university of groningen

faculty of science and engineering

A Bayesian bivariate response mixed-effects model for zero-inflated count data containing large outliers

Master Project Mathematics

October 2020

Student: Y.B. van Oppen

First supervisor: Dr. M.A. Grzegorczyk

Second supervisor: Dr. W.P. Krijnen

Abstract

Although numerous techniques for modeling count data have been proposed in a univariate-response setting, generalizations to two or more dimensions are scarce. This thesis proposes a bivariate response mixed-effects approach with a Bayesian specification for modeling zero-inflated, two-dimensional count data containing large outliers. A zero-inflated bivariate geometric distribution is derived and reparameterized it in terms of its marginal medians, which can subsequently be modeled directly using both fixed and random effects. This configuration allows increasing robustness with respect to large observations as the zero-inflation fraction grows to 0.5. The model's development is motivated by an observational study on green hawker populations in the northern Netherlands. The covariates include the host plant's presence and abiotic factors relating to the water condition. Two competing ways of measuring population sizes are compared in a bivariate response setting to simultaneously compare the covariates' effects on the marginal sizes as well as their correlation. Doing so provides a new perspective on their (dis)similarity given external factors. The relatively small sample size, the inclusion of repeated measurements, and the presence of large outliers incite the need for a Bayesian mixed-effects model that is comparatively insensitive to excessively high counts. An extension of JAGS with a custom distribution module to estimate the posterior parameter distribution using a Metropolis-Hastings MCMC algorithm is implemented and tested. With it, the model's validity and practical use are demonstrated in a simulation study. An application to the empirical data suggests that the correlation between the population measurements solely depends on the thickness of the sludge layer formed by decayed host plants. The extent to which the measures correlate is estimated to be 25% lower at minimal than median to maximal amounts of sludge. Using average covariate levels, the predicted correlations in 2015–2017 are 0.15-0.18 lower at minimal amounts of sludge, implying the measures are not interchangeable in such conditions.

Contents

Acknowledgements

I want to thank my supervisors Marco Grzegorczyk and Wim Krijnen for countless talks on statistical modeling ideas and meaningful interpretation of the results. Their highly involved and supportive supervision has been indispensable in this research.

I am also very grateful to have been given the opportunity to work on novel data from important research by Bureau Biota. Spending four summers doing fieldwork must take quite some perseverance, and I feel honored to provide the statistical analysis. Special thanks should be given to Gabi Milder-Mulderij, Rink Wiggers, and Christoff Brochard for their patient and enthusiastic attitude towards the, admittedly, lengthy process of finding an adequate model.

Furthermore, I want to express my gratitude towards Team HPC for letting me burden the Peregrine cluster with countless simulations. The timely completion of this thesis would have been impossible without their services.

Last but foremost, I want to thank my friends, family, and significant other, for their endless support and encouragement motivated me to overcome any obstacles I feared would be insurmountable.

General introduction

The motivating research problem of this thesis was to define a tailored model for the observational study (Milder-Mulderij et al., 2019) on dragonfly (green hawker) populations in the northern Netherlands by Bureau Biota (hereafter referred to as the client). The study covers 17 locations supervised by five managers from 2015–2018. It includes measurements on the dragonflies (counts, subdivided into categories pertaining to sex, growth stage, and occupation upon recording), the water surface coverage of their host plant (water soldier), the thickness of the sludge layer formed by decayed water soldiers, and some abiotic factors such as the water's oxygen or acidity levels. The study's central hypothesis is that a novel host plant management strategy could benefit the dragonfly prevalence. The current practice is to remove the vegetation on one side of the ditch, but it is speculated that doing so in a 'zipper' (or two-row 'checkerboard') pattern is superior. The underlying rationale is that more vegetation border length is created, which is the preferred spot for the green hawkers, and that the removal pattern is more reminiscent of nature. An essential aspect of the newly suggested procedure is that it should not cost significantly more resources than currently needed.

However, it turns out that the data do not suggest that the new strategy leads to increased dragonfly prevalence. There is an indication that it decreases the sludge layer's thickness, but a new study with an adjusted removal pattern would be required to assess this decrease's practical significance. That does not mean no interesting ecological insights may be obtained from analyzing the data. Apart from finding a well-fitting and parsimonious model for the dragonfly counts, some quantities could be modeled bivariately. The green hawkers were counted in the traditional way of tallying live specimen, but also using a new method by collecting exuviae: exoskeletons shed by the dragonflies upon transitioning from the larval to the adult stage. There is some disagreement on which approach should be preferred (Hardersen et al., 2017). It is possible to gain more detailed insight into the interrelationship between the counting methods by fitting a bivariate response model that contains a correlation parameter. Modeling this parameter should allow us to uncover covariates that influence the counts' correlation.

The fact that the available data set is of small size, includes repeated measurements, and contains large outliers insinuates that a Bayesian mixed-effects model with appropriate robustness could provide more reliable results than maximum likelihood estimates for conventional bivariate count models. It turns out that such a model has not been defined or implemented yet in literature. For this reason, this thesis is divided into two parts. The first is dedicated to rigorously defining the model, implementing an estimating procedure using the Bayesian analysis program JAGS, and verifying its validity and practicability using a simulation study. The second part comprises a statistical data analysis using the observational study in Milder-Mulderij et al. (2019),

following a conventional structure by starting with a data preparation and description, followed by an exploratory and a formal analysis, and concluded with a examination of the model validity. The formal analysis also includes the results of fitting the proposed bivariate Bayesian mixed effects model to the data, as well as a brief discussion of the possible ecological implications.

The contribution of this thesis is intended to extend beyond the statistical analysis of the green hawker populations. In principle, the proposed bivariate Bayesian mixed-effects model is applicable to any count data that are more or less geometrically distributed and supports both negative and positive correlation, zero-inflation, and large outliers. It is possible to generalize the approach by deriving a bivariate counterpart of the discrete Weibull distribution, which is also parameterizable by its median, to accommodate under- or over-dispersion as well. Such models could also be applied to count data with components that follow a Poisson or a negative binomial distribution, and the components' distributions would not have to be of similar shape. Moreover, regardless of whether the underlying distribution is geometric or discrete Weibull, there is a potential extension to more than two dimensions, should future research demand such models.

This thesis is organized as follows. Chapter 1 describes the robust Bayesian bivariate response model together with the necessary preliminaries and presents the estimation and simulation study results. The statistical analysis of the dragonfly populations is presented in Chapter 2. Appendix A contains additional tables and figures that would clutter the text when shown in place. All relevant source code is listed in Appendix B, along with the necessary implementation details. Finally, a report to the client (in Dutch and targeted at a more general audience) is attached in Appendix C.

Chapter 1

A Bayesian bivariate response mixed model

1.1 Introduction

Linear modeling is a simple and powerful tool in statistical analysis, but it is inadequate when dealing with count, binary value, or survival time responses. Moreover, in the presence of repeated measurements, one often needs to include a large number of parameters to satisfy the requirement of conditionally independent observations. To deal with these matters, it is often necessary to consider *Generalized Linear Mixed Models* (*GLMM*s) (Bolker, 2015). Compared to alternatives such as neural networks (Ellacott et al., 2012), segmented regression (Bellman and Roth, 1969; McZgee and Carleton, 1970) or cubic splines (Rice and Rosenblatt, 1983), and support vector machines (Friedman et al., 2001, chap. 12), GLMMs are relatively simple and easy to interpret, while simultaneously offering sufficient flexibility to be applicable in a broad range of data analyses (Brooks et al., 2019; Faes et al., 2006; Wang et al., 2015). In many cases, it is natural to consider multiple variables as responses and model them concurrently; for example, by considering responses at different time points (Das et al., 2016), multidimensional quality measures (Rivaz and Khaledi, 2017), or multiple genera within a family of organisms (Chauvet et al., 2019). Numerous techniques for defining and estimating multivariate-response GLMMs are already documented. Gueorguieva (2001) and Jaffa et al. (2016) propose to define a separate GLMM for each response, where the components' random effects are assumed to follow some joint distribution. The `MCMCglmm` package in R (Hadfield et al., 2010) offers estimation by including a covariance matrix that dictates the so-called trait structure, where a trait either corresponds to a response or a zero-inflation parameter. The approach is strictly Bayesian, but Dworkin and Bolker (2019) presents a trick to define models similarly using the `lme4` package in R. Yet another option is to define a copula (Nikoloulopoulos, 2015; Zhang et al., 2019) which completely describes the responses' covariance structure, but the results are generally hard to interpret.

Issues commonly arise when explicitly dealing with count data. Besides often being overdispersed or zero-inflated, such data frequently contain large outliers, which can severely influence model estimates. It is not always valid to exclude them from the analysis, so an alternative procedure may be required that is robust with respect to nonconforming data points. In a Bayesian setting, Burger et al. (2020) present a mixed-effects model for zero-inflated and highly skewed longitudinal data using a discrete Weibull distribution, which is robust with re-

spect to extreme outliers through direct modeling of median instead of mean counts. At the moment, such a model does not exist for a bivariate response. Famoye (2019) likely comes closest by proposing a bivariate exponentiated-exponential geometric regression model, albeit without including random effects or a Bayesian specification. This model supports positively or negatively correlated, zero-altered, and under- or over-dispersed count data. However, no robustness with respect to excessively large observations is possible through median modeling, as there is no closed-form expression for the marginal medians. A way to obtain a bivariate-response model with the desired properties could be to find a bivariate discrete Weibull distribution. However, this thesis explores a simpler alternative by considering a bivariate geometric distribution. As a special case of the discrete Weibull distribution, the geometric distribution is also parameterizable in terms of its median. Therefore, a bivariate variant may be expressed according to its marginal medians, besides a correlation parameter. Even though doing so sacrifices the opportunity to account for additional under- or over-dispersion, the resulting model is simpler by excluding two distribution parameters.

The inclusion of random effects complicates the likelihood maximization process in frequentist inference, as the likelihood needs to be marginalized with respect to these effects before it may be optimized. The frequent lack of a closed-form expression for the associated integral necessitates the approximation of the integrand. Popular approaches are penalized quasi-likelihood or quasi-likelihood, Gauss-Hermite quadrature, and Laplace approximation (Tuerlinckx et al., 2006; Bolker et al., 2009). The latter can efficiently be implemented using packages such as `TMB` (Template Model Builder, see Kristensen et al. (2015)) in R. This complication does not occur in Bayesian inference since the additional random effects distribution becomes another level in a hierarchical model, which is one of the main reasons we will give a Bayesian specification of the proposed model. Note, however, that frequentist inference remains possible through direct marginalized likelihood maximization. Often, imposing a Bayesian model often means that the posterior distribution of the parameters needs to be estimated through lengthy MCMC algorithms, as a closed-form expression of the normalizing constant is rarely available. The upside is that the results are readily interpretable, make use of all available information, and do not rely as heavily on large sample properties as their frequentist counterparts do.

The motivating data stems from the observational study by Milder-Mulderij et al. (2019) on green hawker (a species of dragonfly) populations in the northern Netherlands. The dragonflies were counted in two ways, where the novel way is speculated to be superior, although there is some disagreement (Hardersen et al., 2017). An appropriate bivariate response model may yield valuable insight into their (dis)similarity while simultaneously modeling the counts in terms of the available covariates. A bivariate geometric distribution is likely a good fit, as univariate geometric distributions appropriately model the responses separately (cf. Section 2.4). The critical aspect is that the data, besides being highly skewed and zero-inflated, contain large outliers. For this reason, it makes sense to model the marginal medians instead of the marginal means. The application of the model to empirical data will be postponed until Section 2.5. Rather, this chapter will include various simulations to demonstrate model identifiability and superiority compared to a competing model in an idealized situation.

This chapter starts with a review of preliminaries on statistical modeling, generalized linear (mixed) models, frequentist inference, MCMC algorithms, and Bayesian inference in Section 1.2. A concrete model is subsequently defined in Section 1.3 and tested in a simulation study in Section 1.4. Critical aspects of the results are discussed in Section 1.5, after which some concluding remarks on what has been achieved and further research comprise Section 1.6.

1.2 Preliminaries

1.2.1 Notation

The following notation (that is not universal in mathematics) will be used:

- $\mathbf{x} \cdot \mathbf{y}$: the *dot product* of two vectors $\mathbf{x}, \mathbf{y}$ of equal dimension;
- $\mathbb{1}\{A\}$: the *indicator* function that equals 1 if A is true and 0 otherwise;
- $\mathrm{N}(\mu, \sigma^2)$: the *normal* distribution with mean μ and variance σ^2;
- $\mathrm{Unif}(\Omega)$: the *uniform* distribution on a set Ω;
- $\mathrm{Poi}(\lambda)$: the *Poisson* distribution with rate λ;
- $\mathrm{Geom}(p)$: the *geometric* distribution on $0, 1, 2, \ldots$ with success probability p, counting the number of failures before the first success;
- $\mathrm{MN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$: the *multivariate normal* distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$;
- $\mathrm{W}(\mathbf{V}, n)$: the *Wishart* distribution with scale matrix $\mathbf{V}$ and n degrees of freedom.;

1.2.2 Generalized linear (mixed) models

The remainder of this section serves to provide the necessary background on generalized linear mixed models, maximum likelihood estimation, and Bayesian inference. A basic understanding of probability, statistics, and measure theory is assumed, for which Billingsley (2012) may serve as a reference.

Statistical models and data

There are numerous ways to define a generalized linear (mixed) model. Philosophical definitions are sometimes criticized for imparing intuition, whereas more practical definitions often lack complete generality. We try to find some middle ground by maintaining a moderately abstract formulation while providing examples along the way to emphasize its practical use. The following resembles the exposition in Sections 1 and 3.1 of McCullagh (2002).

Definition 1. A *statistical model* is a pair $(\mathcal{S}, \mathcal{P})$, where $\mathcal{S}$ is a sample space and $\mathcal{P}$ is a set of probability distributions on $\mathcal{S}$. In nearly all cases, $\mathcal{P} := \mathcal{P}(\Theta)$ is parameterized by some parameter space Θ, so that each element $\mathscr{P} := \mathscr{P}_{\boldsymbol{\theta}} \in \mathcal{P}$ is identified by a unique parameter vector $\boldsymbol{\theta} \in \Theta$. The set $\mathcal{P}$ is chosen to contain a distribution that sufficiently resembles the unknown, true distribution on $\mathcal{S}$.

Remark 1. A *Bayesian model* differs from a statistical model by additionally including a prior distribution $\mathscr{Q}$ on Θ. Therefore, it is given by a triple $(\mathcal{S}, \mathcal{P}(\Theta), \mathscr{Q})$ instead.

Example 1. One of the simplest nontrivial examples of a statistical model has $\mathcal{S} = \{0, 1\}$ and the set of Bernoulli distributions on $\mathcal{S}$ as $\mathcal{P}$. Note that $\mathcal{P}$ is necessarily parameterized by $\Theta \subset [0, 1]$.

Assume there is a countable set of *statistical units* $\mathcal{U}$, where we label the elements of $\mathcal{U}$ as $1, 2, \ldots$ Although covariates may technically be random, it is conventional to consider them as fixed, taking values in some *covariate space* $\mathcal{X}$. The statistical units are related to the covariate space by a map $x : \mathcal{U} \to \mathcal{X}$, sending each unit $i \in \mathcal{U}$ to a point $x_i \in \mathcal{X}$. We refer to a quantity that takes values on a (not necessarily strict) subspace of $\mathcal{X}$ as a *covariate* (also called *regressor*, *explanatory variable*, *predictor* or *independent variable*).

Similarly, we denote the (not necessarily one-dimensional) *response space* by $\mathcal{Y}$, which is related to the statistical units through a map $y : \mathcal{U} \to \mathcal{Y}$ that sends each $i \in \mathcal{U}$ to a point $y_i \in \mathcal{Y}$. The sample space $\mathcal{S}$ is the collection of all such maps. We refer to a quantity taking values on a subspace of $\mathcal{Y}$ as a *response* (also called *outcome* or *dependent variable*). In all practical cases, the set of probability distributions $\mathcal{P} := \mathcal{P}_x$ on $\mathcal{S}$ will depend on the map x.

Example 2. Suppose $\mathcal{U} = \{1, 2\}$ and $\mathcal{X} = \mathcal{Y} = \{0, 1\}$. There are four possible configurations for $x : \mathcal{U} \to \mathcal{X}$, as each $i \in \mathcal{U}$ can independently be mapped to either 0 or 1. Similarly, $\mathcal{X}$ admits 2^n possible maps $\mathcal{U} \to \mathcal{X}$ when there are $n \in \mathbb{N}$ statistical units. Such is the case when there is a single predictor that indicates membership in one of two categories. In empirical studies, 1 and 0 often respectively represent the treatment and the control group.

The same holds for $y : \mathcal{U} \to \mathcal{Y}$. Returning to the case where $\mathcal{U} = \{1, 2\}$, the sample space $\mathcal{S}$ is bijective to $\{0, 1\} \times \{0, 1\}$, whereby $\mathcal{P}_x$ must be a subset of the set of bivariate Bernoulli distributions. A popular modeling choice is a logistic regression model; letting $(s_1, s_2) \in \mathcal{S}$, it is given by

$$\mathbb{P}\{s_i = 1\} = \frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)} \quad \text{for} \quad i = 1, 2 \tag{1.1}$$

for $(\beta_0, \beta_1) \in \mathbb{R}^2 =: \Theta$. In this model, the components of (s_1, s_2) are independent, so (1.1) completely specifies $\mathcal{P}_x$.

We will refer to a *data set* $\mathcal{D}$ as a collection of points (y_i, x_i), where y_i and x_i are the respective images of $i \in \mathcal{U}$ under $x : \mathcal{U} \to \mathcal{X}$ and $y : \mathcal{U} \to \mathcal{Y}$ for some set of statistical units $\mathcal{U}$. Out of practical considerations, $\mathcal{U}$ will be of finite size $n \in \mathbb{N}$. The points in $\mathcal{D}$ will be called *observations*. In light of the statistical model definition given above, this insinuates that the points $y_1, \ldots, y_n$ are realizations of some random elements $Y_1, \ldots, Y_n$, which will be implicitly assumed if not stated. Note that we need not specify whether the points $x_1, \ldots, x_n$ are values attained by random elements, as we are concerned with distributions on $\mathcal{S}$ conditional on x. It is perfectly reasonable to regard some covariates as deterministic (experimental design factors, for instance) and others as random elements (such as uncontrollable exogenous variables).

In practice, the observations in a data set $\mathcal{D}$ will be bijective to a subset of $\mathbb{R}^k$ for some $k \in \mathbb{N}$, so it is without loss of generality to assume that $\mathcal{D}$ has a matrix representation in $\mathbb{R}^{n \times k}$. This assertion is straightforward to see when some covariates or responses attain discrete values, as we can simply label the values using the natural numbers. Higher-dimensional (e.g., complex-valued) variables can be decomposed into one-dimensional components, and variables that may become infinite in magnitude can be augmented with indicators that signify finiteness. Such a representation is necessary when a data set is to be analyzed using computer software.

Generalized linear models

Although not strictly required, response distributions are typically members of the *exponential family* of distributions in generalized linear models. Distributions in this family have various useful properties, like being expressable in terms of sufficient statistics and having closed-form

expressions for their expected value, variance, and information. We give a definition for completeness' sake, but as zero-inflated distributions generally do not lie in the exponential family, we redirect the interested reader to (Dobson and Barnett, 2018, ch. 3).

Definition 2. The *exponential family* is the set of distributions with cumulative distribution function $F(\mathbf{x} \mid \boldsymbol{\theta})$ satisfying

$$dF(\mathbf{x} \mid \boldsymbol{\theta}) = \exp(\boldsymbol{\eta}(\boldsymbol{\theta}) \cdot \mathbf{T}(\mathbf{x}) - A(\boldsymbol{\theta}))\, dH(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^k, \quad \boldsymbol{\theta} \in \Theta \subset \mathbb{R}^\ell,$$

where $\boldsymbol{\eta} : \Theta \to \mathbb{R}^m$ is the *natural parameter*, $\mathbf{T} : \mathbb{R}^k \to \mathbb{R}^m$ is a *sufficient statistic*, $H : \mathbb{R}^m \to \mathbb{R}$ has non-negative partial derivatives, and $k, \ell, m \in \mathbb{N}$. The function H is a Lebesgue-Stieltjes integrator for the reference measure of the exponential family that it generates. The *normalizing function* $A : \Theta \to \mathbb{R}$ is implicitly defined by $\boldsymbol{\eta}$, $\mathbf{T}$ and H.

Example 3. For fixed $n \in \mathbb{N}$ and $p \in (0, 1)$, the CDF $F(x \mid p)$ of the binomial distribution $\text{Bin}(n, p)$ has

$$dF(x \mid p) = \exp\left(\log\left(\frac{p}{1-p}\right) \cdot x + n \log(1-p)\right)\, dH(x), \quad H(x) = \sum_{i=0}^{n} \binom{n}{i} \mathbb{1}\{x \geqslant i\}.$$

It follows that for a known number of trials, the binomial distribution belongs to the exponential family.

Suppose we have a data set of $n \in \mathbb{N}$ observations $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i)\}_{i=1}^n$, where each $\mathbf{y}_i \in \mathbb{R}^m$ is a value attained by a random vector $\mathbf{Y}_i$ and each $x_i \in \mathbb{R}^k$ for some $k \in \mathbb{N}$. For $m = 1$, a *Generalized Linear Model* (GLM) is a model of the form

$$\mathbf{Y}_i := Y_i \sim \mathscr{P}_i, \quad g(\mathbb{E}\, Y_i) = \mathbf{x}_i \cdot \boldsymbol{\beta} \quad (\boldsymbol{\beta} \in \mathbb{R}^k) \quad \text{for} \quad i = 1, \ldots, n. \tag{1.2}$$

The *linear predictor* $\mathbf{x}_i \cdot \boldsymbol{\beta}$ is denoted by η_i for each i, and the *link function* g must be invertible and differentiable. One typically takes the laws $\mathscr{P}_i$ to lie in the exponential family, parameterized by elements $\boldsymbol{\theta}_i$ of some parameter space Θ. That is,

$$\mathscr{P}_i := \mathscr{P}(\boldsymbol{\theta}_i) \quad \text{for} \quad i = 1, \ldots, n.$$

Moreover, the Y_i are assumed to be independent. For maximum likelihood inference, laws $\mathscr{P}_i$ must be parameterizable in terms of the means $\mathbb{E}\, Y_i$, and any remaining nuisance parameters must to either be fixed or be simultaneously maximized.

Example 4. A *linear regression model* is a special case of a GLM, where

$$Y_i \sim \text{N}(\mu_i, \sigma^2), \quad g(\mu_i) = \mu_i = \mathbf{x}_i \cdot \boldsymbol{\beta} \quad \text{for} \quad i = 1, \ldots, n,$$

for some variance σ^2 that is constant across observations.

When dealing with multivariate or multiple responses, it is natural to model multiple parameters concurrently. Thus, it makes sense to extend GLMs to models of the form

$$\mathbf{Y}_i \sim \mathscr{P}(\theta_{i1}, \ldots, \theta_{i\ell}), \quad g_j(\theta_{ij}) = \mathbf{x}_i \cdot \boldsymbol{\beta}_j \quad (\boldsymbol{\beta}_j \in \mathbb{R}^k) \quad \text{for} \quad \begin{matrix} i = 1, \ldots, n, \\ j = 1, \ldots, \ell. \end{matrix} \tag{1.3}$$

Analogously, each linear predictor $\mathbf{x}_i \cdot \boldsymbol{\beta}_j$ is denoted η_{ij}, each link function g_j must be invertible and differentiable, the laws $\mathscr{P}(\theta_{i1}, \ldots, \theta_{i\ell})$ typically form a subset of the exponential family, and the $\mathbf{Y}_i$ are assumed to be independent.

Example 5. For independent Poisson random variables U, V, W with respective means λ, μ, ν,

$$\mathbf{Y} := (U + W, V + W)$$

follows a *bivariate Poisson distribution* with parameters λ, μ, ν (Marshall and Olkin, 1985). The covariance of the components is

$$\operatorname{Cov}(U + W, V + W) = \operatorname{Cov}(W, W) = \nu.$$

Assuming fixed, unit covariance, an extended GLM of the form (1.3) may be given by

$$\mathbf{Y}_i \sim \mathscr{P}(\lambda_i, \mu_i), \quad \log \lambda_i = \mathbf{x}_i \cdot \boldsymbol{\beta}_1, \quad \log \mu_i = \mathbf{x}_i \cdot \boldsymbol{\beta}_2, \quad \text{for} \quad i = 1, \ldots, n,$$

where $\mathscr{P}(\lambda_i, \mu_i)$ denotes the bivariate Poisson distribution with parameters $\lambda_i, \mu_i, 1$.

The parameter vectors $\boldsymbol{\beta}_j$ are referred to as *fixed effects*. These effects describe how changes in covariate levels affect the parameter θ_{ij} and, hence, the predictive distribution of the Y_i. The precise nature of the effects on the parameter of interest depends on the link function.

Random effects

In a generalized linear model, the responses are assumed to be independent. This assumption might be violated if a covariate is not included in the model. Suppose there is a discrete covariate that attains a large number of values. Adding a parameter β_{ij} for each value usually renders the model too complex, especially when dealing with a multitude of such covariates. A solution is to consider the parameters as *random effects*. The idea is to assume the parameters are random variables following some distribution $\mathscr{R}(\boldsymbol{\xi})$, parameterized by a parameter vector $\boldsymbol{\xi}$. It is convenient if $\mathscr{R}(\boldsymbol{\xi})$ describes a family of distributions centered at zero since this allows a covariate effect to be decomposed into a fixed effect and random deviations. For this reason, it is common to assume $\mathscr{R}(\boldsymbol{\xi}) = \mathscr{R}(\boldsymbol{\Sigma}) = \mathrm{MN}(\mathbf{0}, \boldsymbol{\Sigma})$ for some covariance matrix $\boldsymbol{\Sigma}$, but this is by no means the only option. A model containing both fixed and random effects is called a *mixed-effects model*. In particular, a GLM will be called a *Generalized Linear Mixed Model* (GLMM) in this setting.

Example 6. Suppose the weight and blood pressure of ten patients is measured at three visits during a study, so the observations will be of the form

$$(\text{pressure}_i, \text{weight}_i, \text{visit}_i, \text{patient}_i) \quad \text{for} \quad i = 1, \ldots, 30.$$

Let blood pressure be a response, weight and visit number be fixed effects, and the patient identifier be a random effect. Denoting the random variable of which pressure_i are observed values by Y_i for each i, if we would just like to account for a blood pressure offset per patient, our model would be of the form

$$\begin{aligned} Y_i \sim \mathscr{P}(\mu_i) \quad \text{with} \quad \mu_i &:= \mathbb{E}\, Y_i, \\ g(\mu_i) = \beta_0 + \beta_1 \text{weight}_i + \beta_2 \mathbb{1}\{\text{visit}_i = 2\} + \beta_3 \mathbb{1}\{\text{visit}_i = 3\} &+ \sum_{j=1}^{10} u_j \mathbb{1}\{\text{patient}_i = j\}, \\ (u_1, \ldots, u_{10}) \sim \mathscr{R}(\boldsymbol{\xi}) & \end{aligned} \tag{1.4}$$

for $i = 1, \ldots, 30$. This way, the only random effects are the *random intercepts* $u_1, \ldots, u_{10}$. In practice, it is useful to assume the u_j are i.i.d., and to simplify the final line in (1.4) to

$$u_1, \ldots, u_{10} \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$$

for some fixed variance σ^2. One may suspect there is also a variable weight effect per patient, in which case we may change $g(\mu_i)$ to

$$g(\mu_i) = \beta_0 + \cdots + \sum_{j=1}^{10} u_j \mathbb{1}\{\text{patient}_i = j\} + \sum_{j=1}^{10} v_j \mathbb{1}\{\text{patient}_i = j\} \times \text{weight}_i$$

in (1.4). The new assumption on the random effects u_j, v_j becomes

$$(u_1, \ldots, u_{10}, v_1, \ldots, v_{10}) \sim \mathscr{R}'(\boldsymbol{\xi}). \tag{1.5}$$

Again, it is practical to assume the (u_j, v_j) are i.i.d. So for instance, we may simplify (1.5) to

$$(u_1, v_1), \ldots, (v_{10}, v_{10}) \overset{\text{iid}}{\sim} \mathrm{MN}\,(\mathbf{0}, \boldsymbol{\Sigma})$$

for some covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{2\times 2}$.

For the moment, let us restrict to a single grouping factor z_i, so that $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i, z_i)\}_{i=1}^n$. Note the vectors $(\mathbf{x}_i, z_i)$ are now the observed values for the covariates. Assume that each $z_i \in \{1, \ldots, d\}$ for some $d \in \mathbb{N}$. Let $\mathbf{w}_i$ denote the vector containing the components of $\mathbf{x}_i$ that relate to the random effects for each i. A GLMM is a model of the form

$$\begin{aligned} \mathbf{Y}_i &\sim \mathscr{P}(\theta_{i1}, \ldots, \theta_{i\ell}), \\ g_j(\theta_{ij}) &= \mathbf{x}_i \cdot \boldsymbol{\beta}_j + \mathbf{w}_i \cdot (\mathbf{B}_j)_{z_i}, \\ \mathrm{vec}(\mathbf{B}_j) &\sim \mathscr{R}(\boldsymbol{\xi}_j) \qquad\qquad \text{for} \quad i = 1, \ldots, n, \quad j = 1, \ldots, \ell. \end{aligned} \tag{1.6}$$

The $\mathbf{B}_j$ are random effect matrices, with each column corresponding to a group and each row to a covariate. The notation $(\mathbf{B}_j)_{z_i}$ represents column z_i of $\mathbf{B}_j$, and $\mathrm{vec}(\mathbf{B}_j)$ is the vectorization of $\mathbf{B}_j$. Since it is generally impractical to consider nonzero correlations between random effects of different groups, we will always use the assumption

$$(\mathbf{B}_j)_1, \ldots, (\mathbf{B}_j)_d \overset{\text{iid}}{\sim} \mathscr{R}'(\boldsymbol{\xi}'_j) \quad \text{for} \quad j = 1, \ldots, \ell,$$

and often, even

$$(\mathbf{B}_j)_1, \ldots, (\mathbf{B}_j)_d \overset{\text{iid}}{\sim} \mathrm{MN}(\mathbf{0}, \boldsymbol{\Sigma}_j) \quad \text{for} \quad j = 1, \ldots, \ell, \tag{1.7}$$

for covariance matrices $\boldsymbol{\Sigma}_j$ of appropriate dimensions.

Remark 2. If we restrict our attention to the random effects, the model specification is equivalent to a Bayesian model in which a prior distribution is imposed on the parameters, with corresponding hyperparameters that are to be optimized.

Example 7. In Example 6, with notation as in (1.6), we have

$$\mathbf{w}_i = (1, \text{weight}_i), \quad \mathbf{B} := \mathbf{B}_1 = \begin{pmatrix} u_1 & \cdots & u_{10} \\ v_1 & \cdots & v_{10} \end{pmatrix}, \quad \text{and} \quad (\mathbf{B})_j = (u_j, v_j)$$

for each i, j.

Random effects need not be constrained to a single grouping factor; in principle, an arbitrarily complex structure may be imposed. That is, random effects may be nested or crossed, or a combination of the two. The nesting-crossing configuration in a hierarchy of random effects is a property of the experimental design.

Example 8. Suppose the patients in Example 6 were distributed among five hospitals but had each of their visits at the same hospital. In that case, the random effects for the hospital and the patient are nested. They would have had to be crossed if some patients had had visits at different hospitals. Had age groups been included, it would be likely that hospitals and age groups were crossed, while patients remained nested under hospital-age groups. The distinction is important when nested groups are not uniquely labeled with respect to groups nested in other groups, e.g., when each hospital takes two patients, labeled patient one and patient two. Incorrectly treating the patients' and hospitals' random effects as crossed implies there are only two patients who visit each hospital three times.

When a unique identifier is chosen for each group, there is no distinction between nesting and crossing, so the model with multiple random effect grouping factors is given by inductively extending (1.6) as it extends (1.3).

1.2.3 Maximum likelihood estimation

Conventional GLM fitting is carried out through likelihood maximization. We define the likelihood using the following concept.

Definition 3. Suppose two measures μ, ν are defined on a measurable space $(\Omega, \mathcal{F})$, and that μ dominates ν. Then there is a non-negative μ-measurable function h such that

$$\nu(A) = \int_A h \, d\mu \quad \text{for all} \quad A \in \mathcal{F}.$$

The function $h := \frac{d\nu}{d\mu}$ is called the *Radon-Nikodym derivative* of ν with respect to μ. If ν is a probability measure, we will also refer to h as the *probability function*, without explicitly mentioning the measure μ.

Remark 3. The Radon-Nikodym derivative is (among other things) a generalization of the concepts of a probability mass or density function. A straightforward application is to obtain probability functions for mixtures of discrete and continuous distributions. For the CDF F of a univariate continuous distribution with probability measure $\mathbb{P}$, it reduces to the probability density function f, since

$$F(x) = \mathbb{P}\{(-\infty, x]\} = \int_{(-\infty, x]} f \, d\lambda = \int_{-\infty}^{x} f(t) \, dt$$

if λ is the Lebesgue measure on $\mathbb{R}$. A similar reasoning can be applied to discrete distributions when replacing λ by the counting measure.

