}{y_i!\Gamma\left(\alpha_i\right)}\int_0^\infty \lambda_i^{y_i+\alpha_i-1}e^{-\lambda_i(t+\beta_i)}d\lambda_i \\
&= \prod_{i=1}^{2}\frac{t^{y_i}\beta_i^{\alpha_i}}{y_i!\Gamma\left(\alpha_i\right)}\frac{\Gamma\left(y_i+\alpha_i\right)}{\left(t+\beta_i\right)^{y_i+\alpha_i}}\int_0^\infty \frac{\left(t+\beta_i\right)^{y_i+\alpha_i}}{\Gamma\left(y_i+\alpha_i\right)}\lambda_i^{y_i+\alpha_i-1}e^{-\lambda_i(t+\beta_i)}d\lambda_i \\
&= \prod_{i=1}^{2}\frac{t^{y_i}\beta_i^{\alpha_i}\Gamma\left(y_i+\alpha_i\right)}{y_i!\Gamma\left(\alpha_i\right)\left(t+\beta_i\right)^{y_i+\alpha_i}}, \qquad (3.6)
\end{aligned}$$

where $\boldsymbol{\lambda}=(\lambda_1,\lambda_2)$. Note that this is the product of two independent negative binomial distributions.

Thus, with this information we can return to our rejection region found in Equation 3.1 and, using Equations 3.3 and 3.4, show that

$$\begin{aligned}
W &= \left\{\mathbf{y}: P\left(\varphi=1|\mathbf{Y}=\mathbf{y}\right)\geq cP\left(\varphi=0|\mathbf{Y}=\mathbf{y}\right)\right\} \\
&= \left\{\mathbf{y}: P\left(\mathbf{Y}=\mathbf{y}|\varphi=1\right)\pi_1\geq cP\left(\mathbf{Y}=\mathbf{y}|\varphi=0\right)\pi_0\right\},
\end{aligned}$$

which is again similar to the work of Zhao et al. (2011). Further, we can use Equations 3.5 and 3.6 to show that our rejection region for this test consists of all the points that satisfy Inequality 3.7.

$$\frac{\Gamma\left(\alpha\right)\left(2t+\beta\right)^{y_1+y_2+\alpha}}{\beta^{\alpha}\Gamma\left(y_1+y_2+\alpha\right)}\prod_{i=1}^{2}\frac{\beta_i^{\alpha_i}\Gamma\left(y_i+\alpha_i\right)}{\Gamma\left(\alpha_i\right)\left(t+\beta_i\right)^{y_i+\alpha_i}} \quad\geq\quad c\frac{\pi_0}{\pi_1} \qquad (3.7)$$

Note that the left side of 3.7 is our test statistic and Bayes factor, B. The right side remains our decision rule.

Next, we use this rejection region to find the required sample size needed to meet certain operating characteristics. In Section 3.4, we consider the case where we want to reach a given EBP for a fixed value of the loss function constant, c. We can then solve for t and find the EBSL. In Section 3.5, we control the EBSL for a fixed value of c, allowing us to solve for t and compute the EBP. Despite both of these cases taken independently seeming impractical in the real world, these applications are crucial for the development of the last two cases of interest. In Section 3.6, we consider the case where we are not concerned with the value of c and would like to reach a given EBP while still controlling the EBSL. We can do this by fixing the two desired operating characteristics and solving iteratively for both c and t. Lastly, in Section 3.7, we consider the case where we know our loss function constant, but still would like to be able to control for both EBSL and EBP while solving for t. Note that we are restricting t to be a whole number; however, this does not necessarily need to be the case.

3.4 Finding Sample Size from a Specified EBP and Loss Function

3.4.1 General Algorithm

In this section, we aim to find a solution to the sample size determination problem when we specify our loss function constant and our desired EBP, $1 - \beta_0$. We define a candidate solution as any value of t such that Inequality 3.8 holds for t but not $t - 1$.

$$\begin{aligned} P\left(\mathbf{Y} \in W | \varphi = 1\right) &= \sum_{\mathbf{y} \in W} P\left(\mathbf{Y} = \mathbf{y} | \varphi = 1\right) \\ &= \sum_{\mathbf{y} \in W} \prod_{i=1}^{2} \frac{t^{y_i} \beta_i^{\alpha_i} \Gamma\left(y_i + \alpha_i\right)}{y_i! \Gamma\left(\alpha_i\right) \left(t + \beta_i\right)^{y_i + \alpha_i}} \\ &\geq 1 - \beta_0. \end{aligned} \tag{3.8}$$

However, we are really interested in finding the optimal solution, t^*, defined as the smallest candidate solution such that Inequality 3.8 holds for all values larger than

t^*. This is important because we are not guaranteed a nondecreasing EBP as our sample size increases due to the discrete nature of the problem.

Despite the lack of a formal proof, it is a very reasonable conjecture to make that t^* exists; if this is true, it is unique by definition. Our aim is to find this value; however, this has proven to be analytically intractable. Thus, we define t_T^* to be the smallest value of $t \leq T$ such that Inequality 3.8 holds for all values between t_T^* and T. This is clearly a candidate solution that is an approximation of t^*. It should be obvious that $t_T^* = t^*$ for a sufficiently large T, and further, that for increasing T, t_T^* approaches t^*. However, it should also be noted that there is no guarantee that t_T^* exists. Thus, if it does not, we simply report the first candidate solution greater than T that we find.[6] [7]

Note that we condition our data on the alternative hypothesis being true.[8] For a fixed sample size, we could, in theory, set up a matrix of all the possible data values that we could observe. However, because the infinite support of the Poisson distribution will not allow this, we must first determine how many values of $\mathbf{y}$ to consider. Noting that the predictive distribution of $\mathbf{y}$ is negative binomial under the alternative hypothesis, we set up a predictive distribution matrix. For any values of π_0, c and t, this matrix of the reasonable values of $\mathbf{y}$ (found by using small and large percentiles of the negative binomial distribution) looks like Equation 3.9.

$$P(\mathbf{Y} = \mathbf{y}|\varphi = 1) = \begin{bmatrix} P(\mathbf{Y} = (a_1, a_2)|\varphi = 1) & \cdots & P(\mathbf{Y} = (a_1, z_2)|\varphi = 1) \\ \vdots & \ddots & \vdots \\ P(\mathbf{Y} = (z_1, a_2)|\varphi = 1) & \cdots & P(\mathbf{Y} = (z_1, z_2)|\varphi = 1) \end{bmatrix} \quad (3.9)$$

Note that the form of these values is found in Equation 3.6, and that a_i and z_i indicate reasonably small and large possible values, respectively, for y_i. Further, this

[6] This is not necessarily the smallest candidate solution.

[7] This process is detailed in Algorithm 3.1.

[8] Hence, we use the prior structures specified such that $\lambda_i \sim \text{gamma}(\alpha_i, \beta_i)$.

Algorithm 3.1: When specifying the desired EBP, $1-\beta_0$, the loss function constant, c, and T, this algorithm finds the approximate solution, t_T^*, if it exists; if it does not, it finds a candidate solution greater than T

1: Fix $t = T$.
2: Enumerate the predictive probabilities of all reasonable values of $\mathbf{y}$ using Equation 3.9; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 3.7) and divide by f (the sum of the probabilities of all reasonable values). This is the approximate EBP for the fixed value of t.
3: If the resulting EBP is greater than $1-\beta_0$, $t = t-1$; otherwise, skip to step 5.
4: Repeat steps 2 and 3 until the value of t provides an EBP less than $1-\beta_0$; t_T^* is the last value of t that did not cause the EBP to fall below $1-\beta_0$, and the algorithm ends.
5: t_T^* does not exist; fix $t_0 = T$ and $t_1 = 2T$.
6: Repeat step 2 for t_1.
7: If the resulting EBP of t_1 is less than $1-\beta_1$, $t_0 = t_1$ and $t_1 = 2t_1$; go to step 6.
8: Fix $t = (t_0 + t_1)/2$ and repeat step 2.
9: If the resulting EBP is greater than $1-\beta_0$, $t_1 = n$; else, $t_0 = t$.
10: Repeat steps 8 and 9 until $t_1 - t_0 = 1$; select t_1 as the candidate solution.

matrix provides the probabilities of observing the values of $\mathbf{y}$ under the alternative hypothesis, and summing all of these values up should provide a number very close to one.[9] However, denoting this sum f and dividing our approximate EBP by it accounts for the fact that we do not have a true predictive density. Algorithm 3.1 shows how we use this in order to determine a sample size.

It should be noted that it becomes increasingly difficult to find the optimal solution as T gets large because the predictive matrix becomes exponentially large. Further, it should be noted that empirical evidence suggests that the likelihood of a substantial decrease in EBP occurring at larger sample sizes decreases as t increases. This is because EBP is only decreasing because of the discrete nature of the problem; namely, a sample size change of one occasionally affects the dimensions of our rejection region such that we successfully reject a false null hypothesis a smaller percentage of the time than previously. However, as the sample size increases, the changes in the rejection region become less extreme. Thus, it becomes improbable

[9] Note that it will not equal exactly one because we do not extend the matrix out indefinitely.

for us to see a large decrease in EBP as t increases. It is for these reasons that we default the algorithm at $T = 50$; however, this option can be changed by the user if desired.

3.4.2 Example

Due to our belief that this methodology will be mostly implemented in situations where a Poisson rate, λ_1, is already established as the industry standard, we focus our research on situations where an "old" process produces a rate of λ_1 and a "new" process has a rate of λ_2. Thus, we place a gamma(α_1, β_1) prior on λ_1, a gamma(α_2, β_2) prior on λ_2, and when we assume that the null hypothesis is true, we say that $\lambda = \lambda_1 = \lambda_2$ has the same prior as λ_1. Thus, our rejection region is altered slightly from 3.7 such that it consists of the points that satisfy Inequality 3.10.

$$\frac{\beta_2^{\alpha_2}\left(2t+\beta_1\right)^{y_1+y_2+\alpha_1}}{\Gamma\left(\alpha_2\right)\Gamma\left(y_1+y_2+\alpha_1\right)}\prod_{i=1}^{2}\frac{\Gamma\left(y_i+\alpha_i\right)}{\left(t+\beta_i\right)^{y_i+\alpha_i}} \quad \geq \quad c\frac{\pi_0}{\pi_1} \tag{3.10}$$

For this example, we place a gamma$(8, 4)$ prior on λ_1 and a gamma$(4, 4)$ prior on λ_2. This can be thought of as λ_1 having a rate of eight occurrences in four person years and λ_2 having a rate of four occurrences in four person years. Note that the first process will have a mean of two, a mode of 1.75, and a standard deviation of 0.71; the second process will have a mean of one, a mode of 0.75, and a standard deviation of 0.5. Figure 3.1 shows both of these priors graphically.

We place a prior probability of 0.5 on both $H_0 : \lambda_1 = \lambda_2$ and $H_1 : \lambda_1 \neq \lambda_2$, set $c = 1$, and attempt to find the sample size needed to have an EBP of at least 0.8.

To explain the general method for finding EBP given a fixed sample size, we first consider the case of a sample size of two. We create a predictive density matrix of reasonable values that extends from 0 to 17 for y_1 and 0 to 13 for y_2. There are 252 points in this matrix, each of which has a prior predictive density assigned to it. The sum of this 18 by 14 matrix is 0.9999, indicating that the probability of observing a value outside of this matrix is very small. Then, using the formulation

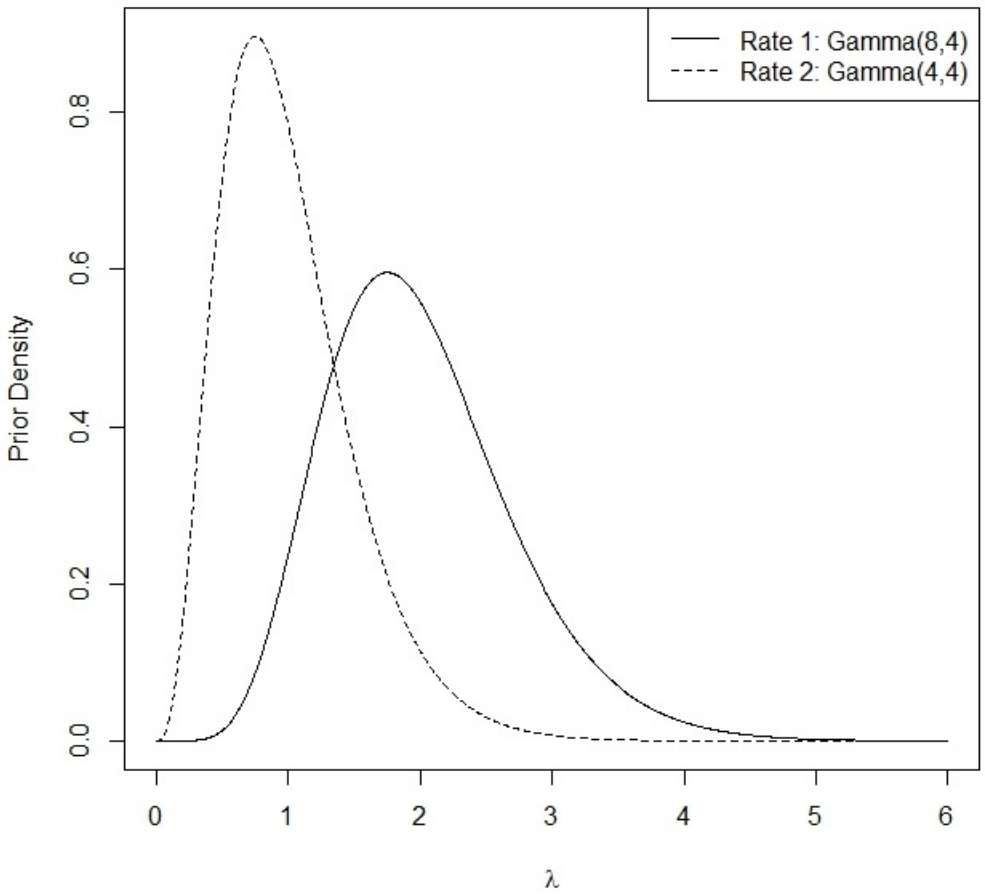


Figure 3.1: Prior structures used in Poisson sample size determination example

above, we determine which points are part of the rejection region. The sum of the predictive probabilities of the points included in the rejection region is 0.694, and dividing by the sum of the entire matrix leaves us with an approximate EBP of 0.694 for a sample size of two.[10]

Clearly we need to try larger values of t in order to reach our desired EBP of 0.8. The algorithm continues by finding the EBP for our defaulted value of T, 50; if we have not found the correct sample size to meet our desired EBP at this point, it continues by using the bi-sectional approach described in Algorithm 3.1. For this example, a sample size of 50 provides an approximate EBP of 0.815, so we simply have to go backwards until we find the first value that does not exceed 0.8. This

[10] While this seems abnormally large, it is explainable given the constructs of the problem.

gives us $T_T^* = 40$, and it provides an approximate EBP of 0.801 with an approximate EBSL of 0.060.

Figure 3.2 shows the values of EBP for a sample size of up to 90 in our example, and how EBP generally increases as sample size increases. The ridges should provide evidence that candidate solutions initially appearing to be optimal solutions are not guaranteed to be optimal, and that small sample sizes should be checked for validation due to their erratic behavior.

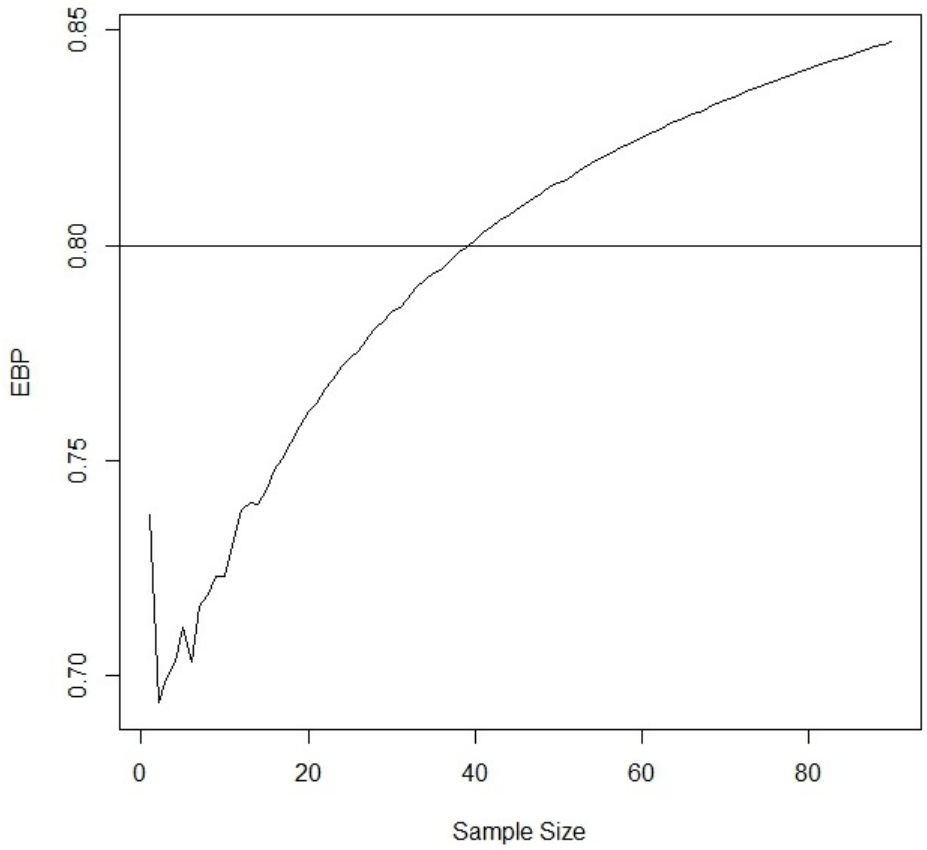


Figure 3.2: EBP curve for Poisson sample size determination example

We can also verify these results via simulation. We generate one million random values of λ_1 and λ_2 from the priors given above. Then using a sample size of 40, we generate a value of y_1 and y_2 for each random value of λ_1 and λ_2. Then, using the rejection region that we created earlier, we determine how many of the one million experiments end in correctly rejecting the null hypothesis. Running this

simulation produces an approximate EBP of 0.801, which is the same value we find when computing it analytically.

3.4.3 Results Using Alternate Priors

Next, Table 3.1 displays various results of the algorithm for different gamma priors where we fix our desired EBP and only consider the $0-1$ loss function such that $c = 1$. We show the results and operating characteristics using a desired EBP of 0.8 with $\pi_0 = 0.5$. The last column in the table is the time, in minutes, that the algorithm takes to run.

Table 3.1: Various results for Poisson sample size determination when controlling EBP and setting $c = 1$

$\lambda = \lambda_1$		λ_2		EBP	EBSL	t	time
α_1	β_1	α_2	β_2				
8	4	4	4	0.801	0.060	40	1
4	4	8	4	0.801	0.064	37	1
1	1	1.5	1	0.801	0.042	60	5
10	10	15	10	0.800	0.054	164	3
1	1	1.7	1	0.801	0.043	51	7
10	10	17	10	0.800	0.081	49	<1
1	1	1.9	1	0.802	0.045	43	3
10	10	19	10	0.803	0.140	13	1

Notice that the first case listed is the one outlined in the example. For the second case, we simply flip which set of prior beliefs corresponds to the "old" process and which one is the "new" process. Notice that we do end up with a slightly different required sample size due to this change because we have flipped which set of values belongs to the null hypothesis.

The next three sets of cases provide an interesting discussion. Here we are interested in comparing a "new" process to an established processes believed to have a rate of one occurrence per person year. We then calculate the necessary sample size in order to reach our desired EBP when the "new" process has a mean of 1.5, 1.7 and 1.9 occurrences per person year. However, for each of these three

comparisons we analyze what happens when we increase or decrease the precision of these priors. When considering the case comparing a mean of one and 1.5, we see that our required sample size for a more precise prior is nearly three times as large as the more diffuse prior. However, when we increase our prior mean to 1.7, we see our required sample sizes are nearly the same. But by the time we look at the case of a prior mean of 1.9, the more diffuse prior is actually the one that requires a larger sample size by over three fold.

