

UNIVERSITY OF THE FREE STATE



UFS·UV

**Statistical Inference for  
Ectoparasiticide Efficacy in  
Animal trials**

by

Chandré Laverne Teise

(2009022779)

Mini-thesis submitted in partial fulfillment of the degree of  
Master's in Mathematical Statistics

in the

Faculty of Natural and Agricultural Sciences  
Department of Mathematical Statistics and Actuarial Science

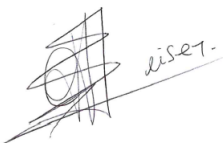
April 2018

Promoter: Dr. Divan Aristo Burger  
Co-Promoter: Dr. Janet van Niekerk  
Co-Promoter: Prof. Robert Schall



# Declaration of Authorship

I hereby declare that this work, submitted for the degree Magister Scientiae, at the University of the Free State, is my own original work and has not previously been submitted, for degree purposes or otherwise, to any other institution of higher learning. I further declare that all sources cited or quoted are indicated and acknowledged by means of a comprehensive list of references. Copyright hereby cedes to the University of the Free State.

Signed:  \_\_\_\_\_

Date: 6 April 2018

# *Abstract*

## **Statistical Inference for Ectoparasiticide Efficacy in Animal trials**

Chandré Laverne Teise  
(2009022779)

In controlled animal trials of ectoparasiticides the efficacy of treatments is estimated based on the number of surviving parasites with which experimental animals have been infected. Guidelines for the conduct and analysis of animal trials published by regulatory authorities require that the efficacy of the test treatment (as determined by the Abbott formula) should be at least 90%, for the test treatment to be declared efficacious. This decision rule, therefore, is simply based on a point estimate of efficacy and does not take into account the precision of the estimate; specifically, proper statistical inference on the efficacy of the test treatment in question is not required. As a consequence, the Type I error probability of falsely declaring a non-efficacious product to be efficacious can be overinflated. In the proposed research project we investigate the use of appropriate statistical decision rules for the efficacy which control the Type I error at a specified low level, say 5%. The statistical model for the data assumes a beta-binomial distribution which can accommodate the binomial overdispersion typically associated with such data. A Bayesian approach for implementing the analysis of ectoparasiticide efficacy data is explored.

# *Acknowledgments*

I am sincerely thankful to following individuals who made the completion of this study possible:

Dr. Divan Burger, leading supervisor, who worked through several preliminary versions of my script. His effort, patience, guidance, motivation, healthy criticism, and invaluable suggestions added considerably to the end result.

Dr. Janet van Niekerk, for her guidance, sharing of her Bayesian expertise, and for being dedicated to the role of secondary supervisor.

Prof. Robert Schall, for his expertise, comments and suggestions. It has been a great honor and privilege to have worked alongside him.

My family and friends, for their love, support, and constant encouragement.

# Contents

<b>Declaration of Authorship</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Acknowledgments</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem Statement . . . . .	1
1.2 Study Objectives . . . . .	2
1.3 Study Overview . . . . .	3
<b>2 Preliminary Theory</b>	<b>5</b>
2.1 Introduction . . . . .	5
2.2 Bayesian Statistics . . . . .	6
2.2.1 Prior Distributions . . . . .	7
2.2.1.1 Conjugate Priors . . . . .	8
2.2.1.2 Uninformative Priors . . . . .	8
2.2.1.3 Informative Priors . . . . .	8
2.2.2 Model Discrimination Statistics . . . . .	9
2.3 Linear Models . . . . .	10
2.3.1 Generalized Linear Models . . . . .	10
2.3.2 Hierarchical Generalized Linear Models . . . . .	12
2.4 Overdispersion . . . . .	13
2.5 Controlled Animal Studies for Efficacy . . . . .	14
2.5.1 Efficacy Point Estimates . . . . .	16
2.5.2 Limitations of Efficacy Estimation . . . . .	17
2.6 Framework for Efficacy Analysis . . . . .	19
2.6.1 Efficacy Evaluation . . . . .	20
2.7 Conclusion . . . . .	21

---

<b>3</b>	<b>Bayesian Model for Ectoparasiticide Efficacy</b>	<b>22</b>
3.1	Introduction . . . . .	22
3.2	Beta-Binomial Distribution . . . . .	23
3.2.1	Beta-Binomial Distribution in Animal Studies . . . . .	25
3.2.2	Beta-Binomial as Hierarchical Generalized Linear Model . . . . .	26
3.3	Integrated Nested Laplace Approximations . . . . .	28
3.3.1	Latent Gaussian Models . . . . .	28
3.3.2	Gaussian Approximation . . . . .	31
3.3.3	Methodology . . . . .	32
3.3.4	Model Discrimination Statistics . . . . .	35
3.3.4.1	Bayes Factors . . . . .	35
3.3.4.2	Deviance Information Criterion . . . . .	36
3.3.4.3	Predictive Integral Transform . . . . .	37
3.4	INLA of the Beta-Binomial Distributions . . . . .	38
3.5	Conclusion . . . . .	39
<b>4</b>	<b>Applications</b>	<b>40</b>
4.1	Introduction . . . . .	40
4.1.1	Generalized Linear Model Fit . . . . .	41
4.1.2	Fit of Bayesian HGLM by INLA . . . . .	41
4.2	Results and Discussion . . . . .	42
<b>5</b>	<b>Conclusion</b>	<b>50</b>
	<b>Bibliography</b>	<b>52</b>
<b>A</b>	<b>Programming Code</b>	<b>56</b>
A.1	SAS <sup>®</sup> Code . . . . .	56
A.2	R Code . . . . .	58

# Chapter 1

## Introduction

The aim of this chapter is to give a description of the research problem at hand, explain the scope of this study, and to give an outline of this mini dissertation.

### 1.1 Problem Statement

Ectoparasites are parasites that live on the surface of the hosts' body such as ticks, fleas, leeches and lice. Possible consequences of infestation may include blood loss, skin irritation, hypersensitivity, and transmission of pathogens. Ectoparasiticides might be used to provide short-term relief or long-term control against infestation. The term ectoparasiticides include insecticidal and acaracidal compounds (active substances that kill fleas and ticks, respectively), and insect growth regulators (IGRs) ([Marchiondo et al., 2013](#)). The response produced by a treatment against parasites can be referred to as the efficacy of the product.

Animal trials of parasiticides typically involve an untreated control (C), an investigational veterinary product (IVP; tests parasiticide, T) and a control veterinary

product (CVP; reference treatment, R). To establish the efficacy of an ectoparasiticide, studies are conducted that involve IVP and C treatments. In the case of a comparative evaluation of ectoparasiticide efficacy, each of the three treatments (C, T and R) are tested and compared in these studies (Schall and Luus, 2011).

To demonstrate the efficacy of a treatment, two types of animal studies must be conducted: A laboratory study followed by a clinical field study (CVMP, 2005).

During the laboratory study, several efficacy studies are conducted using experimentally induced animal infestations under controlled conditions. These studies are conducted to determine the ideal dose-rate of a test product, as well as to confirm the persistent and immediate efficacy of ectoparasiticide treatments. Subsequently, a clinical field evaluation is conducted to confirm the findings of the laboratory study. Such field studies are necessary due to many potential interactions which laboratory testing may not anticipate. Moreover, laboratory studies are seldomly able to adequately replicate the experimental conditions in which treatments (and animals) are being tested (Marchiondo et al., 2013).

For a treatment to be declared efficacious, guidelines published by international regulatory authorities require a minimum treatment efficacy of 90%. However, this criterion is simply based on point estimates that do not consider the standard error or precision of the estimates.

## 1.2 Study Objectives

The overall aim of this research project is to explore a statistical inference approach for the evaluation of ectoparasiticide efficacy. More specifically, the objectives of this research project are to:

1. Design an adequate model for ectoparasiticide efficacy evaluation (statistical inferences).
2. Propose a Bayesian approach that can accommodate small sample sizes and consider potential variation of survival probabilities across animals.

### 1.3 Study Overview

The content of this mini-thesis is summarized as follows:

A brief overview is given on the preliminary theory in Chapter 2. The literature on Bayesian statistics is reviewed in Section 2.2 as follows: The appropriate choice of prior distributions are outlined in Section 2.2.1, and the tools for model selection are provided in Section 2.2.2. The development of the conventional linear model is presented in Section 2.3 and includes the following: GLMs are discussed in Section 2.3.1, and hierarchical GLMs (HGLMs) are discussed in Section 2.3.2. A brief introduction to the concept of overdispersion is presented in Section 2.4. In Section 2.5 an overview of controlled animal studies of efficacy is presented. The literature includes the following: The conventional statistical evaluation of ectoparasiticide efficacy is outlined in Section 2.5.1, and the limitations associated with point estimation are discussed in Section 2.5.2. A statistical framework for efficacy evaluation using a GLM is presented in Section 2.6. Some chapter concluding remarks are presented in Section 2.7.

A literature review, and methods applied in this research project are presented in Chapter 3. Section 3.2 presents an introduction of the beta-binomial distribution. The implementation of the beta-binomial model in ectoparasiticide animal

trials is presented in Section 3.2.1. The beta-binomial model as a special case of an HGLM is described in Section 3.2.2. Section 3.3 presents an overview of integrated nested Laplace approximation (INLA) of the joint posterior distribution of model parameters, and furthermore includes the following: Latent Gaussian models (LGM) are discussed in Section 3.3.1, and the Gaussian approximation is outlined in Section 3.3.2. The steps for implementation of the INLA method are given in Section 3.3.3. Model evaluation statistics for the INLA method are introduced in Section 3.3.4. Further discussions on the beta-binomial model in the INLA framework are provided in Section 3.4. Some chapter concluding remarks are given in Section 3.5.

Chapter 4 provides applications of the GLM, conventional binomial and beta-binomial models to data of recently published animal studies. An overview of the SAS<sup>®</sup> procedure PROC GENMOD is presented in Section 4.1.1. Section 4.1.2 gives a summary of the INLA procedure (implemented in the R software). Section 4.2 summarizes the results of the application of the two procedures.

Chapter 5 presents some possible areas of future research, as well as final concluding remarks.

Appendix A presents the relevant SAS<sup>®</sup> and R code for the implementation of the statistical methods introduced in this research report.

# Chapter 2

## Preliminary Theory

### 2.1 Introduction

Statistics can be classified into two broad classes, namely frequentist and Bayesian statistics. The Bayesian approach considers all unknown parameters as random variables (therefore following probability distributions) conditional on the data. Conversely, the frequentist (“classical” or likelihood-based) approach considers the unknown parameters to be fixed ([McNeish, 2016](#)).

Within the likelihood-based framework, the method of maximum likelihood (ML) estimation is the most popular estimation method. One desirable property of ML estimation is that the parameter estimates are often unbiased, or at least consistent. Secondly, ML estimates exhibit asymptotic normality which is useful for statistical inferences. Both of these properties, however, rely on large sample sizes ([McNeish, 2016](#)). Consequently, the parameter estimates from ML estimation are often unreliable for small samples ([Fong et al., 2010](#)). To account for problems

associated with small sample sizes, alternatives for ML estimation have been proposed (e.g. the method of restricted maximum likelihood (REML)) (McNeish, 2016).

Sampling-based Markov Chain Monte Carlo (MCMC) methods are commonly used in Bayesian inference (Ntzoufras, 2009). Since MCMC methods do not rely on asymptotic approximations, Bayesian methods are more suitable to model small sample data. However, for small sample sizes the choice of a prior specification is vital. The latter is due to the limited information that is made available by the likelihood function of the data; priors could therefore play a key role in the contribution towards posterior distributions. A further advantage of Bayesian methods over “classical” approaches is that prior information can be incorporated (*via* a prior distribution) into the model, and may aid with accuracy of predictions. As a disadvantage of Bayesian approaches, the fact that the parameters are random implies that their interpretations are probabilistic, and may cause difficulty with hypothesis testing, model selection and model comparison (McNeish, 2016).

## 2.2 Bayesian Statistics

Bayesian inference is based on the process of applying prior information in the context of the Bayes theorem to observable events. Within this framework, all unknown parameters are considered as random variables which can be characterized by prior distributions (Blangiardo and Cameletti, 2015). The posterior distribution is of primary interest and can be obtained as follows:

Consider an independent, identically distributed (i.i.d.) sample of size  $n$ , and observed data  $\mathbf{y} = \{y_1, y_2, \dots, y_n\}$ , with joint probability distribution (Ntzoufras,

2009):

$$\pi(\mathbf{y}|\boldsymbol{\theta}) = \prod_{i=1}^n \pi(y_i|\boldsymbol{\theta}) \quad \text{for } i = 1, 2, \dots, n$$

where  $\boldsymbol{\theta}$  is a vector of model parameters. The joint probability distribution  $\pi(\mathbf{y}|\boldsymbol{\theta})$  is referred to as the likelihood of the model and summarizes the available information provided by the observed sample. Furthermore, let  $\pi(\boldsymbol{\theta})$  be the prior distribution for  $\boldsymbol{\theta}$ . Based on the Bayes theorem, the posterior distribution in question can be expressed as follows:

$$\begin{aligned} \pi(\boldsymbol{\theta}|\mathbf{y}) &= \frac{\pi(\mathbf{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\int \pi(\mathbf{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}} \\ &\propto \pi(\mathbf{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \end{aligned} \tag{2.1}$$

The term in the denominator of Equation (2.1) is called the normalizing constant (Ando, 2010). Note that the data  $\mathbf{y}$  enter the model only through the likelihood function (Ntzoufras, 2009). As previously indicated, Bayesian methods (opposed to classical approaches) take into account prior knowledge that could stem from operational data from past experiments, intuition or personal and expert belief. Bayesian inferences can also be appealing in cases where limited data are available (small sample sizes), provisional on the availability of sufficient prior information (Ando, 2010).

### 2.2.1 Prior Distributions

An important task in Bayesian modeling is the choice of prior distributions, as it is representative of the information that is available for the parameters of interest. In many cases, a “natural” candidate that can be used as a prior distribution is

obtainable ([Blangiardo and Cameletti, 2015](#)). The subsequent sections discuss a few commonly used priors.

### **2.2.1.1 Conjugate Priors**

A prior distribution can be classified as a conjugate prior, if both the posterior and prior distributions are members of the same probability distribution. The advantage of conjugate priors is that the analytical form of the posterior distribution is known ([Ando, 2010](#)). Consequently, the summary statistics of the associated posterior distribution (e.g. mean, median and Bayesian credibility intervals) and other quantities of interest, are easily obtainable. However, it is not always possible to implement conjugate priors in practice since not all likelihoods are accompanied by a conjugate prior distribution ([Blangiardo and Cameletti, 2015](#)).

### **2.2.1.2 Uninformative Priors**

Commonly used priors are uninformative (flat, or diffuse) priors. Uninformative priors are commonly defined as a constant, in which case the posterior distribution is simply a constant multiplied by the likelihood function ([Ando, 2010](#)). Such specification can be appealing in that they allow the data to “speak for itself”, in some sense.

### **2.2.1.3 Informative Priors**

Unlike uninformative priors, informative priors are constructed from prior knowledge, opinion, belief or intuition. Informative priors may have a large effect on the

posterior distribution, since they are not necessarily (significantly) dominated by the likelihood function (Ando, 2010).

## 2.2.2 Model Discrimination Statistics

Consider the following marginal likelihood:

$$\pi(\mathbf{y}|M) = \int \pi(\mathbf{y}, \theta_M|M)d\theta_M = \int \pi(\mathbf{y}|\theta_M, M)\pi(\theta_M|M)d\theta_M$$

where  $\theta_M$  is the set of parameters of the likelihood  $\pi(\mathbf{y}|\theta_M, M)$  under models  $M = \{m_1, m_2\}$ . Similarly,  $\pi(\theta_M|M)$  is the prior distribution of  $\theta_M$  under model  $M$ . The ratio  $\pi(\mathbf{y}|m_1)/\pi(\mathbf{y}|m_2)$  is defined as the Bayes factor of models  $m_1$  versus  $m_2$  (Ntzoufras, 2009). Bayes factors are commonly used as a quantity for comparing models which are in favor of the model  $M$  with the largest marginal likelihood (Ando, 2010).

The Akaike information criterion (AIC) is computed as  $-2$  times the maximum value of the likelihood plus twice the number of parameters estimated by the model. Spiegelhalter et al. (2002) proposed the deviance information criterion (DIC) as an extension of the AIC (as cited in Ando 2010). The DIC has the following structure:

$$\text{DIC} = D(\bar{\theta}_M, M) + 2p_M$$

where  $p_M$  is the number of “effective” parameters. The quantity  $D(\bar{\theta}_M, M)$  in (2.2.2) is called the deviance (defined as twice the difference of the maximum attainable log-likelihood and the log-likelihood of the fitted model) of the posterior mean of the parameters (Myer et al., 2010). For one-stage models the values for AIC and DIC are in some cases similar. Differences between AIC and DIC are

evident with hierarchical models, where the DIC uses the number of “effective” parameters as a substitute of the actual number of parameters of the model (Ntzoufras, 2009). The DIC is mostly used as a tool for model selection, in which case smaller values of DICs may indicate an improved model fit to the data (Ando, 2010).

## 2.3 Linear Models

Linear regression models are models in which a set of observed data  $y_1, y_2, \dots, y_n$  can be modeled by a linear combination of explanatory variables,  $x_{i1}, x_{i2}, \dots, x_{ip}$ . An ordinary linear model takes on the form:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_k x_{ik} + \dots + \beta_p x_{ip} + \epsilon_i \quad (2.2)$$

for  $i = 1, 2, \dots, n$  and  $k = 1, \dots, p$ . The  $\epsilon_i$  term represents the model residuals, and  $\beta_k$  the regression coefficients of the linear model. The linear model often assumes normally distributed residuals with a mean of zero, and for which the expected value of the observations is modeled by a linear combination of the unknown parameters  $\beta_k$ . The unknown parameters  $\beta_k$  are typically estimated using least squares and maximum likelihood estimation.