Example 9. Suppose a fraction $p \in (0, 1)$ of the probability mass of a distribution $\mathscr{P}$ is concentrated at zero, and the remainder is uniformly distributed on the open unit interval. Consider the measure μ that is the sum of the Lebesgue measure on $\mathbb{R}$ and the measure ν on $\mathbb{R}$ defined by $\nu(A) = \mathbb{1}\{0 \in A\}$ for $A \in \mathbb{R}$. Then μ dominates the probability measure $\mathbb{P}$ of $\mathscr{P}$ for all p, and so the Radon-Nikodym derivative of $\mathbb{P}$ with respect to μ is defined by

$$\frac{d\mathbb{P}}{d\mu}(x) = p\mathbb{1}\{x = 0\} + (1 - p)\mathbb{1}\{x \in (0, 1)\} \quad \text{for} \quad x \in \mathbb{R}.$$

Definition 4. Assume a distribution $\mathscr{P}(\boldsymbol{\theta})$ is parameterized by $\boldsymbol{\theta} \in \Theta$, and that a value y has been observed for a random variable $Y \sim \mathscr{P}(\boldsymbol{\theta})$. Let $\mathbb{P}_{\boldsymbol{\theta}}$ denote the probability measure associated $\mathscr{P}(\boldsymbol{\theta})$, and let μ be some measure that dominates $\mathbb{P}_{\boldsymbol{\theta}}$ for each $\boldsymbol{\theta} \in \Theta$. The *likelihood* $\mathscr{L}(\boldsymbol{\theta} \mid y)$ of $\boldsymbol{\theta}$ given y is

$$\mathscr{L}(\boldsymbol{\theta} \mid y) := \frac{d\mathbb{P}_{\boldsymbol{\theta}}}{d\mu}(y).$$

Unless stated otherwise, it will be implicit that the likelihood is defined with respect to a domination measure μ, which will always be the Lebesgue or the counting measure for continuous or discrete distributions, respectively. The corresponding *maximum likelihood estimate* $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$ is defined as

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \mathscr{L}(\boldsymbol{\theta} \mid y).$$

Optimizing with respect to all random effects is complicated by their distributional restriction. Moreover, doing so would defeat the purpose of reducing the number of parameters. However, we may maximize a *marginal likelihood* instead, obtained from the marginal distribution in which any random effects have been integrated out.

Definition 5. Assume a distribution $\mathscr{P}(\boldsymbol{\theta}, \boldsymbol{\zeta})$ with probability measure $\mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\zeta}}$ is parameterized by $\boldsymbol{\theta}, \boldsymbol{\zeta}$, and that $\boldsymbol{\zeta} \in \mathcal{Z}$ follows a distribution $\mathscr{R}(\boldsymbol{\xi})$ with probability measure $\mathbb{P}_{\boldsymbol{\xi}}$. Suppose y is an observed value for $Y \sim \mathscr{P}(\boldsymbol{\theta}, \boldsymbol{\zeta})$ with support $\mathcal{Y}$. Then the *marginal likelihood* $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\xi} \mid y)$ is

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\xi} \mid y) := \frac{d\mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\xi}}}{d\mu}(y), \quad \text{where} \quad \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\xi}}\{A\} = \int_{\mathcal{Z}} \mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\zeta}}\{A\}\, d\mathbb{P}_{\boldsymbol{\xi}}(\boldsymbol{\zeta}) \quad \text{for} \quad A \in \mathcal{Y}$$

and μ is some measure dominating $\mathbb{P}_{\boldsymbol{\theta},\boldsymbol{\xi}}$ for all $\boldsymbol{\theta}, \boldsymbol{\xi}$.

When modeling data $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i, z_i)\}_{i=1}^n$ with a GLMM of the form (1.6), the $\mathbf{Y}_i$ are assumed to be independent, and so

$$\mathcal{L}(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_\ell, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_\ell \mid \mathbf{y}_1, \ldots, \mathbf{y}_n) = \prod_{i=1}^n \mathcal{L}(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_\ell, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_\ell \mid \mathbf{y}_i). \tag{1.8}$$

For ease of exposition, we consider only the case where $\ell = 1$, i.e., a single parameter is modeled. It should be clear how to extend the following discussion to multiple parameter modeling. If we assume a random effects distribution (1.7), the marginal likelihoods in the right-hand side of (1.8) have the simpler form

$$\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\Sigma} \mid \mathbf{y}_i) = \int_{\mathbb{R}^r} \mathcal{L}(\boldsymbol{\beta}, (\mathbf{B})_{z_i} \mid \mathbf{y}_i)\, d\Phi_{\boldsymbol{\Sigma}}\left((\mathbf{B})_{z_i}\right), \tag{1.9}$$

where r is the number of random effects, and $\Phi_{\boldsymbol{\Sigma}}$ is the CDF of the $\mathrm{MN}(\mathbf{0}, \boldsymbol{\Sigma})$ distribution.

Except for a few special cases, the integral in (1.9) does not have an analytic solution. Since $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ are unknown, it is inefficient to approximate the integral in (1.9) numerically. Although still possible, a grid search quickly becomes too computationally demanding as the number of fixed or random effects grows. It is more appropriate to replace the integral by an approximating function of $\boldsymbol{\beta}, \boldsymbol{\Sigma}$, such as a *Laplace approximation*. Denote the PDF of the $\mathrm{MN}(\mathbf{0}, \boldsymbol{\Sigma})$ distribution by $\phi_{\boldsymbol{\Sigma}}$ and let

$$f(\mathbf{b}_i) := \mathcal{L}(\boldsymbol{\beta}, \mathbf{b}_i \mid \mathbf{y}_i)\, \phi_{\boldsymbol{\Sigma}}(\mathbf{b}_i) \quad \text{for} \quad \mathbf{b}_i \in \mathbb{R}^r,$$

where $\mathbf{b}_i$ abbreviates $(\mathbf{B})_{z_i}$ for each i. The second-order Taylor series expansion of $\log f$ about

$$\hat{\mathbf{b}}_i := \arg\max_{\mathbf{b}_i} \log f(\mathbf{b}_i) \tag{1.10}$$

is given by

$$\log f(\mathbf{b}_i) \approx \log f\big(\hat{\mathbf{b}}_i\big) + \tfrac{1}{2}\big(\mathbf{b}_i - \hat{\mathbf{b}}_i\big) \cdot \mathbf{H}\big(\hat{\mathbf{b}}_i\big)\big(\mathbf{b}_i - \hat{\mathbf{b}}_i\big). \tag{1.11}$$

In (1.11), $\mathbf{H}$ denotes the Hessian matrix of $\log f$, that is,

$$\mathbf{H}(\mathbf{x}) = \frac{d^2 \log\left(\mathcal{L}(\boldsymbol{\beta}, \mathbf{b}_i \mid \mathbf{y}_i)\, \phi_{\boldsymbol{\Sigma}}\left(\mathbf{b}_i\right)\right)}{d\mathbf{b}_i d\mathbf{b}_i^T}(\mathbf{x}) \quad \text{for} \quad \mathbf{x} \in \mathbb{R}^r.$$

This approximation yields the Laplace approximation as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\Sigma} \mid \mathbf{y}_i) &= \int_{\mathbb{R}^r} \exp(\log f(\mathbf{b}_i))\, d\mathbf{b}_i \\
&\approx f\big(\hat{\mathbf{b}}_i\big) \int_{\mathbb{R}^r} \exp\left(\tfrac{1}{2}\big(\mathbf{b}_i - \hat{\mathbf{b}}_i\big) \cdot \mathbf{H}\big(\hat{\mathbf{b}}_i\big)\big(\mathbf{b}_i - \hat{\mathbf{b}}_i\big)\right) d\mathbf{b}_i \\
&= (2\pi)^{r/2} \mathcal{L}\big(\boldsymbol{\beta}, \hat{\mathbf{b}}_i \,\big|\, \mathbf{y}_i\big) \phi_{\boldsymbol{\Sigma}}\big(\hat{\mathbf{b}}_i\big) \big| -\mathbf{H}\big(\hat{\mathbf{b}}_i\big)\big|^{-\frac{1}{2}}
\end{aligned} \tag{1.12}$$

since the integrand on the second line is proportional to the PDF of the $\text{MN}\big(\hat{\mathbf{b}}_i, -\mathbf{H}\big(\hat{\mathbf{b}}_i\big)^{-1}\big)$ distribution.

Higher-order Taylor series approximations may also be used to improve accuracy (Raudenbush et al., 2000). Maximizing the approximation (1.12) involves defining the optimum $\hat{\mathbf{b}}_i$ as a function of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$. If doing so is challenging or infeasible, one may alternatively use current estimates of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ in an iterative scheme as in adaptive Gauss-Hermite quadrature (Tuerlinckx et al., 2006).

1.2.4 Model validation using randomized quantile residuals

A typical choice for GL(M)M validation is a visual inspection of deviance or Pearson residuals. However, in contrast to the linear case, the residuals may be far from Gaussian with homogeneous variance. The situation becomes further complicated when the response attains only a few values, and parallel lines are formed that make it impossible to distinguish a good from a bad fit by looking at residual plots (Dunn and Smyth, 1996). To circumvent this problem, we introduce a set of standardized residuals called *randomized quantile residuals*. Note that the following discussion applies to a broader family of models, which includes GLMs and GLMMs.

Suppose we have modeled data $\mathcal{D} = \{(y_i, \mathbf{x}_i)\}_{i=1}^n$, where Y_i is a random variable with attained value y_i for each i, as

$$Y_i \sim \mathscr{P}(\boldsymbol{\theta}_i) \quad \text{for} \quad i = 1, \ldots, n,$$

where each $\boldsymbol{\theta}_i$ is an element of some parameter space Θ that may depend on $\mathbf{x}_i$. Fix $i \in \{1, \ldots, n\}$, and assume that $\mathscr{P}_i^*$ is the unknown, true distribution of Y_i with CDF F_i^*. If each $\mathscr{P}_i^*$ is continuous, then

$$F_i^*(Y_i) \sim \text{Unif}([0, 1]).$$

Let $F_{\boldsymbol{\theta}_i}$ denote the CDF of $\mathscr{P}(\boldsymbol{\theta}_i)$, and let $\hat{\boldsymbol{\theta}}_i$ be the maximum likelihood estimate for $\boldsymbol{\theta}_i$. If $\mathscr{P}(\hat{\boldsymbol{\theta}}_i)$ is a good approximation of $\mathscr{P}_i^*$, then approximately,

$$F_{\hat{\boldsymbol{\theta}}_i}(Y_i) \sim \text{Unif}([0, 1]).$$

The value $F_{\hat{\boldsymbol{\theta}}_i}(Y_i)$ is called a *quantile residual*.

On the other hand, let a be a discontinuity in the support of F_i^*. If there are no other discontinuities, then the distribution of $F_i^*(\mathbf{Y}_i)$ will have a probability mass

$$\mathbb{P}\{Y_i = a\} = F_i^*(a) - F_i^*(a-).$$

at $F_i^*(a)$, and the remainder will be distributed uniformly on $[0, F_i^*(a-)] \cup [F_i^*(a), 1]$ (here, $F_i^*(a-)$ is the *left limit* of F_i^* at a). Define

$$U_i^* \sim \text{Unif}\left([F_i^*(Y_i-), F_i^*(Y_i)]\right),$$

so that U_i^* is distributed as $F_i^*(Y_i)$ on $[0, F_i^*(a-)] \cup (F_i^*(a), 1]$. More importantly, this definition ensures that

$$U_i^* \sim \text{Unif}([0,1]).$$

Similarly, if $\mathscr{P}(\hat{\theta}_i)$ is a good approximation of $\mathscr{P}_i^*$, the same holds approximately for

$$\hat{U}_i \sim \text{Unif}\left(\left[F_{\hat{\boldsymbol{\theta}}_i}(Y_i-), F_{\hat{\boldsymbol{\theta}}_i}(Y_i)\right]\right).$$

Note that the definitions of U_i^* and $\hat{U}_i$ are applicable to any countable number of discontinuities in F_i^*, and their distributions remain (approximately) uniform on the unit interval. We refer to $\hat{U}_i$ as a *randomized quantile residual*.

Using the data points in $\mathcal{D}$, we can form a set of standardized residuals $\hat{u}_i$ from the observed values y_i, that is,

$$\hat{u}_i \sim \text{Unif}\left(\left[F_{\hat{\boldsymbol{\theta}}_i}(y_i-), F_{\hat{\boldsymbol{\theta}}_i}(y_i)\right]\right).$$

This approach is adopted in Hartig (2020), where the CDFs $F_{\hat{\boldsymbol{\theta}}_i}$ are approximated by empirical CDFs. The empirical CDFs are computed using values simulated from the predictive distributions $\mathscr{P}(\hat{\boldsymbol{\theta}}_i)$. We may extend this method to multivariate response models using multivariate CDFs, justified in a manner analogous to the above discussion.

1.2.5 Bayesian inference

A Bayesian variant of the GLMM (1.6) imposes an additional prior distribution $\mathscr{Q}$ on the parameter vectors $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_\ell, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_\ell$. In practice, the $\boldsymbol{\xi}_j$ will often be covariance matrices of multivariate normal distributions with zero mean vectors, for which a Wishart prior distribution is a popular choice. It is common to take a multivariate normal prior distribution for the fixed effects $\boldsymbol{\beta}_j$.

Bayesian inference is primarily concerned with posterior distributions of the parameters given the observed data $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i)\}_{i=1}^n$ instead of fixed parameter estimates. Recall that the $\mathbf{y}_i$ are assumed to be realizations of random quantities $\mathbf{Y}_i$ with common support $\mathcal{Y}$. Denoting

$$(\boldsymbol{\beta}, \boldsymbol{\xi}) := (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_\ell, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_\ell)$$

and the corresponding parameter space by $\mathcal{B} \times \Xi$, let $\mathbb{P}$ be a probability measure on $\mathcal{Y}^n \times \mathcal{B} \times \Xi$ for which the marginal probability measure on $\mathcal{B} \times \Xi$ agrees with $\mathscr{Q}$. Bayes' rule is used to define the posterior distribution $\mathbb{P}\{\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n\}$ as

$$\mathbb{P}\{(\boldsymbol{\beta}, \boldsymbol{\xi}) \in A \mid (\mathbf{Y}_1, \ldots, \mathbf{Y}_n) = (\mathbf{y}_1, \ldots, \mathbf{y}_n)\} = \frac{\mathbb{P}\{(\mathbf{Y}_1, \ldots, \mathbf{Y}_n) = (\mathbf{y}_1, \ldots, \mathbf{y}_n), (\boldsymbol{\beta}, \boldsymbol{\xi}) \in A\}}{\mathbb{P}\{(\mathbf{Y}_1, \ldots, \mathbf{Y}_n) = (\mathbf{y}_1, \ldots, \mathbf{y}_n)\}} \tag{1.13}$$

for $A \in \mathcal{B} \times \Xi$.

Taking a measure μ that dominates $\mathbb{P}$, the conditional and marginal probability measures derived from $\mathbb{P}$ are dominated by the analogous conditional and marginal measures derived from μ. For ease of exposition, we slightly abuse notation and let p denote the Radon-Nikodym derivative with respect to μ of $\mathbb{P}$ and the conditional or marginal measures derived from $\mathbb{P}$. It should be clear which derivative is implied from the arguments of p. Doing so expresses (1.13) in the more familiar notation

$$\begin{aligned} p(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n) &= \frac{p(\mathbf{y}_1, \ldots, \mathbf{y}_n, \boldsymbol{\beta}, \boldsymbol{\xi})}{p(\mathbf{y}_1, \ldots, \mathbf{y}_n)} \\ &= \frac{p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}, \boldsymbol{\xi})\, p(\boldsymbol{\beta}, \boldsymbol{\xi})}{\int_{\mathcal{B}\times\Xi} p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}, \boldsymbol{\xi})\, dp(\boldsymbol{\beta}, \boldsymbol{\xi})}. \end{aligned} \tag{1.14}$$

Conventionally, $p(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n)$, $p(\boldsymbol{\theta}, \boldsymbol{\xi})$, $p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}, \boldsymbol{\xi})$ and $p(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ are termed the *posterior probability*, the *prior probability*, the *likelihood* of the data, and the *marginal likelihood* of the data, respectively. We will adopt this convention, even though $p(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n)$ and $p(\boldsymbol{\theta}, \boldsymbol{\xi})$ need not be probabilities (but could rather probability densities).

In practice, the integral in the denominator of (1.14) may not have an analytic solution. Although it could be approximated with a Laplace approximation analogous to (1.12), note it is a *normalizing constant* since it is independent of $\boldsymbol{\beta}, \boldsymbol{\xi}$. The implication is that

$$p(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n) \propto p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}, \boldsymbol{\xi})\, p(\boldsymbol{\beta}, \boldsymbol{\xi}). \tag{1.15}$$

Relation (1.15) facilitates Markov Chain Monte Carlo (MCMC) methods. Monte Carlo approximations are computational methods that use repeated random sampling to approximate deterministic results. A Markov chain is a stochastic process on a (usually countable) state space, where the transition distribution depends only on the current state. An important property of a Markov chain is that it converges to a unique stationary distribution if the chain is irreducible (any state can be reached from any other state in finitely many steps with positive probability) and aperiodic (the greatest common divisor of all lengths of paths with positive probability from a state to itself is one). Markov chains are interesting objects with fascinating properties, but the listed aspect is the most relevant to this thesis. For a general introduction, see Norris (1998).

MCMC methods involve a sampling scheme to dictate the transition probabilities from any state to the next, which uniquely determines the Markov chain. Such a sampling scheme is the Metropolis-Hastings algorithm. It ensures that the chain forms a sample from the distribution of interest once stationarity is reached. The basic idea is to start at any point in the state space, and at each step, sample a point from a proposal probability distribution (usually conditional on the current state) and accept it with a certain acceptance probability. The acceptance probability must be chosen so that the stationary distribution equals the distribution of interest. At the same time, the resulting Markov chain must be irreducible and aperiodic in order to obtain a unique stationary distribution. The resulting transition probability from the current state to any other state is the proposal probability multiplied with the acceptance probability of the corresponding state.

Specifically, denote the state space by $\mathcal{S}$, the current state by x, the sampled candidate next state by x^*, the transition probability function from x to x^* by $t(x^* \mid x)$, the proposal probability function at x^* by $q(x^* \mid x)$, the corresponding acceptance probability function by $a(x^* \mid x)$, and the probability function of the desired distribution at x by $p(x)$. Start at any $x \in \mathcal{S}$, and iteratively proceed as follows.

1. Sample x^* from the proposal distribution conditional on x.
2. There exists a stationary distribution when the *equation of detailed balance*

$$t(x^* \mid x)\, p(x) = t(x \mid x^*)\, p(x^*) \tag{1.16}$$

is satisfied. The *Metropolis acceptance ratio*

$$a(x^* \mid x) = \min\left\{1, \frac{q(x \mid x^*)}{q(x^* \mid x)} \frac{p(x^*)}{p(x)}\right\}$$

is a common choice that satisfies (1.16).

3. Accept x^* as the next state with probability $a(x^* \mid x)$. Otherwise, the next state will be x.
4. Repeat steps 1–3 until convergence to the stationary distribution is reached.

Applying the above to approximate the posterior parameter distribution given by (1.15) proceeds as follows.

a. Start at any $(\boldsymbol{\beta}, \boldsymbol{\xi}) \in \mathcal{B} \times \Xi$, for instance, $(\boldsymbol{\beta}, \boldsymbol{\xi}) = \mathbf{0}$.

b. Sample $(\boldsymbol{\beta}^*, \boldsymbol{\xi}^*)$ from an adequate proposal distribution. The fixed effects $\boldsymbol{\beta}^*$ could be drawn from a $\text{MN}(\boldsymbol{\beta}, \boldsymbol{\Sigma})$ distribution for some covariance matrix $\boldsymbol{\Sigma}$. Assuming that $\boldsymbol{\xi}$ is a vector of covariance matrices, each covariance matrix $\boldsymbol{\xi}_j$ may be drawn from a $\text{W}(\tau^{-1}\boldsymbol{\xi}_j, \tau)$ distribution for some $\tau > 0$.

c. The next state becomes $(\boldsymbol{\beta}^*, \boldsymbol{\xi}^*)$ with acceptance probability

$$\begin{aligned} a\left(\boldsymbol{\beta}^*, \boldsymbol{\xi}^* \mid \boldsymbol{\beta}, \boldsymbol{\xi}\right) &= \min\left\{1, \frac{q\left(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \boldsymbol{\beta}^*, \boldsymbol{\xi}^*\right)}{q\left(\boldsymbol{\beta}^*, \boldsymbol{\xi}^* \mid \boldsymbol{\beta}, \boldsymbol{\xi}\right)} \frac{p(\boldsymbol{\beta}^*, \boldsymbol{\xi}^* \mid \mathbf{y}_1, \ldots, \mathbf{y}_n)}{p(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \mathbf{y}_1, \ldots, \mathbf{y}_n)}\right\} \\ &= \min\left\{1, \frac{q\left(\boldsymbol{\beta}, \boldsymbol{\xi} \mid \boldsymbol{\beta}^*, \boldsymbol{\xi}^*\right)}{q\left(\boldsymbol{\beta}^*, \boldsymbol{\xi}^* \mid \boldsymbol{\beta}, \boldsymbol{\xi}\right)} \frac{p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}^*, \boldsymbol{\xi}^*)\, p(\boldsymbol{\beta}^*, \boldsymbol{\xi}^*)}{p(\mathbf{y}_1, \ldots, \mathbf{y}_n \mid \boldsymbol{\beta}, \boldsymbol{\xi})\, p(\boldsymbol{\beta}, \boldsymbol{\xi})}\right\}, \end{aligned}$$

and otherwise remains at $(\boldsymbol{\beta}, \boldsymbol{\xi})$.

d. Repeat steps b and c until the stationary distribution is reached.

We avoid evaluating the integral in the marginal likelihood $p(\mathbf{y}_1, \ldots, \mathbf{y}_n)$ by approximating the posterior parameter distribution using a Metropolis-Hastings MCMC simulation, as the common factor cancels in the fraction appearing in the acceptance probability.

1.3 Model

1.3.1 A bivariate zero-inflated geometric distribution

Burger et al. (2020) define a robust Bayesian GLMM using a discrete Weibull distribution to model the response. This distribution facilitates robustness under extreme outliers as it can be parameterized in terms of the median, which can consequently be modeled directly. To mimic this approach to define an analogous bivariate response model, we may either derive a bivariate discrete Weibull distribution and parameterize it in terms of its component's medians or find another bivariate count distribution that allows such a parameterization.

The *geometric distribution* Geom(p) describes the distribution of the number of failures when performing independent experiments with success probability p until a success is observed (there is an equivalent distribution on the total number of trials, but the support of this distribution inconveniently excludes 0). Its CDF F is given by

$$F(x) = 1 - (1-p)^{x+1} \quad \text{for} \quad x = 0, 1, 2, \ldots$$

Moreover, its mean is $\mu := \frac{1}{p} - 1$, so we may also parameterize the distribution in terms of μ (denoted Geom(μ)). It is easy to show that its median $[M]$ is defined as

$$[M] = \left\lceil \frac{-1}{\log_2(1-p)} \right\rceil - 1 = \left\lceil \frac{1}{\log_2\left(\frac{\mu+1}{\mu}\right)} \right\rceil - 1.$$

It follows we may also parameterize Geom(p) in terms of its median if we relax the restriction to non-negative integers. The resulting value technically does not equal the median, but it should be close enough to draw meaningful interpretations in a regression framework. As such, we will refer to the median as the solution to $F(M) = \frac{1}{2}$ in $\mathbb{R}$, i.e.,

$$M = \frac{1}{\log_2\left(\frac{\mu+1}{\mu}\right)} - 1.$$

In the same way that distributions are uniquely determined by their moment generating function, distributions on the non-negative integers are also uniquely determined by their *Probability-Generating Function* (PGF). For a random variable X following such a distribution, it is defined as $\pi(s) := \mathbb{E}\{s^X\}$. The PGF of the Geom$(\mu)$ is given by

$$\pi(s) = \frac{1}{1 + \mu(1-s)} \quad \text{for} \quad |s| < \frac{\mu+1}{\mu}.$$

A natural extension of $\pi(s)$ (Jayakumar and Mundassery, 2007) defines a bivariate geometric distribution BGeom(μ, ν, θ) by its PGF

$$\pi(s,t) = \frac{1}{(1+\mu(1-s))(1+\nu(1-t)) - \theta\mu\nu(1-s)(1-t)} \quad \text{for} \quad |s| < \frac{\mu+1}{\mu}, \quad |t| < \frac{\nu+1}{\nu}, \tag{1.17}$$

where the PGF of a bivariate random variable (X, Y) (with support $\{0, 1, \ldots\}^2$) is defined by $\pi(s,t) = \mathbb{E}\{s^X t^Y\}$. It is a valid PGF, since $\pi(1,1)$ correctly evaluates to $\mathbb{E}\{1\} = 1$. The parameters $\mu, \nu > 0$ correspond to the marginal means of the components of the BGeom(μ, ν, θ) distribution, and $\theta \in [-1, 1]$ is such that the covariance of the components equals $\theta\mu\nu$. To see this, note that for $(X, Y) \sim$ BGeom(μ, ν, θ),

$$\mathrm{Cov}\{X, Y\} = \mathbb{E}\{XY\} - \mathbb{E}\{X\}\ \mathbb{E}\{Y\} = \frac{\partial^2 \pi}{\partial s \partial t}(1,1) - \mu\nu = (1+\theta)\mu\nu - \mu\nu = \theta\mu\nu.$$

Remark that the property $\mathbb{E}\{XY\} = \frac{\partial^2 \pi}{\partial s \partial t}(1,1)$ follows from Tonelli's theorem. Note that the marginal distributions are Geom(μ) and Geom(ν), respectively, since

$$\mathbb{E}\{s^X\} = \pi(s,1) = \frac{1}{1+\mu(1-s)} \quad \text{and} \quad \mathbb{E}\{t^Y\} = \pi(1,t) = \frac{1}{1+\nu(1-t)}.$$

For large μ, ν, the parameter θ resembles the correlation coefficient ρ of the components, since

$$\rho = \frac{\operatorname{Cov}\{X,Y\}}{\sqrt{\operatorname{Var} X}\sqrt{\operatorname{Var} Y}} = \frac{\theta\mu\nu}{\sqrt{\mu(\mu+1)}\sqrt{\nu(\nu+1)}} = \theta\sqrt{\frac{\mu\nu}{(\mu+1)(\nu+1)}} \xrightarrow{\mu,\nu\to\infty} \theta.$$

If more dimensions are desired, a PGF of the form

$$\pi(s_1,\ldots,s_n) = \frac{1}{\prod_{i=1}^{n}(1+\mu_i(1-s_i)) - \sum_{i<j}\theta_{ij}\mu_i\mu_j(1-s_i)(1-s_j)} \quad \text{for} \quad |s_i| < \frac{\mu_i+1}{\mu_i}, \tag{1.18}$$

where each $\mu_i > 0$ and $\theta_i \in [-1,1]$, may be used as a starting point instead.

Proposition 1. *The PMF of the BGeom(μ,ν,θ) distribution is given by*

$$\mathbb{P}\{(X,Y)=(x,y)\} = \sum_{j=0}^{\min\{x,y\}} (-1)^j \binom{x+y-j}{j,x-j,y-j} ((1-\theta)\mu\nu)^j (\mu+(1-\theta)\mu\nu)^{x-j} \times (\nu+(1-\theta)\mu\nu)^{y-j}(1+\mu+\nu+(1-\theta)\mu\nu)^{-(x+y-j+1)}.$$

Proof. The PMF can be derived using (1.17) and the relation (cf. Tonelli's theorem)

$$\mathbb{P}\{(X,Y)=(x,y)\} = \frac{1}{x!y!}\frac{\partial^{x+y}\pi}{\partial s^x \partial t^y}(0,0).$$

Let $f(s,t) := 1/\pi(s,t)$ for each s,t, and denote

$$\begin{aligned} f_s(s,t) &:= \frac{\partial f}{\partial s}(s,t) = -\mu - (1-\theta)\mu\nu(1-t), \\ f_t(s,t) &:= \frac{\partial f}{\partial t}(s,t) = -\nu - (1-\theta)\mu\nu(1-s), \\ f_{st}(s,t) &:= \frac{\partial^2 f}{\partial s \partial t}(s,t) = (1-\theta)\mu\nu. \end{aligned}$$

Notice that higher-order derivatives with respect to s or t are zero. We claim that

$$\frac{\partial^x \pi}{\partial s^x}(s,t) = (-1)^x\, x! \frac{f_s^x}{f^{x+1}}(s,t) \tag{1.19}$$

for all integers $x \geqslant 0$. The base case where $x = 0$ holds superfluously, so suppose (1.19) holds for an arbitrary integer $x \geqslant 0$. Then, suppressing the arguments s,t for clarity,

$$\frac{\partial^{x+1}\pi}{\partial s^{x+1}} = \frac{\partial}{\partial s}\left((-1)^x\, x! \frac{f_s^x}{f^{x+1}}\right) = (-1)^x\, x! \frac{-(x+1)f_s^x\, f_s\, f^x}{f^{2(x+1)}} = (-1)^{x+1}\,(x+1)! \frac{f_s^{x+1}}{f^{x+2}}.$$

The claim (1.19) now follows by mathematical induction.

Next, we claim that for any integer $x \geqslant 0$,

$$\frac{\partial^{x+y}\pi}{\partial s^x \partial t^y} = x!y! \sum_{j=0}^{\min\{x,y\}} (-1)^{x+y-j} \binom{x+y-j}{j,x-j,y-j} \frac{f_{st}^j\, f_s^{x-j}\, f_t^{y-j}}{f^{x+y-j+1}} \tag{1.20}$$

for all integers $y \geqslant 0$. The base case where $y = 0$ reduces to (1.19), so suppose that (1.20) holds for an arbitrary integer $y \geqslant 0$. Then

$$\begin{aligned}\frac{\partial^{x+(y+1)}\pi}{\partial s^x \partial t^{y+1}} &= \frac{\partial}{\partial t}\left(x!y! \sum_{j=0}^{\min\{x,y\}} (-1)^{x+y-j} \binom{x+y-j}{j, x-j, y-j} \frac{f_{st}^j \, f_s^{x-j} \, f_t^{y-j}}{f^{x+y-j+1}} \right) \\ &= x!y! \sum_{j=0}^{\min\{x,y\}} (-1)^{x+y-j} \binom{x+y-j}{j, x-j, y-j} \frac{\partial}{\partial t}\left(\frac{f_{st}^j \, f_s^{x-j} \, f_t^{y-j}}{f^{x+y-j+1}} \right),\end{aligned}$$

where, if $j < x$,

$$\begin{aligned}&\frac{\partial}{\partial t}\left(\frac{f_{st}^j \, f_s^{x-j} \, f_t^{y-j}}{f^{x+y-j+1}} \right) \\ &\quad= \frac{(x-j) f_{st}^{j+1} f_s^{x-j-1} f_t^{y-j} f^{x+y-j+1} - (x+y-j+1) f_{st}^j \, f_s^{x-j} \, f_t^{y-j+1} f^{x+y-j}}{f^{2(x+y-j+1)}} \\ &\quad= (x-(j+1)+1) \frac{f_{st}^{j+1} f_s^{x-(j+1)} f_t^{(y+1)-(j+1)}}{f^{x+(y+1)-(j+1)+1}} - (x+(y+1)-j) \frac{f_{st}^j \, f_s^{x-j} \, f_t^{(y+1)-j}}{f^{x+(y+1)-j+1}}.\end{aligned}$$

and if $j = x$,

$$\frac{\partial}{\partial t}\left(\frac{f_{st}^j \, f_s^{x-j} \, f_t^{y-j}}{f^{x+y-j+1}} \right) = -(x+(y+1)-j) \frac{f_{st}^j \, f_s^{x-j} \, f_t^{(y+1)-j}}{f^{x+(y+1)-j+1}}.$$

Hence, $\min\{x, y\} = x$ implies that taking the partial derivative in the final summand where $j = x$ does not generate an additional summand in terms of $j + 1 = x + 1$. Moreover, it follows $\min\{x, y+1\} = x$ as well. On the other hand, if no summand has $j = x$, then $y = \min\{x, y\} < x$. It follows $y + 1 \leqslant x$, and so $\min\{x, y + 1\} = y + 1$. Taking partial derivatives in the summand with $j = y$ generates a quantity in terms of $j + 1 = y + 1$, given by

$$\begin{aligned}&(-1)^{x+y-j} \binom{x+y-j}{j, x-j, y-j} (x-(j+1)+1) \frac{f_{st}^{j+1} f_s^{x-(j+1)} f_t^{(y+1)-(j+1)}}{f^{x+(y+1)-(j+1)+1}} \\ &\quad= (-1)^{x+(y+1)-(j+1)} \underbrace{(j+1)}_{y+1} \binom{x+(y+1)-(j+1)}{j+1, x-(j+1), (y+1)-(j+1)} \\ &\qquad\qquad \times \frac{f_{st}^{j+1} f_s^{x-(j+1)} f_t^{(y+1)-(j+1)}}{f^{x+(y+1)-(j+1)+1}}.\end{aligned}$$

So, if $\min\{x, y\} = x$, we obtain

$$\begin{aligned}\frac{\partial^{x+y+1}\pi}{\partial s^x \partial t^{y+1}} &= x!y! \sum_{j=0}^{\min\{x,y+1\}} -(-1)^{x+y-j} (x+(y+1)-j) \binom{x+y-j}{j, x-j, y-j} \frac{f_{st}^j \, f_s^{x-j} \, f_t^{(y+1)-j}}{f^{x+(y+1)-j+1}} \\ &\quad + (-1)^{x+y-(j-1)} (x-j+1) \binom{x+y-(j-1)}{j-1, x-(j-1), y-(j-1)} \frac{f_{st}^j \, f_s^{x-j} \, f_t^{(y+1)-j}}{f^{x+(y+1)-j+1}} \\ &= x!y! \sum_{j=0}^{\min\{x,y+1\}} (-1)^{x+(y+1)-j} \left((x+(y+1)-j) \binom{x+y-j}{j, x-j, y-j} \right.\end{aligned}$$

$$+(x-j+1)\binom{x+y-(j-1)}{j-1,x-(j-1),y-(j-1)}\Bigg)\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}$$

$$= x!y!\sum_{j=0}^{\min\{x,y+1\}}(-1)^{x+(y+1)-j}\Bigg((y+1-j)\binom{x+(y+1)-j}{j,x-j,(y+1)-j}$$
$$+j\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\Bigg)\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}$$

$$= x!y!\sum_{j=0}^{\min\{x,y+1\}}(-1)^{x+(y+1)-j}(y+1)\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}$$

$$= x!(y+1)!\sum_{j=0}^{\min\{x,y+1\}}(-1)^{x+(y+1)-j}\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}$$

by collecting terms. If $\min\{x,y\}<x$, doing so leads to the same results, i.e.,

$$\frac{\partial^{x+y+1}\pi}{\partial s^{x}\partial t^{y+1}} = x!(y+1)!\sum_{j=0}^{\min\{x,y\}}(-1)^{x+(y+1)-j}\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}$$
$$+\left[x!y!(-1)^{x+(y+1)-j}(y+1)\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\frac{f_{st}^{j}f_{s}^{x-j}f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}\right]_{j=y+1}$$

$$= x!(y+1)!\sum_{j=0}^{\min\{x,y+1\}}(-1)^{x+(y+1)-j}\binom{x+(y+1)-j}{j,x-j,(y+1)-j}\frac{f_{st}^{j}\,f_{s}^{x-j}\,f_{t}^{(y+1)-j}}{f^{x+(y+1)-j+1}}.$$

Again, the claim follows by mathematical induction. Substituting

$$f_{st}(0,0)=(1-\theta)\mu\nu, \qquad f_{t}(0,0)=-(\nu+(1-\theta)\mu\nu),$$
$$f_{s}(0,0)=-(\mu+(1-\theta)\mu\nu), \qquad f(0,0)=1+\mu+\nu+(1-\theta)\mu\nu,$$

into (1.20), consequently, yields

$$\frac{1}{x!y!}\frac{\partial^{x+y}\pi}{\partial s^{x}\partial t^{y}}(0,0) = \sum_{j=0}^{\min\{x,y\}}(-1)^{x+y-j}\binom{x+y-j}{j,x-j,y-j}\frac{f_{st}^{j}(0,0)\,f_{s}^{x-j}(0,0)\,f_{t}^{y-j}(0,0)}{f^{x+y-j+1}(0,0)}$$

$$= \sum_{j=0}^{\min\{x,y\}}(-1)^{2x+2y-3j}\binom{x+y-j}{j,x-j,y-j}((1-\theta)\mu\nu)^{j}(\mu+(1-\theta)\mu\nu)^{x-j}$$
$$\times(\nu+(1-\theta)\mu\nu)^{y-j}(1+\mu+\nu+(1-\theta)\mu\nu)^{-(x+y-j+1)}.$$

Noting that $(-1)^{2x+2y-3j}=(-1)^{j}$ for any j completes the proof. □

It is not uncommon for data to be *zero-inflated* with respect to a certain distribution; this happens when zeros occur significantly more frequently than prescribed by the distribution. To be able to model such data appropriately, we define a zero-inflated variant of our distribution of interest. For univariate distributions, this simply entails adding a point mass at zero and scaling the probability mass on the original support (which may also include zero). For multivariate distributions, we can impose zero inflation on any subspace of the support. For a bivariate distribution, the most general definition is the following. Suppose a $\mathscr{P}$-distributed random variable (X, Y) has respective marginal distributions $\mathscr{P}_X$ and $\mathscr{P}_Y$. Then a zero-inflated variant $\mathbf{Z}$ is distributed as

$$\mathbf{Z} \sim \begin{cases} (0,0) & \text{with probability } p_{00}, \\ (\mathscr{P}_X, 0) & \text{with probability } p_{10}, \\ (0, \mathscr{P}_Y) & \text{with probability } p_{01}, \\ \mathscr{P} & \text{with probability } p_{11} = 1 - p_{00} - p_{10} - p_{01}. \end{cases}$$

Hence, we will define a Zero-Inflated Bivariate Geometric distribution $\text{ZIBG}(\mu, \nu, \theta, p, q, r)$ by

$$\text{ZIBG}(\mu, \nu, \theta, p, q, r) \sim \begin{cases} (0,0) & \text{with probability } p, \\ (\text{Geom}(\mu), 0) & \text{with probability } q, \\ (0, \text{Geom}(\nu)) & \text{with probability } r, \\ \text{BGeom}(\mu, \nu, \theta) & \text{with probability } 1 - p - q - r. \end{cases} \tag{1.21}$$

Control over the zero-inflation is particularly important for modeling with the geometric distribution (multivariate or not) since there is no dispersion parameter. Let $f(x, y)$ be the PMF of the $\text{ZIBG}(\mu, \nu, \theta, p, q, r)$ distribution, and denote the PMF of the $\text{BGeom}(\mu, \nu, \theta)$ distribution by $g(x, y)$ (for non-negative integers x, y). Then

$$\begin{aligned} f(x,y) = p\, \mathbb{1}\{x = y = 0\} + q\, \mathbb{1}\{y = 0\} \frac{\mu^x}{(\mu+1)^{x+1}} + r\, \mathbb{1}\{x = 0\} \frac{\nu^y}{(\nu+1)^{y+1}} \\ + (1 - p - q - r)\, g(x, y). \end{aligned}$$

1.3.2 Marginal median reparameterization

Since the marginal means μ, ν of the $\text{BGeom}(\mu, \nu, \theta)$ distribution can be expressed in terms of the respective marginal medians M, N (recall we do not require them to be integers), we can completely parameterize the distribution in terms of M, N, and θ. Consequently, the same holds for the $\text{ZIBG}(\mu, \nu, \theta, p, q, r)$ distribution.

