ABSTRACT

Network Meta-analysis with Rare Events and Misclassified Response

Wenqi Wu, Ph.D.

Mentors: James D. Stamey, Ph.D., David Kahle, Ph.D.

Count data are subject to considerable sources of what is often referred to as non-sampling error. Errors such as misclassification, measurement error, and unmeasured confounding can lead to substantially biased estimators. It is strongly recommended that epidemiologists not only acknowledge these sorts of errors in data but also incorporate sensitivity analyses into part of the total data analysis. In this dissertation, we extend previous work on Poisson regression models that allow for misclassification by thoroughly discussing the basis for the models and allowing for extra-Poisson variability in the form of random effects. Markov chain Monte Carlo methods are applied to perform the computations needed to draw inferences and make model assessments. Through simulation, we show the improvements in inference that are brought about by accounting for both misclassification and overdispersion.

Network meta-analysis is increasingly popular in clinical trials and provides both direct and indirect treatment comparisons. One common issue in network meta-analysis is zero outcomes, which will lead to biased estimates and low coverage probabilities. We consider both the binomial distribution and the Poisson distribution to model data. Four network patterns are considered, which are star, loop, ladder, and one-closed loop geometry. The Bayesian approach is used as our method

of inference. Through simulation, we evaluate two contiunity correction methods for different geometry patterns. The performance of continuity correction depends on the geometry pattern and the underlying distribution assumption.

We also consider misclassification in the network meta-analysis for binary outcomes. Sensitivity and specificity are introduced to adjust misclassified data. Through simulation, we demonstrate the importance of accounting for misclassification. We also assess the robustness of different values for sensitivity and specificity. We find that the the posterior inferences are very sensitive to misclassification rates.

Network Meta-analysis with Rare Events and Misclassified Response

by

Wenqi Wu, Ph.D.

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., Mentor

_____

David Kahle, Ph.D., Mentor

_____

Carson F. Mencken, Ph.D.

_____

Dean M. Young, Ph.D.

_____

Dennis A. Johnston, Ph.D.

Accepted by the Graduate School
May 2016

_____

J. Larry Lyon, Ph.D., Dean

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

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I would like to thank my committee members for their insightful suggestions and comments for improving this thesis. I would never have been able to finish my dissertation without their guidance.

My deepest gratitude goes to my advisor, Dr. Stamey, for continuous support of my Ph.D study and related research. His excellent guidance and patience make it possible for me to work on the topics that are of great interest to me. I could not imagine having a better advisor and mentor for my Ph.D study. I would like to thank my co-advisor, Dr. Kahle. I am extremely grateful to him for consecutive assistance throughout my dissertation.

I would also like to thank the faculty at both Baylor and Southern Methodist University for helping me develop my background in statistics, guiding me in my research, and various forms of support during my graduate study in the U.S. My sincere thanks to Alissa G. Wiseman for reading my dissertation, commenting on countless revisions of my writings, and helping me finish the dissertation.

I thank my former and current classmates for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the tough and happy moments we had during the last five years.

Finally, I would like to thank my parents and wife. Their continuous encouragement and guidance helped me overcome all the difficulties during my Ph.D study. They made me who I am now.

DEDICATION

To my wife, Yuan Shi

CHAPTER ONE

Introduction

In this dissertation we discuss two topics. First, accounting for misclassification in Poisson regression model and network meta-analysis. Second, comparisons of continuity correction methods for the zero outcomes in the network meta-analysis.

1.1 Misclassification and Rare Event

In the epidemiologic studies, one major source of error is misclassification in counted data due to misreporting by subjects or the use of imperfect measurement devices in data collection. Misclassification in the counted data refers to the classification of an individual into a category other than that to which it should be assigned. Without appropriate adjustment, misclassification could lead to substantially biased estimates, and can therefore underestimate risk of certain disease. Tu, Litvak, and Pagano 1994 discussed that the imperfect measurement of HIV led to incorrectly estimation of prevalence in New York.

Rare event is one of the major concerns for binary data in clinical trial, for example the adverse events associated with medical treatments. Even though meta-analysis may be the only way to obtain reliable evidence of the effects of healthcare interventions (Bradburn et al. 2006), there are still difficulties in estimating an effect and standard error when no events occur in either arm. For example, if zero events exist in one or more cells in a 2×2 table, the commonly used Mantel-Haenszel and Peto methods are inappropriate, espcially, the calculation of odds ratio, which is not defined in this situation. Risk difference is an option when zero events exist in one or both arms. However, risk difference may highly depend on the underlying risk of events.

1.2 Plan of the Dissertation

The purpose of this dissertaion is to adjust misclassification and zero events in Poisson regression model and network meta-analysis. The performance of adjustment is assessed through simulation studies and Bayesian approach is used to estimate the parameters of interest.

In Chapter Two, we adjust misclassification by incorporating specificity and sensitivity in the Poisson regression. In addition, we consider adding random effects in the Poisson regression models that allow for extra-Poisson variability, which is commonly encountered in observational study. We use the more accessible graphical models to illustrate the assumptions and the form of the models. Through simulation studies, we demonstrate that our proposed model over performs the naive model that ignores the misclassificaiton.

Chapter Three consists of comparisons of two continuity correction methods for zero events in network meta-analysis for counted data. A small number (0.5 or 1) is added to both arms from a trial where either outcome is zero. We assume that data follows binomial distribution and use odd ratio to measure the relative efficacy between treatments. We also provide brief descriptions of network meta-analysis, such as basic parameters and functional parameters, and consider four different types of networks (star, loop, ladder, and one closed-loop geometry). Best rank probability is introduced to assess the ability to detect the best treatment among all the treatments. In the simulation studies, we vary the percentage of zero outcomes in the data. The simulation results confirm that with more zero in the outcome , the bias of estimates increased. Continuity correction methods do reduce the bias of esitmates, but there is no universal best method for all types of network, the performance varies from case to case.

In Chapter Four, we compare the continuity correction methods for zero events in network meta-analysis for Poisson data. We perform a very similar analysis as

what we do in Chapter Three, four types of network patterns and two continuity corrections methods. Besides, we assume the underlying distribution for data is Poisson distribution and use risk ratio to measure the efficacy between treatments.

In Chapter Five, we incorporate sensitivity and specificity in the network meta-analysis for binary data when misclassification exists. In this chapter, only star geometry is considered, the other geometry can be easily extended. In the simulation studies, we investigate the danger of ignoring misclassification, which understimates the odds ratio between treatments. We also perform a sensitivity analysis of using imperfect misclassification parameters and simulation suggests that the use of slightly overestimated misclassification paramters will provide better results than that of using underestimated misclassification parameters.

Appendices are provided at the end of dissertation including the relative codes through this dissertation.

CHAPTER TWO

A Bayesian Method to Account for Misclassified and Overdispersed Data

This chapter published as: Wu, Wenqi, James Stamey, and David Kahle. 2015 "A Bayesian Approach to Account for Misclassification and Overdispersion in Count Data." *International Journal of Environmental Research and Public Health* 12(9):10648-10661.

2.1 Introduction

Epidemiologic studies often have data that are subject to a wide array of different types of error. Measurement error, unmeasured confounding, and selection bias are all examples of biased estimator sources and cause reduced power for hypothesis tests (Edwards et al. 2013; Pierce and VanderWeele 2012). For continuous covariates, the problem of imperfect assessment is referred to as measurement error. A thorough review of measurement error, including remedial measures, can be found in Carroll et al. (2006). When considering discrete covariates, binary exposure variables are often measured with error, and such error is known to yield biased estimators (Batterman et al. 2014; Burstyn, Yang, and Schnatter 2014). Misclassification in ordinal covariates is also considered by Küchenhoff, Mwalili, and Lesaffre (2006). In this binary setting, misclassification is communicated in the language of diagnostic tests: sensitivities (se) and specificities (sp). On the response side of the model, imperfect assessment for both binary/categorical variables (response misclassification) and count variables are also taken into account (Sposto et al. 1992; Luta et al. 2013).

In the case of Poisson regression, two approaches taken to correct for misclassification error: one frequentist and one Bayesian. From the frequentist perspective, Edwards et al. (2013) consider the problem using maximum likelihood (ML) techniques assuming fixed and known se and sp. From the Bayesian perspective, Stamey,

Young, and Seaman (2008) relax the assumption that the *se* and *sp* are known by making use of validation data or informative priors. Both approaches have strengths and weaknesses. Treating the *se* and *sp* as known, as done by Edwards et al. (2013), is a strong assumption when these values are determined from small prior data sets or expert opinion. However, the "fixed" *se* and *sp* approach is usually enhanced by performing a sensitivity study, where a range of values for the *se* and *sp* are plugged in and the impact to the estimated parameters of interest is noted. On the other hand, the Bayesian model of Stamey et al. (2008), which fully accounts for the uncertainty in estimation, may require considerable computational time for the Markov chain Monte Carlo (MCMC) to converge and might also require a sensitivity analysis. Assuming these parameters are fixed and known reintroduces identifiability into the model so that the parameters are estimable from either the ML or Bayesian perspectives.

In some cases, extra-variability beyond what is allowed for with a Poisson distribution is observed in the data; this effect is called overdispersion. Gorman et al. (2014) consider the effect of non-response on the estimation of alcohol-related outcomes using a Poisson model with overdispersion. Milner, Morrell, and LaMontagne (2014) consider this problem in an analysis of how the 2007 recession impacted suicide rates in Australia. Paulino, Silva, and Achcar (2005) considered a misclassified binomial model with overdispersion and allowed for extra variability in their model with a random effect. In this chapter, we focus on the cases where sampling is done either in clusters or longitudinally, motivating the need for a random effects model.

Statisticians have done considerable research developing methods for correcting biases in observational data due to misclassification, measurement error, unmeasured confounding, etc. In this chapter, we also focus on the important case of count data with misclassification and provide a cohesive estimating procedure for inference for a range of models of interest, specifically fixed and random effects models and

the cases of known and unknown misclassification rates. Our goal is to demonstrate and show the value of methods that account for bias in epidemiologic models with the Bayesian approach.

In Section 2.2, we overview the Poisson model with misclassification for both fixed and random effects. Then in Section 2.3, we discuss the prior distributions used for all parameters in the models and methods used for posterior inference. Next, we consider the analysis of a single synthetic dataset in Section 2.4. We then discuss the results of the simulation studies in Section 2.5 and give some concluding comments in Section 2.6.

Wenqi Wu and James Stamey conceptualized the simulation study. Wenqi Wu performed the simulation. David Kahle conceptualized and constructed the graphical models and other visuals. All authors wrote the paper.

2.2 The Model

In this section, we introduce the model of interest using an example from Edwards et al. (2014). To aid in the description, we build the model with increasing levels of complexity communicated through the language of directed graphical models, also called Bayesian networks. In that language, fixed known quantities are represented as dots and variables are represented as nodes (circles), which are shaded if they are to be observed as data. The directed edges (arrows) represent dependent relationships, with the defining property being that a node is conditionally independent of any node that is not one of its descendants given its parents. Nodes that have a double lining are considered deterministic/known given their parents. For more background on graphical models, we find that Koller and Friedman (2009) is an excellent, extensive treatment.

2.2.1 The No-Misclassification Poisson Regression Model

The overarching question of interest in this chapter is to understand the rate at which individuals succumb to diseases like lung cancer, and we are later interested in determinants of that rate. To begin modeling death due to lung cancer or alternative causes, we temporarily overlook covariates. The data are based on death certificate information, which consists of the number of deaths due to lung cancer (y_{1i}) and those due to other causes (y_{2i}). Here, the index i runs from 1 to n and represents a number of observations, e.g., counties where death certificate information is obtained. Both of these are counts gathered over the same opportunity size (t_i). In these scenarios, it is standard practice to assume that each of these counts follow a Poisson distribution with rate parameters λ and μ, respectively. The graphical model corresponding to this situation is presented in Figure 2.1. Note that the rectangle (plate notation) is graphical shorthand for iterated relationships so that the same λ and μ are the rates in each location. This is a naive assumption that will be relaxed shortly. This model yields the likelihood function,

$$p(\boldsymbol{y_1}, \boldsymbol{y_2}) \propto \prod_{i=1}^{n} \left(\lambda^{y_{1i}} \exp(-\lambda) \times \mu^{y_{2i}} \exp(-\mu)\right).$$

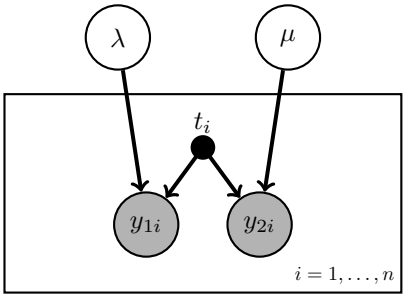


Figure 2.1: The naive baseline model: The number of deaths due to cancer (y_{1i}) and non-cancer (y_{2i}) follow a Poisson distribution with constant parameters.

To make the model more realistic, we let the rates vary depending on various covariates. In particular, we assume the rates are related to the covariates through

log-linear models:

$$\log(\lambda_i) \quad = \quad \beta_0 + \beta_1 X_i + \sum_{j=1}^{p} \beta_j Z_{ij} \quad = \quad \beta_0 + \beta_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\beta} \tag{2.1}$$

and

$$\log(\mu_i) \quad = \quad \gamma_0 + \gamma_1 X_i + \sum_{j=1}^{p} \gamma_j Z_{ij} \quad = \quad \gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\gamma}. \tag{2.2}$$

Here, X_i is the main exposure of interest for observation i, while Z_{ij} are other covariates associated with it; p is the number of non-treatment covariates. This model is diagramed in Figure 2.2. Note that bolded symbols represent vector quantities. The likelihood for the observed data is

$$p(y_{1i}, y_{2i}) \propto \lambda_i^{y_{1i}} \exp(-\lambda_i) \times \mu_i^{y_{2i}} \exp(-\mu_i)$$

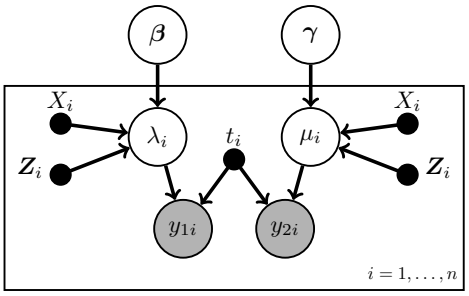


Figure 2.2: The no-misclassification Poisson regression model.

2.2.2 The Misclassification Poisson Regression Model

Outcomes, such as lung cancer deaths, are based on death certificate information, which is well known to be potentially misclassified (Sposto et al. 1992; Hinchliffe, Abrams, and Lambert 2013). In particular, we note that it may be naive to consider the observed quantities to be the true cancer and non-cancer death counts, as indicated in the two previous models. Instead, we add an additional layer of complexity to the model by considering the observed data to be the counts of deaths that are labeled as due to lung cancer (w_{1i}) or other causes (w_{2i}). The true number

of deaths due to lung cancer y_{1i} and non-lung cancer y_{2i} are thus considered unobserved. Since y_{1i} and y_{2i} are not observed, the error-prone data w_{1i} and w_{2i} are used in the analysis instead. The misclassified counts w_{1i} and w_{2i} depend on the true counts y_{1i} and y_{2i} and misclassified counts u_{1i} and u_{2i} (also unobserved) in the following ways: the count u_{1i} is the number of cancer deaths incorrectly labeled as non-cancer, and u_{2i} is the number of non-cancer deaths incorrectly labeled as due to cancer. We then have

$$w_{1i} = y_{1i} - u_{1i} + u_{2i},$$

$$w_{2i} = y_{2i} + u_{1i} - u_{2i},$$

and

$$w_{1i} + w_{2i} = y_{1i} + y_{2i}.$$

The relationships between the observed variables w_{1i} and w_{2i} and the unobserved variables are shown in diagram and table form in Figure 2.3. Note that CC_i represents the true cancer deaths classified as cancer deaths (correct), $C\bar{C}_i$ represents the true cancer deaths classified as non-cancer deaths (incorrect), and the classifications are similar for $\bar{C}C_i$ and $\bar{C}\bar{C}_i$. Another way to describe the potential misclassification is to use *se* and *sp*. Here, *se* is the probability that a lung cancer death is correctly labeled as lung cancer; *sp* is the probability that a death due to all other causes are correctly labeled as not being due to lung cancer. Therefore, the true rate parameter for observed data for the counts of deaths that are labeled as due to lung cancer (w_{1i}) is

$$\lambda_i se + \mu_i(1 - sp),$$

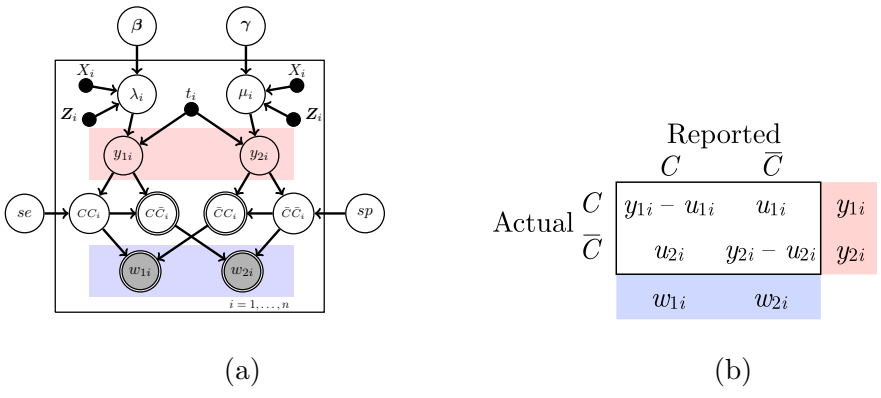
and the true rate parameter for observed data for the counts of deaths that are labeled as due to other causes (w_{2i}) is

$$\lambda_i(1 - se) + \mu_i sp,$$

where λ_i and μ_i are the covariate specific death rates for the ith observational unit. Thus, w_{1i} and w_{2i} are biased for the rates λ_i and μ_i. For instance, w_{1i} only provides information about the quantity $\lambda_i se+\mu_i(1-sp)$, and without additional information, there is no way to disentangle a direct estimate for λ_i. In other words, accounting for misclassification with the two parameters of se and sp overparameterizes the model in a way that demands to be addressed. A simple derivation reveals that like the unobserved counts, the observed counts also follow Poisson distributions, but the rates are functions of all the parameters. The likelihood for the observed data is

$$p(w_{1i}, w_{2i}) \propto (\lambda_i se + \mu_i(1-sp))^{w_{1i}}(\lambda_i(1-se) + \mu_i sp)^{w_{2i}}\exp\{-(\lambda_i+\mu_i)n_i\}. \quad (2.3)$$

Edwards et al. (2014) consider se and sp known, and they provide a method to obtain the maximum likelihood estimators (MLEs). Stamey et al. (2008) assume information about se and sp exists, not as point estimates, but rather in the form of probability distributions with which one can perform a Bayesian analysis, e.g., prior distributions. Here, we investigate both the fixed and unknown approaches via the Bayesian paradigm.



(a) (b)

Figure 2.3: The Poisson regression model with misclassification

Figure 2.3 shows the Poisson regression model with misclassification. (**a**) is the graphical representation of the model, se denotes the sensitivity of the classifier,

and *sp* denotes its specificity. (**b**) is the contingency table representation of the data. y_{1i} (y_{2i}) is the true number of deaths due to lung cancer (non-lung cancer). u_{1i} (u_{2i}) is the true number of lung cancer (non-lung cancer) deaths misclassified. w_{1i} (w_{2i}) is the observed number of deaths due to lung cancer (non-lung cancer). Note that C and $\bar{C}$ denote correctly classified and misclassified deaths, respectively.

2.2.3 The Misclassification Poisson Regression Model with Extra Variability

The model described above is the same one used in both Stamey et al. (2008) and Edwards et al. (2014) and does not allow for extra Poisson variability. If the sampling is done in clusters or if there are repeated measures, a random effects model may be appropriate as an alternative to the fixed effects model already described. When using random effects, the log-linear models in Equations 2.1 and 2.2 become

$$\log(\lambda_i) = \beta_0 + \beta_1 X_i + \sum_{j=1}^{p} \beta_j Z_{ij} + e_{\lambda_{k[i]}} = \beta_0 + \beta_1 X_i + \boldsymbol{Z}_i'\boldsymbol{\beta} + e_{\lambda_{k[i]}} \quad (2.4)$$

and

$$\log(\mu_i) = \gamma_0 + \gamma_1 X_i + \sum_{j=1}^{p} \gamma_j Z_{ij} + e_{\mu_{k[i]}} = \gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i'\boldsymbol{\gamma} + e_{\mu_{k[i]}}. \quad (2.5)$$

Here, $e_{\lambda_{k[i]}} \sim \mathcal{N}(0, \sigma^2)$ and $e_{\mu_{k[i]}} \sim \mathcal{N}(0, \sigma^2)$. This model is sometimes referred to as a random intercept model. If a more complicated hierarchical structure is desired, the slopes could also be modeled, but that is beyond what we are interested in for this work.

Combining Equation 2.3 with Equations 2.4 and 2.5, the likelihood function for the log normal structure in the product-Poisson setting for the data (w_{1i}, w_{2i}) is

$$
\begin{aligned}
L(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{e}_\lambda, \boldsymbol{e}_\mu, \sigma^2 | w_{1i}, w_{2i}) &= \prod_{i=1}^{n} p(w_{1i}, w_{2i}) f(e_{\lambda_{k[i]}}, e_{\mu_{k[i]}}) \\
&\propto \prod_{i=1}^{n} (\lambda_i se + \mu_i(1-sp))^{w_{1i}} (\lambda_i(1-se) + \mu_i sp)^{w_{2i}} \\
&\times \exp\{-(\lambda_i + \mu_i) n_i\} \\
&\times \exp[-(e^2_{\lambda_{k[i]}} + e^2_{\mu_{k[i]}})/(2\sigma^2)]/\sigma^2,
\end{aligned}
$$

where λ_i and μ_i are presented by Equations 2.4 and 2.5.

Here, we consider the usual representation of the prior information for a generalized linear mixed model (GLMM):

$$
\boldsymbol{\beta} \sim \boldsymbol{N}_{p+2}(\boldsymbol{b}_\beta, \boldsymbol{B}_\beta), \quad \boldsymbol{\gamma} \sim \boldsymbol{N}_{p+2}(\boldsymbol{b}_\gamma, \boldsymbol{B}_\gamma), \quad \sigma^2 \sim IG(a_0, a_1).
$$

That is to say, the fixed effects parameter with every hyperparameter pre-specified has a normal distribution and an inverse gamma distribution for σ^2, independent of $\boldsymbol{\beta}$. Furthermore, the misclassification probabilities are each given an independent beta distribution with known hyperparameters:

$$
se \sim \text{Beta}(a_{se}, b_{se}), \quad sp \sim \text{Beta}(a_{sp}, b_{sp}).
$$

Previous studies have shown that for the purpose of MCMC application, it is inconvenient to sample using only misclassified data. This issue can be addressed by introducing latent variables. Let u_{1i} and u_{2i} be the number of lung cancer and non-lung cancer deaths misclassified for the i^{th} covariate pattern, and let v_{1i} and v_{2i} be the number of lung cancer and non-lung cancer deaths correctly classified for the i^{th} covariate pattern, and let y_{1i} and y_{2i} be the true number of deaths due to lung cancer and other cause. Note that w_{1i} and w_{2i} are the partial sums that are observed. Therefore,

$$
v_{1i} = w_{1i} - u_{2i}, \quad v_{2i} = w_{1i} + u_{2i},
$$

and

$$
y_{1i} = w_{1i} + u_{1i} - u_{2i}, \quad y_{2i} = w_{2i} - u_{1i} + u_{2i}.
$$

The likelihood of the augmented data is

$$L(\boldsymbol{\beta}, \boldsymbol{\gamma}, se, sp, \sigma^2 | u, w) \propto \prod_{i=1}^{n} \lambda_i^{y_{1i}} e^{-\lambda_i} \times \mu_i^{y_{2i}} e^{-\mu_i} \times se^{v_{1i}} (1-se)^{u_{1i}} sp^{v_{2i}} (1-sp)^{u_{2i}}. \quad (2.6)$$

Hence, the augmented data are distributed conditionally upon the observed data w_{1i} and w_{2i} as independent Poisson distributions.

If we combine Equation 2.6 with all prior distributions, the joint posterior distribution given the augmented data is

$$\begin{aligned}
q(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{e}_\lambda, \boldsymbol{e}_\mu, \sigma^2 | \boldsymbol{w}, \boldsymbol{u}) \quad \propto \quad & \prod_{i=1}^{n} \exp[(w_{1i} + u_{1i} - u_{2i})(\beta_0 + \beta_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\beta} + e_{\lambda_{k[i]}}) \\
& - \exp(\beta_0 + \beta_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\beta} + e_{\lambda_{k[i]}})] \\
& \times \exp[-(\boldsymbol{\beta} - \boldsymbol{b}_\beta)' \boldsymbol{B}_\beta (\boldsymbol{\beta} - \boldsymbol{b}_\beta)/2] \\
& \times \prod_{i=1}^{n} \exp[(w_{2i} - u_{1i} + u_{2i})(\gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\gamma} + e_{\mu_{k[i]}}) \\
& - \exp(\gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\gamma} + e_{\mu_{k[i]}})] \\
& \times \exp[-(\boldsymbol{\gamma} - \boldsymbol{b}_\gamma)' \boldsymbol{B}_\gamma (\boldsymbol{\gamma} - \boldsymbol{b}_\gamma)/2] \\
& \times \prod_{i=1}^{n} se^{w_{1i} - u_{2i} + a_{se} - 1} (1 - se)^{u_{1i} + b_{se} - 1} \\
& \times \prod_{i=1}^{n} sp^{u_{2i} + a_{sp} - 1} (1 - sp)^{w_{2i} - u_{1i} + b_{sp} - 1} \\
& \times \exp\left[-\frac{1}{\sigma^2} \left(a_1 + \frac{1}{2} \sum_{i=1}^{n} (e_{\lambda_{k[i]}}^2 + e_{\mu_{k[i]}}^2) \right) \right] \times \sigma^{-2(a_0 + n + 1)}.
\end{aligned}$$

The posterior samples can be drawn by Gibbs sampling algorithm, and the conditional parameter distributions are given as follows:

(1) $q(\boldsymbol{\beta} | \boldsymbol{e}_\lambda, \sigma^2, w, u) \propto \prod_{i=1}^{n} \exp[(w_{1i} + u_{1i} - u_{2i})(\beta_0 + \beta_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\beta} + e_{\lambda_{k[i]}}) - \exp(\beta_0 + \beta_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\beta} + e_{\lambda_{k[i]}})] \times \exp[-(\boldsymbol{\beta} - \boldsymbol{b}_\beta)' \boldsymbol{B}_\beta (\boldsymbol{\beta} - \boldsymbol{b}_\beta)/2]$,

(2) $q(\boldsymbol{\gamma} | \boldsymbol{e}_\mu, \sigma^2, w, u) \propto \prod_{i=1}^{n} \exp[(w_{2i} - u_{1i} + u_{2i})(\gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\gamma} + e_{\mu_{k[i]}}) - \exp(\gamma_0 + \gamma_1 X_i + \boldsymbol{Z}_i' \boldsymbol{\gamma} + e_{\mu_{k[i]}})] \times \exp[-(\boldsymbol{\gamma} - \boldsymbol{b}_\gamma)' \boldsymbol{B}_\gamma (\boldsymbol{\gamma} - \boldsymbol{b}_\gamma)/2]$,

(3) $se | \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{e}_\lambda, \boldsymbol{e}_\mu, \sigma^2, w, u \sim \text{Beta}(\sum_{i=1}^{n} (w_{1i} - u_{2i}) + a_{se}, \sum_{i=1}^{n} u_{i1} + b_{se})$,

(4) $sp|\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{e}_\lambda, \boldsymbol{e}_\mu, \sigma^2, w, u \sim \text{Beta}(\sum_{i=1}^n u_{2i} + a_{sp}, \sum_{i=1}^n (w_{2i} - u_{1i}) + b_{se})$,

(5) $\sigma^2|\boldsymbol{e}_\lambda, \boldsymbol{e}_\mu, w, u \sim IG(a_0 + n, a_1 + \frac{1}{2}\sum_{i=1}^n (e^2_{\lambda_{k[i]}} + e^2_{\mu_{k[i]}})$,

(6) $u_{1i} \sim \text{Binomial}(w_{1i}, \mu_i(1-se)/(\lambda_i se + \mu_i(1-sp)))$,

(7) $u_{2i} \sim \text{Binomial}(w_{2i}, \lambda_i(1-sp)/(\lambda_i(1-se) + \mu_i sp))$.

Simulating values from (3), (4), (5), (6), and (7) is straightforward; for sampling from (1) and (2), we can use the Metropolis Hastings algorithm.

We initially assume the random effects come from a common distribution to limit the number of parameters required to be estimated, and in general, these variances would unlikely be largely different. However, this is a very strong assumption, so we also provide a more flexible model. For this more general model, we assume $e_{\lambda_{k[i]}}$ and $e_{\mu_{k[i]}}$ are bivariate normal,

$$\boldsymbol{e} \sim \mathcal{N}_2 \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \right), \tag{2.7}$$

where ρ is the Pearson's correlation coefficient of the two groups.