### 2.3.1 Generalized Linear Models

One of the most widely used types of models in statistics is the generalized linear model (GLM) which is an extension of the conventional linear model (McCullagh and Nelder, 1989). The popularity of GLMs is due to their generality and wide

range of applications (Ntzoufras, 2009). Within the GLM class, the observations are assumed to be independent of one another and that the data are not clustered (McCullagh and Nelder, 1989). GLMs can be used to model non-normal data such as discrete or binary data. A significant unifying concept underlying GLMs is the exponential family of distributions (Myer et al., 2010). Here one assumes that the observed responses  $y_i$  follow distributions within the exponential family that can take on the canonical form:

$$f(y_i; \theta_i, \phi) = \exp \left\{ \frac{y_i \theta_i - a(\theta_i)}{b(\phi)} + c(y_i, \phi) \right\} \quad (2.3)$$

where  $a(\theta_i)$ ,  $b(\phi)$  and  $c(y_i, \phi)$  are specific functions (McCullagh and Nelder, 1989). Here, the parameters  $\theta_i$  are referred to as the natural location parameters and  $\phi$  is called the dispersion parameter. The researcher must accordingly deal with the extra parameter ( $\phi$ ) in the presence of overdispersion or underdispersion (Myer et al., 2010). The exponential family of distributions includes, among others, the normal, Poisson, binomial, gamma and inverse Gaussian distributions (Myer et al., 2010).

A GLM consists of the following three-part specification (McCullagh and Nelder, 1989):

1. Error structure (or distribution):

The  $y_i$  are assumed to be independent (uncorrelated) with corresponding means  $\mu_i$ , and share a common distribution within the exponential family of distributions.

2. Linear predictor:

A set of predictor variables  $x_{ik}$ , intercept  $\beta_0$  and unknown parameters  $\beta_k$

such that:

$$\eta_i = \beta_0 + \sum_{k=1}^p x_{ik}\beta_k \quad (2.4)$$

3. Link function:

A monotonic and differentiable function  $g$  which connects the mean of the responses to the linear predictor in the following way:

$$g(\mu_i) = \eta_i = \beta_0 + \sum_{k=1}^p x_{ik}\beta_k \quad (2.5)$$

where  $\mu_i = E(y_i)$  and  $x_{ik}$  is the  $k^{\text{th}}$  covariate of observation  $i$ .

### 2.3.2 Hierarchical Generalized Linear Models

[Lee and Nelder \(1996\)](#) introduced a hierarchical generalized linear model (HGLM) as an extension of generalized linear mixed models (GLMMs). This class of models allows one or more random components (additional to the fixed effects) in the linear predictor of a GLM. HGLMs differ from GLMMs in the sense that the random effects are from a conjugate exponential family of distribution that are not confined only to the normal distribution ([Myer et al., 2010](#)). Introducing random effect terms allows one to model hierarchically organized data (e.g. clusters or repeated measures ([Kassahun, 2014](#))), and to account for overdispersion that may be present in the data ([Walker and Mallick, 1997](#)).

An HGLM can be defined as follows: Let  $y_1, y_2, \dots, y_i, \dots, y_n$  represent the observed data and  $u_i$  the unobserved univariate random effect. Assuming a distribution for  $u_i$ , conjugate to the exponential family,  $y_i|u_i$  accordingly follow distributions

(within the exponential family) which satisfy the following:

$$E(y_i|u_i) = \mu_i^* \quad \text{and} \quad \text{Var}(y_i|u_i) = \phi_i V(\mu_i^*) \quad (2.6)$$

The link function of an HGLM with random component  $u_i$  is given by:

$$\begin{aligned} g(\mu_i^*) &= \eta_i^* = \eta_i + \nu(u_i) \\ &= \beta_0 + \sum_{k=1}^p x_{ik} \beta_k + \nu(u_i) \end{aligned} \quad (2.7)$$

where  $\nu(u_i)$  are strictly monotonic functions of the random component  $u_i$  (Lee and Nelder, 1996). In addition, Lee and Nelder (1996) noticed that the distributions used for  $\nu_i$  in conjugate HGLMs have similar properties as that of the normal distribution (e.g. continuous, unimodal, or symmetrically distributed). Analyses using HGLMs are therefore often similar to those of GLMMs.

## 2.4 Overdispersion

Overdispersion occurs when the variance of the data  $y_i$  appears to be greater than what is expected under some reference model (McCullagh and Nelder, 1989). Say  $y_i$  follow binomial distributions with sample sizes  $n$  and probability parameter  $\pi$ , then the observed variability of the data cannot always be fully accounted for by the theoretical variance of the binomial distribution, namely  $n\pi(1 - \pi)$ . For example, Myer et al. (2010) introduced an additional parameter so that the variance of the data is explained by  $n\pi(1 - \pi)\phi$ . Although it may be difficult to locate the precise cause (or underlying process) of overdispersion, some common causes include clustered sampling, and the presence of correlation between observations

and omitted unobserved variables. Consequently, the standard errors (of model parameters) obtained from the assumed model may be inappropriate, and may lead to incorrect conclusions concerning the statistical significance of regression parameters (Hinde and Demétrio, 1998). When analyzing binary or count data, it is good practice to check for overdispersion before carrying out analyses of the data. Consider the Pearson chi-square statistic:

$$X^2 = \sum_i \frac{(O_i - E_i)^2}{E_i}$$

where  $O_i$  and  $E_i$  respectively denote the observed and fitted values. The Pearson chi-square statistic together with the deviance (see Section 2.2.2) can be used to estimate the overdispersion parameter of the exponential family of distributions. Values for the ratio of the deviance, divided by its associated degrees of freedom, larger than 1 indicate that overdispersion may be present in the data (SAS Institute Inc., 2008).

## 2.5 Controlled Animal Studies for Efficacy

A controlled animal study is recommended to assess the efficacy of a parasiticide against flea and tick infestations and involves at least two treatment groups, namely an investigational veterinary product (IVP) and a control (C) (Schall et al., 2016). In such studies, the study subjects are randomized to the different treatment groups, which typically consist of 8 to 10 animals per group (Schall and Luus, 2011). Thereafter, the efficacy of the test treatment is determined (under standardized conditions) by comparing the number of “alive” parasites of the treated

and untreated groups (Marchiondo et al., 2013). A typical trial schedule for tick and flea infestations of cats or dogs is as follows.

## Tick Studies

After an acclimatization period of at least 7 days, a pre-allocation infestation is conducted on Day -7 to establish the infestation rate. Thereafter the animals are ranked according to the total number of attached parasites on each experimental animal. Subsequently, the animals are randomized to the different study treatments involved (Marchiondo et al., 2013). The “actual” infestation of 50 unfed adult ticks occurs on Day -2 followed by an application of the randomized test substance on Day 0 (Schall and Luus, 2011). To assess the immediate efficacy of a test treatment, an *in situ* efficacy test is conducted 48 hours after application (CVMP, 2016). To test the therapeutic efficacy for up to 4 weeks (i.e. short-term efficacy) (Marchiondo et al., 2013), treatments with a claimed protective effect for up to 4 weeks are used (e.g. shampoo and spray). This process requires weekly re-infestations of ticks followed by an efficacy test 48 hours after each challenge (CVMP, 2016). To test the therapeutic efficacy for up to 3 months (i.e. long-term efficacy) (Marchiondo et al., 2013) of the test product, treatments with a claimed protective effect for more than 4 weeks are used (e.g. collars and tablets). In the latter case, the animals should be re-infested every 4 weeks over the period of claimed protection, reducing to bi-weekly re-infestations during the last month of expected protection. An *in situ* efficacy testing follows the re-infestation 48 hours after each challenge (CVMP, 2016).

## Flea Studies

Following an acclimatization period, the experimental animals are infested to assess their ability to sustain a flea population. On Day -1 the “actual” infestation of 50 to 100 unfed adult fleas occurs, and on Day 0 the randomized test substance is administered. To test the immediate efficacy of the test product, an *in situ* efficacy test is conducted 24 hours after the treatment has been administered (CVMP, 2016). Similar to the tick studies, in order to test the therapeutic efficacy for up to 4 weeks (i.e. short-term efficacy) (Marchiondo et al., 2013), treatments with a claimed protective effect for up to 4 weeks are used. In the latter case, the animals undergo weekly re-infestations over the period of claimed protection. The *in situ* efficacy testing follows the re-infestation 24 hours after each challenge (CVMP, 2016). To test the therapeutic efficacy for up to 3 months (i.e. long-term efficacy) (Marchiondo et al., 2013) of the test product, treatments with a claimed protective effect for more than 4 weeks are used. The animals are re-infested every 4 weeks over the period of claimed protection, followed by *in situ* efficacy testing 24 hours after each challenge (CVMP, 2016).

### 2.5.1 Efficacy Point Estimates

Abbott et al. (1925) proposed the following formula to calculate the efficacy of an IVP:

$$E_T(\%) = \frac{m_C - m_T}{m_C} \times 100 \quad (2.8)$$

where  $m_C$  is the mean number of “live” parasites in the control group (C), and  $m_T$  is the mean number of “live” parasites in the test group (T). For a label claim, the minimum required efficacy of an IVP should be at least 90% for ticks, and

95% for fleas (Marchiondo et al., 2013).

The efficacy (2.8) can be based on estimators such as arithmetic or geometric means. The geometric mean is considered to be a more reliable estimate of central tendency for comparison, applicable to cases where the distribution of the surviving parasite counts within each group is skewed (Marchiondo et al., 2013). In addition, a geometric mean of a right skewed sample is smaller than the corresponding arithmetic mean. In such cases, the Abbott estimator based on the geometric mean may yield higher efficacy estimates than the corresponding arithmetic mean estimates. The Abbott estimator based on the arithmetic mean is therefore often perceived to yield lower efficacy estimates (Schall et al., 2016). However, there is an alternative view that the arithmetic mean is the only appropriate tool for estimating the efficacy of an ectoparasiticide (Marchiondo et al., 2013). It is therefore good practice to record the Abbott estimates based on both arithmetic and geometric means. Alternatively, Schall et al. (2016) considered an efficacy estimator based on the median number of surviving parasites, and a maximum likelihood estimator. Schall et al. (2016) showed that maximum likelihood estimation of efficacy produces results that are similar to those of the arithmetic mean estimator. The results also revealed that the estimator based on geometric means is potentially biased upwards. For these reasons, Schall et al. (2016) suggested that arithmetic means should rather be used when estimating the efficacy of ectoparasiticides.

### 2.5.2 Limitations of Efficacy Estimation

Schall and Luus (2011) discussed the potential limitations when comparing the empirical efficacy  $E_T$  with an absolute standard of 90% to 95%. Firstly, when

$E_T$  is solely based on an observed quantity there is no indication of precision with which the efficacy is estimated (e.g. standard errors or confidence intervals). Secondly, as per expression in Equation (2.8), there is no statistical decision rule to test the statistical significance of the observed efficacy. Let  $\varepsilon_T$  represent the true efficacy of the test treatment as follows (Schall and Luus, 2011):

$$\varepsilon_T = \left(1 - \frac{\pi_T}{\pi_C}\right) \times 100 \quad (2.9)$$

where  $\pi_j$  is the proportion of surviving parasites of treatment  $j = C, T$ . The point estimate for  $\pi_j$  can be calculated as  $p_j = \frac{\sum_{i=1}^N y_{ij}}{kN}$ , where  $y_{ij}$  denotes the number of surviving parasites on animal  $i$  assigned to treatment  $j = C, T$ ,  $N$  the number of animals assigned to each of the treatments ( $i = 1, 2, \dots, N$ ), and  $k$  is the number of parasites on animal  $i$ . Consequently, the empirical efficacy of a test treatment can also be expressed as follows:

$$E_T = \frac{p_C - p_T}{p_C} \times 100 \quad (2.10)$$

However, when comparing the test treatment (T) against an untreated control (C) the conventional hypotheses are as follows:

$$H_0 : \pi_C - \pi_T = 0 \quad \text{against} \quad H_A : \pi_C - \pi_T \neq 0 \quad (2.11)$$

at a two-sided significance level of  $\alpha = 5\%$ .

Dividing both sides of the hypotheses (Equation (2.11)) by  $\pi_C$  results in (analogous to Equation (2.9)):

$$H_0 : \varepsilon_T = 0 \quad \text{against} \quad H_A : \varepsilon_T \neq 0$$

Consequently, by rejecting the null-hypothesis of no difference as in Equation (2.11) implies that the test treatment (T) has an efficacy that is different from zero. Hence, a significant difference between the two groups does not necessarily guarantee that the test treatment meets the minimum requirements for a label claim (say  $\varepsilon_T > 90\%$ ).

## 2.6 Framework for Efficacy Analysis

For the animal study described in Section 2.5, Schall and Luus (2011) proposed the following statistical model:

If each experimental animal  $i$  ( $i = 1, 2, \dots, N$ ) in a given treatment group  $j$  is infested with  $k$  parasites, then the number of surviving parasites on animal  $i$  in group  $j$  has a binomial distribution, denoted by  $y_{ij} \sim \text{Bin}(\pi_{ij}, k)$ . Schall and Luus (2011) assumed that:

$$E(\pi_{ij}) = \pi_j \quad \text{and} \quad \text{Var}(\pi_{ij}) = \pi_j(1 - \pi_j)\tau^2 \quad \text{for } j = \text{C,R,T}$$

The total number of surviving parasites in each group,  $y_j = \sum_{i=1}^N y_{ij}$ , therefore has the following unconditional mean and variance (Schall and Luus, 2011):

$$E(y_j) = m\pi_j \quad \text{and} \quad \text{Var}(y_j) = m\phi\pi_j(1 - \pi_j)$$

where  $m = kN$  and the dispersion parameter  $\phi = 1 + \tau^2(k - 1)$  may be regarded as the “extra-binomial” variation (due to clustering). Therefore, it follows that  $y_j$  can be modeled using a binomial type of distribution that accounts for overdispersion (Schall and Luus, 2011).

### 2.6.1 Efficacy Evaluation

Recall from Section 2.5.2 that the conventional null-hypothesis only suggests that the efficacy of a test substance (T) is different from zero (Schall and Luus, 2011). Moreover, the hypothesis does not guarantee that the efficacy meets the minimum requirement for a label claim of e.g. 90%. Consequently, Schall and Luus (2011) proposed a “shifted” null-hypothesis of the following form:

$$H_0 : \varepsilon_T < \Lambda \quad \text{against} \quad H_A : \varepsilon_T \geq \Lambda \quad (2.12)$$

at a one-sided significance level of  $\alpha = 2.5\%$  and where  $\Lambda$  is the required minimum level of efficacy.

A two-sided confidence interval (CI) to test  $H_0$  in Equation (2.12) can be obtained by fitting a model of the following form (Schall and Luus, 2011):

$$\log(\pi_j) = \beta_0 + \beta_{treat(j)} \quad \text{for} \quad j = C, T \quad (2.13)$$

Schall and Luus (2011) assumed a binomial distribution to the counts of surviving parasites  $y_{ij}$ . The log-link function was specified for the GLM, and the Pearson chi-square statistic was used to estimate the dispersion parameter  $\phi$ . A 95% CI was constructed for the quantity  $\log(\pi_T) - \log(\pi_C) = \log(\pi_T/\pi_C)$ . Suppose that L and U are the lower and the upper limits of the 95% CI for  $\log(\pi_T/\pi_C)$ , respectively, then the 95% CI for  $\varepsilon_T$  is (Schall and Luus, 2011):

$$[(1 - \exp(U)) \times 100; (1 - \exp(L)) \times 100] \quad (2.14)$$

$H_0$  in Equation (2.12) is rejected if the lower limit of the acquired 95% CI is greater than  $\Lambda$ ; it is concluded that  $\varepsilon_T$  is at minimum  $\Lambda$ .

## 2.7 Conclusion

In the current chapter, we motivated the use of Bayesian statistics for small sample sizes. An introduction of Bayesian methods, and the specification of linear models were presented. The phenomenon of overdispersion, and ways of detecting its presence, were discussed. A brief summary of controlled animal studies of efficacy was provided, and included the limitations associated with the conventional methods of establishing the efficacy of ectoparasiticide treatments (that is, efficacy point estimation is conventionally not accompanied by statistical hypothesis testing). The chapter concludes with a summary of the current statistical framework: More specifically, the specification of a GLM for efficacy evaluation of ectoparasiticides (proposed by [Schall and Luus \(2011\)](#)).

# Chapter 3

## Bayesian Model for Ectoparasiticide Efficacy

### 3.1 Introduction

Bayesian models often do not provide closed form analytical expressions for quantities such as posterior means, and therefore, approximations are required. The conventional estimation method for Bayesian models is Markov Chain Monte Carlo (MCMC) sampling schemes. MCMC sampling is a flexible method for sampling from posterior distributions ([Holand et al., 2013](#)), but can be computationally cumbersome ([Blangiardo and Cameletti, 2015](#)). For example, MCMC sampling in some cases needs a hundred times more computational time to improve an estimate by at least one digit ([Martins and Rue, 2014](#)). Software such as BUGS, JAGS and BayesX are available for performing Bayesian inference using MCMC sampling.

As an alternative, [Rue et al. \(2009\)](#) introduced the integrated nested Laplace approximation (INLA) method for performing fully Bayesian inference for the class of latent Gaussian models (LGMs). INLA is based on direct numerical integration, and therefore not on simulations as in the case of MCMC sampling ([Holand et al., 2013](#)). INLA circumvents the need for simulations by using extensive Gaussian approximations that draw from the properties of LGMs ([Martins and Rue, 2014](#)). INLA has shown to outperform MCMC sampling in terms of both computational speed and accuracy (of estimation) ([Holand et al., 2013](#)). Nevertheless, it should be noted that the computational cost associated with INLA increases exponentially as the number of hyperparameters in the model increases ([Rue et al., 2009](#)). The R package that implements the INLA approach, R-INLA, is available from [www.r-inla.org](http://www.r-inla.org).