We start by deriving the mean μ (before zero-inflation) of a zero-inflated geometric distribution with zero-inflation parameter p in terms of its median M (taking zero-inflation into account). It has CDF

$$F(x) = p + (1 - p) \left(1 - \left(1 - \frac{1}{\mu + 1} \right)^{x+1} \right) \quad \text{for} \quad x = 0, 1, 2, \ldots,$$

and its median M solves $\frac{1}{2} = F(M)$. It is important to keep in mind that necessarily $M = 0$ when $p \geqslant \frac{1}{2}$, so assume the contrary. Applying some straightforward transformations yields

$$M = \frac{\log(2(1-p))}{\log\left(\frac{\mu+1}{\mu}\right)} - 1 \quad \text{or} \quad \mu = \left((2(1-p))^{\frac{1}{M+1}} - 1 \right)^{-1}. \tag{1.22}$$

The marginal zero-inflations for the first and second component of the ZIBG$(\mu, \nu, \theta, p, q, r)$ distribution are $p + r$ and $p + q$ (cf. (1.21)), respectively. By (1.22), we may parameterize the distribution in terms of its marginal medians M, N by letting

$$\mu = \left((2(1-p-r))^{\frac{1}{M+1}} - 1\right)^{-1} \quad \text{and} \quad \nu = \left((2(1-p-q))^{\frac{1}{N+1}} - 1\right)^{-1}. \tag{1.23}$$

We will show this reparameterization facilitates robustness with respect to large outliers in a marginal sense by analyzing the maximum likelihood estimates of a univariate zero-inflated geometric distribution. Suppose we observed $x_1, \ldots, x_n$ for $X_1, \ldots, X_n \overset{\text{iid}}{\sim} \text{ZIG}(p, \mu)$. Without loss of generality, let

$$x_1, \ldots, x_m > 0 \quad \text{and} \quad x_{m+1}, \ldots, x_n = 0.$$

Denoting $y := \sum_{i=1}^{n} x_i$ and $k := n - m$, the joint probability mass of $x_1, \ldots, x_n$ becomes

$$f(x_1, \ldots, x_n) = (1-p)^m \frac{\mu^y}{(\mu+1)^{y+m}} \left(p + \frac{1-p}{\mu+1}\right)^k.$$

As such, the log likelihood of p, μ is given by

$$l(p, \mu) = m \log(1-p) + y \log \mu - (y+m) \log(\mu+1) + k \left(\log(p\mu+1) - \log(\mu+1)\right).$$

Setting its first-order partial derivatives to zero yields

$$\begin{aligned} \frac{\partial l}{\partial p} &= \frac{-m}{1-p} + \frac{k\mu}{p\mu+1} \\ &\propto k\mu(1-p) - m(p\mu+1) \\ &= k\mu - m - np\mu = 0 \end{aligned} \tag{1.24}$$

and

$$\begin{aligned} \frac{\partial l}{\partial \mu} &= \frac{y}{\mu} - \frac{y+m}{\mu+1} + \frac{kp}{p\mu+1} - \frac{k}{\mu+1} \\ &\propto y(\mu+1)(p\mu+1) - (y+n)\mu(p\mu+1) + kp\mu(\mu+1) - k\mu(p\mu+1) \\ &= -mp\mu^2 + ((k+y)p - n)\mu + y = 0. \end{aligned} \tag{1.25}$$

The respective solutions to (1.24) and (1.25) expressed in μ and p are

$$p = \frac{k\mu - m}{n\mu} \quad \text{and} \quad \mu = \frac{\sqrt{((k+y)p-n)^2 + 4mpy} + (k+y)p - n}{2mp}.$$

Substituting the expression for p into the expression for μ and solving for μ gives us

$$\mu = y/m - 1 \quad \text{and} \quad p = \frac{k(y/m-1) - m}{n(y/m-1)}.$$

We may verify that the maximum likelihood estimator for (p, μ), therefore, is

$$(\hat{p}, \hat{\mu}) = \left(\frac{\alpha\bar{X} - 1}{\bar{X} - 1}, \bar{X} - 1\right),$$

where $\bar{X} := Y/m$ is the mean of the nonzero observations and $\alpha := k/n$ is the fraction of zeros in the data. Assuming $\bar{X} > 1$ and $\alpha < (\bar{X}+1)/(2\bar{X})$, it follows the maximum likelihood estimator for M is

$$\hat{M} = \frac{\log(2(1-\hat{p}))}{\log\left(\frac{\hat{\mu}+1}{\hat{\mu}}\right)} - 1 = \frac{\log\left(2\left(\frac{(1-\alpha)\bar{X}}{(\bar{X}-1)}\right)\right)}{\log\left(\frac{\bar{X}}{\bar{X}-1}\right)} - 1 = \frac{\log\left(2(1-\alpha)\right)}{\log\left(\frac{\bar{X}}{\bar{X}-1}\right)}.$$

Let us compare the sensitivity of maximum likelihood estimators for the mean and the median (taking zero-inflation into account) with respect to $\bar{X}$. For the mean, it is given by

$$\tilde{\mu} := (1-p)\mu = (\bar{X}-1)\frac{(1-\alpha)\bar{X}}{(\bar{X}-1)} = (1-\alpha)\bar{X}.$$

Its derivative with respect to $\bar{X}$ is, therefore, $\frac{d\tilde{\mu}}{d\bar{X}} = 1-\alpha$. For the median, we have

$$\frac{d\hat{M}}{d\bar{X}} = \frac{\log\left(2\frac{m}{n}\right)}{\bar{X}(\bar{X}-1)\log^2\left(\frac{\bar{X}}{\bar{X}-1}\right)} = \frac{\log(2(1-\alpha))}{\bar{X}(\bar{X}-1)\log^2\left(\frac{\bar{X}}{\bar{X}-1}\right)} \sim \log(2(1-\alpha))$$

for large enough $\bar{X}$. Note that the equivalence does not hold for arbitrarily large $\bar{X}$, but it suffices for practical purposes (cf. the left panel of Figure 1.1). The 'asymptotic' ratio $\frac{d\tilde{\mu}}{d\bar{X}} / \frac{d\hat{M}}{d\bar{X}}$ rapidly increases as α tends to $\frac{1}{2}$ (cf. the right panel of Figure 1.1), showing robustness with respect to large outliers that increases with zero-inflation.

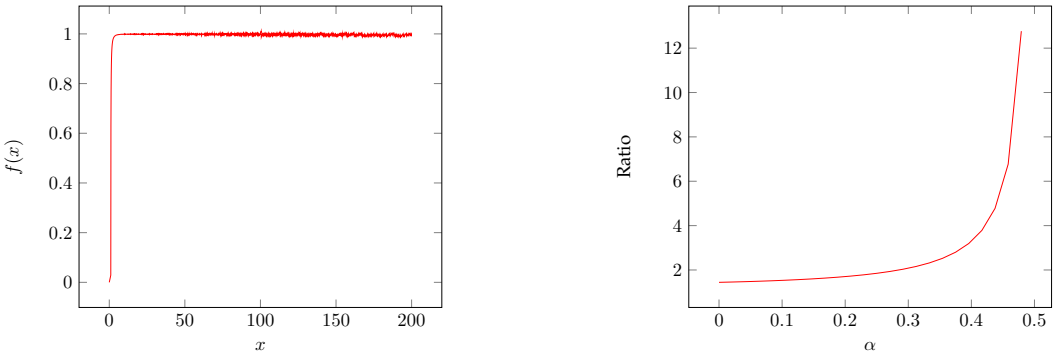


Figure 1.1: The function $f : x \mapsto x(x-1)\log^2\left(\frac{x}{x-1}\right)$ (left) and the 'asymptotic' ratio $\frac{d\tilde{\mu}}{d\bar{X}} / \frac{d\hat{M}}{d\bar{X}}$ with respect to α (right).

1.3.3 Model specification

Let $\mathcal{D} = \{(\mathbf{y}_i, \mathbf{x}_i, z_i)\}_{i=1}^n$ be a data set of $n \in \mathbb{N}$ observations, where each $\mathbf{y}_i$ is an attained value of a 2-dimensional random variable $\mathbf{Y}_i$, each $\mathbf{x}_i$ is a k-dimensional vector of covariate levels, and each z_i is a grouping factor taking values in $\{1, \ldots, d\}$. Consider the GLMM of the form (1.6), given by

$$
\begin{aligned}
\mathbf{Y}_i &\sim \text{ZIBG}(M_i, N_i, \theta_i, p, q, r), \\
\log M_i &= \mathbf{x}_i \cdot \boldsymbol{\beta}_M + \mathbf{w}_i \cdot (\mathbf{B}_M)_{z_i}, \quad \log N_i = \mathbf{x}_i \cdot \boldsymbol{\beta}_N + \mathbf{w}_i \cdot (\mathbf{B}_N)_{z_i}, \\
\theta_i &= 2\,\text{logit}^{-1}(\mathbf{x}_i \cdot \boldsymbol{\beta}_\theta) - 1, \\
p &= \frac{e^{\beta_p}}{2C}, \quad q = \frac{e^{\beta_q}}{2C}, \quad r = \frac{e^{\beta_r}}{2C}, \quad \text{where} \quad C = 1 + e^{\beta_p} + e^{\beta_q} + e^{\beta_r}, \\
(\mathbf{B}_M)_1, \ldots, (\mathbf{B}_M)_d &\overset{\text{iid}}{\sim} \text{MN}(\mathbf{0}, \boldsymbol{\Sigma}_M), \\
(\mathbf{B}_N)_1, \ldots, (\mathbf{B}_N)_d &\overset{\text{iid}}{\sim} \text{MN}(\mathbf{0}, \boldsymbol{\Sigma}_N), \quad 0 < \boldsymbol{\Sigma}_M, \boldsymbol{\Sigma}_N \in \mathbb{R}^{m \times m}
\end{aligned}
\tag{1.26}
$$

for $i = 1, \ldots, n$, where $m \leqslant k$ is the respective numbers of random effects for the M_i, N_i. As in (1.6), $\boldsymbol{\beta}_N, \boldsymbol{\beta}_N, \boldsymbol{\beta}_\theta, \beta_p, \beta_r, \beta_q$ are (vectors of) fixed effects and $\mathbf{B}_M, \mathbf{B}_N$ are random effect matrices; see Section 1.2.2 for a more detailed description of their structures. The equations for p, q, r ensure that $p + q, p + r < p + q + r < \frac{1}{2}$, so that (1.23) is a valid reparameterization.

Note (1.26) is certainly not the only way to specify the distribution parameters in terms of the covariates. For instance, one might choose to add random effects to the model for the θ_i or to have non-constant zero-inflation parameters. The configuration in (1.26) is chosen to provide a large degree of freedom when modeling the marginal medians. Moreover, the non-constant correlation parameters are motivated by the empirical data so that we may analyze the population measures' similarity given external factors. We reduce model complexity by holding zero-inflation parameters constant and restricting random effects to the marginal median models.

We use a Bayesian specification similar to the one presented in Section 3.3 of Burger et al. (2020). This means we will attribute uncertainty to the fixed effects by specifying a $\text{N}(0, \sigma^2 = 10)$ prior distribution for each component. The inverse covariance matrices $(\boldsymbol{\Sigma}^{(M)})^{-1}, (\boldsymbol{\Sigma}^{(N)})^{-1}$ are assigned an MGH-$t(A = 10, 2, m)$ (*Multivariate Generalized Hyperbolic t*) prior distribution. The MGH-$t(A, v, d)$ distribution is defined as a $\text{W}((2v\boldsymbol{\Omega})^{-1}, v + d - 1)$ distribution, where

$$
\boldsymbol{\Omega} = \text{diag}(\omega_1, \ldots, \omega_d), \quad \text{where each} \quad \omega_i \overset{\text{iid}}{\sim} \Gamma(0.5, 1/A^2).
$$

As Burger et al. (2020) argue, close-to-zero variability in random effects over time leads to inferences on the corresponding variances that are sensitive to overly vague priors. The resut may be an upward bias in the estimated variance components, causing excessively large confidence intervals. Using an MGH-t prior distribution for the inverse random effect covariance matrices should reduce this effect.

Let $f(\mathbf{y}_i \mid M_i, N_i, \theta_i, p, q, r)$ denote the PMF of $\mathbf{Y}_i$ at $\mathbf{y}_i$ for $i = 1, \ldots, n$. The resulting joint posterior probability of the parameters is

$$
\begin{aligned}
&p(\boldsymbol{\beta}_M, \boldsymbol{\beta}_N, \boldsymbol{\beta}_\theta, \beta_p, \beta_q, \beta_r, \mathbf{B}_N, \mathbf{B}_M, \boldsymbol{\Sigma}_N, \boldsymbol{\Sigma}_M \mid \mathbf{y}_1, \ldots, \mathbf{y}_n) \\
&\quad \propto \left(\prod_{i=1}^{n} f(\mathbf{y}_i \mid M_i, N_i, \theta_i, p, q, r) \right) p(\boldsymbol{\beta}_N)\, p(\boldsymbol{\beta}_M)\, p(\boldsymbol{\beta}_\theta)\, p(\beta_p)\, p(\beta_r)\, p(\beta_q) \\
&\qquad\qquad \times\; p(\mathbf{B}_M \mid \boldsymbol{\Sigma}_M)\, p(\boldsymbol{\Sigma}_M^{-1} \mid \boldsymbol{\Omega})\, p(\mathbf{B}_N \mid \boldsymbol{\Sigma}_N)\, p(\boldsymbol{\Sigma}_N^{-1} \mid \boldsymbol{\Omega})\, p(\boldsymbol{\Omega}),
\end{aligned}
$$

where

$$
\begin{aligned}
p(\boldsymbol{\beta}) &\propto \exp\left(-\tfrac{1}{2}\sigma^{-2}\boldsymbol{\beta}\cdot\boldsymbol{\beta}\right) \quad \text{for each} \quad \boldsymbol{\beta} \in \{\boldsymbol{\beta}_M, \boldsymbol{\beta}_N, \boldsymbol{\beta}_\theta, \beta_p, \beta_q, \beta_r\},\\
p(\mathbf{B}_M \mid \boldsymbol{\Sigma}_M) &\propto \prod_{j=1}^{d} |\boldsymbol{\Sigma}_M|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2}(\mathbf{B}_M)_j \cdot (\boldsymbol{\Sigma}_M)^{-1}(\mathbf{B}_M)_j\right),\\
p(\mathbf{B}_N \mid \boldsymbol{\Sigma}_N) &\propto \prod_{j=1}^{d} |\boldsymbol{\Sigma}_N|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2}(\mathbf{B}_N)_j \cdot (\boldsymbol{\Sigma}_N)^{-1}(\mathbf{B}_N)_j\right),\\
p\left(\boldsymbol{\Sigma}_M^{-1} \mid \boldsymbol{\Omega}\right) &\propto \exp\left(-2\,\mathrm{tr}\left(\boldsymbol{\Omega}\boldsymbol{\Sigma}_M^{-1}\right)\right), \quad p\left(\boldsymbol{\Sigma}_N^{-1} \mid \boldsymbol{\Omega}\right) \propto \exp\left(-2\,\mathrm{tr}\left(\boldsymbol{\Omega}\boldsymbol{\Sigma}_N^{-1}\right)\right)\\
p(\boldsymbol{\Omega}) &\propto \prod_{j=1}^{d} \omega_j^{-\frac{1}{2}} \exp\left(-A^{-2}\omega_j\right).
\end{aligned}
$$

1.4 Estimation and simulation study

A simulation study is needed to verify that (1) we are able to retrieve a set of parameters from data generated accordingly, and (2) the proposed model shows better performance in certain situations than the closest published model (Famoye, 2019). The first point also serves as a verification that the implementation of the estimation procedure is free of programming errors, while the second justifies the need for a novel modeling technique.

1.4.1 Parameter retrieval

We consider a simplified version of (1.26) in which we also hold the N_i and θ_i constant across observations. A single random intercept is added to the M_i, so we will restrict our attention to models of the form

$$
\begin{aligned}
\mathbf{Y}_i &\sim \mathrm{ZIBG}(M_i, N, \theta, p, q, r),\\
\log M_i &= \beta_M + b_{z_i}, \quad \log N = \beta_N,\\
\theta &= 2\,\mathrm{logit}^{-1}(\beta_\theta) - 1,\\
p &= \frac{e^{\beta_p}}{2C}, \quad q = \frac{e^{\beta_q}}{2C}, \quad r = \frac{e^{\beta_r}}{2C}, \quad \text{where} \quad C = 1 + e^{\beta_p} + e^{\beta_q} + e^{\beta_r},\\
b_1, \ldots, b_d &\overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2).
\end{aligned}
\tag{1.27}
$$

Since we are not dealing with close-to-zero random effect variability, we may use a simpler prior distribution $\sigma^{-2} \sim \mathrm{W}(1,1) = \Gamma(1/2, 1)$. To make sure that the prior distribution on (p, q, r) is more or less uniformly distributed on the simplex $\{(p,q,r) \geqslant \mathbf{0} \mid p + q + r < \frac{1}{2}\}$, we let $\beta_p, \beta_q, \beta_r \sim \mathrm{N}(0, 4)$. Similarly, letting $\beta_\theta \sim \mathrm{N}(0, 4)$ makes θ approximately uniformly distributed on $[-1, 1]$. As before, the prior distributions for β_M, β_N are $\mathrm{N}(0, 100)$.

We take $d = 6$ treating groups (this number is considered a minimum for random effects, cf. Bolker (2015)) and set the random effect standard deviation to $\sigma = 0.5$. The random effects $b_1, \ldots, b_6$ are taken to be the $\frac{1}{7}, \frac{2}{7}, \ldots, \frac{6}{7}$ quantiles of the $\mathrm{N}(0, \sigma^2)$ distribution. Setting $\beta_M = 1$, $\beta_N = 0.75$, and $\beta_\theta, \beta_p, \beta_q, \beta_r$ such that $\theta = 0.5, p = 0.05, q = 0.1, r = 0.2$, we generate 200 observations for each group to obtain a total of $n = 1200$ data points. This means we can simply let $z_1, \ldots, z_{200} = 1$, $z_{201}, \ldots, z_{400} = 2$, and so on. The observations are generated using the Metropolis-Hastings MCMC algorithm given in Listing B.1. They are displayed in a scatter plot in Figure 1.2

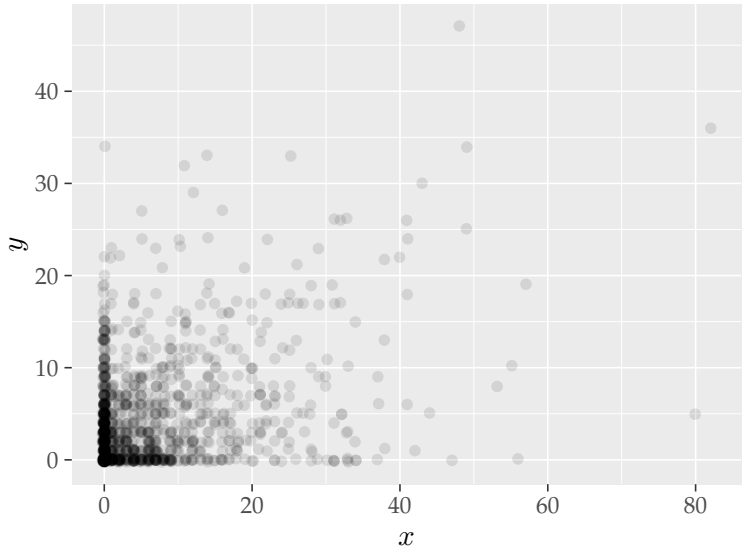


Figure 1.2: Data sampled from the model (1.27) using the MCMC algorithm given in Listing B.1.

To estimate the posterior parameter distributions, we use the Metropolis-Hastings MCMC algorithm provided by *JAGS* (*Just Another Gibbs Sampler*, see Plummer (2017)). JAGS uses a model definition written in the BUGS language (Lunn et al., 2012), which can be found in Listing B.4. Note in the model definition, the stochastic relation with the distribution `dzibg()` requires the custom module `ZIBGeometric`. The source code for this module can be found in Appendix B.3. We invoke JAGS in R using the package `runjags` (Denwood et al., 2016). To be able to assess convergence, we generate eight independent chains in parallel with respective seeds 50 005, 12 572, 24 342, 90 334, 16 634, 38 441, 36 514, 83 179 for the random number generator `base::Mersenne-Twister`. For each chain, all parameters are initialized at zero, the samplers are optimized for 100 iterations, a burn-in period runs for 500 iterations, and the remaining 5000 iterations are thinned by 50 to obtain samples of size 100.

Summary statistics on the parameter posterior distribution estimates are reported in Table 1.1. The sample autocorrelation with a lag of 50 is shown in the second-to-last column. The effective sample size in the third-to-last column is a correction of the sample size, taking into account the sample autocorrelation; it is computed using the `effectiveSize()` function in the `coda` package. The associated Monte Carlo standard error in the seventh column is the standard deviation in the sixth column divided by the square root of the effective sample size. It should be small compared to the standard deviation, which is the case for each parameter. The last column shows the potential scale reduction factor of the Gelman-Rubin statistic (Gelman et al., 1992), which should be close to one (this indicates the between-chains variance is small compared to the within-chain variance). No values are alarmingly high, as all are well below 1.05.

Since all 95% confidence intervals include the original parameter values (which are even quite close to the posterior means and medians), we may safely assume the model is identifiable and the estimation procedure is free of notable programming errors.

	Original	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_M	1.00	0.15	0.83	1.48	0.83	0.343	0.021	276	-0.039	1.008
β_N	0.75	0.67	0.79	0.91	0.79	0.062	0.002	775	-0.102	1.002
β_θ	1.10	0.87	1.07	1.25	1.07	0.099	0.003	1021	-0.037	1.004
β_p	-1.10	-1.38	-0.89	-0.35	-0.89	0.261	0.009	841	-0.066	1.001
β_q	-0.41	-0.78	-0.30	0.23	-0.30	0.268	0.009	911	-0.007	1.004
β_r	0.29	0.23	0.58	1.02	0.59	0.206	0.007	883	0.031	1.003
b_1	-0.53	-1.24	-0.53	0.12	-0.54	0.350	0.017	407	-0.053	1.008
b_2	-0.28	-1.05	-0.33	0.35	-0.33	0.355	0.018	382	-0.039	1.006
b_3	-0.09	-0.94	-0.28	0.44	-0.27	0.350	0.019	348	-0.032	1.006
b_4	0.09	-0.56	0.07	0.76	0.08	0.346	0.017	408	-0.027	1.009
b_5	0.28	-0.13	0.53	1.25	0.54	0.344	0.016	442	-0.041	1.010
b_6	0.53	-0.13	0.51	1.24	0.53	0.344	0.017	433	-0.050	1.008
σ^{-2}	1.00	0.24	1.69	4.17	1.94	1.127	0.041	773	-0.018	1.002

Table 1.1: Summary statistics on the posterior parameter distributions along with the original values.

1.4.2 Comparison with existing models

Famoye (2019) introduces the Bivariate Expontiated-Exponential Geometric (BEEG) regression model to accommodate negative, zero, or positive correlation, zero-inflation, and over- or under-dispersion in two-dimensional count data. The PMF of the BEEG distribution is given by

$$g(x_1, x_2) := \left(1 + \lambda \prod_{t=1}^{2} \left(e^{-x_t} - c_t\right)\right) \prod_{t=1}^{2} \left(\left(1 - \theta_t^{x_t+1}\right)^{b_t} - \left(1 - \theta_t^{x_t}\right)^{b_t}\right) \quad \text{for} \quad x_1, x_2 = 0, 1, 2, \ldots, \tag{1.28}$$

where $\theta_1, \theta_2 \in (0, 1)$, $b_1, b_2 > 0$, and $\lambda \in \mathbb{R}$. Moreover, $c_t = \mathbb{E}\{e^{-X_t}\}$ for $t = 1, 2$ and $(X_1, X_2) \sim$ BEEG$(\theta_1, \theta_2, b_1, b_2, \lambda)$, which means

$$c_t = \sum_{r=0}^{\infty} \frac{b_t(b_t - 1) \cdots (b_t - r + 1)(-1)^r}{r!} \frac{\theta_t^r - 1}{1 - \theta_t^r e^{-1}} \quad \text{for} \quad t = 1, 2. \tag{1.29}$$

The infinite sum in (1.29) is well approximated by its first 10 terms in most cases. Note a bivariate geometric distribution is obtained by setting $b_1 = b_2 = 1$, which is different from the BGeom(μ, ν, θ) distribution proposed in Section 1.3.1. A Zero-Inflated BEEG (ZIBEEG) distribution is defined as

$$\text{ZIBEEG}(\theta_1, \theta_2, b_1, b_2, \lambda, p) \sim \begin{cases} (0, 0) & \text{with probability } p, \\ \text{BEEG}(\theta_1, \theta_2, b_1, b_2, \lambda) & \text{with probability } 1 - p, \end{cases}$$

and has PMF

$$f(x_1, x_2) := p\, \mathbb{1}\{x_1, x_2 = 0\} + (1 - p)\, g(x_1, x_2) \quad \text{for} \quad x_1, x_2 = 0, 1, 2, \ldots$$

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_M	0.68	1.11	1.48	1.09	0.206	0.012	281	-0.062	1.010
β_N	-0.01	0.50	0.99	0.47	0.258	0.015	301	-0.078	1.010
β_θ	0.44	0.82	1.19	0.82	0.192	0.006	1121	-0.051	0.999
β_p	-0.36	0.17	0.75	0.18	0.276	0.017	262	-0.069	1.011

Table 1.2: Summary statistics on the posterior parameter distributions of the ZIBG$(M, N, \theta, p, 0, 0)$ model applied to the ZIBG data.

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_M	0.58	1.07	1.47	1.07	0.232	0.011	453	-0.059	1.003
β_N	-1.22	-0.27	0.46	-0.34	0.454	0.034	177	-0.034	1.006
β_θ	-0.01	0.52	0.98	0.50	0.262	0.011	573	0.014	1.004
β_p	-0.05	0.50	1.02	0.50	0.284	0.023	155	-0.058	1.001

Table 1.3: Summary statistics on the posterior parameter distributions of the ZIBG$(M, N, \theta, p, 0, 0)$ model applied to the ZIBEEG data.

Using observed values $\mathbf{y}_1, \ldots, \mathbf{y}_n$ from random quantities $\mathbf{Y}_1, \ldots, \mathbf{Y}_n$, we will compare the models

$$\begin{aligned} \mathbf{Y}_1, \ldots, \mathbf{Y}_n &\sim \text{ZIBG}(M, N, \theta, p, 0, 0), & \mathbf{Y}_1, \ldots, \mathbf{Y}_n &\sim \text{ZIBEEG}(\theta_1, \theta_2, b_1, b_2, \lambda, p), \\ \log M &= \beta_M, & \operatorname{logit} \theta_t &= \beta_{\theta_t} \quad \text{for} \quad t = 1, 2, \\ \log N &= \beta_N, & \log b_t &= \beta_{b_t} \quad \text{for} \quad t = 1, 2, \\ \theta &= \frac{2\exp(\beta_\theta)}{1+\exp(\beta_\theta)} - 1, & \lambda &= \beta_\lambda, \\ p &= \frac{e^{\beta_p}}{2(1+e^{\beta_p})}; & p &= \frac{e^{\beta_p}}{2(1+e^{\beta_p})}. \end{aligned}$$

We generate $n = 200$ data points from each model with respective parameters (see Figure 1.3)

$$(\beta_M, \beta_N, \beta_\theta, \beta_p) = (1.25, 0.75, 0.85, 0) \quad \text{and} \quad (\beta_{\theta_1}, \beta_{\theta_2}, \beta_{b_1}, \beta_{b_2}, \beta_\lambda, \beta_p) = (2.5, 1.5, -0.3, 0, 5, 0). \tag{1.30}$$

The parameter values were chosen so that the generated data are comparable in magnitude and covariance. The data are generated using the algorithm in Listing B.1. Subsequently, we estimate the posterior parameter distributions using JAGS (see the corresponding model specifications in Listings B.5 and B.6), applying each model to each data set to obtain four sets of results. These results can be found in Tables 1.2 to 1.5, which show the MCMC algorithms have adequately converged. A small hiccup needed to be taken care of to generate the results: too large values of $|\lambda|$ lead to invalid probability distributions, and there are no explicit bounds given by Famoye (2019). If this is neglected, the Markov chains tend to infeasible values of λ due to the large (but incorrect) likelihoods they produce. This problem is mitigated by setting $\lambda = 0$ in the computation of the PMF at $(0, 0)$.

We compare the model fits using the *Deviance Information Criterion* (*DIC*, see Spiegelhalter et al. (2002)). The DIC is a generalization of the *Akaike Information Criterion* (*AIC*, see Akaike (1973)) to hierarchical modeling. Like the AIC, the DIC estimates the information lost (the

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_{θ_1}	2.19	2.43	2.65	2.43	0.119	0.004	769	0.020	1.001
β_{θ_2}	1.54	1.76	2.02	1.77	0.121	0.005	659	-0.043	1.003
β_{b_1}	-0.41	-0.15	0.07	-0.15	0.120	0.004	736	0.019	1.000
β_{b_2}	-0.09	0.17	0.40	0.17	0.128	0.005	694	0.003	1.003
β_λ	0.00	0.71	2.20	0.86	0.671	0.025	749	-0.060	0.999
β_p	-0.27	0.19	0.72	0.20	0.258	0.008	1123	-0.003	1.008

Table 1.4: Summary statistics on the posterior parameter distributions of the ZIBEEG$(\theta_1, \theta_2, b_1, b_2, \lambda, p)$ model applied to the ZIBG data. Chains 5 and 6 were dropped due to poor mixing.