2.3 Priors and Posterior Inference

We consider four different models. Model 1 is a Bayesian version of the model in Edwards et al. (2014). That is, it is the fixed effects Poisson model, and the diagnostic parameters *se* and *sp* are assumed to be known as it is in Figure 2.3, with *se* and *sp* fixed. Model 2 is the model of Stamey et al. (2008), which is also a fixed effects Poisson model, but it allows for uncertainty in *se* and *sp* by replacing the fixed values with beta priors for those two parameters; this is seen in Figure 2.3. Model 3 extends Model 1 by adding a random effect to account for clustered sampling designs, which is shown in Figure 2.4 with fixed *se* and *sp* , and Model 4 adds a random effect to Model 2, which is also shown in Figure 2.4.

For all models, we assume relatively diffuse independent normal priors for the regression coefficients. Specifically, we have

$$\beta_j \sim \mathcal{N}(0, 10), \quad j = 0, 1, ..., p$$

and

$$\gamma_j \sim \mathcal{N}(0, 10), \quad j = 0, 1, ..., p.$$

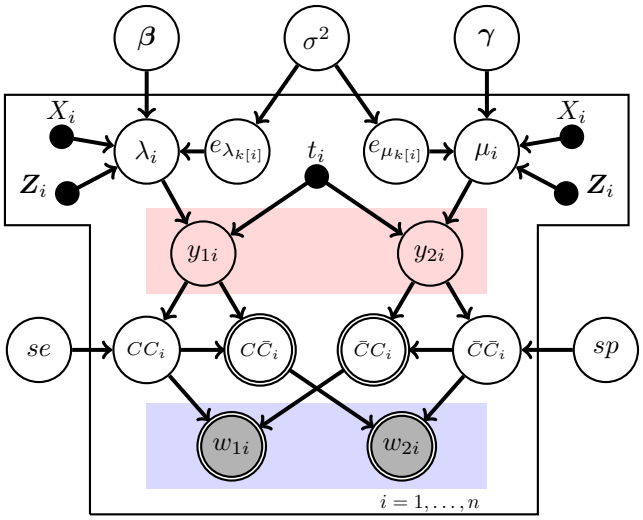


Figure 2.4: The random-intercept Poisson regression model with misclassification

For all the simulations we consider, a prior standard deviation of 10 for the regression parameters leads to a very diffuse prior relative to the likelihood. In practice, the user should consider likely values for these parameters when choosing the standard deviation for these priors. For a Poisson regression, priors allowing for values from –20 to 20 would almost always be sufficient; on the scale of the Poisson rate, this would allow for multiplicative effects ranging from zero to about 500 million.

For the diagnostic parameters *se* and *sp* in Models 2 and 4 where they are considered unknown, we assume independent beta priors,

$$se \sim \text{Beta}(a_{se}, b_{se}) \quad \text{and} \quad sp \sim \text{Beta}(a_{sp}, b_{sp}).$$

The beta is a flexible distribution allowing for a wide variety of shapes. Also, it is the conjugate prior for a binomial likelihood, so if validation data for both the true and fallible results are available, the values can be used to specify the a's and b's. For instance, Sposto et al. (1992) consider the impact of misclassification on cancer rate estimates in Hiroshima and Nagasaki. Most of their data is based on fallible death certificate information, but for a subset of the deaths, autopsies were performed and could be treated as a "gold standard" (e.g., infallible or perfectly classified data). For instance, suppose that in the validation data, there are m autopsied subjects known to have died from cancer, but the death certificates only correctly labeled w of them. The likelihood for this data is thus binomial:

$$\binom{m}{w} se^w(1-se)^{m-w}. \tag{2.8}$$

This data leads directly to a Beta$(w+1, m-w+1)$ prior for the se. In combining the autopsy and death certificate information, the same sort of prior for sp can be obtained. Alternatively, the beta priors can be viewed as a mechanism to perform a sensitivity analysis. While Edwards et al. (2014) assumed the se and sp are known, they considered a wide range of values as part of a sensitivity analysis. Using similar logic, the prior parameters for se and sp can be selected so that the priors reflect the range for the se and sp of interest.

The random effects standard deviation in Models 3 and 4 is the final parameter requiring a prior distribution. The conjugate prior for a variance in this situation is the inverse gamma, so an InvGamma$(0.001, 0.001)$ is often used. Gelman (2006) finds that both a half-Cauchy distribution and a uniform distribution perform better as priors than the inverse gamma. Here, we use a Unif$(0, D)$ prior where D is an upper value for the support that is chosen to minimize influence on the posterior. A value of 5 will often be large enough but should be checked in each unique situation.

If the correlated random effects model is used instead of the equal variance

model, then priors for σ_1, σ_2, and ρ are required. In the absence of substantial prior information, uniform priors for all three parameters would often be used. Specifically, $\sigma_1 \sim \text{Unif}(0, B_1)$, $\sigma_2 \sim \text{Unif}(0, B_2)$, and $\rho \sim \text{Unif}(-1, 1)$.

We fit the models using MCMC methods via the free package OpenBUGS. The OpenBUGS code used for fitting the models and the R code used to generate all the data for simulations are in the appendix. As always when using MCMC methods, care must be taken to ensure the validity of the results. For our simulated models where the misclassification parameters are assumed unknown, there are times when the chains did not mix sufficiently, indicating a lack of convergence clearly visible in the trace plots of the MCMC. This is not unusual for overparameterized models such as these. When chains illustrate a lack of convergence, remedial measures, including increasing the number of burn-in iterations and thinning the chains, improved convergence. Another important issue is starting values for the chains. The test parameters *se* and *sp* must have a sum greater than 1, or the classifying technique is actually worse than random guessing. Thus, starting values for *se* and *sp* should be chosen so that $se + sp > 1$.

2.4 Simulated Example

In this section, we consider a simulated example to illustrate the new random effects model and how the models can be used for sensitivity analysis. We imagine a scenario where interest is in the relationship between lung cancer deaths due to a particular exposure. We suppose that the data are clustered, with each cluster containing four observations. Thus, the random effects model is appropriate. We generate the data using Equations 2.4 and 2.5 with three binary covariates, a total sample size of $N = 32$ observations and $K = 8$ clusters. The parameter values chosen result in models

$$\log(\lambda_i) = -1 + 0.5x_{1ij} - 0.5x_{2ij} + 0.1x_{ij} + e_{\lambda_{k[i]}}$$

and

$$\log(\mu_i) = -2 - 0.3x_{1ij} + 0.2x_{2ij} + 0.5x_{ij} + e_{\mu_{k[i]}}.$$

To generate the data, instead of assuming a common standard deviation for the random effects, we assume $(e_{\lambda_{k[i]}}, e_{\mu_{k[i]}})$ comes from a bivariate normal distribution with means of 0, $\sigma_1 = 0.2$, $\sigma_2 = 0.4$, and $\rho = 0.5$. We assume 1000 person-months for each observation. Finally, we assume the true *se* is 0.75 and the true *sp* is 0.8. The counts for each observation range from a low of 148 to a high of 613, so this classifies as a relatively large observational study.

Suppose an expert thinks the most likely value for the *se* is 0.7, and it could (with a 5% chance) be as low as 0.6 and as high as 0.8. For the *sp*, the most likely value is 0.8 with a 5% chance of being lower than 0.7 and as high as 0.9. These beliefs could be roughly summarized into $se \sim \text{Beta}(35, 15)$ and $sp \sim \text{Beta}(40, 10)$.

Before discussing the overall results and illustrating how to use the methods to perform a sensitivity analysis, we compare estimates for the correlated random effects and single standard deviation models. Even though σ_2 is twice σ_1, the posterior estimates for the regression parameters are almost identical. For instance, the primary parameter of interest is β_1. The posterior mean and 95% interval are 0.541 and (0.454, 0.646) for the correlated model and 0.540 and (0.452, 0.643) for the single variance model. Due to the over-parameterization that is already in the model because of misclassification, unless strong evidence against the equal standard deviation model exists, we recommend using it instead of the correlated random effects model.

One advantage of using informative priors on *se* and *sp* instead of using fixed values is that the analysis of the data doubles as a sensitivity analysis. There is essentially no information in the data on *se* and *sp*. Thus, the posterior distributions are approximately the same as the prior distributions. If we desire to assure that

the priors and posteriors for the misclassification parameters completely match, the cut function in WinBUGS and OpenBUGS could be used so that the priors for *se* and *sp* exactly match the posteriors. In this case, the Bayesian analysis is a version of Monte Carlo sensitivity advocated for in Steenland and Greenland (2004). As mentioned before, β_1 is the primary parameter of interest.

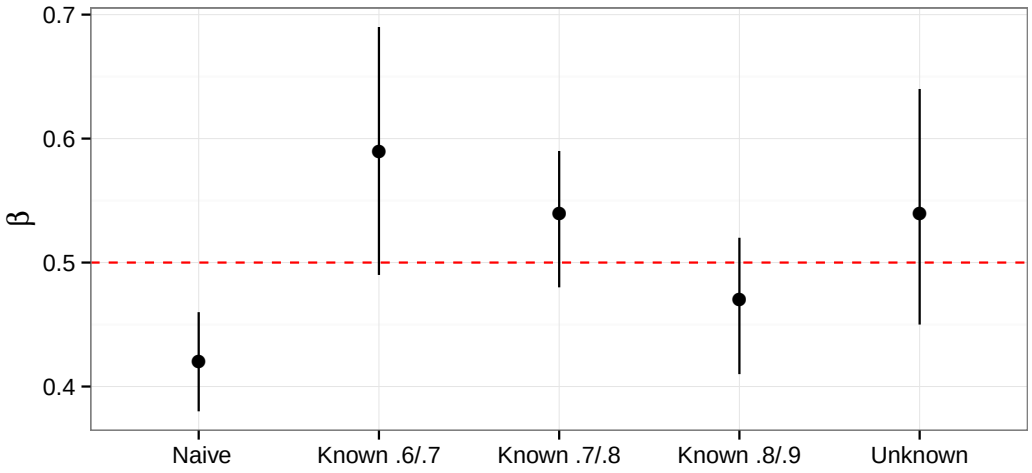


Figure 2.5: Posterior means and 95% credible sets for sensitivity analysis of β_1 (with true value 0.5)

In Figure 2.5, we plot the posterior means and the 95% intervals for β_1 for the naive model where the misclassification is ignored (Model 4), for when *se* and *sp* are given prior distributions, and for several versions of Model 3. For Model 3, where *se* and *sp* are fixed, we consider the following (se, sp) pairs: (0.8, 0.9), (0.7, 0.8), and (0.6, 0.7). These pairs represent the most optimistic, the most likely, and the most pessimistic values, according to the expert. The misclassification in the data biases the estimates towards the null value of 0, which is why the naive model has the lowest posterior mean. Note also that the interval for the naive model does not contain the true value. For Model 3, the pessimistic choice of (0.6, 0.7) shifted β_1 upwards the most. One approach to the sensitivity analysis would be to take the upper limit of the interval for the (0.6, 0.7) posterior and the lower limit of the (0.8,

0.9) posterior. This would yield an interval of (0.41, 0.69). Another option would be to simply use the interval that corresponds to the most likely pair of (0.7, 0.8). This would yield an interval of (0.48, 0.59). What is interesting is that the Monte Carlo sensitivity analysis of Model 4 with a 95% interval of (0.45, 0.64) provides a very nice intermediate step between these two extremes. In the next section, we investigate the operating characteristics of these procedures via simulation.

2.5 Simulation Study

We conducted a series of simulations to illustrate the performance of the four models under various situations. For these simulations, we focused on inference, specifically the average of the posterior means, along with the width and coverage of 95% intervals. The code we used could easily be modified to include hypothesis testing and monitor quantities, such as Type I error rates and power. For the simulation, we assumed three binary covariates, and for each covariate pattern, we assumed the person-time $t_i = 1000$. In anticipation of analyzing both fixed and random effects models, we actually generated three counts for each covariate pattern. For the fixed effects model, these counts were independent, but for the random effects model, they were correlated. For each simulation configuration, we generated 1000 data sets with 32 observations each. The regression parameters were kept the same throughout the simulations and are provided in the following equations:

$$\log(\lambda_i) = -2 - 0.3X_1 + 0.2Z_1 + 0.5Z_2, \tag{2.9}$$

and

$$\log(\mu_i) = -1 + 0.5X_1 - 0.5Z_1 + 0.1Z_2. \tag{2.10}$$

We considered various values of the *se* and *sp* and the random effect variance, σ^2.

Previously, Edwards et al. (2014) and Stamey et al. (2008) performed fixed effects model simulations for the situation of correctly-known misclassification and prior information in the form of validation data that is centered on the true value.

We repeated simulations similar to theirs and verified that the fixed effects models perform very well in terms of bias and coverage when either of the correct values for *se* and *sp* are used (Model 1) and when the priors are centered at the true values (Model 2). That is, when the misclassification is correctly modeled, there is very little bias, and coverage of the 95% intervals is close to nominal. We next considered this same situation for the random effects model. Specifically, we generated three correlated counts of each covariate pattern for the random effect variance values of 0.1, 0.25, 0.5, and 0.75. We did this for $se = 0.9$ and $sp = 0.8$ and for $se = 0.9$ and $sp = 0.6$. We analyzed the data both with the naive model, where the misclassification is ignored, and with the model where the misclassification is accounted for, with priors for *se* of Beta$(45, 5)$ and *sp* of Beta$(40, 10)$ for both cases of misclassification parameters. These priors had means of 0.9 and 0.8 with standard deviations of 0.042 and 0.056, respectively. We focused on the results for β_1 and γ_1, but the general patterns were similar for all parameters.

Table 2.1: Average Posterior Means Across 1000 Simulations (Truth: $\beta_1 = -0.3$, $\gamma_1 = 0.5$)

$se = \mathbf{0.9}$, $sp = \mathbf{0.8}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Naive)**	$\boldsymbol{\beta_1}$ **(Model 4)**	$\boldsymbol{\gamma_1}$ **(Naive)**	$\boldsymbol{\gamma_1}$ **(Model 4)**
	0.10	–0.11	–0.30	0.43	0.51
	0.25	–0.11	–0.32	0.42	0.50
	0.50	–0.10	–0.33	0.40	0.50
	0.75	–0.08	–0.34	0.39	0.50
$se = \mathbf{0.9}$, $sp = \mathbf{0.6}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Naive)**	$\boldsymbol{\beta_1}$ **(Model 4)**	$\boldsymbol{\gamma_1}$ **(Naive)**	$\boldsymbol{\gamma_1}$ **(Model 4)**
	0.10	–0.07	–0.31	0.36	0.50
	0.25	–0.06	–0.32	0.35	0.50
	0.50	–0.05	–0.33	0.34	0.49
	0.75	–0.05	–0.33	0.32	0.49

Table 2.2: Average Width of 95% Intervals Across 1000 Simulations

$se = 0.9$, $sp = 0.8$	σ^2	β_1 (Naive)	β_1 (Model 4)	γ_1 (Naive)	γ_1 (Model 4)
	0.10	0.52	0.66	0.46	0.53
	0.25	0.70	0.88	0.66	0.80
	0.50	0.93	1.15	0.89	1.06
	0.75	1.09	1.37	1.02	1.28
$se = 0.9$, $sp = 0.6$	σ^2	β_1 (Naive)	β_1 (Model 4)	γ_1 (Naive)	γ_1 (Model 4)
	0.10	0.53	0.76	0.44	0.56
	0.25	0.70	0.95	0.63	0.78
	0.50	0.91	1.21	0.86	1.08
	0.75	1.07	1.44	1.02	1.31

Table 2.3: Average Coverage of 95% Intervals Across 1000 Simulations

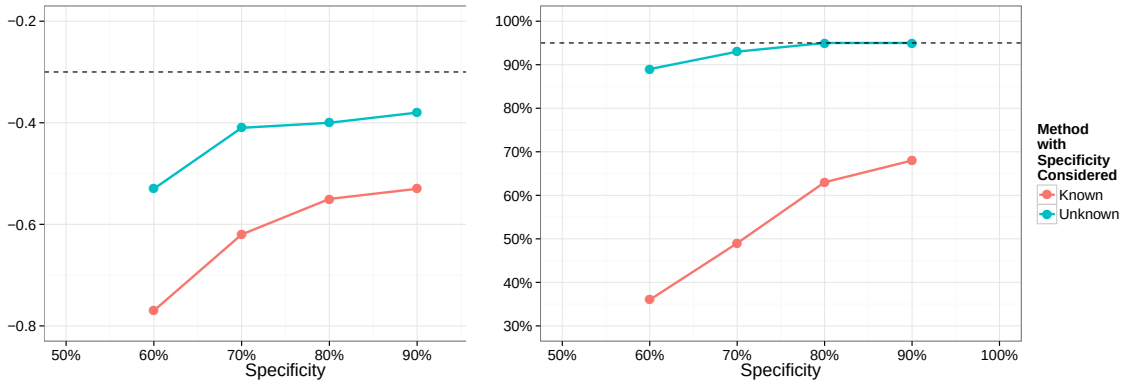
$se = 0.9$, $sp = 0.8$	σ^2	β_1 (Naive)	β_1 (Model 4)	γ_1 (Naive)	γ_1 (Model 4)
	0.10	0.72	0.95	0.90	0.96
	0.25	0.81	0.95	0.91	0.94
	0.50	0.88	0.96	0.91	0.95
	0.75	0.90	0.96	0.93	0.95
$se = 0.9$, $sp = 0.6$	σ^2	β_1 (Naive)	β_1 (Model 4)	γ_1 (Naive)	γ_1 (Model 4)
	0.10	0.59	0.96	0.74	0.95
	0.25	0.74	0.94	0.86	0.95
	0.50	0.81	0.96	0.86	0.94
	0.75	0.85	0.96	0.87	0.94

In Table 2.1, we display the averages of the posterior means for both the naive and "corrected" models for the $se = 0.9$ and $sp = 0.8$ and the $se = 0.9$ and $sp = 0.6$ cases across the values for σ^2. For the same scenarios, Tables 2.2 and 2.3

report 95% interval widths and coverages, respectively. In both cases and for all parameters, the naive model yielded biased estimators with an empirical coverage probability below nominal, while the corrected model had posterior means close to the truth and coverage close to 95%. It is also interesting that the naive model had narrower intervals. Accounting for the misclassification increased the uncertainty in the model, leading to wider intervals. The narrower intervals for the naive case contributed to the below nominal coverage, as it leads to estimates being "precisely wrong" (biased and overly confident).

2.5.1 Robustness Considerations

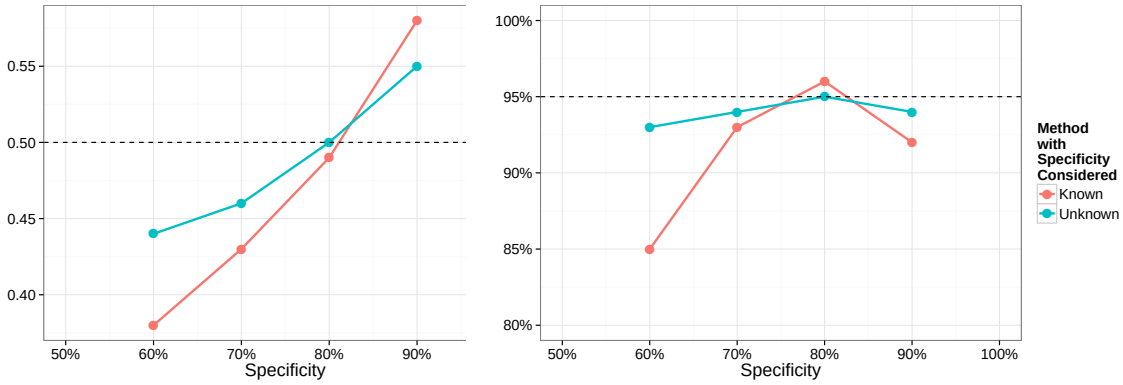
We next investigated robustness. Specifically, we were interested in the impact of imperfect estimation of *se* and *sp*. We focus on Models 3 and 4 here, but the results were similar for Models 1 and 2. For Model 3, we assumed a value of 0.7 for *se* and of 0.8 for *sp*. For Model 4, we centered the priors on these same values, with a Beta$(35, 15)$ for *se* and a Beta$(40, 10)$ for *sp* (with means of 0.7 and 0.8 and standard deviations of 0.064 and 0.056, respectively). For the simulation, we fixed *se* at 0.75, shifting it mildly from the true value. For the *sp*, we considered a range of values: $0.9, 0.8, 0.7$, and 0.6.



(a) (b)

Figure 2.6: Posterior means (**a**) and coverage rates (**b**) for γ; $se = 0.75$

In Figures 2.6 and 2.7, we provided the average posterior means along with the coverage of the nominal 95% intervals for both β_1 and γ_1. The most notable aspect of the results is that while the posterior means were biased for both models, Model 4 was not nearly as biased and held the coverage close to nominal. Conversely, Model 3 seemed to be surprisingly sensitive. It is interesting to note that if we generated the data from the exact distribution assumed (that is, a value of 0.7 for *se* and 0.8 for *sp*), then the estimation in Model 3 exhibited little bias and had a 95% coverage. However, we see that the estimation of γ_1 was quite poor in every case, and the coverage for β_1 dipped for lower values of *sp*.



(a) (b)

Figure 2.7: Posterior means (**a**) and coverage rates (**b**) for β; $se = 0.75$

We were also interested in the impact of ignoring overdispersion in observed data. Specifically in the simulation study, we generated data with a random effect, but we used a fixed effects Poisson model in the analysis. For simplicity, we assumed both diagnostic parameters *se* and *sp* were known. This could easily be extended to the case when both diagnostic parameters are unknown. For data generation, we considered two values of the random effect variance, $\sigma^2 = 0.1, 0.25$, and several pairs of *se* and *sp*, (*se*, *sp*) = (0.9,0.9),(0.9,0.7),(0.8,0.8),(0.7,0.9). In Table 2.4, we display the average of the posterior means and coverages for both β_1 and γ_1. In

all cases, the estimators of both β_1 and γ_1 were close to true values. However, in terms of coverage probability, they were all well below nominal. The cases with $\sigma^2 = 0.1$ provided better coverage probabilities than that with $\sigma^2 = 0.25$. It is also interesting that the smaller *se* and *sp* gave larger coverage probabilities for both β_1 and γ_1. Because smaller *se* and *sp* increased the uncertainty in the model leading to wider intervals, the coverage probabilities increased.

Table 2.4: Posterior Summaries Across 1000 Simulations (Truth: $\beta_1 = -0.3$, $\gamma_1 = 0.5$)

$se = \mathbf{0.9}$, $sp = \mathbf{0.9}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Mean)**	$\boldsymbol{\beta_1}$**(Coverage)**	$\boldsymbol{\gamma_1}$ **(Mean)**	$\boldsymbol{\gamma_1}$ **(Coverage)**
	0.10	–0.30	0.72	0.50	0.63
	0.25	–0.26	0.37	0.50	0.20
$se = \mathbf{0.9}$, $sp = \mathbf{0.7}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Mean)**	$\boldsymbol{\beta_1}$**(Coverage)**	$\boldsymbol{\gamma_1}$ **(Mean)**	$\boldsymbol{\gamma_1}$ **(Coverage)**
	0.10	–0.29	0.77	0.5	0.70
	0.25	–0.29	0.61	0.51	0.50
$se = \mathbf{0.8}$, $sp = \mathbf{0.8}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Mean)**	$\boldsymbol{\beta_1}$**(Coverage)**	$\boldsymbol{\gamma_1}$ **(Mean)**	$\boldsymbol{\gamma_1}$ **(Coverage)**
	0.10	–0.30	0.79	0.5	0.73
	0.25	–0.29	0.65	0.51	0.52
$se = \mathbf{0.7}$, $sp = \mathbf{0.9}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\beta_1}$ **(Mean)**	$\boldsymbol{\beta_1}$**(Coverage)**	$\boldsymbol{\gamma_1}$ **(Mean)**	$\boldsymbol{\gamma_1}$ **(Coverage)**
	0.10	–0.30	0.82	0.51	0.75
	0.25	–0.27	0.61	0.50	0.54

2.6 Conclusion

In this chapter, we have extended previous work on count regression models with misclassification by including a random effect to allow for overdispersion commonly encountered in observational data. Using graphical models, we have attempted to make the assumptions and form of the models more accessible to a general

audience. Simulation results confirmed the performance of the model, demonstrating the improvement over the naive model, which could ignore the impacts of the misclassification substantially. We hope this work motivates researchers to account for misclassification, to consider the wide range of non-sampling bias that can be found in their observational data, and to apply appropriate tools to fully address the problems that can arise. Future work includes the development of software for epidemiologists and public health researchers in order to address misclassification in Poisson and related count models used for public health data.

CHAPTER THREE

Bayesian Network Meta-analysis for Binary Outcomes with Rare Events

3.1 Introduction

Meta-analysis is a method for systematically combining results from several studies in order to increase precision in estimating a parameter of interest. One advantage of meta-analysis is that inferences often have greater statistical power than tests based on any single study. Specifically, meta-analysis has more power to detect a small but significant effect (Greco et al. 2013). For example, when comparing drug A to drug B, there may be no significant difference in any single study, but if we combine all the studies together, drug A may be significantly superior than drug B.

One extension of standard meta-analysis is network meta-analysis. Standard meta-analysis focuses on comparing only two treatments. Network meta-analysis is an extension of standard meta-analysis that can handle the cases in which there are more than two treatments and not all pairwise comparisons are available. Furthermore, if two particular treatments have never been compared against each other but have been compared to a common comparator, then the treatment effect can be indirectly estimated by the direct effects of the two treatments versus the comparator.

One common issue in both standard meta-analysis and network meta-analysis is having zero outcomes when the response is a count. Difficulties arise when the analysis is done, either at the patient level using individual patient data or at the study level using only summary counts from each trial. Sankey et al. (1996) compare the corrected method, in which one half was added to each cell. The authors find that the uncorrected method performs better only when using the Mantel-Haenszel odds ratio with very little heterogeneity present. For all other sparse data applications, the

continuity correction performs better and is recommended for use in meta-analyses of similar scope. Bradburn et al. (2007) evaluate the performance of 12 methods for pooling rare events through simulation. They found that most of the commonly used meta-analytical methods are biased when data are sparse. Bhaumik et al. (2012) compare various methods for random effects meta-analysis. In the absence of heterogeneity, the Mantel and Haenszel method with the empirical continuity correction performs well, and it is to be recommended for moment-based fixed-effects meta-analysis. The simple average method, which was introduced by Shuster (2010), with a 0.5 continuity correction is recommended for sparse data with heterogeneity.

Compared to frequentist meta-analysis methods, Bayesian approaches have several advantages. Sweeting, Sutton, and Lambert (2004) compare several meta-analysis methods for combining odds ratios (using various classical and Bayesian methods of estimation) on sparse event data, and they find that the Bayesian model using vague priors (which does not require continuity correction factors) performs consistently well irregardless of group size imbalance. This, however, is sometimes outperformed by the Mantel-Haenszel method with a continuity correction. Sutton and Abrams (2001) show that the advantages of Bayesian meta-analysis include full allowance for all parameter uncertainty in the model, the ability to include other pertinent information that would otherwise be excluded, and the ability to extend the models to accommodate more complex situations.

In this chapter, we focus on dealing with rare events in network meta-analysis for count data. Our goal is to compare continuity correction methods to adjust for zero outcomes in a network meta-analysis. We use a Bayesian framework and focus on binary outcomes. In Section 3.2, we overview four types of network patterns and the common outcome measures in network meta-analysis. In addition, we provide brief descriptions of network meta-analysis and the continuity correction. Section 3.3 presents simulation details, including simulation design, performance evaluation,

and simulation results. We discuss the results of simulation studies and give some concluding comments in Section 3.4.

3.2 Methods

3.2.1 Geometry of Network

Because there are more than two treatments in network meta-analysis, it is important to understand the distribution of included studies. A network diagram is very helpful to display the network configuration. It provides a visualized explanation of all direct comparisons between treatments. Figure 3.2.1 is an example of a network diagram, specifically (the configuration of the star geometry).

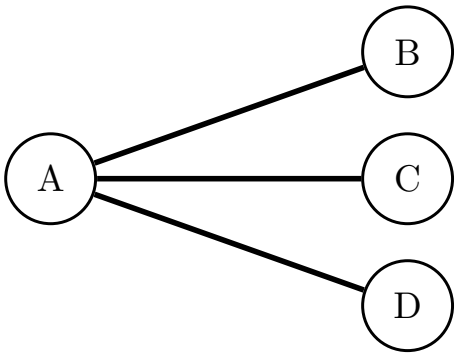


Figure 3.2.1: Star Geometry

The nodes in Figure 3.2.1 represent interventions or treatments, and the lines linking the two interventions represent that these two interventions have a direct comparison, such as A-B, A-C, and A-D. Because B, C, and D link with common comparator A, we can obtain indirect treatment comparisons among them.