## 3.2 Beta-Binomial Distribution

The beta-binomial distribution is a natural extension of the binomial distribution, and is indexed (or described) by two parameters. The compound distribution is obtained by assuming a beta prior for the probability parameter of a finite sum of Bernoulli random variables ([Najera-Zuloaga et al., 2017](#)). The beta-binomial distribution for the present problem can be specified as follows: Suppose that each animal  $i$  ( $i = 1, 2, \dots, N$ ) in a given treatment group  $j$  is infected with  $k$  parasites, then  $y_{ij}$  represents the number of surviving parasites on the  $i^{\text{th}}$  animal in the  $j^{\text{th}}$  group. Consequently,  $y_{ij}|p_{ij} \sim \text{Bin}(k, p_{ij})$  where  $p_{ij}$  are random and assigned a conjugate beta distribution. That is,  $p_{ij} \sim \text{Beta}(a_j, b_j)$ . The marginal density of

$y_{ij}$  is given by (Martinez et al., n.d.):

$$\begin{aligned} \text{BB}(y_{ij}|k, a_j, b_j) &= \int_0^1 \text{Bin}(y_{ij}|k, p_{ij})\text{Beta}(p_{ij}|a_j, b_j)dp_{ij} \\ &= \binom{k}{y_{ij}} \frac{\Gamma(a_j + b_j)\Gamma(a_j + y_{ij})\Gamma(b_j + k - y_{ij})}{\Gamma(a_j + b_j + k)\Gamma(a_j)\Gamma(b_j)} \end{aligned} \quad (3.1)$$

where  $B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$  is the beta function,  $p_{ij}$  are the proportions of a "success", and the hyperparameters  $a_j$  and  $b_j$  are both positive numbers (Najera-Zuloaga et al., 2017). The mean and variance of a beta-binomial random variable are given by (Martinez et al., n.d.):

$$\begin{aligned} \text{E}(y_{ij}|k, a_j, b_j) &= k \left( \frac{a_j}{a_j + b_j} \right) \quad \text{and} \\ \text{Var}(y_{ij}|k, a_j, b_j) &= \frac{ka_jb_j(a_j + b_j + k)}{(a_j + b_j)^2(a_j + b_j + 1)} \end{aligned} \quad (3.2)$$

Ennis and Bi (1998) suggest the use of the following parametrization:

$$\pi_j = \frac{a_j}{a_j + b_j} \quad \text{and} \quad \rho_j = \frac{\theta_j}{1 + \theta_j} \quad (3.3)$$

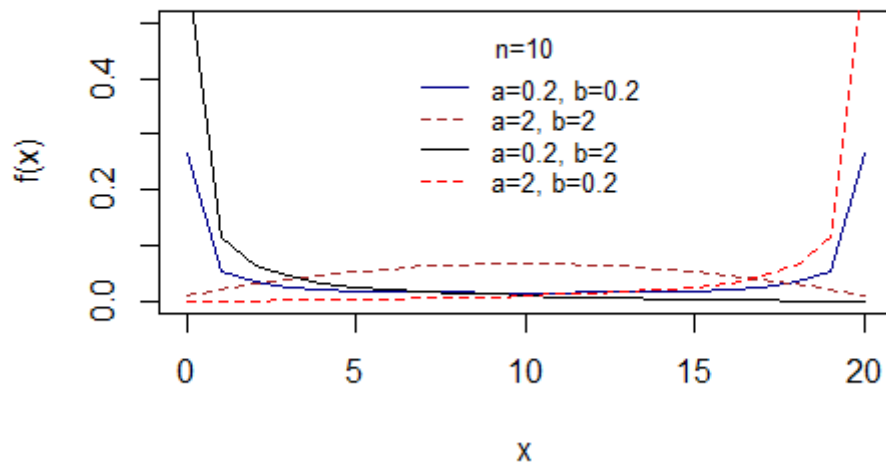
where  $\pi_j$  is the (mean) survival probability within group  $j$ , and  $\theta_j = \frac{1}{a_j + b_j}$ . Consequently, the mean and variance of the beta-binomial distribution are:

$$\text{E}(y_{ij}|k, a_j, b_j) = k\pi_j \quad \text{and} \quad \text{Var}(y_{ij}|k, a_j, b_j) = k\pi_j(1 - \pi_j)[1 + (k - 1)\rho_j] \quad (3.4)$$

It follows that the mean and variance of the beta-binomial distribution are of the same form as that of the binomial distribution. However, the variance of the beta-binomial distribution has a multiplicative overdispersion factor  $1 + (k - 1)\rho_j$  which, as stated by McCullagh and Nelder (1989), "enables us to proceed as if the

observations were binomially distributed”. Note that the parameters  $\rho_j$  are a measure of the pairwise correlation between the Bernoulli random variables, and thus model the overdispersion in the data. For the beta-binomial distribution, a large value of the correlation parameter  $\rho_j$  indicates a larger degree of overdispersion in the data (?). The beta-binomial distribution reduces to the binomial distribution when  $\theta_j = 0$  (or  $\rho_j = 0$ ) (Ennis and Bi, 1998).

Figure 3.1 shows a beta-binomial density function for various parameter values of  $a_j$  and  $b_j$ .



**Figure 3.1:** Beta-Binomial Density Function for Various Parameter Values

### 3.2.1 Beta-Binomial Distribution in Animal Studies

Consider the animal study in Section 2.5, for which Schall et al. (2016) considered the following model for the evaluation of ectoparasiticide efficacy. If each experimental animal  $i$  in a given treatment group  $j$  is infested with  $k$  parasites,

then, when conditioned on the surviving probabilities, the number of surviving parasites  $y_{ij}|p_{ij} \sim \text{Bin}(p_{ij}, k)$  distribution. The survival probabilities are allowed to vary across the animals within each of the treatment groups. Since the probabilities are random, the variability in  $p_{ij}$  has to be modeled. [Schall et al. \(2016\)](#) therefore assumed that  $p_{ij} \sim \text{Beta}(a_j, b_j)$ . Consequently, the number of surviving parasites  $y_{ij}$  follows a  $\text{BB}(k, a_j, b_j)$  distribution. Based on the properties of the beta-binomial distribution, it follows that the mean survival probability in a given treatment group is:

$$E(p_{ij}) = \pi_j = \frac{a_j}{a_j + b_j}$$

As a result, the efficacy of the test treatment (using Abbott's formula) can be expressed as:

$$\varepsilon_T = \left(1 - \frac{\pi_T}{\pi_C}\right) \cdot 100 = \left(1 - \frac{a_T/(a_T + b_T)}{a_C/(a_C + b_C)}\right) \cdot 100 \quad (3.5)$$

where the pairs  $(a_C, b_C)$  and  $(a_T, b_T)$  are the parameters of the beta-binomial distribution assumed for the control and test treatment groups, respectively. [Schall et al. \(2016\)](#) investigated the accuracy and precision of point estimates (of efficacy) for different estimators of  $\varepsilon_T$ .

### 3.2.2 Beta-Binomial as Hierarchical Generalized Linear Model

The linear predictor of a logistic model (with binomially distributed observations) is given by ([McCullagh and Nelder, 1989](#)):

$$\log\left(\frac{\pi_j}{1 - \pi_j}\right) = \beta_0 + \sum_{k=1}^n x_{ik}\beta_k \quad (3.6)$$

The beta-binomial distribution described in Section 3.2 can be viewed as a special case of a HGLM. On the grounds that the beta distribution is conjugate to the binomial distribution, the conditioned response follows a distribution within the exponential-family (Najera-Zuloaga et al., 2017). Consider the following HGLM model for the beta-binomial distribution: When the numbers of surviving parasites in a given treatment group are conditioned on random components  $u_{ij}$ , then  $y_{ij}|u_{ij} \sim \text{Bin}(k, p_{ij})$  Schall et al. (2016). If one assumes that  $u_{ij} \sim \text{Beta}(a_j, b_j)$  then a conjugate distribution for the random component is accordingly specified. However, note that a distribution for  $p_{ij}$  is not specified but instead  $p_{ij}$  and  $u_{ij}$  are connected with the linear predictor of the HGLM (Najera-Zuloaga et al., 2017). Therefore the linear predictor of the beta-binomial HGLM is given by:

$$\begin{aligned} \eta_j^* &= \eta + \nu(u_{ij}) \\ \log\left(\frac{\pi_j}{1 - \pi_j}\right) &= \beta_0 + \sum_{k=1}^n x_{ik}\beta_{ik} + \nu(u_{ij}) \end{aligned} \quad (3.7)$$

Note that Equation (3.7) is defined by adding random components to Equation (3.6) (that is, the GLM for binary responses). The term  $\nu(u_{ij})$  is the scale on which the random components are included in the linear predictor. In this case  $\nu(u_{ij}) = \log\left(\frac{u_{ij}}{1-u_{ij}}\right)$  represents the random effects attributed to treatment  $j$  (Lee and Nelder, 1996). Using  $\nu(u_{ij})$  instead of  $u_{ij}$  “allows the model to maintain invariance of inference with respect to equivalent approaches” (Najera-Zuloaga et al., 2017). To maintain the structure of the model, Lee and Nelder (2001) proposed to restrict the random components of the conjugate HGLM. In the case of the beta-binomial conjugate HGLM, the authors suggest that the mean of the distribution of  $u_{ij}$  should be set to that value of  $u_{ij}$ , for which the scale  $\nu(u_{ij})$  transforms to zero. Therefore the constraint placed on the  $u_{ij}$  is  $E(u_{ij}) = 1/2$ .

Consequently, from the properties of the beta distribution,  $a_j = b_j$  (symmetric) and the selection of the value of  $a_j$  depends on the variability of the  $u_{ij}$ . It therefore follows that  $u_{ij} \sim \text{Beta}(1/a_j, 1/a_j)$  for  $a_j > 0$ . The aforementioned results in the previously defined parameters  $\theta_j = a_j/2$  and  $\rho_j = a_j/(a_j + 2)$  (correlation parameter) (Najera-Zuloaga et al., 2017).

For the application of the beta-binomial model, which is the aim of this study, two methods will be employed: GLM (as implemented through SAS PROC GENMOD) and the Bayesian HGLM (as implemented through R-INLA).

### 3.3 Integrated Nested Laplace Approximations

Rue et al. (2009) proposed the INLA as a non-sampling alternative to MCMC schemes when dealing with the subclass of structured additive regression models, namely LGMs. More specifically, the INLA method performs approximate Bayesian inference for the class of LGMs. The underlying theory of INLA has been well established by Rue et al. (2009), and the method continues to be an active research field (Martins and Rue, 2014).

#### 3.3.1 Latent Gaussian Models

The class of latent Gaussian models can be defined using the following hierarchical structure (Rue et al., 2017):

Stage 1:

The data points  $\{y_i\}; i = 1, 2, \dots, n_d$  are assumed to be conditionally independent given a latent field  $\mathbf{x}$  and possible hyperparameters  $\boldsymbol{\theta}_1$ , to form the

likelihood function:

$$\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}_1 \sim \pi(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}_1) = \prod_i \pi(y_i|x_i, \boldsymbol{\theta}_1)$$

Stage 2:

The latent field  $\mathbf{x}$ , conditional on the hyperparameters  $\boldsymbol{\theta}_2$ , is assumed to follow a joint multivariate Gaussian distribution, i.e.:

$$\mathbf{x}|\boldsymbol{\theta}_2 \sim \pi(\mathbf{x}|\boldsymbol{\theta}_2) = \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\theta}_2), \mathbf{Q}(\boldsymbol{\theta}_2)^{-1})$$

Stage 3:

The last stage is constructed by assuming a (suitable) prior distribution for the hyperparameters of the model (namely  $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\}$ ):

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$$

There may be additional linear constraints for the latent field  $\mathbf{x}$ , which are in the form of  $\mathbf{A}\mathbf{x} = \mathbf{c}$ . Here the matrix  $\mathbf{A}$  is  $k \times n$  of rank  $k$ , where  $k$  is the number of constraints, and  $n$  is the size of  $\mathbf{x}$  (Rue et al., 2009). The posterior distribution of the unknowns can therefore be expressed as:

$$\begin{aligned} \pi(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y}) &\propto \pi(\boldsymbol{\theta})\pi(\mathbf{x}|\boldsymbol{\theta}) \prod_{i=1} \pi(y_i|x_i, \boldsymbol{\theta}) \\ &\propto \pi(\boldsymbol{\theta})|\mathbf{Q}(\boldsymbol{\theta})|^{n/2} \exp \left\{ \frac{1}{2} \mathbf{x}' \mathbf{Q}(\boldsymbol{\theta}) \mathbf{x} + \sum_{i=1}^{n_d} \log \{y_i|x_i, \boldsymbol{\theta}\} \right\} \end{aligned} \quad (3.8)$$

where  $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ .

The majority of the LGMs mentioned in the literature satisfy the following properties:

- (i) The latent field is often high dimensional ( $n = 10^2$  to  $10^5$ ) and, given the conditional independence properties, is a Gaussian Markov random field (GMRF) with a sparse precision matrix. This in turn leads to the use of numerical methods for sparse matrices which are significantly faster than numerical calculations for general dense matrices.
- (ii) The dimension of the hyperparameters  $\boldsymbol{\theta}$  is generally small, say  $m = \boldsymbol{\theta} \leq 6$ .

Normally, both of these properties are required in order to yield fast inference, but exceptions do exist (Rue et al., 2009).

Latent Gaussian models include a variety of statistical models. For example, interpreting the likelihood such that  $y_i$  relates to a single  $x_i$  yields the GLM setup (Rue et al., 2017). For structured additive regression models, the responses  $y_i$  are assumed to belong to some distribution family which does not necessarily have to be part of the exponential family of distributions (Martins and Rue, 2014). The mean  $\mu_i$  is linked to an additive predictor  $\eta_i$  through a link function  $g(\mu_i)$  in the following way (Rue et al., 2009):

$$\eta_i = g(\mu_i) = \beta_0 + \sum_{k=1}^{n_f} f(u_{ij}) + \sum_{j=1}^N \beta_j x_{ij} + \varepsilon_i \quad \text{for } i = 1, 2, \dots, n \quad (3.9)$$

where

$f(u_{ij})$  are functions of the random effects  $\mathbf{u}$

$\beta_k$  are the linearly structured parameters of covariates  $\mathbf{x}$

$\varepsilon_i$  are the residual effects.

### 3.3.2 Gaussian Approximation

An integral part of the INLA methodology is the Gaussian prior assumption of the latent field (Stage 2). The method relies greatly on this assumption, as well as on the choice of approximations being used (Martins and Rue, 2014). As previously indicated in Section 3.3.1, an important contribution of the Gaussian assumption of  $\mathbf{x}|\boldsymbol{\theta}_2$ , is that the conditional independence properties translate into a sparse precision matrix (Martins et al., 2013). This in turn implies that numerical methods for sparse matrices may be used (Rue et al., 2009), and as numerical methods for sparse matrices are faster than that of a dense matrix, the Gaussian assumption leads to less computational time (Martins and Rue, 2014). The Gaussian approximation is used for densities of the form:

$$\pi(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y}) \propto \exp\left\{-\frac{1}{2}\mathbf{x}'\mathbf{Q}(\boldsymbol{\theta})\mathbf{x} + \sum_i p_i(x_i)\right\} \quad (3.10)$$

where, in this specific setting,  $p_i(x_i) = \log\{y_i|x_i, \boldsymbol{\theta}\}$  (Rue et al., 2009). One way of estimating Equation (3.10) by a Gaussian distribution is to perform a Taylor expansion up to the second order of  $p_i$  based on an initial guess,  $\mu_i^{(0)}$ . The approximation is of the following form:

$$p_i(x_i) \approx p_i(\mu_i^{(0)}) + b_i x_i - \frac{1}{2} c_i x_i^2$$

where  $c_i$  and  $b_i$  depend on  $\mu_i^{(0)}$ . The Taylor expansion results in a Gaussian approximation having precision matrix  $\mathbf{Q} + \text{diag}(\mathbf{c})$ , and for which the mode is the solution of  $\mathbf{b} = \{\mathbf{Q} + \text{diag}(\mathbf{c})\}\boldsymbol{\mu}^{(1)}$ . Note: The vectors  $\mathbf{b}$  and  $\mathbf{c}$  are formed by  $b_i$  and  $c_i$ 's. The aforementioned procedure is performed until it converges to a Gaussian distribution, denoted by  $\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$  (Martins and Rue, 2014).

### 3.3.3 Methodology

For the hierarchical LGM in Section 3.3.1, the posterior marginal densities of interest are as follows:

$$\begin{aligned}\pi(x_i|\mathbf{y}) &= \int \pi(x_i|\boldsymbol{\theta}, \mathbf{y})\pi(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta} \quad \text{for } i = 1, 2, \dots, n \\ \pi(\theta_j|\mathbf{y}) &= \int \pi(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta}_{-j} \quad \text{for } j = 1, 2, \dots, m\end{aligned}\tag{3.11}$$

where  $m = \dim(\boldsymbol{\theta})$ , and  $\boldsymbol{\theta}_{-j}$  denotes the vector  $\boldsymbol{\theta}$  minus its  $j^{\text{th}}$  element (Rue et al., 2009). The approximated posterior marginals returned by INLA have the following form:

$$\begin{aligned}\hat{\pi}(x_i|\mathbf{y}) &= \sum_k \hat{\pi}(x_i|\boldsymbol{\theta}^{(k)}, \mathbf{y})\hat{\pi}(\boldsymbol{\theta}^{(k)}|\mathbf{y})\Delta\boldsymbol{\theta}^{(k)} \quad \text{for } i = 1, 2, \dots, n \\ \hat{\pi}(\theta_j|\mathbf{y}) &= \int \hat{\pi}(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta}_{-j} \quad \text{for } j = 1, 2, \dots, m\end{aligned}\tag{3.12}$$

where  $\{\hat{\pi}(\boldsymbol{\theta}^{(k)}|\mathbf{y})\}$  are  $k$  values obtained while doing a grid exploration of  $\{\hat{\pi}(\boldsymbol{\theta}|\mathbf{y})\}$  (Martins and Rue, 2014). Furthermore, Martins and Rue (2014) discussed the following simplified algorithm for the approximations:

**Step 1:** Select a set of  $k$  points.  $\Theta = \{\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(k)}\}$

Assume  $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_k\} \in \mathbb{R}^m$  then the Laplace approximation of the posterior of  $\boldsymbol{\theta}$  is given by:

$$\hat{\pi}(\boldsymbol{\theta}|\mathbf{y}) \propto \frac{\pi(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y})}{\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})} \Big|_{\mathbf{x}=\mathbf{x}^*(\boldsymbol{\theta})}\tag{3.13}$$

where  $\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$  is the Gaussian approximation as discussed in Section 3.3.2.