This phenomenon can be explained by considering the shape of these distributions. While their centers (noted above) and variances (clearly decreasing on the more informative priors) tell a large percentage of the story, they do not tell it all. For the more informative priors, we need larger sample sizes as the priors become centered closer together; this is exactly what we would expect. However, for the less informative priors, the shape of the gamma$(1, 1)$ distribution is not the typical curve that can be approximated by a normal distribution. Rather, this curve starts at a vertical height of zero, decreasing constantly as the values of the parameter increase. This, in turn, actually creates less overlap between the priors in line three of Table 3.1 than in line four, which would be counter intuitive if only considering their variances. Further, note that a similar effect can be seen with relation to the values of our EBSL.

It should also be pointed out that we did not consider other values of our loss function constant due to space limitations. However, due to the fact that we have defined c to be the ratio of how much worse a Type I error is in relation to a Type II error, increasing this value will decrease our EBSL while increasing the necessary sample size, while decreasing c will have the opposite effect.

3.5 Finding Sample Size from a Specified EBSL and Loss Function

3.5.1 General Algorithm

This algorithm follows similarly to that of the one previously; however, here we aim to find a solution to the sample size determination problem when we specify our loss function constant and our desired EBSL, α_0. We now define a candidate solution as any value of t such that Inequality 3.11 holds for t but not $t-1$.

$$\begin{aligned} P(\mathbf{Y} \in W|\varphi = 0) &= \sum_{\mathbf{y} \in W} P(\mathbf{Y} = \mathbf{y}|\varphi = 0) \\ &= \sum_{\mathbf{y} \in W} \frac{t^{y_1+y_2}\beta^{\alpha}\Gamma(y_1+y_2+\alpha)}{y_1!y_2!\Gamma(\alpha)(2t+\beta)^{y_1+y_2+\alpha}} \\ &\leq \alpha_0. \end{aligned} \tag{3.11}$$

However, we are really interested in finding the optimal solution, t^*, defined as the smallest candidate solution such that Inequality 3.11 holds for all values larger than t^*. This is again important because we are not guaranteed a nonincreasing EBSL as our sample size increases due to the discrete nature of the problem.

Despite the lack of a formal proof, it is again a very reasonable conjecture to make that t^* exists; if this is true, it is unique by definition. The aim of our algorithm is to find this value, but this is again analytically intractable. Thus, we define t_T^* to be the smallest value of $t \leq T$ such that Inequality 3.11 holds for all values between t_T^* and T. This is clearly a candidate solution that is an approximation of t^*. It should again be obvious that $t_T^* = t^*$ for a sufficiently large T, and further, that for increasing T, t_T^* approaches t^*. However, it should still be noted that there is no guarantee that t_T^* exists. Thus, if it does not, we again report the first candidate solution greater than T that we find.[11] [12]

[11] This is not necessarily the smallest candidate solution.

[12] This process is again detailed in Algorithm 3.2.

Note that now we condition our data on the null hypothesis being true.[13] As before, we must first determine how many values of $\mathbf{y}$ to consider; thus, for any values of π_0, c and t, this matrix of reasonable values of $\mathbf{y}$ looks like equation 3.12.

$$P(\mathbf{Y}=\mathbf{y}|\varphi=0) = \begin{bmatrix} P(\mathbf{Y}=(a,a)|\varphi=0) & \cdots & P(\mathbf{Y}=(a,z)|\varphi=0) \\ \vdots & \ddots & \vdots \\ P(\mathbf{Y}=(z,a)|\varphi=0) & \cdots & P(\mathbf{Y}=(z,z)|\varphi=0) \end{bmatrix} \quad (3.12)$$

Note that the form of these values is found in Equation 3.5, and that a and z indicate reasonably small and large possible values, respectively, for our data. Further, this matrix provides the probabilities of observing the values of $\mathbf{y}$ under the null hypothesis, and summing all of these values up should provide a number very close to one.[14] Algorithm 3.2 shows how we use this in order to determine a sample size.

Algorithm 3.2: When specifying the desired EBSL, α_0, the loss function constant, c, and T, this algorithm finds the approximate solution t_T^* it is exists; if it does not, it finds a candidate solution greater than T

1: Fix $t = T$.
2: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 3.12; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 3.7) and divide by f (the sum of the probabilities of all reasonable values). This is the approximate EBSL for the fixed value of t.
3: If the resulting EBSL is less than α_0, $t = t - 1$; otherwise, skip to step 5.
4: Repeat steps 2 and 3 until the value of t provides an EBSL greater than α_0; t_T^* is the last value of t that did not cause the EBSL to rise above α_0.
5: t_T^* does not exist; fix $n_0 = T$ and $t_1 = 2T$.
6: Repeat step 2 for t_1.
7: If the resulting EBSL of t_1 is greater than α_0, $t_0 = t_1$ and $t_1 = 2t_1$; go to step 6.
8: Fix $t = (t_0 + t_1)/2$ and repeat step 2.
9: If the resulting EBSL is less than α_0, $t_1 = t$; else, $t_0 = t$.
10: Repeat steps 8 and 9 until $t_1 - t_0 = 1$; select t_1 as the candidate solution.

It should again be noted that it becomes increasingly difficult to find the optimal solution as T gets large because the predictive matrix becomes exponentially

[13] Hence, we use the prior structure specified such that $\theta \sim \text{gamma}(\alpha, \beta)$.

[14] As before, it will not equal exactly one because we do not extend the matrix out indefinitely. However, we again denote this sum f and divide our approximate EBSL to account for the fact that we do not have a true predictive density.

large. Further, it should be noted that empirical evidence suggests that the likelihood of a substantial increase in EBSL occurring at larger sample sizes decreases as t increases. This is because EBSL is only increasing because of the discrete nature of the problem; namely, a sample size change of one occasionally affects the dimensions of our rejection region such that we incorrectly reject a true null hypothesis a larger percentage of the time than previously. However, as the sample size increases, the changes in the rejection region become less extreme. Thus, it becomes improbable for us to see a large increase in EBSL as t increases. It is for these reasons that we again default the algorithm at $T = 50$; however, this can be changed by the user.

3.5.2 Example

As in the previous example, we will continue with a gamma$(8, 4)$ prior on λ_1 and a gamma$(4, 4)$ prior on λ_2. These priors can be seen graphically in Figure 3.1.

As before, we will place a prior probability of 0.5 on both $H_0 : \lambda_1 = \lambda_2$ and $H_1 : \lambda_1 \neq \lambda_2$, keep $c = 1$, and attempt to find the sample size needed to have an EBSL no greater than 0.05. We do this by applying a similar approach as before; however, instead of EBP increasing as our sample size increases, we are now dealing with decreasing EBSL as t increases. We again start with our defaulted T of 50, which does not provide an EBSL less than our threshold of 0.05. However, a t of 100 provides an approximate EBSL 0.034. Using the bi-sectional approach described in Algorithm 3.2 suggests that the required sample size is 54, as it provides an approximate EBSL of 0.050 with an approximate EBP of 0.819.

Figure 3.3 shows the values of EBSL for a sample size of up to 104 in our example, and how EBSL generally decreases as sample size increases. The ridges should provide clear evidence that candidate solutions initially appearing to be optimal solutions are not guaranteed to be optimal, and that small sample sizes should be checked for validation due to their erratic behavior.

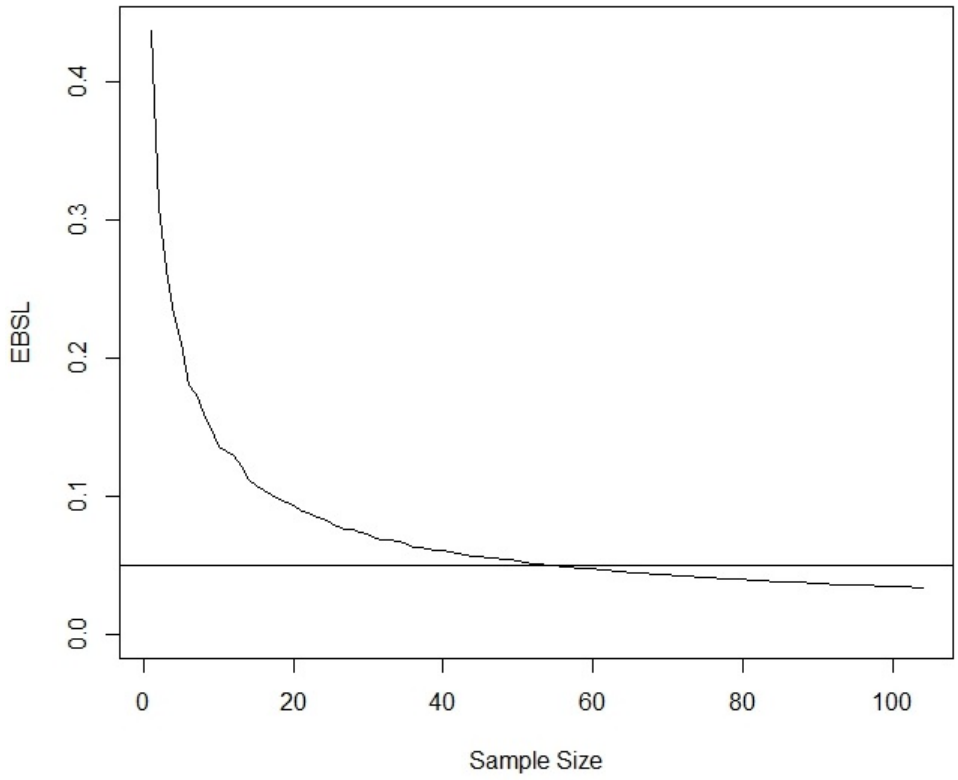


Figure 3.3: EBSL curve for Poisson sample size determination example

We can also verify these results via simulation. Because under H_0 we believe that λ_2 will follow the same distribution as λ_1, we generate one million values of λ_1. We then use a sample size of 54 to generate values of y_1 and y_2, and determine how many times we falsely reject a true null hypothesis. Running this simulation three times produces an EBSL of 0.050, which is the same value we find when computing it analytically.

3.5.3 Results Using Alternate Priors

Next, 3.2 displays various results of the algorithm for different gamma priors where we fix our desired EBSL and only consider the $0-1$ loss function such that $c = 1$. We show the results and operating characteristics using a desired EBSL of 0.05 with $\pi_0 = 0.5$. The last column in the table is the time, in minutes, that the algorithm takes to run.

Table 3.2: Various results for Poisson sample size determination when controlling EBSL and setting $c = 1$

$\lambda = \lambda_1$		λ_2		EBP	EBSL	t	time
α_1	β_1	α_2	β_2				
8	4	4	4	0.819	0.050	54	1
4	4	8	4	0.823	0.049	57	1
1	1	1.5	1	0.780	0.050	45	2
10	10	15	10	0.807	0.050	183	3
1	1	1.7	1	0.783	0.050	40	3
10	10	17	10	0.839	0.050	104	1
1	1	1.9	1	0.787	0.050	35	4
10	10	19	10	0.874	0.050	61	<1

Notice that the first case listed is the one outlined in the example. For the second case, we again flip which set of prior beliefs corresponds to the "old" process and which one is the "new" process. Notice that we again end up with a slightly different required sample size due to this change because we have flipped which set of values belongs to the null hypothesis. Further, when looking at the last six cases, we see a similar phenomenon we noticed earlier when controlling for EBP. Clearly the shapes and precisions of these distributions is creating a similar effect with regard to how much their likelihoods overlap.

As before, even though we did not consider other values of our loss constant, the same discussion about changing c remains. Due to the fact that we have defined c to be the ratio of how much worse a Type I error is in relation to a Type II error, increasing this value will decrease our EBP while increasing the necessary sample size, while decreasing c will have the opposite effect.

3.6 Finding Sample Size from a Specified EBSL and EBP

In this section, we aim to find a solution to the sample size determination problem when we specify our desired EBSL and EBP.[15] Here, we define our solution as the first value of t that satisfies Inequalities 3.8 and 3.11. We are not concerning

[15] Note that we are not specifying our loss function constant, c, in this case.

ourselves with an optimal solution because of the effect that c has on these values; any increase in t and subsequent change to c can create a situation where our operating characteristics are not met to satisfaction. Thus, the result from the algorithm is simply the smallest solution out of an infinite number to the situation given, which we notate t^*_{min}.

Further, we force c to be greater than one due to the belief that any practical use of these methods will require the penalty for a Type I error to be greater than that of a Type II error. Thus, a situation that would require c to be less than one is of no consequence to us. Algorithm 3.3 shows how we determine a sample size for this problem.

Algorithm 3.3: When specifying the desired EBSL, α_0, and EBP, $1-\beta_0$, this algorithm finds t^*_{min}

1: Fix $t = 2$.
2: Fix $c = 1/c = 1$.
3: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 3.9; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 3.7) and divide by f (the sum of the probabilities of all reasonable values). This is the approximate EBP for the fixed values of t and $1/c$.
4: If the resulting EBP is less than $1-\beta_0$, $t = t+1$; repeat steps 3 and 4.
5: Fix $1/c_0 = 0$ and $1/c_1 = 1$.
6: Fix $1/c = (1/c_0 + 1/c_1)/2$.
7: Repeat step 3 for t and $1/c$.
8: If the resulting EBP is less than $1-\beta_0$, $1/c_0 = 1/c$; else $1/c_1 = 1/c$.
9: Repeat steps 6 and 7 until $|\text{EBP} - (1-\beta_0)|$ is less than some threshold.[16]
10: Enumerate the predictive probabilities of all possible values of $\mathbf{y}$ using Equation 3.12; sum the probabilities of the points that are part of the rejection region (those that satisfy Equation 3.7) and divide by f (the sum of the probabilities of all reasonable values). This is the approximate EBSL for the fixed values of t and $1/c$.
11: If the resulting EBSL is greater than α_0, $t = t+1$ and go to step 5; otherwise $t^*_{min} = t$.

For the same prior structure that we have been considering throughout this chapter, we run this algorithm in order to find the minimum sample size needed in

Table 3.3: Various results for Poisson sample size determination when controlling EBSL and EBP

$\lambda = \lambda_1$		λ_2		EBP	EBSL	c	t	time
α_1	β_1	α_2	β_2					
8	4	4	4	0.800	0.048	1.13	45	5
4	4	8	4	0.800	0.050	1.15	43	5
1	1	1.5	1	0.800	0.042	1.01	60	24
10	10	15	10	0.800	0.049	1.05	172	56
1	1	1.7	1	0.800	0.043	1.01	51	15
10	10	17	10	0.800	0.049	1.26	69	10
1	1	1.9	1	0.801	0.043	1.02	43	10
10	10	19	10	0.800	0.048	1.65	30	2

order to reach an EBP of 0.8 and an EBSL of 0.05. Recall that we are still letting $\pi_0 = 0.5$. For example, letting t be any value less than 40 requires c to be less than one; because of this restriction, there is no value of c that will bring our EBP above 0.8. However, when $t = 41$, the algorithm suggests that $c = 1.03$ brings down our approximate EBSL 0.002 from its previous value when $c = 1$. The solution to this scenario is to select $c = 1.13$, where we can reach both of our operating characteristics with a sample size of 45.[17] This will provide us with an approximate EBP of 0.800 and an approximate EBSL of 0.048. We can again verify these results via simulation as illustrated in previous sections.

Next, Table 3.3 displays various results of the algorithm for different gamma priors where we fix our desired EBSL and EBP while letting our loss function constant, c, vary. We show the results and operating characteristics using a desired EBSL of 0.05, a desired EBP of 0.8, and $\pi_0 = 0.5$. The last column in the table is the time, in minutes, that the algorithm takes to run.

Note that essentially what we have done is what was previously mentioned with regards to our loss function constant, c. Because we know that increasing c will decrease EBSL or EBP while increasing the necessary sample size, we can

[17] This is an improvement to the required sample size of 54 previously required when controlling for EBSL and using $c = 1$.

compare these results to previous results and see that effect occurring. Note how in certain cases we could have decreased c to create the opposite effect (decreasing our necessary sample size to increase EBSL); however, the restriction that c should be at least one prevents this from occurring.

3.7 Finding Sample Size from a Specified EBSL, EBP, and Loss Function

Discussion from industry statisticians prompted the next discussion. Essentially, what if we only desire one unknown instead of two? This section details how to find a minimum necessary sample size in the case where we have a fixed loss function constant and still want to reach a desired EBP and EBSL. Algorithm 3.4 shows how we accomplish this.

Algorithm 3.4: When specifying the desired EBSL, α_0, EBP, $1-\beta_0$, loss function constant, c, and T, this algorithm finds the approximate solution t_T^* it is exists; if it does not, it finds a candidate solution greater than T

1: Run Algorithm 3.1.
2: Run Algorithm 3.2.
3: Choose the larger sample size of the two provided by the two algorithms.

It seems to reason that the smaller of the two sample sizes would only meet one of the two operating characteristics. This can be seen in the example using the same prior structure covered throughout the chapter. If we fix our loss function constant at $c = 1$, the first algorithm to control for EBP gives that a sample size of 40 produces an approximate EBP of 0.801 and an approximate EBSL of 0.060. When running the second algorithm to control for EBSL, we find that a sample size of 54 produces an approximate EBSL of 0.050 with an approximate EBP of 0.819. Thus, we need to take the larger of the two values because it is the one that meets both criteria. This will hold for varying values of c.

Next, Table 3.4 displays various results using this approach for different gamma priors where we fix our desired EBSL, EBP, and loss function constant such that

Table 3.4: Various results for Poisson sample size determination when controlling EBSL, EBP, and setting $c = 1$

$\lambda = \lambda_1$		λ_2		EBP	EBSL	t	time
α_1	β_1	α_2	β_2				
8	4	4	4	0.819	0.050	54	1
4	4	8	4	0.823	0.049	57	1
1	1	1.5	1	0.801	0.042	60	5
10	10	15	10	0.807	0.050	183	2
1	1	1.7	1	0.801	0.043	51	7
10	10	17	10	0.839	0.050	104	1
1	1	1.9	1	0.802	0.045	43	3
10	10	19	10	0.874	0.050	61	<1

$c = 1$. We show the results and operating characteristics using a desired EBSL of 0.05, a desired EBP of 0.7, and $\pi_0 = 0.6$. The last column in the table is the time, in minutes, that the algorithm takes to run.

It can be clearly seen that some of the operating characteristics are not only met, but well exceeded. This is a product of only having one unknown with two equations; because we have fixed the other three values, there might not be an intersection where we barely eclipse both operating characteristics.

3.8 Conclusion

To recap, we have used conjugate prior structures in order to assess our beliefs about a rate parameter in a two sample Poisson trial a priori in order in to find the minimal sample size needed to reach certain operating characteristics. By the use of a loss function constant, we are able to control for at least two properties between how much worse a Type I error is in relation to a Type II error, desired expected Bayesian significance level, and desired expected Bayesian power. This type of analysis had never been considered previously.

It is of note that we were not able to consider the use of analysis priors in this research. Ideally, we would be able to adapt this research to account for the fact that researchers often times use one set of priors when conducting sample size

analyses, but a more vague or non-informative set of priors when actually analyzing the experiment. This adaptation would provide better sample size and operating characteristic estimates, though for now we are limited to the case where we use the same prior structure throughout. Further, this process could be expanded to consider non-conjugate priors as well. However, the analytical tractability of conjugate priors made it an ideal use, and modeling prior beliefs of a Poisson rate with a gamma distribution is not an unreasonable thing to do.

It also should be noted that time considerations, while already improved throughout the process, can always continue to improve. One improvement to current methods involve replacing the bi-sectional approaches described in Algorithms 3.2 and 3.2 with one that approximates the EBP curve with some logarithmic function; this improvement should get us in the ballpark of a candidate solution much quicker. Future work also includes a more in depth look at how expected Bayesian error rates compare to typical frequentist ones, and potentially an in-depth look at the different sample size determination and testing methods in order to determine the relative advantages and disadvantages of each. Further, we could generalize the algorithm such that we are not looking at a common sample size $t = t_1 = t_2$, but rather two different sample sizes t_1 and t_2 such that they do not need to be equal.