In the following simulation study, we consider four network patterns, namely star, loop, one-closed loop, and ladder, given in Figures 3.2.1 to 3.2.4

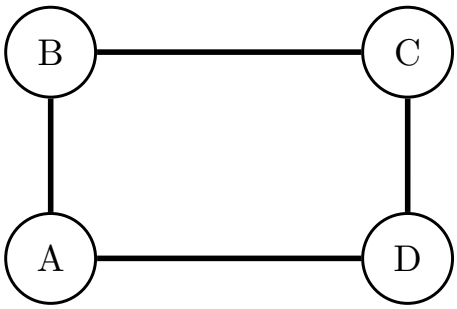


Figure 3.2.2: Loop Geometry

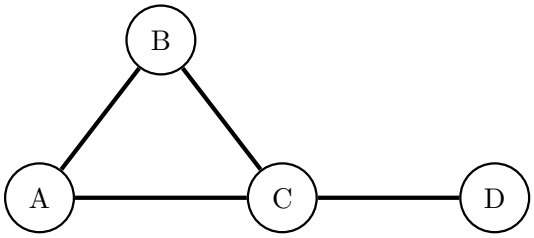


Figure 3.2.3: One-closed Loop Geometry

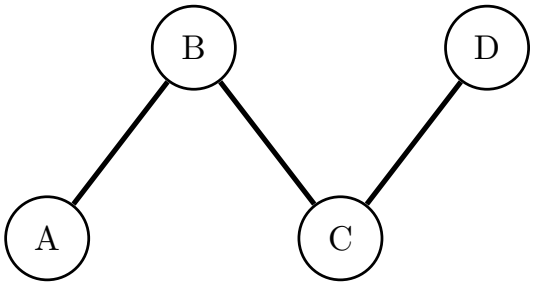


Figure 3.2.4: Ladder Geometry

In each figure, there are four nodes representing four treatments; there are three direct treatment comparisons for both star geometry and ladder geometry and four direct treatment comparisons for loop geometry and one-closed loop geometry.

3.2.2 Effect Measure

For binary outcomes, three outcome measures are commonly used in meta-analysis: risk difference (RD), relative risk (RR), and odds ratio (OR). RD has an easy interpretation and is most useful in decision making; however, the limitation of

RD is that it is not as consistent as the other two measurements across all studies, especially when the events are rare. RD is preferred when the incidence rates are close through all studies. When the incidence rates vary widely across studies, RD is not recommended. The limitations of RD can be overcome by using RR and OR. Both of these provide a more consistent result than RD. Lane (2013) suggests to use OR rather than RR because it is conceptually more appropriate for modeling risks bound in the range [0,1]. It is also more common in statistical analysis. Therefore, in the following simulation study, we use OR as the outcome measurement.

3.2.3 Statistical Details

Following the notation used in Greco et al. (2013), suppose we have N randomized controlled trials making mixed comparisons among K treatments. Define r_{ik} and n_{ik} as the number of events and the total observations on treatment k in the i^{th} trial, respectively. Furthermore, let p_{ik} be the probability of event occurrence. For the binary outcomes, we assume the number of events, r_{ik}, has a binomial distribution,

$$r_{ik} \sim \text{binomial}(p_{ik}, n_{ik}), \quad i = 1, 2, \ldots, N; k = 1, 2, \ldots, K.$$

It is common to use logistic regression to model the probability of event occurrence p_{ik}:

$$\text{logit}(p_{ib}) = \log\left(\frac{p_{ib}}{1-p_{ib}}\right) = \mu_i, \quad i = 1, 2, \ldots, N; k = b = 1, 2, \ldots, K-1,$$

$$\text{logit}(p_{ik}) = \log\left(\frac{p_{ik}}{1-p_{ik}}\right) = \mu_i + \delta_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k. \quad (3.1)$$

μ_i are the trial-specific baselines and represent the log odds of events in the reference treatment ($k = b$), while $\delta_{i,bk}$ are the trial-specific log odds ratios of event occurrence of the treatment group k compared with the reference treatment.

The nature of the effect $\delta_{i,bk}$ depends on the underlying assumptions. Two models that are commonly used in meta-analysis are the fixed effects and random

effects models. The difference between these two models is in the way between-study variations are accounted for. With the fixed effects model, the assumptions are made that each observed individual study has a shared common treatment effect and that differences between studies are caused by chance. Individual studies are simply weighted by their precision. For the fixed effects model, Equation 3.1 can be replaced as follows:

$$\text{logit}(p_{ik}) = \mu_i + d_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k,$$

where μ_i are the trial-specific baselines and $d_{i,bk}$ are fixed log odds ratios of the event occurrence of the treatment group k compared with the reference treatment. In this model, the between study variation is equal to zero.

The random effects model allows for the existence of between-study heterogeneity. In other words, the underlying effect for each study is different. In addition, it is often assumed that these true effects are described by a normal distribution, and our interest is in estimating the mean of this normal distribution. For a random effects model, the trial-specific log odds ratio is assumed:

$$\delta_{i,bk} \sim N(d_{bk}, \sigma^2).$$

Both the fixed and random effects models have limitations. For example, the fixed effects estimator produces confidence intervals with poor coverage when applied to the populations that may not be entirely identical. Random effects, on the other hand, are highly sensitive to the accuracy of the estimate of the between-study variance. Hunter and Schmidt (2000) also show that fixed effects models yield confidence intervals for mean effect sizes that are narrower than their nominal width, therefore overstating the degree of precision in meta-analysis findings. They recommend that random effects methods routinely be employed in meta-analysis in preference to fixed effects methods. In the following simulation study, we use a random effects meta-analysis model. Some typical assumptions of random effects

meta-analysis models are that individual study results are exchangeable and can be described as a sample from a common distribution.

In terms of comparison between treatments, there are direct comparisons and indirect comparisons. Direct comparisons are made between treatments having head-to-head randomized studies. For example, in the star geometry of Figure 3.2.1, A-B, A-C, and A-D directly compare with each other, so d_{AB}, d_{AC}, and d_{AD} can be estimated from these studies. In the language of network meta-analysis, these are called basic parameters of the model. Indirect comparisons are made between treatments in the absence of head-to-head randomized studies but have one common comparator. For example, treatments B, C, and D are linked via a common comparator A, so d_{BC}, d_{BD}, and d_{CD} can be calculated based on the pooled estimates for the basic parameters. Those parameters, like d_{BC}, d_{BD}, and d_{CD}, are called functional parameters.

The key assumption between direct and indirect comparisons is the consistency assumption. It is important that the indirect estimate is unbiased and that there are no discrepancies between the direct and indirect comparisons. For example, in one-closed loop geometry, d_{BC} can be directly estimated from studies B-C, but it also can be indirectly calculated from d_{AB} and d_{AC}. The consistency assumption requires that the following equation be satisfied: $d_{BC} = d_{AC} - d_{AB}$. In general, for the estimate of indirect treatment comparison d_{st}, we have

$$d_{st} = d_{bt} - d_{bs}, \quad b = 1, 2, \ldots, K; s = 2, 3, \ldots, K; t = 3, 4, \ldots, K; s < t.$$

In the Bayesian framework, we assume prior distributions for unknown parameters. It is common to set weakly-informative prior distributions to the basic parameters. Usually $\mu_i, d_{bk} \sim N(0, 10^2)$. In terms of the standard deviation of $\delta_{i,bk}$, because the standard deviation has to be positive, we take the positive part of the normal distribution. That is,

$$\sigma \sim N(0, B^2)^+,$$

where B is a large value so that the prior information for σ is vague. We set $B = 10$ in the following simulation.

3.2.4 Continuity Correction

When zero counts occur in a meta-analysis, a common strategy is to do a continuity correction. One of the most popular methods is to add a small number c to the zeros to make them non-zero. The number c is usually added to both arms from a study where either outcome is zero in order to reduce bias. For example, if we compare treatment b and treatment k and if either r_{ib} or r_{ik} is zero, we would add c to both r_{ib} and r_{ik} and add $2c$ to both n_{ib} and n_{ik}. The most widespread constant that is chosen is $c = 0.5$ because when using the odds ratio as the effect measure, $c = 0.5$ provides the least biased estimator of the true log odds in a single treatment group situation. We also consider $c = 1$ because it is assumed that the number of events has a binomial distribution. Thus, the number of events is more reasonable as an integer.

3.3 Simulation Study

3.3.1 Simulation Design

In the simulation study, we aimed to compare the performance of two continuity correction methods in four types of networks: star geometry, loop geometry, one-closed loop geometry, and ladder geometry. We also investigated how the hierarchical Bayesian model identifies the most effective treatment under different network geometries. We considered four treatments and assumed one of them to be the reference treatment and the others to be competing treatments.

We chose the random effects model in the network meta-analysis because the fixed effects model could be treated as a special case of the random effects model in which the between-study variance is zero. Let us denote the treatments under

investigation by treatment A (T_1), which we considered to be the reference treatment, treatment B (T_2), treatment C (T_3), and treatment D (T_4).

Table 3.3.1 describes the simulation scenarios. Within each network pattern, we let μ_i vary from -3 to -7, where μ_i are the trial-specific baselines. We considered two different sample sizes (n_{ik}): 100 and 200. It was more likely to have zero outcomes in the study where n_{ik} was 100.

In terms of the number of studies, we set the studies for each comparison as equal. For example, for the star geometry, we did five studies for all comparisons $(T_1\&T_2), (T_1\&T_3)$, and $(T_1\&T_4)$, thus the total number of studies for the star geometry was 15. Similarly, the total number of studies for the ladder geometry, the loop geometry, and the one-closed loop were 15, 20, and 20, respectively.

Table 3.3.1: Simulation Scenarios

Parameter	Value	
Network Patterns	Star, Ladder, Loop, One-Closed Loop	
Baseline, μ_i	-3,-4,-4.5,-5,-5.5,-6,-6.4,-6.6,-7	
Total Observations(n_{ik})	100,200	
Number of	Star, Ladder	Loop, One-Closed Loop
Studies	15	20

Table 3.3.2 shows the true means of the odds ratios between treatments for all network patterns. The trial-specific log odds ratios $\delta_{i,bk}$ were generated from a $N(\log(d_{bk}), \sigma^2)$, and we then used the inverse-logit transformation to get the corresponding probability of event occurrence p_{ik}. The number of events, r_{ik}, was then generated from a binomial(p_{ik}, n_{ik}).

Table 3.3.2: True Value of Odds Ratio

Study	Star	Ladder	Loop	One-Closed Loop
b_{21}	1.5	1.5	2	1.5
b_{31}	2	3	3	2
b_{41}	3	9	4	4.5
b_{32}	1.33	1.5	1.5	1.33
b_{42}	2	6	2	3
b_{43}	1.5	3	1.33	2.5

Based on the procedures we described above, we generated the number of events for each of the simulation scenarios, then we fit the hierarchical Bayesian random effects model on each data set and performed statistical inference. Inferences were based on the posterior samples using MCMC. In each simulation, we drew 13000 posterior samples with a burn-in of 3000 samples to remove the impact of the initial values on the posterior distribution. We repeated each scenario 100 times.

3.3.2 Performance Evaluation

The hierarchical Bayesian approach to mixed treatment comparisons provides a straightforward way to calculate the probability that each treatment is best. In each MCMC run, each treatment in the study can be ranked based on its estimated magnitude. Then, the proportion of MCMC iterates in which the treatment k ranks first gives the probability that the specific treatment is best among all competing treatments in the study. Similarly, the other probabilities can be calculated for being second best and the third best, as well as the 95% posterior interval for treatment k's rank.

Salanti et al. (2011) and Groce et al. (2013) propose another way to assess the rank probability. They discuss graphical and numerical summary of cumulative ranking and suggest surface under the cumulative rank curve (SUCRA) to show the

cumulative rank probabilities for each treatment. For each treatment k in the K treatments in the network, $Cum_{k,w}$ is the cumulative probabilities of the w^{th} best for the k treatment. Where $w = 1, \ldots, K$, the SUCRA statistic will be

$$SUCRA_k = \frac{\sum_{w=1}^{K-1} Cum_{k,w}}{K-1}.$$

The SUCRA statistic goes to 1 if the treatment is the best and goes to 0 if it is the worst.

Besides the best rank probabilities of each treatment and the 95% posterior intervals of their ranks, we were also interested in the odds ratio of all treatment comparisons, including direct treatment comparisons and indirect treatment comparisons in each scenario. In addition, the length of the 95% posterior intervals of the odds ratios and coverage probability were also recorded. We also calculated the bias of the estimates, which is the absolute difference between the true and estimated values.

3.3.3 Simulation Results

Figure 3.3.1 shows the zero rates in our simulated data. The true value of the odds ratio between treatments are listed in Table 3.3.2. As the baseline log odds ratio decreased, the number of zero outcomes increased. when the baseline log odds ratio was -7 and the number of total observations was 200, star and one-closed loop geometry provided more than 70% of the zero outcomes, and ladder and loop geometry provided more than 60% of the zero outcomes.

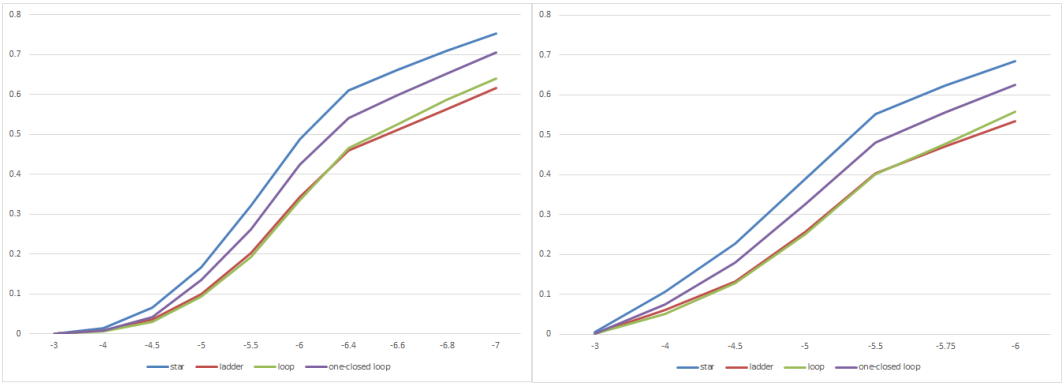


Figure 3.3.1: The left plot is the zero rate when the total observation is 200, the right plot is the zero rate when the total obervation is 100, and the horizontal axis represents the baseline log odds ratio μ.

3.3.3.1 Star Geometry

Table 3.3.3: No Continuity Correction for Star Geometry

(μ_i, n_i)	Para.	Prob.	Length	Bias	(μ_i, n_i)	Prob.	Length	Bias
(-4.5,200)	b_{12}	0.96	4.66	0.70	(-5,200)	0.91	11.47	1.85
	b_{13}	0.97	5.19	0.67		0.97	179.68	46.62
	b_{14}	0.96	6.82	0.74		0.97	11.89	1.57
	b_{23}	0.94	5.52	0.67		0.95	142.04	43.78
	b_{24}	0.94	7.75	0.87		0.93	13.80	1.74
	b_{34}	0.99	5.36	0.58		0.97	9.64	1.27
(-4,100)	b_{12}	0.95	5.81	0.93	(-4.5,100)	0.94	15.92	2.41
	b_{13}	0.97	6.23	0.84		0.97	61.96	9.37
	b_{14}	0.96	9.52	1.26		0.93	14.57	1.95
	b_{23}	0.96	6.86	0.89		0.94	57.43	8.59
	b_{24}	0.94	10.00	1.22		0.93	17.33	2.26
	b_{34}	0.98	7.27	0.88		0.96	11.94	1.53

Table 3.3.3 shows part of the simulation results for the star geometry without any continuity correction. When the total number of observations was $n = 200$, the $\mu \geq -4.5$ results were reasonable. However, as μ decreased, the results became worse, especially for the estimates of b_{13} and b_{23}. When the total number of observations was $n = 100$, $\mu = -4$ still provided reasonable results, but when $\mu = -4.5$, the width of the 95% posterior interval and the bias increased dramatically.

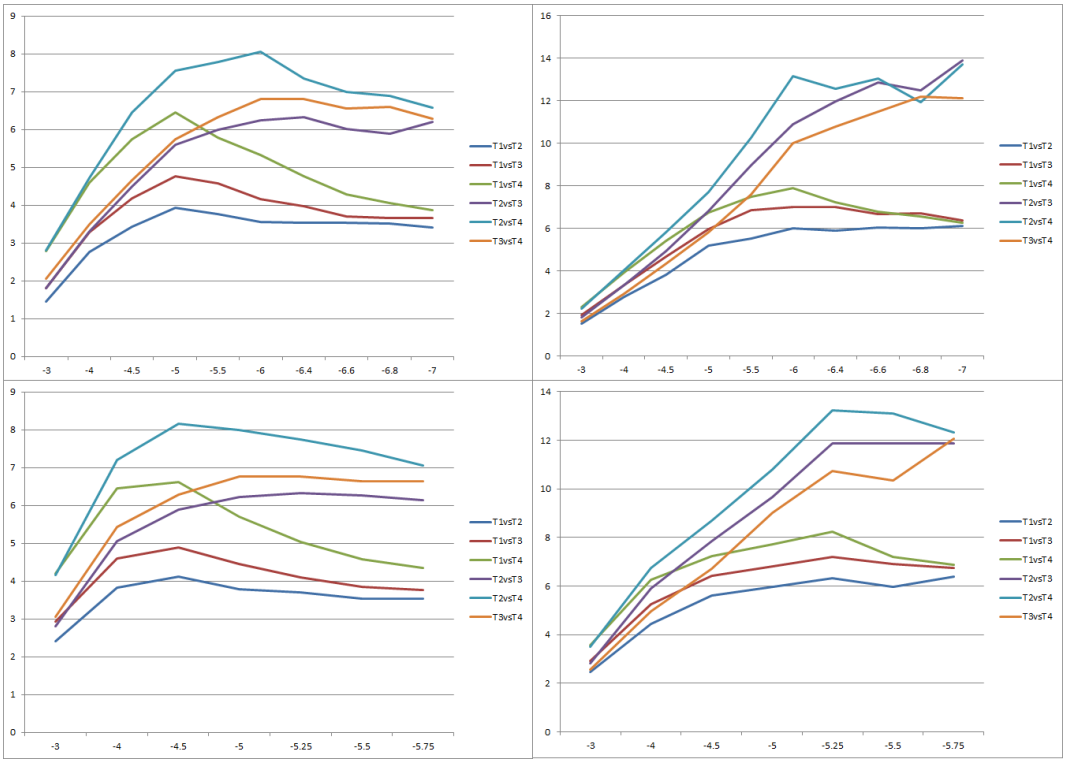


Figure 3.3.2: The left plot is the length of the 95% posterior interval of star geometry for $c = 1$, the right plot is the length of the 95% posterior interval of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

Figures 3.3.2 and 3.3.3 provide the plots of the 95% posterior interval widths and the bias of the estimates for the star geometry for both $c = 0.5$ and $c = 1$. Both length and bias were reduced, compared to the no continuity correction cases. In addition, the studies of indirect comparisons (T_2&T_3), (T_2&T_4), and (T_3&T_4) provided larger 95% posterior interval lengths than those of direct comparison studies

for both types of continuity correction. Across all of the studies, $c = 0.5$ gave a larger length than $c = 1$.

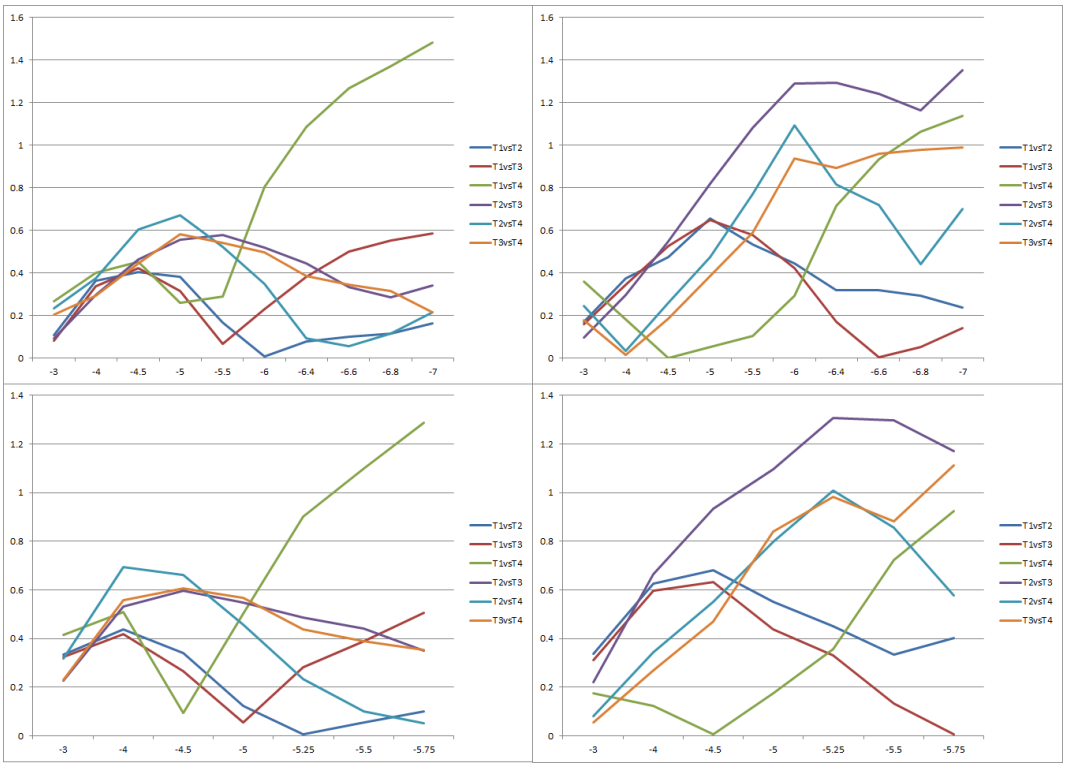


Figure 3.3.3: The left plot is the bias of star geometry for $c = 1$, the right plot is the bias of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

For the case of $c = 1$, the study of $(T_1\&T_4)$ had relatively larger bias than others, and all of the other studies' biases were smaller than 0.8. For the case of $c = 0.5$, the biases were larger than they were for $c = 1$, and indirect comparisons $(T_2\&T_3)$, $(T_2\&T_4)$, and $(T_3\&T_4)$ performed worse than direct comparisons. For coverage probability, adding $c = 1$ to zeros provided low coverage for study $(T_1\&T_4)$ when there were several zeros in the generated data, while $c = 0.5$ performed well in all studies.

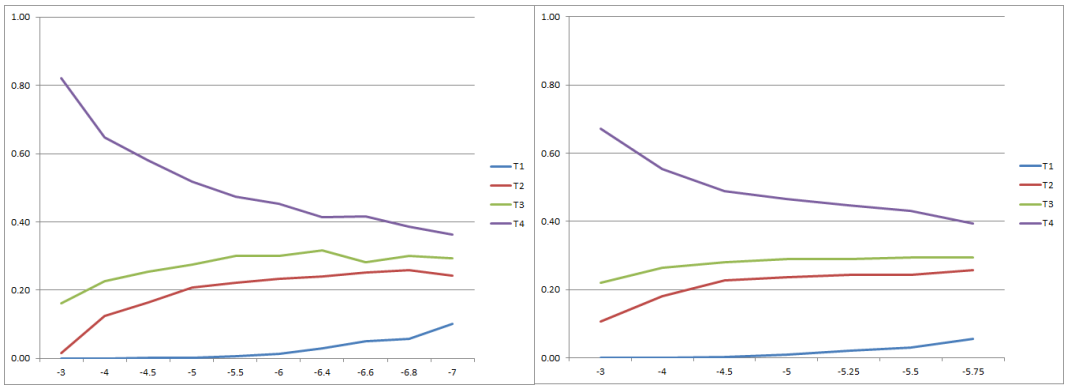


Figure 3.3.4: Best rank probabilities of star geometry for no continuity correction; the horizontal axis represents the baseline log odds ratio μ.

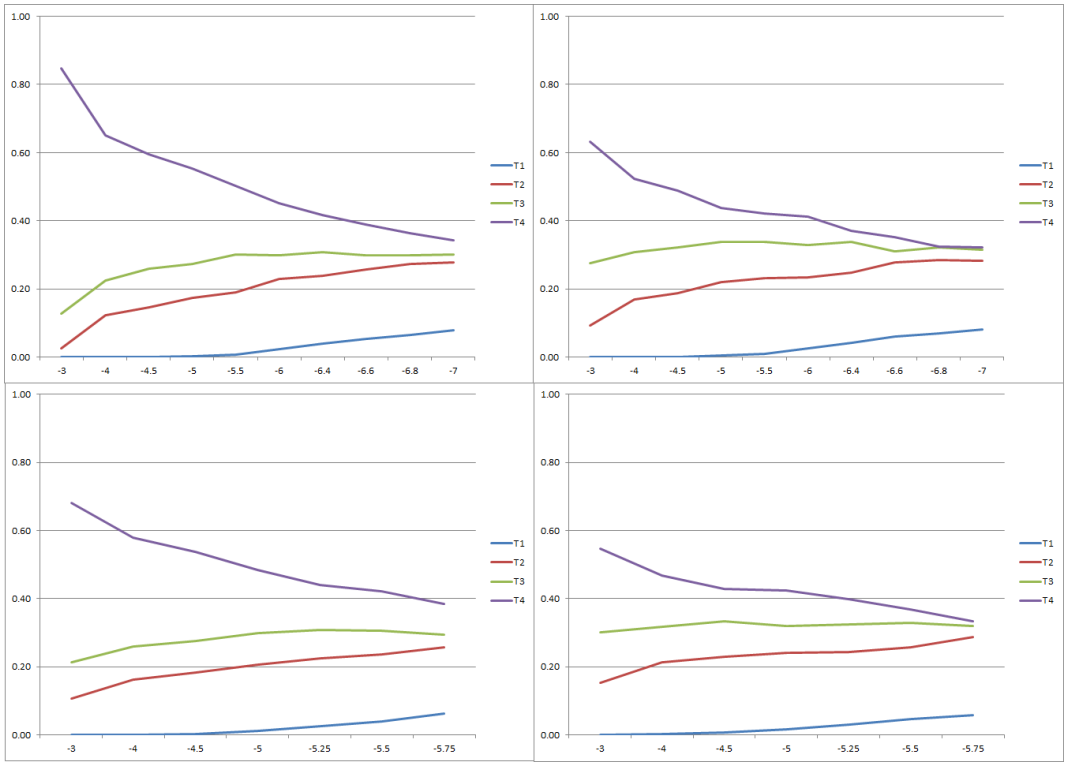


Figure 3.3.5: The left plot is the best rank probabilities of star geometry for $c = 1$, the right plot is the best rank probabilities of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

In terms of the best rank probability, in the star geometry, T_4 is the largest and should have the highest probability. According to Figures 3.3.4 and 3.3.5, as μ decreased, the first rank probability for T_4 decreased, which means that as the num-

ber of zeros increased, it was harder to correctly recognize T_4 as the best treatment. In addition, $c = 1$ with no continuity correction provided similar results, while the case of $c = 0.5$ yielded the lowest best rank probabilities.

3.3.3.2 Ladder Geometry

Table 3.3.4 shows part of the simulation results for ladder geometry without any continuity correction, and we had similar results as the star geometry pattern. For $n = 100$, results became markedly worse when μ was less than -4. For $n = 200$, values of μ less than -4.5 lead to particularly poor results.

Table 3.3.4: No Continuity Correction for Ladder Geometry

(μ_i, n_i)	Para.	Prob.	Length	Bias	(μ_i, n_i)	Prob.	Length	Bias
(-4.5,200)	b_{12}	0.96	4.38	0.75	(-5,200)	0.92	9.96	1.65
	b_{23}	0.97	3.74	0.49		0.98	6.01	0.83
	b_{34}	0.98	3.99	0.29		0.99	5.39	0.51
	b_{13}	0.94	13.51	2.17		0.96	33.21	5.56
	b_{24}	0.96	16.08	2.00		0.97	27.55	4.07
	b_{14}	0.95	49.43	7.93		0.94	127.86	21.14
(-4,100)	b_{12}	0.95	5.72	0.87	(-4.5,100)	0.94	12.98	2.69
	b_{23}	0.98	4.36	0.50		0.97	6.90	0.90
	b_{34}	0.99	4.48	0.38		0.97	6.60	0.75
	b_{13}	0.94	18.18	3.00		0.93	45.86	8.52
	b_{24}	0.97	19.19	2.54		0.95	34.84	5.09
	b_{14}	0.95	67.26	11.17		0.94	188.70	34.84

Figures 3.3.6 and 3.3.7 display the length of the 95% posterior intervals and the bias of estimates for both $c = 1$ and $c = 0.5$.

Figure 3.3.6: The left plot is the length of the 95% posterior interval of ladder geometry for $c = 1$, the right plot is the length of the 95% posterior interval of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

In terms of length, the studies of indirect comparison (T_2&T_3) and (T_2&T_4) provided a wider 95% posterior interval than those of the direct comparison studies for both types of continuity correction. Also, $c = 1$ provided a narrower width than $c = 0.5$. In terms of bias, $c = 1$ performed better than $c = 0.5$, and the direct comparison had smaller biases than the indirect comparisons.

In terms of the best rank probability, in the ladder geometry, T_4 is again largest and should have the highest probability. According to Figures 3.3.8 and 3.3.9, as μ decreased, the first rank probability for T_4 decreased, which means that if we have more zeros in our datasets, it is harder to recognize T_4 as the best treatment. Making a continuity correction increased the best rank probability, especially in the situations where there were more zero outcomes.