Then:

- (i) Obtain the mode of  $\hat{\pi}(\boldsymbol{\theta}|\mathbf{y})$  (namely,  $\boldsymbol{\theta}^*$ ) and compute the negative Hessian matrix  $\mathbf{H}$  at the modal configuration.
- (ii) Calculate the eigen-decomposition  $\boldsymbol{\Sigma} = \mathbf{V}\boldsymbol{\Lambda}^{1/2}\mathbf{V}'$  where  $\boldsymbol{\Sigma} = \mathbf{H}^{-1}$ .
- (iii) Define  $\mathbf{z}$  such that:

$$\boldsymbol{\theta}(\mathbf{z}) = \boldsymbol{\theta}^* + \mathbf{V}\boldsymbol{\Lambda}^{1/2}\mathbf{z}$$

The variable  $\mathbf{z} = (z_1, z_2, \dots, z_m)$  is known to be standardized and its components are mutually orthogonal. This reparametrization is used to explore  $\hat{\pi}(\boldsymbol{\theta}|\mathbf{y})$  in order to produce a grid of  $k$  points related to the bulk of the mass. The  $z$ -parametrization also emends for scale and rotation, and reduces the complexity of the numerical integration process (Rue et al., 2009).

**Step 2:** Approximate  $\hat{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y})$

For each  $k$ , evaluate  $\hat{\pi}(\boldsymbol{\theta}|\mathbf{y})$  for the computation of  $\hat{\pi}(\boldsymbol{\theta}^{(k)}|\mathbf{y})$ , from which it is possible to approximate  $\hat{\pi}(x_i|\boldsymbol{\theta}^{(k)}, \mathbf{y})$  as a function of  $x_i$  for all  $i = 1, 2, \dots, n$ . Three possible methods are available to approximate  $\hat{\pi}(x_i|\boldsymbol{\theta}^{(k)}, \mathbf{y})$ .

Gaussian approximation:  $\pi_G(x_i|\boldsymbol{\theta}, \mathbf{y})$

When evaluating  $\hat{\pi}(\boldsymbol{\theta}|\mathbf{y})$  in Step 1, the Gaussian approximation  $\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$  is computed. Therefore it seems natural to obtain the approximate marginal densities from  $\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$ .

Laplace approximation:  $\hat{\pi}_{LA}(x_i|\boldsymbol{\theta}, \mathbf{y})$

This step is a more accurate approach which entails the use of a Laplace approximation of the form:

$$\hat{\pi}_{LA}(x_i|\boldsymbol{\theta}, \mathbf{y}) \propto \frac{\pi(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y})}{\pi_{GG}(\mathbf{x}_{-i}|x_i, \boldsymbol{\theta}, \mathbf{y})} \Big|_{\mathbf{x}_{-i}=\mathbf{x}_{-i}^*(x_i, \boldsymbol{\theta})} \quad (3.14)$$

where  $\pi_{GG}$  is the Gaussian approximation for  $\mathbf{x}_{-i}|x_i, \boldsymbol{\theta}, \mathbf{y}$  and  $\mathbf{x}_{-i}^*(x_i, \boldsymbol{\theta})$  is the modal configuration. However, Equation (3.14) requires that  $\pi_{GG}$  should be computed for each value of  $x_i$ . Thus Rue et al. (2009) proposed to present the density of  $\hat{\pi}_{LA}(x_i|\boldsymbol{\theta}, \mathbf{y})$  by the following:

$$\hat{\pi}_{LA}(x_i|\boldsymbol{\theta}, \mathbf{y}) \propto \mathcal{N}\{x_i; \mu_i(\boldsymbol{\theta}), \sigma_i^2(\boldsymbol{\theta})\} \exp\{\text{cubic spline}(x_i)\}$$

Simplified Laplace approximation:  $\hat{\pi}_{SLA}(x_i|\boldsymbol{\theta}, \mathbf{y})$

For this approximation Rue et al. (2009) proposed a Taylor series expansion of both the numerator and denominator of Equation (3.14) around  $x_i = \mu_i(\boldsymbol{\theta})$ , up to the third order. The expansion allows to correct the Gaussian approximation  $\hat{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y})$  for location and skewness, at a much lower computational cost than  $\hat{\pi}_{LA}$ .

The Gaussian approximation algorithm is faster since it requires only additional computation of marginal variances. On the one hand the Gaussian approximation often gives reasonable results, and on the other, there can be inaccuracies in the calculation of location parameters should the data be skewed. The Laplace approximation is generally preferred, thus a substantial simplification of Laplace approximations rectifies for the loss in accuracy. Once  $\hat{\pi}(x_i|\boldsymbol{\theta}, \mathbf{y})$  is obtained,  $\hat{\pi}(x_i|\boldsymbol{\theta}^{(k)}, \mathbf{y})$  is computed for each  $k$ , as a function of  $x_i$  for all  $i$ .

**Step 3:** Compute  $\hat{\pi}(x_i|\mathbf{y})$

This step combines Step 1 and Step 2 to obtain an approximation of  $\pi(x_i|\mathbf{y})$ .

The approximation is of the following form:

$$\hat{\pi}(x_i|\mathbf{y}) = \sum_k \hat{\pi}(x_i|\boldsymbol{\theta}^{(k)}, \mathbf{y}) \hat{\pi}(\boldsymbol{\theta}^{(k)}|\mathbf{y}) \Delta\boldsymbol{\theta}^{(k)} \quad (3.15)$$

as a function of  $x_i$  for all  $i = 1, 2, \dots, n$ .

Additionally, one can compute an approximation for  $\pi(\theta_j|\mathbf{y})$ . The latter approximation is obtained by using the density values  $\{\hat{\pi}(\boldsymbol{\theta}^{(k)}|\mathbf{y})\}$  to construct an interpolant,  $\mathbf{I}(\boldsymbol{\theta}|\mathbf{y})$ , after which numerical integration is used to obtain the marginals. Therefore the approximation is of the following form (Rue et al., 2009):

$$\hat{\pi}(\theta_j|\mathbf{y}) = \int \mathbf{I}(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta}_{-j} \quad (3.16)$$

### 3.3.4 Model Discrimination Statistics

#### 3.3.4.1 Bayes Factors

Bayes factors are used for model comparison (see Section 2.2.2), and is defined as the quotient of marginal likelihoods of  $\mathbf{y}$  under two competing models (Ntzoufras, 2009). For the approximation of the marginal likelihoods, Rue et al. (2009) proposed the use of the normalizing constant of  $\hat{\pi}(\theta_j|\mathbf{y})$ . That is:

$$\hat{\pi}(\mathbf{y}) = \int \frac{\pi(\mathbf{x}, \boldsymbol{\theta}, \mathbf{y})}{\pi_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})} \Big|_{\mathbf{x}=\mathbf{x}^*(\boldsymbol{\theta})} d\boldsymbol{\theta} \quad (3.17)$$

Additionally, INLA computes a cruder alternative estimate of the marginal likelihood by assuming that  $\boldsymbol{\theta}|\mathbf{y}$  is Gaussian distributed (Rue et al., 2009).

### 3.3.4.2 Deviance Information Criterion

The DIC can be defined as the deviance of the posterior mean of the parameters plus twice the number of effective parameters, where the deviance is given by:

$$D(\mathbf{x}, \boldsymbol{\theta}) = -2 \sum_{i \in I} \log\{\pi(y_i | x_i, \boldsymbol{\theta})\} + \text{const.}$$

Since  $\pi(\mathbf{x} | \mathbf{y}, \boldsymbol{\theta})$  asymptotically follows a Gaussian distribution, [Rue et al. \(2009\)](#) proposed the following approximation for the number of effective parameters:

$$p_M \approx n - \text{tr}\{\mathbf{Q}(\boldsymbol{\theta})\mathbf{Q}^*(\boldsymbol{\theta})^{-1}\}$$

where  $n = \dim(\mathbf{x})$ , and  $\mathbf{Q}(\boldsymbol{\theta})$  &  $\mathbf{Q}^*(\boldsymbol{\theta})^{-1}$  are respectively the precision and posterior covariance matrix of the Gaussian approximation. The posterior mean of the deviance can be calculated in two steps. In the first step, the conditional mean (conditioned on  $\boldsymbol{\theta}$ ) of the deviance is computed using numerical integration for each observation. That is (see Equation (3.8)):

$$\begin{aligned} E_{\mathbf{x}|\boldsymbol{\theta}}[D(\mathbf{x}, \boldsymbol{\theta})] &= \int D(\mathbf{x}, \boldsymbol{\theta})\pi(x_i | \mathbf{y})dx_i \\ &= \int D(\mathbf{x}, \boldsymbol{\theta})\pi(\boldsymbol{\theta} | \mathbf{y})\pi(x_i | \boldsymbol{\theta}, \mathbf{y})d\boldsymbol{\theta}dx_i \end{aligned} \tag{3.18}$$

In the second step,  $\boldsymbol{\theta}$  is eliminated by integrating  $E_{\mathbf{x}|\boldsymbol{\theta}}[D(\mathbf{x}, \boldsymbol{\theta})]$  with respect to  $\pi(\boldsymbol{\theta} | \mathbf{y})$ .

### 3.3.4.3 Predictive Integral Transform

In addition to the main computations of INLA, the predictive integral transform (PIT) can be obtained. The PIT is defined as:

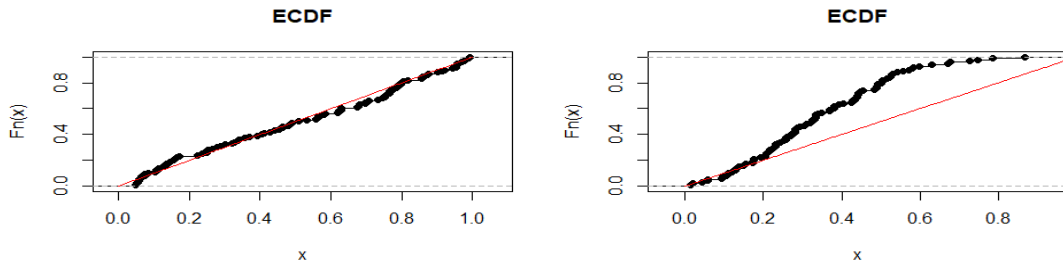
$$\text{PIT}_i = \text{Prob}(y_i^{\text{new}} \leq y_i | \mathbf{y}_{-i}) \quad (3.19)$$

Note that by the omission of  $y_i$  from the data results in:

$$\pi(y_i | \mathbf{y}_{-i}, \boldsymbol{\theta}) = \int \pi(y_i | x_i, \boldsymbol{\theta}) \pi(x_i | \mathbf{y}_{-i}, \boldsymbol{\theta}) dx_i$$

Here,  $\boldsymbol{\theta}$  can be integrated out using the same device as Equation (3.15) (Rue et al., 2009). The PIT is a measure that relies on the following: If a random variable  $x$  has a continuous cumulative distribution function,  $F_x$ , then the random variable  $Y = F_x(x)$  is uniformly distributed on the interval  $[0, 1]$ . The model can therefore be questioned, if the  $\text{PIT}_i$  deviates too far from a uniform distribution (Rue et al., 2009).

In Figure 3.2, the ECDF of the uniform distribution is plotted against the ECDF of some unknown distribution to assess the fit of the uniform distribution to the underlying distribution.

**Figure 3.2:** Good Versus Bad Fits of the Uniform Distribution

The left-hand side of Figure 3.2 illustrates closely scattered simulated points of the  $U(0, 1)$  distribution. The latter is considered a good fit of the uniform distribution to the (true) underlying distribution. On the right-hand side of Figure 3.2, majority of the simulated points are above the ECDF graph of the uniform distribution. In the latter case, the standard uniform distribution is considered a bad fit to the underlying distribution.

### 3.4 INLA of the Beta-Binomial Distributions

Lee and Nelder (1996) noticed that the distributions of  $u_{ij}$  used in conjugate HGLMs follow closely the normal distribution. Therefore the authors concluded that conjugates for HGLM and GLMM often yield similar results. The INLA method has been proven to be highly effective when applied to GLMMs (see for example Fong et al. (2010) and Holand et al. (2013)). Recall from Section 3.2.2 that the beta-binomial distribution can be viewed as a conjugate HGLM. As an end result, the restrictions imposed on the random part of the model (to maintain its structure) resulted in  $u_{ij} \sim \text{Beta}(1/a_j, 1/a_j)$ . Based on the properties of the

beta distribution,  $u_{ij}$  are symmetric about the point  $1/a_j$ . INLA can thus be applied to the beta-binomial distribution to perform approximation inferences.

### 3.5 Conclusion

In this chapter we discussed a justification for the use of Bayesian INLA methods as alternatives to traditional MCMC procedures. The beta-binomial distribution was introduced, together with a discussion of its application to the assessment of efficacy of ectoparasiticide treatments. Specifically, it was shown that the beta-binomial distribution can be viewed as a special case of HGLMs. An overview of the INLA methodology was provided, together with its associated model discrimination statistics. The chapter is concluded by explaining how the beta-binomial distribution fits into the INLA framework.

# Chapter 4

## Applications

### 4.1 Introduction

In this chapter, the efficacy of investigational veterinary products (IVPs) is analyzed for two previously published animal trials. In each trial, eight dogs were randomized to each of three study treatments involved, namely an investigational veterinary product (IVP; test treatment (T)), a control veterinary product (CVP; reference treatment (R)) and an untreated control (C). Each dog underwent weekly infestations of  $k = 50$  ticks, and on each of the assessment days (Days 2, 9, 16, 23 and 30, wherever applicable) the surviving ticks on each dog were counted ([Schall and Luus, 2011](#)). The PROC GENMOD (SAS<sup>®</sup>) procedure is employed to reproduce the GLM results obtained by [Schall and Luus \(2011\)](#). In addition, the R package R-INLA was used to fit the Bayesian model to the data. The latter uses an HGLM to approximate the marginal posterior distributions of the hyperparameters, and the marginal posterior distribution of the latent vector ([Rue et al., 2009](#)).

### 4.1.1 Generalized Linear Model Fit

Schall and Luus (2011) investigated the efficacy (separately by study day) using the following specifications: The binomial distribution for the counts  $y_{ij}$ , Pearson scale and the log-link function. The data are arranged as follows: The response variable  $y_{ij}$  contains the number of surviving parasites,  $k$  is the corresponding number of parasites with which each experimental animal was infested, and **TREATMENT** specifies the treatment group associated with each count  $y_{ij}$ . PROC GENMOD fits a GLM to the data using maximum likelihood estimates of the parameters  $\beta_{ik}$  in the linear predictor. The generated point estimates are used to compute the ratio's  $\log(\pi_T)/\log(\pi_C)$  and  $\log(\pi_R)/\log(\pi_C)$ . Thus, to obtain the efficacy, the authors used the following transformation:  $\varepsilon_T = 100(1 - \log(\pi_T)/\log(\pi_C))$  and  $\varepsilon_R = 100(1 - \log(\pi_R)/\log(\pi_C))$ . 95% Wald confidence intervals were obtained using Equation (2.14).

### 4.1.2 Fit of Bayesian HGLM by INLA

To illustrate the proposed INLA methodology, an HGLM was fitted (separately by study day) using the following specifications: The beta-binomial distribution for the likelihood function for the counts  $y_{ij}$ , and Gaussian priors for the hyperparameters (overdispersion parameter)  $\theta = \{\rho_T, \rho_C, \rho_R\}$ . Subsequently, 10,000 posterior samples were simulated from the approximated posterior distribution. A latent vector (containing the linear predictors) is extracted from each sample. Vague prior distributions were assumed for the overdispersion parameters. The latter serves as an approximation to the survival probabilities of each study treatment  $\pi_j$ ,  $j = C, R, T$ . Subsequently, the efficacy for each treatment was calculated using

Equation (2.9). The posterior mean was used as an estimate for the efficacy of each study treatment.

## 4.2 Results and Discussion

Table 4.1 presents the estimated efficacy (%) and corresponding confidence intervals for the test (T) and reference (R) study treatment, for each assessment day. The results obtained are from both methods employed, namely PROC GENMOD and R-INLA .

**Table 4.1:** Efficacy (%) Estimates and Corresponding 95% CIs

Study	Day	PROC GENMOD		INLA	
		Estimate (95% CI)		Estimate (95% CI)	
		Reference	Test	Reference	Test
1	2	55.3 (26.3; 72.8)	77.4 (54.5; 88.7)	53.1 (29.2; 69.6)	72.3 (37.6; 89.2)
	9	97.2 (89.6; 99.2)	96.7 (89.0; 99.0)	95.5 (83.5; 98.9)	94.3 (78.5; 98.7)
	16	83.9 (72.7; 90.5)	87.8 (77.6; 93.4)	82.3 (68.4; 89.8)	85.6 (69.2; 93.3)
	23	60.3 (40.3; 73.6)	74.6 (57.8; 84.7)	57.2 (27.4; 76.3)	73.0 (58.0; 82.4)
	30	58.2 (38.6; 71.5)	57.3 (37.6; 70.8)	56.8 (41.0; 68.5)	54.7 (29.8; 71.5)
2	2	34.2 (7.8; 53.0)	45.6 (20.4; 62.8)	37.2 (7.2; 62.7)	43.2 (19.2; 61.8)
	9	95.5 (81.1; 98.9)	94.9 (80.4; 98.7)	91.0 (62.3; 98.9)	89.7 (57.8; 98.8)
	16	99.1 (96.9; 99.7)	98.2 (95.8; 99.3)	98.1 (91.8; 99.7)	97.6 (93.8; 99.3)
	23	64.7 (44.1; 77.7)	64.3 (43.6; 77.4)	63.1 (39.6; 79.1)	60.8 (37.2; 76.8)
	30	85.8 (68.4; 93.6)	84.2 (66.3; 92.6)	83.3 (64.8; 92.5)	80.0 (50.9; 93.2)
3	2	51.5 (13.5; 72.8)	49.2 (10.4; 71.2)	48.1 (10.8; 70.8)	45.0 (4.2; 70.7)
	9	82.6 (60.3; 92.3)	67.4 (42.1; 81.7)	79.2 (57.3; 90.6)	65.6 (37.4; 84.1)
	16	77.5 (58.4; 87.8)	64.2 (41.0; 78.2)	74.5 (50.7; 86.6)	59.4 (21.5; 79.2)

Note: CI: Confidence interval.