Lastly, while the general code used for sample size determination can be found in the Appendices, the entire package will be made available for download soon in the software package R. It will be able to not only handle sample size determination for both the Poisson and binomial cases, but also provide various graphics along with the actual test that would be conducted after data collection.

CHAPTER FOUR

Underreporting in Mixed Treatment Comparisons Meta-Analysis for Poisson Data

4.1 Introduction

Meta-analysis has become a popular method in statistics because of its ability to combine separate trials in order to make comparisons about parameters. This enables us to take several studies that might have smaller sample sizes and use them in a way so that we can use all of the relevant data at hand in analyses. Recently, efforts have been made to extend this to mixed treatment comparisons meta-analysis, which is also known as indirect comparisons or network meta-analysis. The idea behind this is that we can make comparisons on drugs that have not actually been directly compared. For instance, in a traditional meta-analysis setting, all of the drugs that we want to compare need to have been compared in each study. In mixed treatment comparisons meta-analysis, we can take several studies that only compare some of the drugs and, by combining the data, are able to make comparisons among all the treatments.[1] Most methods of meta-analysis use a generalized linear model, providing data models for normal, binomial, multinomial and Poisson data. In this chapter, we focus on Poisson data where we use the typical log link function. We extend the general indirect comparisons problem to the case where there is underreporting present, considering estimation from the Bayesian perspective.

Whittemore and Gong (1991) accounted for misclassification in classical Poisson regression, which was an early look at how to handle data that lacks accuracy. Stamey et al. (2008) and Powers et al. (2010) extended these efforts to the Bayesian paradigm. However, none of these considered meta-analysis. Bayesian approaches

[1] Example: If we only have studies comparing drugs A to B and A to C, we can use this methodology to compare B and C despite those two never having been directly compared.

to meta-analysis and mixed treatment comparisons meta-analysis has been well developed; see, for instance, Smith et al. (1995), Cooper et al. (2006), Jansen et al. (2011), and Dias et al. (2012). Though these authors consider several different models and scenarios, one area that has not been researched as much is the case of inaccurate data, such as the Poisson model with underreporting of Whittemore and Gong (1991). Here, we combine both of these ideas in order to create a methodology for handling mixed treatment comparisons meta-analysis when underreporting is present. Even though we only consider this problem for Poisson data, the extension to other data types should be straightforward.

This chapter is arranged in the following way. First, we outline the typical Bayesian mixed treatment comparisons meta-analysis model. We then move on to the case where we add a constant underreporting rate for each treatment. Lastly, we add a different underreporting rate for each treatment to the model and show how differing circumstances in data specifications can alter results.

4.2 Traditional Mixed Treatment Comparisons Meta-Analysis

4.2.1 Model

Suppose we have N_S two-arm Poisson data studies involving N_T treatments. We define $i = 1, ..., N \equiv 2N_S$ as the outcome number[2] and r_i the observed count such that

$$r_i \sim \text{Poisson}(\lambda_i).$$

Following Jansen et al. (2011), we assume the λ_i can be modeled by

$$\log(\lambda_i) = \log(y_i) + \mu_{s_i} + \delta_i I\{t_i \neq b_i\},$$

where y_i is the person-year contribution of the subjects, μ_{s_i} represents the baseline effect for the s^{th} study, $s = 1, ..., N_S$ represents the study number, δ_i represents

[2] Thus, for the first study, $i = 1$ would represent the first study and first treatment; $i = 2$ would represent the first study and second treatment, etc.

the random treatment effect of outcome i, $t_i = 1, ..., N_T$ represents the treatment number, and $b_i = 1, ..., N_T - 1$ denotes the "base" treatment number for that study.[3] For the purposes of this chapter, we focus on the case where the placebo is in every study; however, this methodology could easily be adapted to when that is not the case.

We treat the baseline rates, denoted μ_{s_i}, as independent nuisance parameters. These are given independent diffuse normal priors,

$$\mu_{s_i} \sim N(0, 0.005),$$

where the second term is the precision. For the treatment effect,

$$\delta_i \sim N(\theta_i, \tau)$$

where the prior on the precision is induced from that of the standard deviation, specifically,

$$\frac{1}{\sqrt{\tau}} \sim U(0, 3).$$

We focus on the random effects model here, but the fixed effects model can be recovered by setting $\tau = 0$. Further,

$$\theta_i = \varphi_{t_i} - \varphi_{b_i},$$

where this represents the difference between the i^{th} treatment and the baseline treatment for that study where $\varphi[1] = 0$. Because we are only concerning ourselves with the case where each study's baseline is a placebo (that will be designated as treatment one), this yields the simplification that

$$\theta_i = \varphi_{t_i},$$

[3] It is highly unlikely that a large number of treatments will be considered the baseline, so there should actually be less than $N_T - 1$ baselines.

though as we noted, this does not necessarily need to be the case. Thus for this case, the distribution of the δ_i is

$$\delta_i \sim N(\varphi_{t_i}, \tau).$$

For the φ_{t_i}'s, we assume that

$$\varphi_{t_i} \sim N(0, 0.005).$$

4.2.2 Example

To illustrate the problem, we begin with an example. For this first example we assume we are comparing four active drugs to a placebo so that $N_T = 5$. We use the program R to generate a data set such that each treatment of the four has a random number of studies conducted on it. We generate a random value from a normal distribution centered at ten with a standard deviation of two, and round the result for each treatment. If any of the values are less than 4, we adjust this value so that there are four studies for that treatment. We also generate a random number of person years for each study such that

$$y_i \sim N(250, 0.0004),$$

where 0.0004 is again the precision. We set an average log placebo effect of -2, which corresponds to μ_{s_i}. We also set average log treatment effects of -0.25, -0.5, -0.75, and -1 for the four treatments, which correspond to the five values for φ_{t_i} where, again, $\varphi[1] = 0$. This creates relative risks seen in the Table 4.1.

Table 4.1: Relative risks for mixed treatment comparisons meta-analysis example

	Treatment 2	Treatment 3	Treatment 4	Treatment 5
Treatment 2	1	0.779	0.607	0.472
Treatment 3		1	0.779	0.607
Treatment 4			1	0.779
Treatment 5				1

Note that the first treatment is the placebo. We then generate a treatment effect from the average treatment effects listed above such that

$$\delta_i \sim N(\varphi_{t_i}, 100).$$

We then compute the true λ_i for this data set and use that to generate random Poisson counts.

For example, one randomly generated data set creates 9, 10, 10 and 12 studies for the four treatments, respectively. The person years for those 41 studies ranges from 117 to 386 with a median of 260. The randomly generated Poisson counts range from 5 to 59, with a median of 24.5, a mean of 26.7, and a standard deviation of 12.8. The R code used to generate this data set is available in Appendix D. The data is then analyzed in WinBUGS, where the model is available in Appendix E.[4] We run a chain of length 60,000 where the first 10,000 iterations are discarded as the burn-in and we thin by 5 in order to leave us with a chain of 10,000 iterations used for inference. We can then summarize that chain by finding posterior credible intervals and summary statistics on all parameters of interest. The results for the run are listed in Table 4.2.

Table 4.2: Simulation results of one run of the traditional model

Parameter	Mean	2.5% Percentile	Median	97.5% Percentile	Interval Width
RR[2,3]	0.759	0.564	0.750	1.009	0.445
RR[2,4]	0.636	0.468	0.629	0.857	0.389
RR[2,5]	0.569	0.400	0.561	0.784	0.384
RR[3,4]	0.848	0.622	0.836	1.139	0.517
RR[3,5]	0.758	0.530	0.750	1.041	0.511
RR[4,5]	0.905	0.631	0.894	1.238	0.607
σ	0.182	0.016	0.184	0.347	0.331

When comparing the results to the table of relative risks that we are trying to estimate, it is clear that the procedure has worked well, as every interval has

[4] All models used in this chapter are available in this appendix.

captured the true value of the parameter. However, because we have randomly generated some values, we run this same procedure 500 times. This allows us to assess the variability and accuracy of the procedure. Note that each new generated data set will have a different number of studies and person years per study for each treatment, but the mean relative risks and the random effects standard deviation will remain the same. Table 4.3 summarizes the results.

Table 4.3: Simulation results for the traditional model

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.787	0.795	0.452	0.982
RR[2,4]	0.607	0.605	0.612	0.361	0.954
RR[2,5]	0.472	0.475	0.480	0.293	0.956
RR[3,4]	0.779	0.776	0.785	0.476	0.952
RR[3,5]	0.607	0.609	0.616	0.385	0.954
RR[4,5]	0.779	0.794	0.805	0.520	0.954
σ	0.1	0.123	0.128	0.268	0.984

The operating characteristics in the table resemble those that we would hope to see. This is not a surprising result, but this simulation experiment illustrates that if no underreporting is present, posterior medians are close to the true parameter values and coverage of the 95% intervals are close to nominal.

4.3 Constant Underreporting in Mixed Treatment Comparisons Meta-Analysis

4.3.1 Model

Next, we consider how the model is affected by underreporting. For the first scenario that we consider where we allow for underreporting, we assume the underreporting probability is the same for all treatments. We use the same general notation as before, but add another parameter: the reporting probability, p. We assume a prior distribution on p such that

$$p \sim \text{beta}(\alpha, \beta).$$

Now λ_i can be modeled by

$$\log(\lambda_i) = \log(p) + \log(y_i) + \mu_{s_i} + \delta_i I\{t_i \neq b_i\},$$

where everything else remains as notated previously.

4.3.2 Example

We continue with the same specifications from the previous example, but impose underreporting on each treatment; we use a reporting rate of 0.7. In order to see how analyses with underreporting work, we analyze the underreported data with the method previously considered and the new method just described where we place a

$$p \sim \text{beta}(42, 18)$$

prior on the reporting rate. Thus our prior has a prior mean of 0.7 and is centered around the truth, but has a prior 95% interval of (0.58, 0.81). This prior is relatively informative as it has a prior equivalent sample size of 60 observations where 42 were reported and 18 were unreported.

We again run chain lengths of 60,000 where we discard the first 10,000 iterations as a burn-in and thin by 5 in order to leave us with a chain of 10,000 iterations, repeating this for 500 data sets. The summaries of the incorrect analysis of the data is provided in the Table 4.4, while the summaries from the correctly analyzed data is displayed in the Table 4.5.

It should be noted that the results are not only similar, but both are accurate. This is because we are interested in relative risks, which compare two treatments against each other. If both treatments in the comparison have the same reporting rate, the fact that we are not receiving all events will cancel out in the comparison. Thus, we can conclude that if the reporting rates are believed to be similar across all treatments, analyses can be made using traditional methods.

Table 4.4: Simulation results for constant underreporting analyzed with none

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.776	0.787	0.528	0.958
RR[2,4]	0.607	0.603	0.612	0.427	0.960
RR[2,5]	0.472	0.472	0.480	0.349	0.966
RR[3,4]	0.779	0.789	0.803	0.577	0.958
RR[3,5]	0.607	0.618	0.629	0.469	0.952
RR[4,5]	0.779	0.793	0.808	0.624	0.966
σ	0.1	0.142	0.150	0.318	0.974

Table 4.5: Simulation results for constant underreporting analyzed correctly

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.788	0.799	0.530	0.958
RR[2,4]	0.607	0.610	0.619	0.426	0.958
RR[2,5]	0.472	0.473	0.481	0.345	0.948
RR[3,4]	0.779	0.785	0.797	0.564	0.960
RR[3,5]	0.607	0.610	0.620	0.459	0.954
RR[4,5]	0.779	0.789	0.804	0.612	0.958
σ	0.1	0.133	0.141	0.309	0.992

4.4 Varying Underreporting in Mixed Treatment Comparisons Meta-Analysis

4.4.1 Model

Next, we allow for varying underreporting across the treatments. We assume that every treatment (including placebo) has a different level of underreporting in an attempt to see if this form of misclassification will create innacurate results by not accounting for it. We use the same general notation as before, but now we say that each treatment has a reporting probability, p_{t_i}, with a prior distribution on it such that

$$p_{t_i} \sim \text{beta}\left(\alpha_{t_i}, \beta_{t_i}\right).$$

Now λ_i can be modeled by

$$\log\left(\lambda_i\right) = \log\left(p_{t_i}\right) + \log\left(y_i\right) + \mu_{s_i} + \delta_i I\left\{t_i \neq b_i\right\}.$$

However, for this case, we require a slightly less diffuse prior structure for a few of the parameters in order for the simulations to converge each time. Thus, we give

the baseline rates independent diffuse normal priors of

$$\mu_{s_i} \sim N(0, 0.05),$$

the baseline treatment effects priors of

$$\varphi_{t_i} \sim N(0, 0.05),$$

and the prior on the standard deviation a prior of

$$\frac{1}{\sqrt{\tau}} \sim U(0.001, 2).$$

This informativeness is expected to have little effect on results, but allows each MCMC simulation to converge.

4.4.2 Example

We continue with the same basic specifications as before, but change the reporting probabilities. For the placebo, we give it a reporting probability of 0.8. For the remaining four treatments, we place reporting probabilities of 0.75, 0.7, 0.65 and 0.6. For all five priors, we place priors that have an equivalent sample size of 60 that are centered at the true reporting probability, similar to how we did in the previous case. However, in an effort to see how the new methods compare, we analyze the same specifications using both previous methods and the new one. We again run chain lengths of 60,000 where we discard the first 10,000 iterations as a burn-in and thin by 5 in order to leave us with a chain of 10,000 iterations, repeating this for 500 data sets. Table 4.6 shows the results when the data was analyzed ignoring the underreporting.

Notice the considerable bias for a few of the parameter estimates and how some do not come anywhere close to nominal coverage on a 95% credible interval. Also, note that the further away the relative risk is from one, the worse the method performs.[5] Next, we analyze the data using the previous method using constant

[5] A quick run of relative risks greater than one showed the same phenomenon.

Table 4.6: Simulation results for split underreporting analyzed traditionally

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.740	0.750	0.485	0.944
RR[2,4]	0.607	0.538	0.545	0.369	0.906
RR[2,5]	0.472	0.379	0.385	0.280	0.762
RR[3,4]	0.779	0.736	0.748	0.524	0.940
RR[3,5]	0.607	0.519	0.528	0.395	0.862
RR[4,5]	0.779	0.717	0.731	0.569	0.934
σ	0.1	0.131	0.139	0.305	0.986

Table 4.7: Simulation results for split underreporting analyzed as if constant

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.739	0.749	0.482	0.930
RR[2,4]	0.607	0.527	0.535	0.364	0.888
RR[2,5]	0.472	0.381	0.387	0.277	0.764
RR[3,4]	0.779	0.722	0.734	0.517	0.940
RR[3,5]	0.607	0.522	0.531	0.393	0.868
RR[4,5]	0.779	0.733	0.748	0.581	0.926
σ	0.1	0.132	0.140	0.305	0.992

underreporting. Thus, even though they have different underreporting values, we place a beta $(42, 18)$ prior[6] on the constant underreporting value.Table 4.7 shows the results of the analysis across the 500 data sets using the same chain specifications as before.

Notice the poor results should be expected given that these two methods performed similarly in the constant underreporting case. Table 4.8 show the results when we analyze the data correctly using the split prior approach.

As can be seen, analyzing with the correct method leads to better results and operating characteristics. Further, the fact that our 95% credible sets actually have better than 95% coverage is not only surprising, but should be expected. Because we have so much uncertainty, our intervals are fairly conservative, allowing us to capture the true parameters more often. Stamey et al. (2007) provide an example where this phenomenon was observed.

[6] This is the average of all the underreporting values.

Table 4.8: Simulation results for split underreporting analyzed correctly

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.798	0.814	0.649	0.990
RR[2,4]	0.607	0.609	0.623	0.522	0.990
RR[2,5]	0.472	0.478	0.491	0.439	0.980
RR[3,4]	0.779	0.774	0.793	0.691	0.988
RR[3,5]	0.607	0.607	0.624	0.578	0.990
RR[4,5]	0.779	0.797	0.821	0.791	0.990
σ	0.1	0.138	0.146	0.310	0.976

4.4.3 Various Results

Next, we generate different data sets in order to see how changing certain characteristics affects results. First, we increase the number of active drugs to eight so that there are nine total treatments. The reporting probabilities in this case still range from 0.6 to 0.8, but this time with intervals of 0.025. We again run chain lengths of 60,000 where we discard the first 10,000 iterations as a burn-in and thin by 5 in order to leave us with a chain of 10,000 iterations, repeating this for 500 data sets. Table 4.9 shows that changing the number of drugs does not appear to have a negative effect, as again, the posterior medians are close to the truth and the coverage is again conservative.

Next, we change the standard deviation on the prior of our random effect from 0.1 to 0.4 and return to the case of only four drugs (plus the placebo). Table 4.10 displays the results of the analysis across the 500 data sets using the same chain specifications as before. As can be seen, we have slightly wider intervals, and the posterior medians are slightly farther from the truth, but that is to be expected. We still have good coverage properties, leading us to believe that increasing the random effects standard deviation increases the uncertainty, but does not seem to be detrimental to the estimation procedure.

Next, we change the number of studies that we generate from. Previously, we were generating data such that each treatment had roughly ten studies and floored

Table 4.9: Simulation results for split underreporting and increased treatments

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.882	0.890	0.905	0.642	0.988
RR[2,4]	0.779	0.786	0.800	0.583	0.990
RR[2,5]	0.687	0.689	0.702	0.528	0.986
RR[2,6]	0.607	0.617	0.629	0.488	0.990
RR[2,7]	0.535	0.537	0.548	0.438	0.988
RR[2,8]	0.472	0.476	0.487	0.401	0.974
RR[2,9]	0.417	0.421	0.432	0.372	0.992
RR[3,4]	0.882	0.891	0.907	0.677	0.986
RR[3,5]	0.779	0.782	0.797	0.613	0.984
RR[3,6]	0.687	0.699	0.713	0.564	0.990
RR[3,7]	0.607	0.609	0.623	0.508	0.992
RR[3,8]	0.535	0.540	0.553	0.464	0.982
RR[3,9]	0.472	0.478	0.490	0.429	0.996
RR[4,5]	0.882	0.870	0.905	0.713	0.988
RR[4,6]	0.779	0.792	0.809	0.654	0.986
RR[4,7]	0.687	0.691	0.706	0.589	0.994
RR[4,8]	0.607	0.612	0.627	0.536	0.990
RR[4,9]	0.535	0.542	0.556	0.496	0.992
RR[5,6]	0.882	0.905	0.925	0.768	0.992
RR[5,7]	0.779	0.790	0.809	0.692	0.990
RR[5,8]	0.687	0.701	0.718	0.627	0.974
RR[5,9]	0.607	0.620	0.637	0.579	0.986
RR[6,7]	0.882	0.884	0.906	0.788	0.988
RR[6,8]	0.779	0.784	0.804	0.718	0.982
RR[6,9]	0.687	0.694	0.713	0.662	0.988
RR[7,8]	0.882	0.902	0.927	0.852	0.974
RR[7,9]	0.779	0.798	0.822	0.784	0.980
RR[8,9]	0.882	0.902	0.930	0.904	0.982
σ	0.1	0.113	0.117	0.234	0.988

Table 4.10: Simulation results for split underreporting and increased random effect standard deviation

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.805	0.836	0.904	0.976
RR[2,4]	0.607	0.629	0.654	0.730	0.978
RR[2,5]	0.472	0.488	0.509	0.588	0.978
RR[3,4]	0.779	0.804	0.839	0.957	0.980
RR[3,5]	0.607	0.624	0.652	0.769	0.980
RR[4,5]	0.779	0.799	0.838	1.019	0.986
σ	0.4	0.409	0.415	0.372	0.930

Table 4.11: Simulation results for split underreporting and decreased studies

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.790	0.826	0.971	0.996
RR[2,4]	0.607	0.621	0.652	0.806	0.970
RR[2,5]	0.472	0.473	0.498	0.646	0.988
RR[3,4]	0.779	0.805	0.849	1.084	0.999
RR[3,5]	0.607	0.614	0.649	0.867	0.992
RR[4,5]	0.779	0.789	0.841	1.189	0.986
σ	0.1	0.170	0.188	0.452	0.986

our number of studies at four. Now we generate data such that each treatment has roughly five studies and floor the number of studies at two.[7] Table 4.11 shows the results of the analysis across the 500 data sets using the same chain specifications as before. As before, we see a slight increase in interval width here due to the added uncertainty of small study sizes. However, the method still seems to be fairly robust against this change.