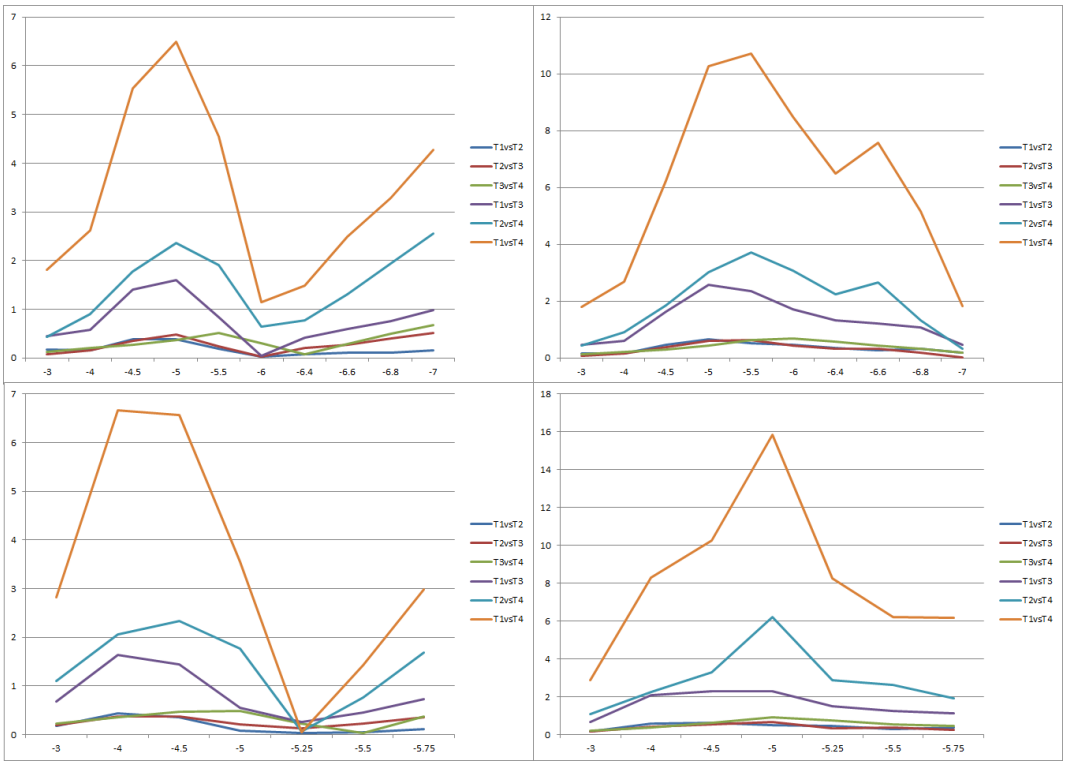


Figure 3.3.7: The left plot is the bias of ladder geometry for $c = 1$, the right plot is the bias of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

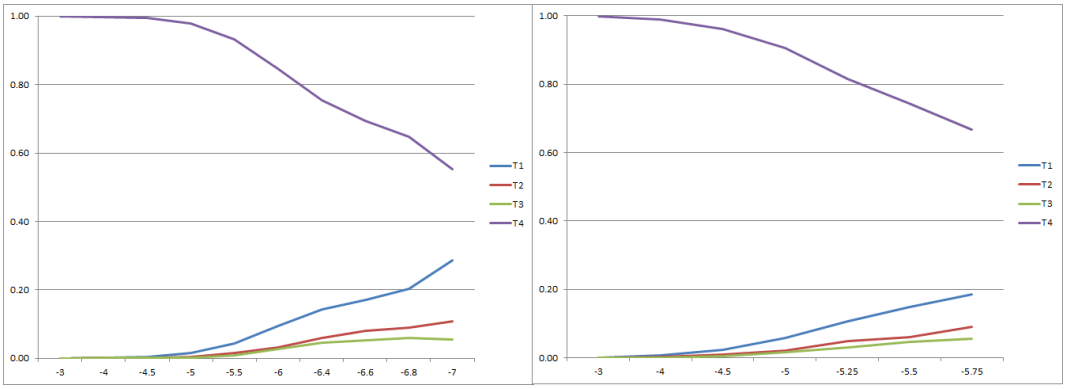


Figure 3.3.8: Best rank probabilities of ladder geometry for no continuity correction; the horizontal axis represents the baseline log odds ratio μ.

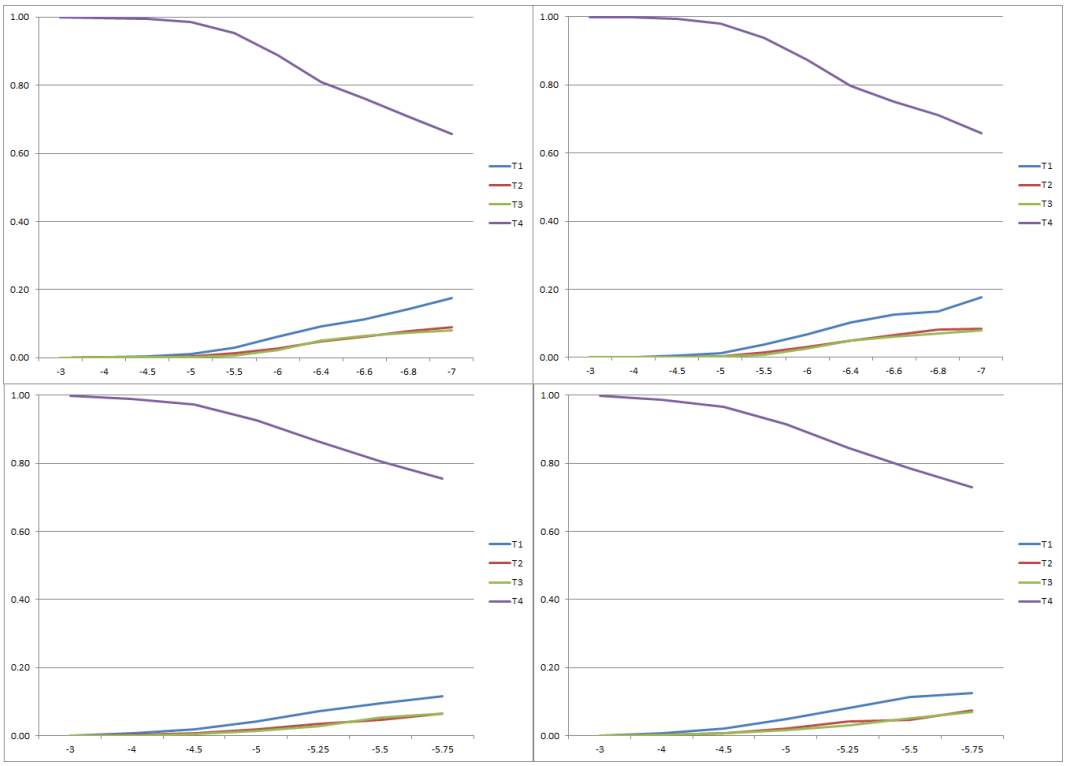


Figure 3.3.9: The left plot is the best rank probabilities of ladder geometry for $c = 1$, the right plot is the best rank probabilities of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

3.3.3.3 Loop Geometry

Table 3.3.5 shows part of the simulation results for the loop geometry without any continuity correction. The coverage probabilities and bias were not as bad in this network pattern. However, the length of the 95% posterior intervals increased considerably when μ decreased from -5 to -5.5 in both of the situations where the sample sizes were $n = 100$ and $n = 200$. It indicated that although the point estimates were closed to the truth, the variation of them were very large.

Table 3.3.5: No Continuity Correction for Loop Geometry

(μ_i, n_i)	Para.	Prob.	Length	Bias	(μ_i, n_i)	Prob.	Length	Bias
(-5,200)	b_{12}	0.96	4.37	0.42	(-5.5,200)	0.92	7.72	1.04
	b_{14}	0.96	8.99	0.96		0.97	15.38	1.94
	b_{23}	0.97	2.68	0.27		0.96	3.91	0.45
	b_{34}	0.96	1.95	0.13		0.97	2.66	0.21
	b_{13}	0.98	7.28	0.80		0.94	12.19	1.55
	b_{24}	0.94	4.45	0.46		0.95	6.49	0.78
(-5,100)	b_{12}	0.94	8.90	-1.11	(-5.5,100)	0.95	21.05	-3.12
	b_{14}	0.97	17.79	2.19		0.95	51.31	8.16
	b_{23}	0.96	4.79	0.60		0.94	9.01	1.19
	b_{34}	0.97	3.20	0.31		0.97	4.83	0.55
	b_{13}	0.95	14.33	1.83		0.94	35.75	5.41
	b_{24}	0.94	8.23	1.07		0.94	15.77	2.18

Figures 3.3.10 and 3.3.11 show the simulation results with the continuity correction. Both the length of the 95% posterior intervals and the bias were reduced, compared to the results without the continuity correction. Generally speaking, $c = 1$ performed better in the length of the 95% posterior interval, while $c = 0.5$ provided less biased estimates. The studies of (T_1&T_4), (T_1&T_3), and (T_2&T_4) had wider 95% posterior intervals than the others. In terms of bias, (T_1&T_4) and (T_1&T_3) performed worse than the others. As μ decreased, the coverage probabilities decreased considerably for both continuity correction methods. The correction $c = 1$ provided even worse coverage probabilities than $c = 0.5$, especially when μ was less than -5.5.

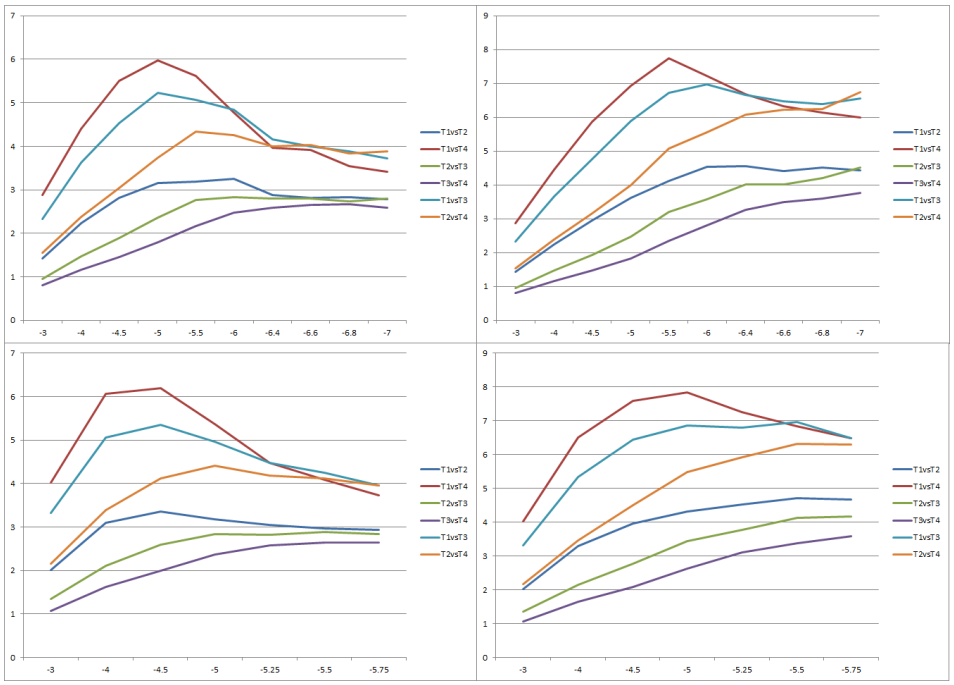


Figure 3.3.10: The left plot is the length of the 95% posterior interval of loop geometry for $c = 1$, the right plot is the length of the 95% posterior interval of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

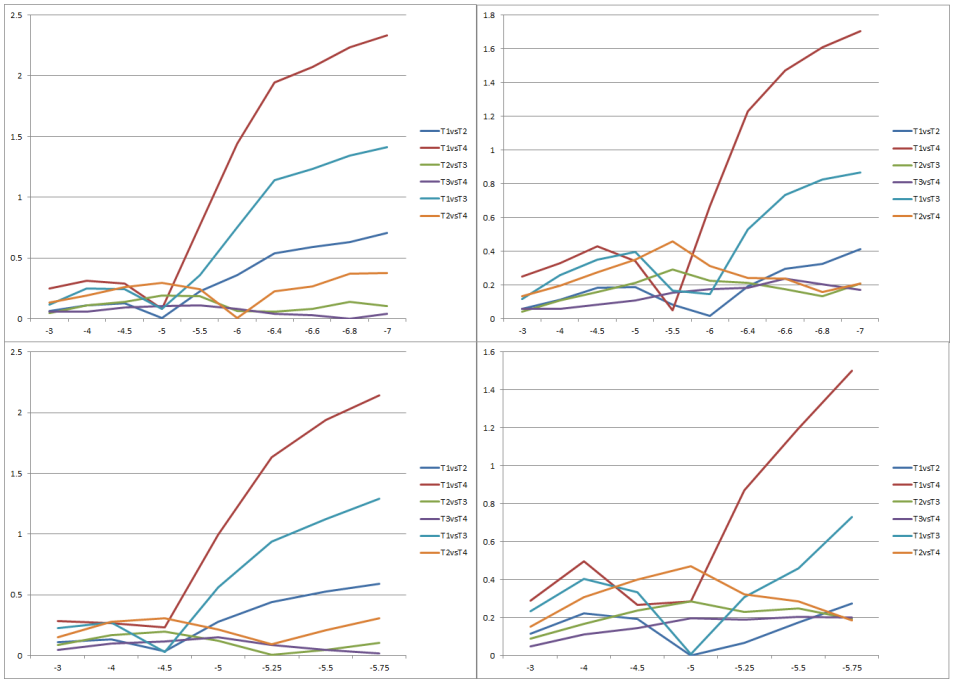


Figure 3.3.11: The left plot is the bias of loop geometry for $c = 1$, the right plot is the bias of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

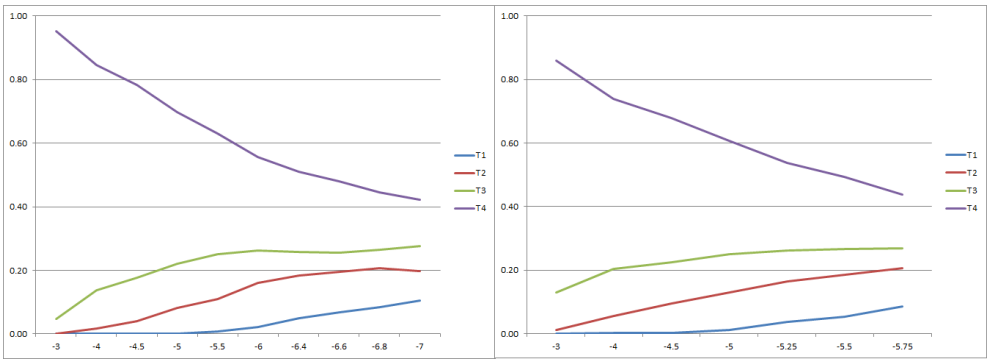


Figure 3.3.12: Best rank probabilities of loop geometry for no continuity correction; the horizontal axis represents the baseline log odds ratio μ.

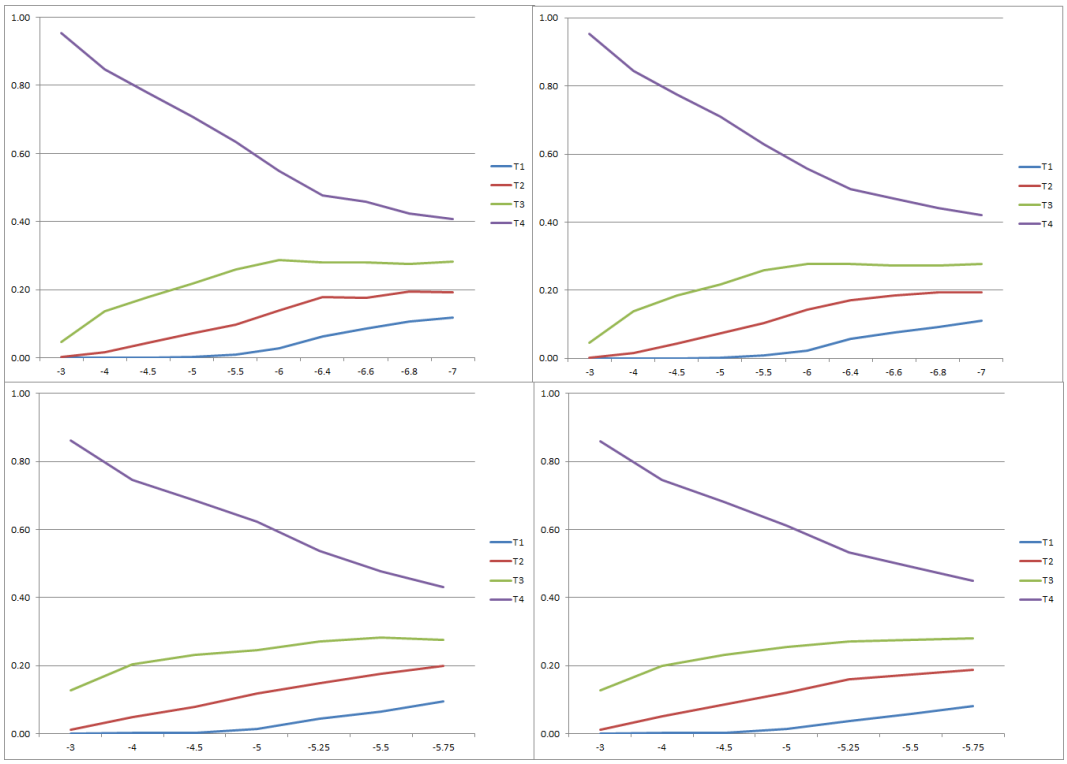


Figure 3.3.13: The left plot is the best rank probabilities of loop geometry for $c = 1$, the right plot is the best rank probabilities of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

In terms of the best rank probability, Figures 3.3.12 and 3.3.13 show that continuity correction did not provide better results than those with no continuity correction, no matter whether we added 0.5 or 1 to zero outcomes.

3.3.3.4 One Closed Loop Geometry

For the one-closed loop geometry pattern, Table 3.3.6 shows part of the results without the continuity correction. When the sample size was $n = 200$ and μ decreased from -6 to -6.4, the bias did not change much; however, the length of the 95% posterior interval increased considerably. When the sample size was $n = 100$ and μ decreased from -5 to -5.5, both the width of the 95% posterior intervals and the bias increased substantially.

Table 3.3.6: No Continuity Correction for One-closed Loop Geometry

(μ_i, n_i)	Para.	Prob.	Length	Bias	(μ_i, n_i)	Prob.	Length	Bias
(-6,200)	b_{12}	0.93	9.65	1.28	(-6.4,200)	0.92	17.44	0.22
	b_{13}	0.93	10.20	1.06		0.95	15.57	0.94
	b_{23}	0.96	7.00	1.00		0.93	16.38	0.33
	b_{34}	0.95	13.12	1.64		0.94	41.61	0.86
	b_{14}	0.92	60.72	8.71		0.95	153.72	4.21
	b_{24}	0.95	61.01	5.56		0.96	153.85	1.73
(-5,100)	b_{12}	0.94	7.12	0.93	(-5.5,100)	0.91	15.62	2.08
	b_{13}	0.94	7.91	0.84		0.93	18.21	2.41
	b_{23}	0.93	4.55	0.49		0.94	22.55	6.63
	b_{34}	0.97	9.71	1.25		0.96	17.39	2.31
	b_{14}	0.95	42.42	6.46		0.95	125.53	21.74
	b_{24}	0.99	42.83	2.81		0.94	125.78	17.06

Figures 3.3.14 and 3.3.15 show the simulation results when incorporating the continuity correction. Generally speaking, $c = 0.5$ performed better in the length of the 95% posterior intervals, while $c = 1$ provided less biased estimates. We found that the estimates of indirect comparisons (T_1&T_4) and (T_2&T_4) had larger 95% posterior interval lengths than those of the direct comparisons. In terms of bias,

when $c = 0.5$, the study (T_1&T_4) had the worst performance. When $c = 1$, (T_1&T_4) and (T_2&T_4) had relatively larger biases than others. For the coverage probabilities, $c = 1$ gave low coverage probabilities for (T_1&T_4) when μ was small.

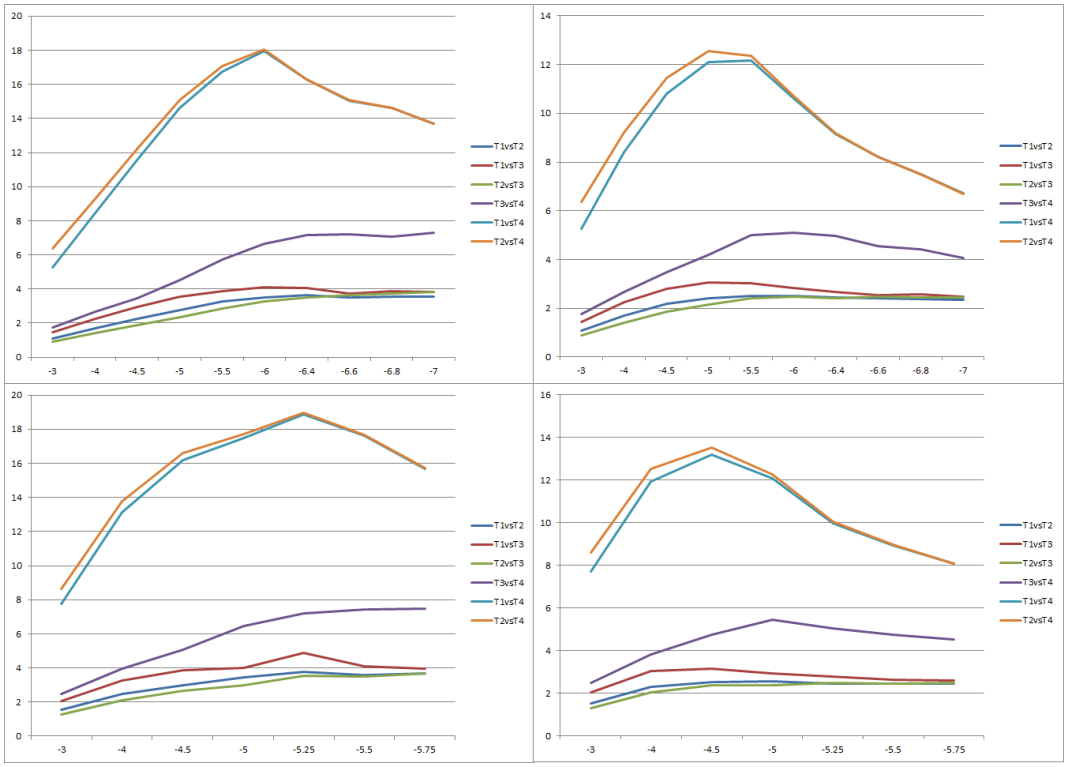


Figure 3.3.14: The left plot is the length of the 95% posterior interval of one-closed loop geometry for $c = 1$, the right plot is the length of the 95% posterior interval of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

In terms of the best rank probability, T_4 was again the best among all of the treatments. Figures 3.3.16 and 3.3.17 show that both with and without the continuity correction, it provided almost the same results for the best rank probability. Incorporating the continuity correction increased the best rank probability by a modest amount in the situations where μ was small.

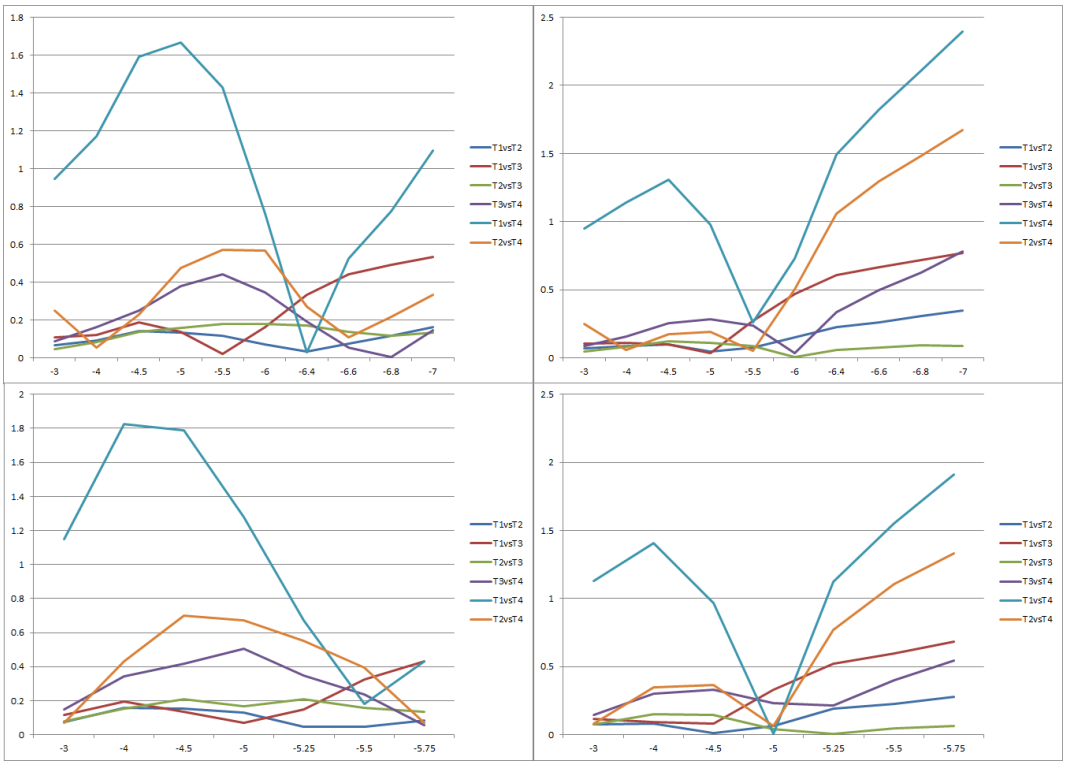


Figure 3.3.15: The left plot is the bias of one-closed loop geometry for $c = 1$, the right plot is the bias of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

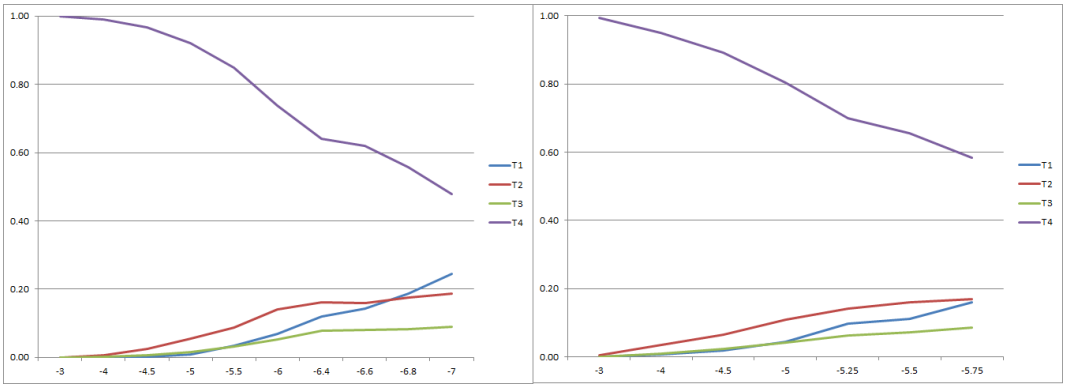


Figure 3.3.16: Best rank probabilities of one-closed loop geometry for no continuity correction; the horizontal axis represents the baseline log odds ratio μ.

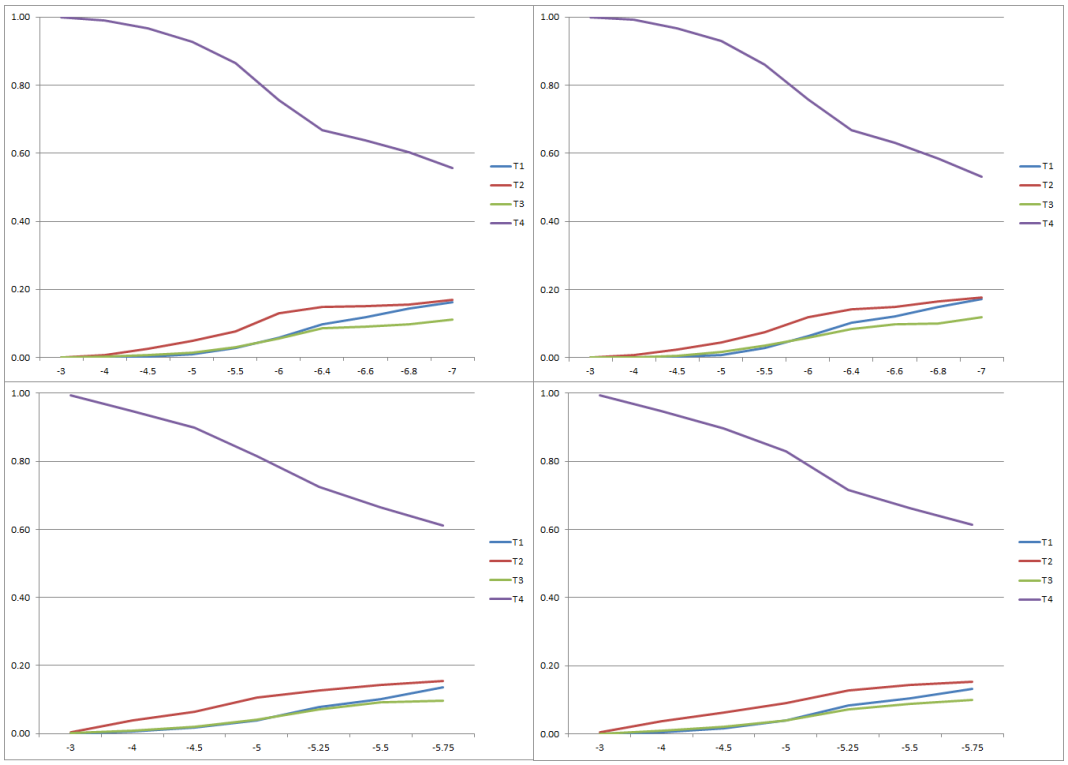


Figure 3.3.17: The left plot is the best rank probabilities of one-closed loop geometry for $c = 1$, the right plot is the best rank probabilities of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log odds ratio μ.