In most cases the GLM fit (PROC GENMOD procedure) yielded slightly higher efficacy estimates than the fit of the Bayesian HGLM through R-INLA. Nonetheless, the estimates from PROC GENMOD are generally quite similar to those obtained from R-INLA: The maximum difference between the efficacy estimates occurs at Day 9 of Study 2 where the PROC GENMOD reference efficacy estimate differs from the corresponding R-INLA estimate by 4.5%, and the test efficacy estimates differs by 5.2%; similarly, the minimum difference between the efficacy estimates obtained from the two methods employed, occurs on Day 16 of Study 2. In the latter case, the PROC GENMOD reference efficacy estimate differs from the corresponding R-INLA estimate by 1%, whereas the test efficacy estimates differs by 0.6%.

The efficacy estimates of both methods and study treatments generally increase up to Day 9. A substantial increase is observed at Day 9 for Studies 1 and 2, where the efficacy estimates are close to 100% (for both test and reference treatments).

The estimates for the pairwise correlation parameter from R-INLA are presented in Table 4.2 (by study day and treatment). In addition, from the PROC GENMOD procedure, the ratio of the deviance and its associated (asymptotic) degrees of freedom are presented (by study day).

**Table 4.2:** Estimates of Overdispersion Parameters

Study	Day	PROC GENMOD			INLA					
		Deviance	df	$\hat{\phi}$	$\hat{\rho}_T$	$\hat{\rho}_C$	$\hat{\rho}_R$	$1 + (k - 1)\hat{\rho}_j$		
								j=T	j=C	j=R
1	2	113.72	21	5.38	0.06	0.06	0.21	3.94	3.94	11.29
	9	57.25	21	2.68	0.07	0.06	0.10	4.43	3.94	5.90
	16	58.22	21	2.41	0.04	0.04	0.07	2.96	2.96	4.43
	23	74.20	21	3.19	0.02	0.15	0.03	1.98	8.35	2.47
	30	78.88	21	3.68	0.08	0.02	0.10	4.92	1.98	5.90
2	2	165.90	21	6.93	0.10	0.26	0.12	5.90	13.74	6.88
	9	69.35	21	4.21	0.02	0.24	0.24	1.98	12.76	12.76
	16	18.07	21	0.81	0.01	0.03	0.01	1.49	2.47	1.49
	23	125.08	21	5.50	0.02	0.17	0.15	1.98	9.33	8.38
	30	126.00	21	5.85	0.08	0.11	0.24	4.92	6.39	12.76
3	2	106.60	21	4.80	0.14	0.06	0.08	7.86	3.94	4.92
	9	231.21	21	10.29	0.16	0.18	0.34	8.84	9.82	17.66
	16	54.20	21	2.43	0.00	0.05	0.10	1.00	3.45	5.90

*Note:* df: Degrees of freedom. DIC: Deviance information criterion.

The ratio of the deviance and its associated degrees of freedom (namely  $\hat{\rho}$ ) is greater than one for the majority of studies and assessment days, therefore suggesting that overdispersion is present in the data. (Exception: For Day 16 of Study 2, the deviance versus degrees of freedom ratio is 0.86.)

The R-INLA estimates for the pairwise correlation within each treatment group ( $\hat{\rho}_j$  for  $j = T, C, R$ ) are moderate, therefore suggesting that dispersion in the data may be due to clustering effects.

The Bayesian HGLM has a multiplicative overdispersion factor (namely  $1 + \hat{\rho}_j (k -$

1) for  $j = T, C, R$  and  $k = 50$ ; see Section 3.2) that is of the same form as the overdispersion estimator of the (frequentist) GLM (namely  $\hat{\rho}$ ; see Section 2.6). Note that both aforementioned quantities are used in calculating the variance of the respective models. For the majority of studies and assessment days,  $1 + (k - 1)\hat{\rho}_j$  for  $j = T, C, R$  of the Bayesian HGLM is greater than  $\hat{\rho}$  of the GLM. Thus, the variance explained by the Bayesian HGLM would be larger than the variance explained by the GLM. Consequently, the Bayesian HGLM would have a larger degree of overdispersion than the GLM.

To evaluate the fit of the beta-binomial distribution, the marginal log-likelihoods and DICs of the conventional binomial and beta-binomial distribution are compared. The model discrimination statistics are presented in Table 4.3.

**Table 4.3:** Model Discrimination Statistics: Beta-binomial vs Binomial Model

Study	Day	Beta-binomial		Binomial	
		$\log\{\hat{\pi}(y M)\}$	DIC	$\log\{\hat{\pi}(y M)\}$	DIC
1	2	-91.28	154.89	-115.82	204.45
	9	-63.44	103.35	-68.06	113.01
	16	-80.89	134.91	-82.79	139.65
	23	-88.54	148.86	-97.94	168.68
	30	-94.05	159.04	-104.86	181.85
2	2	-105.84	182.35	-148.33	268.33
	9	-60.32	96.71	-68.24	112.97
	16	-51.53	75.58	-46.04	70.63
	23	-97.31	164.69	-125.29	222.81
	30	-87.11	146.64	-115.17	204.04
3	2	-95.18	160.83	-114.71	202.18
	9	-98.05	168.55	-170.76	314.35
	16	-79.32	128.47	-82.28	138.50

Note: DIC: Deviance information criteria.

The DIC favors models with smaller values of DICs. Conversely, the Bayes factor favors models with larger posterior marginal log-likelihoods (see Section 2.2.2). For each of the studies and assessment days, the marginal log-likelihoods of the beta-binomial model are greater those of the binomial likelihood. Also, the beta-binomial model yields smaller values of DIC than those obtained from the conventional binomial model. Here, both model comparison statistics are in favor of the beta-binomial model.

Figure 4.1 to Figure 4.3 depict the empirical cumulative distribution function (ECDF) of the posterior predictive integral transform (PIT). The ECDF of the PITs is plotted against the ECDF of a standard uniform distribution. The left-hand side shows the plots of the beta-binomial distribution, whereas the right-hand side shows the plots of the conventional binomial distribution.

Figure 4.1: Study 1 - ECDF of Posterior Integral Transform

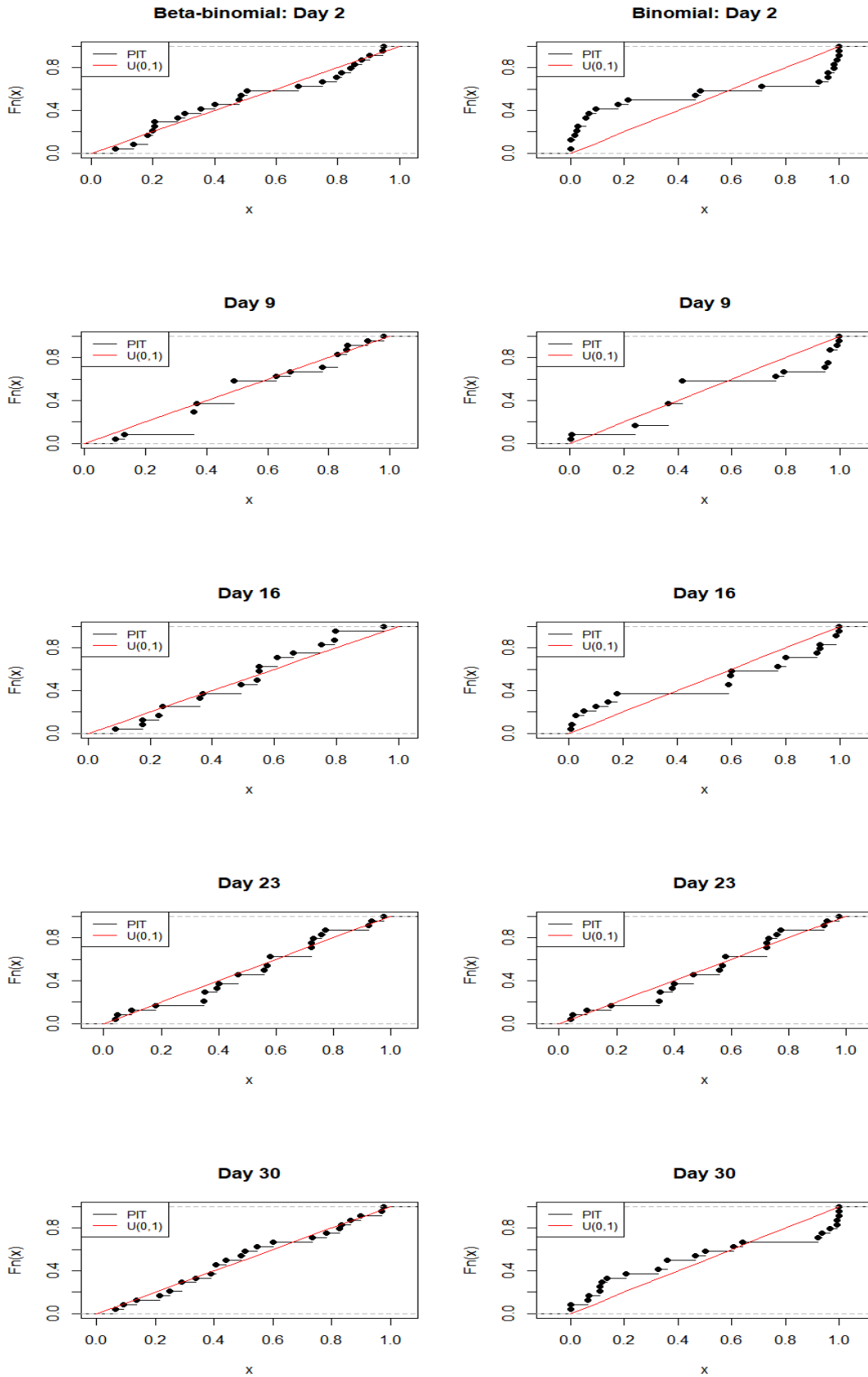
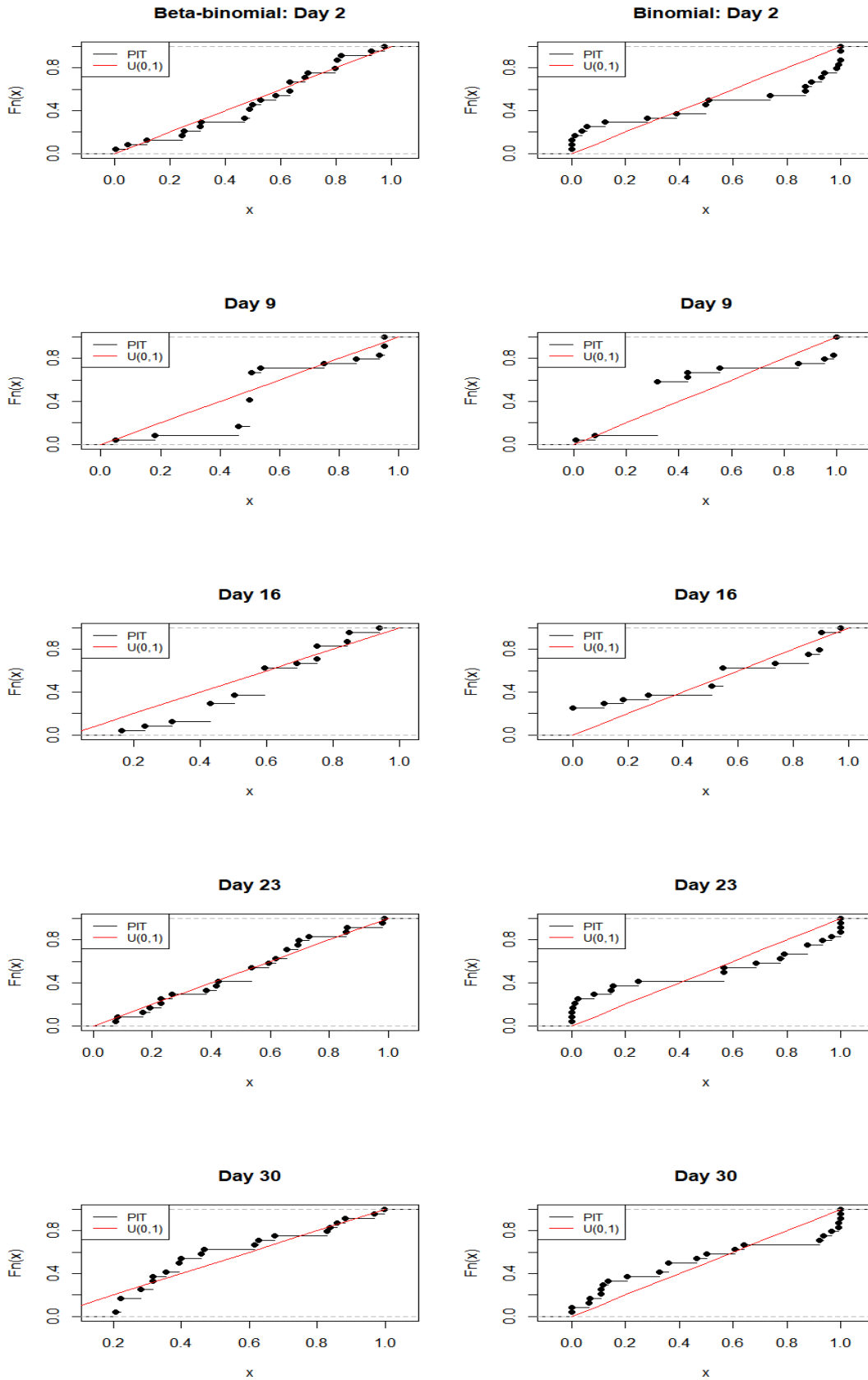
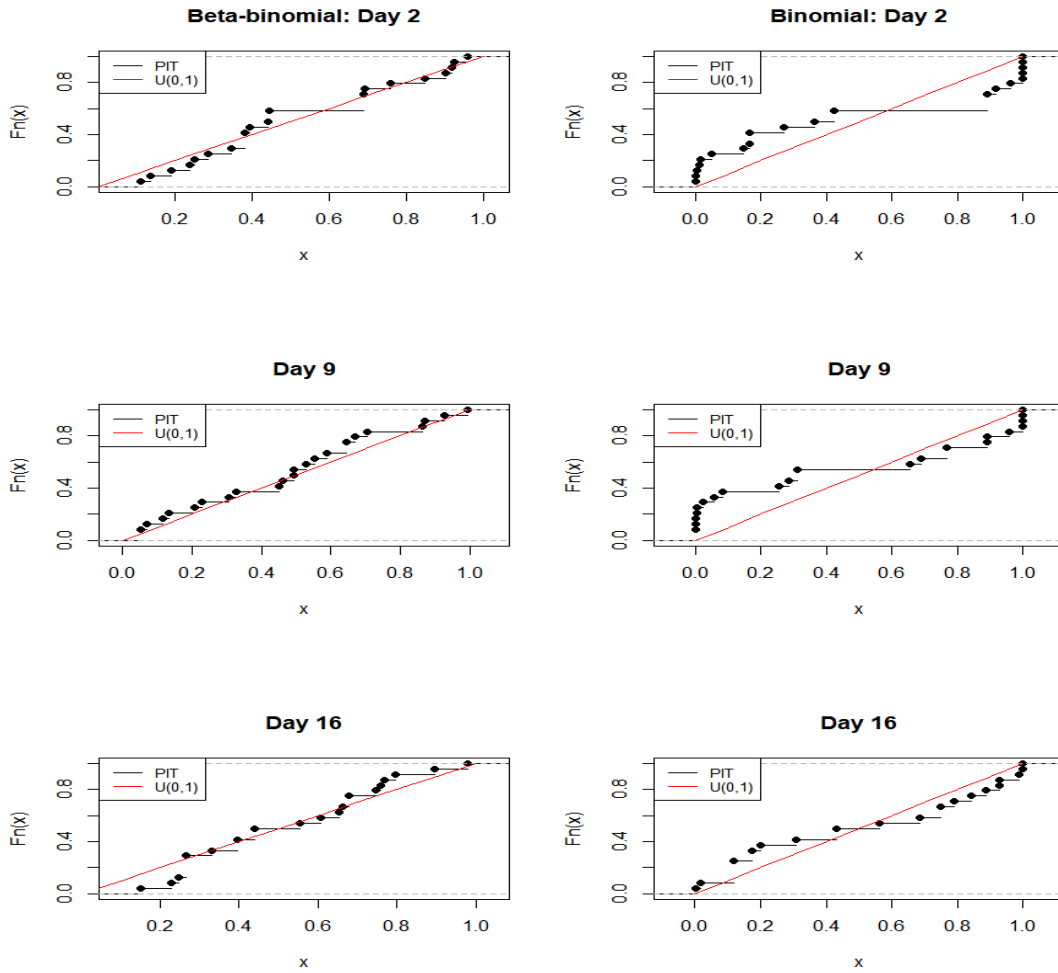


Figure 4.2: Study 2 - ECDF of Posterior Integral Transform



**Figure 4.3:** Study 3 - ECDF of Posterior Integral Transform

The ECDF plots show that the PIT values of the beta-binomial distribution are fairly close to the uniform distribution. The corresponding plots of the conventional binomial distribution show major departures from the uniform distribution. Therefore, the ECDF plots also suggest that the beta-binomial distribution is a more appropriate model for the data, than the binomial distribution.

# Chapter 5

## Conclusion

This study has been set out to investigate a Bayesian approach to evaluate the efficacy of animal ectoparasiticide treatments. More specifically, the integrated nested Laplace approximation (INLA) method was explored, and compared to the generalized linear model (conventional likelihood-based approach). Both methods were applied to data obtained from three trials which investigated the efficacy of an investigational veterinary product (IVP).

A brief overview of Bayesian statistics was presented, which included literature on prior selection and model discrimination statistics. Thereafter, the linear model and extensions thereof were discussed which included the GLM, and the HGLM. A brief discussion of overdispersion was provided, together with an overview of controlled animal studies of efficacy, and a statistical framework for efficacy evaluation.