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_{θ_1}	2.24	2.47	2.70	2.47	0.118	0.004	997	0.035	1.003
β_{θ_2}	1.30	1.53	1.79	1.54	0.121	0.005	578	-0.019	1.010
β_{b_1}	-0.22	-0.00	0.25	-0.00	0.122	0.005	613	0.033	1.002
β_{b_2}	-0.13	0.12	0.37	0.12	0.127	0.005	583	0.005	1.008
β_λ	0.89	3.36	5.65	3.33	1.242	0.044	804	0.033	1.004
β_p	-0.06	0.44	1.06	0.45	0.283	0.010	841	-0.010	1.002

Table 1.5: Summary statistics on the posterior parameter distributions of the ZIBEEG$(\theta_1, \theta_2, b_1, b_2, \lambda, p)$ model applied to the ZIBG data.

Kullback-Leibler divergence, see Kullback and Leibler (1951)) by supposing a particular model. Letting $D(\boldsymbol{\theta})$ denote the deviance of the model parameterized by a random parameter vector $\boldsymbol{\theta}$, it is defined as

$$\text{DIC} = \mathbb{E}\{D(\boldsymbol{\theta})\} + p_D, \quad \text{where} \quad p_D = \mathbb{E}\{D(\boldsymbol{\theta})\} - D(\mathbb{E}\,\boldsymbol{\theta}).$$

The penalty p_D estimates the effective number of model parameters. When a posterior sample is available, it is common to replace the expected values by sample means. As with the AIC, the model with the lowest DIC should be preferred. Naturally, the data should remain constant when making comparisons. When computing the deviances, we neglect the additional term corresponding to the marginal probability of the data. Doing so is valid as we are only interested in differences between deviances, in which the additional terms cancel. For a detailed treatment of information criteria and their underlying philosophy, see Konishi and Kitagawa (2008).

On the ZIBG data, the estimated ZIBG and the ZIBEEG models yield respective DIC values 2076.20 and 2102.69, while on the ZIBEEG data, the respective DIC values are 1993.90 and 1997.47. These results show that the ZIBG model is an improvement on both data sets compared to the ZIBEEG model. The improvement is far greater on the ZIBG data since the DIC values are more than 25 apart (suggesting the model with the lower DIC is over $100\,000$ times more probable), but it is debatable for the ZIBEEG data (where the model with the lower DIC is only 5-10 times more probable). These findings support the claim that a ZIBG model can be preferable when modeling certain data and that it is not significantly worse on data that favors a competing model.

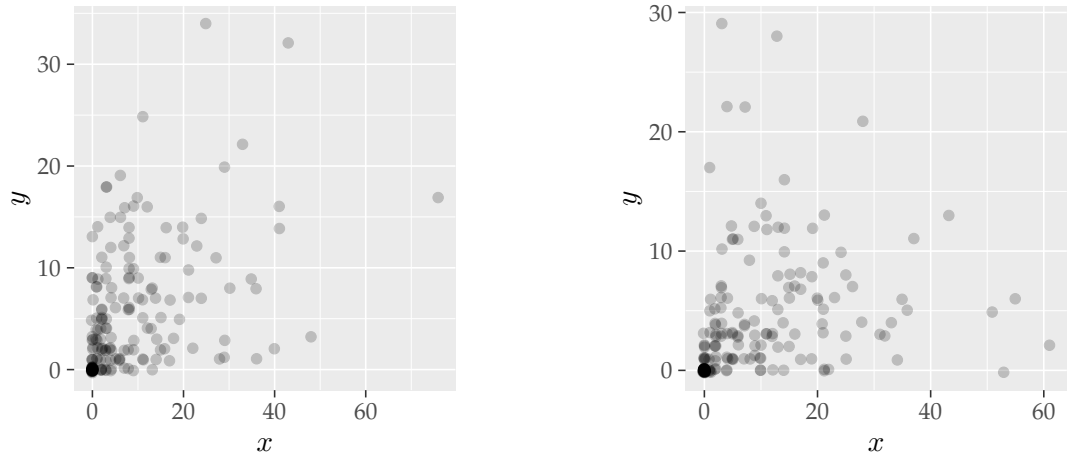


Figure 1.3: Data sampled from the ZIBG (left) and the ZIBEEG (right) models with respective parameter vectors (1.30) using the MCMC algorithm given in Listing B.1.

1.5 Discussion

Perhaps the most important consideration when estimating the proposed Bayesian model is the computational cost, which becomes more evident when a large thinning factor is needed (something unavoidable when including random effects). The underlying reason for this inefficiency lies in our definition of the ZIBG distribution. The sum in the PMF presented in Proposition 1 needs many evaluations when both components are high. In addition, a severe loss of precision due to near-zero denominators and huge multinomial coefficients may lead to highly imprecise probabilities in such cases. As a result, multi-precision floating-point numbers need to be used when implementing the probability evaluations to prevent underflow and rounding of large numbers in intermediate computations, consuming more memory and computing time. Despite this drawback, our definition of a bivariate geometric distribution provides easily interpretable parameters. In particular, the correlation parameter θ resembles the actual correlation when the marginal means are sufficiently large. This feature facilitates meaningful direct modeling of the correlation parameter, which is a necessity in assessing the similarity of the components in the empirical data analysis (cf. Chapter 2).

The process may be sped up by using maximum likelihood estimation instead. Although doing so is straightforward for fixed effect models, computing the Laplace approximations can be arduous when dealing with mixed effects. The `TMB` package in R uses the automatic differentiation library `CppAD` in C++ to efficiently approximate the marginal likelihood in a large range of models. However, the use of multi-precision floating-point numbers in the probability mass computations for the ZIBG distribution is incompatible with `CppAD`. The consequence is that all needed partial derivatives have to be implemented and supplied manually, which would be a lengthy and tedious process. A better alternative is likely to write a Laplace approximation in R or to take on a different integrand approximation altogether.

We have taken the geometric distribution as our starting point in deriving a suitable zero-inflated bivariate count distribution. The primary reason for doing so was its expressibility in terms of the median and its relative simplicity. The negative binomial distribution is a generalization that supports dispersion modeling but lacks a closed-form median expression. The

same criticism applies to overdispersed Poisson and binomial distributions, which also add the disadvantageous need to use quasi-likelihoods. However, the discrete Weibull distribution (the discretized version of the continuous Weibull distribution) simultaneously allows median parameterization and overdispersion modeling. As mentioned earlier, we might extend this idea to two (or more) dimensions, for instance, by discretizing the bivariate Weibull distribution proposed by Sagias and Karagiannidis (2005) (see equations (11) and (17) for a bivariate distribution or equations (23) and (28) for a multivarate variant). Including zero-inflation and median parameterization could then be analogous to the approach followed in this thesis. Despite this possibility, the model that we derived from a geometric distribution is applicable in a range of data sets where over- or underdispersion is modest. Furthermore, the extension to higher dimensions using a PGF of the form (1.18) is conceptually simple. A potential pitfall is finding a closed-form expression for the PMF, which may involve even more nested sums. The parameters correlation parameters may be assigned some covariance structure that solely depends on their indices, as is prevalent in mixed-effects modeling, to reduce the complexity. A successful attempt to extend to higher dimensions could offer a more easily interpretable alternative to copula-based approaches (Nikoloulopoulos, 2015) for multivariate responses in which each component is approximately geometrically distributed.

The proposed ZIBG model was compared with the ZIBEEG model given by Famoye (2019). Since no procedure to estimate the posterior parameter distribution is available, one needed to be implemented. A problem with the given PMF is the lack of bounds on the similarity parameter λ. As the equation of the correlation coefficient (Famoye, 2019, pg. 437) shows, λ cannot be arbitrarily large in magnitude. The issue is that its bounds depend on the failure probabilities and the dispersion parameters. Unrestricted, the estimation procedure may excessively increase λ in data sets containing a large number of $(0, 0)$ points, leading to parameter sets for which the PMF exceeds one at $(0, 0)$. We used an ad-hoc solution that takes λ out of the computation of the PMF at $(0, 0)$ to circumvent this problem. Doing so leads to more reasonable sets of parameters that yield valid probabilities, but there is a possibility that the most optimal feasible solution has not been reached. Neglecting this fact is likely unproblematic, given that the obtained parameters are sufficiently close to those in the data generating mechanism. Still, a more rigorous comparison may be made if proper restrictions on the similarity parameter could be implemented.

1.6 Concluding remarks

In this chapter, we have proposed a Bayesian bivariate response mixed-effects model that is robust with respect to extreme outliers. To do so, we started by deriving a closed-form expression for the PMF of the bivariate geometric distribution given in Jayakumar and Mundassery (2007) while simultaneously extending it to support both positive and negative correlation. We then defined a zero-inflated bivariate geometric distribution analogous to the zero-inflated bivariate Poisson distribution in Li et al. (1999), allowing for zero-inflation at the component axes as well. The distribution was reparameterized in terms of its marginal medians taking the zero-inflation into account. In a univariate setting, we showed that maximum likelihood estimates of the medians are less sensitive than means to changes in the nonzero observations' sample average. The robustness rapidly increases as the proportion of zeros in the data tends to a half. Marginally speaking, this result also applies to a zero-inflated bivariate geometric distribution. Using this distribution, we defined a mixed-effects model that allows random effects in the marginal median models and a non-constant correlation parameter model. We provided a Bayesian specification using non-informative priors and implemented an estimation procedure

using JAGS with a customized module in R. The implementation's validity and the model's identifiability were demonstrated in a simulation study. In addition, it was shown that the model was convincingly superior to a competing model (Famoye, 2019) in certain situations and not significantly worse on data sets that favored the latter.

Considerable improvements to the computational efficiency should be made to the estimation procedure to make the inclusion of random effects feasible on large data sets. An explorable option would be rewriting the probability mass function in a way that rules out underflow or rounding of large numbers in intermediate computations. Doing so would remove the need to use multi-precision libraries that severely slow down the probability computations, but the sum appearing in the probability mass might make this impossible. Another strategy could be to store and reuse the factors from the computation of the previous term in the sum. Although more storage would be required, such a modification could notably reduce the number of multiplications needed per term for observations where both components are large. If Bayesian is not strongly preferred to frequentist inference and the main interest lies in the parameter's posterior means, a substantial gain in performance may be obtained by manually implementing a Laplace (or some other) approximation; proceeding through marginal likelihood maximization. When doing so, it is vital to keep the philosophical distinction between Bayesian and frequentist inference on the randomness of the model parameters in mind, as accurate distributional approximations of maximum likelihood estimates rely heavily on large sample sizes.

The current model definition does not provide dispersion control and is limited to two-dimensional response data. However, an approach to replicate the derivation on a discrete Weibull distribution has been outlined, allowing dispersion modeling while retaining marginal median parameterization. Moreover, a multivariate probability generating function has been suggested from which a multivariate zero-inflated geometric distribution can be derived. Albeit more complex, higher dimensions could also be supported using a multivariate discrete Weibull distribution. While the proposed model already supports a reasonable range of two-dimensional count data with similarly distributed components, such a model would be applicable in a much more general setting.

Chapter 2

Statistical analysis of the green hawker populations

2.1 Introduction

Green hawker (Aeshna viridis) dragonfly habitation relies on its host plant called the water soldier (Stratiotes aloides). The green hawkers inject their eggs into these plants, and after hatching, the larvae remain in the vegetation for one to three years. Endangerment threatens the species, and measures need to be taken to counteract this risk. In the Netherlands, the water soldier is commonly found in ditches that separate patches of agricultural land. Extensive agriculture has severely sped up the species' growth, but the abundance of water soldiers has caused a thickened layer of sludge formed by the decaying plants at the bottom of the ditches they inhabit. The sludge layer leads to a deteriorated water quality, eventually killing off the water soldiers. A fraction of the water soldiers must periodically be removed to prevent this from happening. The current approach is to extract the plants on one side of each ditch at the end of summer. Milder-Mulderij et al. (2019) propose a new method that involves roughly the same resources; square patches are cleared on alternating sides of the ditch instead, creating two rows of a checkered pattern. Doing so should leave the vegetation in a more natural pattern and creates more border length between the plants and the water, which is the preferred spot for the green hawkers. The hypothesis is that this approach will benefit the species.

The data contain green hawker population measurements at 17 locations across Groningen, Friesland, and Drenthe in the Netherlands from 2015–2018. The locations are ditches or a ponds divided into two transects. Each transect is randomly assigned either the regular or the checkered water soldier removal pattern (referred to as treatment). The dragonflies were counted during six sessions each summer from 2015–2018. The first three sessions were primarily focused on collecting exuviae (exoskeletons shed during metamorphosis), while mainly live adults were counted during the remainder. However, adults were also counted in the first sessions and vice versa. The counts in the data correspond to the total numbers recorded by two biologists in 45 minutes. Apart from the dragonfly populations, the host plant emersion was recorded as the water surface coverage percentage during each session. Each year's early spring (except in 2015) and late summer, some abiotic water quality factors were measured, such as the oxygen level, the water temperature, and the acidity.

The study's primary purpose is to analyze the effect of the new water soldier management strategy on the green hawker presence. The statistical analysis presented in this thesis shows

that there is no statistically significant associated improvement. Nevertheless, a secondary research problem is to find a fitting model for the dragonfly counts. Ecological data are frequently modeled using GLMMs (Kruger and Morin, 2020; Reimchen and Bergstrom, 2009; Zhang et al., 2017; Zuur et al., 2009). Doing so, instead of merely applying a linear model, is often required by repeated measurements on individuals and responses that are binary, counts, or represent proportions (Bolker et al., 2009). Although there are instances in literature that impose statistical models on dragonfly or damselfly (larval) population data (Benke and Benke, 1975; Van Buskirk, 1993; Nakanishi et al., 2018; Sherrat et al., 2011; Termaat et al., 2019), there is little record of GLMM usage in this setting. Such is expected in this relatively specific area of expertise, so this statistical analysis should hardly be considered a contribution solely based on the applied model's classification. Rather, the novelty is the modeling of a correlation parameter in a bivariate-response model that incorporates both measures of the dragonfly's presence, namely exuviae and adult counts. It is hard to find literature on generalized linear models for correlations, as correlations are rarely taken to vary with some covariate. The unconventional approach in this thesis analyzes the similarity of the different population measurements as environmental factors vary. It turns out that, according to the model estimates, the sludge thickness has a significant impact on the correlation. In particular, minimum amounts of sludge are associated with a decrease of 25%, on average around 0.15, compared to the correlation at median to maximal sludge thickness. This finding suggests the measures are not substitutes in such conditions.

This chapter is organized as follows. Section 2.2 lists the steps taken in the process of data preparation and additionally serves to provide descriptive statistics. An informal, exploratory analysis is conducted in Section 2.3. The formal analysis involves regressions for conventional univariate-response mixed models an the robust Bayesian bivariate response model proposed in Section 1.3. The results are presented in Section 2.4 and 2.5, respectively. Section 2.6 discusses the critical aspects of the analysis, and some concluding remarks are given in Section 2.7.

2.2 Data preparation and description

2.2.1 Preparation

The initially provided data set is quite extensive and requires some preparation to be suitable for statistical inference. There are three distinct data sets, respectively containing the abiotic factors, the host plant emersion, and the dragonfly counts. We start by listing their modifications separately. Subsequently, we describe the merging of the data sets and the computation of yearly aggregates.

Abiotic factors

Once in 2015 (late summer) and twice in each consecutive year (early spring and late summer) up to and including 2018, some abiotic factors relating to the water quality were recorded on each location. These factors are the water's acidity, redox, oxygen levels, dissolved oxygen levels, electrical conductivity (EC), temperature, transparency, depth, sludge depth, sludge thickness, color, and texture. Although technically not an abiotic factor, the host plant emersion (also simply referred to as *emersion*) was also included in each entry.

Some redundant variables need to be removed. For each entry, the first and second observer is included, but the large number of different observers would make it hard to correct for this factor in any modeling scenario. A similar criticism applies to general comments, water color, and water texture, which all contain over ten levels. Also, since data are only recorded during

times of sufficiently average weather, it makes sense to remove the start and end times (always 45 minutes apart) and any weather-related variables (air temperature, wind, and cloudiness). Additionally, the client has mentioned that host plant submersion, water transparency, and dissolved oxygen measurements are unreliable and should be removed.

Each entry contains ten measurements of water depth, sludge depth, and sludge thickness; we retain only the average of these measurements in each instance. Emersion percentages '< 1', '< 5' and '< 15' are replaced by their mode 0 to keep this variable numeric; alternatively, they could have been replaced by their respective category means 0.5, 2.5, and 7.5.

The units of measurement were changed for some variables that attain relatively large values. In particular, percentages are converted to fractions for host plant emersion and oxygen (divided by 100), redox is expressed in volts instead of millivolts (divided by 1000), electrical conductivity is converted from microsiemens to millisiemens (divided by 1000), and centimeters are converted to meters for water depth, sludge depth and sludge thickness (divided by 100).

Some locations have transect codes not equal to 'a' or 'b' and no treatment has been specified in these cases; such entries are removed. In 2015, some locations were selected to receive the new treatment, even though the host plant management occurred after the population measurements. Therefore, in each entry recorded in 2015, the treatment is set to 'regular'.

Stratiotes coverage

Once in late summer 2015 and six times during the summer in 2016–2018, the host plant emersion was recorded as the water surface coverage percentage. The data set containing these measurements will only be appended to the dragonfly counts, so only the area, transect code, treatment, date, and host plant emersion percentage are relevant. We set transect codes not equal to 'a' or 'b' to NA and treatments not equal to 'regular' or 'checker' to 'none'. As before, we convert emersion percentages to fractions.

Dragonfly counts

In each summer of 2015–2018, dragonflies were tallied at each location on six occasions. The first three were primarily focused on collecting (and counting) exuviae and the others on counting adults, but in no session was the recording exclusive towards the other dragonfly stage. The variables that signify the session number can be dropped since we will later aggregate the counts to yearly totals.

These data contain counts for numerous dragonfly species besides Aeshna viridis. Since those counts are rarely positive and, if so, very low, we disregard them for this analysis. Only a small fraction of the entries has a comment, so this feature is also removed. The dragonflies were classified according to their stadium (exuviae or adult), their sex (male, female or unknown), and the way they were recorded (estimated, collected, or observed while metamorphosing, flying, egg-laying, or patrolling). We sum exuviae, adult, and egg-laying female counts in each entry. Doing so for exuviae and adults is necessary since the recording methods (collecting remains versus counting live specimens) are intrinsically different, and we are interested in analyzing their (dis)similarity. Considering also the egg-laying female counts was suggested by the client, as these adults are more strongly tied to the location at which they were spotted.

As with the abiotic factors data set, we remove entries where no treatment was specified and set all treatments in 2015 to 'regular'. The two other data sets were merged with the dragonfly counts by appending the entries with the closest dates, thereby assuming that piecewise constant abiotic factors and host plant emersion give adequate representations of reality. Finally, we compute yearly aggregates for each transect of each location by summing the dragonfly

counts and averaging the abiotic factors and the host plant emersion levels. We drop the date and, naturally, only maintain the year.

Prepared data

The first six observations (out of 114) of the resulting complete data set containing yearly aggregates can be found in Table 2.1. Analogously, Table 2.2 shows the first six observations (out of 172) of the data on the abiotic factors.

	Manager	Area	Transect	Treatment	Year	Exuviae	Adults
1	GL	GL1	a	regular	2015	99	11
2	GL	GL1	b	regular	2015	69	11
3	GL	GL1	a	checker	2016	72	38
4	GL	GL1	b	regular	2016	59	32
5	GL	GL1	a	checker	2017	16	22
6	GL	GL1	b	regular	2017	15	24

	Eggl. females	Emersion (fraction)	pH	Redox (V)	Oxygen (fraction)	EC (mS/cm)
1	6	0.65	6.70	0.0354	0.264	0.369
2	4	0.65	6.96	0.0650	0.566	0.376
3	10	0.63	7.40	0.2380	0.690	0.224
4	10	0.63	7.60	0.2250	0.720	0.245
5	11	0.63	7.30	0.1760	0.820	0.706
6	13	0.63	7.10	0.1710	0.630	0.759

	Temperature (C)	Water depth (m)	Sludge depth (m)	Sludge thickness (m)
1	14.80	0.485	0.875	0.390
2	15.50	0.460	1.006	0.546
3	7.30	0.514	0.893	0.379
4	7.20	0.532	1.030	0.498
5	18.20	0.326	0.724	0.398
6	18.10	0.282	0.788	0.506

Table 2.1: The first six observations of the complete data set after preprocessing.

2.2.2 Description

We conclude this section with some descriptive statistics on the acquired data sets. Table 2.3 shows the number of observed levels for each factor in the complete data set. Most locations have eight corresponding measurements, one for each year and transect. Some have less due to the experiment being very unsuccessful (Milder-Mulderij et al., 2019) or because no treatment was applied (this is also the reason for the higher number of measurements in 2015). For each location, the treatments have been randomly assigned to the transects. Because we set all treatments to regular in 2015, there are fewer measurements for the checker treatment in total. Table

	Manager	Area	Transect	Treatment	Year	Month	Date
1	GL	GL1	a	regular	2015	September	18/09/2015
2	GL	GL1	b	regular	2015	September	18/09/2015
3	GL	GL1	a	checker	2016	March	20/03/2016
4	GL	GL1	b	regular	2016	March	20/03/2016
5	GL	GL1	a	checker	2017	March	23/03/2017
6	GL	GL1	b	regular	2017	March	23/03/2017

	Emersion (fraction)	pH	Redox (V)	Oxygen (fraction)	EC (mS/cm)
1	0.65	6.70	0.0354	0.264	0.369
2	0.65	6.96	0.0650	0.566	0.376
3	0.75	7.40	0.2380	0.690	0.224
4	0.75	7.60	0.2250	0.720	0.245
5	0.05	7.20	-0.0398	0.648	0.418
6	0.01	7.41	-0.0140	1.019	0.448

	Temperature (C)	Water depth (m)	Sludge depth (m)	Sludge thickness (m)
1	14.80	0.485	0.875	0.390
2	15.50	0.460	1.006	0.546
3	7.30	0.514	0.893	0.379
4	7.20	0.532	1.030	0.498
5	10.00	0.562	1.143	0.581
6	8.84	0.760	1.084	0.324

Table 2.2: The first six observations of the abiotic factors data set after preprocessing.

2.4 contains the factor level counts for the abiotic factors data set; similar explanations apply to the difference among levels.

Summary statistics on the numeric variables in the complete data set can be found in Table 2.5. Recall that the statistics relate to yearly aggregates: dragonfly counts are summed, and the other variables are averaged. The significantly higher mean and standard for the exuviae compared to the adults and egg-laying females is likely due to two extreme outliers (250 and 344, although the client confirmed their validity), as supported by the relatively low median. Neglecting the exuviae counts, the slightly higher means compared to the medians of the adult and egg-laying female counts indicate these counts are right-skewed.

We observe that, although a total absence of water soldiers occurs, the average surface coverage fraction (emersion) is reasonably high at approximately 0.8. The acidity (pH) of the water is generally neutral, although somewhat acidic and slightly alkaline water is sometimes observed. In well-oxidized water, the redox potential should remain in the range 0.3–0.5 V (Søndergaard, 2009); the summary statistics, therefore, indicate the contrary for most measurements. The oxygen percentage (given as a fraction) ranges from 0 to oversaturated at 1.44. The electrical conductivity (EC) ranges from 0.133 to 1.078 mS/cm (micro-Siemens per centimeter), which contains about double the amounts in the typical drinking water range. Measuring electrical conductivity is a convenient method to estimate ionization, an essential factor relating to a broad range of biochemical processes taking place in the water. Average water temperatures re-

Manager	Area		Transect	Treatment	Year
GL: 24	GL1: 8	SBBF3: 8	a: 57	regular: 73	2015: 32
GV: 22	GL2: 8	SBBG1: 2	b: 57	checker: 41	2016: 28
SBBF: 24	GL3: 8	WHA1: 8			2017: 28
SBBG: 2	GV1: 6	WHA2: 8			2018: 26
WHA: 42	GV1/2: 2	WHA3: 2			
	GV2: 6	WHA4: 8			
	GV3: 8	WHA5: 8			
	SBBF1: 8	WHA6: 8			
	SBBF2: 8				

Table 2.3: Factor level counts in the complete data set.

Manager	Area		Transect	Treatment	Year	Month
GL: 36	GL1: 12	SBBF3: 12	a: 86	regular: 102	2015: 32	March: 84
GV: 34	GL2: 12	SBBG1: 4	b: 86	checker: 70	2016: 30	September: 72
SBBF: 36	GL3: 12	WHA1: 12			2017: 56	October: 16
SBBG: 4	GV1: 10	WHA2: 12			2018: 54	
WHA: 62	GV1/2: 2	WHA3: 2				
	GV2: 10	WHA4: 12				
	GV3: 12	WHA5: 12				
	SBBF1: 12	WHA6: 12				
	SBBF2: 12					

Table 2.4: Factor level counts in the abiotic factors data set.

mained between 5 and 21 degrees Celsius. The water depth, sludge thickness, and sludge depth are roughly related through the sum of the former two equaling the latter; only two measurements do not satisfy this relationship (one is off by 1 cm and the other by 1 mm). To avoid issues caused by multicollinearity, we will not consider the sludge depth in any formal regression.

The summary statistics in Table 2.6 are comparable to those in Table 2.5. The differences are due to the additional measurements, which are incidentally never the closest to the dragonfly counts concerning dates, at the beginning of spring. The consequence is that the averages presented in Table 2.5 only relate to the measurements of the abiotic factors that occurred at the end of summer. This aspect most notably leads to differences in the respective means and medians.

2.3 Exploratory analysis

Before conducting a formal analysis, we will present an exploratory (or *informal*) examination of the data. This practice provides essential orientation and narrows down the set of plausible regression scenarios.

	Mean	Standard deviation	Minimum	Median	Maximum
Exuviae	16.325	42.730	0	3	344
Adults	12.974	13.773	0	8	64
Egg-laying females	5.509	6.875	0	3	32
Emersion (fraction)	0.770	0.246	0.000	0.842	1.000
pH	7.008	0.549	4.900	6.965	8.100
Redox (V)	0.101	0.068	-0.087	0.107	0.285
Oxygen (fraction)	0.560	0.286	0.000	0.559	1.440
EC (mS/cm)	0.521	0.239	0.133	0.486	1.078
Temperature (C)	13.952	4.202	5.500	15.200	21.000
Water depth (m)	0.622	0.182	0.232	0.652	1.098
Sludge depth (m)	0.932	0.218	0.355	0.950	1.414
Sludge thickness (m)	0.311	0.149	0.063	0.280	0.746

Table 2.5: Summary statistics of numeric variables in the complete data set (114 observations).

2.3.1 Empirical dragonfly count distributions

Let us start by inspecting the empirical densities of the exuviae counts for different treatments, managers, and years. As Figure 2.1 shows, there is only a negligible difference between treatments when considering the pooled sample of all observations. Measurements for the year 2015 were left out of the empirical density evaluation to facilitate making a fair comparison; in 2015, counts were substantially higher, and the regular treatment was applied everywhere.

There are more pronounced differences when considering the observations per manager, but none of the treatments consistently yield increased dragonfly populations compared to the other, cf. Figure A.1. The checker treatment is favorable for GV, while the regular treatment gives better results for SBBF and WHA. The empirical densities for each year in Figure A.2 are quite different from the ones shown in Figure A.1. The impression is that the checker treatment performs better in 2017, while the converse holds in 2016 and 2018.

The regular treatment seems to be associated with slightly higher adult counts, cf. Figure 2.1b. At the manager level, Figure A.3 shows that regular treatment leads to relatively higher counts for GL and SBBF, while the checker treatment looks preferable for GV. These indications, interestingly, do not resemble those related to the exuviae counts. Figure A.4 depicts that there is no convincingly superior treatment in any year. The distribution of the egg-laying female counts is more comparable to the exuviae count distribution, as shown in Figure 2.1c. As suggested by the client, a possible explanation could be the stronger tie of the egg-laying females to a specific location compared to arbitrary adults.

Finally, a possible indirect effect of the checker treatment compared with the regular treatment is a decrease in the average thickness of the sludge layer formed by decayed water soldiers. To explore this effect, we examine the empirical densities for each treatment group given in Figure 2.1d. Since the decrease is evident, we omit the manager- and year-specific empirical densities.

	Mean	Standard deviation	Minimum	Median	Maximum
Date	-	-	09/09/2015	-	05/10/2018
Emersion (fraction)	0.420	0.398	0.000	0.300	1.000
pH	7.061	0.549	4.900	7.100	8.250
Redox (V)	0.097	0.067	-0.087	0.104	0.285
Oxygen (fraction)	0.654	0.317	0.000	0.645	1.460
EC (mS/cm)	0.507	0.224	0.052	0.470	1.078
Temperature (C)	12.247	4.349	5.500	12.050	21.000
Water depth (m)	0.625	0.194	0.220	0.637	1.145
Sludge depth (m)	0.933	0.232	0.322	0.945	1.684
Sludge thickness (m)	0.307	0.150	0.063	0.276	0.746

Table 2.6: Summary statistics of numeric variables in the abiotic factors data set (172 observations).

2.3.2 Correlation heatmap

Figure 2.2 shows Spearman's rank correlation coefficients (Spearman's *rho*) for relevant variables. Contrary to Pearson correlation, Spearman's rho can assess monotonic relationships that are not necessarily linear. It is defined as the Pearson correlation between the rank values (i.e., the ordering of the observations) of two variables. When some ranks are tied, ranks may be replaced by so-called *midranks*: averaged ranks when considering all slight perturbations of the values. A precise formula for Spearman's rho can be derived using this prescription (Press et al., 1988, Section 14.6.1). Corresponding p-values may be computed using an appropriate t-distribution approximation.

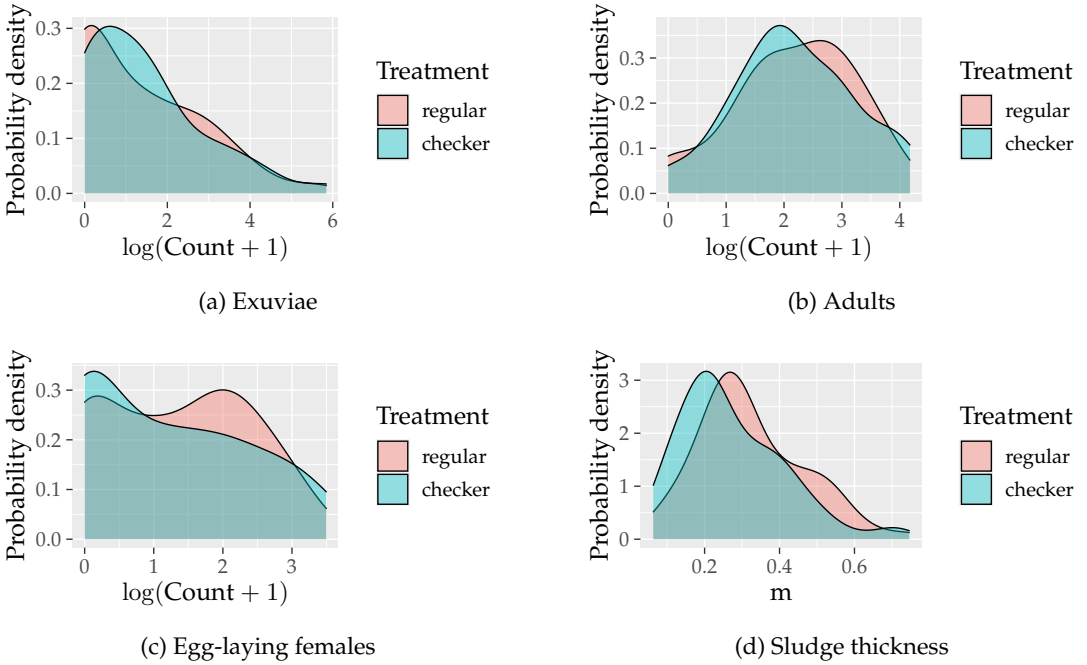
The exact coefficients and their p-values can be found in Table A.1 and Table A.2, respectively. It interesting to see that the negative correlation between exuviae counts and the checker treatment is statistically significant. A possible explanation is that the counts were highest in 2015, in which only the regular treatment is applied. We will see that this relation's statistical significance does not persist in a regression framework in Section 2.4. On the contrary, the negative correlation between the checker treatment and the sludge thickness (and as a result, the sludge depth) is demonstrable in a formal analysis.

Notably, the correlations between the dragonfly counts are all positive and statistically significant. All counts are moreover negatively correlated with the year, in line with their observed decline from 2015. Other considerably high and statistically significant correlations are those of year and redox (-0.307), year and conductivity (0.434), host plant emersion and water depth (0.284), pH and oxygen (0.413), and pH and conductivity (0.419).

2.3.3 Regression lines

Let us illustrate the effects of some of the continuous variables that are correlated with the dragonfly counts. We restrict our attention to the variables for which this correlation is statistically significant, cf. Table A.2, and discriminate between treatment groups. Figures 2.3, 2.4, and 2.5 show regression lines for each such a variable. The lines are the predicted means for each data point, obtained by fitting the GLMM

$$\text{count}_{it} \sim \text{NBin}(\mu_{it}, \phi), \quad \log \mu_{it} = \beta_0 + \beta_1 \,\text{variable}_{it} + b_i, \quad b_1, \ldots, b_{17} \sim \text{N}(0, \sigma^2)$$



(a) Exuviae

(b) Adults

(c) Egg-laying females

(d) Sludge thickness

Figure 2.1: Empirical densities for both treatments (2016–2018).

for each location i and each repeated measurement t, for both treatments separately. The negative binomial distribution with dispersion parameter ϕ allows modeling a great variety of count data, so it is a flexible option suitable for the exploratory stage of the statistical analysis. It is important to note that, although random effects are included in the model fitting, they are left out of the predicted means' computation. The 95% confidence bands are computed as the corresponding percentile bootstrap intervals (Wasserman, 2006, Section 3.4) for each data point.

According to Figure 2.3, the conductivity seems to be slightly negatively related to the exuviae counts for the checker treatment group. Some dissimilarity between the treatment groups is visible, but this may be partially due to the measurements in 2015. Although all regression lines are decreasing in their respective covariate, the other effects for the exuviae are not convincingly positive or negative in this setting. This ambiguity is implied by the confidence bands containing both increasing and decreasing regression lines.