3.4 Conclusion

In this chapter, we have extended previous work on standard meta-analysis with rare events to network meta-analysis with rare events. The Bayesian approach was used to get the estimates of odds ratios between treatments because of the advantages of Bayesian methods in dealing with zero outcomes (Sutton and Abrams 2001; Sweeting et al. 2004). In addition, we implemented a continuity correction to further reduce bias. Through simulation studies, we have shown that with more zeros in the outcome, the bias of the odds ratio estimates increased regardless of including the continuity correction or not. In general, indirect comparisons (e.g., T1 vs. T3 and T2 vs. T4 in ladder geometry) provided more biased estimates and wider 95% posterior intervals than direct comparisons (e.g., T1 vs. T2 and T2 vs. T3 in ladder geometry).

Making continuity corrections did reduce the bias of the estimated odds ratio compared to no continuity correction. When comparing the performance of adding 0.5 with adding 1, the amount of improvement varied in different network patterns. Adding 1 to zero outcomes provided less biased estimates of the odds ratio in star, ladder, and one-closed loop geometry and narrower 95% posterior intervals in star, ladder, and loop geometry. Adding 0.5 showed advantages in reducing bias for loop geometry and a narrower 95% posterior interval in one-closed loop geometry.

When it comes to the best rank probability, the continuity correction helped detect the most effective treatment, regardless of the network patterns. The network meta-analysis model was able to identify the most effective treatment, especially when there was a superior treatment among the competing treatments.

CHAPTER FOUR

Bayesian Network Meta-analysis for Poisson Outcomes with Rare Events

4.1 Introduction

In Chapter Three, we extended standard meta-analysis with rare events to network meta-analysis with rare events. Bayesian approaches were used as our method of inference because the Bayesian model using vague priors outperformed frequentist meta-analysis methods. Zero events were adjusted with continuity corrections, which can further reduce the estimates biases. Also in Chapter Three, we used binomial likelihood and logit transformation to conduct our analysis, and the conclusions were based on the estimates of odds ratios. In this chapter, we consider Poisson data.

In order to model count data, the Poisson distribution is widely used in clinical trials and epidemiologic studies. In fact, the Poisson distribution can be obtained by making use of the binomial limit. In the binomial distribution, if the sample size goes to infinity and the event probability goes to zero in such a way that the product of the sample size and the event probability goes to a certain rate, the binomial distribution approaches the Poisson distribution.

Poisson models have already been considered in meta-analysis. Crowther et al. (2012) use Poisson regression models in meta-analysis for individual patient survival data in both classical and Bayesian framework. Compared to the Cox model, the Poisson model is computationally efficient and highly flexible. Also, Bayesian methods perform better in estimating the heterogeneity in the treatment effect.

Poisson models also show some advantages in dealing with rare events. Böhning, Mylona, and Kimber (2015) show that the mixed Poisson regression can handle the occurrence of zero events in clinical trials, even if zeros occur in both arms. In the meta-analysis, Bagos and Nikolopoulos (2009) show that in cases of rare events

(event rate $\leq$5%) and when the number of studies is small ($\leq$10), the Poisson regression method is uniformly more powerful. Spittal, Pirkis, and Gurrin (2015) find that Poisson regression with random intervention effects is a useful method for conducting a meta-analysis of incidence rate data, especially when the data contains structural zeros.

We extend the Poisson model in standard meta-analysis with rare events to the network meta-analysis with rare events. We show the advantages of making a continuity correction for zero events, and we also compare the different continuity correction methods to adjust for zero outcomes in different geometry patterns. Bayesian approach is used to obtain the estimates of parameters of interest. In Section 4.2, we discuss the methodology of network meta-analysis and mixed models. Section 4.3 presents simulation studies, including simulation design, performance evaluation, and simulation results. We discuss the results of the simulation studies and give some concluding comments in Section 4.4.

4.2 Methods

For the treatment network patterns, we consider the same types of network as those considered in the binary outcome chapter: star, loop, one-closed loop, and ladder geometry. Risk ratio is used frequently. In the Poisson model, the risk ratio is the ratio of the event occurring rate in the treatment group to the event occurring rate in the reference group. It measures the relative effect of the outcome in the treatment group to that in the control group. To deal with zero outcomes in the Poisson setting, the same strategy as in the binary cases is performed: we add a small number c to both arms from a study where either outcome is zero. The choice of the small number c is either 0.5 or 1.

Following the notion of the binary case, we have N randomized controlled trials making mixed comparisons among K treatments. Define r_{ik} and n_{ik} as the number

of events and the total person-time on treatment k in the i^{th} trial, respectively. Furthermore, let λ_{ik} be the rate of event occurrence. For the Poisson outcomes, we assume the number of events r_{ik} has a Poisson distribution,

$$r_{ik} \sim \text{Poisson}(n_{ik}\lambda_{ik}), \quad i = 1, 2, \ldots, N; k = 1, 2, \ldots, K.$$

In the Poisson model, it is common to model the rate of event occurrence λ_{ik} in log scale,

$$\log(\lambda_{ib}) = \mu_i, \quad i = 1, 2, \ldots, N; k = b = 1, 2, \ldots, K,$$

$$\log(\lambda_{ik}) = \mu_i + \delta_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k, \tag{4.1}$$

where μ_i are the trial-specific baselines and represent the log rate in the reference treatment ($k = b$). Also, $\delta_{i,bk}$ are the trial-specific log risk ratios of the treatment group k compared with the reference treatment.

The nature of the effect $\delta_{i,bk}$ depends on the underlying assumptions. Two models that are commonly used in meta-analysis are the fixed effects and random effects models. The difference between these two models is the way the between-study variation is accounted for. For the fixed effects model, Equation 4.1 can be replaced as follows:

$$\log(\lambda_{ik}) = \mu_i + d_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k,$$

where μ_i are the trial-specific baselines and $d_{i,bk}$ are the fixed log risk ratios of event occurrence of the treatment group k compared with the reference treatment. In this model, the between-study variation is equal to zero.

The random effects model allows for the existence of between-study heterogeneity. In other words, the underlying effect for each study is different. In addition, it is often assumed that these true effects follow a normal distribution, and our interest is in estimating the mean of this normal distribution. For a random effects model, the trial-specific log risk ratio is assumed $\delta_{i,bk}$,

$$\delta_{i,bk} \sim N(d_{bk}, \sigma^2).$$

As we discussed in Chapter Three, the random effects model is recommended, and the assumptions of the random effects meta-analysis model is that the individual study results are exchangeable and can be described as a sample from a common distribution.

In terms of comparison between treatments, there are direct comparisons and indirect comparisons. Similar to the binomial settings, in the network meta-analysis for Poisson counts, there are basic parameters and functional parameters. Basic parameters of the model can be estimated from direct comparisons, like d_{AB}, d_{AC}, and d_{AD} in the star geometry in Figure 3.2.1. Functional parameters, which are related to indirect comparisons, can be calculated based on the pooled estimates for basic parameters.

The crucial assumption between direct and indirect comparisons is the consistency assumption. Under this assumption, the indirect estimate is unbiased, and there is no discrepancy between the direct and indirect comparisons. In general, for the estimate of the indirect treatment comparison d_{st}, we have

$$d_{st} = d_{bt} - d_{bs}, \quad b = 1, 2, \ldots, K; s = 2, 3, \ldots, K; t = 3, 4, \ldots, K; s < t.$$

In the Bayesian framework, we assume prior distributions for unknown parameters. It is common to set weakly-informative prior distributions for the basic parameters. Usually, $\mu_i, d_{bk} \sim N(0, 10^2)$. Because the standard deviation of $\delta_{i,bk}$ has to be positive, we take the positive part of the normal distribution. That is,

$$\sigma \sim N(0, B^2)^+,$$

where B is a large enough value so that the prior information for σ is vague. We set $B = 10$ in the following simulation.

4.3 Simulation Study

4.3.1 Simulation Design

In the simulation study, our goal was to compare the performance of two continuity correction methods in four types of networks: star geometry, loop geometry, one-closed loop geometry, and ladder geometry. We also investigated how the hierarchical Bayesian model identifies the most effective treatment under different network geometries. We considered four treatments and assumed one of them to be the reference treatment and the others to be competing treatments.

We chose the random effects model in the network meta-analysis because the fixed effects model could be treated as a special case of the random effects model, in which the between-study variance was zero. We denote the treatments under investigation by treatment A (T_1), which we considered to be the reference treatment, treatment B (T_2), treatment C (T_3), and treatment D (T_4).

Table 4.3.1 describes the simulation scenarios. Within each network pattern, we let μ_i vary from -3 to -6, where μ_i were the trial-specific baselines. We varied μ_i to control the number of zeros in the outcomes; the smaller μ_i was, the higher the probability of zeros. In this simulation, we fixed the sample size at $n_{ik} = 100$. We set the number of studies for each comparison equal. For example, for the star geometry, we had five studies for all comparisons $(T_1\&T_2), (T_1\&T_3)$, and $(T_1\&T_4)$. Then, the total number of studies was 15. Similarly, the total number of studies for the ladder geometry, loop geometry, and one-closed loop geometry were 15, 20, and 20, respectively.

Table 4.3.1: Simulation Scenarios

Parameter	Value	
Network Patterns	Star, Ladder, Loop, One-Closed Loop	
Baseline, μ_i	-3,-3.5,-4,-4.5,-5,-5.5,-6	
Total Observations(n_{ik})	100	
Number of	Star, Ladder	Loop, One-Closed Loop
Studies	15	20

Table 4.3.2 shows the true mean of the risk ratio between treatments for all network patterns. As long as we had b_{bk}, the trial-specific log risk ratio $\delta_{i,bk}$ could be generated from $N(\log(d_{bk}), \sigma^2)$; we then used an exponential transformation to get the corresponding rate of event occurrence λ_{ik}. Finally, the number of events (r_{ik}) could be generated from Poisson$(\lambda_{ik} n_{ik})$.

Table 4.3.2: True Value of Risk Ratio

Study	Star	Ladder	Loop	One-Closed Loop
b_{21}	1.5	1.5	2	1.5
b_{31}	2	3	3	2
b_{41}	3	9	4	4.5
b_{32}	1.33	1.5	1.5	1.33
b_{42}	2	6	2	3
b_{43}	1.5	3	1.33	2.5

Based on the procedures we described above, we generated the number of events for each of the simulation scenarios. Then, we fit the hierarchical Bayesian random effects model to each data set and performed statistical inference. The inference was based on the posterior samples by MCMC. In each simulation, we drew 13000 posterior samples with a burn-in of 3000 samples to remove the impact

of initial values on the posterior distribution. We repeated this 100 times for each scenario.

4.3.2 Performance Evaluation

The hierarchical Bayesian approach to mixed treatment comparisons provides a straightforward way to calculate the probability of each treatment being the best. In each MCMC run, every treatment in the study could be ranked based on its estimated magnitude. Then, the proportion of MCMC iterates in which the treatment k ranks first gives the probability that the specific treatment was best among all competing treatments in the study. Similarly, the 95% posterior interval of the treatment k's rank can also be calculated.

Besides the best rank probabilities of each treatment and the 95% posterior interval of its rank, we were also interested in the risk ratio of all treatment comparisons, including direct treatment comparisons and indirect treatment comparisons in each scenario. In addition, the length of the 95% posterior intervals of the risk ratio and its coverage probability were also recorded. We also calculated the bias of the estimates, which is the absolute difference between the true and estimated values.

4.3.3 Simulation Results

Figure 4.3.1 shows the zero rates in our simulated data. The true values of the risk ratio between treatments are listed in Table 4.3.2. As the baseline log risk ratio decreased, the number of zero outcomes increased. In the extreme case when the baseline log risk ratio was -6, star geometry almost provided 70% of the zero outcomes, one-closed loop geometry provided more than 60% of the zero outcomes, and ladder and loop geometry provided around 55% of the zero outcomes.

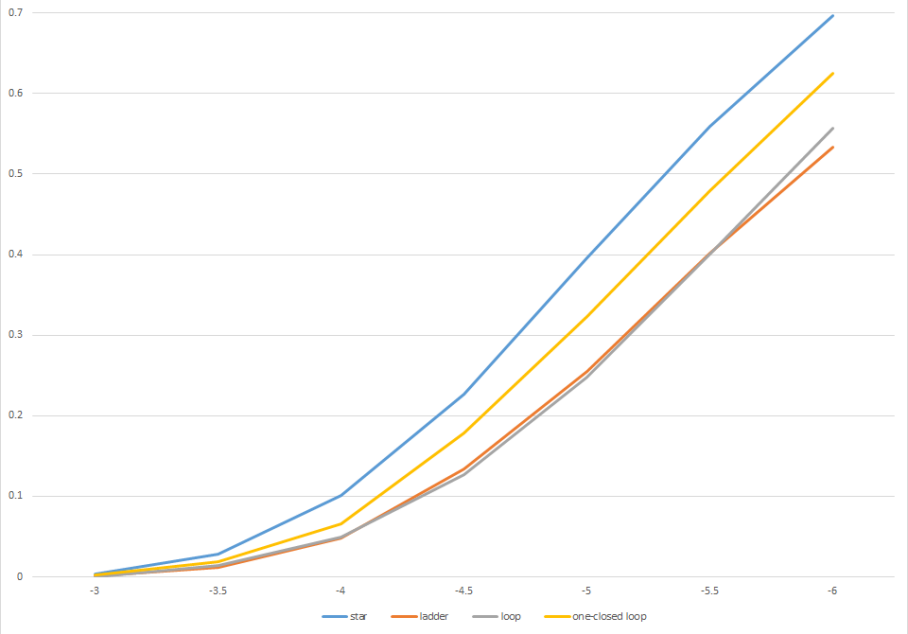


Figure 4.3.1: The zero rate in the simulated data; the horizontal axis represents the baseline log risk ratio μ.

4.3.3.1 Star Geometry

Table 4.3.3 shows some of the simulation results for the star geometry without any continuity correction. In our simulation study, the total number of observations was $n = 100$, $\mu \geq -4$, and the results were reasonable. However, as μ decreased, all of the estimates became worse.

Table 4.3.3: No Continuity Correction for Star Geometry

μ_i	Para.	Prob.	Length	Bias	μ_i	Prob.	Length	Bias
	b_{12}	0.99	5.72	0.66		0.97	166.54	93.32
	b_{13}	0.98	8.93	1.30		0.97	23.13	3.55
	b_{14}	0.93	13.25	2.14		0.94	781.61	352.27
-4	b_{23}	0.97	10.10	1.52	-4.5	0.97	29.68	4.66
	b_{24}	0.97	15.47	2.43		0.94	1280.10	631.20
	b_{34}	0.95	9.93	1.43		0.95	454.53	189.06

Figures 4.3.2 and 4.3.3 provide the plots of the 95% posterior interval widths and the bias of the estimates for the star geometry for both $c = 0.5$ and $c = 1$. Both length and bias were reduced compared to the no continuity correction cases. In addition, the studies of indirect comparisons (T_2&T_3), (T_2&T_4), and (T_3&T_4) provided larger 95% posterior interval lengths than those of the direct comparison studies for both types of continuity correction.

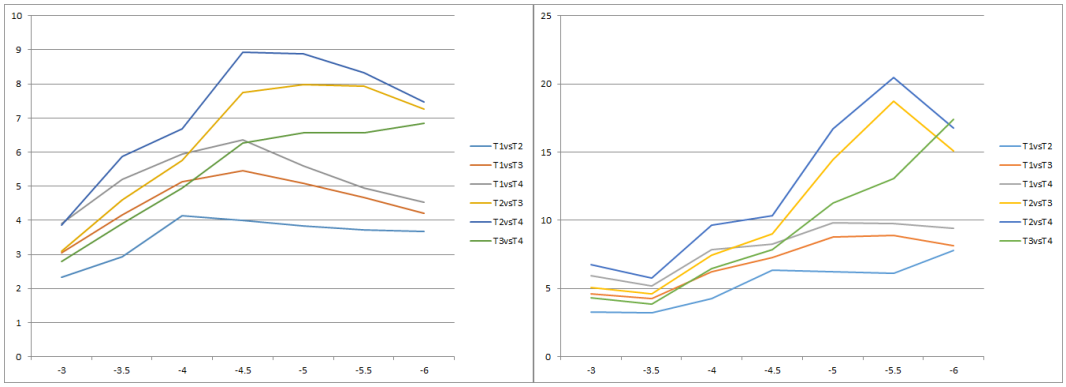


Figure 4.3.2: The left plot is the length of the 95% posterior interval of star geometry for $c = 1$, the right plot is the length of the 95% posterior interval of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

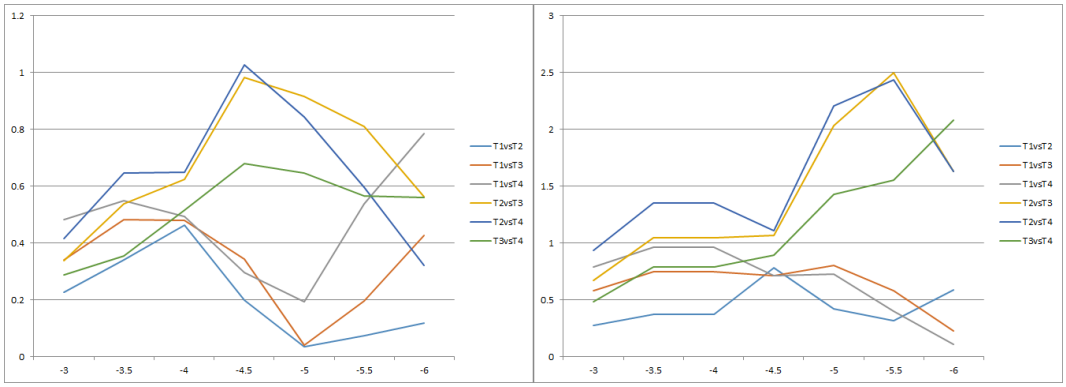


Figure 4.3.3: The left plot is the bias of star geometry for $c = 1$, the right plot is the bias of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

Across all of the studies, $c = 0.5$ gave a wider interval than $c = 1$. For the case of $c = 1$, biases were smaller than 0.8 for almost all of the studies. For the case

of $c = 0.5$, the biases were larger than they were for $c = 1$. and indirect comparisons $(T_2\&T_3)$, $(T_2\&T_4)$, and $(T_3\&T_4)$ performed worse than direct comparisons. For coverage probability, adding $c = 1$ and adding $c = 0.5$ to zeros both provided good results.

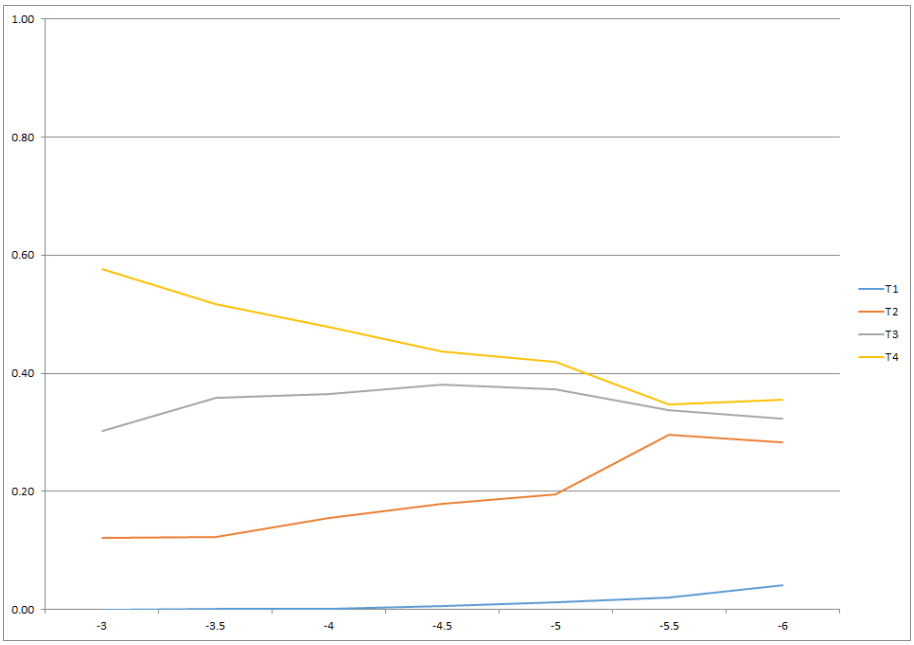


Figure 4.3.4: Best rank probabilities of star geometry for no continuity correction; the horizontal axis represents the baseline log risk ratio μ.

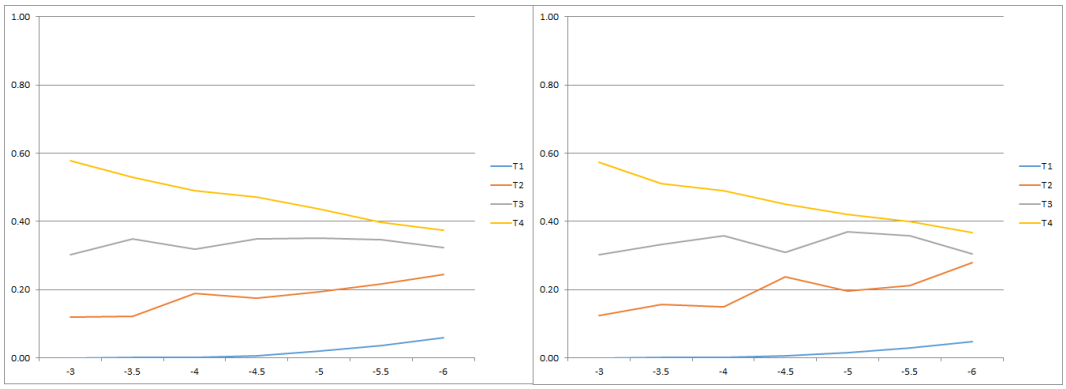


Figure 4.3.5: The left plot is the best rank probabilities of star geometry for $c = 1$, the right plot is the best rank probabilities of star geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

In terms of the best rank probability, in the star geometry, T_4 should be the best among all the treatments. According to Figures 4.3.4 and 4.3.5, as μ decreased, the first rank probability for T_4 decreased, which means that as the number of zeros increased, it was harder to correctly recognize T_4 as the best treatment. In addition, for continuity correction, neither $c = 1$ nor $c = 0.5$ showed any advantages in the best rank probability.

4.3.3.2 Ladder Geometry

Table 4.3.4 shows some of the simulation results for ladder geometry without any continuity correction. We received similar results as the star geometry pattern. For $n = 100$, the results were markedly worse when μ was less than -4.

Table 4.3.4: No Continuity Correction for Ladder Geometry

μ_i	Para.	Prob.	Length	Bias	μ_i	Prob.	Length	Bias
	b_{12}	0.97	2.80	0.19		0.98	21.93	3.72
	b_{13}	0.92	3.19	0.34		0.97	4.26	0.37
	b_{14}	0.98	3.46	0.33		0.94	4.86	0.50
-3.5	b_{23}	0.94	8.61	0.95	-4	0.97	33.15	5.33
	b_{24}	0.97	13.50	1.71		0.98	19.69	2.38
	b_{34}	0.97	32.72	4.26		0.98	127.29	20.95

Figures 4.3.6 and 4.3.7 display the length of the 95% posterior intervals and the bias of estimates for both $c = 1$ and $c = 0.5$. In terms of length, the studies of indirect comparison (T_1&T_4) provided a wider 95% posterior interval than those of the direct comparison studies for both types of continuity correction. Also, $c = 1$ provided a narrower width than $c = 0.5$. In terms of bias, $c = 1$ performed better than $c = 0.5$, and the direct comparison had smaller biases than the indirect comparisons.

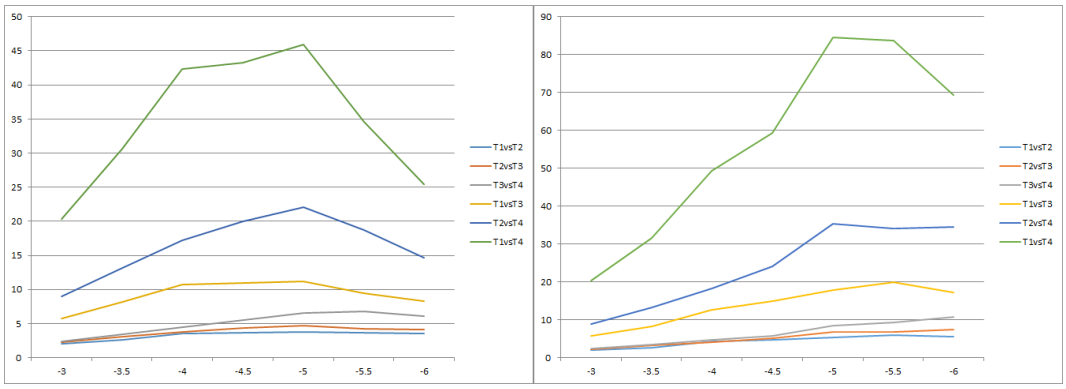


Figure 4.3.6: The left plot is the length of the 95% posterior interval of ladder geometry for $c = 1$, the right plot is the length of the 95% posterior interval of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

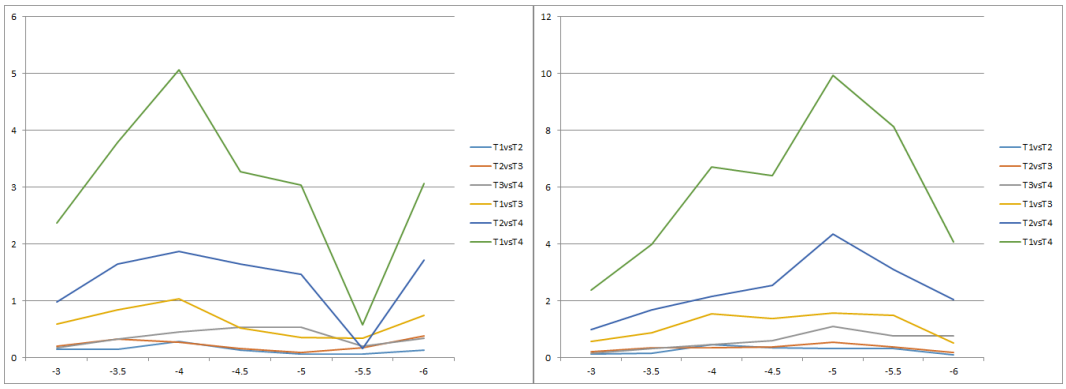


Figure 4.3.7: The left plot is the bias of ladder geometry for $c = 1$, the right plot is the bias of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

In terms of the best rank probability, in the ladder geometry, T_4 should be the best among all of the treatments. According to Figures 4.3.8 and 4.3.9, as μ decreased, the first rank probability for T_4 decreased, which means that if we had more zeros in our datasets, it was harder to recognize T_4 as the best treatment among them. Making a continuity correction increased the best rank probability, especially in the situations where there were more zero outcomes.

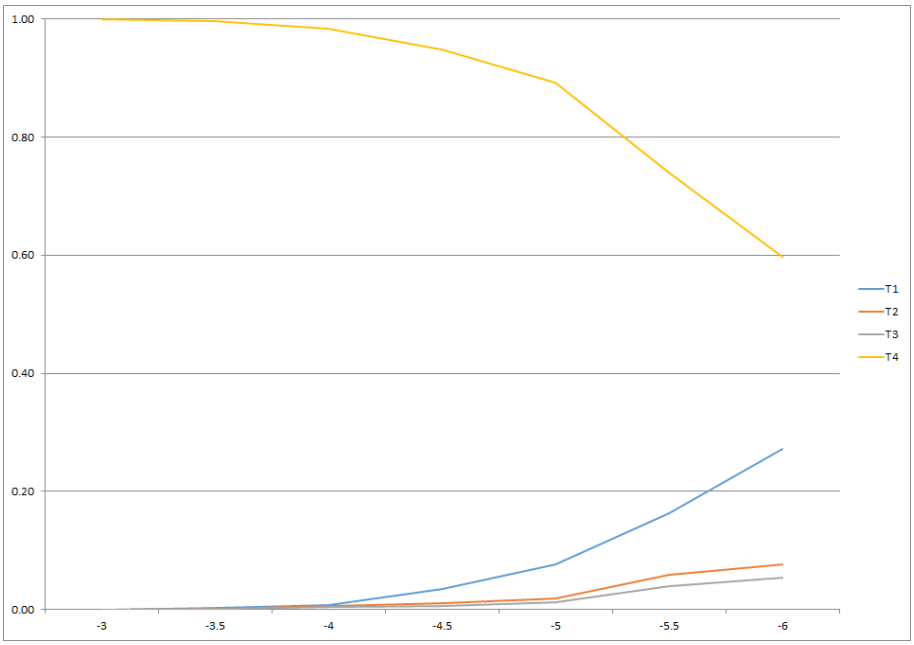


Figure 4.3.8: Best rank probabilities of ladder geometry for no continuity correction; the horizontal axis represents the baseline log risk ratio μ.