Next, we change the reporting probabilities for each treatment to be more spread out. Rather than ranging from 0.6 to 0.8, we spread the probabilities from 0.5 to 0.9 (where, again, the placebo is the highest). Table 4.12 shows how this change affects the results of the analysis across the 500 data sets using the same chain specifications as before. As can be seen, this change also does not affect our results in a negative way.

Table 4.12: Simulation results for split and spead underreporting

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.789	0.804	0.614	0.988
RR[2,4]	0.607	0.610	0.624	0.524	0.998
RR[2,5]	0.472	0.479	0.493	0.466	0.988
RR[3,4]	0.779	0.784	0.804	0.715	0.990
RR[3,5]	0.607	0.614	0.634	0.626	0.990
RR[4,5]	0.779	0.796	0.824	0.866	0.994
σ	0.1	0.133	0.141	0.304	0.986

[7] We keep the standard deviation of number of studies generated per treatment at two.

Table 4.13: Simulation results for split underreporting (smaller prior sample size)

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.791	0.813	0.750	0.996
RR[2,4]	0.607	0.614	0.633	0.619	0.994
RR[2,5]	0.472	0.475	0.493	0.546	0.996
RR[3,4]	0.779	0.786	0.813	0.831	0.996
RR[3,5]	0.607	0.609	0.633	0.688	0.998
RR[4,5]	0.779	0.785	0.819	0.930	1.000
σ	0.1	0.135	0.142	0.308	0.986

Next, we consider a change to our prior structures. Namely, what happens if we have a different prior sample size for our reporting probabilities, p_i? In order to keep results comparable, we have been using a sample size of 60 throughout, but next we consider the case where our sample size is only 30. Table 4.13 shows the results of the analysis across the 500 data sets using the same chain specifications as before. As can be seen, a less informative prior does have some effect on results; we see a slight increase in interval width for each relative risk as well as one in coverage.

Because this decrease in prior sample size only appears to affect posterior variability, we run the analysis again with a prior sample size of 15. Table 4.14 shows the results of the analysis across the 500 data sets using the same chain specifications as before. We see the same effect happening here, as our intervals continue to get wider as our coverage increases. This can be explained by the fact that we are overparameterized. Thus, as we have less and less informativeness in our prior structures, that indecision creates more conservative estimates. That, in turn, allows us to capture the truth more often. It is a safe assumption to assume that the opposite effect would occur as we increase our prior sample size beyond 60. As our certainty for p_i increases, we would see coverage around a nominal level with an even narrower interval.

Lastly, we consider what happens when reporting probabilities are misspecified. Obviously, a large amount of misspecification will hurt results; the idea here

Table 4.14: Simulation results for split underreporting (smallest prior sample size)

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.791	0.827	0.960	1.000
RR[2,4]	0.607	0.616	0.649	0.809	0.996
RR[2,5]	0.472	0.475	0.505	0.678	1.000
RR[3,4]	0.779	0.787	0.831	1.076	1.000
RR[3,5]	0.607	0.607	0.647	0.903	0.998
RR[4,5]	0.779	0.782	0.837	1.212	1.000
σ	0.1	0.136	0.143	0.309	0.984

Table 4.15: Simulation results for split underreporting analyzed with constant misspecification

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.788	0.803	0.613	0.986
RR[2,4]	0.607	0.604	0.617	0.496	0.980
RR[2,5]	0.472	0.463	0.475	0.411	0.976
RR[3,4]	0.779	0.778	0.795	0.669	0.980
RR[3,5]	0.607	0.596	0.612	0.550	0.980
RR[4,5]	0.779	0.780	0.802	0.748	0.980
σ	0.1	0.137	0.144	0.311	0.988

is to test how robust to minor changes these methods are. For the first change, we affect the prior probabilities in such a way that the change is constant among treatments. For example, the five reporting probabilities for the general case we have been outlining are 0.8, 0.75, 0.7, 0.65, and 0.6. Thus, the prior structures for these are a beta$(48, 12)$, beta$(45, 15)$, beta$(42, 18)$, beta$(39, 21)$, and beta$(36, 24)$, respectively.

However, we run a first analyses using prior structures of a beta$(51, 9)$, beta$(48, 12)$, beta$(45, 15)$, beta$(42, 18)$, and beta$(39, 21)$, respectively. This constant misspecification has no tangible effect on results, as can be seen in Table 4.15.[8]

Next, we use prior structures that affect our reporting probabilities differently. Namely, we use a beta$(51, 9)$, beta$(42, 18)$, beta$(45, 15)$, beta$(36, 24)$, and beta$(39, 21)$, respectively. Notice that this change affects treatments in different

[8] As before, when the analysis deviated from the truth by a constant rate across all treatments, this misspecification essentially is wiped out in relative risks because of the division of rates.

Table 4.16: Simulation results for split underreporting analyzed with varied misspecification

Parameter	Actual RR	Median	Mean	Interval Width	Coverage
RR[2,3]	0.779	0.682	0.696	0.546	0.942
RR[2,4]	0.607	0.618	0.634	0.550	0.990
RR[2,5]	0.472	0.408	0.418	0.368	0.956
RR[3,4]	0.779	0.918	0.941	0.821	0.950
RR[3,5]	0.607	0.606	0.621	0.549	0.990
RR[4,5]	0.779	0.671	0.691	0.662	0.942
σ	0.1	0.130	0.138	0.304	0.996

manners, as these priors are centered at 0.85, 0.7, 0.75, 0.6, and 0.65. Thus, the placebo and two of the treatments are believed to have reporting probabilities that are 5% higher than the truth; the other two treatments are believed to have reporting probabilities 5% less than the truth. Table 4.16 summarizes these results.

As can be seen from the table, these methods are still fairly robust to misspecifications of reporting probabilities. However, we can see some shortcomings. The methods that analyze treatments that are both overspecified (or under, of course)[9] still have very good coverage rates. However, in other cases we are seeing a ten percent difference in reporting probabilities because one treatment is under by five percent and the one it is compared to is over by five percent. This is hurting our coverage probability, and it seems reasonable to think that as this gap increases, coverage will continue to decrease. However, this misspecification is still fairly large and a much better case than when analyzing without underreporting, leading to the conclusion that even if an accurate prior cannot be specified in this case, a slightly inaccurate prior is better than ignoring the underreporting

4.5 Checking MCMC Assumptions

We begin our Markov chain Monte Carlo assumptions check with the first case highlighted in this chapter; this is the case where there is no underreporting in the

[9] When even numbered treatments are compared to even numbered treatments and vice versa.

data and we analyze using traditional methods. This should serve as a baseline for what we hope to see in our plots. Recall that when we generate a data set, we had 41 overall studies with studies of 9, 10, 10 and 12 for the four treatments, respectively. The person years ranged from 117 to 386, while the randomly generated Poisson counts ranged from 5 to 59. After running the data through WinBUGS using a chain of length 60,000 where the first 10,000 iterations are discarded as the burn-in and we thin by 5 in order to leave us with a chain of 10,000 iterations, we arrive at three different plots that will aid in checking convergence. The first plot is a history plot that shows each iteration of the chain; here we are hoping to see random variation about the point where the chain has converged.[10] The next plot is a density plot that shows the rough distribution of the chain; because we are using the percentiles as cutoffs for a credible interval, we hope to see a fairly smooth distribution here.[11] The last plot is an autocorrelation plot; here we hope to see no evidence of correlation between future observations.[12] All three of these plots can be seen in Appendix G.

Next, we generate data that comes from the split underreporting model such that we have different reporting probabilities for each treatment. This data set has the same number of studies as before, along with the same set of person years attached to each study. However, because of the difference in reporting probabilities, our randomly generated Poisson counts now range from 3 to 48. The median, mean and standard deviation all decreased as well. Appendix H shows the assumption check plots for this data set analyzed without accounting for underreporting. Appendix I shows the assumption check plots for this data set analyzed using constant underreporting. Appendix J shows the assumption check plots for this data set analyzed using the correct split underreporting model. Appendix K shows the

[10] Failure here would be evidence that the chain has not yet converged.

[11] Failure here would also indicate that the chain has not converged yet.

[12] Failure here would be evidence that we did not discard enough observations in the thinning process.

assumption check plots for this data set analyzed using the incorrect split underreporting model. Note that for every set of plots we have chosen the three relative risks that have the worst appearance; thus, these should be a good representative look at the worst these methods can perform, as relative risks not presented looked better.

It should be noted that the only potential concern with assumptions lie in the odd shape of the posterior standard deviation; ideally, this would be much smoother. However, our primary interest is in the relative risks, so we look at how various priors on σ affects these rates. In an attempt to investigate this, we altered the prior used on our standard deviation from

$$\frac{1}{\sqrt{\tau}} \sim U(0.001, 2)$$

to one of two different priors. The first is

$$\frac{1}{\sqrt{\tau}} \sim U(0.001, 1),$$

while the second is

$$\frac{1}{\sqrt{\tau}} \sim U(0.001, 4).$$

Tables 4.17 and 4.18 display the results of one run of both cases using the same simulated data set in tabular form to show that the relative risks are static to this change.

The effect of prior on the standard deviation can be clearly evidenced in the difference in the tables; however, the relative risks are virtually identical, indicating to us that this is not an issue.

Thus, due to the fact that even the worst plots look satisfactory, there is nothing here that would indicate that our chains have not converged to a stationary distribution and that analyses are correct.

Table 4.17: Simulation results of one run for MCMC convergence check for more informative prior

Parameter	Mean	2.5% Percentile	Median	97.5% Percentile	Interval Width
RR[2,3]	0.657	0.409	0.642	0.979	0.570
RR[2,4]	0.606	0.383	0.588	0.936	0.553
RR[2,5]	0.369	0.229	0.359	0.563	0.334
RR[3,4]	0.944	0.586	0.921	1.436	0.850
RR[3,5]	0.576	0.356	0.562	0.886	0.530
RR[4,5]	0.627	0.381	0.609	0.971	0.590
σ	0.199	0.104	0.202	0.399	0.295

Table 4.18: Simulation results of one run for MCMC convergence check for less informative prior

Parameter	Mean	2.5% Percentile	Median	97.5% Percentile	Interval Width
RR[2,3]	0.661	0.418	0.647	0.987	0.569
RR[2,4]	0.606	0.388	0.591	0.909	0.521
RR[2,5]	0.368	0.227	0.358	0.563	0.336
RR[3,4]	0.940	0.591	0.916	1.438	0.847
RR[3,5]	0.569	0.353	0.554	0.884	0.531
RR[4,5]	0.623	0.375	0.606	0.978	0.603
σ	0.212	0.022	0.214	0.411	0.389

4.6 Conclusion

To recap, we have adjusted the traditional mixed treatment comparisons meta-analysis to account for various types of underreporting. We illustrated how constant underreporting has no effect on relative risk calculations, but failing to account for it when treatments have different level of underreporting can lead to serious errors. We showed several cases various parameters were altered in order to demonstrate the robustness of this method, including analyzing the data with incorrect prior structures in an attempt to show that misspecifying an underreporting prior is still preferred to analyzing with no underreporting.

It should be noted that these methods can easily be extended to the case where not every treatment has a separate underreporting. For instance, if there are six treatments, but it is believed that there are only three different reporting probabilities attached to those, we can easily pool those together to improve performance.

Because that is simply a specific case of what we outlined in this chapter, we chose not to include it. Further, it should be noted that future work includes accounting for varying types of misspecification with other data models.

CHAPTER FIVE

Conclusion

In this dissertation we have investigated and contributed to two important problems in statistical science − sample size determination for select test of statistical hypothesis and underreporting in mixed treatment comparisons meta-analysis. Here we recap some of the contributions we provide and include future directions that may prove useful.

As covered in introductory chapter, sample size determination has become a central issue in applied statistics. Thus, we began this dissertation by introducing improvements to methods for sample size determination in situations where both prior information and knowledge of decision consequences are available. Though these methods were applied from a Bayesian perspective, we kept traditional methods of decision quality in mind; namely, those of significance level and power.[1] This type of consideration had previously been ignored in Bayesian sample size determination problems. Additionally, we have provided novel algorithms and implementations that can be used to handle real-world problems. However, it should be noted that several improvements to these contributions can still be made, including optimization of the fundamental algorithms, adjustment of prior structures, and the extension to other data models besides the binomial and Poisson cases.

We concluded with a look at a mixed treatment comparisons meta-analysis problem. Previous research lacked a way to handle misspecified data, so we introduced a model that accounts for varying levels of underreporting within Poisson data. These methods were also applied from a Bayesian perspective and were shown to have drastic improvements over the current methodology. Several changes were

[1] Note that in this dissertation we analyze these from an expected sense.

made to the general structure of the problem to show that these methods are also fairly robust to moderate alterations. However, the extension to other data models that exhibit misspecification would be a valuable extension to the results demonstrated in this dissertation.

APPENDICES

APPENDIX A

R Code to Find EBP for a Fixed Sample Size

```
sample_power <- function (type, n, c, a1, b1, a2, b2, a = a1, b = b1,
  pi0 = 0.5, pi1 = 1 - pi0) {

  stopifnot(pi0 >= 0)
  stopifnot(pi1 >= 0)
  stopifnot(pi0 + pi1 == 1)
  type <- match.arg(type, c("poisson", "binomial"))

  if (type == "poisson") {
    y1min <- qnbinom(1e-04, size = a1, prob = b1/(n + b1))
    y2min <- qnbinom(1e-04, size = a2, prob = b2/(n + b2))
    y1max <- qnbinom(0.9999, size = a1, prob = b1/(n + b1))
    y2max <- qnbinom(0.9999, size = a2, prob = b2/(n + b2))
    df <- expand.grid(y1 = y1min:y1max, y2 = y2min:y2max)
    value <- function(y1, y2) {
      exp((y1 + y2) * log(n) + a1 * log(b1) + a2 *
        log(b2) + lgamma(y1 + a1) + lgamma(y2 + a2) -
        lfactorial(y1) - lfactorial(y2) - lgamma(a1) -
        lgamma(a2) - (y1 + a1) * log(n + b1) - (y2 +
        a2) * log(n + b2))
      }
    condition <- function(y1, y2) {
      a * log(b) + lgamma(y1 + y2 + a) - lgamma(a) - (y1 +
```

```
      y2 + a) * log(2 * n + b) + lgamma(a1) + lgamma(a2) +
      (y1 + a1) * log(n + b1) + (y2 + a2) * log(n + b2) -
      a1 * log(b1) - a2 * log(b2) - lgamma(y1 + a1) - lgamma(y2 +
      a2) <= log(c * pi1 / pi0)
    }
  reject <- function(y1, y2) {
    if (condition(y1, y2)) return(value(y1, y2)) else return(0)
    }
  f <- sum(apply(df, 1, function(v) value(v[1], v[2])))
  power <- sum(apply(df, 1, function(v) reject(v[1], v[2])))
  p <- power / f
  }

if (type == "binomial") {
  df <- expand.grid(y1 = 0:n, y2 = 0:n)
  value <- function(y1, y2) {
    exp(lchoose(n, y1) + lbeta(y1 + a1, n - y1 + b1) -
      lbeta(a1, b1) + lchoose(n, y2) + lbeta(y2 + a2,
      n - y2 + b2) - lbeta(a2, b2))
    }
  condition <- function(y1, y2) {
    lbeta(y1 + y2 + a, 2 * n - y1 - y2 + b) - lbeta(a, b) -
      lbeta(y1 + a1, n - y1 + b1) + lbeta(a1, b1) -
      lbeta(y2 + a2, n - y2 + b2) + lbeta(a2, b2) <=
      log(c * pi1 / pi0)
    }
  reject <- function(y1, y2) {
```

```
    if (condition(y1, y2)) return(value(y1, y2)) else return(0)
    }
  p <- sum(apply(df, 1, function(v) reject(v[1], v[2])))
  }

p
}
```

APPENDIX B

R Code to Find EBSL for a Fixed Sample Size

```
sample_alpha <- function (type, n, c, a1, b1, a2, b2, a = a1, b = b1,
  pi0 = 0.5, pi1 = 1 - pi0) {

  stopifnot(pi0 >= 0)
  stopifnot(pi1 >= 0)
  stopifnot(pi0 + pi1 == 1)
  type <- match.arg(type, c("poisson", "binomial"))

  if (type == "poisson") {
    ymin <- qnbinom(1e-04, size = a, prob = b/(n + b))
    ymax <- qnbinom(0.9999, size = a, prob = b/(n + b))
    df <- expand.grid(y1 = ymin:ymax, y2 = ymin:ymax)
    value <- function(y1, y2) {
      exp((y1 + y2) * log(n) + a * log(b) + lgamma(y1 + y2 + a) -
        lfactorial(y1) - lfactorial(y2) - lgamma(a) -
        (y1 + y2 + a) * log(2 * n + b))
      }
    condition <- function(y1, y2) {
      a * log(b) + lgamma(y1 + y2 + a) - lgamma(a) - (y1 +
        y2 + a) * log(2 * n + b) + lgamma(a1) + lgamma(a2) +
        (y1 + a1) * log(n + b1) + (y2 + a2) * log(n + b2) -
        a1 * log(b1) - a2 * log(b2) - lgamma(y1 + a1) - lgamma(y2 +
        a2) <= log(c * pi1 / pi0)
```

```
    }
  reject <- function(y1, y2) {
    if (condition(y1, y2)) return(value(y1, y2)) else return(0)
    }
  f <- sum(apply(df, 1, function(v) value(v[1], v[2])))
  alpha <- sum(apply(df, 1, function(v) reject(v[1], v[2])))
  a <- alpha / f
  }

if (type == "binomial") {
  df <- expand.grid(y1 = 0:n, y2 = 0:n)
  value <- function(y1, y2) {
    exp(lchoose(n, y1) + lbeta(y1 + y2 + a, 2 * n - y1 - y2 + b) -
      lbeta(a, b) + lchoose(n, y2))
    }
  condition <- function(y1, y2) {
    lbeta(y1 + y2 + a, 2 * n - y1 - y2 + b) - lbeta(a, b) -
      lbeta(y1 + a1, n - y1 + b1) + lbeta(a1, b1) -
      lbeta(y2 + a2, n - y2 + b2) + lbeta(a2, b2) <=
      log(c * pi1 / pi0)
    }
  reject <- function(y1, y2) {
    if (condition(y1, y2)) return(value(y1, y2)) else return(0)
    }
  a <- sum(apply(df, 1, function(v) reject(v[1], v[2])))
  }
```

```
a
}
```