The effects of redox and temperature on adult counts are more pronounced, as shown in Figure 2.4. All regression lines in the 95% confidence band for redox are increasing, while all are decreasing for temperature. In general, the differences between treatment groups are less visible with adults as the dependent variable for any correlated covariate compared to having exuviae as the response. The differences are negligible for redox and temperature, in particular.

Regarding egg-laying females, the regression lines in Figure 2.5 are to a great extent in agreement with Figure 2.4, but the confidence bands are wider in the former; this especially holds for the counts in relation to the host plant emersion. For both redox and temperature, the checker treatment group's confidence bands contain a small portion of negatively and positively sloped predicted mean lines, respectively. A possible explanation is the higher proportion of zero counts for the egg-laying females compared to the adults.

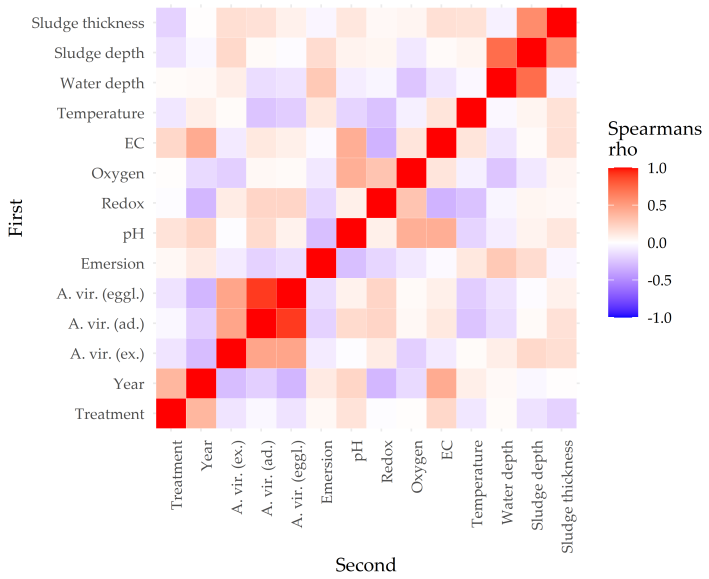
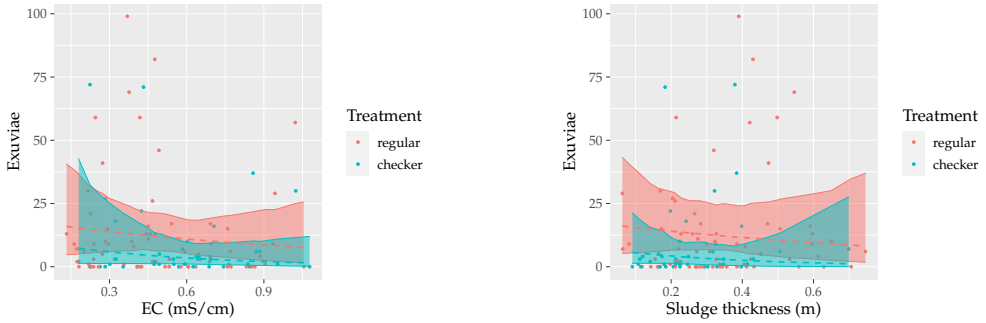


Figure 2.2: Spearman's rank correlation coefficients heatmap for important variables.



(a) (b)

Figure 2.3: Regression lines and bootstrapped 95% confidence bands of exuviae counts with respect to correlated (cf. Figure 2.2 and Table A.2) variables for both treatments (counts above 100 are omitted for clarity).

2.4 Univariate-response regressions

Given the zero-inflated bivariate geometric model proposed in Section 1.3, let us start our formal analysis by considering models of the form (1.6). Specifically, we let

$$\text{count}_{it} \sim \text{ZIG}(\mu_{it}, p), \quad \log \mu_{it} = \mathbf{x}_{it} \cdot \boldsymbol{\beta} + b_i, \quad b_1, \ldots, b_{17} \sim \text{N}(0, \sigma^2), \tag{2.1}$$

for each location i and repeated measurement t. Here, each count_{it} represents either the exuviae, adult, or egg-laying female count, and each covariate vector $\mathbf{x}_{it}$ contains a subset of

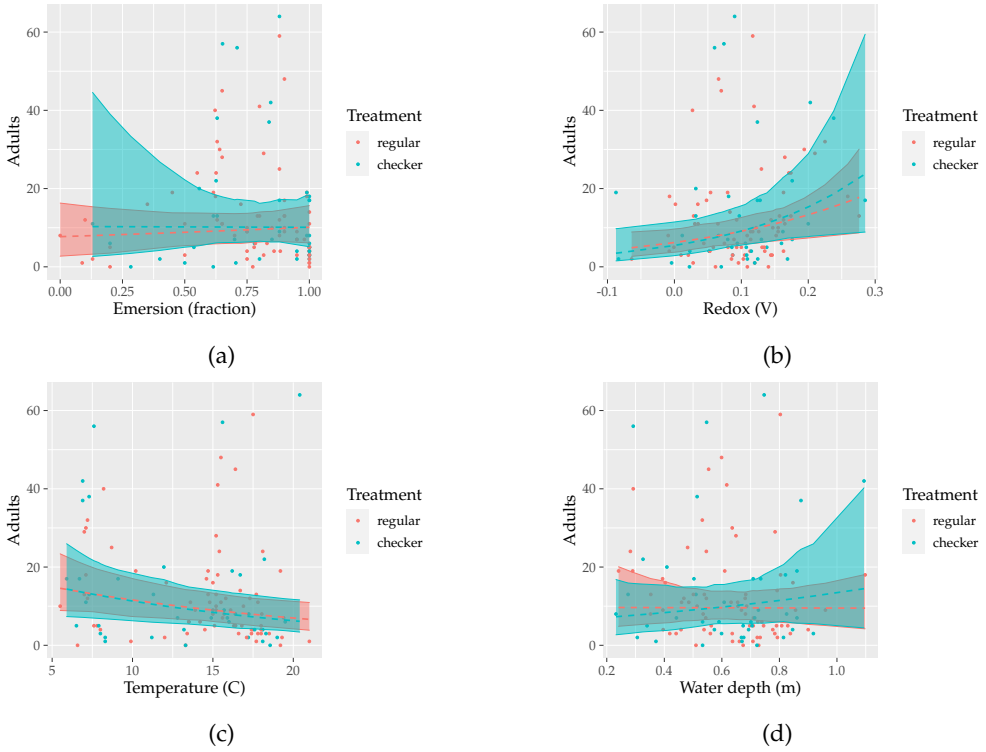


Figure 2.4: Regression lines and bootstrapped 95% confidence bands of adult counts with respect to correlated variables for both treatments.

the covariates manager, treatment, year, pH, redox, oxygen, conductivity, temperature, water depth, and sludge thickness.

In this section, we attempt to find reasonable univariate-response models for the different counts through selection, investigate the presence of any interaction effects, and assess the goodness-of-fit of any promising model. Once appropriate models have been found, the parameter estimates are discussed and interpreted from an ecological perspective.

2.4.1 Model selection

First of all, the most influential variables are selected for each response (exuviae, adult, and egg-laying females counts). The selection heuristic is backward elimination conditional on convergence to a solution, where we force treatment to be included. This procedure is implemented in the `buildmer` package and amounts to the following. The initial model contains all fixed effects and random intercepts that vary with location. The first step consists of finding the largest subset of the fixed effects and the location-varying random intercept such that the corresponding model is estimable; that is, convergence to an optimal set of parameter estimates can be obtained. Subsequently, all effects (including the random intercept) are ordered according to their contribution to the model, where the criterium is the significance of a likelihood ratio test. Elimination continues until the likelihood ratio test indicates that removing the least contributing variable leads to a significant disimprovement.

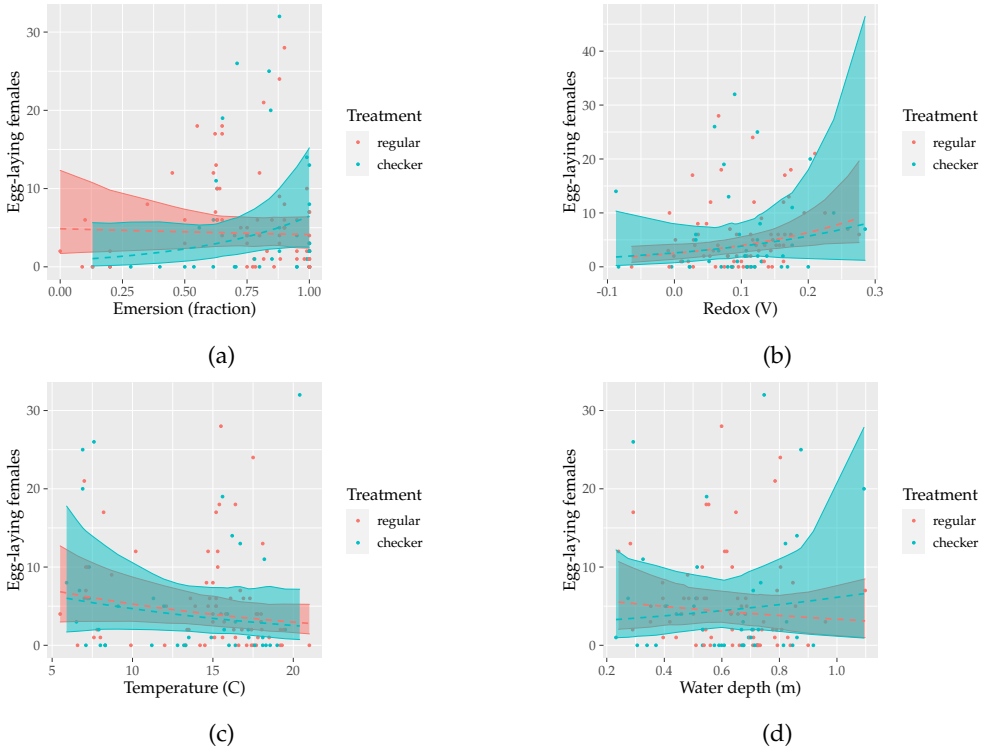


Figure 2.5: Regression lines and bootstrapped 95% confidence bands of egg-laying female counts with respect to correlated variables for both treatments.

The resulting models can be found in Table 2.7. Note that zero-inflation is included, while the estimates indicate it is not necessary. However, we will maintain it for later comparison to the zero-inflated bivariate-response model. Importantly, the estimated coefficient for treatment is not statistically significant in any model, nor is it of substantial magnitude. It is interesting to see that, according to the selected models, the adult and egg-laying female counts do not seem to depend on anything other than time and location.

Remark that no p-values are indicated in Table 2.7, but bootstrapped (basic) 95% confidence intervals are given instead. As Bolker (2017) notes, computing conventional p-values based on Wald tests requires multivariate normal sampling distributions of the parameters and a sampling distribution of the log-likelihood proportional to a χ^2-distribution. The situation is further complicated because no simple, reliable formula to compute the effective degrees of freedom. Although there are approaches to give adequate approximations, they are computationally quite demanding, and bootstrapping offers a conceptually simple and reasonable alternative. Parameter estimates can be considered significant when zero is not included in the 95% confidence interval.

2.4.2 Interaction effects

To assess whether there are interaction effects, we consider every first-order interaction using the remaining variables in Table 2.7 and repeat backward elimination (without forced inclusion of the treatment). The results are found in Table 2.8. For adult and egg-laying female counts, the

only possible interaction is an additional location-specific random effect for each year. Neither models significantly improve when including this effect, so apart from excluding a treatment effect, the ones in Table 2.8 are the same as in Table 2.7. Regarding the exuviae counts, an interaction effect between year and conductivity has been added, and temperature has been removed. Moreover, all parameter estimates are now significant, except for the interaction effect between conductivity and 2016. On the contrary, the estimates for the intercept, 2016, the temperature, the conductivity, and the zero-inflation of the exuviae model in Table 2.7 are not statistically significant. The AIC for the model that includes interaction effects is roughly 25 less, even though the number of parameters has increased by 2.

We assess the goodness-of-fit using randomized quantile residuals (cf. Section 1.2.4) by comparing them to the Unif$([0,1])$ distribution with a Kolmogorov-Smirnov test. The residuals are estimated using empirical distributions of samples of size $R = 2000$ from the predictive distribution for each observation. Doing so by invoking the `DHARMa` package yields respective p-values 0.20, 0.0033, and 0.18. Hence, the models for exuviae and egg-laying female counts are probable, but this cannot be said for adults.

The quantile-quantile plot in Figure 2.6 of the simulated residuals against the Unif$([0,1])$ distribution shows the residuals are generally too high, indicating that the underlying distribution is less skewed than the geometric distribution. Therefore, we consider models of the form

$$\text{adults}_{it} \sim \text{ZIP}(\mu_{it}, p), \quad \log \mu_{it} = \mathbf{x}_{it} \cdot \boldsymbol{\beta} + b_i, \quad b_1, \ldots, b_{17} \overset{\text{iid}}{\sim} \text{N}(0, \sigma^2) \tag{2.2}$$

instead. Performing backward elimination once more yields the results given in Table 2.9. A Kolmogorov-Smirnov test on the scaled residuals gives a p-value of 0.87, so this model is acceptable. Again, the coefficient for treatment is not statistically significant. Including all possible interaction effects of the covariates in Table 2.9 and repeating backward elimination, unfortunately, does not yield a reasonable model, as the Kolmogorov-Smirnov test on the scaled residuals results in a p-value near zero. Thus, we will disregard interaction effects on the adult counts by lack of a convenient systematic way to uncover them.

Since a geometric distribution is a bad fit, only the exuviae and egg-laying females will be considered in the remainder of the formal analysis. Naturally, because the marginal distributions of a bivariate geometric distribution are univariate geometric, adult counts would not be a suitable response in the bivariate regression model proposed in Section 2.5.

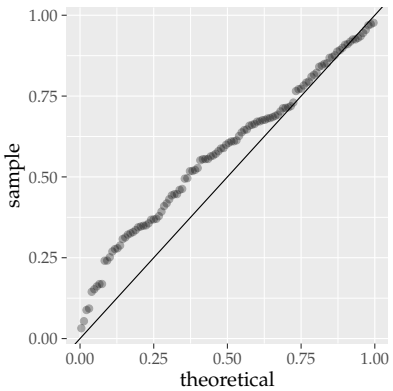


Figure 2.6: QQ-plot of the scaled residuals for the geometric adult count model against the Unif$([0,1])$ distribution, showing the residuals are left-skewed; this indicates there is a downward bias in the model.

2.4.3 Interpretation

The parameter estimates for the exuviae and egg-laying female count models from Table 2.8 will be taken as definitive, and Table 2.9 will be used for the adult counts. Naturally, only statistically significant estimates will be discussed, and all others are deemed as null.

Table 2.8 confirms the observed population decline from 2015–2018, with estimates for 2016–2018 being negative with a relatively large magnitude. Specifically, each year is associated with an approximate decrease of 90% in average counts relative to the previous, ceteris paribus. The water deterioration caused by the sludge seems to negatively impact the exuviae counts, with each additional cm of sludge thickness attributing a 3.5% decrease in average counts. On the contrary, a higher surface coverage fraction of host plants is related to increased average counts: 38% for each additional 10%. This result is not surprising considering the green hawkers' dependence on their host plant.

Regarding abiotic factors, only the water's oxygen level and conductivity seem to be influential. High levels of oxygen relate to decreased numbers of exuviae. To be specific, a 10% oxygen percentage increase is associated with a 25% decrease in average counts. Note that no causal relation is indicated, i.e., it is plausible that large numbers of larvae simply consume fair amounts of oxygen. The relationship between exuviae counts and conductivity, in contrast, is negative in 2015 but positive in 2017 and 2018. This variability is due to the interaction effect between year and conductivity. Each 100 μS/cm increase relates to a decrease of 52%, an increase of 8.8%, and an increase of 18% in 2015, 2017, and 2018, respectively. As before, these effects may be the result of the population decline of the green hawkers over the years.

Notice that zero-inflation is effectively estimated to be zero. Moreover, there is considerable variability between the different locations (areas), as the random intercept variance of 5.33 shows. Using the intercept estimate, we predict the mean of a hypothetical observation in 2015 with all covariates in the model set to zero as 250. Note that this observation does not fall within the observed ranges; for example, only positive sludge thickness and conductivity is observed.

Table 2.9 shows that average adult counts are significantly higher (203%) in 2016 and significantly lower (40%) in 2018 compared with 2015. Although this result agrees with Figure A.4, it is counter-intuitive given the observed population decline of the green hawker. As with exuviae, increased host plant presence leads to higher average adult counts. Specifically, each additional 10% water coverage is associated with a 14% increase. Concerning abiotic factors, water depth, sludge thickness, and water temperature are influential. An additional cm in water depth, cm in sludge thickness, or degree Celcius in water temperature relates to a 0.78% increase, a 1.2% decrease, or a 7.3% increase in average adult counts, respectively. Perhaps the water depth is correlated with the water surface area, whereby deeper water would indirectly enable more green hawker habitation. The amount of sludge is similarly (although more weakly) related to the number of adults compared to exuviae. The temperature increase effect is likely to be valid until some optimal value is reached. Note that the oxygen level and the conductivity effects are nearly statistically significant, as the upper bounds of their 95% confidence intervals are close to zero. Finally, the random intercept variance of 0.74 is notably smaller compared to the exuviae count model.

The average egg-laying female counts are predicted to equal 4.66 in 2015. The counts are significantly smaller in 2018, with an associated decrease of 73%. As with exuviae counts, the zero-inflated is estimated to be practically null. The random intercept variance of 0.76 is relatively small. It is interesting to see that this simple model fits the egg-laying female counts well, but the same does not hold for the adults.

	Dependent variable:		
	Exuviae	Adults	Egglaying females
(Intercept)	1.57	2.28*	1.54*
	[−1.06; 5.22]	[1.90; 2.78]	[1.00; 2.24]
Treatment (checker)	−0.08	0.00	−0.04
	[−0.86; 0.50]	[−0.58; 0.48]	[−0.59; 0.56]
Year (2016)	1.72	0.38	0.26
	[−0.48; 3.69]	[−0.14; 1.18]	[−0.42; 0.96]
Year (2017)	−1.07*	0.00	−0.18
	[−1.97; −0.31]	[−0.65; 0.84]	[−0.95; 0.68]
Year (2018)	−1.62*	−0.68	−1.30*
	[−2.69; −0.50]	[−1.29; 0.23]	[−2.26; −0.48]
Temperature (C)	0.24		
	[−0.00; 0.44]		
Oxygen (fraction)	−3.68*		
	[−5.21; −2.33]		
Sludge thickness (m)	−4.75*		
	[−7.61; −1.87]		
EC (mS/cm)	−2.40		
	[−4.37; 0.18]		
Emersion (fraction)	2.24*		
	[0.31; 3.68]		
Zero-inflation	−4.63	−21.37*	−20.60*
	[−6.83; 12.01]	[−40.41; −21.57]	[−39.12; −20.27]
AIC	719.26	804.71	608.38
Log likelihood	−347.63	−395.36	−297.19
Number of observations	114	114	114
Number of groups	17	17	17
Random intercept variance (Area)	5.41	0.48	0.76

* Null hypothesis value outside the confidence interval.
The reference treatment and year are 'regular' and 2015, respectively.

Table 2.7: Regression results for the models 2.1.

	Dependent variable:		
	Exuviae	Adults	Egglaying females
(Intercept)	5.52^*	2.28^*	1.54^*
	$[3.56; 7.55]$	$[1.85; 2.84]$	$[0.87; 2.20]$
Year (2016)	-2.25^*	0.38	0.24
	$[-4.28; -0.64]$	$[-0.20; 0.98]$	$[-0.40; 0.88]$
Year (2017)	-4.65^*	0.00	-0.20
	$[-6.66; -2.42]$	$[-0.52; 0.59]$	$[-0.77; 0.49]$
Year (2018)	-6.43^*	-0.68^*	-1.31^*
	$[-9.68; -2.75]$	$[-1.35; -0.10]$	$[-2.00; -0.54]$
Sludge thickness (m)	-3.55^*		
	$[-5.85; -0.71]$		
EC (mS/cm)	-7.37^*		
	$[-11.08; -2.36]$		
Emersion (fraction)	3.25^*		
	$[1.84; 4.96]$		
Oxygen (fraction)	-2.83^*		
	$[-4.35; -1.23]$		
Year (2016) $\times$ EC (mS/cm)	4.94		
	$[-0.39; 9.54]$		
Year (2017) $\times$ EC (mS/cm)	8.22^*		
	$[3.09; 12.53]$		
Year (2018) $\times$ EC (mS/cm)	9.05^*		
	$[2.88; 14.41]$		
Zero-inflation	-20.71^*	-21.39^*	-20.59^*
	$[-38.47; -20.02]$	$[-40.40; -21.46]$	$[-39.00; -19.97]$
AIC	695.01	801.65	605.54
Log likelihood	-334.50	-394.83	-296.77
Number of observations	114	114	114
Number of groups	17	17	17
Random intercept variance (Area)	5.33	0.48	0.76

* Null hypothesis value outside the confidence interval.
The reference treatment and year are 'regular' and 2015, respectively.

Table 2.8: Regression results for the models 2.1, including interaction effects.

	Dependent variable:
	Adults
(Intercept)	0.71
	[−0.73; 2.16]
Treatment (checker)	0.03
	[−0.10; 0.18]
Manager (GV)	−1.37
	[−2.65; 0.05]
Manager (SBBF)	0.08
	[−1.55; 1.58]
Manager (SBBG)	1.58
	[−0.34; 3.32]
Manager (WHA)	−0.05
	[−1.28; 1.47]
Year (2016)	1.11*
	[0.55; 1.60]
Year (2017)	0.15
	[−0.12; 0.35]
Year (2018)	−0.51*
	[−0.79; −0.25]
Oxygen (fraction)	−0.42
	[−0.82; 0.02]
Water depth (m)	0.78*
	[0.04; 1.73]
Emersion (fraction)	1.29*
	[0.94; 1.63]
Sludge thickness (m)	−1.20*
	[−1.77; −0.52]
EC (mS/cm)	−0.50
	[−1.04; 0.02]
Temperature (C)	0.07*
	[0.01; 0.11]
Zero-inflation	−3.84
	[−4.97; 14.25]
AIC	829.00
Log likelihood	−397.50
Number of observations	114
Number of groups	17
Random intercept variance (Area)	0.74

* Null hypothesis value outside the confidence interval.
The reference manager, treatment, and year are GL, regular, and 2015, respectively.

Table 2.9: Regression results for the model 2.2.

2.4.4 Regression of sludge thickness

Because a different model and data set are used when the sludge thickness is taken as the response, we give the results in a separate subsection. Because sludge thickness is continuous and non-negative, we will consider models of the form

$$\text{sludge}_{it} \sim \Gamma(\mu_{it}, \tau^2), \quad \log \mu_{it} = \mathbf{x}_{it} \cdot \boldsymbol{\beta} + b_i, \quad b_1, \ldots, b_{17} \sim \mathrm{N}(0, \sigma^2),$$

for each location i and repeated measurement t. Note that the Gamma distributions are parameterized by their mean and variance. Again, each $\mathbf{x}_{it}$ contains measurement from a subset of the covariates manager, treatment, year, pH, redox, oxygen, conductivity, temperature, and water depth. As before, the b_i correspond to location-specific random effects. The data on abiotic factors (cf. Tables 2.4 and 2.6) are used to take early spring measurements into account as well.

Backward elimination, as described in Section 2.4.1, yields the results in Table 2.10, where the dispersion parameter τ^2 is estimated as 0.143. A comparison of randomized quantile residuals to a Unif$([0, 1])$ distribution using a Kolmogorov-Smirnov test produces a p-value of 0.34, so the model is satisfactory. We observe that the checker treatment is associated with a 16.5% decrease in average sludge thickness, and this estimate is statistically significant. Note also the manager SBBG is estimated to have substantially larger average amounts of sludge.

	Dependent variable:
	Sludge thickness (m)
(Intercept)	-1.14^*
	$[-1.46; -0.70]$
Manager (GV)	-0.42
	$[-1.11; 0.12]$
Manager (SBBF)	-0.01
	$[-0.66; 0.45]$
Manager (SBBG)	0.61
	$[-0.20; 1.39]$
Manager (WHA)	0.14
	$[-0.32; 0.57]$
Treatment (checker)	-0.18^*
	$[-0.32; -0.07]$
AIC	-235.24
Log likelihood	125.62
Number of observations	172
Number of groups	17
Random intercept variance (Area)	0.08

* Null hypothesis value outside the confidence interval.
The reference manager and treatment are GL and 'regular', respectively.

Table 2.10: Regression results for the models 2.4.4.

2.5 Bayesian bivariate response model inference

Let $\mathbf{Y}_i$ denote the ith pair of exuviae and egg-laying female counts, and let z_i be the location index of the ith observations. We will apply the model defined in Section 1.3 to the data. Considering the model selections in Section 2.4, we let

$$\begin{aligned}
\mathbf{Y}_i &\sim \text{ZIBG}(M_i, N_i, \theta_i, p, r, s),\\
\log M_i &= \beta_{M0} + \beta_{M1}\,\text{year(2016)}_i + \beta_{M2}\,\text{year(2017)}_i + \beta_{M3}\,\text{year(2018)}_i\\
&\quad + \beta_{M4}\,\text{emersion}_i + \beta_{M5}\,\text{oxygen}_i + \beta_{M6}\,\text{conductivity}_i\\
&\quad + \beta_{M7}\,\text{sludge}_i + \beta_{M8}\,\text{year(2016)}_i \times \text{conductivity}_i\\
&\quad + \beta_{M9}\,\text{year(2017)}_i \times \text{conductivity}_i + \beta_{M10}\,\text{year(2018)}_i \times \text{conductivity}_i\\
&\quad + b_{Mz_i}\\
\log N_i &= \beta_{N0} + \beta_{N1}\,\text{year(2016)}_i + \beta_{N2}\,\text{year(2017)}_i + \beta_{N3}\,\text{year(2018)}_i\\
&\quad + b_{Nz_i}\\
\operatorname{logit}\theta_i &= \beta_{\theta0} + \beta_{\theta1}\,\text{year(2016)}_i + \beta_{\theta2}\,\text{year(2017)}_i + \beta_{\theta3}\,\text{year(2018)}_i\\
&\quad + \beta_{\theta4}\,\text{emersion}_i + \beta_{\theta5}\,\text{oxygen}_i + \beta_{\theta6}\,\text{conductivity}_i\\
&\quad + \beta_{\theta7}\,\text{sludge}_i + \beta_{\theta8}\,\text{year(2016)}_i \times \text{conductivity}_i\\
&\quad + \beta_{\theta9}\,\text{year(2017)}_i \times \text{conductivity}_i + \beta_{\theta10}\,\text{year(2018)}_i \times \text{conductivity}_i\\
p = \frac{e^{\beta_p}}{2C},\quad q &= \frac{e^{\beta_q}}{2C},\quad r = \frac{e^{\beta_r}}{2C},\quad \text{where}\quad C = 1 + e^{\beta_p} + e^{\beta_q} + e^{\beta_r},\\
b_{M1},\ldots,b_{M17} &\overset{\text{iid}}{\sim} \text{N}(0,\sigma_M^2),\\
b_{N1},\ldots,b_{N17} &\overset{\text{iid}}{\sim} \text{N}(0,\sigma_N^2).
\end{aligned} \tag{2.3}$$

We assign uninformative prior distributions as in Section 1.3 to the fixed effects, zero-inflation parameters, and random effect variances. That is,

$$\beta_{M0},\ldots,\beta_{M10},\beta_{N0},\ldots,\beta_{N3},\ldots,\beta_{\theta0},\ldots,\beta_{\theta10},\beta_p,\beta_q,\beta_r \sim \text{N}(0,100),$$

and

$$\sigma_M^{-2},\sigma_N^{-2} \sim \text{MGH-}t(10,2,1).$$

Note that

$$\text{MGH-}t(10,2,1) = \Gamma\left(1,\frac{1}{8\omega}\right),\quad \text{where}\quad \omega \sim \Gamma(0.5,0.01).$$

The indices $j = 1,\ldots,17$ in the random intercepts b_{Mj}, b_{Nj} correspond to the areas as ordered in Table 2.3.

The posterior parameter distributions of the model (2.3) are estimated using a Metropolis-Hastings MCMC algorithm, as in Section 1.4. Eight independent chains are run in parallel to be able to assess then chains' convergence. Each chain is initialized with all fixed effects, random effects, and zero-inflation parameters set to zero and the inverse random effect variances set to one. The algorithm is executed using JAGS, which we call from R using the package `runjags`. We will use the `base::Mersenne-Twister` for random number generation. The

chains' seeds are random numbers between 0 and 100 000 (sampled as 76 107, 70 223, 51 250, 1609, 91 610, 34 600, 50 245, 21 839). The first 1000 iterations comprise an adaptation phase in which sampler behavior is optimized. The next 250 000 iterations are the burn-in period, and the remaining 2 500 000 are thinned by a factor 250 000 to generate a sample of size 100. The script with the model definition and the call to `runjags` can be found in Listing B.7.

The simulation summary can be found in Tables 2.12 and 2.13. For either table, the first and third columns list the lower and upper end of the 95% confidence interval, respectively, computed as the 0.025 and the 0.975 sample quantiles. The sixth column is the associated Monte-Caro standard error, given as the standard deviation divided by the square root of the effective sample size (the sample size summed across chains and adjusted for autocorrelation, given in the seventh column). Column eight lists the autocorrelation of the sample. Finally, the Gelman-Rubin convergence statistic's potential scale reduction factor is given in the last column; values too far from one indicate a lack of convergence. By convention, we take 1.05 as the threshold. Note that no autocorrelation is alarmingly high, and all potential scale reduction factors are below or very close to 1.05. Figures A.7 and A.8 show that no chain visually exhibits poor mixing.

Posterior means can be considered significantly different from zero when their 95% confidence interval excludes zero. Such is the case for all fixed effect parameters β_{Mj}, except when $j = 4, 7, 9$, although the left interval ends for β_{M4} and β_{M9} are relatively close to zero. The posterior means closely resemble the parameter estimates in the first column of Table 2.8, except for the effect β_{M7} of sludge thickness. Regarding the marginal medians of the egg-laying females, the 95% confidence interval nearly excludes zero. The posterior means of the intercept and the year 2018 are positive and negative, respectively, as is the case with the parameter estimates in the third column of Table 2.8.

Concerning the correlation parameter, only the effect $\beta_{\theta 7}$ for sludge thickness is significantly different from zero. The posterior mean is estimated to equal 17.44, so at the minimum, median, and maximum observed sludge thickness, the respective predicted values for θ are 0.75, 0.99, and 1.00. The values are valid under the assumption that the other fixed effects on θ are zero. Moreover, the correlation ρ_i for the ith observation satisfies

$$\rho_i = \theta_i \sqrt{\frac{\mu_i \nu_i}{(\mu_i + 1)(\nu_i + 1)}}, \tag{2.4}$$

where μ_i and ν_i are the marginal means (before zero-inflation) that correspond to parameterization (1.21). Therefore, we can predict the correlation using the mean covariate values per year (see the left panel Table 2.11). Predicting M, N, p, q, and r using the parameters' posterior means in Table 2.12, using the expressions (1.23) to compute μ, ν, and evaluating (2.4) yields the values in the right panel of Table 2.11. Keep in mind that we take the posterior means of β_{M7}, β_{N1}, and β_{N2} as zero because of their confidence interval. The predicted correlation becomes quite small in 2018, due to ν being very close to zero. The correlation's variability is much more pronounced in 2015–2017, where the counts are estimated to be significantly less correlated when the sludge thickness is minimal.

The zero-inflations p, q, and r are all estimated to approximately equal 0.0004. Considering the small number of observations, we may safely treat them as zero. Note this was also the case in the univariate response setting, cf. Tables 2.8 and 2.9. The respective posterior means of the random intercept variances for the exuviae and egg-laying female counts are 3.23 and 1.20. These estimates also resemble the results in Section 2.4, where the variance was also higher for exuviae counts.

Regarding the random intercepts, only the areas GL1, GL2, GV2, and SBBF3 have a significant effect on the median exuviae counts, while only SBBF2 and WHA4 significantly affect

the median egg-laying female counts. Median exuviae counts are estimated to be much higher in GL1 and SBBF3 (respectively 20 and 7.5 times larger than the mean across areas) and much lower in GL2 and GV2 (respectively 81 and 7.8 times smaller). In SBBF2 and WHA4, the median egg-laying female counts are 4.9 and 6.4 times larger, respectively.

	2015	2016	2017	2018
Emersion (fraction)	0.73	0.74	0.81	0.81
Oxygen (fraction)	0.52	0.76	0.57	0.39
EC (mS/cm)	0.38	0.52	0.59	0.62
Predicted μ	4.46	2.89	1.49	1.37
Predicted ν	2.12	2.12	2.12	0.00

	2015	2016	2017	2018
Minimal	0.56	0.53	0.48	0.0090
Median	0.74	0.71	0.63	0.0119
Maximal	0.74	0.71	0.64	0.0120

Table 2.11: Mean covariate values relevant to (2.3) per year along with predicted means (left) and predicted correlations for minimal, median, and maximal sludge thickness in each year (right).

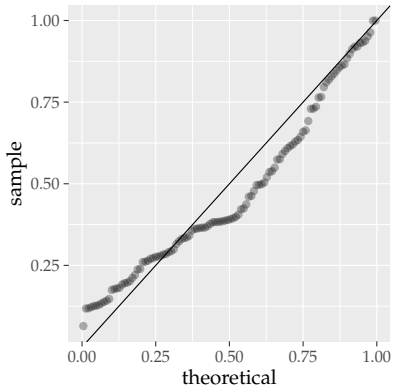


Figure 2.7: QQ-plot of the scaled residuals for the bivariate count model against the Unif$([0, 1])$ distribution.

We conclude this section by assessing the model's validity using randomized quantile residuals computed from the posterior predictive distribution for each observation. Instead of computing this distribution directly, we approximate it by generating 10 of points for each set of parameters in the MCMC sample. The points are generated using the mechanism in Listing B.1 with a thinning factor of 500, totaling 8000 generated points per observation. Each scaled residuals is obtained from the corresponding (two-dimensional) empirical cumulative distribution function at the observed value. Section 1.2.4 argues that the residuals should be approximately uniformly distributed on the unit interval if the model is a good fit. Figure 2.7 presents a quantile-quantile plot that shows the distribution is not compellingly uniform, as confirmed by a Kolmogorov-Smirnov test with a p-value of 0.0432. As the p-value is, at the same time, relatively close to the conventional cut-off value 0.05, it is also not a convincing indication of a bad model fit.