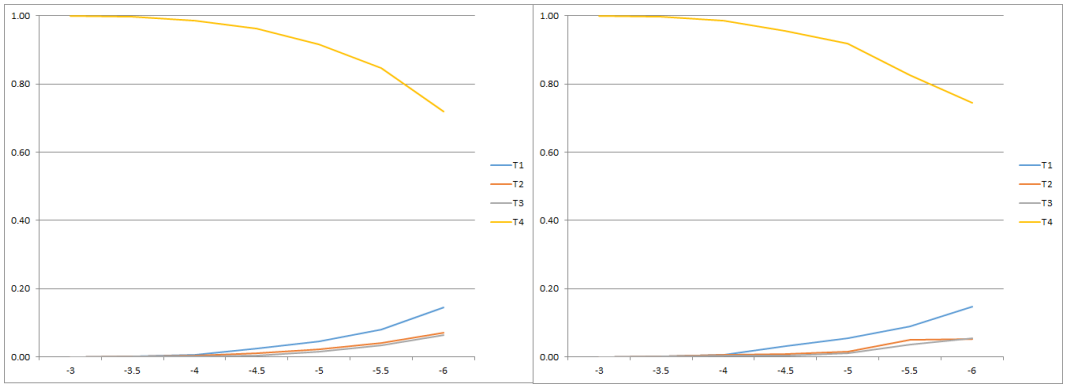


Figure 4.3.9: The left plot is the best rank probabilities of ladder geometry for $c = 1$, the right plot is the best rank probabilities of ladder geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

4.3.3.3 Loop Geometry

Table 4.3.5 shows part of the simulation results for the loop geometry without any continuity correction. The coverage probabilities and bias were not as bad in this network pattern. However, the length of the 95% posterior intervals increased

almost twice as much when μ decreased from -4.5 to -5 in the situations where the total number of observations was $n = 100$.

Table 4.3.5: No Continuity Correction for Loop Geometry

μ_i	Para.	Prob.	Length	Bias	μ_i	Prob.	Length	Bias
-4.5	b_{12}	0.92	6.73	1.00	-5	0.92	12.41	1.90
	b_{13}	0.92	13.14	1.87		0.95	25.71	3.84
	b_{14}	1.00	3.30	0.30		0.95	5.91	0.71
	b_{23}	0.96	2.39	0.16		0.96	3.80	0.37
	b_{24}	0.94	11.40	1.75		0.93	23.21	3.62
	b_{34}	0.94	5.71	0.61		0.91	15.94	2.38

Figures 4.3.10 and 4.3.11 show the simulation results with the continuity correction. Both the length of the 95% posterior intervals and the bias were reduced compared to the results without the continuity correction.

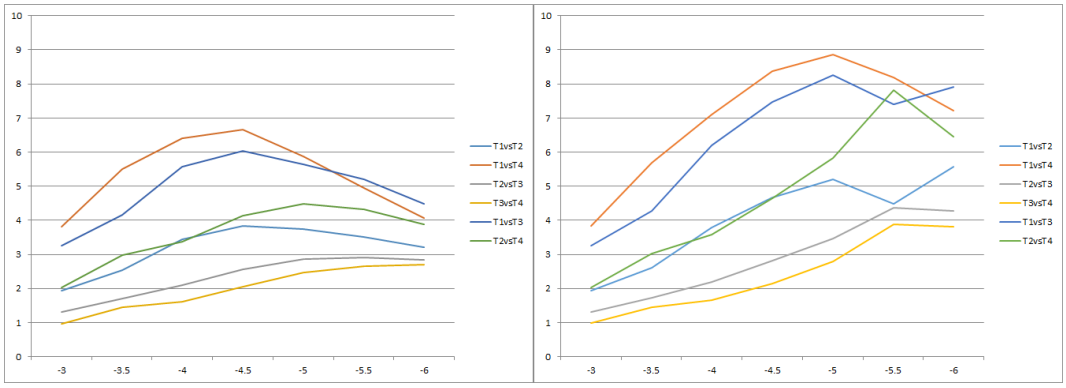


Figure 4.3.10: The left plot is the length of the 95% posterior interval of loop geometry for $c = 1$, the right plot is the length of the 95% posterior interval of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

Generally speaking, $c = 1$ had better performance in the length of the 95% posterior interval, while $c = 0.5$ provided less biased estimates. The studies of $(T_1\&T_4)$, $(T_1\&T_3)$, and $(T_2\&T_4)$ had wider 95% posterior intervals than the others. In

terms of bias, (T_1&T_4) performed worse than the others. As μ decreased, the coverage probabilities decreased for both continuity correction methods, especially for the study (T_1&T_4). The correction $c = 1$ provided even worse coverage probabilities than $c = 0.5$, especially when μ was small.

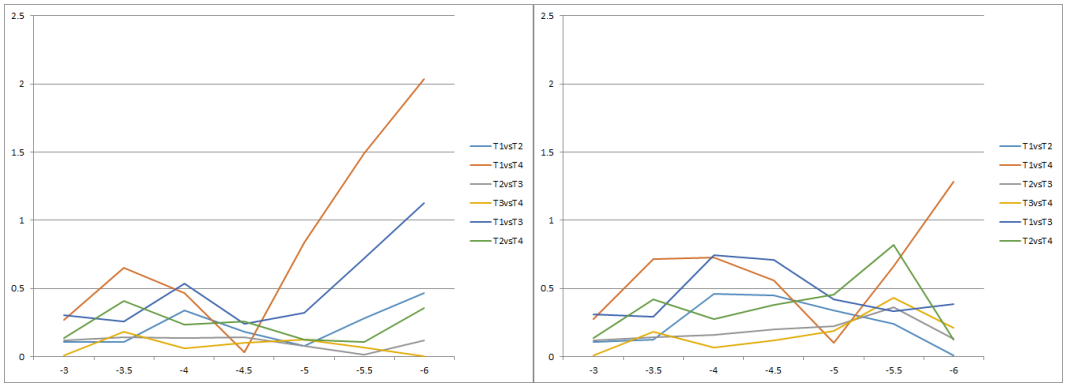


Figure 4.3.11: The left plot is the bias of loop geometry for $c = 1$, the right plot is the bias of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

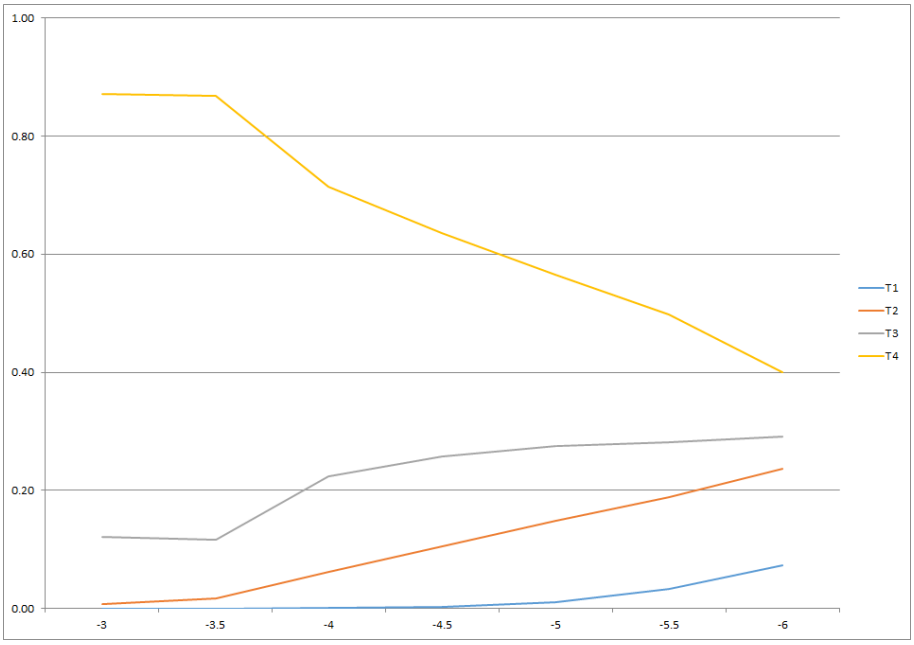


Figure 4.3.12: Best rank probabilities of loop geometry for no continuity correction; the horizontal axis represents the baseline log risk ratio μ.

In terms of the best rank probability, T_4 should once again be the best among all of the treatments. Figures 4.3.12 and 4.3.13 show that continuity correction did not provide better results than those with no continuity correction, no matter whether we added 0.5 or 1 to zero outcomes.

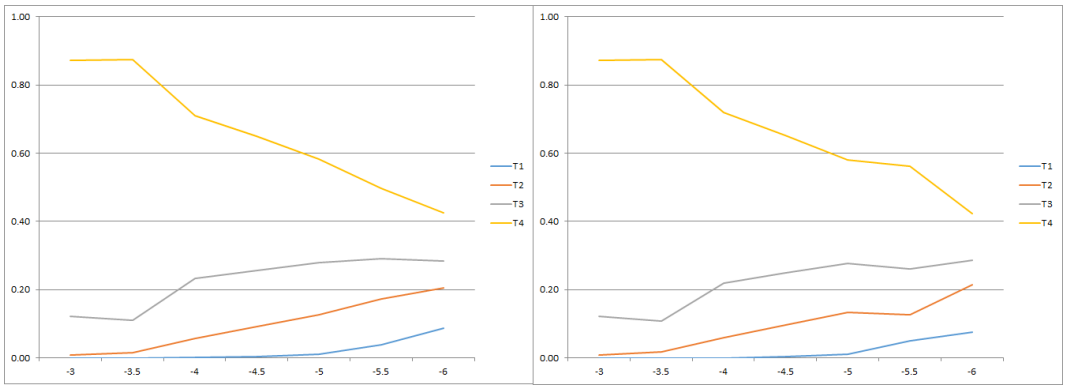


Figure 4.3.13: The left plot is best rank probabilities of loop geometry for $c = 1$, the right plot is the best rank probabilities of loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

4.3.3.4 One Closed Loop Geometry

For the one-closed loop geometry pattern, Table 4.3.6 shows part of the results without the continuity correction.

Table 4.3.6: No Continuity Correction for One-closed Loop Geometry

μ_i	Para.	Prob.	Length	Bias	μ_i	Prob.	Length	Bias
	b_{12}	0.92	4.99	0.69		0.95	7.90	0.82
	b_{13}	0.93	6.52	0.88		0.93	13.89	1.70
	b_{14}	0.93	4.15	0.43		0.96	11.02	1.72
-4.5					-5			
	b_{23}	0.98	6.71	0.68		0.94	17.79	2.83
	b_{24}	0.93	29.23	4.71		0.89	99.23	15.91
	b_{34}	0.95	29.79	3.12		0.94	99.57	12.47

When the number of total observations was 100, as μ decreased from -4.5 to -5, both the 95% posterior intervals and the bias increased substantially. Also, when $\mu = -5$, the coverage probability for the study (T_1&T_4) was below nominal.

Figures 4.3.14 and 4.3.15 show the simulation results when incorporating continuity correction. Generally speaking, indirect comparisons had poor performance in the width of the 95% posterior intervals and the biasness of estimates. Also, $c = 1$ had better performance in both the length of the 95% posterior intervals and the bias of estimates.

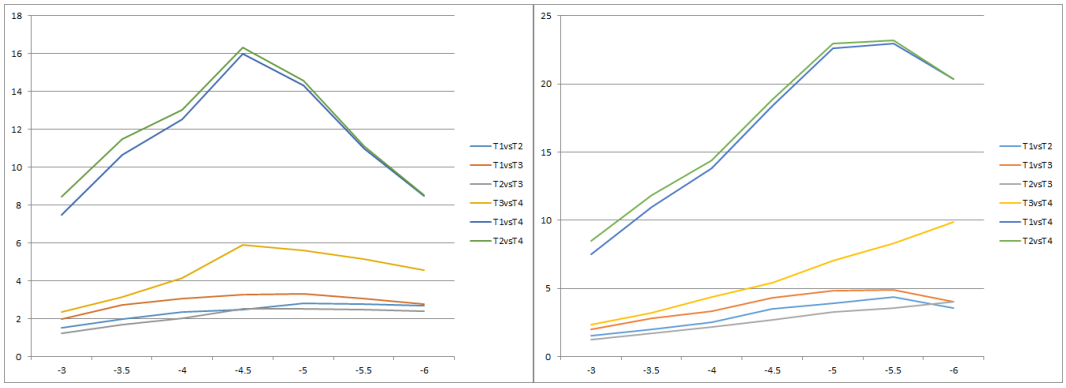


Figure 4.3.14: The left plot is the length of the 95% posterior interval of one-closed loop geometry for $c = 1$, the right plot is the length of the 95% posterior interval of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

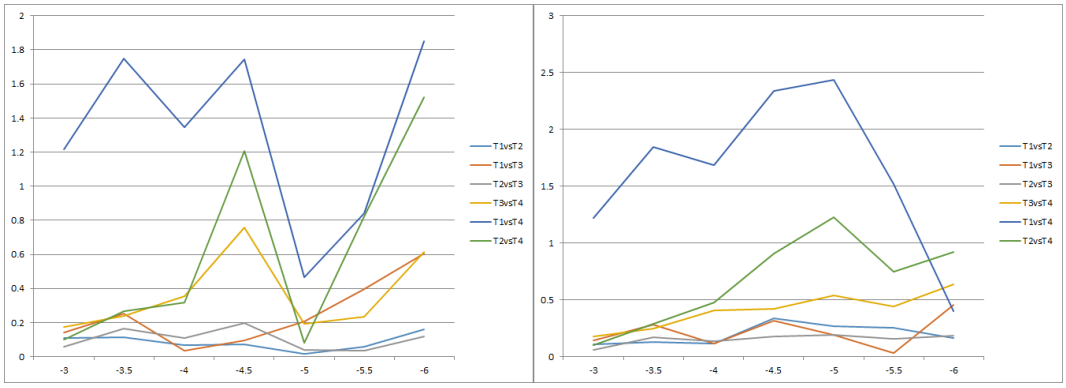


Figure 4.3.15: The left plot is the bias of one-closed loop geometry for $c = 1$, the right plot is the bias of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

We found that the studies of indirect comparison (T_1&T_4) and (T_2&T_4) had larger 95% posterior interval lengths than those of the direct comparisons. In terms of bias, the study (T_1&T_4) had the worst performance in both the situations of $c = 1$ and $c = 0.5$. For the coverage probabilities, $c = 1$ gave low coverage probabilities for (T_1&T_4) when μ was small.

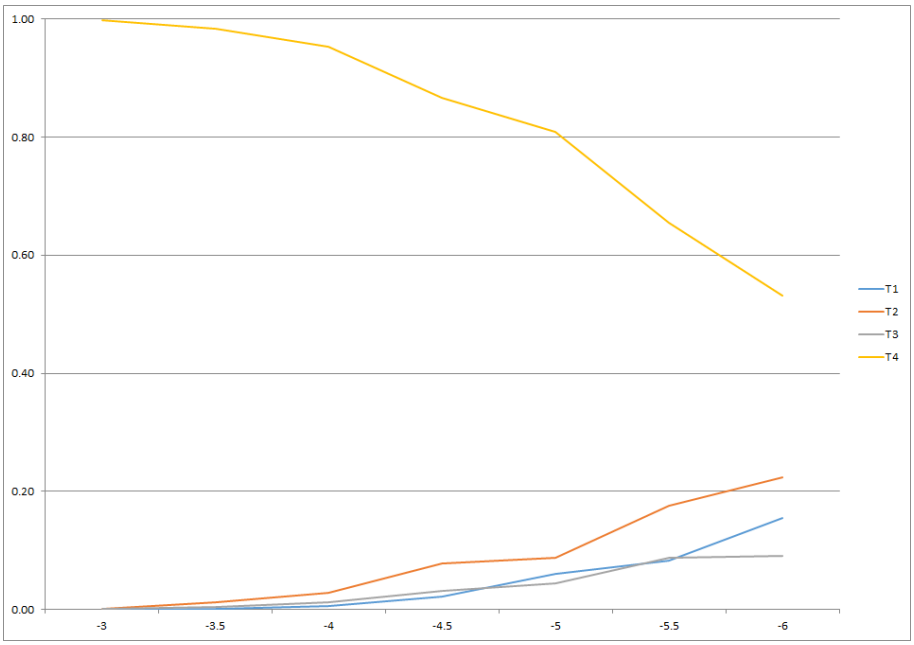


Figure 4.3.16: Best rank probabilities of one-closed loop geometry for no continuity correction; the horizontal axis represents the baseline log risk ratio μ.

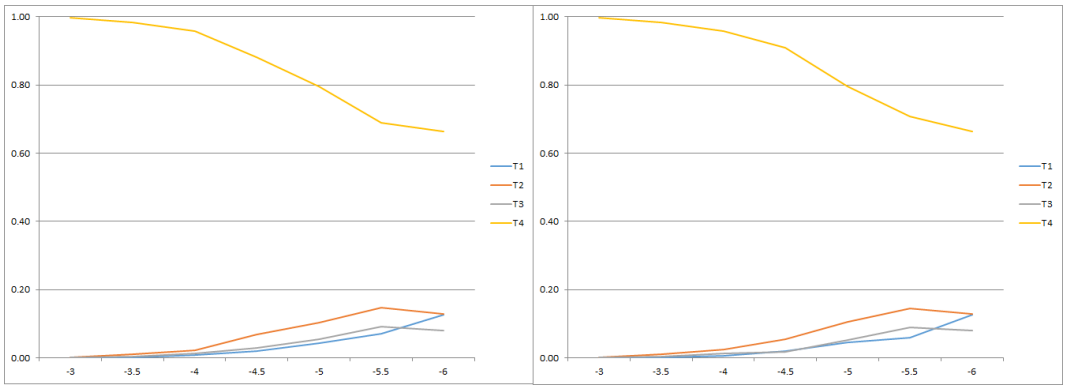


Figure 4.3.17: The left plot is the best rank probabilities of one-closed loop geometry for $c = 1$, the right plot is the best rank probabilities of one-closed loop geometry for $c = 0.5$, and the horizontal axis represents the baseline log risk ratio μ.

In terms of the best rank probability, T_4 was again the best among all of the treatments. Figures 4.3.16 and 4.3.17 show that the two continuity correction methods provided almost the same results in terms of the best rank probability. In the situation of $\mu = -6$, incorporating the continuity correction increased the best rank probability compared to the results from no continuity correction.

4.4 Conclusion

In this chapter, we have extended previous work on standard meta-analysis with rare events to network meta-analysis with rare events in the Poisson model. The Bayesian approach was used to obtain the estimates of the risk ratio between treatments. Continuity correction (adding a small number to both arms from a study where either outcome is zero) was performed to further reduce bias. Through simulation studies, we have shown that with more zeros in the outcome, the bias increases regardless of whether the continuity correction is used or not. In general, indirect comparisons (e.g., T2 vs. T3 & T2 vs. T4 in star geometry) have more biased estimates and wider 95% posterior intervals than direct comparisons (e.g., T1 vs. T2 & T1 vs. T3 in star geometry).

For comparing the performance of the two continuity correction methods, adding 1 to zero outcomes provides narrower 95% posterior intervals in all of the four geometry patterns. It also gives less biased estimates of risk ratio in star, ladder, and one-closed loop geometry. Adding 0.5 to zero outcomes only has advantages in terms of reducing bias for the loop geometry. For the probability to detect the most effective treatment, the continuity correction does not show improvements over no correction, especially in the situation where there is a clear superior treatment.

CHAPTER FIVE

Bayesian Network Meta-analysis for Binary Outcomes with Misclassification

5.1 Introduction

A generalization of meta-analysis is network meta-analysis, which allows for mutiple treatment comparisons in a single analysis. Network meta-analysis can not only provide head-to-head pairwise direct comparisons, but it also provides the indirect comparisons of two treatments which have never been compared with each other but have been compared to a common comparator. One standard assumption in both meta-analysis and network meta-analysis is that the outcomes are measured perfectly. Misclassification frequently occurs due to cost, convenience, misreporting by subjects, or the use of imperfect measurement devices in data collection. For example, a fallible diagnostic test can lead to a healthy subject being incorrectly diagnosed as sick (false-positive) or a sick subject being incorrectly diagnosed as healthy (false-negative). The use of misclassified data will lead to biased estimates and wrong statistical inference. Copeland et al. (1977) demonstrate that when misclassification is equal for the two compared groups, the estimate is biased toward the null value; when differential misclassification occurs, the bias can be in either direction.

There are at least two approaches used to adjust for misclassified data. First, a gold-standard measure is available that always gives the correct classification, but it is too expensive to be applied to everyone in the study. Prescott and Garthwaite (2002) provide a two-stage Bayesian analysis of misclassified binary data. They use the posterior distribution from the first stage as the prior distribution for the second stage, thus transferring all relevant information between the stages. Second, if no gold-standard measure is availiable, misclassification probabilities (e.g., *se* and

sp) should be incorporated to adjust for misclassified outcomes. Edwards et al. (2014) use fixed values of the *se* and *sp* to adjust misclassified counts and propose modified maximum likelihood to estimate the rate ratio of lung cancer death for a Poisson regression model. Paulino, Soares, and Neuhaus (2003) present a Bayesian binomial regression analysis in which the response is subject to an unconstrained misclassification process and informative priors were used.

Both Bayesian methods and likelihood methods are widely used to make misclassification adjustments. Chu (2007) shows that Bayesian methods can provide more reasonable and stable inferences when the resulting data are sparse, which is of particular relevance to small validation data sets in rare exposure contexts. The inclusion of prior information offered by the Bayesian approach can be used to effectively adjust for bias. Rather than committing to non-differential or "fully" differential assumptions concerning the exposure misclassification, a prior can be constructed to represent a "nearly non-differential" assumption.

In this chapter, we focus on adjusting for misclassified binary outcomes in network meta-analysis. The Bayesian approach is used to perform statistical inference. In Section 5.2, we overview the network meta-analysis and incorporate *se* and *sp* in the model. Section 5.3 presents simulation details, including simulation design, performance evaluation, and simulation results. We discuss the results of simulation studies and give some concluding comments in Section 5.4.

5.2 Methods

5.2.1 Fixed and Random Effects Network Meta-analysis Models

Following the notation used in Greco et al. (2013), suppose we have N randomized controlled trials making mixed comparisons among K treatments. Define r_{ik} and n_{ik} as the number of events and the total observations on treatment k in the i^{th} trial, respectively. Furthermore, let p_{ik} be the probability of event occur-

rence. For the binary outcomes, we assume the number of events r_{ik} has a binomial distribution,

$$r_{ik} \sim \text{binomial}(p_{ik}, n_{ik}), \quad i = 1, 2, \ldots, N; k = 1, 2, \ldots, K.$$

It is common to use logistic regression to model the probability of event occurrence p_{ik}:

$$\begin{aligned} \text{logit}(p_{ib}) &= \log\left(\frac{p_{ib}}{1-p_{ib}}\right) = \mu_i, \quad i = 1, 2, \ldots, N; k = b = 1, 2, \ldots, K, \\ \text{logit}(p_{ik}) &= \log\left(\frac{p_{ik}}{1-p_{ik}}\right) = \mu_i + \delta_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k, \end{aligned} \tag{5.1}$$

where μ_i are the trial-specific baselines and represent the log odds ratio of events in the referent treatment ($k = b$). Also, $\delta_{i,bk}$ are the trial-specific log odds ratios of event occurrence of the treatment group k compared with the reference treatment.

The nature of the effect $\delta_{i,bk}$ depends on the underlying assumptions. Two models that are commonly used in meta-analysis are the fixed effects and random effects models. The difference between these two models is the way between-study variation is accounted for. With the fixed effects model, the assumption is made that each observed individual study has a shared common treatment effect and that differences between studies are caused by chance. Individual studies are simply weighted by their precision. For the fixed effects model, Equation 5.1 can be replaced as follows:

$$\text{logit}(p_{ik}) = \mu_i + d_{i,bk}, \quad i = 1, 2, \ldots, N; k = 2, \ldots, K; b < k,$$

where μ_i are the trial-specific baselines and $d_{i,bk}$ are the fixed log odds ratios of event occurrence of the treatment group k compared with the reference treatment. In this model, the between-study variation is equal to zero.

The random effects model allows for the existence of between-study heterogeneity, meaning the underlying effect for each study is different. In addition, it is

often assumed that these true effects are described by a normal distribution, and our interest is in estimating the mean of this normal distribution. For a random effects model, the trial-specific log odds ratio is assumed $\delta_{i,bk}$:

$$\delta_{i,bk} \sim N(d_{bk}, \sigma^2).$$

In terms of comparison between treatments, there are direct comparisons and indirect comparisons. Direct comparisons are made between treatments having head-to-head randomized studies. In the star geometry (Figure 3.2.1), A-B, A-C, and A-D directly compare with each other, so in the language of network meta-analysis, we call the parameters which can be estimated from direct comparisons the basic parameters of the model, like d_{AB}, d_{AC}, and d_{AD}. Indirect comparisons are made between treatments in the absence of head-to-head randomized studies but have one common comparator. For example, treatments B, C, and D are linked via a common comparator A, and then d_{BC}, d_{BD}, and d_{CD} can be calculated based on the pooled estimates for the basic parameters. Those parameters, like d_{BC}, d_{BD}, and d_{CD}, are called functional parameters.

The key assumption in which we can estimate functional parameters (based on the pooled estimates for basic parameters) is the consistency assumption. We assume this because it is important that the indirect estimate is unbiased and that there are no discrepancies between the direct and indirect comparisons. For example, in one-closed loop geometry in Figure 3.2.3, d_{BC} can be directly estimated from studies B-C, but it also can be indirectly calculated from d_{AB} and d_{AC}. The consistency assumption requires that the following equation be satisfied: $d_{BC} = d_{AC} - d_{AB}$.

In general, for the estimate of indirect treatment comparison d_{st}, we have

$$d_{st} = d_{bt} - d_{bs}, \quad b = 1, 2, \ldots, K; s = 2, 3, \ldots, K; t = 3, 4, \ldots, K; s < t.$$

5.2.2 Network Meta-analysis with Misclassification

In some situations, the binary outcomes are not measured perfectly, meaning the number of events observed is misclassified. Therefore, the observed number of events may lead to biased estimation. We assume x_{ik} is the true number of successes on the treatment k in the i^{th} trial.

Because the x_{ik} are not observed, the observed misclassified counts r_{ik} depend on the true counts x_{ik} and misclassified counts u_{ik} and v_{ik}. Let u_{ik} be the number of successes incorrectly labeled as failures, and let v_{ik} be the number of failures incorrectly labeled as successes on the k^{th} treatment in the i^{th} trial. Therefore, we have the observed number of successes and failures:

$$r_{ik} = x_{ik} - u_{ik} + v_{ik}, \quad q_{ik} = y_{ik} + u_{ik} - v_{ik}.$$

Similar to the unobserved counts, the observed counts follow binomial distributions, but the success probabilities are functions of the prevelance, the *se*, and the *sp*. That is,

$$r_{ik} \sim \text{binomial}(n_{ik}, p^*{}_{ik}),$$

and

$$p^*{}_{ik} = p_{ik} \times se_{ik} + (1 - p_{ik}) \times (1 - sp_{ik}),$$

where se_{ik} is the probability a success is correctly labeled as a success on the treatment k in the i^{th} trial. Also, sp_{ik} is the probability a failure is correctly labeled as a failure on the treatment k in the i^{th} trial. Thus, the r_{ik} are biased for the p_{ik}, and r_{ik} provides information only about the quantity $p_{ik} \times se_{ik} + (1 - p_{ik}) \times (1 - sp_{ik})$.

5.2.3 Choice of Priors

In the Bayesian framework, network meta-analysis can be treated as a hierarchical model. Compared to frequestist methods, a Bayesian approach can incorporate prior information on the parameters, and the parameters are viewed as random

variables that are constrained by a prior distribution. Gelman (2006) suggests to use weakly informative priors on scale parameters, which are the trial-specific baselines (μ_i) and the log odds ratios of the treatment group to the control group (d_{bk}). The normal distribution (with a large variance) is widely used as a weakly informative prior. Therefore, the priors for those parameters are

$$\mu_i, d_{bk} \sim N(0, 10^4).$$

In terms of priors for the between-study standard deviation, the inverse gamma prior for the variance is often used because of the conjugacy. However, Gelman (2006) show the serious problem of the inverse gamma prior for the variance parameter and suggest the use of a half-Cauchy distribution instead of the uniform distribution. We use the half-normal distribution to approximate the half-Cauchy distribution, so the prior distribution for the standard deviation is

$$\sigma \sim N(0, 5^2)^+,$$

which is the positive part of the normal distribution with mean 0 and standard deviation 5.

If *se* and *sp* are unknown, a good choice is a beta prior. The beta distribution is flexible, and it also is the conjugate prior for the binomial likelihood. However, for our overparameterized model, the unknown *se* and *sp* leads to convergence issues, especially in the situation with a limited number of observations. Therefore, instead of assuming *se* and *sp* are unknown, we fix both of them and perform a sensitivity analysis.

5.3 Simulation Study

5.3.1 Simulation Design

In the simulation study, we aimed to show the advantages of accounting for misclassification in the network meta-analysis rather than ignoring it. We also inves-

tigated the effect of heterogeneity on the estimation of the odds ratio. In addition, we showed the impact of using imperfect *se* and *sp* in the analysis. For simplicity, we only considered the star geometry with one common placebo (the reference treatment) and three competing treatments. This method can easiliy be extended to other geometries.

We chose the random effects model in the network meta-analysis, because the fixed effects model could be treated as a special case of the random effects model in which the between-study variance is zero. We denote the treatments under investigation by treatment A (T_1), which we considered to be the reference treatment, treatment B (T_2), treatment C (T_3), and treatment D (T_4).