APPENDIX C

R Code to Find Desired Sample Size

```
find_size <- function (type, a1, b1, a2, b2, a = a1, b = b1, pi0 = 0.5,
  pi1 = 1 - pi0, alpha = 0, power = 0, c = 0, c_tol = 0, N = 0) {

  if(c != 0) c <- 1 / c
  stopifnot (pi0 >= 0)
  stopifnot (pi1 >= 0)
  stopifnot (pi0 + pi1 == 1)
  if (alpha > 0 && power > 0 && c > 0) stop(
    "Cannot specify EBSL, EBP and c")
  if (alpha == 0 && power == 0 & c == 0) stop(
    "Must specify two of three between EBSL, EBP and c")
  if (alpha == 0 && power == 0) stop(
    "Must specify more than just c")
  if (alpha == 0 && c == 0) stop(
    "Must specify more than just EBP")
  if (c == 0 && power == 0) stop(
    "Must specify more than just EBSL")
  type <- match.arg(type, c("poisson", "binomial"))
  if(N == 0 && type == "poisson") check <- 50
  if(N == 0 && type == "binomial") check <- 100
  if(c_tol == 0) c_tol <- 0.005
```

```
if (alpha == 0) {
  n <- check
  n_power <- sample_power(type, n, c, a1, b1, a2, b2,
      a, b, pi0, pi1)
  if (n_power > power) {
    i <- 1
    while (n_power[i] > power) {
      n <- c(n, check - i)
      n_power <- c(n_power, sample_power(type, check - i, c,
        a1, b1, a2, b2, a, b, pi0, pi1))
      i <- i + 1
      }
    n <- n[i - 1]
    power <- n_power[i - 1]
    } else {
    n <- c(1, n)
    n_power <- c(0, n_power)
    while (n_power[2] < power) {
      n <- c(n[2], 2 * n[2])
      n_power <- c(n_power[2], sample_power(type, n[2], c,
        a1, b1, a2, b2, a, b, pi0, pi1))
      }
    while (abs(n[1] - n[2]) > 1) {
      n <- c(n, round(mean(n)))
      n_p <- sample_power(type, n[3], c, a1, b1, a2, b2,
        a, b, pi0, pi1)
      if (n_p < power) {
```

```
        n <- n[3:2]
        n_power <- c(n_p, n_power[2])
        } else {
        n <- n[-2]
        n_power <- c(n_power[1], n_p)
        }
      }
    n <- n[2]
    power <- n_power[2]
    }
  alpha <- sample_alpha(type, n, c, a1, b1, a2, b2,
    a, b, pi0, pi1)
  }

if (power == 0) {
  n <- check
  n_alpha <- sample_alpha(type, n, c, a1, b1, a2, b2,
      a, b, pi0, pi1)
  if (n_alpha < alpha) {
    i <- 1
    while (n_alpha[i] < alpha) {
      n <- c(n, check - i)
      n_alpha <- c(n_alpha, sample_alpha(type, check - i, c,
      a1, b1, a2, b2, a, b, pi0, pi1))
      i <- i + 1
      }
    n <- n[i - 1]
```

```
  alpha <- n_alpha[i - 1]
  } else {
  n <- c(1, n)
  n_alpha <- c(1, n_alpha)
  while (n_alpha[2] > alpha) {
    n <- c(n[2], 2 * n[2])
    n_alpha <- c(n_alpha[2], sample_alpha(type, n[2], c,
      a1, b1, a2, b2, a, b, pi0, pi1))
    }
  while (abs(n[1] - n[2]) > 1) {
    n <- c(n, round(mean(n)))
    n_a <- sample_alpha(type, n[3], c, a1, b1, a2, b2,
      a, b, pi0, pi1)
    if (n_a > alpha) {
      n <- n[3:2]
      n_alpha <- c(n_a, n_alpha[2])
      } else {
      n <- n[-2]
      n_alpha <- c(n_alpha[1], n_a)
      }
    }
  n <- n[2]
  alpha <- n_alpha[2]
  }
power <- sample_power(type, n, c, a1, b1, a2, b2,
  a, b, pi0, pi1)
}
```

```
find_c <- function(type, n, power, a1, b1, a2, b2, a, b, pi0, pi1) {
  c <- c(0, 1)
  c_power <- c(0, sample_power(type, n, 1, a1, b1, a2, b2,
    a, b, pi0, pi1))
  if(c_power[2] > power) {
    while (abs(c[1] - c[2]) > c_tol) {
      c <- c(c, mean(c))
      c_p <- sample_power(type, n, c[3], a1, b1, a2, b2,
        a, b, pi0, pi1)
      if (c_p < power) {
        c <- c[3:2]
        c_power <- c(c_p, c_power[2])
        } else {
        c <- c[-2]
        c_power <- c(c_power[1], c_p)
        }
      }
    }
  list(c = c[2], power = c_power[2])
  }


if (c == 0) {
  n <- 1
  n_power <- 0
  n_alpha <- 1
  while (n_alpha > alpha || n_power < power) {
```

```
    n <- n + 1
    new <- find_c(type, n, power, a1, b1, a2, b2,
      a, b, pi0, pi1)
    c <- new$c
    n_power <- new$power
    n_alpha <- sample_alpha(type, n, c, a1, b1, a2, b2,
      a, b, pi0, pi1)
    }
  alpha <- n_alpha
  power <- n_power
  }

c <- 1 / c
list(size = n, c = c, alpha = alpha, power = power)
}
```

APPENDIX D

R Code to Generate Study Data for Meta-Analysis

```
generate <- function(d = 4, sd = 0.1, ns = "large", u = 2) {

  if (u != 0 && u != 1 && u != 2 && u != 3) stop(
  "u must be 0 (no underreporting), 1 (constant underreporting),
  2 (tight split underreporting) or 3 (spread split underreporting)")

  ns <- match.arg(ns, c("small", "large"))
  if (ns == "small") ns <- 5 else ns <- 10

  D <- round(rnorm(d, ns, 2))
  for(i in 1:d) if(D[i] < ceiling(ns / 3)) D[i] <- ceiling(ns / 3)
  avg_y <- 250
  placebo <- -2
  d <- seq(0, -1, length = (d + 1))[-1]

  D <- D * 2
  N <- sum(D)
  NS <- N / 2
  NT <- length(D) + 1
  x <- matrix(0, N, length(D) + 1)
  x[,1] <- 1
  k <- 1
  t <- c()
```

```
for(i in 1:length(D)) {
  for(j in k:cumsum(D)[i]) if(j / 2 == round(j / 2))
    x[j, i + 1] <- 1
  k <- k + D[i]
  t <- c(t, rep(c(1, (i + 1)), D[i] / 2))
  }
s <- c(rbind(1:NS, 1:NS))
b <- rep(1, N)

if(u == 0) p <- rep(1, NT)
if(u == 1) p <- rep(0.7, NT)
if(u == 2) p <- seq(0.8, 0.6, length = length(d) + 1)
if(u == 3) p <- seq(0.9, 0.5, length = length(d) + 1)
alpha <- 60 * p
beta <- 60 - alpha

d <- c(0, d)
RR <- matrix(0, NT, NT)
for(i in 1:(NT - 1)) for(j in (i + 1):NT)
  RR[i, j] <- exp(d[j] - d[i])
RR <- RR[-1,]
rr <- c()
for(i in 1:length(c(RR))) if(c(t(RR))[i] != 0)
  rr <- c(rr, c(t(RR))[i])
d <- c(placebo, d[-1])
sd <- rep(sd, NT)
delta <- c()
```

```
for(i in 1:NT) delta <- cbind(delta, rnorm(NS, d[i], sd[i]))
y <- round(rnorm(N, avg_y, round(avg_y / 5)))
for(i in 1:N) if(y[i]<1) y[i] <- 1
r <- numeric(N)
for(i in 1:N)  r[i] <- rpois(1, p[t[i]] * y[i] * exp((x[i,] %*%
  delta[s[i],])))
list(r = r, y = y, t = t, s = s, b = b, rr = rr, alpha = alpha,
  beta = beta, N = N, NS = NS, NT = NT)
}
```

APPENDIX E

WinBUGS Models for Meta-Analysis

WinBUGS Code for Model with No Underreporting

```
model{
  for(i in 1:N) {
    log(lambda[i]) <- log(y[i]) + mu[s[i]] + delta[i] *
      (1 - equals(t[i], b[i]))
    r[i] ~ dpois(lambda[i])
    delta[i] ~ dnorm(theta[i], tau)
    theta[i] <- phi[t[i]] - phi[b[i]]
    }
  for(j in 1:NS) {
    mu[j] ~ dnorm(0, 0.005)
    }
  phi[1] <- 0
  for (k in 2:NT) {
    phi[k] ~ dnorm(0, 0.005)
    }
  sd ~ dunif(0, 3)
  tau <- 1 / pow(sd, 2)
  for (c in 1:(NT - 1)) for (k in (c + 1):NT)
    rr[c, k] <- exp(phi[k] - phi[c])
  }
```

WinBUGS Code for Model with Constant Underreporting

```
model{
  for(i in 1:N) {
    log(lambda[i]) <- log(p) + log(y[i]) + mu[s[i]] + delta[i] *
      (1 - equals(t[i], b[i]))
    r[i] ~ dpois(lambda[i])
    delta[i] ~ dnorm(theta[i], tau)
    theta[i] <- phi[t[i]] - phi[b[i]]
    }
  for(j in 1:NS) {
    mu[j] ~ dnorm(0, 0.005)
    }
  p ~ dbeta(alpha, beta)
  phi[1] <- 0
  for (k in 2:NT) {
    phi[k] ~ dnorm(0, 0.005)
    }
  sd ~ dunif(0, 3)
  tau <- 1 / pow(sd, 2)
  for (c in 1:(NT - 1)) for (k in (c + 1):NT)
    rr[c, k] <- exp(phi[k] - phi[c])
  }
```

WinBUGS Code for Model with Split Underreporting

```
model{
  for(i in 1:N) {
    log(lambda[i]) <- log(p[t[i]]) + log(y[i]) + mu[s[i]] + delta[i] *
      (1 - equals(t[i], b[i]))
    r[i] ~ dpois(lambda[i])
    delta[i] ~ dnorm(theta[i], tau)
    theta[i] <- phi[t[i]] - phi[b[i]]
    }
  for(j in 1:NS) {
    mu[j] ~ dnorm(0, 0.005)
    }
  for(k in 1:NT) p[k] ~ dbeta(alpha[k], beta[k])
  phi[1] <- 0
  for (k in 2:NT) {
    phi[k] ~ dnorm(0, 0.005)
    }
  sd ~ dunif(0, 3)
  tau <- 1 / pow(sd, 2)
  for (c in 1:(NT - 1)) for (k in (c + 1):NT)
    rr[c, k] <- exp(phi[k] - phi[c]
  }
```

APPENDIX F

R Code to Run Simulations for Meta-Analysis

```
source("Generate.R")
library(R2WinBUGS)

simulation <- function(reps, drugs = 4, sd = 0.1, ns = "large",
  u = 2, a = 2) {

  if (u != 0 && u != 1 && u != 2 && u != 3) stop(
  "Undereporting must be 0 (none), 1 (constant),
  2 (tight split) or 3 (spread split)")

  if (a != 0 && a != 1 && a != 2) stop(
  "Analyzation must be 0 (no underreporting),
  1 (constant underreporting), 2 (correct split underreporting),
  or 3 (incorrect split underreporting)")

  medians <- means <- lowers <- uppers <- c()
  success <- numeric(drugs * (drugs + 1) / 2 + 1 - drugs)

  for(i in 1:reps) {
    generated = generate(d = drugs, sd = sd, ns = ns, u = u)
    r <- generated$r
    t <- generated$t
    s <- generated$s
```

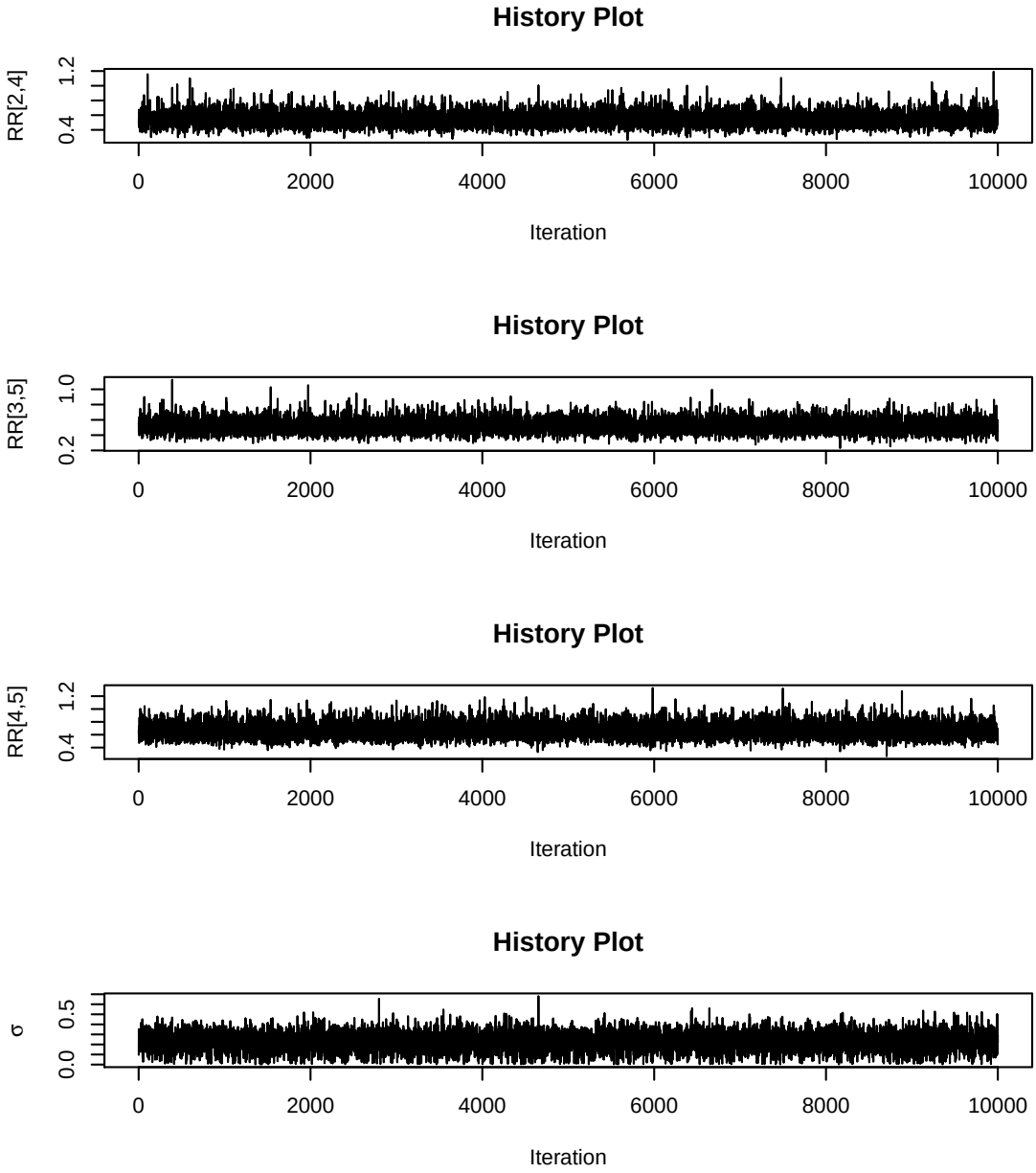
```
b <- generated$b
N <- generated$N
NS <- generated$NS
NT <- generated$NT
y <- generated$y
rr <- generated$rr
Parameters <- c("rr", "sd")
Inits <- list(list(sd = 1))
if(a == 0) {
  Data <- list("r", "t", "s", "b", "N", "NS", "NT", "y")
  Fit <- bugs(Data, Inits, Parameters, "NoUnder.txt", n.chains = 1,
    n.iter = 60000, n.thin = 5, n.burnin = 10000, debug = FALSE)
  }
if(a == 1) {
  alpha <- mean(generated$alpha)
  beta <- mean(generated$beta)
  Data <- list("r", "t", "s", "b", "N", "NS", "NT", "y", "alpha",
    "beta")
  Fit <- bugs(Data, Inits, Parameters, "Under.txt", n.chains = 1,
    n.iter = 60000, n.thin = 5, n.burnin = 10000, debug = FALSE)
  }
if(a == 2) {
  alpha <- generated$alpha
  beta <- generated$beta
  Data <- list("r", "t", "s", "b", "N", "NS", "NT", "y", "alpha",
    "beta")
  Fit <- bugs(Data, Inits, Parameters, "SplitUnder.txt",
```

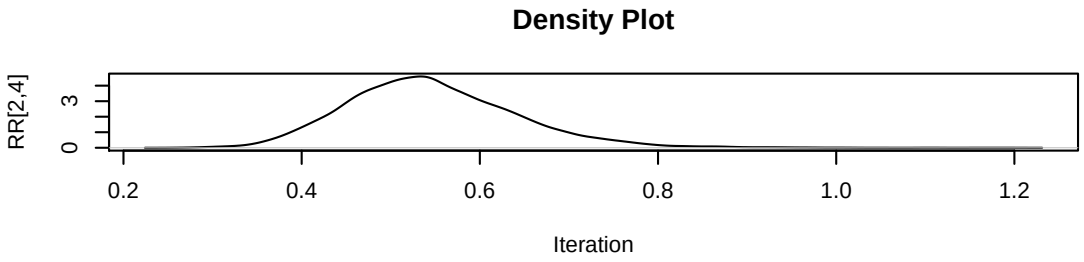
```
      n.chains = 1, n.iter = 60000, n.thin = 5, n.burnin = 10000,
      debug = FALSE)
    }
  if(a == 3) {
    alpha <- generated$alpha
    beta <- generated$beta
    alpha <- alpha + rep(c(3, -3), 10000)[1:length(alpha)]
    beta <- beta + rep(c(-3, 3), 10000)[1:length(beta)]
    Data <- list("r", "t", "s", "b", "N", "NS", "NT", "y", "alpha",
      "beta")
    Fit <- bugs(Data, Inits, Parameters, "SplitUnder.txt",
      n.chains = 1, n.iter = 60000, n.thin = 5, n.burnin = 10000,
      debug = FALSE)
    }
  X <- Fit$summary[1:(drugs * (drugs + 1) / 2 + 1),][-1:-drugs,]
  medians <- rbind(medians, X[, 5])
  means <- rbind(means, X[, 1])
  lowers <- rbind(lowers, X[, 3])
  uppers <- rbind(uppers, X[, 7])
  for(j in 1:length(rr)) if(rr[j] > lowers[i,j] && rr[j] <
    uppers[i,j]) success[j] <- success[j] + 1
  if(sd > lowers[i,length(success)] && sd <
    uppers[i,length(success)]) success[length(success)] <-
    success[length(success)] + 1
  }

print(noquote("Actual Relative Risks and Standard Deviation"))
```

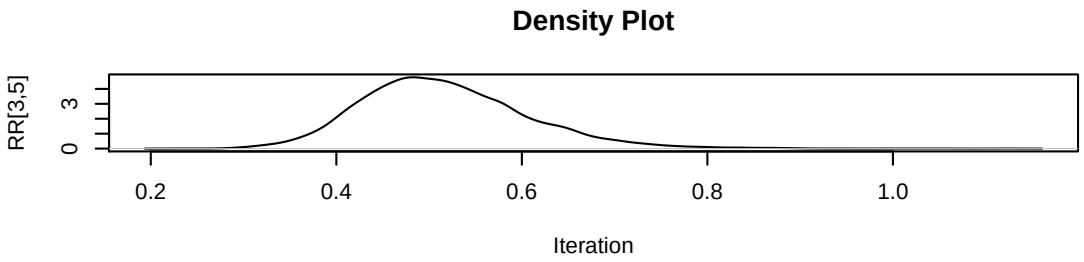
```
print(c(round(rr, 3), sd))
print(noquote("Medians"))
print(round(colMeans(medians), 3))
print(noquote("Means"))
print(round(colMeans(means), 3))
print(noquote("Interval Widths"))
print(round(colMeans(uppers - lowers), 3))
print(noquote("Coverage"))
print(round(success / reps * 100, 1))
}
```