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
β_{M0}	2.85	4.94	7.54	4.94	1.176	0.054	477	-0.097	1.022
β_{M1}	-3.78	-2.10	-0.25	-2.10	0.916	0.036	662	-0.021	1.002
β_{M2}	-6.57	-3.62	-0.81	-3.70	1.519	0.057	711	-0.068	1.011
β_{M3}	-13.67	-5.95	-2.36	-6.92	3.292	0.127	667	-0.084	1.007
β_{M4}	-0.15	1.94	4.06	1.94	1.076	0.041	674	-0.023	1.006
β_{M5}	-5.03	-3.14	-1.44	-3.17	0.926	0.035	714	-0.084	1.026
β_{M6}	-10.14	-5.98	-1.97	-6.01	2.047	0.088	546	-0.088	1.006
β_{M7}	-4.76	-1.52	1.79	-1.56	1.712	0.060	820	-0.002	1.010
β_{M8}	1.23	4.92	9.29	4.96	2.023	0.081	630	-0.024	1.005
β_{M9}	-0.25	4.86	10.08	4.89	2.633	0.105	629	-0.076	1.014
β_{M10}	0.53	7.43	19.42	9.05	5.491	0.216	649	-0.094	1.005
β_{N0}	-0.02	0.81	1.71	0.79	0.446	0.016	744	-0.007	1.002
β_{N1}	-0.26	0.58	1.32	0.57	0.410	0.012	1099	-0.057	1.004
β_{N2}	-1.58	-0.66	0.22	-0.68	0.460	0.015	883	0.008	1.004
β_{N3}	-2.85	-1.66	-0.63	-1.71	0.831	0.029	833	-0.032	1.048
$\beta_{\theta^2 0}$	-16.79	-6.88	3.09	-6.61	5.067	0.372	186	-0.013	1.058
$\beta_{\theta^2 1}$	-3.88	3.25	13.53	3.71	4.391	0.219	400	-0.009	1.012
$\beta_{\theta^2 2}$	-14.01	-0.09	16.18	0.57	7.714	0.280	757	0.009	1.006
$\beta_{\theta^2 3}$	-21.97	-5.13	8.71	-5.52	7.846	0.277	800	-0.027	1.000
$\beta_{\theta^2 4}$	-4.29	6.42	18.15	6.17	5.851	0.383	234	0.010	1.045
$\beta_{\theta^2 5}$	-24.67	-9.80	3.91	-9.79	7.464	0.429	302	0.022	1.026
$\beta_{\theta^2 6}$	-11.71	1.02	14.54	0.95	6.594	0.328	405	0.000	1.009
$\beta_{\theta^2 7}$	4.29	17.73	33.55	17.44	7.660	0.359	456	-0.045	1.043
$\beta_{\theta^2 8}$	-17.02	-2.04	10.58	-2.44	7.202	0.297	589	0.003	1.003
$\beta_{\theta^2 9}$	-11.43	4.19	20.96	4.32	8.461	0.288	861	0.029	1.009
$\beta_{\theta^2 10}$	-23.48	-6.88	10.25	-7.01	8.551	0.310	762	-0.051	0.998
β_p	-20.19	-7.68	-0.83	-8.79	5.840	0.230	646	0.035	1.009
β_q	-21.50	-7.96	-1.14	-9.28	5.894	0.208	800	-0.025	1.000
β_r	-20.76	-7.83	-0.21	-9.05	5.952	0.199	892	0.015	1.000
σ_M^{-2}	0.05	0.26	0.66	0.31	0.183	0.008	556	-0.020	1.003
σ_N^{-2}	0.07	0.69	1.88	0.83	0.569	0.022	681	0.009	1.003
deviance	3905.22	3942.58	3972.61	3943.20	17.679	0.875	409	0.019	1.009

Table 2.12: MCMC simulation results for the model 2.3.

	Lower95	Median	Upper95	Mean	SD	MCerr	SSeff	AC	psrf
b_{M1}	1.42	3.03	4.66	3.03	0.867	0.036	586	0.003	1.007
b_{M2}	-7.16	-4.40	-1.42	-4.43	1.471	0.055	724	-0.020	1.004
b_{M3}	-1.80	-0.23	1.29	-0.25	0.799	0.029	774	-0.010	1.003
b_{M4}	-3.08	-1.42	0.15	-1.43	0.835	0.030	763	-0.022	1.008
b_{M5}	-4.85	-0.79	2.11	-0.97	1.713	0.060	802	-0.010	1.001
b_{M6}	-3.87	-2.05	-0.17	-2.05	0.943	0.040	552	0.049	0.998
b_{M7}	-2.63	-1.15	0.32	-1.15	0.783	0.029	718	-0.029	1.005
b_{M8}	-1.95	-0.67	0.82	-0.62	0.703	0.026	759	-0.004	1.001
b_{M9}	-2.43	-0.52	1.09	-0.55	0.906	0.029	952	0.034	1.003
b_{M10}	0.68	2.02	3.38	2.01	0.684	0.024	799	-0.036	1.008
b_{M11}	-0.20	1.68	4.08	1.76	1.093	0.039	766	-0.016	1.017
b_{M12}	-1.46	-0.28	1.32	-0.24	0.730	0.028	660	0.037	0.998
b_{M13}	-1.23	0.06	1.53	0.07	0.713	0.025	797	-0.037	1.001
b_{M14}	-0.85	0.68	2.80	0.69	0.929	0.033	800	0.017	1.000
b_{M15}	-0.75	0.85	2.42	0.90	0.839	0.030	806	0.013	1.001
b_{M16}	-0.02	1.57	3.10	1.64	0.794	0.029	769	-0.050	1.000
b_{M17}	-0.13	1.94	4.21	1.93	1.137	0.044	669	-0.020	1.003
b_{N1}	-0.44	0.87	2.34	0.91	0.733	0.030	592	-0.008	1.001
b_{N2}	-1.14	0.22	1.41	0.22	0.650	0.023	797	-0.019	0.998
b_{N3}	-1.09	0.01	1.25	0.01	0.586	0.020	876	-0.013	1.007
b_{N4}	-2.98	-1.12	0.46	-1.22	0.926	0.032	836	0.027	1.000
b_{N5}	-2.48	-0.24	1.89	-0.31	1.146	0.042	746	0.003	0.998
b_{N6}	-3.06	-1.24	0.25	-1.31	0.881	0.032	743	0.041	1.006
b_{N7}	-1.97	-0.25	1.32	-0.31	0.860	0.030	800	-0.039	1.011
b_{N8}	-0.57	0.47	1.43	0.46	0.527	0.018	870	-0.014	1.003
b_{N9}	-1.36	-0.19	1.02	-0.20	0.631	0.021	931	-0.056	1.003
b_{N10}	0.52	1.60	2.56	1.59	0.530	0.019	769	-0.009	1.010
b_{N11}	-0.20	1.21	2.67	1.23	0.742	0.024	980	0.004	1.003
b_{N12}	-2.35	-0.77	0.60	-0.83	0.768	0.028	734	0.050	1.015
b_{N13}	-0.98	0.11	1.26	0.15	0.598	0.019	967	0.024	1.003
b_{N14}	-3.02	-0.87	1.16	-0.96	1.065	0.036	874	0.013	1.002
b_{N15}	0.81	1.82	2.96	1.86	0.564	0.019	856	0.009	1.005
b_{N16}	-3.16	-1.35	0.61	-1.42	0.972	0.035	785	0.008	1.006
b_{N17}	-1.47	-0.16	1.15	-0.17	0.654	0.023	783	-0.050	1.005
σ_M^{-2}	0.05	0.26	0.66	0.31	0.183	0.008	556	-0.020	1.003
σ_N^{-2}	0.07	0.69	1.88	0.83	0.569	0.022	681	0.009	1.003
deviance	3905.22	3942.58	3972.61	3943.20	17.679	0.875	409	0.019	1.009

Table 2.13: MCMC simulation results for the random effects of the model 2.3.

2.6 Discussion

Certain aspects of the analysis are deserving of some explanatory comments. First of all, no interaction effects on adult counts are uncovered, even though there might be some. The reason is that we applied a simple but crude heuristic. Even though doing so may not lead to the best fit, the uncovered model is less prone to overfitting; we accept some bias pertaining to the 'true' model in return for a reduction of variance. It is important to remark that we modeled numbers of egg-laying females counts instead of adults together with exuviae. The justification is that no geometric distribution model found by our backward selection heuristic fits the adult counts, and the applied bivariate-response model does not support underlying distributions that vary per component. Despite this limitation, it is not unreasonable to think that egg-laying female counts can be used to accurately measure the green hawker populations. Additionally, as mentioned before, egg-laying females are more strongly tied to the location they were tallied than adult specimens are in general.

Many effects' estimated posterior distributions have relatively wide 95% confidence intervals compared to the magnitude of their mean. To gain details on the actual size of these effects, using more informative prior distributions might be useful. Alternatively, the confidence intervals could be narrowed using more data, given they are obtainable. Moreover, only a positive correlation was considered in (2.3) because the counts under scrutiny were hypothesized to express similarity. In terms of modeling, however, it would have been entirely valid to allow a negative correlation as well. Nevertheless, the positive estimate for the effect of sludge thickness on the correlation parameter and the fact that there are no other statistically significant estimates imply that negative correlation would not likely have been predicted if it would have been permitted.

A compromise has been made in the model definition in Listing B.7, as this version may lead to inaccuracy due to precision issues. The justification is the large thinning factor needed by the MCMC algorithm to reach convergence, which is likely caused by the relatively small data set compared to the number of model parameters. It would have taken an excessive amount of time to generate sufficient iterations if the likelihoods would have been computed using the `ZIBGeometric` module. This solution is tolerable due to marginal estimates that are similar to the univariate response regression results. Nonetheless, it would be interesting to see whether the findings persist if the `ZIBGeometric` module could be optimized.

Finally, it is important to be mindful of the nearly unacceptable fit of the bivariate response model to the data. There are a few possible reasons why this occurs. First of all, we used univariate model selection to find suitable sets of covariates. Although the utilized MCMC algorithms are not nearly fast enough to carry out backward elimination, a suitable marginal likelihood maximization could be utilizable. In such a manner, we might find a parsimonious model that fits the data better. A second explanation would be that the approximations of the scaled residuals are insufficient, either because we did not generate enough points per observation or because the thinning factor in the data generating mechanisms was too small. Without optimization, however, increasing either factor causes the residual approximation to take much longer. Therefore, we will settle for the narrowly permissible fit while keeping in mind the potential improvement strategies.

2.7 Concluding remarks

The main contributions of this thesis are (1) the definition and simulation study of a Bayesian bivariate-response GLMM that is robust under extreme outliers (treated in Chapter 1), (2) the modeling of the data on green hawker populations from Milder-Mulderij et al. (2019), and (3) the quantification of the correlation between green hawker exuviae and egg-laying female counts with respect to the covariates. While the latter points have a scope that is limited to the ecological study of the dragonfly populations, the first may have implications in a broader statistical modeling context.

Both conventional GLMM univariate response regressions and Bayesian inference on the newly proposed bivariate response model have been conducted. The conventional regressions' primary purpose was, besides providing readily generated and easily interpretable comparison, to select an adequate set of regressors for the new model. Although comparisons between the analysis approaches were not the central focus, we have seen that many estimated effects in the conventional regressions have similar posterior means and significances when considering a median-modeled bivariate response variant instead. Since the inclusion of prior knowledge is a principal distinction between frequentist and Bayesian inference, we can safely make such comparisons because of the uninformative prior distributions imposed on the model parameters.

We have observed that the new management strategy is not associated with a statistically significant increase or decrease in green hawker populations. It is, however, related to a decrease of 16.5% in the average sludge amounts. Therefore, one would expect that fewer water soldiers could be removed using an adjusted checkered pattern. Since the vegetation contains larvae, doing so should be beneficial to the green hawkers. It would be essential to test whether the number of removed water soldiers could be significantly smaller in a new study, especially because vegetation positively relates to sludge.

The results indicate that exuviae counts vary with the water's electrical conductivity and the oxygen level aside from the anticipated relationship with the host plant emersion. In particular, these two abiotic factors are inversely related. On the other hand, no covariates in the data set apart from the year or the area seem to influence the egg-laying female counts. Their correlation appears to be influenced by the sludge thickness, where higher values are associated with increased similarity. At the same time, this covariate's effect on the exuviae counts is not statistically significant. As a result, the correlation is predicted to be roughly 25% less at a minimal compared to a median or higher sludge thickness.

It should be noted that the conducted analysis only investigates a small fraction of the possible aspects of the data at hand. Ecological processes can by no means be considered deterministic while maintaining a manageable level of complexity. We did not investigate the interesting relationships that may exist between some of the variables that were taken as covariates. Additionally, further research could focus its attention on modeling the data as unevenly spaced time series. If the locations' coordinates are retrievable, another possibility would be to take distances and similarities between ditches into account.

It would be fascinating to see whether defining and applying a bivariate discrete Weibull distribution would provide a significant improvement. As noted before, doing so allows additionally controlling for overdispersion, a feature that is lost by not being able to use a bivariate negative binomial distribution if the median is to be modeled directly. Moreover, applying backward elimination on univariate discrete Weibull distributions may reveal a different set of relevant covariates. If achievable, the analogous bivariate-response model would be applicable in a much more general setting by supporting a majority of distributions appearing in count data.

Bibliography

Akaike, Hirotugu (1973). Information theory and an extension of the maximum likelihood principle,[w:] proceedings of the 2nd international symposium on information, bn petrow, f. *Czaki, Akademiai Kiado, Budapest*.

Bellman, Richard and Robert Roth (1969). Curve fitting by segmented straight lines. *Journal of the American Statistical Association 64*(327), 1079–1084.

Benke, Arthur C. and Susan S. Benke (1975). Comparative dynamics and life histories of coexisting dragonfly populations. *Ecology 56*(2), 302–317.

Billingsley, Patrick (2012). Probability and measure (anniversary ed.).

Bolker, Benjamin (2015). Linear and generalized linear mixed models. In *Ecological statistics: contemporary theory and application*. Oxford University Press, USA.

Bolker, B (2017). Glmm faq: Inference and confidence intervals.

Bolker, Benjamin M, Mollie E Brooks, Connie J Clark, Shane W Geange, John R Poulsen, M Henry H Stevens, and Jada-Simone S White (2009). Generalized linear mixed models: a practical guide for ecology and evolution. *Trends in ecology & evolution 24*(3), 127–135.

Brooks, Mollie E., Kasper Kristensen, Maria Rosa Darrigo, Paulo Rubim, Maria Uriarte, Emilio Bruna, and Benjamin M. Bolker (2019). Statistical modeling of patterns in annual reproductive rates. *Ecology 100*(7), e02706.

Burger, Divan Aristo, Robert Schall, Johannes Theodorus Ferreira, and Ding-Geng Chen (2020). A robust bayesian mixed effects approach for zero inflated and highly skewed longitudinal count data emanating from the zero inflated discrete weibull distribution. *Statistics in Medicine*.

Chauvet, Jocelyn, Catherine Trottier, and Xavier Bry (2019). Component-based regularization of multivariate generalized linear mixed models. *Journal of Computational and Graphical Statistics 28*(4), 909–920.

Das, Kalyan, Mohamad Elmasri, and Arusharka Sen (2016). A skew-normal copula-driven glmm. *Statistica Neerlandica 70*(4), 396–413.

Denwood, Matthew J et al. (2016). runjags: An r package providing interface utilities, model templates, parallel computing methods and additional distributions for mcmc models in jags. *Journal of Statistical Software 71*(9), 1–25.

Dobson, Annette J and Adrian G Barnett (2018). *An introduction to generalized linear models* (4 ed.). Chapman and Hall/CRC.

Dunn, Peter K and Gordon K Smyth (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics 5*(3), 236–244.

Dworkin, Ian and Ben Bolker (2019). *Multivariate modeling via mixed models*.

Ellacott, Stephen W, John C Mason, and Iain J Anderson (2012). *Mathematics of neural networks: models, algorithms and applications*, Volume 8. Springer Science & Business Media.

Faes, Christel, Marc Aerts, Helena Geys, Luc Bijnens, Luc Ver Donck, and Wim JEP Lammers (2006). Glmm approach to study the spatial and temporal evolution of spikes in the small intestine. *Statistical Modelling 6*(4), 300–320.

Famoye, Felix (2019). Bivariate exponentiated-exponential geometric regression model. *Statistica Neerlandica 73*(3), 434–450.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani (2001). *The elements of statistical learning*, Volume 1. Springer series in statistics New York.

Gelman, Andrew, Donald B Rubin, et al. (1992). Inference from iterative simulation using multiple sequences. *Statistical science 7*(4), 457–472.

Gueorguieva, Ralitza (2001). A multivariate generalized linear mixed model for joint modelling of clustered outcomes in the exponential family. *Statistical Modelling 1*(3), 177–193.

Hadfield, Jarrod D et al. (2010). Mcmc methods for multi-response generalized linear mixed models: the mcmcglmm r package. *Journal of Statistical Software 33*(2), 1–22.

Hardersen, Sönke, Serena Corezzola, Gabriele Gheza, Alessandro Dell'Otto, and Gianandrea La Porta (2017). Sampling and comparing odonate assemblages by means of exuviae: statistical and methodological aspects. *Journal of Insect Conservation 21*(2), 207–218.

Hartig, Florian (2020). *DHARMa: Residual Diagnostics for Hierarchical (Multi-Level / Mixed) Regression Models*. R package version 0.3.1.

Jaffa, Miran A, Mulugeta Gebregziabher, Deirdre K Luttrell, Louis M Luttrell, and Ayad A Jaffa (2016). Multivariate generalized linear mixed models with random intercepts to analyze cardiovascular risk markers in type-1 diabetic patients. *Journal of applied statistics 43*(8), 1447–1464.

Jayakumar, K and Davis Antony Mundassery (2007). On bivariate geometric distribution. *Statistica 67*(4), 389–404.

Konishi, Sadanori and Genshiro Kitagawa (2008). *Information criteria and statistical modeling*. Springer Science & Business Media.

Kristensen, Kasper, Anders Nielsen, Casper W Berg, Hans Skaug, and Brad Bell (2015). Tmb: automatic differentiation and laplace approximation. *arXiv preprint arXiv:1509.00660*.

Kruger, Ariel and Peter J Morin (2020). Predators induce morphological changes in tadpoles of hyla andersonii. *Copeia 108*(2), 316–325.

Kullback, Solomon and Richard A Leibler (1951). On information and sufficiency. *The annals of mathematical statistics 22*(1), 79–86.

Li, Chin-Shang, Jye-Chyi Lu, Jinho Park, Kyungmoo Kim, Paul A Brinkley, and John P Peterson (1999). Multivariate zero-inflated poisson models and their applications. *Technometrics 41*(1), 29–38.

Lunn, David, Chris Jackson, Nicky Best, Andrew Thomas, and David Spiegelhalter (2012). *The BUGS book: A practical introduction to Bayesian analysis*. CRC press.

Marshall, Albert W and Ingram Olkin (1985). A family of bivariate distributions generated by the bivariate bernoulli distribution. *Journal of the American Statistical Association 80*(390), 332–338.

McCullagh, Peter (2002). What is a statistical model? *Annals of statistics*, 1225–1267.

McZgee, Victor E and Willard T Carleton (1970). Piecewise regression. *Journal of the American Statistical Association 65*(331), 1109–1124.

Milder-Mulderij, G, C Brochard, R Wiggers, and S de Vries (2019). Alternatief krabbenscheerbeheer in fryslân, groningen en drenthe. interpretatie op basis van vier jaar onderzoek op diverse locaties. Technical report.

Nakanishi, Kosuke, Hiroyuki Yokomizo, and Takehiko I. Hayashi (2018). Were the sharp declines of dragonfly populations in the 1990s in japan caused by fipronil and imidacloprid? an analysis of hill's causality for the case of sympetrum frequens. *Environmental Science and Pollution Research 25*(35), 35352–35364.

Nikoloulopoulos, Aristidis K (2015). A mixed effect model for bivariate meta-analysis of diagnostic test accuracy studies using a copula representation of the random effects distribution. *Statistics in medicine 34*(29), 3842–3865.

Norris, James R (1998). *Markov chains*. Number 2. Cambridge university press.

van Oppen, Y (2020, oct). JAGS ZIBGeometric module. `https://sourceforge.net/projects/jags-zibg/`.

Plummer, Martyn (2017). Jags version 4.3.0 user manual. *See `https://sourceforgenet/projects/mcmc-jags/files/Manuals/4x`*.

Press, William H, Saul A Teukolsky, William T Vetterling, and Brian P Flannery (1988). Numerical recipes in c.

Raudenbush, Stephen W, Meng-Li Yang, and Matheos Yosef (2000). Maximum likelihood for generalized linear models with nested random effects via high-order, multivariate laplace approximation. *Journal of computational and Graphical Statistics 9*(1), 141–157.

Reimchen, Thomas E and Carolyn A Bergstrom (2009). The ecology of asymmetry in stickleback defense structures. *Evolution: International Journal of Organic Evolution 63*(1), 115–126.

Rice, John and Murray Rosenblatt (1983). Smoothing splines: regression, derivatives and deconvolution. *The annals of Statistics*, 141–156.

Rivaz, Firoozeh and Majid Jafari Khaledi (2017). Likelihood inference in a multivariate spatial glmm with skew gaussian random effects using the slice-sab algorithm. *Journal of Statistical Theory and Applications 16*(1), 108–116.

Sagias, Nikos C and George K Karagiannidis (2005). Gaussian class multivariate weibull distributions: theory and applications in fading channels. *IEEE Transactions on Information Theory 51*(10), 3608–3619.

Sherrat, T, C Hassall, R Laird, D Thompson, and A Cordero-Rivera (2011). A comparative analysis of senescence in adult damselflies and dragonflies (odonata). *Journal of Evolutionary Biology 24*(4), 810–822.

Søndergaard, Martin (2009). Redox potential. In *Encyclopedia of Inland Waters*, pp. 852–859. Pergamon Press.

Spiegelhalter, David J, Nicola G Best, Bradley P Carlin, and Angelika Van Der Linde (2002). Bayesian measures of model complexity and fit. *Journal of the royal statistical society: Series b (statistical methodology) 64*(4), 583–639.

Termaat, Tim, Arco J. van Strien, Roy H. A. van Grunsven, Geert De Knijf, Ulf Bjelke, Klaus Burbach, Klaus-Jürgen Conze, Philippe Goffart, David Hepper, Vincent J. Kalkman, Grégory Motte, Marijn D. Prins, Florent Prunier, David Sparrow, Gregory G. van den Top, Cédric Vanappelghem, Michael Winterholler, and Michiel F. WallisDeVries (2019). Distribution trends of european dragonflies under climate change. *Diversity and Distributions 25*(6), 936–950.

Tuerlinckx, Francis, Frank Rijmen, Geert Verbeke, and Paul De Boeck (2006). Statistical inference in generalized linear mixed models: A review. *British Journal of Mathematical and Statistical Psychology 59*(2), 225–255.

Van Buskirk, Josh (1993). Population consequences of larval crowding in the dragonfly aeshna juncea. *Ecology 74*(7), 1950–1958.

Wabersich, Dominik and Joachim Vandekerckhove (2014). Extending jags: A tutorial on adding custom distributions to jags (with a diffusion model example). *Behavior research methods 46*(1), 15–28.

Wang, Xia, Ming-Hui Chen, Rita C Kuo, and Dipak K Dey (2015). Bayesian spatial-temporal modeling of ecological zero-inflated count data. *Statistica Sinica 25*(1), 189.

Wasserman, Larry (2006). *All of nonparametric statistics*. Springer Science & Business Media.

Zhang, Weiping, MengMeng Zhang, and Yu Chen (2019). A copula-based glmm model for multivariate longitudinal data with mixed-types of responses. *Sankhya B*, 1–27.

Zhang, Xinyan, Himel Mallick, Zaixiang Tang, Lei Zhang, Xiangqin Cui, Andrew K Benson, and Nengjun Yi (2017). Negative binomial mixed models for analyzing microbiome count data. *BMC bioinformatics 18*(1), 4.

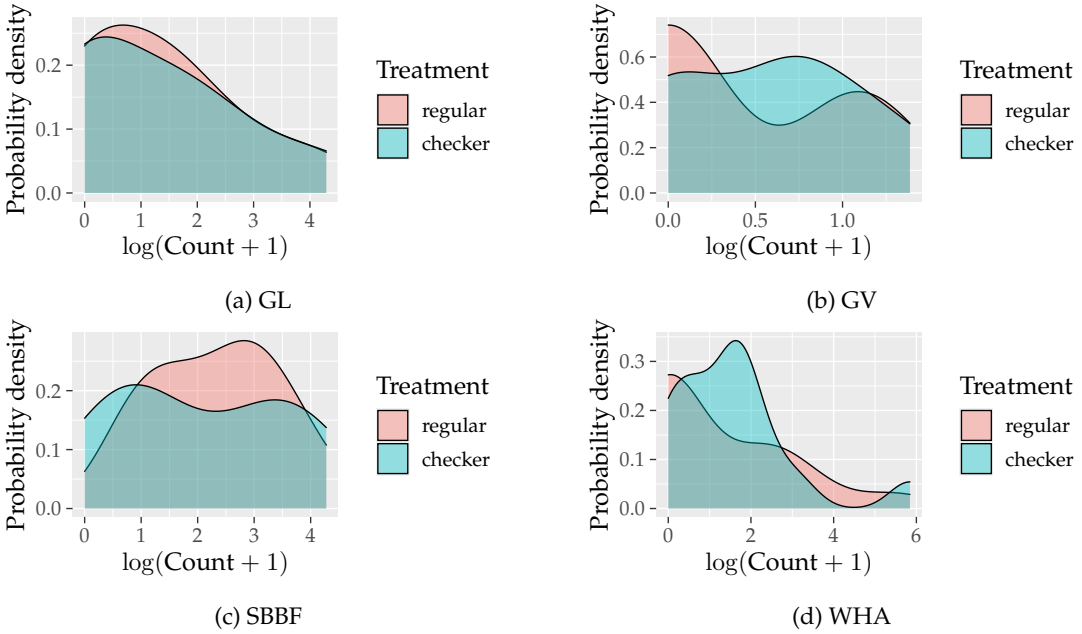
Zuur, Alain, Elena N Ieno, Neil Walker, Anatoly A Saveliev, and Graham M Smith (2009). *Mixed effects models and extensions in ecology with R*. Springer Science & Business Media.

Appendix A

Additional tables and figures

A.1 Manager- and year-specific empirical densities

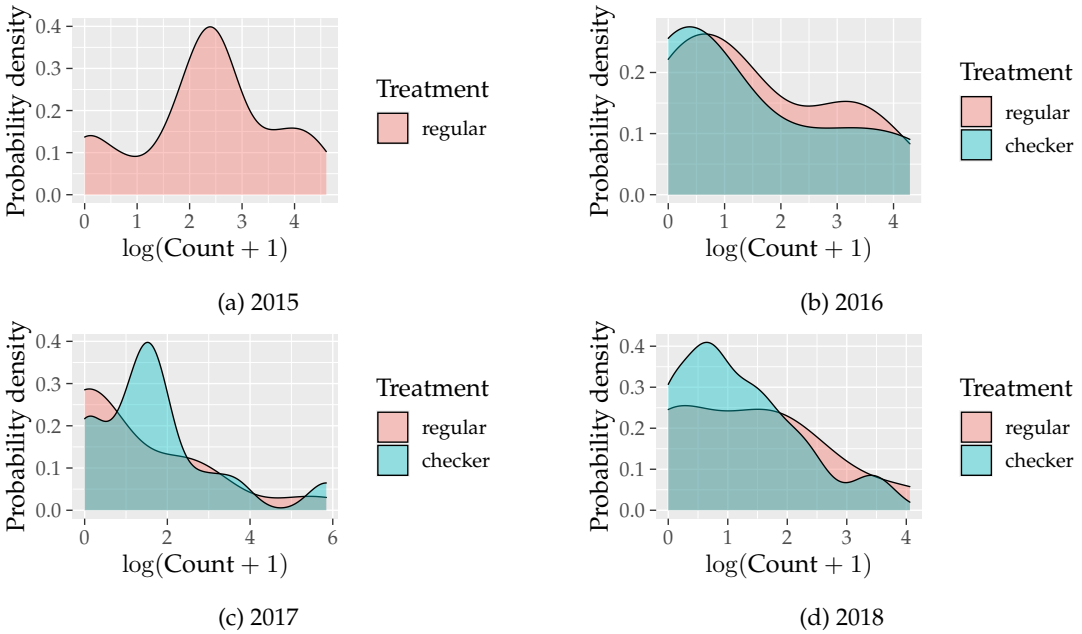
A.1.1 Exuviae



(a) GL (b) GV

(c) SBBF (d) WHA

Figure A.1: Manager-specific empirical densities of exuviae counts for both treatments in 2016 – 2018. SBBG is not included, as there is only data from 2015 for this manager.

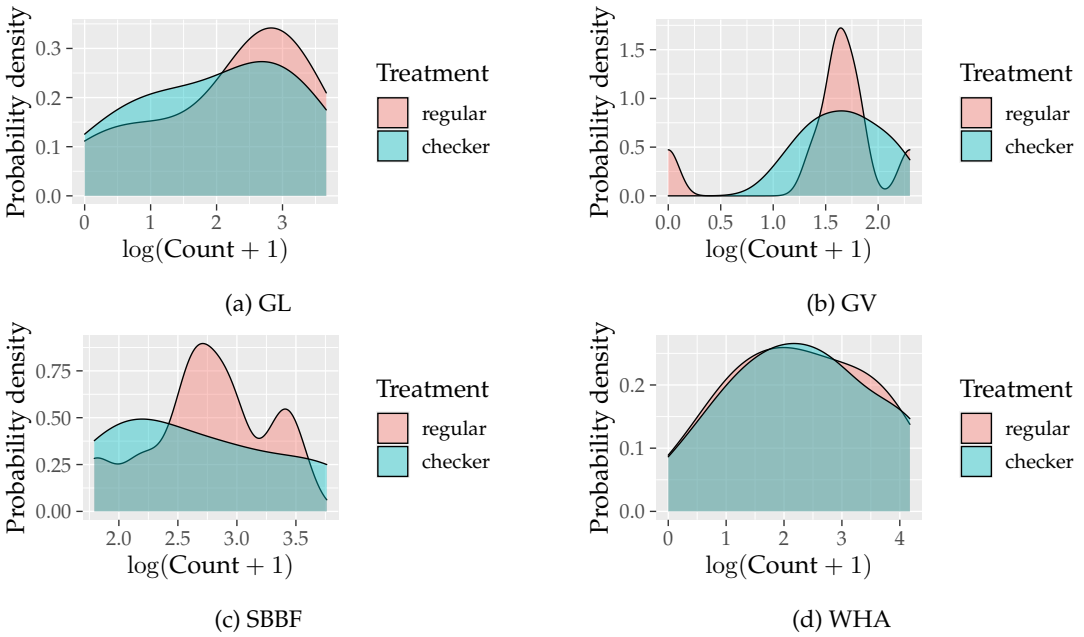


(a) 2015 (b) 2016

(c) 2017 (d) 2018

Figure A.2: Year-specific empirical densities of exuviae counts for both treatments.

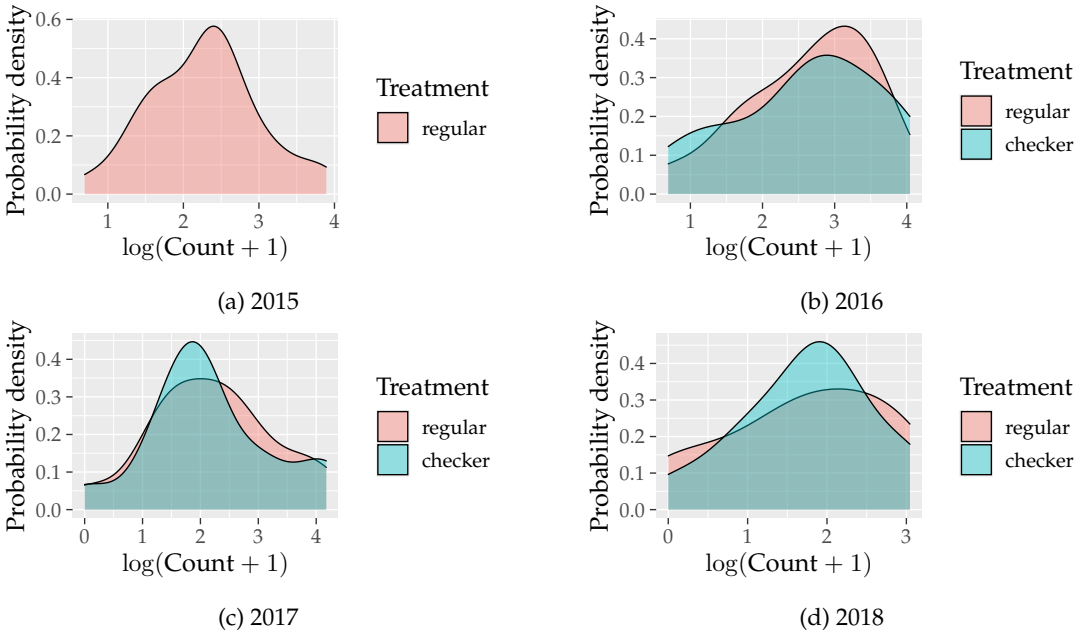
A.1.2 Adults



(a) GL (b) GV

(c) SBBF (d) WHA

Figure A.3: Manager-specific empirical densities of adult counts for both treatments in 2016 – 2018. SBBG is not included, as there is only data from 2015 for this manager.