Table 5.3.1: True Value of Odds Ratio

Study	b_{21}	b_{31}	b_{41}	b_{32}	b_{42}	b_{43}
Star Geometry	1.5	2	2.5	1.33	1.67	1.25

Table 5.3.1 shows the true mean of the odds ratios between treatments for all network patterns. For example, $b_{21} = 1.5$ means that the odds ratio of treatment B to treatment A (reference treatment) was 1.5.

Table 5.3.2: Simulation Scenarios

Parameter	Value
sensitivity	0.9,0.8
specificity	0.9,0.8,0.7,0.6
σ	0.1,0.25,0.5,0.75

In this simulation study, we assumed the study sample size was 200, and we had five studies for each comparison, which brought the number of studies to 15. Table 5.3.2 shows the values of *se*, *sp*, and the between-study standard deviation.

We chose relatively higher *se* values, 0.9 and 0.8, and let *sp* vary from 0.9 to 0.6. The between study standard deviation σ varied from 0.1 to 0.75

We randomly sampled 1000 data sets for each of the simulation scenarios. Then, we fit the hierarchical Bayesian mixed treatment comparison model on each data set and performed statistical inference. In our situation, we discarded the first 20000 iterations as a burn-in to reduce the effect of initial values on the inference. Auto-correlation between posterior samples can lead to poor mixing or slow convergence of the Markov chain. Also, we thinned the Markov chain by keeping every twenty-fifth simulated draw from each MCMC sequence. Therefore, our inference about the parameters of interest was based on 20000 interations after a burn-in first discarded 20000.

5.3.2 Simulation Results

First, we show the advantages of incorporating the *se* and *sp* in the analysis. Table 5.3.3 shows the average posterior means of two pairs of *se* and *sp*, which were $se = 0.9, sp = 0.9$ and $se = 0.9, sp = 0.7$, with two between-study standard deviations of $\sigma = 0.1$ and $\sigma = 0.25$.

Table 5.3.3: Average Posterior Means Across 1000 Simulations

	$se = 0.9, sp = 0.9$		$se = 0.9, sp = 0.7$	
σ	0.1	0.25	0.1	0.25
$\hat{b_{21}}$	1.53(1.39)	1.52(1.39)	1.56(1.30)	1.58(1.30)
$\hat{b_{31}}$	2.07(1.74)	2.05(1.74)	2.09(1.63)	2.13(1.57)
$\hat{b_{41}}$	2.58(2.05)	2.56(2.06)	2.67(1.80)	2.69(1.80)
$\hat{b_{32}}$	1.40(1.27)	1.40(1.28)	1.42(1.21)	1.45(1.23)
$\hat{b_{42}}$	1.74(1.50)	1.75(1.52)	1.80(1.40)	1.84(1.41)
$\hat{b_{43}}$	1.29(1.19)	1.30(1.22)	1.36(1.17)	1.37(1.17)

The values in the parentheses are the average posterior means from the "naive" models, which ignore the misclassification in the analysis, and the values out of the parentheses are the average posterior means from the "correct" models, which accounted for se and sp in the analysis.

It is clear from Table 5.3.3 that the models accounting for misclassification provided better estimates of the odds ratios than that of the "naive" models in all situations. In addition, for the odds ratios of direct comparisons (b_{21}, b_{31}, and b_{41}), the "correct" models had little bias. However, for the "naive" model, even in the situations of $se = 0.9, sp = 0.9$, the estimates of odds ratios were very biased, and their performance worsened when se and sp decreased. For the odds ratios of indirect comparisons (b_{32}, b_{42}, and b_{43}), both "navie" and "correct" models provided biased estimates, but the "correct" model still provided better results than the "naive" model.

Table 5.3.4: Average Coverage of the 95% Intervals Across 1000 Simulations

	$se = 0.9, sp = 0.9$		$se = 0.9, sp = 0.7$	
σ	0.1	0.25	0.1	0.25
$\hat{b_{21}}$	0.97(0.82)	0.95(0.87)	0.98(0.84)	0.96(0.68)
$\hat{b_{31}}$	0.98(0.53)	0.95(0.73)	0.97(0.06)	0.95(0.32)
$\hat{b_{41}}$	0.97(0.23)	0.95(0.59)	0.97(0.00)	0.95(0.08)
$\hat{b_{32}}$	0.98 (0.92)	0.94(0.93)	0.98(0.85)	0.96(0.89)
$\hat{b_{42}}$	0.97(0.82)	0.96(0.89)	0.97(0.60)	0.96(0.75)
$\hat{b_{43}}$	0.97 (0.94)	0.95(0.95)	0.97(0.92)	0.95(0.89)

Table 5.3.4 shows the average coverage of the 95% intervals across 1000 simulations in the same setting as Table 5.3.3. The "correct" model provided very good coverage probabilities, which were all around 95%, while the "naive" model performed poorly in terms of coverage probability, especially when the sp and the

between-study standard deviation were both small. It is interesting that for the "naive" model, the coverage probabilities of the 95% intervals of indirect comparisons were closer to 95% than those from the direct comparisons.

Table 5.3.5 shows the average width of the 95% intervals across 1000 simulations, also in the same setting as Table 5.3.3. The "naive" models had a narrower interval than the "correct" models. This is because accounting for misclassification increases uncertainty, and the estimates should have wider posterior intervals. Although the "naive" models provided narrower intervals, the average means of odds ratios were very biased, and coverage probabilities were below nominal (Tables 5.3.3 and 5.3.4). Therefore, the "naive" models provided precise but biased estimates.

Table 5.3.5: Average Width of the 95% Intervals Across 1000 Simulations

	$se = 0.9, sp = 0.9$		$se = 0.9, sp = 0.7$	
σ	0.1	0.25	0.1	0.25
$\hat{b_{21}}$	0.86(0.41)	0.88(0.62)	1.13(0.39)	1.29(0.52)
$\hat{b_{31}}$	1.18(0.52)	1.19(0.77)	1.54(0.47)	1.76(0.62)
$\hat{b_{41}}$	1.51(0.62)	1.50(0.92)	2.02(0.55)	2.27(0.72)
$\hat{b_{32}}$	1.12 (0.54)	1.15(0.81)	1.47(0.51)	1.69(0.69)
$\hat{b_{42}}$	1.42(0.63)	1.45(0.96)	1.90(0.60)	2.16(0.79)
$\hat{b_{43}}$	1.06 (0.51)	1.08(0.77)	1.44(0.50)	1.63(0.66)

Table 5.3.6 shows the results when we fixed *se* and *sp* at 0.9 and 0.8, respectively, and it also shows the results from when we varied between-study standard deviation from 0.1 to 0.75. Each cell contains the average posterior and the average width of the 95% intervals (in the parentheses). As the between-study standard deviation increased, the estimates of the odds ratios were more biased. In addition, it is not surprising that the width of the 95% interval increased as the variation between studies increased.

Table 5.3.6: Average Posterior Means and Average Widths of the 95% Intervals Across 1000 Simulations

σ	0.1	0.25	0.5	0.75
$\hat{b_{21}}$	1.55(0.98)	1.54(0.92)	1.63(1.76)	1.73(2.51)
$\hat{b_{31}}$	2.08(1.35)	2.06(1.24)	2.16(2.36)	2.72(3.35)
$\hat{b_{41}}$	2.65(1.76)	2.57(1.57)	2.75(3.06)	2.85(4.24)
$\hat{b_{32}}$	1.41(1.28)	1.39(1.19)	1.53(2.38)	1.72(3.61)
$\hat{b_{42}}$	1.79(1.65)	1.73(1.49)	1.96(3.09)	2.15(4.53)
$\hat{b_{43}}$	1.33(1.24)	1.30(1.12)	1.48(2.32)	1.65(3.47)

5.3.3 Sensitivity Analysis

We next investigated robustness. Specifically, we were interested in the impact from imperfect estimation of the *se* and *sp*. First, we assumed that *se* was known and that *sp* was unknown. We assumed a value of 0.9 for both *se* and *sp*, and the between-study standard deviation was 0.1. For the simulation, we fixed *se* at the true value 0.9 and considered a range of values for *sp* ($sp = 0.8, 0.7, 0.6$).

Table 5.3.7: Posterior Results of Imperfect Specificity ($se = 0.9$)

	Mean			Coverage			Width		
sp	0.8	0.7	0.6	0.8	0.7	0.6	0.8	0.7	0.6
$\hat{b_{21}}$	1.64	1.86	2.69	0.95	0.88	0.66	1.05	1.46	3.66
$\hat{b_{31}}$	2.24	2.66	4.34	0.94	0.76	0.34	1.45	2.09	5.94
$\hat{b_{41}}$	2.85	3.49	6.62	0.92	0.70	0.20	1.87	2.77	8.10
$\hat{b_{32}}$	1.43	1.54	1.95	0.97	0.97	0.96	1.30	1.71	3.65
$\hat{b_{42}}$	1.82	2.01	2.85	0.97	0.96	0.92	1.67	2.25	4.91
$\hat{b_{43}}$	1.34	1.40	1.86	0.97	0.97	0.96	1.23	1.56	3.01

Table 5.3.7 shows the average posterior means, coverage rates, and widths of the 95% posterior intervals for different values of sp. The results show that when $sp = 0.8$, which is close to the true value, the average means of the odds ratios showed little bias from the true value, and the coverage rates were close to nominal rates. However, as sp decreased, the estimates of the odds ratios were more biased, the coverage rates of the 95% posterior intervals decreased dramatically, and the widths of the 95% intervals became wider. Therefore, the posterior results were quite sensitive to the sp. It is interesting that the coverage rates of indirect comparisons were close to nominal, even though sp was far away from the true value.

Next, we considered that both se and sp were imperfect, but neither of them were far away from the true value. The between-study standard deviation was again fixed at 0.1. In this situation, we assumed both se and sp were 0.1 away from the true value. For example, if we assumed $se = 0.9$ and $sp = 0.8$, we used $se = 0.8$ and $sp = 0.7$ for the simulation. We considered several pairs of se and sp in the setting: $se = 0.9, sp = 0.9$; $se = 0.9, sp = 0.8$; and $se = 0.9, sp = 0.7$. Therefore, in this simulation, we used $se = 0.8, sp = 0.8$; $se = 0.8, sp = 0.7$; and $se = 0.8, sp = 0.6$, respectively, to do our analysis.

Table 5.3.8: Posterior Results with Imperfect Sensitivity and Specificity

	$se = 0.9, sp = 0.9$			$se = 0.9, sp = 0.8$			$se = 0.9, sp = 0.7$		
	Mean	Coverage	Width	Mean	Coverage	Width	Mean	Coverage	Width
$\hat{b_{21}}$	1.82	0.90	1.44	1.92	0.89	1.82	2.14	0.87	2.55
$\hat{b_{31}}$	2.84	0.71	2.40	1.92	0.89	1.82	3.74	0.63	5.08
$\hat{b_{41}}$	4.10	0.48	3.77	4.86	0.42	5.74	1.84E10	0.37	1.96E10
$\hat{b_{32}}$	1.67	0.93	1.94	1.77	0.94	2.47	2.04	0.94	3.69
$\hat{b_{42}}$	2.41	0.86	2.94	2.79	0.84	4.24	1.47E10	0.82	9.63E10
$\hat{b_{43}}$	1.57	0.94	1.95	1.77	0.92	2.75	3.90E9	0.92	7.62E9

Table 5.3.8 shows the posterior results of these three situations. It provides the posterior means, coverage rates, and widths of the 95% posterior intervals of the odds ratios. In the situation of $se = 0.9$ and $sp = 0.9$, using imperfect se and sp led to slightly biased estimates, and the coverage rates of the 95% intervals were below nominal. When se and sp decreased, the results became worse, posterior estimates were more biased, coverage rates decreased, and the width of the 95% intervals increased, especially in the situation of $se = 0.9, sp = 0.7$.

Table 5.3.9 shows the results after the se and sp used in the analysis were larger than the true value by 0.1. In these three cases, the use of imperfect se and sp provided very similar results. The posterior means of all the odds ratios were moderately biased from the true value. Also, the coverage rates of $\hat{b}_{31}$ and $\hat{b}_{41}$ were below nominal, but the rest were all close to 95%. Compared to Table 5.3.8, all of the posterior means, coverage rates, and widths of the 95% posterior intervals had considerable improvements, especially in the situations with low se and sp.

Table 5.3.9: Posterior Results with Imperfect Sensitivity and Specificity

	$se = 0.9, sp = 0.8$			$se = 0.9, sp = 0.7$			$se = 0.8, sp = 0.8$		
	Mean	Coverage	Width	Mean	Coverage	Width	Mean	Coverage	Width
$\hat{b_{21}}$	1.39	0.97	0.67	1.38	0.91	0.73	1.39	0.92	0.76
$\hat{b_{31}}$	1.72	0.77	0.84	1.70	0.75	0.90	1.70	0.78	0.95
$\hat{b_{41}}$	2.03	0.59	0.99	1.99	0.58	1.06	2.00	0.62	1.13
$\hat{b_{32}}$	1.28	0.95	0.87	1.27	0.95	0.95	1.27	0.96	0.99
$\hat{b_{42}}$	1.50	0.92	1.03	1.48	0.91	1.12	1.49	0.92	1.18
$\hat{b_{43}}$	1.21	0.97	0.83	1.20	0.97	0.91	1.22	0.95	0.97

5.4 Conclusion

In this chapter, we extended previous work on network meta-analysis that allows for misclassification. The Bayesian approach was used to get the estimates of

odds ratios between treatments. The important advantage of the Bayesian method in network meta-analysis is that there was an allowance to incorporate prior belief about the parameters of interest in the analysis. In Section 5.2, we discussed the network meta-analysis in both fixed and random effects models. In addition, we adjusted misclassified counts by incorporating *se* and *sp*. In Section 5.3, we described the design of our simulation and presented the simulation results.

In this simulation study, we have shown that under the fixed *se* and *sp*, the increasing of the standard deviation for the between-study standard deviations lead to more biased estimates and wider 95% posterior intervals. We also investigated the danger of ignoring misclassification, which can lead to wrong conclusions as it underestimated the odds ratio between treatments. Although ignoring misclassification provided narrower 95% posterior intervals, the coverage of the 95% posterior intervals was well below nominal.

In the situation where *se* and *sp* are unknown, the common strategy in the Bayesian approach is to assume two independent beta prior distributions for *se* and *sp*. However, for the network meta-analysis (the overparameterized model), the unknown *se* and *sp* leads to convergence issues. Therefore, we investigated the robustness. The simulation study shows that if either *se* or *sp* were perfect, the other was imperfect. If the imperfect one slightly departed from the true value, we still got good estimates and coverage probabilities. As it shifted more from the true value, the estimates were more biased with low coverage probabilities. It is surprising that the results of the indirect comparisons were better than those of the direct comparisons, especially with the coverage probabilities.

It is very dangerous to use the imperfect *se* and *sp*, even though they are slightly departed from the true value. The results are very sensitive to the misclassification rates. Our simulation study showed that the use of slightly overestimated *se* and *sp* will provide better estimates than that of using underestimated *se* and

sp, especially when *se* and *sp* are low. Therefore, in a real world analysis, it is very important to detect the true *se* and *sp*. We should at least find one of the true values so that we can get correct or less biased estimates. In the case where both *se* and *sp* are unknown, using slightly larger *se* and *sp* would provide more consistent results.

APPENDICES

APPENDIX A

R and BUGS Programs for Misclassified and Overdispersed Poisson Data

The programs presented here were used for Bayesian analysis to account for misclassification and overdispersion in the Poisson model, which was discribed in Chapter Two. We generated data for the simulation using R, and we implemented the Bayesian approach by WinBUGS.

```
#################parameters in the R function#########################
##k: number of groups
##b0, b1, a0, a1: coefficients
##x: exposure
##n: person year
##sp, se: specificity and sensitivity
##err: random error
##lam: true death rate due to lung cancer
##mu: true death rate due to other cancer
##rate1: observed death rate due to lung cancer
##rate2: observed death rate due to other cancer
##w: observed death number due to lung cancer
##e: observed death number due to other cancer
##model: random or fixed model
##sesp: sp and se known or unknown
##a.se, b.se, a.sp, b.sp: prior parameters for se and sp

pmisc = function(a0, a1, a2, a3, b0, b1, b2, b3, sig, n, se, sp,
               model, sesp, a.se=90, b.se=10, a.sp=90, b.sp=10, M)
```

```
{
  simout <- as.list(rep(NA, M))
  ##covariates in the Poisson model
  a = c(0,1)
  b = c(0,1)
  d = c(0,1)
  x = matrix(NA, ncol = 3, nrow = 8)
  y = matrix(NA, ncol = 3, nrow = 8)
  z = matrix(NA, ncol = 3, nrow = 8)
  v = matrix(NA, ncol = 3, nrow = 8)
  i = 1
  for (j in 1:2)
  { for (k in 1:2)
    { for (h in 1:2) {
        x[i,] <- c(a[j],b[k],d[h])
        y[i,] <- c(a[j],b[k],d[h])
        z[i,] <- c(a[j],b[k],d[h])
        v[i,] <- c(a[j],b[k],d[h])
        i = i + 1
      } } }
  x.4 = rbind(x,y,z,v)
  k = dim(x.4)[1]
  x1 = x.4[,1]
  x2 = x.4[,2]
  x3 = x.4[,3]
```

```
################random effects model with known se and sp##############
  if (model == 'random')
  {if (sesp == 'known')
  {for (i in 1:M) {
    err.b = rnorm(k, 0, sig^0.5)
    err.a = rnorm(k, 0, sig^0.5)
    lam = exp(b0 + b1*x1 + b2*x2 + b3*x3 + err.b)
    mu = exp(a0 + a1*x1 + a2*x2 + a3*x3 + err.a)
    rate1 = lam*se + mu*(1-sp)
    rate2 = lam*(1-se) + mu*sp
    w = rpois(k, rate1*n)
    e = rpois(k, rate2*n)
    parameters = list('a0','a1','a2','a3','b0','b1','b2','b3','sd')
    data = list('w', 'e', 'x1','x2','x3', 'k','n', 'se', 'sp')
    inits = list(a0 = 0,a1 = 0,a2 = 0,a3 = 0,b0 = 0,b1 = 0,b2 = 0,
          b3 = 0,err.a = rep(0,k),err.b = rep(0,k),sd = 1)
    inits = list(inits)
    ##call WinBUGS in R
    sim = bugs(data, inits, parameters, 'code_rk.txt',
        n.chains = 1, n.burnin = 50000, n.iter = 500000,
        n.thin = 25,bugs.directory="c:/WinBUGS/WinBUGS14/")
    df <- as.data.frame(sim$summary)
    df$should <- c(a0, a1, a2, a3, b0, b1, b2, b3, sig^0.5)
    simout[[i]] <- t(apply(df, 1, function(row){ c(
            est = unname( row["mean"] ),
            cover = unname( row["2.5%"] <= row["should"]
                  && row["should"] <= row["97.5%"] ),
```

```
            length = unname( row["97.5%"] - row["2.5%"] ) ) }))
      }}
###############random effects model with unknown se and sp#############
  if (sesp == 'unknown')
  {for (i in 1:M)  {
     err.b = rnorm(k, 0, sig^0.5)
      err.a = rnorm(k, 0, sig^0.5)
      lam = exp(b0 + b1*x1 + b2*x2 + b3*x3 + err.b)
      mu = exp(a0 + a1*x1 + a2*x2 + a3*x3 + err.a)
      rate1 = lam*se + mu*(1-sp)
      rate2 = lam*(1-se) + mu*sp
      w = rpois(k, rate1*n)
      e = rpois(k, rate2*n)
      parameters = list('a0','a1','a2','a3','b0','b1','b2',
                    'b3', 'sd', 'se', 'sp')
      data = list('w','e','x1','x2','x3','k','n','a.se',
                                'b.se','a.sp','b.sp')
      inits = list(a0 = 0,a1 = 0,a2 = 0,a3 = 0,b0 = 0,b1 = 0,
               b2 = 0,b3 = 0,err.a = rep(0,k),err.b = rep(0,k),
                                        sd = 1,se=0.5,sp=0.5)
      inits = list(inits)
      sim = bugs(data, inits, parameters, 'code_ru.txt',
           n.chains = 1,n .burnin = 50000, n.iter = 500000,
           n.thin = 25,bugs.directory="c:/WinBUGS/WinBUGS14/")
      df <- as.data.frame(sim$summary)
      df$should <- c(a0,a1,a2,a3,b0,b1,b2,b3,sig^0.5,se,sp)
      simout[[i]] <- t(apply(df, 1, function(row){c(
```

```
          est = unname( row["mean"] ),
          cover = unname( row["2.5%"] <= row["should"]
             && row["should"] <= row["97.5%"] ),
          length = unname( row["97.5%"] - row["2.5%"] ) ) }))
    }   }   }
################fixed effects model with known se and sp#############
 if (model == 'fixed')
 { if (sesp == 'known')
 {  for (i in 1:M)  {
   lam = exp(b0 + b1*x1 + b2*x2 + b3*x3)
   mu = exp(a0 + a1*x1 + a2*x2 + a3*x3)
   rate1 = lam*se + mu*(1-sp)
   rate2 = lam*(1-se) + mu*sp
   w = rpois(k, rate1*n)
   e = rpois(k, rate2*n)
   parameters = list('a0','a1','a2','a3','b0','b1','b2','b3')
   data = list('w', 'e', 'x1','x2','x3', 'k','n', 'se', 'sp')
   inits = list(a0=0,a1=0,a2=0,a3=0,b0=0,b1=0,b2=0,b3=0)
   inits = list(inits)
   sim = bugs(data, inits, parameters, 'code_fk.txt',
       n.chains = 1, n.burnin = 50000, n.iter = 500000,
      n.thin = 25,bugs.directory="c:/WinBUGS/WinBUGS14/")
   df <- as.data.frame(sim$summary)
   df$should <- c(a0, a1, a2, a3, b0, b1, b2, b3)
   simout[[i]] <- t(apply(df, 1, function(row){c(
           est = unname( row["mean"] ),
           cover = unname( row["2.5%"] <= row["should"]
```

```
            && row["should"] <= row["97.5%"] ),
          length = unname( row["97.5%"] - row["2.5%"] ) ) }))
    }   }
##############fixed effects model with unknown se and sp###############
 if (sesp == 'unknown')
 {  for (i in 1:M) {
    lam = exp(b0 + b1*x1 + b2*x2 + b3*x3)
    mu = exp(a0 + a1*x1 + a2*x2 + a3*x3)
    rate1 = lam*se + mu*(1-sp)
    rate2 = lam*(1-se) + mu*sp
    w = rpois(k, rate1*n)
    e = rpois(k, rate2*n)
    parameters = list('a0','a1','a2','a3','b0','b1','b2','b3',
                                              'se','sp')
    data = list('w','e','x1','x2','x3','k','n','a.se','b.se',
                                             'a.sp','b.sp')
    inits = list(a0 = 0, a1 = 0, a2 = 0, a3 = 0, b0 = 0, b1 = 0,
              b2 = 0, b3 = 0, se=0.5, sp=0.5)
    inits = list(inits)
    sim = bugs(data, inits, parameters, 'code_fu.txt',
        n.chains = 1, n.burnin = 50000, n.iter = 500000,
        n.thin = 25,bugs.directory="c:/WinBUGS/WinBUGS14/")
    df <- as.data.frame(sim$summary)
    df$should <- c(a0, a1, a2, a3, b0, b1, b2, b3, se, sp)
    simout[[i]] <- t(apply(df, 1, function(row){c(
        est = unname( row["mean"] ),
        cover = unname( row["2.5%"] <= row["should"]
```

```
                && row["should"] <= row["97.5%"] ),
          length = unname( row["97.5%"] - row["2.5%"] ) )  }))
    }   }  }

  ## format and return
  simTogether <- arrange(melt(simout), Var1, Var2)
  simTogether <- ddply(simTogether, c("Var1", "Var2"), function(df){
  mean(df$value) })
  simTogether <- dcast(simTogether, Var1 ~ Var2 , value.var = "V1")
  row.names(simTogether) <- simTogether$Var1
  simTogether <- simTogether[,-1]
  simTogether  }

######WinBUGS code for random effects model with known se and sp#######
model {
  for (i in 1:k) {
     log(lam[i]) <- b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] +err.b[i]
     log(mu[i]) <- a0 + a1*x1[i] + a2*x2[i] + a3*x3[i] +err.a[i]
     rate1[i] <- lam[i]*se + mu[i]*(1-sp)
     rate2[i] <- lam[i]*(1-se) + mu[i]*sp
     r1[i] <- rate1[i]*n
     r2[i] <- rate2[i]*n
     w[i] ~ dpois(r1[i])
     e[i] ~ dpois(r2[i])
     err.b[i] ~ dnorm(0, tau)
     err.a[i] ~ dnorm(0, tau)  }
     a0 ~ dnorm(0, 0.1)
```

```
    b0 ~ dnorm(0, 0.1)
    a1 ~ dnorm(0, 0.1)
    b1 ~ dnorm(0, 0.1)
    a2 ~ dnorm(0, 0.1)
    b2 ~ dnorm(0, 0.1)
    a3 ~ dnorm(0, 0.1)
    b3 ~ dnorm(0, 0.1)
    sd ~ dnorm(0,1)I(0.001,)
    tau <- 1/pow(sd,2)  }
#####WinBUGS code for random effects model with unknown se and sp######
model {
  for (i in 1:k) {
    log(lam[i]) <- b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] +err.b[i]
    log(mu[i]) <- a0 + a1*x1[i] + a2*x2[i] + a3*x3[i] +err.a[i]
    rate1[i] <- lam[i]*se + mu[i]*(1-sp)
    rate2[i] <- lam[i]*(1-se) + mu[i]*sp
    r1[i] <- rate1[i]*n
    r2[i] <- rate2[i]*n
    w[i] ~ dpois(r1[i])
    e[i] ~ dpois(r2[i])
    err.b[i] ~ dnorm(0, tau)
    err.a[i] ~ dnorm(0, tau)  }
    a0 ~ dnorm(0, 0.1)
    b0 ~ dnorm(0, 0.1)
    a1 ~ dnorm(0, 0.1)
    b1 ~ dnorm(0, 0.1)
    a2 ~ dnorm(0, 0.1)
```

```
    b2 ~ dnorm(0, 0.1)
    a3 ~ dnorm(0, 0.1)
    b3 ~ dnorm(0, 0.1)
    se ~ dbeta(a.se, b.se)
    sp ~ dbeta(a.sp, b.sp)
    sd ~ dnorm(0,1)I(0.001,)
    tau <- 1/pow(sd,2)  }

#######WinBUGS code for fixed effects model with known se and sp#######
model {
  for (i in 1:k) {
    log(lam[i]) <- b0 + b1*x1[i] + b2*x2[i] + b3*x3[i]
    log(mu[i]) <- a0 + a1*x1[i] + a2*x2[i] + a3*x3[i]
    rate1[i] <- lam[i]*se + mu[i]*(1-sp)
    rate2[i] <- lam[i]*(1-se) + mu[i]*sp
    r1[i] <- rate1[i]*n
    r2[i] <- rate2[i]*n
    w[i] ~ dpois(r1[i])
    e[i] ~ dpois(r2[i])   }
    a0 ~ dnorm(0, 0.1)
    b0 ~ dnorm(0, 0.1)
    a1 ~ dnorm(0, 0.1)
    b1 ~ dnorm(0, 0.1)
    a2 ~ dnorm(0, 0.1)
    b2 ~ dnorm(0, 0.1)
    a3 ~ dnorm(0, 0.1)
    b3 ~ dnorm(0, 0.1)  }
```

```
######WinBUGS code for fixed effects model with unknown se and sp######
model {
  for (i in 1:k) {
     log(lam[i]) <- b0 + b1*x1[i] + b2*x2[i] + b3*x3[i]
     log(mu[i]) <- a0 + a1*x1[i] + a2*x2[i] + a3*x3[i]
     rate1[i] <- lam[i]*se + mu[i]*(1-sp)
     rate2[i] <- lam[i]*(1-se) + mu[i]*sp
     r1[i] <- rate1[i]*n
     r2[i] <- rate2[i]*n
     w[i] ~ dpois(r1[i])
     e[i] ~ dpois(r2[i])   }
     a0 ~ dnorm(0, 0.1)
     b0 ~ dnorm(0, 0.1)
     a1 ~ dnorm(0, 0.1)
     b1 ~ dnorm(0, 0.1)
     a2 ~ dnorm(0, 0.1)
     b2 ~ dnorm(0, 0.1)
     a3 ~ dnorm(0, 0.1)
     b3 ~ dnorm(0, 0.1)
     se ~ dbeta(a.se, b.se)
     sp ~ dbeta(a.sp, b.sp) }
```

APPENDIX B

R and Stan Programs for Network Meta-analysis with Rare Events

In this appendix, we provide the R and Stan programs for the network meta-analysis with rare events for both the binomial and Poisson models, which were discussed in Chapter Three and Chapter Four. Star geometry in the binomial setting and loop geometry in the Poisson setting were shown as examples, but the other geometries could be easily extended through these programs.