The beta-binomial distribution was presented, as well as literature on the INLA method. A presentation of the beta-binomial distribution as a Bayesian HGLM was given, which therefore suggests that the beta-binomial distribution fits into

the INLA framework. As such, INLA can be applied to the beta-binomial distribution, to perform approximate Bayesian inference.

An investigation of the performance of INLA (when applied to veterinarian trials of IVPs) was carried out. The results obtained from the estimation of efficacy of an IVP using INLA were compared to those obtained from the (frequentist) GLM fit using PROC GENMOD. The GLM fit yields higher efficacy estimates compared to the Bayesian HGLM fit, and indicated that the survival counts are overdispersed (data clustering). However, with the assessment of model fits, the model selection statistics suggests that INLA is the preferred approach (over conventional binomial regression).

### **Disclaimer**

This work is based on the research supported wholly by the National Research Foundation of South Africa (Grant Numbers 102634). Any opinions, findings and conclusions or recommendations expressed in the material is that of the author(s) and the NRF does not accept any liability in the regard.



# Bibliography

Abbott, W. et al. (1925), ‘A method of computing the effectiveness of an insecticide’, *J. econ. Entomol* **18**(2), 265–267.

Ando, T. (2010), *Bayesian model selection and statistical modeling*, CRC Press.

Blangiardo, M. and Cameletti, M. (2015), *Spatial and spatio-temporal Bayesian models with R-INLA*, John Wiley & Sons.

CVMP (2005), ‘Specific efficacy requirements for ectoparasiticides in cattle’.

**URL:** [http://www.ema.europa.eu/docs/en\\_GB/document\\_library/Scientific\\_guideline/2009/10/WC500004643.pdf](http://www.ema.europa.eu/docs/en_GB/document_library/Scientific_guideline/2009/10/WC500004643.pdf)

CVMP (2016), ‘Guideline for the testing and evaluation of the efficacy of antiparasitic substances for the treatment and prevention of tick and flea infestation in dogs and cats’.

**URL:** [http://www.ema.europa.eu/docs/en\\_GB/document\\_library/Scientific\\_guideline/2016/07/WC500210927.pdf](http://www.ema.europa.eu/docs/en_GB/document_library/Scientific_guideline/2016/07/WC500210927.pdf)

Ennis, D. M. and Bi, J. (1998), ‘The beta-binomial model: Accounting for inter-trial variation in replicated difference and preference tests’, *Journal of Sensory Studies* **13**(4), 389–412.

Fong, Y., Rue, H. and Wakefield, J. (2010), ‘Bayesian inference for generalized linear mixed models’, *Biostatistics* **11**(3), 397–412.

Hinde, J. and Demétrio, C. G. (1998), ‘Overdispersion: models and estimation’, *Computational Statistics & Data Analysis* **27**(2), 151–170.

Holand, A. M., Steinsland, I., Martino, S. and Jensen, H. (2013), ‘Animal models and integrated nested laplace approximations’, *G3: Genes, Genomes, Genetics* **3**(8), 1241–1251.

**URL:** <http://www.g3journal.org/content/3/8/1241>

Kassahun, W. (2014), Modeling Hierarchical Data, Allowing for Overdispersion and Zero Inflation, in Particular Excess Zeros, PhD thesis.

Lee, Y. and Nelder, J. A. (1996), ‘Hierarchical generalized linear models’, *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 619–678.

Lee, Y. and Nelder, J. A. (2001), ‘Hierarchical generalised linear models: a synthesis of generalised linear models, random-effect models and structured dispersions’, *Biometrika* pp. 987–1006.

Marchiondo, A., Holdsworth, P., Fourie, L., Rugg, D., Hellmann, K., Snyder, D. and Dryden, M. (2013), ‘World association for the advancement of veterinary parasitology (waavp): guidelines for evaluating the efficacy of parasiticides for the treatment, prevention and control of flea and tick infestations on dogs and cats’, *Vet parasitol* **194**(1), 84–97.

Martinez, E. Z., Achcar, J. A. and Aragon, D. C. (n.d.), ‘Parameter estimation of the beta-binomial distribution: an application using the sas software’.

Martins, T. G. and Rue, H. (2014), ‘Extending integrated nested laplace approximation to a class of near-gaussian latent models’, *Scandinavian Journal of Statistics* **41**(4), 893–912.

Martins, T. G., Simpson, D., Lindgren, F. and Rue, H. (2013), ‘Bayesian computing with inla: new features’, *Computational Statistics & Data Analysis* **67**, 68–83.

McCullagh, P. and Nelder, J. A. (1989), *Generalized linear models (Second edition)*, London: Chapman & Hall.

McNeish, D. (2016), ‘On using bayesian methods to address small sample problems’, *Structural Equation Modeling: A Multidisciplinary Journal* **23**(5), 750–773.  
**URL:** <http://dx.doi.org/10.1080/10705511.2016.1186549>

Myer, R., Montgomery, D., Vining, G. and Robinson, T. (2010), *Generalized Linear Models: With Applications in Engineering and the Sciences: Second Edition*, John Wiley & Sons Inc.

Najera-Zuloaga, J., Lee, D.-J. and Arostegui, I. (2017), ‘Comparison of beta-binomial regression model approaches to analyze health-related quality of life data’, *Statistical Methods in Medical Research* p. 0962280217690413.

Ntzoufras, I. (2009), *Bayesian modeling using WinBUGS*, John Wiley & Sons Inc.

Rue, H., Martino, S. and Chopin, N. (2009), ‘Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations’, *Journal of the royal statistical society: Series b (statistical methodology)* **71**(2), 319–392.

Rue, H., Riebler, A., Srbye, S., Illian, J., Simpson, D. and Lindgren, F. (2017), ‘Bayesian computing with inla: A review’, *Annual Review of Statistics and its Application* **4**(1).

SAS Institute Inc. (2008), *SAS/STAT<sup>®</sup> 9.2 User's Guide*, SAS Institute Inc., Cary, North Carolina.

Schall, R., Burger, D. A. and Luus, H. G. (2016), 'Accuracy and precision of alternative estimators of ectoparasiticide efficacy', *Veterinary Parasitology* **223**(223), 50–56.

Schall, R. and Luus, H. G. (2011), 'Comparative evaluation of ectoparasiticide efficacy: Non-inferiority and superiority assessment', *Veterinary Parasitology* **11**(1), 173–179.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P. and Van der Linde, A. (2002), 'Bayesian measures of model complexity and fit (with discussion)', *Journal of the Royal Statistical Society* **64**, 583–640.

Walker, S. and Mallick, B. (1997), 'Hierarchical generalized linear models and frailty models with bayesian nonparametric mixing', *Journal of the Royal Statistical Society. Series B (Methodological)* **59**(4), 845–860.

# Appendix A

## Programming Code

### A.1 SAS<sup>®</sup> Code

```
option nodate center nobyline nonumber msglevel = I missing = " ";
%let ROOT = C:\Users\ctk68\Desktop\STSR7900_2017\LaTeX Thesis\Programs;
%let COVER = 95;
data ticksdata;
    set C:\Users\ctk68\Desktop\STSR7900_2017\LaTeX Thesis\Programs\Data\ticks.sas7bdat;
run;
proc format;
    value TREATX
        1 = "A"
        2 = "B"
        3 = "C";
run;
roc format;
    value TREATY
        1 = 'Control'
        2 = 'Reference'
        3 = 'Test';
run;
data TICK01;
    set TICKSDATA;
```

```

format TREAT TREATY.;
run;
proc means data = TICK01;
var TREAT;
output out = NULL01 max = MAXTRT;
run;
data NULL02;
set NULL01;
call symput("MAXTRT", strip(put(MAXTRT, BEST.)));
run;
%put &MAXTRT.;
ods trace on;
proc genmod data = TICK01;
by STUDY TIME;
class TREAT;
model Y/N = TREAT / dist = BINOMIAL link = LOG scale = PEARSON;
estimate 'log(pi R) - log(pi C)' TREAT -1 1 0 / e alpha = 0.05;
estimate 'log(pi T) - log(pi C)' TREAT -1 0 1 / e alpha = 0.05;
ods output ESTIMATES = TICK02;
run;
ods trace off;
data TICK03;
length RSL $50.;
set TICK02;
EST = 100*(1 - exp(LBETAESTIMATE));
LOW = 100*(1 - exp(LBETAUPPERCL));
UPP = 100*(1 - exp(LBETALOWERCL));
RSL = strip(put(EST, 10.1)) || " (" || strip(put(LOW, 10.1)) || "; " || strip(put(UPP, 10.1))
|| ")";
if LABEL = "log(pi R) - log(pi C)" then LABNO = 1; else
if LABEL = "log(pi T) - log(pi C)" then LABNO = 2;
run;
proc transpose data = TICK03 prefix = EFF out = TICK04;
by STUDY TIME;
var RSL;
id LABNO;
idlabel LABEL;
run;
data TICK05;
length DISPLAY $200.;

```

```
set TICK04;
by STUDY TIME;
DISPLAY = strip(put(STUDY, BEST.)) || " & " || strip(put(TIME, BEST.)) || " & " || strip(EFF1)
|| " & " || strip(EFF2) || " \\";
run;
title;
ods listing file = "&ROOT.\GENMOD_RESULT.txt";
proc report data = TICK05 (drop = _NAME_) noheader nowd;
column DISPLAY;
define DISPLAY / width = 65 " ";
run;
ods listing close;
```

## A.2 R Code

```
library(INLA)
setwd("C:/Users/ctk68/Desktop/STSR7900_2017/RData")
TICKS <- read.csv('Ticks.csv', sep=',')
attach(TICKS)

group <- factor(rep(c('1', '2', '3'), each=8))
blank <- rep(NA, 8)
n <- rep(50, 24)
nsims <- 10000

# 1 = test ; 2 = control ; 3 = ref
# Day 2
# Study 1

s1.d2.test <- c(y[1:8])
s1.d2.control <- c(y[9:16])
s1.d2.ref <- c(y[17:24])

s1.d2.mat <- matrix(c(s1.d2.test, blank, blank, blank, s1.d2.control, blank, blank, blank, s1.d2.ref)
, 24, 3)
s1.d2.data <- data.frame(s1.d2.mat, n, group)
```

```

s1.d2.model <- inla(s1.d2.mat ~ group -1,data=s1.d2.data,family=rep("betabinomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T),control.
  family = list(list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian', param =
  c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05))))
s1.d2.model <- inla.hyperpar(s1.d2.model)
  #Model Info
s1.d2.dic <- s1.d2.model$dic$dic
s1.d2.ML <- s1.d2.model$mlik
s1.d2.od <- s1.d2.model$summary.hyperpar$mean

plot(s1.d2.model,single=T)

s1.d2.sims <- inla.posterior.sample(nsims, s1.d2.model)

#Binomial
s1.d2.bin.model <- inla(s1.d2.mat ~ group -1,data=s1.d2.data,family=rep("binomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s1.d2.bin.model <- inla.hyperpar(s1.d2.bin.model)
  #Model Info
s1.d2.bin.dic <- s1.d2.bin.model$dic$dic
s1.d2.bin.ML <- s1.d2.bin.model$mlik

#Bin likelihood has no hyperparameters
s1.d2.bin.od <- s1.d2.bin.model$summary.hyperpar$mean

#Day 9
s1.d9.test <- c(y[25:32])
s1.d9.control <- c(y[33:40])
s1.d9.ref <- c(y[41:48])

s1.d9.mat <- matrix(c(s1.d9.test,blank,blank,blank,s1.d9.control,blank,blank,blank,s1.d9.ref)
  ,24,3)
s1.d9.data <- data.frame(s1.d9.mat,n,group)

s1.d9.model <- inla(s1.d9.mat ~ group -1,data=s1.d9.data,family=rep("betabinomial",3),Ntrials=n,
  control.compute = list(config=T, dic=T, cpo=T),control.family = list(list(prior = 'gaussian',
  param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
  param = c(0, 0.05))))
s1.d9.model <- inla.hyperpar(s1.d9.model)

```

```

#Model Info
s1.d9.dic <- s1.d9.model$dic$dic
s1.d9.ML <- s1.d9.model$mlik
s1.d9.od <- s1.d9.model$summary.hyperpar$mean

#Binomial
s1.d9.bin.model <- inla(s1.d9.mat ~ group -1,data=s1.d9.data,family=rep("binomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s1.d9.bin.model <- inla.hyperpar(s1.d9.bin.model)

s1.d9.bin.dic <- s1.d9.bin.model$dic$dic
s1.d9.bin.ML <- s1.d9.bin.model$mlik

s1.d9.sims <- inla.posterior.sample(nsims, s1.d9.model)
#summary(s1.d9.model)

#Day 16
s1.d16.test <- c(y[49:56])
s1.d16.control <- c(y[57:64])
s1.d16.ref <- c(y[65:72])

s1.d16.mat <- matrix(c(s1.d16.test,blank,blank,blank,s1.d16.control,blank,blank,blank,s1.d16.ref)
  ,24,3)
s1.d16.data <- data.frame(s1.d16.mat,n,group)

s1.d16.model <- inla(s1.d16.mat ~ group -1, data=s1.d16.data, family=rep("betabinomial",3),Ntrials
  =n,control.compute = list(config=T,cpo=T, dic=T),control.family = list(list(prior = 'gaussian
  ', param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian'
  , param = c(0, 0.05))))
s1.d16.model <- inla.hyperpar(s1.d16.model)

#Model Info
s1.d16.dic <- s1.d16.model$dic$dic
s1.d16.ML <- s1.d16.model$mlik
s1.d16.od <- s1.d16.model$summary.hyperpar$mean

#Binomial
s1.d16.bin.model <- inla(s1.d16.mat ~ group -1,data=s1.d16.data,family=rep("binomial",3),Ntrials=n
  ,control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s1.d16.bin.model <- inla.hyperpar(s1.d16.bin.model)

```

```

#Model Info
s1.d16.bin.dic <- s1.d16.bin.model$dic$dic
s1.d16.bin.ML <- s1.d16.bin.model$mlik

s1.d16.sims <- inla.posterior.sample(nsims, s1.d16.model)

#Day 23
s1.d23.test <- c(y[73:80])
s1.d23.control <- c(y[81:88])
s1.d23.ref <- c(y[89:96])

s1.d23.mat <- matrix(c(s1.d23.test,blank,blank,blank,s1.d23.control,blank,blank,blank,s1.d23.ref)
,24,3)
s1.d23.data <- data.frame(s1.d23.mat,n,group)

s1.d23.model <- inla(s1.d23.mat ~ group -1, data=s1.d23.data, family=rep("betabinomial",3),Ntrials
=n,control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian
', param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian'
, param = c(0, 0.05))))
s1.d23.model <- inla.hyperpar(s1.d23.model)

#Model Info
s1.d23.dic <- s1.d23.model$dic$dic
s1.d23.ML <- s1.d23.model$mlik
s1.d23.od <- s1.d23.model$summary.hyperpar$mean

#Binomial
s1.d23.bin.model <- inla(s1.d23.mat ~ group -1,data=s1.d23.data,family=rep("binomial",3),Ntrials=n
,control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s1.d23.bin.model <- inla.hyperpar(s1.d23.model)

#Model Info
s1.d23.bin.dic <- s1.d23.bin.model$dic$dic
s1.d23.bin.ML <- s1.d23.bin.model$mlik

s1.d23.sims <- inla.posterior.sample(nsims, s1.d23.model)

#Day 30
s1.d30.test <- c(y[97:104])
s1.d30.control <- c(y[105:112])
s1.d30.ref <- c(y[113:120])

```

```

s1.d30.mat <- matrix(c(s1.d30.test,blank,blank,blank,s1.d30.control,blank,blank,blank,s1.d30.ref)
,24,3)
s1.d30.data <- data.frame(s1.d30.mat,n,group)

s1.d30.model <- inla(s1.d30.mat ~ group -1,data=s1.d30.data,family=rep("betabinomial",3),Ntrials=n
,control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian',
param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
param = c(0, 0.05))))
s1.d30.model <- inla.hyperpar(s1.d30.model)

s1.d30.sims <- inla.posterior.sample(nsims, s1.d30.model)

#Model Info
s1.d30.dic <- s1.d30.model$dic$dic
s1.d30.ML <- s1.d30.model$mlik
s1.d30.od <- s1.d30.model$summary.hyperpar$mean

#Binomial
s1.d30.bin.model <- inla(s1.d30.mat ~ group -1,data=s1.d23.data,family=rep("binomial",3),Ntrials=n
,control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s1.d30.model <- inla.hyperpar(s1.d30.model)

#Model Info
s1.d30.bin.dic <- s1.d30.bin.model$dic$dic
s1.d30.bin.ML <- s1.d30.bin.model$mlik
#
#####

#Simulations
#
#####

s1.d2.pi0 <- s1.d2.pi1 <- s1.d2.pi2 <- s1.d2.r.efficacy <- s1.d2.t.efficacy <- rep(0,nsims)
s1.d9.pi0 <- s1.d9.pi1 <- s1.d9.pi2 <- s1.d9.r.efficacy <- s1.d9.t.efficacy <- rep(0,nsims)
s1.d16.pi0 <- s1.d16.pi1 <- s1.d16.pi2 <- s1.d16.r.efficacy <- s1.d16.t.efficacy <- rep(0,nsims)
s1.d23.pi0 <- s1.d23.pi1 <- s1.d23.pi2 <- s1.d23.r.efficacy <- s1.d23.t.efficacy <- rep(0,nsims)
s1.d30.pi0 <- s1.d30.pi1 <- s1.d30.pi2 <- s1.d30.r.efficacy <- s1.d30.t.efficacy <- rep(0,nsims)