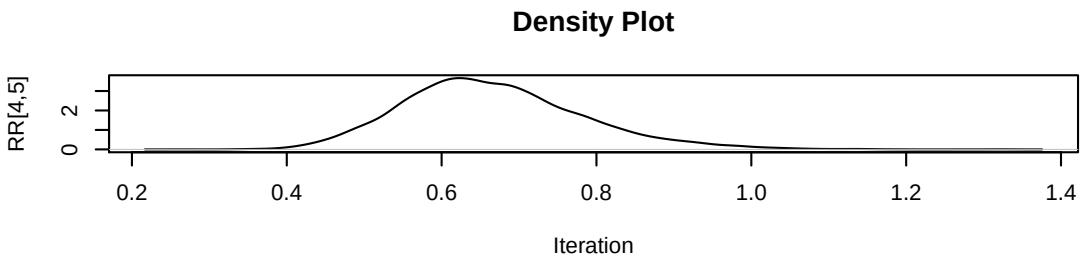
APPENDIX G

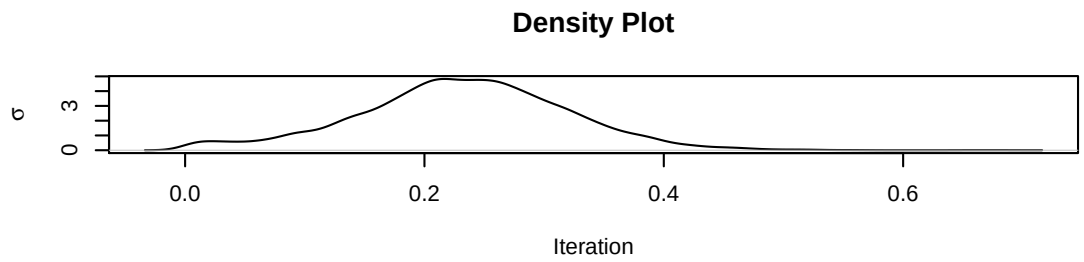
MCMC Assumption Plots for Traditional Model

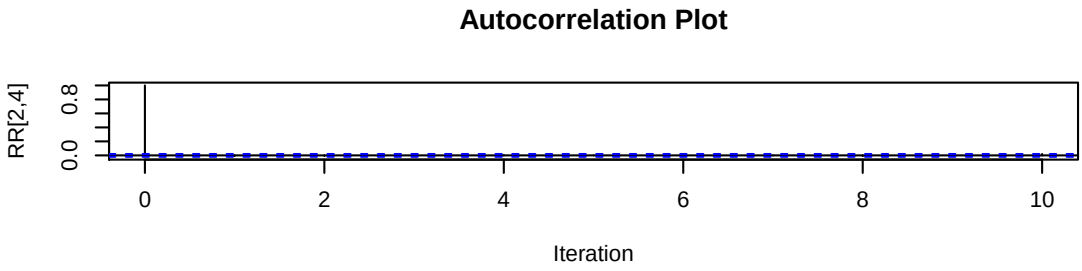


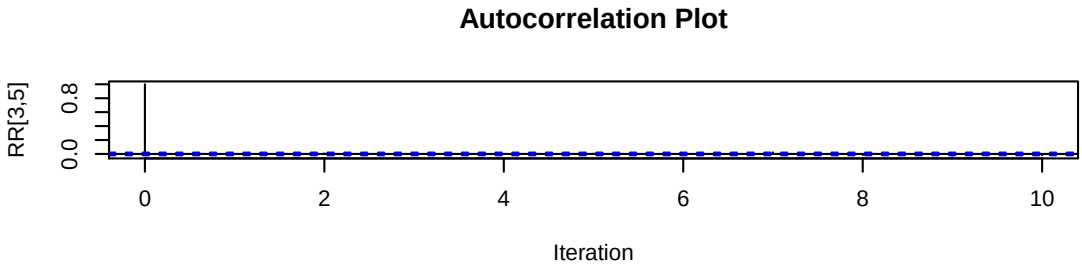

Density Plot
RR[2,4]
0
3
0.2
0.4
0.6
0.8
1.0
1.2
Iteration

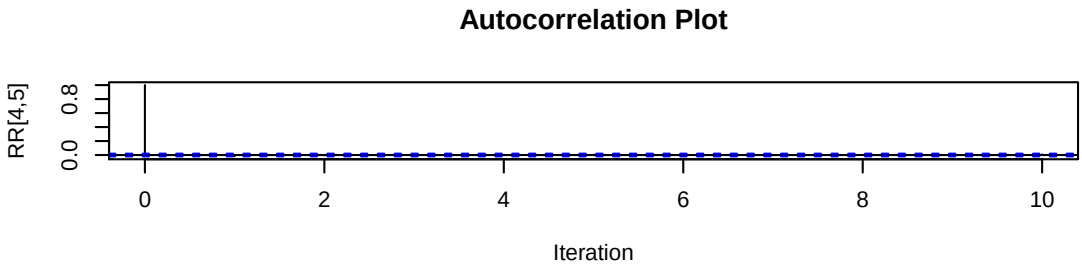

Density Plot
RR[3,5]
0
3
0.2
0.4
0.6
0.8
1.0
Iteration

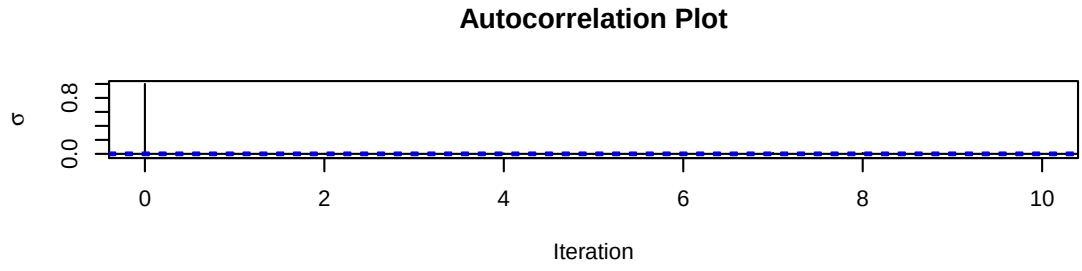

Density Plot
RR[4,5]
0
2
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Iteration


Density Plot
σ
0
3
0.0
0.2
0.4
0.6
Iteration


Autocorrelation Plot
RR[2,4]
0.0
0.8
0
2
4
6
8
10
Iteration

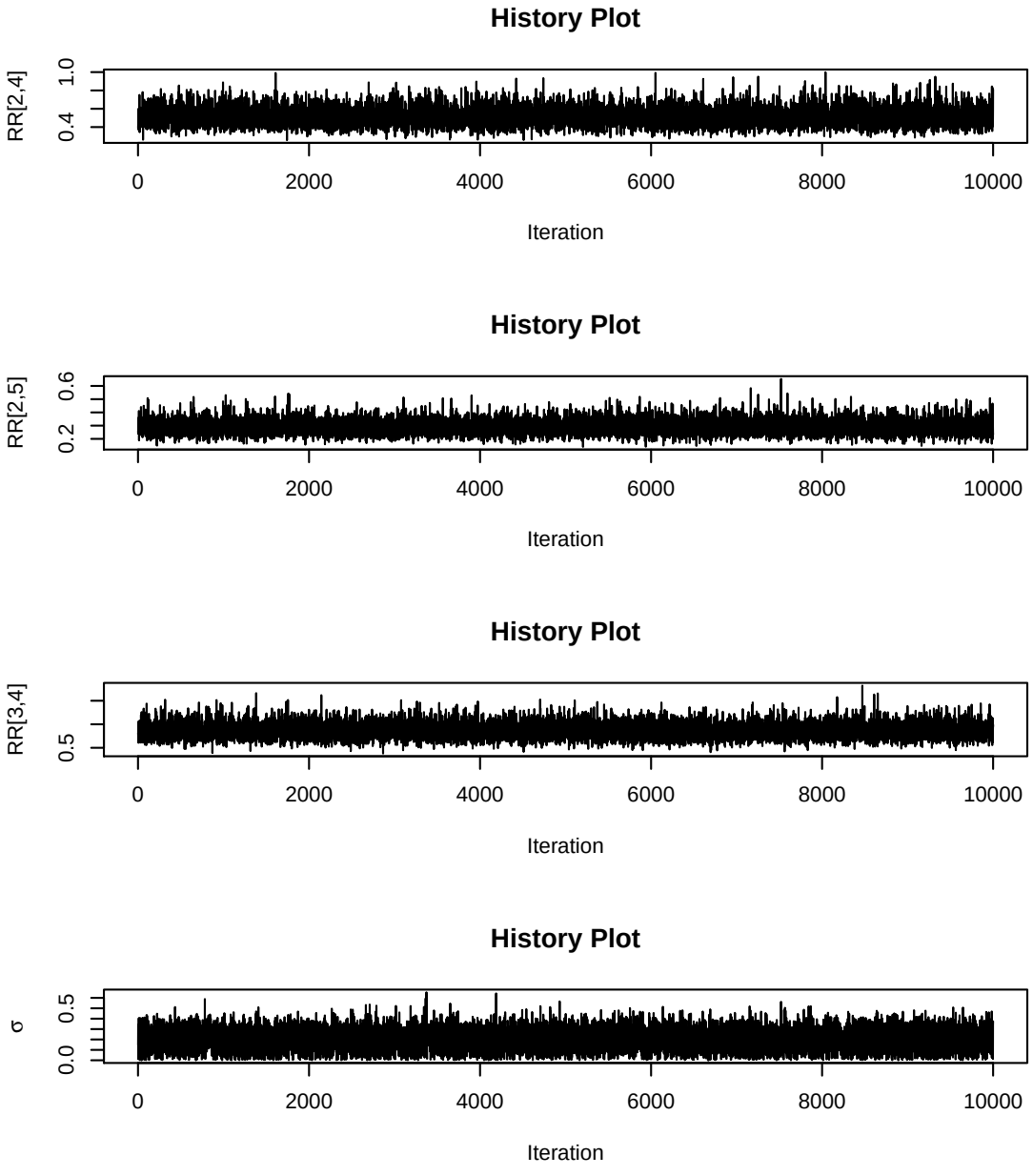

Autocorrelation Plot
RR[3,5]
0.0
0.8
0
2
4
6
8
10
Iteration

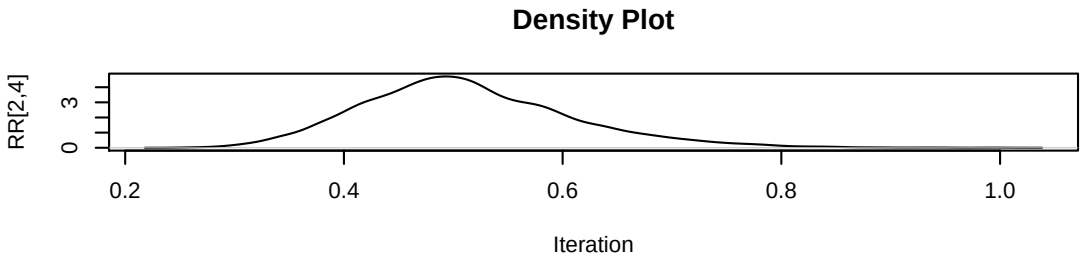

Autocorrelation Plot
RR[4,5]
0.0
0.8
0
2
4
6
8
10
Iteration

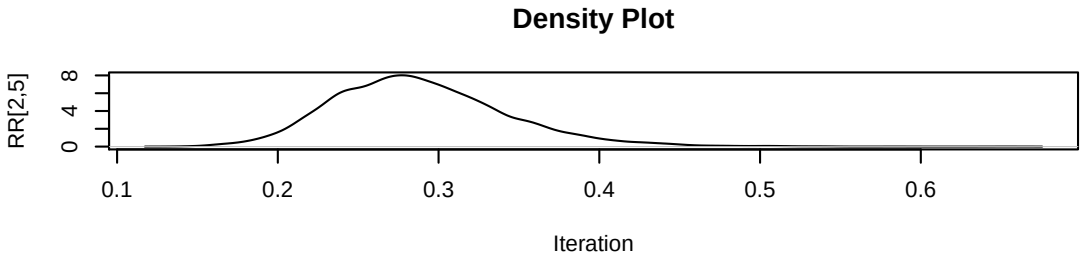

Autocorrelation Plot
σ
0.0
0.8
0
2
4
6
8
10
Iteration

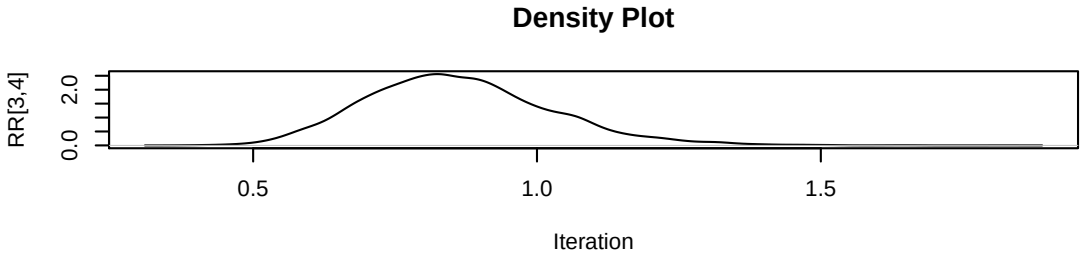
APPENDIX H

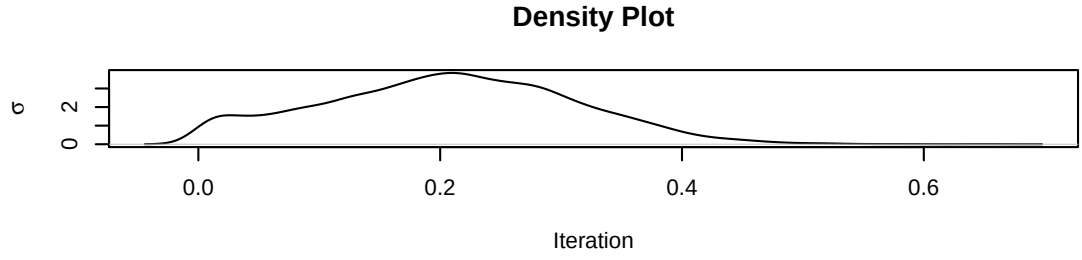
MCMC Assumption Plots for Split Underreported Data Analyzed with No Underreporting