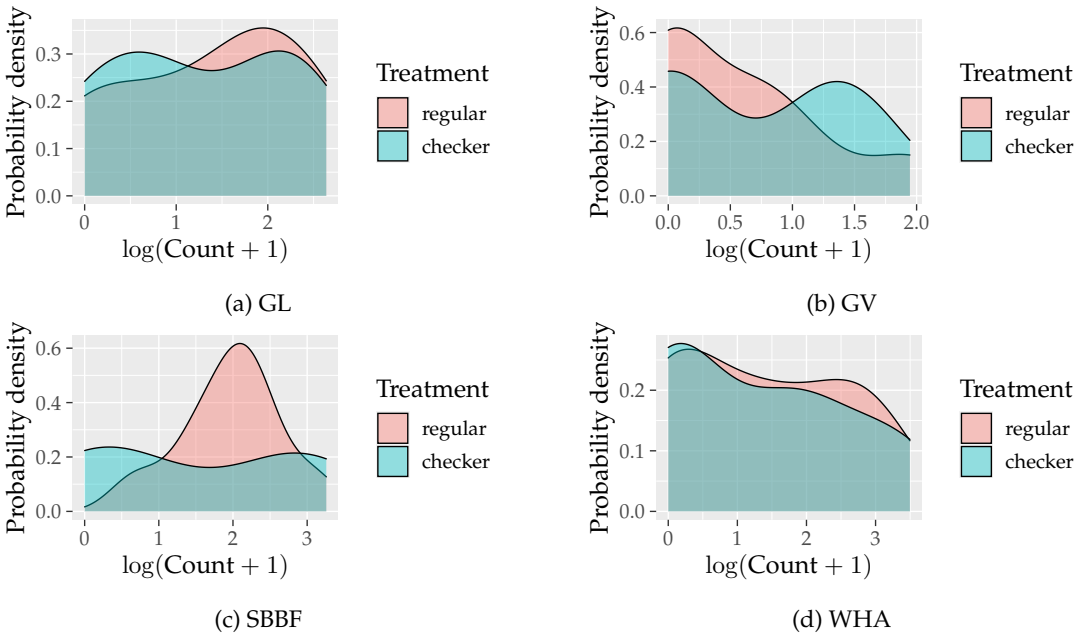


(a) 2015 (b) 2016

(c) 2017 (d) 2018

Figure A.4: Year-specific empirical densities of adult counts for both treatments.

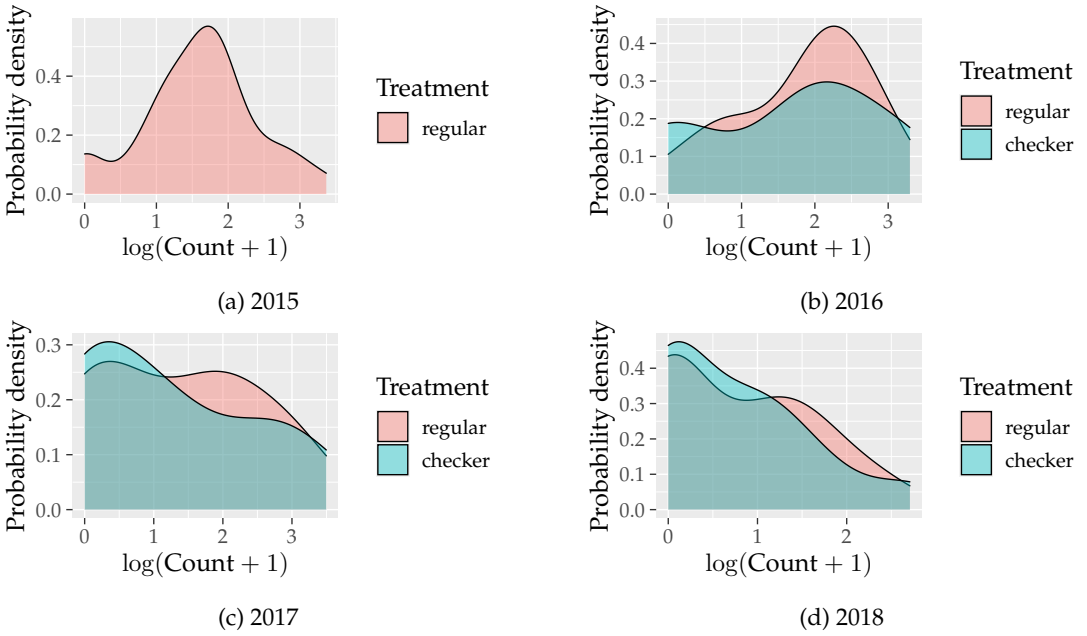
A.1.3 Egg-laying females



(a) GL (b) GV

(c) SBBF (d) WHA

Figure A.5: Manager-specific empirical densities of egg-laying female counts for both treatments in 2016 – 2018. SBBG is not included, as there is only data from 2015 for this manager.



(a) 2015 (b) 2016

(c) 2017 (d) 2018

Figure A.6: Year-specific empirical densities of egg-laying female counts for both treatments.

A.2 Spearman's rank correlation coefficients

A.2.1 Coefficients

	Treatment	Year	A. vir. (ex.)	A. vir. (ad.)	A. vir. (eggl.)
Treatment	1.000				
Year	0.377	1.000			
A. vir. (ex.)	-0.115	-0.283	1.000		
A. vir. (ad.)	-0.034	-0.204	0.465	1.000	
A. vir. (eggl.)	-0.117	-0.314	0.465	0.911	1.000
Emersion	0.038	0.108	-0.084	-0.191	-0.146
pH	0.148	0.216	-0.014	0.189	0.068
Redox	-0.013	-0.307	0.100	0.219	0.218
Oxygen	0.010	-0.158	-0.206	0.038	0.028
EC	0.204	0.434	-0.087	0.118	0.079
Temperature	-0.103	0.086	0.023	-0.253	-0.212
Water depth	0.023	0.030	0.090	-0.146	-0.119
Sludge depth	-0.123	-0.035	0.201	0.026	-0.018
Sludge thickness	-0.192	0.008	0.168	0.155	0.073

	Emersion	pH	Redox	Oxygen	EC	Temperature
Emersion	1.000					
pH	-0.275	1.000				
Redox	-0.173	0.080	1.000			
Oxygen	-0.097	0.413	0.306	1.000		
EC	-0.024	0.419	-0.326	0.135	1.000	
Temperature	0.122	-0.183	-0.261	-0.064	0.139	1.000
Water depth	0.284	-0.078	-0.043	-0.246	-0.111	-0.032
Sludge depth	0.183	0.060	0.049	-0.096	0.024	0.052
Sludge thickness	-0.041	0.127	0.035	0.050	0.163	0.154

	Water depth	Sludge depth	Sludge thickness
Water depth	1.000		
Sludge depth	0.727	1.000	
Sludge thickness	-0.061	0.588	1.000

Table A.1: Spearman's rank correlation coefficients for relevant variables. Coefficients that are significant at the 0.05 level are marked red.

A.2.2 P-values

	Treatment	Year	A. vir. (ex.)	A. vir. (ad.)	A. vir. (eggl.)
Treatment	-				
Year	2.45e-03	-			
A. vir. (ex.)	4.91e-04	1.23e-03	-		
A. vir. (ad.)	1.64e-01	8.12e-03	1.07e-02	-	
A. vir. (eggl.)	1.22e-01	6.07e-03	8.70e-03	4.59e-09	-
Emersion	6.81e-01	1.14e-01	3.42e-01	2.51e-04	1.84e-03
pH	1.35e-01	3.03e-01	1.40e-01	2.88e-01	4.09e-01
Redox	1.31e-01	2.17e-04	8.46e-02	2.92e-03	9.98e-03
Oxygen	9.35e-01	6.59e-01	5.43e-01	1.26e-01	1.35e-01
EC	1.10e-01	4.10e-03	3.20e-02	4.83e-01	4.74e-01
Temperature	7.94e-01	6.12e-02	3.92e-01	1.47e-02	4.49e-02
Water depth	8.64e-01	4.09e-01	8.99e-01	4.08e-02	3.53e-02
Sludge depth	4.92e-02	5.43e-01	4.92e-02	9.35e-01	8.76e-01
Sludge thickness	6.54e-03	3.42e-01	4.28e-02	1.35e-01	1.02e-01

	Emersion	pH	Redox	Oxygen	EC	Temperature
Emersion	-					
pH	2.09e-02	-				
Redox	4.10e-03	6.70e-01	-			
Oxygen	4.82e-03	7.04e-03	1.02e-01	-		
EC	7.14e-01	5.63e-03	7.22e-02	2.40e-01	-	
Temperature	8.70e-03	3.42e-01	1.11e-03	2.53e-01	2.33e-01	-
Water depth	1.37e-03	1.14e-02	1.22e-01	2.51e-04	1.64e-01	2.40e-01
Sludge depth	2.53e-01	4.70e-02	8.29e-01	1.69e-01	8.14e-02	2.88e-01
Sludge thickness	3.92e-01	7.48e-01	7.03e-01	5.13e-01	6.70e-01	2.74e-01

	Water depth	Sludge depth	Sludge thickness
Water depth	-		
Sludge depth	2.89e-02	-	
Sludge thickness	4.36e-01	1.26e-01	-

Table A.2: P-values for Spearman's rank correlation coefficients in Table A.1; values below 0.05 are marked red.

A.3 MCMC trace plots using empirical data

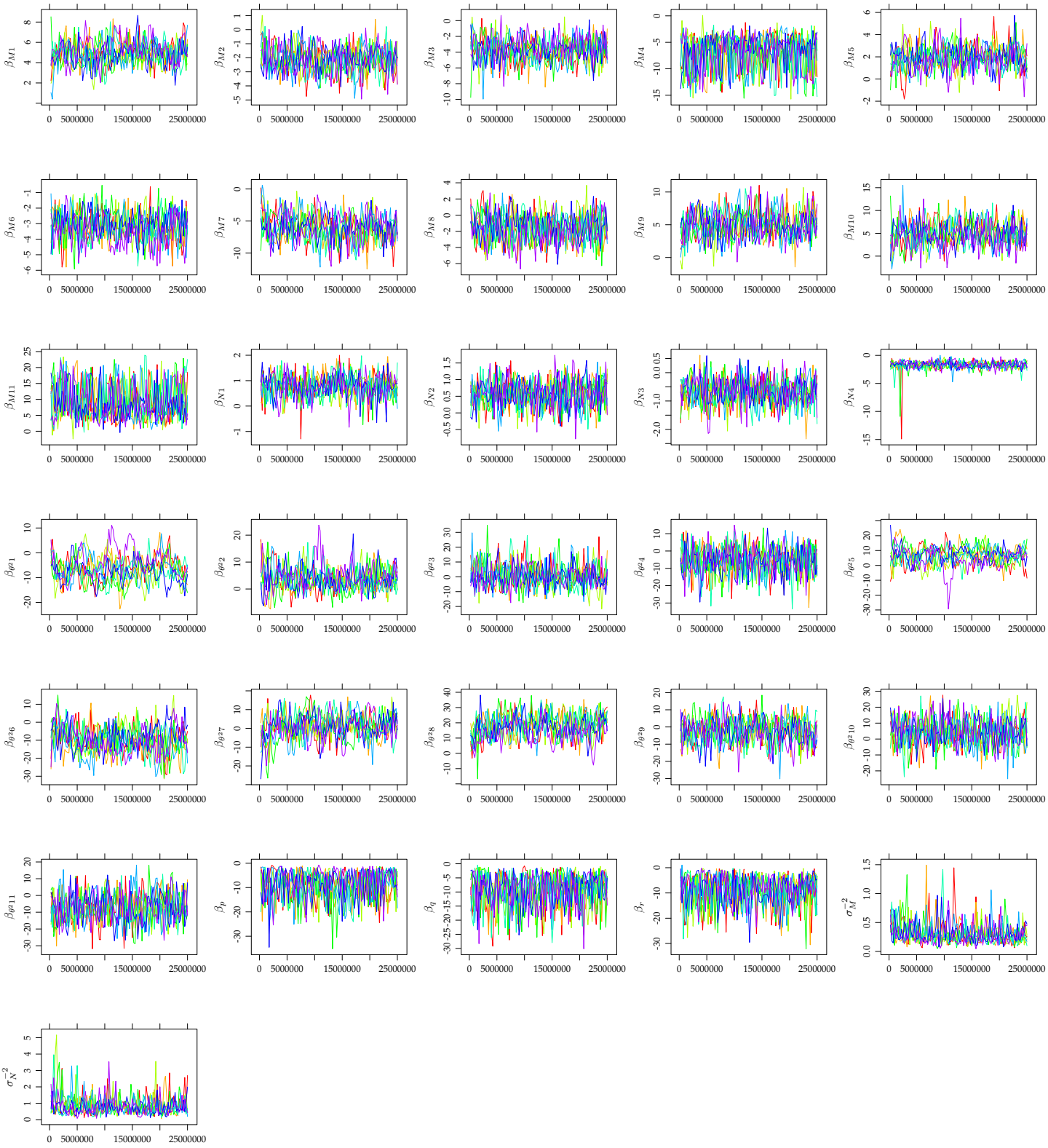


Figure A.7: Trace plots of the fixed effect parameters in (2.3) (iteration number on the horizontal axis).

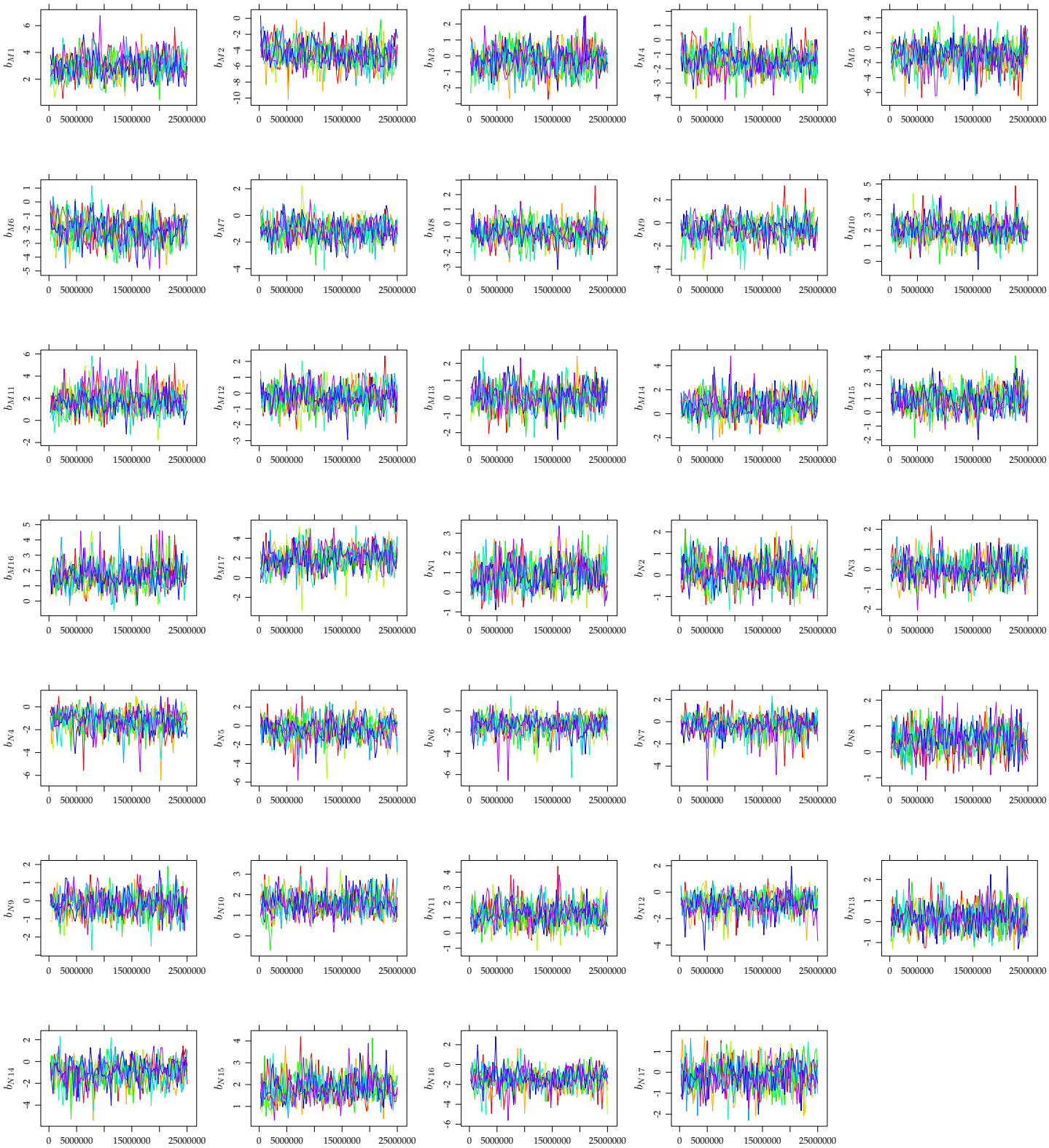


Figure A.8: Trace plots of the random intercepts in (2.3) (iteration number on the horizontal axis).

Appendix B

Source code

B.1 Data generation

Listing B.1: A Metropolis-Hastings MCMC algorithm to generate two-dimensional count data.

```
library(Rcpp)

rm(list = ls())
sourceCpp("dzibg.cpp")
load("dzibeeg.rda")

neighbors <- function(current, number = T) {     # return the (number of) neighbors of current
    if (number)
        return(8 - 3 * any(!current) - 2 * all(!current))
    else {
        nbors <- rbind(current + c(0, 1), current + c(1, 0), current + c(1, 1))
        if (current[1])
            nbors <- rbind(nbors, rbind(current + c(-1, 0), current + c(-1, 1)))
        if (current[2])
            nbors <- rbind(nbors, rbind(current + c(0, -1), current + c(1, -1)))
        if (all(as.logical(current)))
            nbors <- rbind(nbors, current + c(-1, -1))
        return(nbors)
    }
}

Q <- function(from, to) {                        # proposal distribution going Q(to | from)
    if (any(abs(from - to) > 1) | any(to < 0))
        return(0)
    else
        return(1 / neighbors(from))
}

propose <- function(current) {                   # sample neighbor from the proposal distribution
    nbors <- neighbors(current, number = F)
    return(nbors[sample(1:nrow(nbors), 1), ])
}

A <- function(from, to, pars, pmass)             # acceptance probability A(to | from)
    return( min(1, pmass(to, pars) * Q(to, from) / (pmass(from, pars) * Q(from, to))) )

                                                 # generate sample using MCMC algorithm
metrosample <- function(n, pars, pmass, thin = 1, burnin = 0) {
    points <- matrix(0, n, 2)
    temp   <- matrix(0, thin, 2)

    for (point in 1:n) {                         # loop over n * thin iterations
        for (iter in 1:thin) {
            current <- temp[ifelse(iter == 1, thin, iter-1), ]
            proposal <- propose(current)         # propose new point

                                                 # accept with probability A(proposal | current)
            if (A(current, proposal, pars, pmass) > runif(1))
                temp[iter, ] <- proposal
            else
                temp[iter, ] <- current
        }
        points[point, ] <- temp[thin, ]          # store every thin iterations
    }

    return(points)
}
                                                 # generate ZIBG data
metroZIBG <- function(n, pars, thin = 1, burnin = 0)
    metrosample(n, pars, dzibg, thin, burnin)
```

Listing B.2: A C++ function to compute the PMF of a ZIBG distribution.

```cpp
#include <Rcpp.h>                                       // connect with R

#include <cmath>                                        // isfinite
#include <limits>                                       // most negative double
#include <boost/math/special_functions/binomial.hpp>    // binomial coefficient
#include <boost/multiprecision/cpp_bin_float.hpp>       // high precision floats

namespace mp = boost::multiprecision;                   // abbreviate for clarity
typedef mp::number<mp::cpp_bin_float<100> > myfloat;

myfloat multinom(int a, int b, int c) {                 // multinomial coefficient
    return boost::math::binomial_coefficient<myfloat>(a + b + c, b + c) *
           boost::math::binomial_coefficient<myfloat>(b + c, c);
}

// [[Rcpp::depends(BH)]]                                // library dependencies
// [[Rcpp::export]]                                     // make available in R
double dzibg(Rcpp::IntegerVector x, Rcpp::NumericVector par, bool log = false) {
    double y = x[0], z = x[1];
    double mu = par[0], nu = par[1], th = par[2], p = par[3], q = par[4], r = par[5];

    myfloat pmf = 0;                                    // bivariate geometric pmf
    for (int j = 0; j <= std::min(y, z); ++j) {
        pmf += pow(-1, j) * multinom(y - j, z - j, j)
                          * mp::pow(myfloat((1 - th) * mu * nu), j)
                          * mp::pow(myfloat(mu + (1 - th) * mu * nu), y - j)
                          * mp::pow(myfloat(nu + (1 - th) * mu * nu), z - j)
                          / mp::pow(myfloat(1 + mu + nu + (1 - th) * mu * nu), y + z - j + 1);
    }
                                                        // add zero-inflation
    pmf = (1 - p - q - r) * pmf + p * (y == 0 && z == 0)
                                + q * (z == 0) * mp::pow(myfloat(mu / (mu + 1)), y) / (mu + 1)
                                + r * (y == 0) * mp::pow(myfloat(nu / (nu + 1)), z) / (nu + 1);

    if (log)                                            // return (log) pmf
        return isfinite(mp::log(pmf).convert_to<double>()) ? mp::log(pmf).convert_to<double>()
                                                           : std::numeric_limits<double>::lowest();
    else
        return pmf.convert_to<double>();
}
```

Listing B.3: A function to compute the PMF of a ZIBEEG distribution.

```r
dzibeeg <- function(x, pars, log = FALSE) { # compute the PMF of the ZIBEEG(pars) distribution
    th <- pars[1:2]
    b  <- pars[3:4]
    la <- pars[5]
    p  <- pars[6]

    c1 <- c2 <- 0                             # approximate E{exp(-Yt)} for t = 1, 2
    for (r in 0:10) {
        c1 <- c1 + prod(b[1]:(b[1]-r+1)) * (-1)^r / exp(lfactorial(r)) * (th[1]^r - 1) /
                                           (1 - th[1]^r * exp(-1))
        c2 <- c2 + prod(b[2]:(b[2]-r+1)) * (-1)^r / exp(lfactorial(r)) * (th[2]^r - 1) /
                                           (1 - th[2]^r * exp(-1))
    }

                                           # compute the PMF
    pmf <-  prod( ((1 - th^(x+1))^b - (1 - th^x)^b) ) *
            (1 + la * (exp(-x[1]) - c1) * (exp(-x[2]) - c2))
    pmf <- p * (sum(x) == 0) + (1 - p) * pmf

    return( ifelse(log, log(pmf), pmf) )
}

save(dzibeeg, file = "dzibeeg.rda")
```

B.2 JAGS R scripts

JAGS (Just Another Gibbs Sampler, see Plummer (2017)) is a program to conduct Bayesian analysis of hierarchical models using MCMC simulation. It uses a dialect of the BUGS language (Bayesian inference Using Gibbs Sampling, see Lunn et al. (2012)), and is designed to work closely with R. Its useful features include compatibility with all major operating systems and extensibility that allows its users to write custom functions, distributions, and samplers.

A hierarchical model is specified in a `model {...}` block. Inside this block, the `<-` operator indicates a *deterministic* relation, while `~` represents a *stochastic* relation. Vectors, matrices, and arrays are indexed as in R. Many functions that resemble their counterparts in R are available, and most of them are automatically vectorized. Functions that are also (inverse) link functions form an exception; this applies in particular to `log()` and `exp()`.

Data may be supplied in an additional `data {...}` block preceding the model definition. The data will be taken into account to estimate the posterior parameter distribution using Gibss sampling if the full conditional likelihood can be computed. Otherwise, JAGS will resort to a Metropolis-Hastings MCMC algorithm. The JAGS user manual (Plummer, 2017) provides a brief yet comprehensive introduction to effectively using the program; the interested reader is redirected.

Listing B.4: An R script to estimate the model (1.27) using JAGS.

```
library(coda)                           # detailed MCMC output
library(parallel)                       # multithreading
library(runjags)                        # JAGS in R

load("resynthdata.rda")
DATA <- list(Y = y, len = dim(y)[1], Z = rep(1:6, each = dim(y)[1] / 6))

MODEL <- '
model {
    for (i in 1:len) {
        Y[i, 1:2] ~ dzibg(MU[Z[i]], NU, TH, P, Q, R)
    }

    for (j in 1:6) {
        MU[j] <- 1 / (pow(2 * (1 - P - R), 1 / (exp(BETA_M + B[j]) + 1)) - 1)
        B[j]   ~ dnorm(0, ISIG2)
    }
    ISIG2 ~ dgamma(1/2, 1)

    NU <- 1 / (pow(2 * (1 - P - Q), 1 / (exp(BETA_N) + 1)) - 1)
    TH <- 2 * ilogit(BETA_TH) - 1
    P  <- exp(BETA_P) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2
    Q  <- exp(BETA_Q) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2
    R  <- exp(BETA_R) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2

    BETA_M ~ dnorm(0, 0.1)
    BETA_N ~ dnorm(0, 0.1)

    BETA_TH ~ dnorm(0, 0.5)
    BETA_P  ~ dnorm(0, 0.5)
    BETA_Q  ~ dnorm(0, 0.5)
    BETA_R  ~ dnorm(0, 0.5)
}'

PARAMS <- c('BETA_M', 'BETA_N', 'BETA_TH', 'BETA_P', 'BETA_Q', 'BETA_R', 'B', 'ISIG2')

INITIAL <- replicate(8, list(list(.RNG.name = 'base::Mersenne-Twister',
                                  .RNG.seed = sample.int(n = 100000, size = 1))))

SAMPLE <- run.jags(model = MODEL, inits = INITIAL, data = DATA, n.chains = 8,
                   monitor = PARAMS, adapt = 100, burnin = 500, thin = 50, sample = 100,
                   method = 'parallel', modules = c('ZIBGeometric', 'glm'))
```

Listing B.5: An R script to estimate the constant ZIBG$(M, N, \theta, p, 0, 0)$ model using JAGS.

```
library(coda)                           # detailed MCMC output
library(parallel)                       # multithreading
library(runjags)                        # JAGS in R

load("zibgdata.rda")
# load("zibeegdata.rda")
DATA <- list(Y = y, LEN = dim(y)[1])

MODEL <- '
model {
    for (i in 1:LEN) {
        Y[i, 1:2] ~ dzibg(MU, NU, TH, P, 0, 0)
    }

    MU <- 1 / (pow(2 * (1 - P), 1 / (exp(BETA_M) + 1)) - 1)
    NU <- 1 / (pow(2 * (1 - P), 1 / (exp(BETA_N) + 1)) - 1)
    TH <- 2 * ilogit(BETA_TH) - 1
    P  <- ilogit(BETA_P) / 2

    BETA_M ~ dnorm(0, 0.1)
    BETA_N ~ dnorm(0, 0.1)

    BETA_TH ~ dnorm(0, 0.5)
    BETA_P  ~ dnorm(0, 0.5)
}'

PARAMS <- c('BETA_M', 'BETA_N', 'BETA_TH', 'P')

INITIAL <- replicate(8, list(list(.RNG.name = 'base::Mersenne-Twister',
                                  .RNG.seed = sample.int(n = 100000, size = 1))))

SAMPLE <- run.jags(model = model, inits = INITIAL, data = DATA, n.chains = 8,
                   monitor = PARAMS, adapt = 100, burnin = 50, thin = 5, sample = 100,
                   method = 'parallel', modules = c('ZIBGeometric', 'glm'))
```

Listing B.6: An R script to estimate the constant ZIBEEG$(\theta_1, \theta_2, b_1, b_2, \lambda, p)$ model using JAGS.

```
library(coda)                           # detailed MCMC output
library(parallel)                       # multithreading
library(runjags)                        # JAGS in R

# load("zibgdata.rda")
load("zibeegdata.rda")
DATA <- list(Y = y, LEN = dim(y)[1])

MODEL <- '
model {
    for (i in 1:LEN) {
        Y[i, 1:2] ~ dzibeeg(TH[1], TH[2], B[1], B[2], LA, P)
    }

    logit(TH[1]) <- BETA_TH1
    logit(TH[2]) <- BETA_TH2
    log(B[1])    <- BETA_B1
    log(B[2])    <- BETA_B2
    P            <- ilogit(BETA_P) / 2

    BETA_TH1 ~ dnorm(0, 0.5)
    BETA_TH2 ~ dnorm(0, 0.5)
    BETA_B1  ~ dnorm(0, 0.1)
    BETA_B2  ~ dnorm(0, 0.1)
    BETA_LA  ~ dnorm(0, 0.1)
    BETA_P   ~ dnorm(0, 0.5)
}'

PARAMS <- c('BETA_TH1', 'BETA_TH2', 'BETA_B1', 'BETA_B2', 'BETA_LA', 'BETA_P')

INITIAL <- replicate(8, list(list(.RNG.name = 'base::Mersenne-Twister',
                                  .RNG.seed = sample.int(n = 100000, size = 1))))
```

```
SAMPLE <- run.jags(model = MODEL, inits = INITIAL, data = DATA, n.chains = 8,
                   monitor = PARAMS, adapt = 100, burnin = 50, thin = 5, sample = 100,
                   method = 'parallel', modules = c('ZIBGeometric', 'glm'))
```

Listing B.7: An R script to estimate the model (2.3) using JAGS. The sparse array `MNOM` contains the needed pre-calculated multinomial coefficients for computational efficiency.

```
library(coda)                           # detailed MCMC output
library(parallel)                       # multithreading
library(runjags)                        # JAGS in R

load("df.rda")
load("mnom.rda")
DATA <- with(totals, list(Y = cbind(A_vir_ex, A_vir_ad_f_eggl),
                          Z = as.numeric(area),
                          X = cbind(rep(1, nrow(totals)),
                                    as.numeric(year == "2016"),
                                    as.numeric(year == "2017"),
                                    as.numeric(year == "2018"),
                                    emers_frac,
                                    O2_frac,
                                    EC_mS_cm,
                                    sludge_thickness_m,
                                    as.numeric(year == "2016") * EC_mS_cm,
                                    as.numeric(year == "2017") * EC_mS_cm,
                                    as.numeric(year == "2018") * EC_mS_cm),
                          PRIORMU  = rep(0, 11),
                          PRIORISIG = diag(rep(0.1, 11)),
                          MNOM = mnom))

MODEL <- '
data {
    for (i in 1:114) {
        ONES[i] <- 1
    }
}
model {
    for (i in 1:114) {
        LIKE[i]  <- min(1, max(1e-308, equals(Y[i, 1], 0) * equals(Y[i, 2], 0) * P
                                       + equals(Y[i, 2], 0) * Q * MARG1[i]
                                       + equals(Y[i, 1], 0) * R * MARG2[i]
                                       +         (1 - P - Q - R) * PMF[i]))
        ONES[i]  ~  dbern(LIKE[i])

        MARG1[i] <- pow(MU[i], Y[i, 1]) / pow(MU[i] + 1, Y[i, 1] + 1)
        MARG2[i] <- pow(NU[i], Y[i, 2]) / pow(NU[i] + 1, Y[i, 2] + 1)

        PMF[i] <- sum((pow(-1, 0:min(Y[i, 1], Y[i, 2]))
                       * MNOM[Y[i, 1] + 1, Y[i, 2] + 1, 0:min(Y[i, 1], Y[i, 2]) + 1]
                       * pow((1 - TH[i]) * MU[i] * NU[i],
                             0:min(Y[i, 1], Y[i, 2]))
                       * pow(MU[i] + (1 - TH[i]) * MU[i] * NU[i],
                             Y[i, 1] - 0:min(Y[i, 1], Y[i, 2]))
                       * pow(NU[i] + (1 - TH[i]) * MU[i] * NU[i],
                             Y[i, 2] - 0:min(Y[i, 1], Y[i, 2]))
                       * pow(1 + MU[i] + NU[i] + (1 - TH[i]) * MU[i] * NU[i],
                             -(Y[i, 1] + Y[i, 2] - 0:min(Y[i, 1], Y[i, 2]) + 1))))

        MU[i]     <- pow(pow(2 * (1 - P - R), 1 / (M[i] + 1)) - 1, -1)
        NU[i]     <- pow(pow(2 * (1 - P - Q), 1 / (N[i] + 1)) - 1, -1)

        log(M[i])     <- inprod(BETA_M, X[i, ])    + B_M[Z[i]]
        log(N[i])     <- inprod(BETA_N, X[i, 1:4]) + B_N[Z[i]]
        logit(TH[i]) <- ilogit(inprod(BETA_TH, X[i, ]))
    }

    for (i in 1:17) {
        B_M[i] ~ dnorm(0, ISIGMA2_M)
        B_N[i] ~ dnorm(0, ISIGMA2_N)
```

```
    }

    P <- exp(BETA_P) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2
    Q <- exp(BETA_Q) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2
    R <- exp(BETA_R) / (1 + exp(BETA_P) + exp(BETA_Q) + exp(BETA_R)) / 2

    BETA_M  ~ dmnorm(PRIORMU,      PRIORISIG)
    BETA_N  ~ dmnorm(PRIORMU[1:4], PRIORISIG[1:4, 1:4])
    BETA_TH ~ dmnorm(PRIORMU,      PRIORISIG)
    BETA_P  ~ dnorm(PRIORMU[1],    PRIORISIG[1, 1])
    BETA_Q  ~ dnorm(PRIORMU[1],    PRIORISIG[1, 1])
    BETA_R  ~ dnorm(PRIORMU[1],    PRIORISIG[1, 1])

    OM        ~ dgamma(0.5, 1E-2)
    ISIGMA2_M ~ dgamma(1,   1/(8*OM))
    ISIGMA2_N ~ dgamma(1,   1/(8*OM))
}'


PARAMS <- c('BETA_M', 'BETA_N', 'BETA_TH', 'BETA_P', 'BETA_Q', 'BETA_R',
            'B_M', 'B_N', 'ISIGMA2_M', 'ISIGMA2_N')

INITIAL <- replicate(8, list(list(.RNG.name = 'base::Mersenne-Twister',
                                  .RNG.seed = sample.int(n = 100000, size = 1))))

SAMPLE <- run.jags(model = MODEL, data = DATA, inits = INITIAL, monitor = PARAMS,
                   n.chains = 8, burnin = 250000, thin = 250000, sample = 100,
                   method = 'parallel', modules = c('ZIBGeometric', 'glm'),
                   factories = 'bugs::MNormal sampler off')
```

B.3 ZIBGeometric JAGS module

The modular and extensible character of JAGS makes it relatively easy to write custom modules, provided the user has a basic understanding of class inheritance in C++. A great tutorial is given in Wabersich and Vandekerckhove (2014), which we will use to write a module `ZIBGeometric` that implements both a ZIBG and a ZIBEEG distribution class. In our working directory, we insert the following file structure.

```
./
├── m4
├── src
│   ├── distributions
│   │   ├── DZIBG.h
│   │   ├── DZIBG.cc
│   │   ├── DBEEG.h
│   │   ├── DBEEG.cc
│   │   └── Makefile.am
│   ├── ZIBGeometric.cc
│   └── Makefile.am
├── configure.ac
└── Makefile.am
```

The contents of the file `src/ZIBGeometric.cc` can be found in Listing B.8. It defines a module namespace `ZIBGeometric` within the `jags` namespace. A module class `ZIBGModule`, which publicly inherits from the `Module` class, is declared and implemented within the `jags` namespace. Its constructor `ZIBGModule()` instantiates a `DZIBG` and a `DBEEG` distribution

class object, representing the ZIBG and the ZIBEEG distributions, respectively. Its destructor `~ZIBGModule()` removes all distribution objects, and the final line initializes a `ZIBGModule` object.