```

```

for (i in 1:nsims) {
  #Day2
  s1.d2.pi0[i] <- plogis(s1.d2.sims[[i]]$latent[25]); s1.d2.pi1[i] <- plogis(s1.d2.sims[[i]]$
    latent[26]); s1.d2.pi2[i] <- plogis(s1.d2.sims[[i]]$latent[27]);
  s1.d2.r.efficacy[i] <- (1 - s1.d2.pi1[i]/s1.d2.pi0[i])*100 #ref
  s1.d2.t.efficacy[i] <- (1 - s1.d2.pi2[i]/s1.d2.pi0[i])*100 #test

  #Day9
  s1.d9.pi0[i] <- plogis(s1.d9.sims[[i]]$latent[25]); s1.d9.pi1[i] <- plogis(s1.d9.sims[[i]]$
    latent[26]); s1.d9.pi2[i] <- plogis(s1.d9.sims[[i]]$latent[27]);
  s1.d9.r.efficacy[i] <- (1 - s1.d9.pi1[i]/s1.d9.pi0[i])*100 #ref
  s1.d9.t.efficacy[i] <- (1 - s1.d9.pi2[i]/s1.d9.pi0[i])*100 #test

  #Day16
  s1.d16.pi0[i] <- plogis(s1.d16.sims[[i]]$latent[25]); s1.d16.pi1[i] <- plogis(s1.d16.sims[[i]]$
    latent[26]); s1.d16.pi2[i] <- plogis(s1.d16.sims[[i]]$latent[27]);
  s1.d16.r.efficacy[i] <- (1 - s1.d16.pi1[i]/s1.d16.pi0[i])*100 #ref
  s1.d16.t.efficacy[i] <- (1 - s1.d16.pi2[i]/s1.d16.pi0[i])*100 #test

  #Day23
  s1.d23.pi0[i] <- plogis(s1.d23.sims[[i]]$latent[25]); s1.d23.pi1[i] <- plogis(s1.d23.sims[[i]]$
    latent[26]); s1.d23.pi2[i] <- plogis(s1.d23.sims[[i]]$latent[27]);
  s1.d23.r.efficacy[i] <- (1 - s1.d23.pi1[i]/s1.d23.pi0[i])*100 #ref
  s1.d23.t.efficacy[i] <- (1 - s1.d23.pi2[i]/s1.d23.pi0[i])*100 #test

  #Day30
  s1.d30.pi0[i] <- plogis(s1.d30.sims[[i]]$latent[25]); s1.d30.pi1[i] <- plogis(s1.d30.sims[[i]]$
    latent[26]); s1.d30.pi2[i] <- plogis(s1.d30.sims[[i]]$latent[27]);
  s1.d30.r.efficacy[i] <- (1 - s1.d30.pi1[i]/s1.d30.pi0[i])*100 #ref
  s1.d30.t.efficacy[i] <- (1 - s1.d30.pi2[i]/s1.d30.pi0[i])*100 #test

}

#Mean Estimates and CI's
#Day2
s1.d2.r <- mean(s1.d2.r.efficacy)
s1.d2.rCI <- quantile(s1.d2.r.efficacy)
s1.d2.t <- mean(s1.d2.t.efficacy)

```

```
s1.d2.tCI <- quantile(s1.d2.t.efficacy ,c(0.025 ,0.975))

#Day9
s1.d9.r <- mean(s1.d9.r.efficacy)
s1.d9.rCI <- quantile( s1.d9.r.efficacy ,c(0.025 ,0.975))

s1.d9.t <- mean( s1.d9.t.efficacy)
s1.d9.tCI <- quantile( s1.d9.t.efficacy,c(0.025 ,0.975))
#Day16
s1.d16.r <- mean(s1.d16.r.efficacy)
s1.d16.rCI <- quantile( s1.d16.r.efficacy,c(0.025 ,0.975))

s1.d16.t <- mean( s1.d16.t.efficacy)
s1.d16.tCI <- quantile( s1.d16.t.efficacy ,c(0.025 ,0.975))
#Day23
s1.d23.r <- mean(s1.d23.r.efficacy)
s1.d23.rCI <- quantile( s1.d23.r.efficacy ,c(0.025 ,0.975))

s1.d23.t <- mean( s1.d23.t.efficacy)
s1.d23.tCI <- quantile(s1.d23.t.efficacy ,c(0.025 ,0.975))
#Day30
s1.d30.r <- mean(s1.d30.r.efficacy)
s1.d30.rCI <- quantile(s1.d30.r.efficacy ,c(0.025 ,0.975))

s1.d30.t <- mean( s1.d30.t.efficacy)
s1.d30.tCI <- quantile( s1.d30.t.efficacy,c(0.025 ,0.975))

#
#####

#Study 2
# 1 = test ; 2 = control ; 3 = ref

# Day 2
s2.d2.test <- c(y[121:128])
s2.d2.control <- c(y[129:136])
s2.d2.ref <- c(y[137:144])
```

```

s2.d2.mat <- matrix(c(s2.d2.test,blank,blank,blank,s2.d2.control,blank,blank,blank,s2.d2.ref)
,24,3)
s2.d2.data <- data.frame(s2.d2.mat,n,group)

s2.d2.model <- inla(s2.d2.mat ~ group -1, data=s2.d2.data, family=rep("betabinomial",3),Ntrials=n,
control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian',
param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
param = c(0, 0.05))))
s2.d2.model <- inla.hyperpar(s2.d2.model)

#Model Info
s2.d2.dic <- s2.d2.model$dic$dic
s2.d2.mD <- s2.d2.model$dic$mean.deviance
s2.d2.Dm <- s2.d2.model$dic$deviance.mean
s2.d2.ML <- s2.d2.model$mlik
s2.d2.od <- s2.d2.model$summary.hyperpar$mean

#Binomial
s2.d2.bin.model <- inla(s2.d2.mat ~ group -1,data=s2.d2.data,family=rep("binomial",3),Ntrials=n,
control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s2.d2.bin.model <- inla.hyperpar(s2.d2.bin.model)

#Model Info
s2.d2.bin.dic <- s2.d2.bin.model$dic$dic
s2.d2.bin.mD <- s2.d2.bin.model$dic$mean.deviance
s2.d2.bin.Dm <- s2.d2.bin.model$dic$deviance.mean
s2.d2.bin.ML <- s2.d2.bin.model$mlik
s2.d2.bin.od <- s2.d2.bin.model$summary.hyperpar$mean

s2.d2.sims <- inla.posterior.sample(nsims, s2.d2.model)

#Day 9
s2.d9.test <- c(y[145:152])
s2.d9.control <- c(y[153:160])
s2.d9.ref <- c(y[161:168])

s2.d9.mat <- matrix(c(s2.d9.test,blank,blank,blank,s2.d9.control,blank,blank,blank,s2.d9.ref)
,24,3)
s2.d9.data <- data.frame(s2.d9.mat,n,group)

```

```

s2.d9.model <- inla(s2.d9.mat ~ group -1, data=s2.d9.data, family=rep("betabinomial",3),Ntrials=n,
  control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian',
  param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
  param = c(0, 0.05))))
s2.d9.model <- inla.hyperpar(s2.d9.model)

#Model Info
s2.d9.dic <- s2.d9.model$dic$dic
s2.d9.mD <- s2.d9.model$dic$mean.deviance
s2.d9.Dm <- s2.d9.model$dic$deviance.mean
s2.d9.ML <- s2.d9.model$mlik
s2.d9.od <- s2.d9.model$summary.hyperpar$mean

#Binomial
s2.d9.bin.model <- inla(s2.d9.mat ~ group -1,data=s2.d9.data,family=rep("binomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s2.d9.bin.model <- inla.hyperpar(s2.d9.bin.model)
#Model Info
s2.d9.bin.dic <- s2.d9.bin.model$dic$dic
s2.d9.bin.mD <- s2.d9.bin.model$dic$mean.deviance
s2.d9.bin.Dm <- s2.d9.bin.model$dic$deviance.mean
s2.d9.bin.ML <- s2.d9.bin.model$mlik
s2.d9.bin.od <- s2.d9.bin.model$summary.hyperpar$mean

s2.d9.sims <- inla.posterior.sample(nsims, s2.d9.model)

#Day 16
s2.d16.test <- c(y[169:176])
s2.d16.control <- c(y[177:184])
s2.d16.ref <- c(y[185:192])

s2.d16.mat <- matrix(c(s2.d16.test,blank,blank,blank,s2.d16.control,blank,blank,blank,s2.d16.ref)
  ,24,3)
s2.d16.data <- data.frame(s2.d16.mat,n,group)

s2.d16.model <- inla(s2.d16.mat ~ group -1, data=s2.d16.data, family=rep("betabinomial",3),Ntrials
  =n,control.compute = list(config=T,dic=T,cpo=T),control.family = list(list(prior = 'gaussian'
  , param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
  param = c(0, 0.05))))
s2.d16.model <- inla.hyperpar(s2.d16.model)

```

```

#Model Info
s2.d16.dic <- s2.d16.model$dic$dic
s2.d16.mD <- s2.d16.model$dic$mean.deviance
s2.d16.Dm <- s2.d16.model$dic$deviance.mean
s2.d16.ML <- s2.d16.model$mlik
s2.d16.od <- s2.d16.model$summary.hyperpar$mean

#Binomial
s2.d16.bin.model <- inla(s2.d16.mat ~ group -1,data=s2.d16.data,family=rep("binomial",3),Ntrials=n
,control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s2.d16.bin.model <- inla.hyperpar(s2.d16.bin.model)

#Model Info
s2.d16.bin.dic <- s2.d16.bin.model$dic$dic
s2.d16.bin.mD <- s2.d16.bin.model$dic$mean.deviance
s2.d16.bin.Dm <- s2.d16.bin.model$dic$deviance.mean
s2.d16.bin.ML <- s2.d16.bin.model$mlik
s2.d16.bin.od <- s2.d16.bin.model$summary.hyperpar$mean

s2.d16.sims <- inla.posterior.sample(nsims, s2.d16.model)

#Day 23
s2.d23.test <- c(y[193:200])
s2.d23.control <- c(y[201:208])
s2.d23.ref <- c(y[209:216])

s2.d23.mat <- matrix(c(s2.d23.test,blank,blank,blank,s2.d23.control,blank,blank,blank,s2.d23.ref)
,24,3)
s2.d23.data <- data.frame(s2.d23.mat,n,group)

s2.d23.model <- inla(s2.d23.mat ~ group -1, data=s2.d23.data, family=rep("betabinomial",3),Ntrials
=n,control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian
', param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian'
, param = c(0, 0.05))))
s2.d23.model <- inla.hyperpar(s2.d23.model)

#Model Info
s2.d23.dic <- s2.d23.model$dic$dic
s2.d23.mD <- s2.d23.model$dic$mean.deviance

```

```

s2.d23.Dm <- s2.d23.model$dic$deviance.mean
s2.d23.ML <- s2.d23.model$mlik
s2.d23.od <- s2.d23.model$summary.hyperpar$mean

#Binomial
s2.d23.bin.model <- inla(s2.d23.mat ~ group -1,data=s2.d23.data,family=rep("binomial",3),Ntrials=n
,control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s2.d23.bin.model <- inla.hyperpar(s2.d23.bin.model)
#Model Info
s2.d23.bin.dic <- s2.d23.bin.model$dic$dic
s2.d23.bin.mD <- s2.d23.bin.model$dic$mean.deviance
s2.d23.bin.Dm <- s2.d23.bin.model$dic$deviance.mean
s2.d23.bin.ML <- s2.d23.bin.model$mlik
s2.d22.bin.od <- s2.d23.bin.model$summary.hyperpar$mean

s2.d23.sims <- inla.posterior.sample(nsims, s2.d23.model)

#Day 30
s2.d30.test <- c(y[217:224])
s2.d30.control <- c(y[225:232])
s2.d30.ref <- c(y[233:240])

s2.d30.mat <- matrix(c(s2.d30.test,blank,blank,blank,s2.d30.control,blank,blank,blank,s2.d30.ref)
,24,3)
s2.d30.data <- data.frame(s2.d30.mat,n,group)

s2.d30.model <- inla(s2.d30.mat ~ group -1, data=s2.d30.data, family=rep("betabinomial",3),Ntrials
=n,control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian
', param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian'
, param = c(0, 0.05))))
s2.d30.model <- inla.hyperpar(s2.d30.model)

#Model Info
s2.d30.dic <- s2.d30.model$dic$dic
s2.d30.mD <- s2.d30.model$dic$mean.deviance
s2.d30.Dm <- s2.d30.model$dic$deviance.mean
s2.d30.ML <- s2.d30.model$mlik
s2.d30.od <- s2.d30.model$summary.hyperpar$mean

```

```

#Binomial
s2.d30.bin.model <- inla(s2.d30.mat ~ group -1,data=s2.d30.data,family=rep("binomial",3),Ntrials=n
  ,control.predictor = list(compute=T),control.compute = list(config=T, dic=T, cpo=T))
s2.d30.bin.model <- inla.hyperpar(s2.d30.bin.model)
#Model Info
s2.d30.bin.dic <- s2.d30.bin.model$dic$dic
s2.d30.bin.mD <- s2.d30.bin.model$dic$mean.deviance
s2.d30.bin.Dm <- s2.d30.bin.model$dic$deviance.mean
s2.d30.bin.ML <- s2.d30.bin.model$mlik
s2.d30.bin.od <- s2.d30.bin.model$summary.hyperpar$mean

s2.d30.sims <- inla.posterior.sample(nsims, s2.d30.model)

#
#####

#Simulations
#
#####

s2.d2.pi0 <- s2.d2.pi1 <- s2.d2.pi2 <- s2.d2.r.efficacy <- s2.d2.t.efficacy <- rep(0,nsims)
s2.d9.pi0 <- s2.d9.pi1 <- s2.d9.pi2 <- s2.d9.r.efficacy <- s2.d9.t.efficacy <- rep(0,nsims)
s2.d16.pi0 <- s2.d16.pi1 <- s2.d16.pi2 <- s2.d16.r.efficacy <- s2.d16.t.efficacy <- rep(0,nsims)
s2.d23.pi0 <- s2.d23.pi1 <- s2.d23.pi2 <- s2.d23.r.efficacy <- s2.d23.t.efficacy <- rep(0,nsims)
s2.d30.pi0 <- s2.d30.pi1 <- s2.d30.pi2 <- s2.d30.r.efficacy <- s2.d30.t.efficacy <- rep(0,nsims)

for (i in 1:nsims) {
  #Day 2
  s2.d2.pi0[i] <- plogis(s2.d2.sims[[i]]$latent[25]); s2.d2.pi1[i] <- plogis(s2.d2.sims[[i]]$
    latent[26]); s2.d2.pi2[i] <- plogis(s2.d2.sims[[i]]$latent[27]);
  s2.d2.r.efficacy[i] <- (1 - s2.d2.pi1[i]/s2.d2.pi0[i])*100 #ref
  s2.d2.t.efficacy[i] <- (1 - s2.d2.pi2[i]/s2.d2.pi0[i])*100 #test

  #Day 9
  s2.d9.pi0[i] <- plogis(s2.d9.sims[[i]]$latent[25]); s2.d9.pi1[i] <- plogis(s2.d9.sims[[i]]$
    latent[26]); s2.d9.pi2[i] <- plogis(s2.d9.sims[[i]]$latent[27]);
  s2.d9.r.efficacy[i] <- (1 - s2.d9.pi1[i]/s2.d9.pi0[i])*100 #ref

```

```
s2.d9.t.efficacy[i] <- (1 - s2.d9.pi2[i]/s2.d9.pi0[i])*100 #test

#Day 16
s2.d16.pi0[i] <- plogis(s2.d16.sims[[i]]$latent[25]); s2.d16.pi1[i] <- plogis(s2.d16.sims[[i]]$
  latent[26]); s2.d16.pi2[i] <- plogis(s2.d16.sims[[i]]$latent[27]);
s2.d16.r.efficacy[i] <- (1 - s2.d16.pi1[i]/s2.d16.pi0[i])*100 #ref
s2.d16.t.efficacy[i] <- (1 - s2.d16.pi2[i]/s2.d16.pi0[i])*100 #test

#Day 23
s2.d23.pi0[i] <- plogis(s2.d23.sims[[i]]$latent[25]); s2.d23.pi1[i] <- plogis(s2.d23.sims[[i]]$
  latent[26]); s2.d23.pi2[i] <- plogis(s2.d23.sims[[i]]$latent[27]);
s2.d23.r.efficacy[i] <- (1 - s2.d23.pi1[i]/s2.d23.pi0[i])*100 #ref
s2.d23.t.efficacy[i] <- (1 - s2.d23.pi2[i]/s2.d23.pi0[i])*100 #test

#Day 30
s2.d30.pi0[i] <- plogis(s2.d30.sims[[i]]$latent[25]); s2.d30.pi1[i] <- plogis(s2.d30.sims[[i]]$
  latent[26]); s2.d30.pi2[i] <- plogis(s2.d30.sims[[i]]$latent[27]);
s2.d30.r.efficacy[i] <- (1 - s2.d30.pi1[i]/s2.d30.pi0[i])*100 #ref
s2.d30.t.efficacy[i] <- (1 - s2.d30.pi2[i]/s2.d30.pi0[i])*100 #test

#Mean estimates and CI's
#Day 2
s2.d2.r <- mean(s2.d2.r.efficacy)
s2.d2.rCI <- quantile(s2.d2.r ,c(0.025 ,0.975))

s2.d2.t <- mean(s2.d2.t.efficacy)
s2.d2.tCI <- quantile(s2.d2.t.efficacy ,c(0.025 ,0.975))
#Day9
s2.d9.r <- mean(s2.d9.r.efficacy)
s2.d9.rCI <- quantile(s2.d9.r ,c(0.025 ,0.975))

s2.d9.t <- mean(s2.d9.t.efficacy)
s2.d9.tCI <- quantile(s2.d9.t.efficacy ,c(0.025 ,0.975))
#Day16
s2.d16.r <- mean(s2.d16.r.efficacy)
s2.d16.rCI <- quantile(s2.d16.r ,c(0.025 ,0.975))

s2.d16.t <- mean(s2.d16.t.efficacy)
s2.d16.tCI <- quantile(s2.d16.t.efficacy ,c(0.025 ,0.975))
```

```

#Day23
s2.d23.r <- mean(s2.d23.r.efficacy)
s2.d23.rCI <- quantile(s2.d23.r ,c(0.025 ,0.975))

s2.d23.t <- mean(s2.d23.t.efficacy)
s2.d23.tCI <- quantile(s2.d23.t.efficacy ,c(0.025 ,0.975))
#Day30
s2.d30.r <- mean(s2.d30.r.efficacy)
s2.d30.rCI <- quantile(s2.d30.r ,c(0.025 ,0.975))

s2.d30.t <- mean(s2.d30.t.efficacy)
s2.d30.tCI <- quantile(s2.d30.t.efficacy ,c(0.025 ,0.975))

}

#
#####

#Study 3

#Day 2
s3.d2.test <- c(y[241:248])
s3.d2.control <- c(y[249:256])
s3.d2.ref <- c(y[257:264])

s3.d2.mat <- matrix(c(s3.d2.test,blank,blank,blank,s3.d2.control,blank,blank,blank,s3.d2.ref)
,24,3)
s3.d2.data <- data.frame(s3.d2.mat,n,group)

s3.d2.model <- inla(s3.d2.mat ~ group -1, data=s3.d2.data, family=rep("betabinomial",3),Ntrials=n,
control.compute = list(config=T, dic=T,cpo=T),control.family = list(list(prior = 'gaussian',
param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
param = c(0, 0.05))))
s3.d2.model <- inla.hyperpar(s3.d2.model)

#Model Info
s3.d2.dic <- s3.d2.model$dic$dic
s3.d2.mD <- s3.d2.model$dic$mean.deviance
s3.d2.Dm <- s3.d2.model$dic$deviance.mean