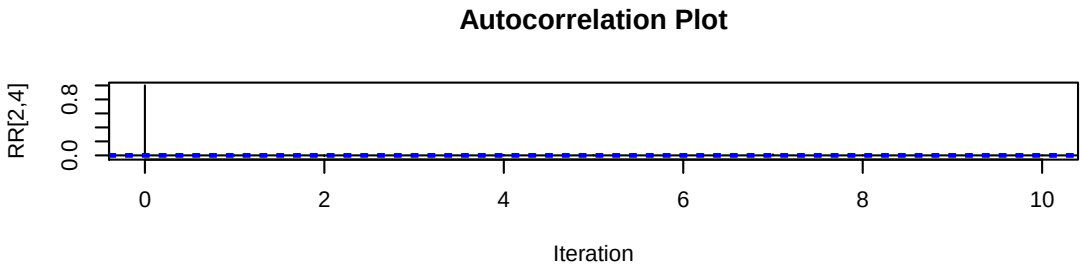


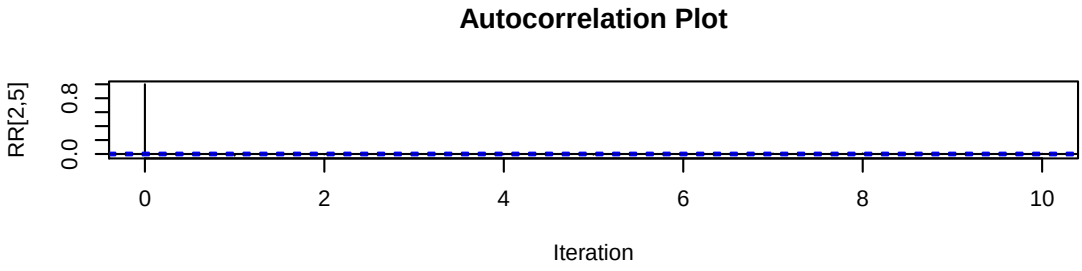

Density Plot
RR[2,4]
0
3
0.2
0.4
0.6
0.8
1.0
Iteration

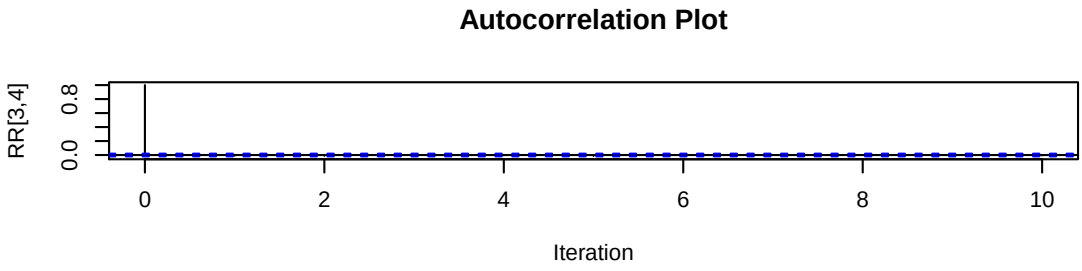

Density Plot
RR[2,5]
0
4
8
0.1
0.2
0.3
0.4
0.5
0.6
Iteration

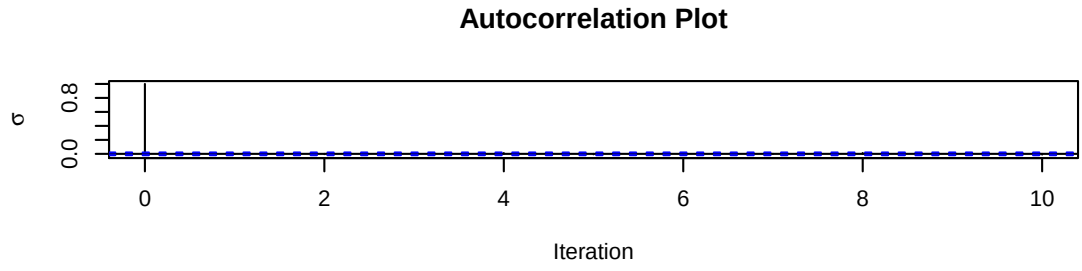

Density Plot
RR[3,4]
0.0
2.0
0.5
1.0
1.5
Iteration


Density Plot
σ
0
2
0.0
0.2
0.4
0.6
Iteration


Autocorrelation Plot
RR[2,4]
0.0
0.8
0
2
4
6
8
10
Iteration

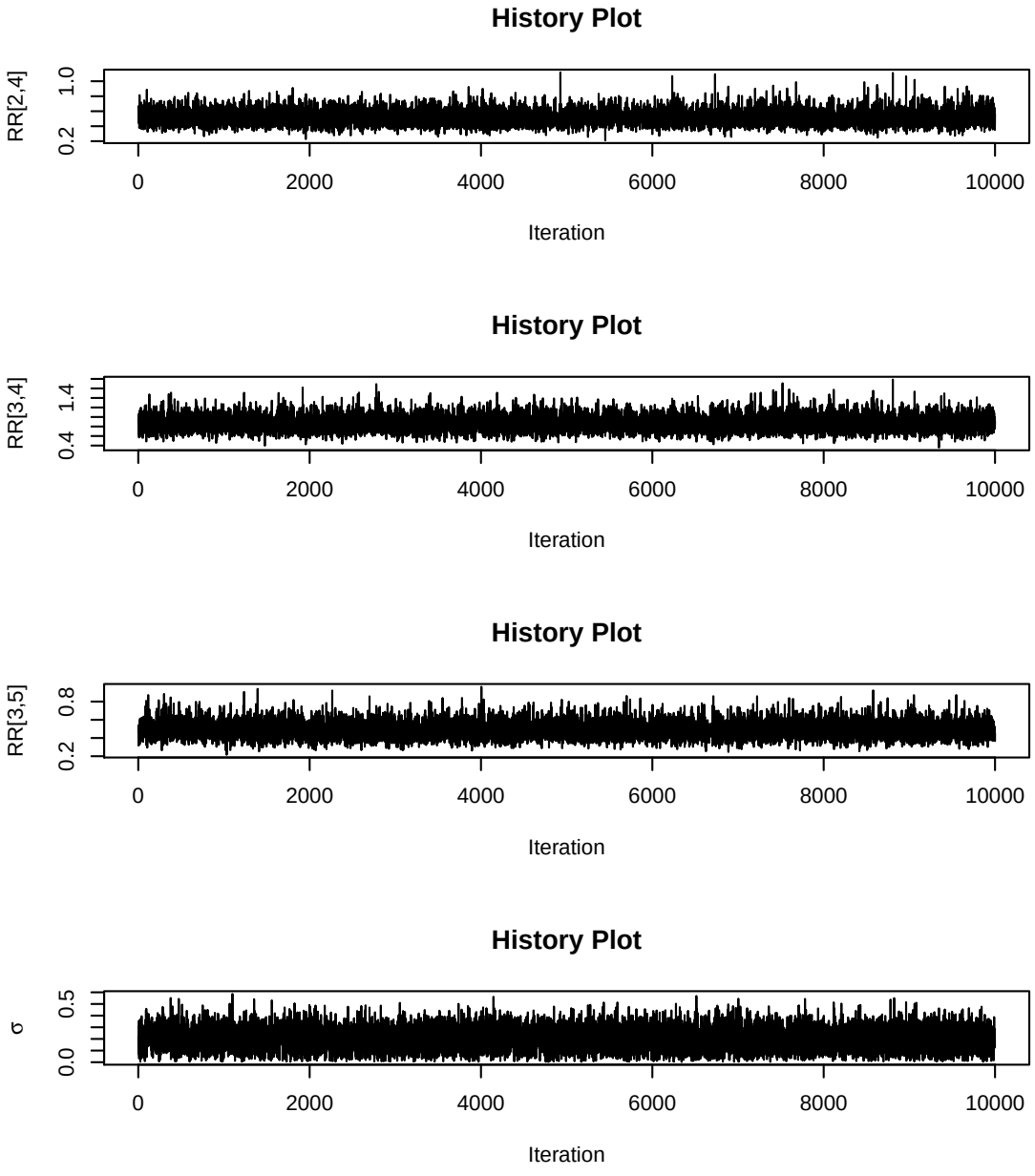

Autocorrelation Plot
RR[2,5]
0.0
0.8
0
2
4
6
8
10
Iteration

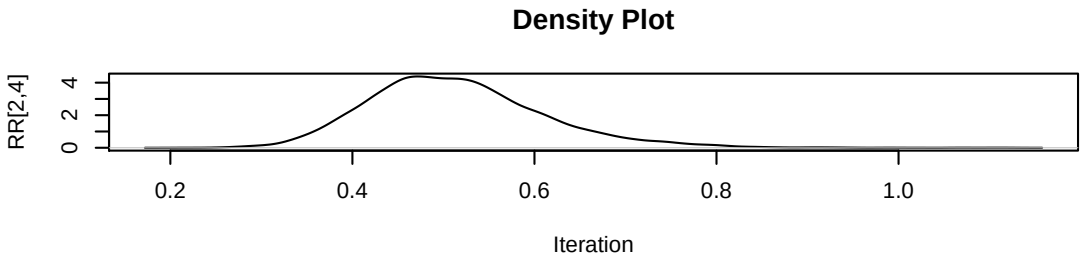

Autocorrelation Plot
RR[3,4]
0.0
0.8
0
2
4
6
8
10
Iteration

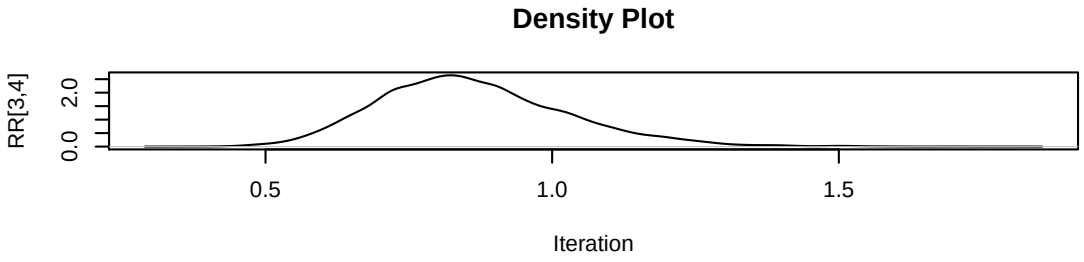

Autocorrelation Plot
σ
0.0
0.8
0
2
4
6
8
10
Iteration

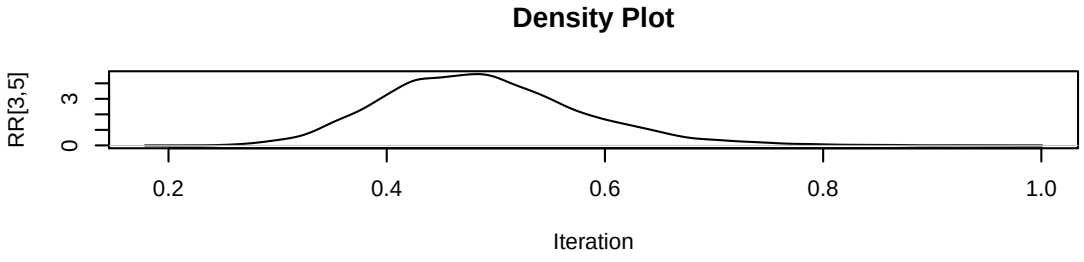
APPENDIX I

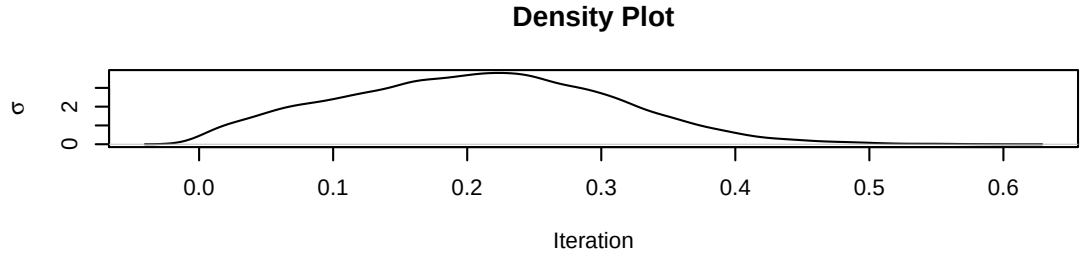
MCMC Assumption Plots for Split Underreported Data Analyzed with Constant Underreporting

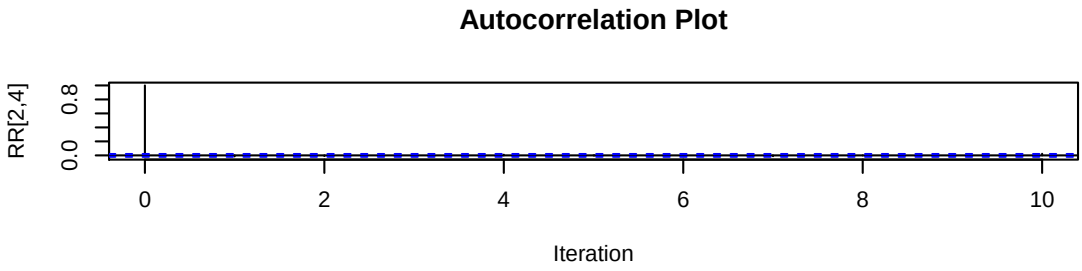


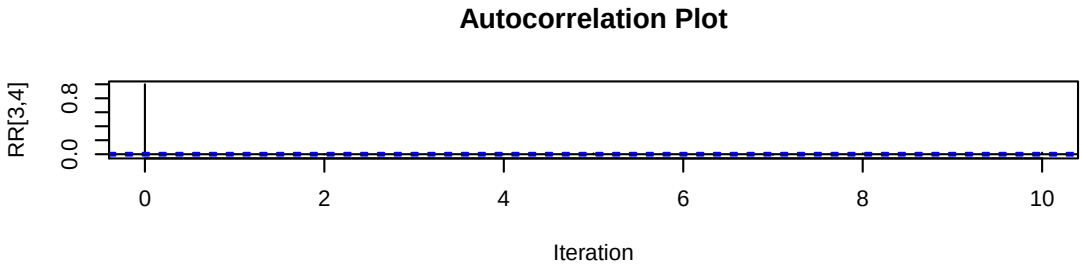

Density Plot
RR[2,4]
0
2
4
0.2
0.4
0.6
0.8
1.0
Iteration

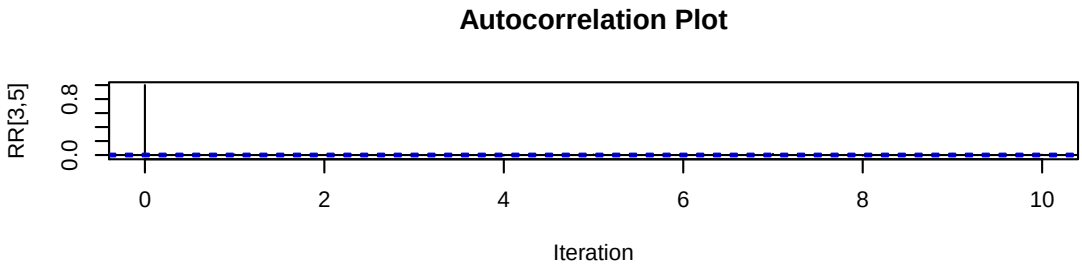

Density Plot
RR[3,4]
0.0
2.0
0.5
1.0
1.5
Iteration

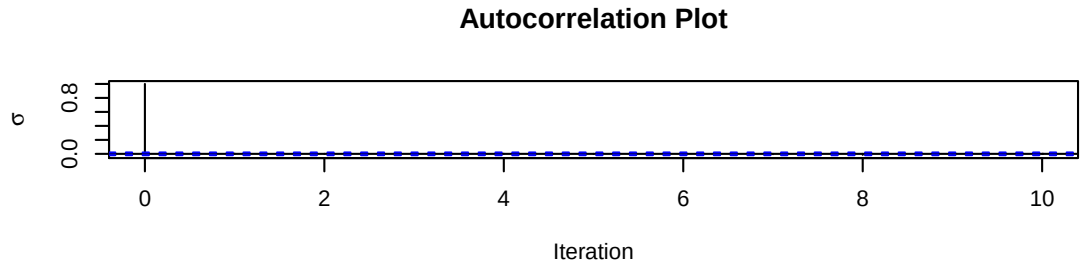

Density Plot
RR[3,5]
0
3
0.2
0.4
0.6
0.8
1.0
Iteration


Density Plot
σ
0
2
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Iteration


Autocorrelation Plot
RR[2,4]
0.0
0.8
0
2
4
6
8
10
Iteration

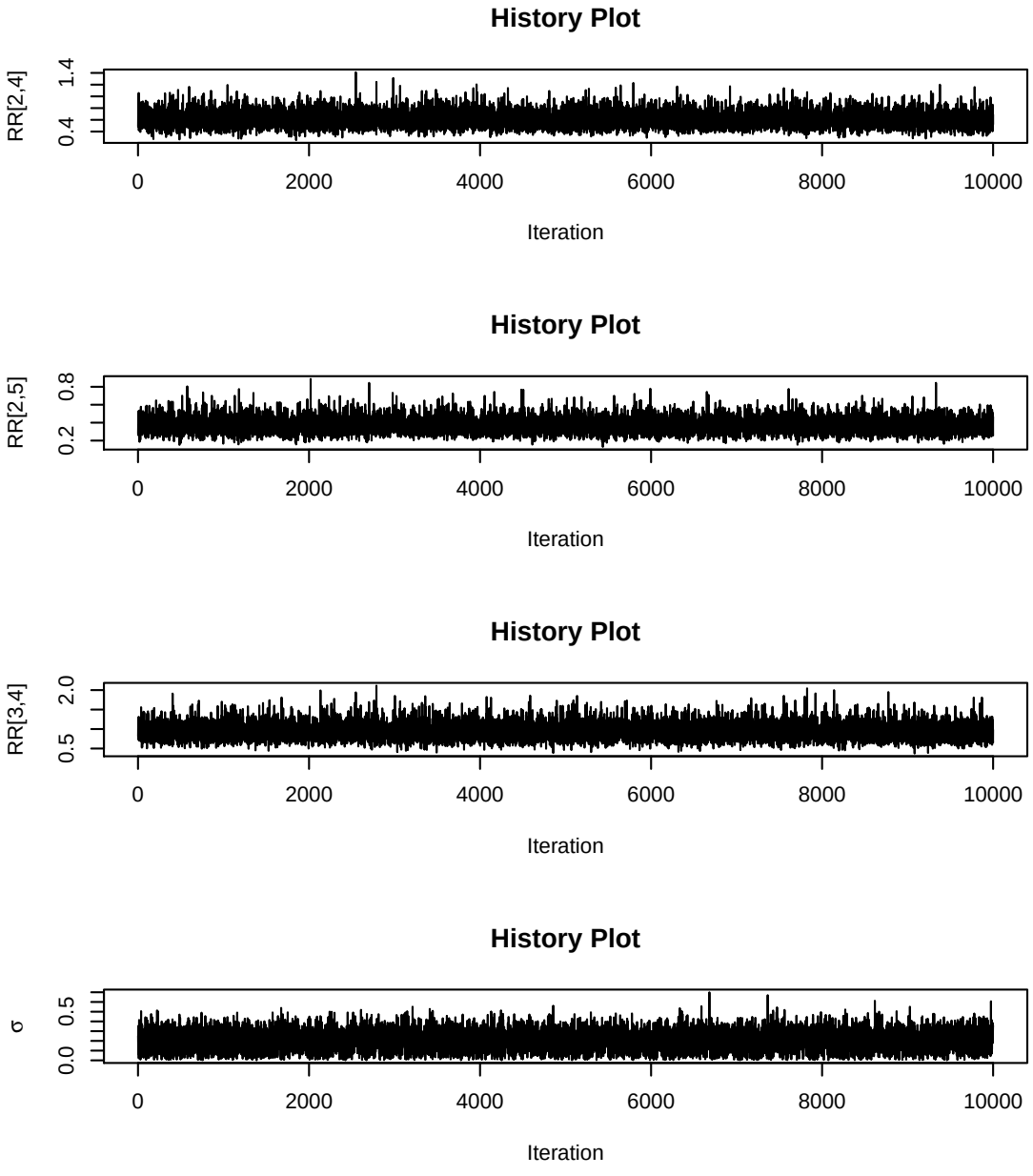

Autocorrelation Plot
RR[3,4]
0.0
0.8
0
2
4
6
8
10
Iteration

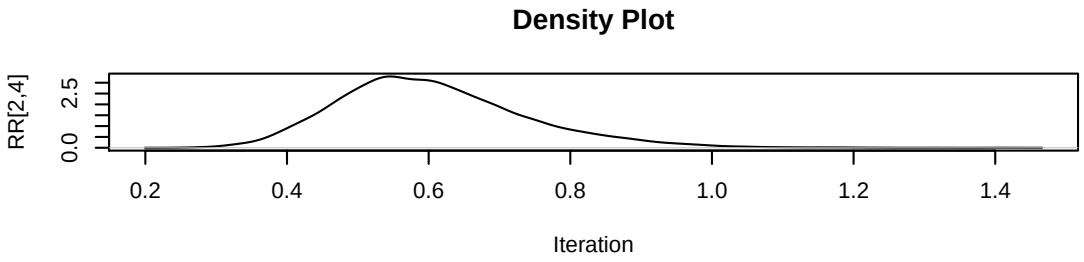

Autocorrelation Plot
RR[3,5]
0.0
0.8
0
2
4
6
8
10
Iteration

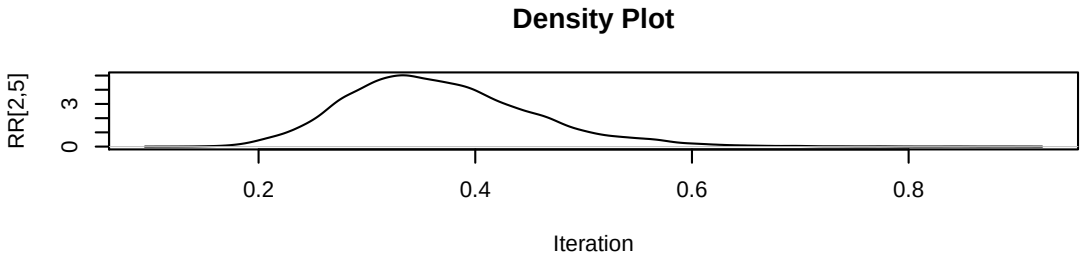

Autocorrelation Plot
σ
0.0
0.8
0
2
4
6
8
10
Iteration

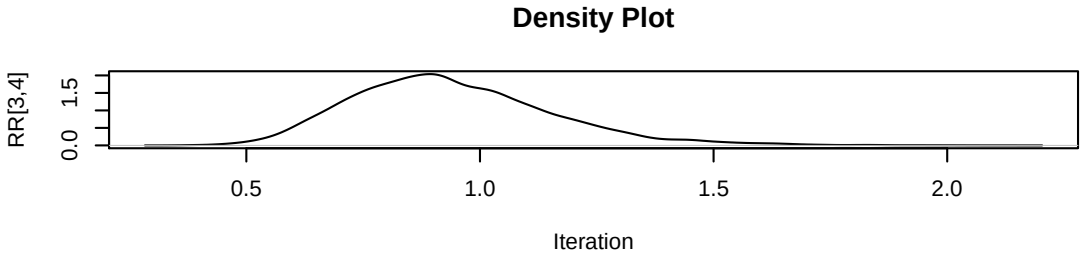
APPENDIX J

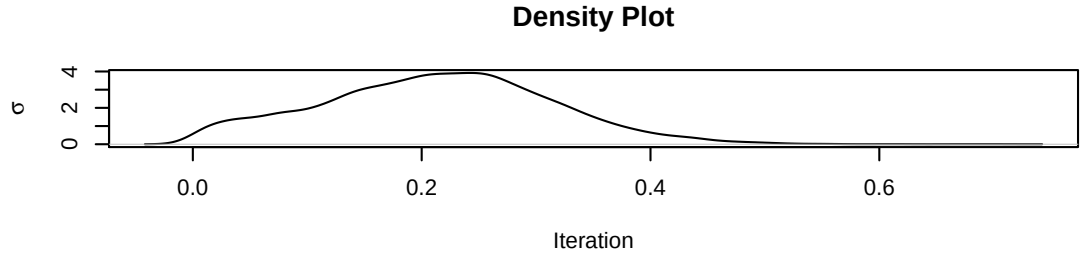
MCMC Assumption Plots for Split Underreported Data Analyzed with Correct Split Underreporting

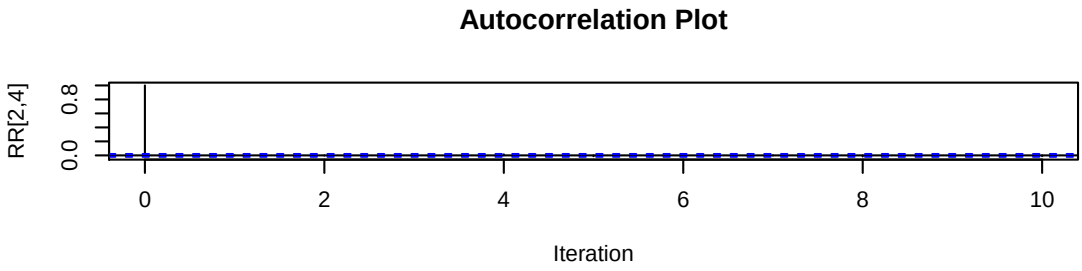


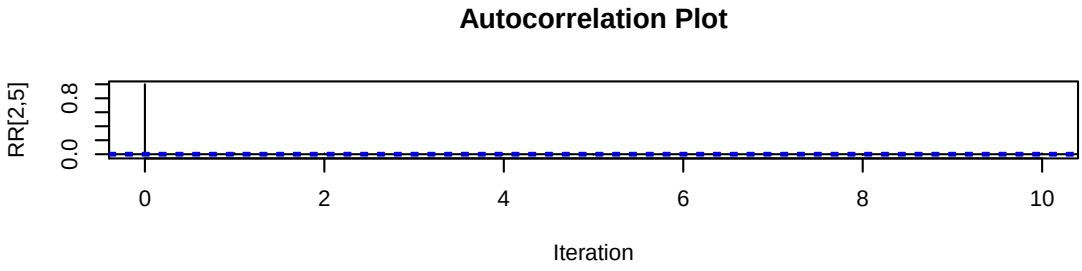

Density Plot
RR[2,4]
0.0
2.5
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Iteration