B.1 Binomial Model

```
################parameters in the R function#########################
##m: number of simulation
##mu: baseline log odds ratio
##N: number of outcomes
##NS: number of studies (N/2)
##NT: number of treatments
##n: sample size in each study
##d.AB, d.AC, d.AD: log odds ratio
##sig: standard deviation of study
##cc: continuity correction
star_rank<-function(m,mu,N,NS,NT,n,d.AB,d.AC,d.AD,sig,seed,cc)
  {  set.seed(seed)
    simout <- as.list(rep(NA, m))
    simout.rank = as.list(rep(NA,m))
##design matrix for star geometry
  x<-matrix(NA, N, NT)
```

```
  x[,1]=1
  x[,2]=0
  x[,3]=0
  x[,4]=0
  x[2,2]=1
  x[4,2]=1
  x[6,2]=1
  x[8,2]=1
  x[10,2]=1
  x[12,3]=1
  x[14,3]=1
  x[16,3]=1
  x[18,3]=1
  x[20,3]=1
  x[22,4]=1
  x[24,4]=1
  x[26,4]=1
  x[28,4]=1
  x[30,4]=1
  n=rep(n, N)
  r<-vector("numeric", N)
##odds ratio
  OR.BC=exp(d.AC-d.AB)
  OR.BD=exp(d.AD-d.AB)
  OR.CD=exp(d.AD-d.AC)
  OR.AB=exp(d.AB)
  OR.AC=exp(d.AC)
```

```
 OR.AD=exp(d.AD)
##data generation
 for (i in 1:m){
   delta.AB=rnorm(NS, d.AB, sig)
   delta.AC=rnorm(NS, d.AC, sig)
   delta.AD=rnorm(NS, d.AD, sig)
   s=c(1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10,
          11,11,12,12,13,13,14,14,15,15)
   t=c(1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3,
          1,4,1,4,1,4,1,4,1,4)
   b=c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
          1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
   teb = 1 - 1*(t==b)
   for(j in 1:NS)
   { r[2*j-1] <- rbinom(1,n[2*j-1], 1/(1+exp(-(x[(2*j-1),] %*%
    c(mu, delta.AB[s[2*j-1]],delta.AC[s[2*j-1]],delta.AD[s[2*j-1]])))))
     r[2*j] <- rbinom(1,n[2*j], 1/(1+exp(-(x[2*j,] %*%
     c(mu, delta.AB[s[2*j]],delta.AC[s[2*j]],delta.AD[s[2*j]])))))
    ##continuity correction
     if(r[2*j-1]==0||r[2*j]==0)
     {   r[2*j-1] = r[2*j-1]+cc
          r[2*j] = r[2*j]+cc    }   }
##Stan code begins here
   data = list(N=N, NS=NS, NT=NT, n=n, t=t, b=b, teb=teb, r=r, s=s)
   code = "data {
   int<lower=0> N;
   int<lower=0> NS;
```

```
int<lower=0> NT;
int<lower=0> n[N];
real<lower=0> r[N];
int<lower=0> s[N];
int<lower=0> t[N];
int<lower=0> b[N];
int<lower=0> teb[N];   }
parameters {
real mu[NS];
real da[NT];
real delta[N];
real<lower=0> sdp;    }
transformed parameters {
real<lower=0> or12;
real<lower=0> or13;
real<lower=0> or14;
real<lower=0> or23;
real<lower=0> or24;
real<lower=0> or34;
real md[N];
real mus[N];
real d[NT];
real ra[NT];
for (g in 2:NT) d[g] <- da[g];
for (z in 1:NT) ra[z] <- inv_logit(d[z]+0.5);
d[1] <- 0;
or12 <- exp(d[2]-d[1]);
```

```
    or13 <- exp(d[3]-d[1]);
    or14 <- exp(d[4]-d[1]);
    or23 <- exp(d[3]-d[2]);
    or24 <- exp(d[4]-d[2]);
    or34 <- exp(d[4]-d[3]);
    for (h in 1:N) {
    md[h] <- d[t[h]] - d[b[h]];
    mus[h] <- mu[s[h]];   }    }
    model {
    for (i in 1:NS)  mu[i] ~ normal(0,10);
    for (j in 2:NT) da[j] ~ normal(0,10);
    for (k in 1:N) delta[k] ~ normal(md[k],sdp);
    sdp ~ normal(0, 1) T[0.00001,];
    for (l in 1:N)
    increment_log_prob(r[l]*log(inv_logit(mus[l] + delta[l]*teb[l]))
      +(n[l] - r[l])*log(1 - inv_logit(mus[l] + delta[l]*teb[l])));}"
    fit = stan(model_code = code, data=data,iter=13000,
                 chains=1, warmup = 3000, pars = c("or12","or13",
                "or14","or23","or24","or34","ra"))
    ss.sim = matrix(NA, 6, 4)
    rank.matrix = matrix(NA,10000,NT)
    best = matrix(NA,10000,NT)
##extract posterior samples
    post = extract(fit)
    or12 = post$or12
    or13 = post$or13
    or14 = post$or14
```

```
or23 = post$or23
or24 = post$or24
or34 = post$or34
ra = post$ra
rank.re = matrix(NA,4,NT)
for (l in 1:10000)
{ rank.matrix[l,] = 5-rank(ra[l,])
   best[l,] = (rank.matrix[l,]==1)*1   }
ss.sim[1,] = c(mean(or12),
  quantile(or12,0.025),quantile(or12,0.975),OR.AB)
ss.sim[2,] = c(mean(or13),
  quantile(or13,0.025),quantile(or13,0.975),OR.AC)
ss.sim[3,] = c(mean(or14),
  quantile(or14,0.025),quantile(or14,0.975),OR.AD)
ss.sim[4,] = c(mean(or23),
 quantile(or23,0.025),quantile(or23,0.975),OR.BC)
ss.sim[5,] = c(mean(or24),
 quantile(or24,0.025),quantile(or24,0.975),OR.BD)
ss.sim[6,] = c(mean(or34),
 quantile(or34,0.025),quantile(or34,0.975),OR.CD)
df = as.data.frame(ss.sim)
simout[[i]] <- t(apply(df, 1, function(row){ c(
    est = unname( row[1] ),
    cover = unname( row[2] <= row[4] && row[4] <= row[3] ),
    length = unname( row[3] - row[2] )   )    }))
rank.re[1,] = colMeans(best)
rank.re[2,] = colMeans(rank.matrix)
```

```
    rank.re[3,] = c(quantile(rank.matrix[,1],0.025),
     quantile(rank.matrix[,2],0.025),quantile(rank.matrix[,3],0.025),
     quantile(rank.matrix[,4],0.025))
    rank.re[4,] = c(quantile(rank.matrix[,1],0.975),
     quantile(rank.matrix[,2],0.975),quantile(rank.matrix[,3],0.975),
     quantile(rank.matrix[,4],0.975))
    simout.rank[[i]] = rank.re  }
simTogether <- arrange(melt(simout), Var1, Var2)
simTogether <- ddply(simTogether, c("Var1", "Var2"), function(df){
  mean(df$value)  })
## format and return
simTogether <- dcast(simTogether, Var1 ~ Var2 , value.var = "V1")
row.names(simTogether)<-c("1vs2","1vs3","1vs4","2vs3","2vs4","3vs4")
simTogether <- simTogether[,-1]
simTogether.rank <- arrange(melt(simout.rank), Var1, Var2)
simTogether.rank <- ddply(simTogether.rank, c("Var1", "Var2"),
function(df){ mean(df$value) })
simTogether.rank <- dcast(simTogether.rank,Var1~Var2,value.var="V1")
row.names(simTogether.rank) <- c("Best Prob","rank","lower rank",
               "upper rank")
simTogether.rank <- simTogether.rank[,-1]
list(simTogether, simTogether.rank) }
```

B.2 Poisson Model

```
################parameters in the R function#########################
##m: number of simulation
##mu: baseline log odds ratio
##N: number of outcomes
```

```
##NS: number of studies (N/2)
##NT: number of treatments
##n1: sample size in each study
##d.AB, d.AC, d.AD: log odds ratio
##sig: standard deviation of study
##cc: continuity correction
loop_rank<-function(m, mu, N, NS, NT, n1, d.AB, d.AD,
                                d.BC, d.CD, sig, seed, cc)
{ set.seed(seed)
   simout <- as.list(rep(NA, m))
   simout.rank = as.list(rep(NA,m))
  ##design matrix for loop geometry
  x<-matrix(NA, N, 4)
  x[,1]=1
  x[,2]=0
  x[,3]=0
  x[,4]=0
  x[2,2]=1
  x[4,2]=1
  x[6,2]=1
  x[8,2]=1
  x[10,2]=1
  x[12,4]=1
  x[14,4]=1
  x[16,4]=1
  x[18,4]=1
  x[20,4]=1
```

```
 x[21,2]=1
 x[23,2]=1
 x[25,2]=1
 x[27,2]=1
 x[29,2]=1
 x[22,3]=1
 x[24,3]=1
 x[26,3]=1
 x[28,3]=1
 x[30,3]=1
 x[31,3]=1
 x[33,3]=1
 x[35,3]=1
 x[37,3]=1
 x[39,3]=1
 x[32,4]=1
 x[34,4]=1
 x[36,4]=1
 x[38,4]=1
 x[40,4]=1
 n=rep(n1, N)
 r = vector("numeric", N)
##risk rate for two treatments
 R.AC=exp(d.BC+d.AB)
 R.BD=exp(d.CD+d.BC)
 R.AB=exp(d.AB)
 R.AD=exp(d.AD)
```

```
  R.BC=exp(d.BC)
  R.CD=exp(d.CD)
 ##data generation
  for (i in 1:m){
    delta.AB=rnorm(NS, d.AB, sig)
    delta.AD=rnorm(NS, d.AD, sig)
    delta.AC=rnorm(NS, (d.BC+d.AB), sig)
    s=c(1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9,10,10,11,11,12,12,
         13,13,14,14,15,15,16,16,17,17,18,18,19,19,20,20)
    t=c(1,2,1,2,1,2,1,2,1,2,1,4,1,4,1,4,1,4,1,4,2,3,2,3,
         2,3,2,3,2,3,3,4,3,4,3,4,3,4,3,4)
    b=c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,2,2,
         2,2,2,2,2,2,3,3,3,3,3,3,3,3,3,3)
    teb = 1 - 1*(t==b)
    for(j in 1:NS)
    {r[2*j-1] = rpois(1,n[2*j-1]*(exp((x[(2*j-1),] %*%
       c(mu,delta.AB[s[2*j-1]],delta.AC[s[2*j-1]],delta.AD[s[2*j-1]])))))
     r[2*j] = rpois(1,n[2*j]*(exp((x[2*j,] %*%
       c(mu,delta.AB[s[2*j]],delta.AC[s[2*j]],delta.AD[s[2*j]])))))
##continuity correction
      if(r[2*j-1]==0||r[2*j]==0)
      {   r[2*j-1] = r[2*j-1]+cc
           r[2*j] = r[2*j]+cc   }   }
##Stan code begins bere
    data = list(N=N, NS=NS, NT=NT, n=n, t=t, b=b, teb=teb, r=r, s=s)
    code = "data {
    int<lower=0> N;
```

```
int<lower=0> NS;
int<lower=0> NT;
int<lower=0> n[N];
real<lower=0> r[N];
int<lower=0> s[N];
int<lower=0> t[N];
int<lower=0> b[N];
int<lower=0> teb[N];  }
parameters {
real mu[NS];
real da[NT];
real delta[N];
real<lower=0> sdp;    }
transformed parameters {
real<lower=0> r12;
real<lower=0> r14;
real<lower=0> r23;
real<lower=0> r34;
real<lower=0> r13;
real<lower=0> r24;
real md[N];
real mus[N];
real d[NT];
real ra[NT];
for (g in 2:NT) d[g] <- da[g];
for (z in 1:NT) ra[z] <- exp(d[z]+0.5);
d[1] <- 0;
```

```
    r12 <- exp(d[2]-d[1]);
    r14 <- exp(d[4]-d[1]);
    r23 <- exp(d[3]-d[2]);
    r34 <- exp(d[4]-d[3]);
    r13 <- exp(d[3]-d[1]);
    r24 <- exp(d[4]-d[2]);
    for (h in 1:N) {
    md[h] <- d[t[h]] - d[b[h]];
    mus[h] <- mu[s[h]];  }  }
    model {
    for (i in 1:NS)  mu[i] ~ normal(0,10);
    for (j in 2:NT) da[j] ~ normal(0,10);
    for (k in 1:N) delta[k] ~ normal(md[k],sdp);
    sdp ~ normal(0, 1) T[0.00001,];
    for (l in 1:N)
     increment_log_prob(r[l] * log(n[l]*exp(mus[l] + delta[l] * teb[l]))
        -n[l]*exp(mus[l] + delta[l] * teb[l]));  }  "
    fit = stan(model_code = code, data=data,iter=20000, chains=4,
         cores = getOption("mc.cores", 4),warmup = 10000,
         pars = c("r12","r14","r23","r34","r13","r24","ra"))
    ss.sim = matrix(NA, 6, 4)
    rank.matrix = matrix(NA,10000,NT)
    best = matrix(NA,10000,NT)
##extract posterior samples
    post = extract(fit)
    or12 = post$r12
    or14 = post$r14
```

```
or23 = post$r23
or34 = post$r34
or13 = post$r13
or24 = post$r24
ra = post$ra
rank.re = matrix(NA,4,NT)
for (l in 1:10000)
{ rank.matrix[l,] = 5-rank(ra[l,])
   best[l,] = (rank.matrix[l,]==1)*1   }
ss.sim[1,] = c(mean(or12),
            quantile(or12,0.025),quantile(or12,0.975),R.AB)
ss.sim[2,] = c(mean(or14),
            quantile(or14,0.025),quantile(or14,0.975),R.AD)
ss.sim[3,] = c(mean(or23),
            quantile(or23,0.025),quantile(or23,0.975),R.BC)
ss.sim[4,] = c(mean(or34),
            quantile(or34,0.025),quantile(or34,0.975),R.CD)
ss.sim[5,] = c(mean(or13),
            quantile(or13,0.025),quantile(or13,0.975),R.AC)
ss.sim[6,] = c(mean(or24),
            quantile(or24,0.025),quantile(or24,0.975),R.BD)
 df = as.data.frame(ss.sim)
simout[[i]] <- t(apply(df, 1, function(row){  c(
    est = unname( row[1] ),
    cover = unname( row[2] <= row[4] && row[4] <= row[3] ),
    length = unname( row[3] - row[2] )   )  }))
rank.re[1,] = colMeans(best)
```

```
    rank.re[2,] = colMeans(rank.matrix)
    rank.re[3,] = c(quantile(rank.matrix[,1],0.025),
     quantile(rank.matrix[,2],0.025),quantile(rank.matrix[,3],0.025),
     quantile(rank.matrix[,4],0.025))
    rank.re[4,] = c(quantile(rank.matrix[,1],0.975),
     quantile(rank.matrix[,2],0.975),quantile(rank.matrix[,3],0.975),
     quantile(rank.matrix[,4],0.975))
    simout.rank[[i]] = rank.re  }
  simTogether <- arrange(melt(simout), Var1, Var2)
  simTogether <- ddply(simTogether, c("Var1", "Var2"), function(df){
    mean(df$value)  })
## format and return
  simTogether <- dcast(simTogether, Var1 ~ Var2 , value.var = "V1")
  row.names(simTogether)<-c("1vs2","1vs4","2vs3","3vs4","1vs3","2vs4")
  simTogether <- simTogether[,-1]
  simTogether.rank <- arrange(melt(simout.rank), Var1, Var2)
  simTogether.rank <- ddply(simTogether.rank, c("Var1", "Var2"),
    function(df){ mean(df$value) })
  simTogether.rank <- dcast(simTogether.rank,Var1~Var2,value.var="V1")
  row.names(simTogether.rank) <- c("Best Prob","rank",
                                   "lower rank","upper rank")
  simTogether.rank <- simTogether.rank[,-1]
  list(simTogether, simTogether.rank)  }
```

APPENDIX C

R and BUGS Programs for Network Meta-analysis with Misclassification

In this appendix, we provide R and BUGS programs that implemented our model in Chapter Four. As we discussed in Chapter Five, only star geometry was considered here, but the other geometries could be extended by slightly modifying these programs.

```
################parameters in the R function#########################
##m: number of simulation
##mu: baseline log odds ratio
##N: number of outcomes
##NS: number of studies (N/2)
##NT: number of treatments
##n: sample size in each study
##d.AB, d.AC, d.AD: log odds ratio
##sig: standard deviation of study
##se, sp: sensitivity and specificity
metamisc = function(m,N,NS,NT,n,mu,d.AB,d.AC,d.AD,se,sp,seed,sig)
{ set.seed(seed)
  simout <- as.list(rep(NA, m))
  simout.rank = as.list(rep(NA,m))
  ##odds ratios
  OR.BC=exp(d.AC-d.AB)
  OR.BD=exp(d.AD-d.AB)
  OR.CD=exp(d.AD-d.AC)
  OR.AB=exp(d.AB)
```

```
OR.AC=exp(d.AC)

OR.AD=exp(d.AD)

##design matrix for star geometry

x<-matrix(NA, N, NT)

x[,1]=1

x[,2]=0

x[,3]=0

x[,4]=0

x[2,2]=1

x[4,2]=1

x[6,2]=1

x[8,2]=1

x[10,2]=1

x[12,3]=1

x[14,3]=1

x[16,3]=1

x[18,3]=1

x[20,3]=1

x[22,4]=1

x[24,4]=1

x[26,4]=1

x[28,4]=1

x[30,4]=1

n=rep(n, N)

##data generation

for (i in 1:m)

{ delta.AB = rnorm(NS, d.AB, sig)
```

```
  delta.AC = rnorm(NS, d.AC, sig)
  delta.AD = rnorm(NS, d.AD, sig)
  s=c(1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9,10,10,
         11,11,12,12,13,13,14,14,15,15)
  t=c(1,2,1,2,1,2,1,2,1,2,1,3,1,3,1,3,1,3,1,3,
        1,4,1,4,1,4,1,4,1,4)
  b=c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
        1,1,1,1,1,1,1,1,1,1)
  teb = 1 - 1*(t==b)
  r = vector("numeric", N)
  p = vector("numeric", N)
  p.t = vector("numeric", N)
  for(j in 1:N)
  { p[j] = 1/(1+exp(-(x[j,] %*%
         c(mu, delta.AB[s[j]], delta.AC[s[j]], delta.AD[s[j]]))))
##incorporate se and sp
    p.t[j] = p[j]*se+(1-p[j])*(1-sp)
    r[j] = rbinom(1,n[j],p.t[j])    }
  parameters <- list("or", "sd")
  data <- list("N", "NS", "NT", "n", "r", "s", "t", "b", "se", "sp")
  inits <- list(d=c(NA,0,0,0),mu=rep(0,15), sd=.6)
 ## call WinBUGS in R
  ss.sim <- bugs(
    data = data, inits = list(inits),
    parameters.to.save = parameters,
    model.file = "binom misc.txt",
    n.chains = 1, n.burnin = 20000, n.iter = 500000,
```

```
            n.thin = 25, bugs.directory="c:/WinBUGS/WinBUGS14/")
     ## collect the results from the WinBUGS
    df <- as.data.frame(ss.sim$summary)
    df$should <- c(OR.AB, OR.AC, OR.AD, OR.BC,OR.BD, OR.CD, sig)
    simout[[i]] <- t(apply(df, 1, function(row){  c(
        est = unname( row["mean"] ),
        cover = unname( row["2.5%"] <= row["should"]
                    && row["should"] <= row["97.5%"] ),
        length = unname( row["97.5%"] - row["2.5%"] ) ) }))  }
 ##format and return
 simTogether <- arrange(melt(simout), Var1, Var2)
  simTogether <- ddply(simTogether, c("Var1", "Var2"),
function(df){mean(df$value)  })
  simTogether <- dcast(simTogether, Var1 ~ Var2 , value.var = "V1")
  row.names(simTogether) <- simTogether$Var1
  simTogether <- simTogether[,-1]
  simTogether }
#####WinBUGS code for network meta-analysis with misclassification#####
model{
for(i in 1:N)  {
logit(p.f[i])<-mu[s[i]]+ delta[i]  * (1-equals(t[i],b[i]))
p[i] <- p.f[i]*se + (1-p.f[i])*(1-sp)
r[i]~dbin(p[i],n[i])
delta[i] ~ dnorm(md[i],tau)
md[i] <- d[t[i]] - d[b[i]]
rhat[i] <- p[i] * n[i]
dev[i] <- 2 * (r[i] * (log(r[i])-log(rhat[i]))+
```

```
(n[i]-r[i]) * (log(n[i]-r[i]) - log(n[i]-rhat[i]))  }
for(j in 1:NS){ mu[j]~dnorm(0,.01) }
totresdev <- sum(dev[])
d[1]<-0
for (k in 2:NT)  {d[k] ~ dnorm(0,.01) }
sd~dnorm(0,4)I(0.001,)
tau<-1/pow(sd,2)
for (c in 1:(NT-1))
   {for (k in (c+1):NT)
      { lor[c,k] <- d[k] - d[c]
         log(or[c,k]) <- lor[c,k] } } }
```

BIBLIOGRAPHY

Bagos, Pantelis G., and Georgios K. Nikolopoulos. 2009. "Mixed-effects Poisson Regression Models for Meta-analysis of Follow-up Studies with Constant or Varying Durations." *The International Journal of Biostatistics* 5(1).

Batterman, Stuart, Janet Burke, Vlad Isakov, Toby Lewis, Bhramar Mukherjee, and Thomas Robins. 2014. "A Comparison of Exposure Metrics for Traffic-related Air Pollutants: Application to Epidemiology Studies in Detroit, Michigan." *International Journal of Environmental Research and Public Health* 11(9):9553-9577.

Bhaumik, Anup Amatya, Sharon-Lise T. Normand, Joel Greenhouse, Eloise Kaizar, Brian Neelon, and Robert D. Gibbons. 2012. "Meta-analysis of Rare Binary Adverse Event Data." *Journal of the American Statistical Association* 107(498):555-567.

Böhning, Dankmar, Kalliopi Mylona, and Alan Kimber. 2015. "Meta-analysis of Clinical Trials with Rare Events." *Biometrical Journal* 57(4):633-648.

Bradburn, Michael J., Jonathan J. Deeks, Jesse A. Berlin, and A. Russell Localio. 2007. "Much Ado About Nothing: A Comparison of the Performance of Meta-analytical Methods with Rare Events." *Statistics in Medicine* 26(1):53-77.

Burstyn, Igor, Yunwen Yang, and A. Robert Schnatter. 2014. "Effects of Non-differential Exposure Misclassification on False Conclusions in Hypothesis-generating Studies." *International Journal of Environmental Research and Public Health* 11(10):10951-10966.

Carroll, Raymond J., David Ruppert, Leonard A. Stefanski, and Ciprian M. Crainiceanu. 2006. "Measurement Error in Nonlinear Models: A Modern Perspective." Boca Raton, FL: CRC press.

Chu, Rong. 2007. "Bayesian Adjustment for Exposure Misclassification in Case-control Studies." PhD Dissertation, Department of Statistics, The University of British Columbia, British Columbia.

Copeland, Karen T., Harvey Checkoway, Anthony J. McMichael, and Robert H. Holbrook. 1977. "Bias Due to Misclassification in the Estimation of Relative Risk." *American Journal of Epidemiology* 105(5):488-495.

Crowther, Michael J., Richard D. Riley, Jan A. Staessen, Jiguang Wang, Francois Gueyffer, and Paul C. Lambert. 2012. "Individual Patient Data Meta-analysis of Survival Data Using Poisson Regression Models." *BMC Medical Research Methodology* 12(1):1.

Edwards, Jessie K., Stephen R. Cole, Haitao Chu, Andrew F. Olshan, and David B. Richardson. 2014. "Accounting for Outcome Misclassification in Estimates of the Effect of Occupational Asbestos Exposure on Lung Cancer Death." *American Journal of Epidemiology* 179(5):641-647.

Edwards, Jessie K., Stephen R. Cole, Melissa A. Troester, and David B. Richardson. 2013. "Accounting for Misclassified Outcomes in Binary Regression Models Using Multiple Imputation with Internal Validation Data." *American Journal of Epidemiology* 177(9):904-912.

Gelman, Andrew. 2006. "Prior Distributions for Variance Parameters in Hierarchical Models (Comment on Article by Browne and Draper)." *Bayesian Analysis* 1(3):515-534.

Gorman, Emma, Alastair H. Leyland, Gerry McCartney, Ian R. White, Srinivasa Vittal Katikireddi, Lisa Rutherford, Lesley Graham, and Linsay Gray. 2014. "Assessing the Representativeness of Population-sampled Health Surveys through Linkage to Administrative Data on Alcohol-related Outcomes." *American Journal of Epidemiology* 180(9):941-948.

Greco, Teresa, Giovanni Landoni, Giuseppe Biondi-Zoccai, Fabrizio D'Ascenzo, and Alberto Zangrillo. 2013. "A Bayesian Network Meta-analysis for Binary Outcome: How to Do It." *Statistical Methods in Medical Research* 0(0):1-17.

Hinchliffe, Sally R., Keith R. Abrams, and Paul C. Lambert. 2013. "The Impact of Under- and Over-recording of Cancer on Death Certificates in a Competing Risks Analysis: A Simulation Study." *Cancer Epidemiology* 37(1):11-19.

Hunter, John E., and Frank L. Schmidt. 2000. "Fixed Effects vs. Random Effects Meta-analysis Models: Implications for Cumulative Research Knowledge." *International Journal of Selection and Assessment* 8(4):275-292.

Kibret, Taddele. 2014 "A Bayesian Network Meta-analysis for Binary Outcome: A Simulation Study." PhD Dissertation, Department of Mathematics & Statistics, McMaster University, Ontario

Koller, Daphne, and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT press.

Küchenhoff, Helmut, Samuel M. Mwalili, and Emmanuel Lesaffre. 2006. "A General Method for Dealing with Misclassification in Regression: The Misclassification SIMEX." *Biometrics* 62(1):85-96.

Lane, Peter W. 2013. "Meta-analysis of Incidence of Rare Events." *Statistical Methods in Medical Research* 22(2):117-132.

Luta, George, Melissa B. Ford, Melissa Bondy, Peter G. Shields, and James D. Stamey. 2013. "Bayesian Sensitivity Analysis Methods to Evaluate Bias Due to Misclassification and Missing Data using Informative Priors and External Validation Data." *Cancer Epidemiology* 37(2):121-126.

Milner, Allison, Stephen Morrell, and Anthony D. LaMontagne. 2014. "Economically Inactive, Unemployed and Employed Suicides in Australia by Age and Sex over a 10-year Period: What Was the Impact of the 2007 Economic Recession?." *International Journal of Epidemiology* 43(5):1500-1507.

Paulino, Carlos Daniel, Giovani Silva, and Jorge Alberto Achcar. 2005. "Bayesian Analysis of Correlated Misclassified Binary Data." *Computational Statistics & Data Analysis* 49(4):1120-1131.

Paulino, Carlos Daniel, Paulo Soares, and John Neuhaus. 2003. "Binomial Regression with Misclassification." *Biometrics* 59(3):670-675.

Pierce, Brandon L., and Tyler J. VanderWeele. 2012. "The Effect of Non-differential Measurement Error on Bias, Precision and Power in Mendelian Randomization Studies." *International Journal of Epidemiology* 41(5):1383-1393.

Prescott, Gordon J., and Paul H. Garthwaite. 2002. "A Simple Bayesian Analysis of Misclassified Binary Data with a Validation Substudy." *Biometrics* 58(2):454-458.

Salanti, Georgia, A. E. Ades, and John P. Ioannidis. 2011. "Graphical methods and numerical summaries for presenting results from multiple-treatment meta-analysis: an overview and tutorial." *Journal of clinical epidemiology* 64(2): 163-171.

Sankey, Steadman S., Lisa A. Weissfeld, Michael J. Fine, and Wishwa Kapoor. 1996. "An Assessment of the Use of the Continuity Correction for Sparse Data in Meta-analysis." *Communications in Statistics-Simulation and Computation* 25(4):1031-1056.

Shuster, Jonathan J. 2010. "Empirical vs. Natural Weighting in Random Effects Meta-analysis." *Statistics in Medicine* 29(12):1259-1265.

Spittal, Matthew J., Jane Pirkis, and Lyle C. Gurrin. 2015. "Meta-analysis of Incidence Rate Data in the Presence of Zero Events." *BMC Medical Research Methodology* 15(1):1.

Sposto, Richard, Dale L. Preston, Yukiko Shimizu, and Kiyohiko Mabuchi. 1992. "The Effect of Diagnostic Misclassification on Non-cancer and Cancer Mortality Dose Response in A-bomb Survivors." *Biometrics*48(2): 605-617.

Stamey, James D., Dean M. Young, and John W. Seaman. 2008. "A Bayesian Approach to Adjust for Diagnostic Misclassification between Two Mortality Causes in Poisson Regression." *Statistics in Medicine* 27(13):2440-2452.

Steenland, Kyle, and Sander Greenland. 2004. "Monte Carlo Sensitivity Analysis and Bayesian Analysis of Smoking as an Unmeasured Confounder in a Study of Silica and Lung Cancer." *American Journal of Epidemiology* 160(4):384-392.

Sutton, Alex J., and Keith R. Abrams. 2001. "Bayesian Methods in Meta-analysis and Evidence Synthesis." *Statistical Methods in Medical Research* 10(4):277-303.

Sweeting, Michael J., Alexander J. Sutton, and Paul C. Lambert. 2004. "What to Add to Nothing? Use and Avoidance of Continuity Corrections in Meta-analysis of Sparse Data." *Statistics in Medicine* 23(9):1351-1375.

Tu, Xin M., Litvak, E. and Pagano, M. 1994. "Screening tests: Can we get more by doing less?". *Statistics in Medicine* 13(19): 19051919.