```

```
s3.d2.ML <- s3.d2.model$mlik
s3.d2.od <- s3.d2.model$summary.hyperpar$mean

#Binomial
s3.d2.bin.model <- inla(s3.d2.mat ~ group -1,data=s3.d2.data,family=rep("binomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T,cpo=T))
s3.d2.bin.model <- inla.hyperpar(s3.d2.bin.model)
#Model Info
s3.d2.bin.dic <- s3.d2.bin.model$dic$dic
s3.d2.bin.mD <- s3.d2.bin.model$dic$mean.deviance
s3.d2.bin.Dm <- s3.d2.bin.model$dic$deviance.mean
s3.d2.bin.ML <- s3.d2.bin.model$mlik
s3.d2.bin.od <- s3.d2.bin.model$summary.hyperpar$mean

s3.d2.sims <- inla.posterior.sample(nsims, s3.d2.model)

#Day 9
s3.d9.test <- c(y[265:272])
s3.d9.control <- c(y[273:280])
s3.d9.ref <- c(y[281:288])

s3.d9.mat <- matrix(c(s3.d9.test,blank,blank,blank,s3.d9.control,blank,blank,blank,s3.d9.ref)
  ,24,3)
s3.d9.data <- data.frame(s3.d9.mat,n,group)

s3.d9.model <- inla(s3.d9.mat ~ group -1, data=s3.d9.data, family=rep("betabinomial",3),Ntrials=n,
  control.compute = list(config=T,dic=T,cpo=T),control.family = list(list(prior = 'gaussian',
  param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian',
  param = c(0, 0.05))))
s3.d9.model <- inla.hyperpar(s3.d9.model)

#Model Info
s3.d9.dic <- s3.d9.model$dic$dic
s3.d9.mD <- s3.d9.model$dic$mean.deviance
s3.d9.Dm <- s3.d9.model$dic$deviance.mean
s3.d9.ML <- s3.d9.model$mlik
s3.d9.od <- s3.d9.model$summary.hyperpar$mean

#Binomial
```

```

s3.d9.bin.model <- inla(s3.d9.mat ~ group -1,data=s3.d9.data,family=rep("binomial",3),Ntrials=n,
  control.predictor = list(compute=T),control.compute = list(config=T, dic=T, cpo=T))
s3.d9.bin.model <- inla.hyperpar(s3.d9.bin.model)

#Model Info
s3.d9.bin.dic <- s3.d9.bin.model$dic$dic
s3.d9.bin.mD <- s3.d9.bin.model$dic$mean.deviance
s3.d9.bin.Dm <- s3.d9.bin.model$dic$deviance.mean
s3.d9.bin.ML <- s3.d9.bin.model$mlik
s3.d9.bin.od <- s3.d2.bin.model$summary.hyperpar$mean

s3.d9.sims <- inla.posterior.sample(nsims, s3.d9.model)

#Day 16
s3.d16.test <- c(y[289:296])
s3.d16.control <- c(y[297:304])
s3.d16.ref <- c(y[305:312])

s3.d16.mat <- matrix(c(s3.d16.test,blank,blank,blank,s3.d16.control,blank,blank,blank,s3.d16.ref)
  ,24,3)
s3.d16.data <- data.frame(s3.d16.mat,n,group)

s3.d16.model <- inla(s3.d16.mat ~ group -1, data=s3.d16.data, family=rep("betabinomial",3),Ntrials
  =n,control.compute = list(config=T,dic=T, cpo=T),control.family = list(list(prior = 'gaussian
  ', param = c(0, 0.05)), list(prior = 'gaussian', param = c(0, 0.05)), list(prior = 'gaussian'
  , param = c(0, 0.05))))
s3.d16.model <- inla.hyperpar(s3.d16.model)

#Model Info
s3.d16.dic <- s3.d16.model$dic$dic
s3.d16.mD <- s3.d16.model$dic$mean.deviance
s3.d16.Dm <- s3.d16.model$dic$deviance.mean
s3.d16.ML <- s3.d16.model$mlik
s3.d16.od <- s3.d16.model$summary.hyperpar$mean

#Binomial
s3.d16.bin.model <- inla(s3.d16.mat ~ group -1,data=s3.d16.data,family=rep("binomial",3),Ntrials=n
  ,control.predictor = list(compute=T),control.compute = list(config=T, dic=T, cpo=T))
s3.d16.bin.model <- inla.hyperpar(s3.d16.bin.model)

#Model Info

```

```

s3.d16.bin.dic <- s3.d16.bin.model$dic$dic
s3.d16.bin.mD <- s3.d16.bin.model$dic$mean.deviance
s3.d16.bin.Dm <- s3.d16.bin.model$dic$deviance.mean
s3.d16.bin.ML <- s3.d16.bin.model$mlik
s3.d16.bin.od <- s3.d2.bin.model$summary.hyperpar$mean

s3.d16.sims <- inla.posterior.sample(nsims, s3.d16.model)

#Simulations
s3.d2.pi0 <- s3.d2.pi1 <- s3.d2.pi2 <- s3.d2.r.efficacy <- s3.d2.t.efficacy <- rep(0,nsims)
s3.d9.pi0 <- s3.d9.pi1 <- s3.d9.pi2 <- s3.d9.r.efficacy <- s3.d9.t.efficacy <- rep(0,nsims)
s3.d16.pi0 <- s3.d16.pi1 <- s3.d16.pi2 <- s3.d16.r.efficacy <- s3.d16.t.efficacy <- rep(0,nsims)

for (i in 1:nsims) {
  #Day 2
  s3.d2.pi0[i] <- plogis(s3.d2.sims[[i]]$latent[25]); s3.d2.pi1[i] <- plogis(s3.d2.sims[[i]]$
    latent[26]); s3.d2.pi2[i] <- plogis(s3.d2.sims[[i]]$latent[27]);
  s3.d2.r.efficacy[i] <- (1 - s3.d2.pi1[i]/s3.d2.pi0[i])*100 #ref
  s3.d2.t.efficacy[i] <- (1 - s3.d2.pi2[i]/s3.d2.pi0[i])*100 #test

  #Day 9
  s3.d9.pi0[i] <- plogis(s3.d9.sims[[i]]$latent[25]); s3.d9.pi1[i] <- plogis(s3.d9.sims[[i]]$
    latent[26]); s3.d9.pi2[i] <- plogis(s3.d9.sims[[i]]$latent[27]);
  s3.d9.r.efficacy[i] <- (1 - s3.d9.pi1[i]/s3.d9.pi0[i])*100 #ref
  s3.d9.t.efficacy[i] <- (1 - s3.d9.pi2[i]/s3.d9.pi0[i])*100 #test

  #Day 16
  s3.d16.pi0[i] <- plogis(s3.d16.sims[[i]]$latent[25]); s3.d16.pi1[i] <- plogis(s3.d16.sims[[i]]$
    latent[26]); s3.d16.pi2[i] <- plogis(s3.d16.sims[[i]]$latent[27]);
  s3.d16.r.efficacy[i] <- (1 - s3.d16.pi1[i]/s3.d16.pi0[i])*100 #ref
  s3.d16.t.efficacy[i] <- (1 - s3.d16.pi2[i]/s3.d16.pi0[i])*100 #test

  #Mean estimates and CI's
  #Day 2
  s3.d2.r <- mean(s3.d2.r.efficacy)
  s3.d2.rCI <- quantile( s3.d2.r ,c(0.025 ,0.975))

  s3.d2.t <- mean(s3.d2.t.efficacy)
  s3.d2.tCI <- quantile(s3.d2.t ,c(0.025 ,0.975))

```

```

#Day 9
s3.d9.r <- mean(s3.d9.r.efficacy)
s3.d9.rCI <- quantile( s3.d9.r ,c(0.025 ,0.975))

s3.d9.t <- mean(s3.d9.t.efficacy)
s3.d9.tCI <- quantile(s3.d9.t ,c(0.025 ,0.975))

#Day 16
s3.d16.r <- mean(s3.d16.r.efficacy)
s3.d16.rCI <- quantile( s3.d16.r ,c(0.025 ,0.975))

s3.d16.t <- mean(s3.d16.t.efficacy)
s3.d16.tCI <- quantile(s3.d16.t ,c(0.025 ,0.975))

}

#
#####

#Export INLA results
#Study1
s1.efficacies <- rbind(s1.d2.r,s1.d2.t,s1.d9.r,s1.d9.t,s1.d16.r,s1.d16.t,s1.d23.r,s1.d23.t,s1.d30.
r,s1.d30.t)
s1.confints <- rbind(s1.d2.rCI,s1.d2.tCI,s1.d9.rCI,s1.d9.tCI,s1.d16.rCI,s1.d16.tCI,s1.d23.rCI,s1.
d23.tCI,s1.d30.rCI,s1.d30.tCI)
write.table(s1.efficacies,"S1.Efficacy.txt",sep="\t",row.names=F)
write.table(s1.confints,"S1.CI.txt",sep="\t",row.names=F)

s1.dic <- rbind(s1.d2.dic,s1.d9.dic,s1.d16.dic,s1.d23.dic,s1.d30.dic,s1.d2.bin.dic,s1.d9.bin.dic,
s1.d16.bin.dic,s1.d23.bin.dic,s1.d30.bin.dic)
s1.ML <- rbind(s1.d2.ML,s1.d9.ML,s1.d16.ML,s1.d23.ML,s1.d30.ML,s1.d2.bin.ML,s1.d9.bin.ML,s1.d16.
bin.ML,s1.d23.bin.ML,s1.d30.bin.ML)
s1.phi <- rbind(s1.d2.od,s1.d9.od,s1.d16.od,s1.d23.od,s1.d30.od)

write.table(s1.ML,"S1.bin.ML.txt",sep="\t",row.names=T, )
write.table(s1.dic,"S1.DIC.txt",sep="\t",row.names=T)
write.table(s1.phi,"S1.OD.txt",sep="\t",row.names=T)

```

```
#Study2
s2.efficacies <- rbind(s2.d2.r,s2.d2.t,s2.d9.r,s2.d9.t,s2.d16.r,s2.d16.t,s2.d23.r,s2.d23.t,s2.d30.
  r,s2.d30.t)
s2.confints <- rbind(s2.d2.rCI,s2.d2.tCI,s2.d9.rCI,s2.d9.tCI,s2.d16.rCI,s2.d16.tCI,s2.d23.rCI,s2.
  d23.tCI,s2.d30.rCI,s2.d30.tCI)
write.table(s2.efficacies,"S2.Efficacy.txt",sep="\t",row.names=T)
write.table(s2.confints,"S2.CI.txt",sep="\t",row.names=T)

s2.dic <- rbind(s2.d2.dic,s2.d9.dic,s2.d16.dic,s2.d23.dic,s2.d30.dic,s2.d2.bin.dic,s2.d9.bin.dic,
  s2.d16.bin.dic,s2.d23.bin.dic,s2.d30.bin.dic)
s2.ML <- rbind(s2.d2.ML,s2.d9.ML,s2.d16.ML,s2.d23.ML,s2.d30.ML,s2.d2.bin.ML,s2.d9.bin.ML,s2.d16.
  bin.ML,s2.d23.bin.ML,s2.d30.bin.ML)
s2.phi <- rbind(s2.d2.od,s2.d9.od,s2.d16.od,s2.d23.od,s2.d30.od)

write.table(s2.dic,"S2.dic.txt",sep="\t",row.names=T)
write.table(s2.ML,"S2.ML.txt",sep="\t",row.names=T)
write.table(s2.phi,"S2.OD.txt",sep="\t",row.names=T)

#Study3
s3.efficacies <- rbind(s3.d2.r,s3.d2.t,s3.d9.r,s3.d9.t,s3.d16.r,s3.d16.t)
s3.confints <- rbind(s3.d2.rCI,s3.d2.tCI,s3.d9.rCI,s3.d9.tCI,s3.d16.rCI,s3.d16.tCI)
write.table(s3.efficacies,"S3.Efficacy.txt",sep="\t",row.names=T)
write.table(s3.confints,"S3.CI.txt",sep="\t",row.names=T)

s3.dic <- rbind(s3.d2.dic,s3.d9.dic,s3.d16.dic,s3.d2.bin.dic,s3.d9.bin.dic,s3.d16.bin.dic)
s3.ML <- rbind(s3.d2.ML,s3.d9.ML,s3.d16.ML,s3.d2.bin.ML,s3.d9.bin.ML,s3.d16.bin.ML)
s3.od <- rbind(s3.d2.od,s3.d9.od,s3.d16.od)
write.table(s3.dic,"S3.dic.txt",sep="\t",row.names=T)
write.table(s3.ML,"S3.ML.txt",sep="\t",row.names=T )
write.table(s3.od,"S3.OD.txt",sep="\t",row.names=T)

#BB Graph
x <- seq(0,20)
a <- rep(c(0.2,2),2)
b <- c(0.2,2,2,0.2)
type <- c(1, 2, 1, 2)
labels <- c("a=0.2, b=0.2","a=2, b=2","a=0.2, b=2","a=2, b=0.2")
```

```

kalir <- c('darkblue','brown','black','red')
par(cex =1)
plot(x,lty=3,lwd=5,type='n',xlab="x",ylab ="f(x)",xlim=c(0,20),ylim=c(0,0.5))
for(i in 1:4){
  lines(x,dbetabinom.ab(x,20,a[i],b[i]),lwd=1.8, col=kalir[i],lty=type[i])
}
legend('top',labels,inset=.05,title="n=10",box.lwd= 0,box.col =" white ",lty=c(1, 2, 1, 2),col=
  kalir,cex=0.8)

#-----
#ECDF graphs
#-----

#Beta binomial
s1.d2.x <- s1.d2.model$cpo$pit
plot(ecdf(s1.d2.x),main="Beta-binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d9.x <- s1.d9.model$cpo$pit
plot(ecdf(s1.d9.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d16.x <- s1.d16.model$cpo$pit
plot(ecdf(s1.d16.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d23.x <- s1.d23.model$cpo$pit
plot(ecdf(s1.d23.x),main="Day 23")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d30.x <- s1.d30.model$cpo$pit
plot(ecdf(s1.d30.x),main="Day 30")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

#Binomial
s1.d2.bin.x <- s1.d2.bin.model$cpo$pit

```

```
plot(ecdf(s1.d2.bin.x),main="Binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d9.bin.x <- s1.d9.bin.model$cpo$pit
plot(ecdf(s1.d9.bin.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d16.bin.x <- s1.d16.bin.model$cpo$pit
plot(ecdf(s1.d16.bin.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d23.bin.x <- s1.d23.bin.model$cpo$pit
plot(ecdf(s1.d23.bin.x),main="Day 23")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s1.d30.bin.x <- s1.d30.bin.model$cpo$pit
plot(ecdf(s1.d30.bin.x),main="Day 30")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

#-----#

#Study 2

s2.d2.x <- s2.d2.model$cpo$pit
plot(ecdf(s2.d2.x),main="Beta-binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d9.x <- s2.d9.model$cpo$pit
plot(ecdf(s2.d9.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d16.x <- s2.d16.model$cpo$pit
```

```
plot(ecdf(s2.d16.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d23.x <- s2.d23.model$cpo$pit
plot(ecdf(s2.d23.x), main="Day 23")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d30.x <- s2.d30.model$cpo$pit
plot(ecdf(s2.d30.x),main="Day 30")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

#Binomial
s2.d2.bin.x <- s2.d2.bin.model$cpo$pit
plot(ecdf(s2.d2.bin.x),main="Binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d9.bin.x <- s2.d9.bin.model$cpo$pit
plot(ecdf(s2.d9.bin.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d16.bin.x <- s2.d16.bin.model$cpo$pit
plot(ecdf(s2.d16.bin.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d23.bin.x <- s2.d23.bin.model$cpo$pit
plot(ecdf(s2.d23.bin.x),main="Day 23")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s2.d30.bin.x <- s2.d30.bin.model$cpo$pit
plot(ecdf(s2.d30.bin.x),main="Day 30")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)
```

```
#-----#

#Study 3

s3.d2.x <- s3.d2.model$cpo$pit
plot(ecdf(s3.d2.x),main="Beta-binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s3.d9.x <- s3.d9.model$cpo$pit
plot(ecdf(s3.d9.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s3.d16.x <- s3.d16.model$cpo$pit
plot(ecdf(s3.d16.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

#Binomial

s3.d2.bin.x <- s3.d2.bin.model$cpo$pit
plot(ecdf(s3.d2.bin.x ),main="Binomial: Day 2")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s3.d9.bin.x <- s3.d9.bin.model$cpo$pit
plot(ecdf(s3.d9.bin.x),main="Day 9")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)

s3.d16.bin.x <- s3.d16.bin.model$cpo$pit
plot(ecdf(s3.d16.bin.x),main="Day 16")
lines(seq(0, 1, by=.1), punif(seq(0, 1, by=.1)), col=2)
legend('topleft',legend=c("PIT","U(0,1)"),col=c("black", "red"), lty=1, cex=0.8)
```