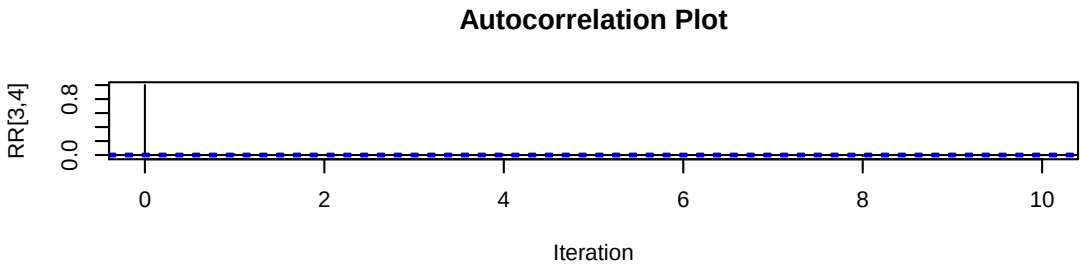

Density Plot
RR[2,5]
0
3
0.2
0.4
0.6
0.8
Iteration

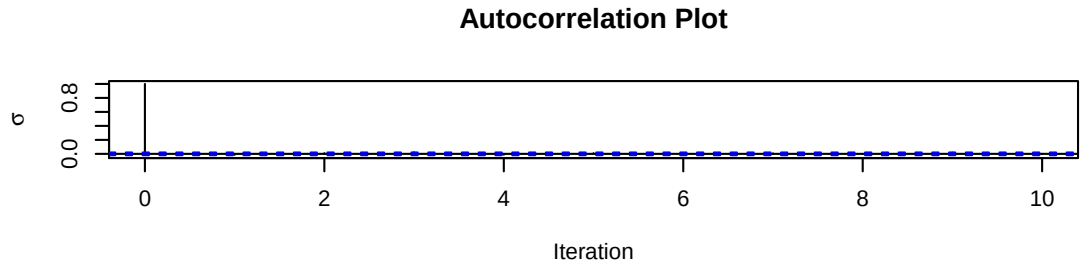

Density Plot
RR[3,4]
0.0
1.5
0.5
1.0
1.5
2.0
Iteration


Density Plot
σ
0
2
4
0.0
0.2
0.4
0.6
Iteration


Autocorrelation Plot
RR[2,4]
0.0
0.8
0
2
4
6
8
10
Iteration

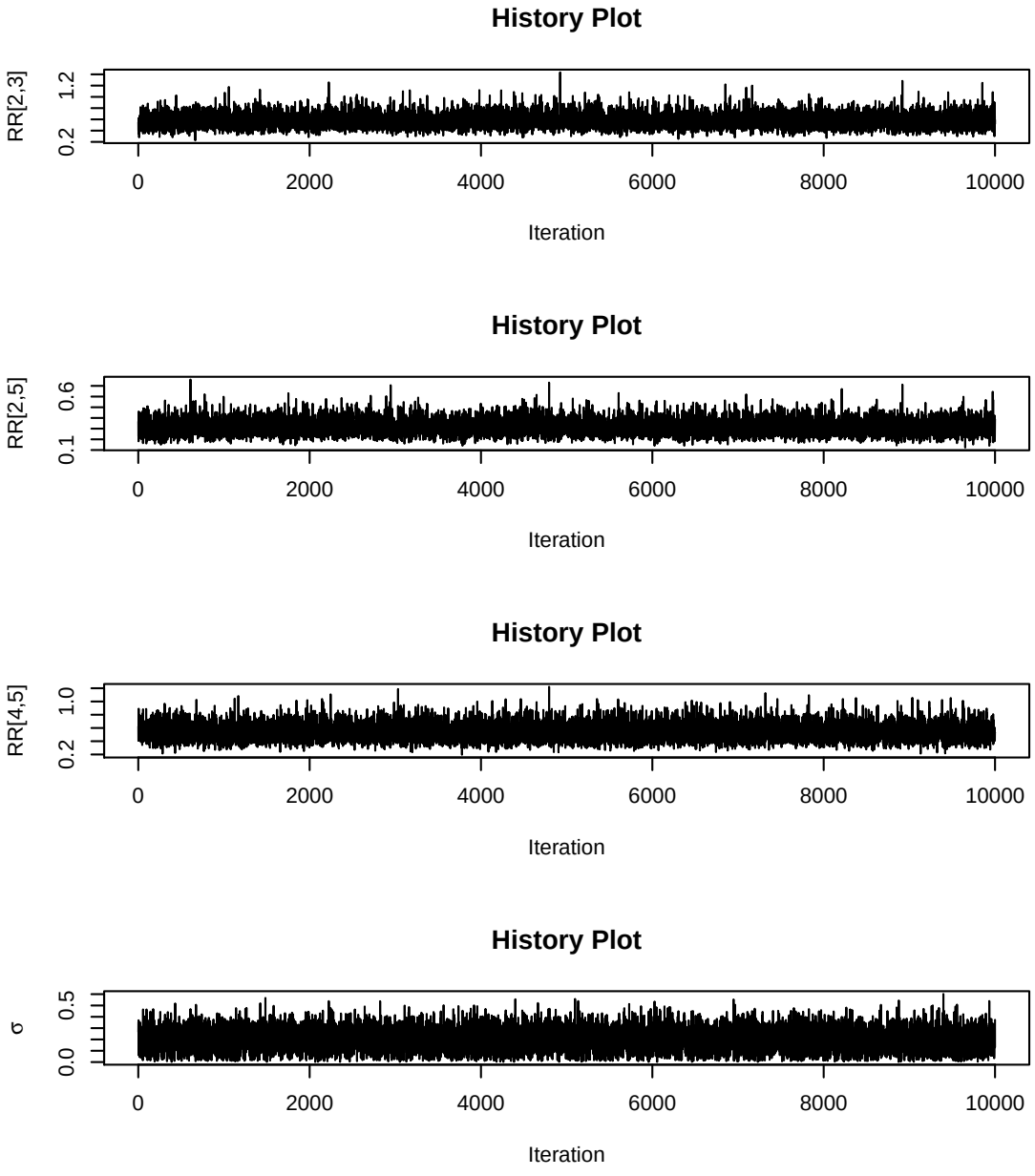

Autocorrelation Plot
RR[2,5]
0.0
0.8
0
2
4
6
8
10
Iteration

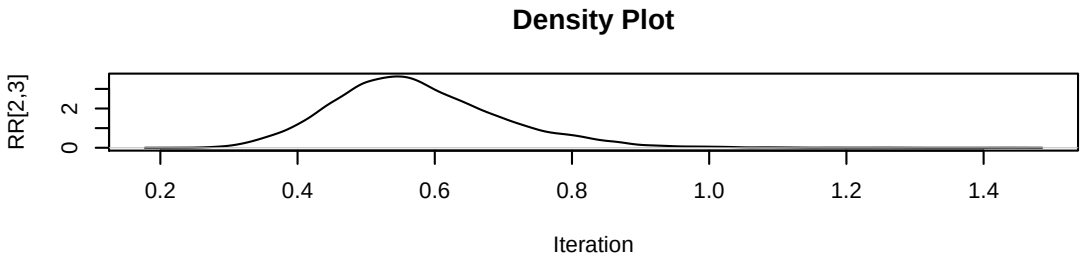

Autocorrelation Plot
RR[3,4]
0.0
0.8
0
2
4
6
8
10
Iteration

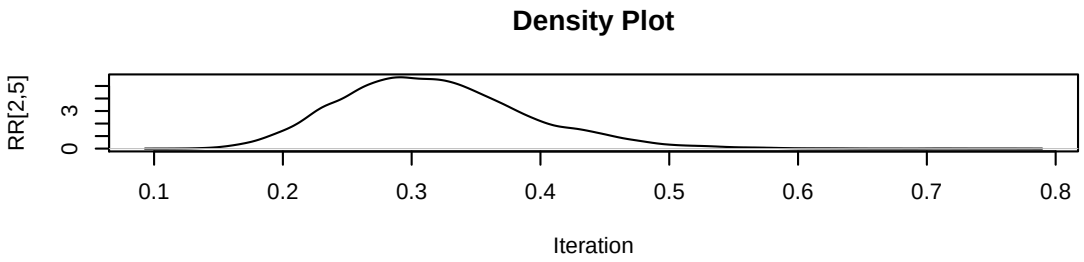

Autocorrelation Plot
σ
0.0
0.8
0
2
4
6
8
10
Iteration

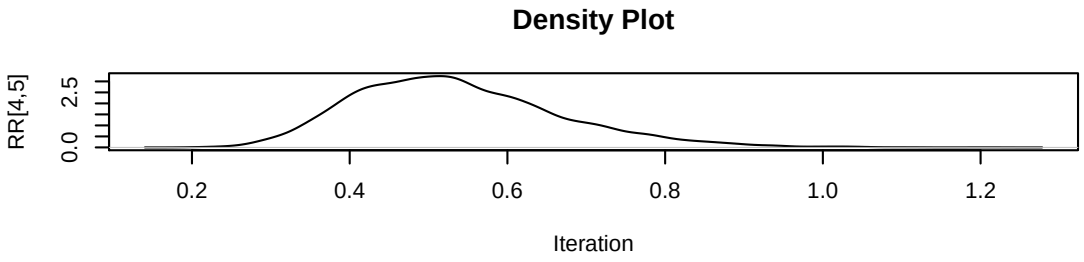
APPENDIX K

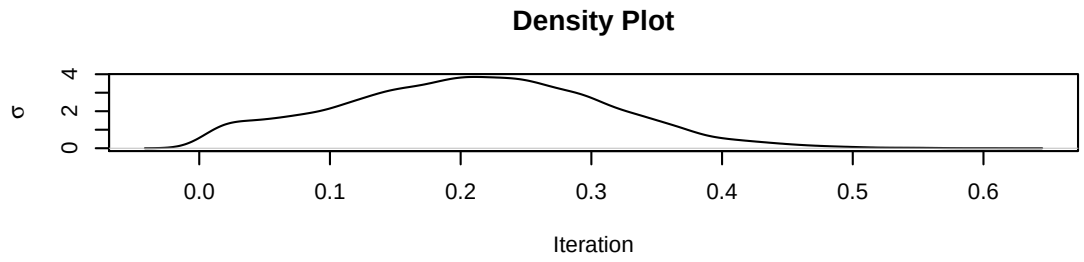
MCMC Assumption Plots for Split Underreported Data Analyzed with Incorrect Split Underreporting

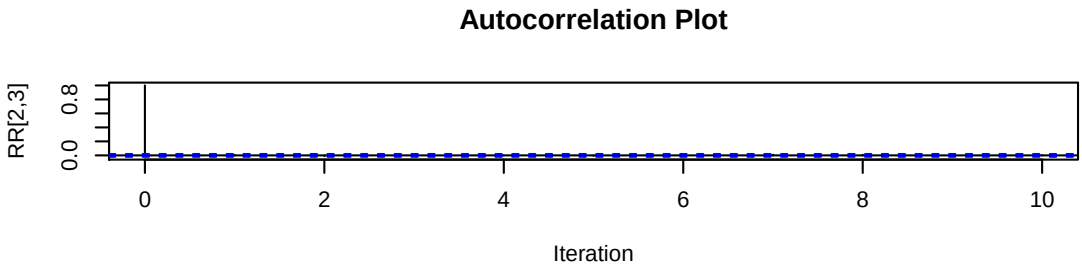


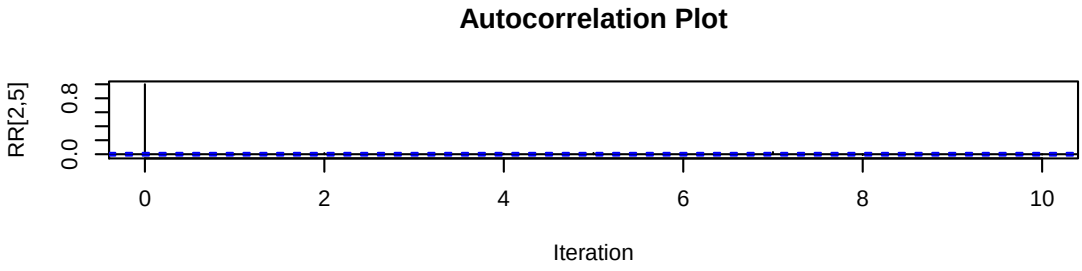

Density Plot
RR[2,3]
0
2
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Iteration

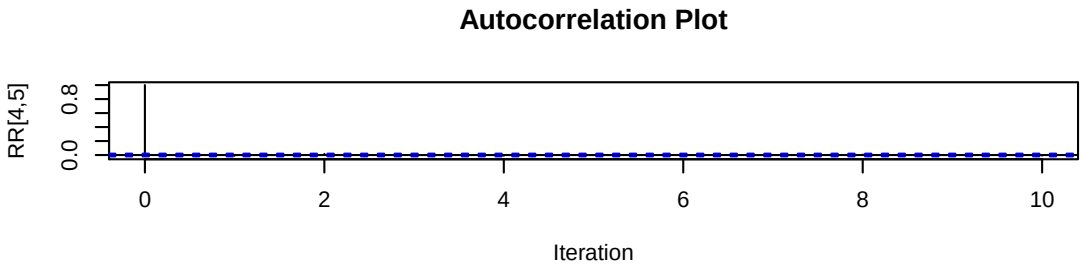

Density Plot
RR[2,5]
0
3
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Iteration

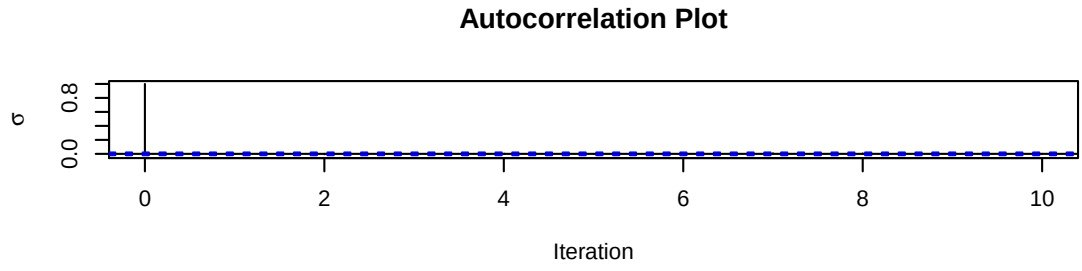

Density Plot
RR[4,5]
0.0
2.5
0.2
0.4
0.6
0.8
1.0
1.2
Iteration


Density Plot
σ
0
2
4
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Iteration


Autocorrelation Plot
RR[2,3]
0.0
0.8
0
2
4
6
8
10
Iteration


Autocorrelation Plot
RR[2,5]
0.0
0.8
0
2
4
6
8
10
Iteration


Autocorrelation Plot
RR[4,5]
0.0
0.8
0
2
4
6
8
10
Iteration


Autocorrelation Plot
σ
0.0
0.8
0
2
4
6
8
10
Iteration

BIBLIOGRAPHY

Adcock, C. J. (1997), "Sample Size Determination: A Review," *The Statistician*, 46, 261–283.

Bayarri, M. J. and Berger, J. O. (2004), "The Interplay of Bayesian and Frequentist Analysis," *Statistical Science*, 19, 58–80.

Cooper, N., Sutton, A., Guobing, L., and Khunti, K. (2006), "Mixed Comparison of Stroke Prevention Treatments in Individuals with Nonrheumatic Atrial Fibrillation," *Archives of Internal Medicine*, 166, 1269–1275.

De Santis, F. and Perone Pacifico, M. (2003), "Two Experimental Settings in Clinical Trials: Predictive Criteria for Choosing the Sample Size in Interval Estimation," in *Applied Bayesian Statistical Studies in Biology and Medicine*, Kluwer Academic Publishers.

De Santis, F., Perone Pacifico, M., and Sambucini, V. (2004), "Optimal Predictive Sample Size for Case-Control Studies," *Journal of Applied Statistics*, 53, 427–441.

Dias, S., Sutton, A., Ades, A. E., and Welton, N. (2012), "Evidence Synthesis for Decision Making 2: A Generalized Linear Modeling Framework for Pairwise and Network Meta-Analysis of Randomized Controlled Trials," *Medical Decision Making*, 33, 607–617.

Hand, A., Stamey, J., and Young, D. (2011), "Bayesian Sample-Size Determination for Two Independent Poisson Rates," *Computer Methods and Programs in Biomedicine*, 104, 271–277.

Jansen, J., Fleurence, R., Devine, B., Itzler, R., Barrett, A., Hawkins, N., Lee, K., Boersma, C., Annemans, L., and Cappelleri, J. (2011), "Interpreting Indirect Treatment Comparisons and Network Meta-Analysis for Health-Care Decision Making: Report of the ISPOR Task Force on Indirect Treatment Comparisons Good Research Practices: Part 1," *Value in Health*, 14, 417–428.

Joseph, L., Du Berger, R., and Bélisle, P. (1997), "Bayesian and Mixed Bayesian/-Likelihood Criteria for Sample Size Determination," *Statistics in Medicine*, 16, 769–781.

Joseph, L., Wolfson, D., and Du Berger, R. (1995), "Sample Size Calculations for Binomial Proportions via Highest Posterior Density Intervals," *The Statistician*, 44, 143–154.

Kass, R. and Raftery, A. (1995), "Bayes Factors," *Journal of the American Statistical Association*, 90, 773–795.

Katsis, A. and Toman, B. (1999), "Bayesian Sample Size Calculations for Binomial Experiments," *Journal of Statistical Planning and Inference*, 81, 349–362.

M'Lan, C., Joseph, L., and Wolfson, D. B. (2006), "Bayesian Sample Size Determination for Case-Control Studies," *Journal of the American Statistics Association*, 101, 760–772.

— (2008), "Bayesian Sample Size Determination for Binomial Proportions," *Bayesian Analysis*, 2, 269–296.

O'Hagan, A., Stevens, J., and Campbell, M. (2005), "Assurance in Clinical Design," *Pharmaceutical Statistics*, 4, 187–201.

Powers, S., Gerlach, R., and Stamey, J. (2010), "Bayesian Variable Selection for Poisson Regression with Underreported Responses," *Computational Statistics and Data Analysis*, 54, 3289–3299.

Smith, T. C., Spiegelhalter, D. J., and Thomas, A. (1995), "Bayesian Approaches to Random-Effects Meta-Analysis: A Comparative Study," *Statistics in Medicine*, 14, 2685–2699.

Speighelhalter, D. J., Abrams, K. R., and Myles, J. P. (2004), *Bayesian Approaches to Clinical Trials and Health Care Evaluation*, New York: Wiley.

Stamey, J., Seaman, J., and Young, D. (2007), "Bayesian Estimation of Intervention Effect with Pre- and Post-Misclassified Binomial Data," *Journal of Biopharmaceutical Statistics*, 17, 93–108.

Stamey, J., Young, D., and Bratcher, T. (2006), "Bayesian Sample-Size Determination for One and Two Poisson Rate Parameters with Applications to Quality Control," *Journal of Applied Statistics*, 33, 583–594.

Stamey, J., Young, D., and Seaman, J. (2008), "A Bayesian Approach to Adjust for Diagnostic Misclassification between Two Mortality Causes in Poisson Regression," *Statistics in Medicine*, 27, 2440–2452.

Whittemore, A. S. and Gong, G. (1991), "Poisson Regression with Misclassified Counts: Application to Cervical Cancer Mortality Rates," *Applied Statistics*, 40, 81–93.

Zhao, Z., Tang, N., and Li, Y. (2011), "Sample-Size Determination for Two Independent Binomial Experiments," *Journal of Systems Science and Complexity*, 24, 981–990.