The `DZIBG` distribution class is declared in the header file `src/distributions/DZIBG.h`, the contents of which can be found in Listing B.9. As we inherit from the abstract base class `VectorDist` in JAGS, we need to include the relevant header file in line 3. The base class contains the pure virtual member functions (for a description of all arguments, see the JAGS header file `distribution/VectorDist.h`)

- `logDensity()`: evaluate the log probability (mass) at `x` (of dimension `length`) with the supplied `parameters`;
- `randomSample()`: draw a random value from the distribution and store it in `x`;
- `typicalValue()`: set `x` to a typical value, often the mean, median, or mode, to initialize nodes with stochastic relations to the distribution;
- `support()`: store the `lower` and `upper` bounds for the distribution;
- `isSupportFixed()`: indicate whether the support depends on the parameters;
- `checkParameterLength()`: check whether the parameter array sizes are correct;
- `checkParameterValue()`: check whether the parameters lie in the parameter space;
- `length()`: returns the dimension of the distribution depending on the `par`ameters.

All of the above member functions need to be implemented in order for a `DZIBG` object to be instantiable. Their definitions can be found in the file `src/distribution/DZIBG.cc`, shown in Listing B.10 and B.11. In our implementation, we use the `boost::multiprecision` libraries to prevent accuracy issues due to underflow or crude rounding in the calculation of the log probability mass. Moreover, we use the implementation of the binomial coefficient from the `boost::math` libraries to prevent overflow during its computation.

The function `logDensity()` returns the log probability mass given in Proposition 1. Note the type `myfloat`, a floating-point number type with a precision of 50 decimal places, is used throughout the calculation. Binary operators automatically return a result of type `myfloat` when at least one of its arguments is of type `myfloat`. The power function `mp::pow` is the `boost::multiprecision` analog of `pow`. The return statement ensures the value of approximately 2.22e-308 is returned instead of negative infinity (which would lead JAGS to prompt it cannot compute the log-density).

The current implementations of both `randomSample()` and `typicalValue()` fill the vector (pointed to by) `x` with zeros. For the former, it means sampling using JAGS is not yet intended, while it simply returns the mode for the latter. The function `support()` fills the vectors `lower` and `upper` with zeros and positive infinity, respectively, since the support is $\{0, 1, \ldots\}^2$. Hence, `isSupportFixed()` always returns true. The member function `checkParameterLength()` verifies that all parameters are scalar. On the other hand, `checkParameterValue` confirms that `mu` and `nu` are non-negative, `th` lies between -1 and 1, and `p`, `q`, and `r` lie within the standard 3-simplex. Finally, `length()` always returns two since the distribution is two-dimensional.

The `DBEEG` class is very similar to `DZIBG`, so we will only provide the definitions of `logDensity()` and `checkParameterValue()` in Listings B.12 and B.13. Aside from that, it suffices to replace `DZIBG` with `DBEEG` in every instance in `DZIBG.h` and `DZIBG.cc`.

The files `configure.ac`, `Makefile.am`, `src/Makefile.am`, and `src/distributions/Makefile.am` are very similar to those supplied by Wabersich and Vandekerckhove (2014). For this reason, we will not go into all workings in detail here. The file contents can be found in the respective Listings B.14, B.15, B.16, B.17. Note that the `dnl` directive indicates a comment line. It is crucial to make sure line 5 of `configure.ac` contains the correct version and contact information, and lines 6–7 reference the installed version of JAGS. Lines 14–15 contain all relevant source files; more may be added if necessary. Finally, line 41 differs from the corresponding file in Wabersich and Vandekerckhove (2014). Here, it contains the path to the modules folder in the JAGS installation directory.

Assuming GNU autotools and libtool are installed, the program may be built and installed on a Linux distribution using the following command lines from the working directory:

```
autoreconf -fvi
./configure
make
sudo make install
```

In case compilation errors occur due to missing JAGS header files, execute the lines

```
CPP_INCLUDE_PATH = /usr/include/JAGS
export CP_INCLUDE_PATH
```

before running `make`. The module is published as a `.tar.gz` file in van Oppen (2020).

Listing B.8: `src/ZIBGeometric.cc`

```
1  #include <module/Module.h>      // include JAGS module base class
2  #include <distributions/DZIBG.h>// include DZIBG distribution class
3  #include <distributions/DBEEG.h>// include DBEEG distribution class
4
5  namespace jags::ZIBGeometric {  // start defining the module namespace
6
7                                  // module class
8      class ZIBGModule : public Module {
9          public:
10             ZIBGModule();       // constructor
11             ~ZIBGModule();      // destructor
12     };
13
14                                 // constructor implementation
15     ZIBGModule::ZIBGModule() : Module("ZIBGeometric") {
16         insert(new DZIBG);      // inherited function to load objects into JAGS
17         insert(new DBEEG);
18     }
19
20     ZIBGModule::~ZIBGModule() { // destructor implementation
21         std::vector<Distribution*> const &dvec = distributions();
22         for (unsigned int i = 0; i < dvec.size(); ++i)
23             delete dvec[i];     // delete all instantiated distribution objects
24     }
25
26 }                                  // end namespace definition
27
28 jags::ZIBGeometric::ZIBGModule _ZIBGeometric_module;
```

Listing B.9: `src/distributions/DZIBG.h`

```
#ifndef DZIBG_H_
#define DZIBG_H_
#include <distribution/VectorDist.h>    // JAGS vector distribution base class

namespace jags::ZIBGeometric {
                                // vector distribution class
    class DZIBG : public VectorDist {
        public:
            DZIBG();            // constructor

                                // evaluate the log probability mass
            double logDensity(double const *x, unsigned int length, PDFType type,
                              std::vector<double const *> const &parameters,
                              std::vector<unsigned int> const &lengths,
                              double const *lbound, double const *ubound) const;

                                // draw random value - currently always (0, 0)
            void randomSample(double *x, unsigned int length,
                              std::vector<double const *> const &parameters,
                              std::vector<unsigned int> const &lengths,
                              double const *lbound, double const *ubound,
                              RNG *rng) const;

                                // set equal to the mode - always (0, 0)
            void typicalValue(double *x, unsigned int length,
                              std::vector<double const *> const &parameters,
                              std::vector<unsigned int> const &lengths,
                              double const *lbound, double const *ubound) const;

                                // set support for the distribution - (0, inf) x (0, inf)
            void support(double *lower, double *upper, unsigned int length,
                         std::vector<double const *> const &params,
                         std::vector<unsigned int> const &lengths) const;

                                // returns true
            bool isSupportFixed(std::vector<bool> const &fixmask) const;

                                // verify that parameters are scalar
            bool checkParameterLength(std::vector<unsigned int> const &parameters) const;

                                // verify positive means, correlation between -1 and 1, and
                                // probabilities that sum to 1
            bool checkParameterValue(std::vector<double const *> const & parameters,
                                     std::vector<unsigned int> const &lengths) const;

                                // values are two-dimensional
            unsigned int length (std::vector<unsigned int> const &par) const;
    };

}

#endif                          // DZIBG_H_
```

Listing B.10: `src/distributions/DZIBG.cc` (1/2)

```
#include <config.h>             // system configuration file, created by Autoconf
                                // and defined in configure.ac
#include "DZIBG.h"              // header file, containing class prototype
#include <rng/RNG.h>            // provides random functions
#include <util/nainf.h>         // provides na and inf functions etc.

#include <cmath>                // library for standard math operations
#include <algorithm>            // generic algorithms
#include <limits>               // represent the most negative double

#include <boost/math/special_functions/binomial.hpp>    // binomial coefficient
#include <boost/multiprecision/cpp_bin_float.hpp>       // high precision floats

using std::vector;              // vector is used in code
using std::min;                 // min is used in code
using std::max;                 // max is used in code

                                // high-precision (100 decimal places) floats
namespace mp = boost::multiprecision;
typedef mp::cpp_bin_float_50 myfloat;

namespace jags::ZIBGeometric {  // module namespace

                                // constructor taking as arguments (1) the distribution
                                //   node's name as used in BUGS code and (2) the number
                                //   of parameters for that node
    DZIBG::DZIBG() : VectorDist("dzibg", 6) {}


                                // evaluate the log probability mass
    double DZIBG::logDensity(double const *x, unsigned int length, PDFType type,
                             vector<double const *> const &parameters,
                             vector<unsigned int> const &lengths,
                             double const *lbound, double const *ubound) const {

        double y = x[0], z = x[1];  // rename for readability
        double mu = *parameters[0], nu = *parameters[1], th = *parameters[2],
                p = *parameters[3],  q = *parameters[4],  r = *parameters[5];

        myfloat pmf = 0;            // initialize probability mass

        for (int j = 0; j <= std::min(y, z); ++j) {
            pmf += pow(-1, j) * boost::math::binomial_coefficient<myfloat>(y + z - j, z)
                              * boost::math::binomial_coefficient<myfloat>(z, j)
                              * mp::pow(myfloat((1 - th) * mu * nu), j)
                              * mp::pow(myfloat(mu + (1 - th) * mu * nu), y - j)
                              * mp::pow(myfloat(nu + (1 - th) * mu * nu), z - j)
                              / mp::pow(myfloat(1 + mu + nu + (1 - th) * mu * nu), y + z - j + 1);
        }

        pmf = (1 - p - q - r) * pmf + p * (y == 0 && z == 0)
                                    + q * (z == 0) * mp::pow(myfloat(mu / (mu + 1)), y) / (mu + 1)
                                    + r * (y == 0) * mp::pow(myfloat(nu / (nu + 1)), z) / (nu + 1);

        return jags_finite(mp::log(pmf).convert_to<double>())
                    ? mp::log(pmf).convert_to<double>()
                    : std::numeric_limits<double>::lowest();
    }
```

Listing B.11: `src/distributions/DZIBG.cc` (2/2)

```
                                // draw random value - currently always (0, 0)
    void DZIBG::randomSample(double *x, unsigned int length,
                             vector<double const *> const &parameters,
                             vector<unsigned int> const &lengths,
                             double const *lbound, double const *ubound,
                             RNG *rng) const {
        std::fill_n(x, 2, 0);
    }

                                // set equal to the mode - always (0, 0)
    void DZIBG::typicalValue(double *x, unsigned int length,
                             vector<double const *> const &parameters,
                             vector<unsigned int> const &lengths,
                             double const *lbound, double const *ubound) const {
        std::fill_n(x, 2, 0);
    }

                                // set support for the distribution - (0, inf) x (0, inf)
    void DZIBG::support(double *lower, double *upper, unsigned int length,
                        vector<double const *> const &params,
                        vector<unsigned int> const &lengths) const {
        std::fill_n(lower, 2, 0);
        std::fill_n(upper, 2, JAGS_POSINF);
    }

                                // returns true
    bool DZIBG::isSupportFixed(vector<bool> const &fixmask) const {
        return true;
    }

                                // verify that all parameters are scalar
    bool DZIBG::checkParameterLength (vector<unsigned int> const &parameters) const {
        return std::all_of(parameters.begin(),
                           parameters.end(),
                           [](unsigned int len) {return len == 1;});
    }

                                // verify positive means, correlation between -1 and 1, and
                                // probabilities that sum to 1
    bool DZIBG::checkParameterValue(vector<double const *> const &parameters,
                                    vector<unsigned int> const &lengths) const {
                                // rename for readability
        double mu = *parameters[0], nu = *parameters[1], th = *parameters[2],
                p = *parameters[3],  q = *parameters[4],  r = *parameters[5];

        return (mu >= 0 && nu >= 0 && th >= -1 && th <= 1 && p >= 0 &&  p <= 1 &&
                 q >= 0 &&  q <= 1 &&  r >=  0 &&  r <= 1 && p + q + r <= 1);
    }

                                // values are two-dimensional
    unsigned int DZIBG::length(vector<unsigned int> const &par) const {
        return 2;
    }
}
```

Listing B.12: The `DBEEG::logDensity()` function definition.

```
double DBEEG::logDensity(double const *x, unsigned int length, PDFType type,
                         vector<double const *> const &parameters,
                         vector<unsigned int> const &lengths,
                         double const *lbound, double const *ubound) const {

    double y = x[0], z = x[1];  // rename for readability
    double th1 = *parameters[0], th2 = *parameters[1], b1 = *parameters[2],
           b2 = *parameters[3],  la = *parameters[4],  p = *parameters[5];

    double c1 = 0, c2 = 0;
    for (std::size_t r = 1; r <= 20; ++r) {
        c1 += boost::math::falling_factorial(b1, r) / boost::math::factorial<double>(r)
              * pow(-1, r) * (pow(th1, r) - 1) / (1 - pow(th1, r) * exp(-1));
        c2 += boost::math::falling_factorial(b2, r) / boost::math::factorial<double>(r)
              * pow(-1, r) * (pow(th2, r) - 1) / (1 - pow(th2, r) * exp(-1));
    }

    myfloat pmf = (1 + la * (exp(-y) - c1) * (exp(-z) - c2) * (y != 0 || z != 0))
                      * (mp::pow(1 - mp::pow(myfloat(th1), y+1), b1)
                      - mp::pow(1 - mp::pow(myfloat(th1), y), b1))
                      * (mp::pow(1 - mp::pow(myfloat(th2), z+1), b2)
                      - mp::pow(1 - mp::pow(myfloat(th2), z), b2));

    pmf = (1 - p) * pmf + p * (y == 0 && z == 0);

    return jags_finite(mp::log(pmf).convert_to<double>())
               ? mp::log(pmf).convert_to<double>()
               : std::numeric_limits<double>::lowest();
}
```

Listing B.13: The `DBEEG::checkParameterValue()` function definition.

```
bool DBEEG::checkParameterValue(vector<double const *> const &parameters,
                                vector<unsigned int> const &lengths) const {
                            // rename for readability
    double th1 = *parameters[0], th2 = *parameters[1], b1 = *parameters[2],
           b2 = *parameters[3],  la = *parameters[4],  p = *parameters[5];

    return (th1 >= 0 && th1 <= 1 && th2 >= 0 && th2 <= 1 &&
               b1 >= 0 &&  b2 >= 0 &&  la >= 0 &&   p >= 0 && p <= 1);
}
```

Listing B.14: `configure.ac`

```
dnl Process this file with Autoconf to produce a configure script/

AC_PREREQ([2.68])

AC_INIT([JAGS-ZIBG], [1.0], [yulanvanoppen@gmail.com], [JAGS-ZIBG-MODULE])
JAGS_MAJOR=4
JAGS_MINOR=3
AC_SUBST(JAGS_MAJOR)
AC_SUBST(JAGS_MINOR)

AC_CANONICAL_HOST
dnl The following lines check if the required files exist
dnl (more AC_CONFIG_SRCDIR directives may be needed - for all relevant files)
AC_CONFIG_SRCDIR([src/distributions/DZIBG.cc])
AC_CONFIG_SRCDIR([src/distributions/DBEEG.cc])
AC_CONFIG_MACRO_DIR([m4])
dnl The configure process creates a header file called config.h
AC_CONFIG_HEADERS([config.h])
AM_INIT_AUTOMAKE

dnl libtool and ltdl configureation
LT_PREREQ(2.2.6)
LT_CONFIG_LTDL_DIR([libltdl])
LT_INIT([dlopen disable-static win32-dll])
LTDL_INIT([recursive])

dnl Indicate C++
AC_PROG_CXX

dnl Optionally, reference the Rmath library
dnl AC_DEFINE(MATHLIB_STANDALONE, 1, [Define if you have standalone R math library])

case "${host_os}" in
    mingw*)
        win=true ;;
    *)
        win=false ;;
esac
AM_CONDITIONAL(WINDOWS, test x$win = xtrue)

jagsmoddir=/usr/lib/x86_64-linux-gnu/JAGS/modules-${JAGS_MAJOR}
AC_SUBST(jagsmoddir)

AC_CONFIG_FILES([
    Makefile
    libltdl/Makefile
    src/Makefile
    src/distributions/Makefile
])
AC_OUTPUT
```

Listing B.15: `Makefile.am`

```
ACLOCAL_AMFLAGS = -I m4
SUBDIRS = libltdl src
```

Listing B.16: `src/Makefile.am`

```
SUBDIRS = distributions

jagsmod_LTLIBRARIES = ZIBGeometric.la

ZIBGeometric_la_SOURCES = ZIBGeometric.cc

ZIBGeometric_la_CPPFLAGS = -I$(includedir)/JAGS

ZIBGeometric_la_LIBADD = distributions/ZIBGeometricdist.la
if WINDOWS
ZIBGeometric_la_LIBADD += -ljags-$(JAGS_MAJOR)
else
ZIBGeometric_la_LIBADD += -ljags
endif

ZIBGeometric_la_LDFLAGS = -module -avoid-version
if WINDOWS
ZIBGeometric_la_LDFLAGS += no-undefined
endif
```

Listing B.17: `src/distributions/Makefile.am`

```
noinst_LTLIBRARIES = ZIBGeometricdist.la

ZIBGeometricdist_la_CPPFLAGS = -I$(top_srcdir)/src -I$(includedir)/JAGS

ZIBGeometricdist_la_LDFLAGS = -no-undefined -module -avoid-version

ZIBGeometricdist_la_SOURCES = DBEEG.cc DZIBG.cc

noinst_HEADERS = DBEEG.h DZIBG.h
```

Appendix C

Report to the client

university of groningen / faculty of science and engineering

Statistische analyse voor

Alternatief krabbenscheerbeheer in Fryslân, Groningen en Drenthe

Interpretatie op basis van vier jaar onderzoek op diverse locaties[1]

Yulan van Oppen

1e begeleider: Marco Grzegorczyk

2e begeleider: Wim Krijnen

Augustus 2020

Department of Mathematics
Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence
Faculty of Science and Engineering
University of Groningen

Dit verslag is geschreven, als onderdeel van de masterthesis “Statistical Analysis of the Aeshna viridis (green hawker) populations in the northern Netherlands” onder supervisie van de bovengenoemde begeleiders, voor Gabi Milder-Mulderij, Rink Wiggers en Christophe Brochard van Bureau Biota.

1 G. Milder-Mulderij, C. Brochard, R. Wiggers & S. De Vries, 2018. Alternatief krabbenscheerbeheer in Fryslân, Groningen en Drenthe. Interpretatie op basis van vier jaar onderzoek op diverse locaties. Bureau Biota rapport 2020-0xx. In opdracht van Waterschap Hunze en Aa’s, Groninger Landschap, Gemeente Veendam & Staatsbosbeheer.

INHOUDSOPGAVE

1. Samenvatting

In deze samenvatting worden de resultaten en bevindingen uit Paragraaf 2 kort benoemd.

De metingen van abiotische metingen en de krabbenscheer bedekking zijn gekoppeld aan de getelde aantallen libellen met de dichtstbijzijnde datum (op hetzelfde transect). Om een beeld te geven van de structuur zijn de eerste zes rijen van de verkregen dataset getoond in Tabel 1. Sommige eigenschappen van de factoren zijn te zien in Tabel 1 tot en met Tabel 4. Zo is er bijvoorbeeld in 43 van de totale hoeveelheid metingen (114) het ritsbeheer toegepast, en de mediaan van de aantallen volwassen groene glazenmakers is 8.

Er kan meer inzicht in de verdeling van de getelde aantallen libellen en de slibdikte verkregen worden met behulp van Figuur 1. Een punt op de horizontale as (let hier op de eventuele logaritmische schaal), waarbij een hogere waarde in de grafiek hoort, komt vaker voor. Deze interpretatie is vergelijkbaar met die van een histogram. In elke grafiek zijn de hoogste aantallen zijn aanzienlijk minder frequent.

Of er een sterk verband is tussen factoren is zichtbaar in Figuur 2; hier betekent een waarde dicht bij 1 (rood), -1 (blauw) of 0 (wit) dat de factoren respectievelijk sterk positief, sterk negatief of niet gerelateerd zijn. Zo wordt bijvoorbeeld een hoge slibdikte geassocieerd met een hoge slibdiepte, en een hoge pH-waarde met een lage krabbenscheer bedekking. Zo lijkt er dus geen verband te zijn temperatuur en waterdiepte. Figuur 3 suggereert vergelijkbare verbanden door middel van puntgrafieken. Per grafiek komt elk punt overeen met een meting, waarvan twee componenten (bijvoorbeeld slibdikte en temperatuur, laatste kolom tweede rij) zijn weergegeven.

Omdat we voornamelijk geïnteresseerd zijn in een direct effect van de behandeling op de aantallen libellen is een goede eerste stap het weergeven van gemiddelden voor de verschillende behandelingsgroepen. In Figuur 4 is dit gedaan voor elk jaar (links) en voor elke beheerder (rechts). Door de 'scheefheid' in de verdelingen (waarden kleiner dan het gemiddelde komen veel vaker voor dan waarden erboven) van de getelde aantallen libellen geven boxplots een beter beeld van de distributie dan de gemiddelden plusminus standaarddeviaties. Er is geen duidelijk verschil waarneembaar tussen de behandelingsgroepen. Er is daarentegen wel een verminderde slibdikte geassocieerd met het ritsbeheer. Dit is te zien in de linker grafiek in Figuur 5, waar de slibdikte consequent lager is voor het ritsbeheer.

In Paragraaf 2.4 worden de statistische modellen beschreven die gebruikt zijn om formele resultaten te verkrijgen. De reden dat er geen standaard lineaire regressie gebruikt kan worden voor de aantallen libellen en de slibdikte is dat (1) de aantallen niet negatief kunnen zijn en (2) er zich een tijdsfactor in de gegevens bevindt.

De resultaten van het schatten van deze modellen zijn te vinden in Tabel 6. In elke kolom staan geschatte effecten voor een bepaalde afhankelijke variabelen. Deze effecten beschrijven de voorspelde verandering van de (natuurlijke logaritme van de) afhankelijke variabelen wanneer de covariabele in kwestie (bijvoorbeeld zuurstofniveau of EGV) met één stijgt. Positieve en negatieve effecten geven respectievelijk een toename of afname in het gemiddelde aantal van de afhankelijke variabelen aan. Een asterisk duidt aan dat het effect waarschijnlijk significant verschillend is van nul. Zo geeft de coëfficiënt voor beheer (-0.18*) in de laatste kolom aan het ritsbeheer geassocieerd wordt met $\exp(-0.18) = 0.835$ maal zoveel slib, of een vermindering vcan 16.5%. In de andere kolommen staat er geen asterisk achter de coëfficiënten voor beheer, wat betekend dat we met deze modellen en data niet kunnen concluderen dat er een effect is van het ritsbeheer op de aantallen groene glazenmakers. Alle modellen in Tabel 6 zijn beoordeeld op kwaliteit van de fitting, en dit wijst uit dat elk model een redelijk beeld van de werkelijkheid geven.

Als ritsbeheer een vermindering van de hoeveelheid slib teweeg brengt, zou een aangepast ritsbeheer gunstig kunnen zijn voor de groene glazenmaker populaties. Hiervoor zou echter wel een nieuw onderzoek moeten plaatsvinden, al zouden (half-)jaarlijkse metingen van de slibdikte dan volstaan.

2. Statistisch onderzoek

2.1 Introductie

Deze sectie beschrijft de stappen die genomen zijn voor een statistische analyse die toereikend zou moeten zijn om te beoordelen in hoeverre de onderzoeksvraag beantwoord kan worden met behulp van de data. Er blijkt geen *direct* gevolg te zijn van het ritssluitingbeheer (hierna afgekort naar ritsbeheer) op de populatiegrootte van groene glazenmaker.

Dit betekent echter niet dat er geen belangrijke innovatie uit dit onderzoek kan voortvloeien. Een uitkomst van de regressie van slibdikte op beheer en covariabelen is dat ritsbeheer vergeleken met het reguliere beheer naar schatting de slibdikte met circa 16.5% verminderd. Dat zou kunnen betekenen dat er minder krabbenscheer, en daarmee ook een kleiner deel van de larven, verwijderd hoeft te worden. Gezien de data lijkt te bevestigen dat er statistisch gezien geen verschil tussen de aantallen groene glazenmakers in beide beheergroepen is, kan er met een vernieuwd ritsbeheer dus mogelijk een toename in het aantal libellen teweeg worden gebracht. Daar is wel de aanname voor nodig dat de slibdikte hetgeen is dat in bedwang gehouden moet worden, en niet de hoeveelheid krabbenscheer. Daarbij is het mogelijk dat het effect kleiner is dan praktisch merkbaar, aangezien het laten staan van meer vegetatie ook voor meer slip zal zorgen.

De rest van deze paragraaf is als volgt georganiseerd. In Subsectie 2.2 worden de bewerkingen die gedaan zijn op verschillende datasets omschreven en beargumenteerd. Bewerkingen zijn hier bijvoorbeeld het verwijderen van irrelevante of onbruikbare variabelen en observaties, het opsommen of gemiddelden berekenen en het samenvoegen van de sets. Vervolgens dient Subsectie 2.3 om een duidelijk beeld van de beschikbare data te geven. Dit gebeurt met behulp van beschrijvende statistieken als gemiddelden en standaarddeviaties, en tevens door middel van verduidelijkende grafieken. Sommige figuren kunnen bepaalde relaties tussen variabelen suggereren, en dienen als deel van een informele analyse. De methodiek gebruikt in de formele analyse wordt in Subsectie 2.4 beschreven; dit komt neer op het definiëren van de te gebruiken regressiemodellen en het geven van korte motivaties. In Subsectie 2.5 worden de regressieresultaten gepresenteerd en oppervlakkig geïnterpreteerd, waarna de implicaties ervan in Subsectie 2.6 besproken worden. Daarbij worden hier mogelijke extensies genoemd voor het onderzoek wat het statistische deel betreft. Tenslotte zijn fragmenten van scripts in **R**, het statistisch softwarepakket (en programmeertaal) gebruikt bij dit onderzoek, in Appendix 3.1 te vinden.

2.2 Voorbereiding van de data

Alvorens enige statistische analyse is het nodig de data hiervoor te prepareren. Er zijn drie losstaande datasets, namelijk betreffende abiotische gegevens, de krabbenscheer bedekking en de getelde aantallen libellen.

2.2.1 Voorbereiding van de abiotische gegevens en de bedekkingsgraden

In deze gegevens bevinden zich onder andere metingen van de water- en de slibdiepte, en de slibdikte. Zoals eerder genoemd, zijn er voor elk transect steeds tien metingen verricht. De eerste stap is om de gemiddelden uit te rekenen en slechts deze te behouden voor zo precies mogelijke schattingen.

De dataset bevat veel informatie waarvan het onwaarschijnlijk is dat deze nagenoeg constant is gedurende de zomer. Hierbij horen de (lucht)temperatuur, de windkracht en de bewolking. Deze moeten daarom buiten beschouwing worden gelaten, gegeven de lage frequentie waarmee de abiotische factoren zijn gemeten (eens aan het begin van het voor- en van het najaar). Daarbij mist van de submersie en de transparantie tenminste de helft van de gegevens. Voor de kleur en textuur van het water is er te veel variatie in de data voor een zinnige bijdrage; dit geldt ook voor de opmerkingen. De voorgenoemde variabelen worden om deze redenen genegeerd.

Ook zijn veel variabelen gerapporteerd in uiteenlopende orden van grootte, en hierom is het noodzakelijk om enkele ervan in aangepaste eenheden uit te drukken. Zo worden alle percentages veranderd in fracties, redox wordt weergegeven in volt (V) in plaats van millivolt (mV), elektrisch geleidingsvermogen in millisiemens (mS) in plaats van microsiemens (µS) en water en slib metingen in meters (m) in plaats van centimeters (cm).

2.2.2 Voorbereiding van de getelde aantallen groene glazenmakers

Ondanks de aanwezigheid van tellingen met betrekking tot andere libellensoorten, beperkt dit verslag zich tot groene glazenmaker (hierna om deze reden ook naar verwezen als 'libellen'). Dit is, afgezien van de primaire focus van het rapport, ook vanwege de lage getelde aantallen voor andere soorten.

De groene glazenmakers zijn als exuviae en volwassen libellen in verscheidene situaties geobserveerd. Deze analyse concentreert zich op de totale aantallen exuviae en volwassen libellen, evenals op de aantallen eierleggende vrouwtjes. Deze laatste categorie zou sterker met de plek waar ze geteld zijn verbonden zijn dan volwassen exemplaren in het algemeen.

Tenslotte laten we locatie WHA7 buiten beschouwing, aangezien hier geen beheer heeft plaatsgevonden en er daarom geen desbetreffend effect mee geschat kan worden.

2.2.3 Algehele aanpassingen en samenvoeging

In iedere dataset bevinden zich indicators voor zowel reguliere als ritsbeheren. De behandeling wordt echter pas aan het eind van de zomer uitgevoerd, waardoor deze technisch gezien een jaar achterloopt. Daarom is het gepaster om in het jaar 2015 in elke dataset de rits behandelingen in reguliere te veranderen.

Om abiotische factoren en de krabbenscheer bedekking te kunnen gebruiken om het effect van de behandeling op de aantallen libellen gecontroleerd te analyseren, is het nodig om de gegevens samen te voegen. De simpelste manier om dit te doen is door elk getelde aantal libellen te voorzien van abiotische gegevens en de bedekkingsgraad met de dichtstbijzijnde datum (gemeten op hetzelfde transect). Dit vereist slechts dat de abiotische factoren redelijk constant zijn gedurende de zomer; de krabbenscheer bedekkingen zijn daarentegen vaker gemeten (zesmaal per zomer).

Een hoge fractie van de getelde aantallen libellen is gelijk aan nul. Dit maakt dat veel methoden een minder duidelijk beeld kunnen geven van de onderliggende verbanden. Hierom sommen we de getelde aantallen in elk transect op per jaar, aangezien we immers geïnteresseerd zijn in veranderingen van de populatiegrootten op de lange termijn. We berekenen de gemiddelden van de rest van de variabelen, en laten de datum, de ronde, en het rondetype buiten beschouwing.

De uiteindelijke samengevoegde dataset beslaat 17 variabelen en 114 observaties, en bevat geen missende waarden. De eerste zes rijen ervan zijn te vinden in Tabel 1. Tevens zijn er 16 variabelen en 172 observaties in abiotische gegevens.

Tabel 1 De eerste zes rijen van de samengevoegde data na de voorbereidende bewerkingen, met het rijnummer links.

	Manager	Area	Behandeling	Transect	Jaar	A. Vir. (ex.)	A. Vir. (volw.)	A. Vir. (eierl.)
1	GL	GL1	regulier	a	2015	99	11	6
2	GL	GL1	regulier	b	2015	69	11	4
3	GL	GL1	rits	a	2016	72	38	10
4	GL	GL1	regulier	b	2016	59	32	10
5	GL	GL1	rits	a	2017	16	22	11
6	GL	GL1	regulier	b	2017	15	24	13

	Bedekking (fractie)	Zuurtegraad (pH)	Redox (V)	Zuurstof (fractie)	EGV (mS/cm)
1	0.65	6.7	0.035	0.26	0.369
2	0.65	7.0	0.065	0.57	0.376
3	0.63	7.4	0.238	0.69	0.224
4	0.63	7.6	0.225	0.72	0.245
5	0.63	7.3	0.176	0.82	0.706
6	0.63	7.1	0.171	0.63	0.759

	Temperatuur (C)	Waterdiepte (m)	Slibdiepte (m)	Slibdikte (m)
1	14.8	0.485	0.875	0.390
2	15.5	0.460	1.006	0.546
3	7.3	0.514	0.893	0.379
4	7.2	0.532	1.030	0.498
5	18.2	0.326	0.724	0.398
6	18.1	0.282	0.788	0.506

2.3 Beschrijvende statistieken en figuren

In deze sectie worden de te gebruiken datasets zo goed mogelijk in beeld gebracht door middel van beschrijvende statistieken en suggestieve figuren.

2.3.1 Beschrijvende statistieken

Allereerst presenteren we frequenties van factorniveaus in Tabel 2 en gemiddelden, standaarddeviaties, minima en maxima van numerieke variabelen in Tabel 3, uit de samengevoegde dataset.

Tabel 2 Factoren in de samengevoegde dataset en de frequenties van de bijbehorende factorniveaus.

Factor	***Factorniveau*:**	**#Observaties**			
Manager	*GL*: 24	*GV*: 22	*SBBF*: 24	*SBBG*: 2	*WHA*: 42
Gebied	*GL1*: 8	*GL2*: 8	*GL3*: 8	*GV1*: 6	*GV1/2*: 2
	GV2: 6	*GV3*: 8	*SBBF1*: 8	*SBBF2*: 8	*SBBF3*: 8
	SBBG1: 2	*WHA1*: 8	*WHA2*: 8	*WHA3*: 2	*WHA4*: 8
	WHA5: 8	*WHA6*: 8			
Behandeling	*regulier*: 73	*rits*: 41			
Transect	*a*: 57	*b*: 57			
Jaar	*2015*: 32	*2016*: 28	*2017*: 28	*2018*: 26	

Tabel 3 Gemiddelden, standaarddeviaties, minima, medianen, en maxima van numerieke variabelen in de samengevoegde dataset.

Variabele	**Gemiddelde**	**Standaarddeviatie**	**Minimum**	**Mediaan**	**Maximum**
A. Vir. aantal (ex.)	16.325	42.73	0	3	344
A. Vir. aantal (volw.)	12.974	13.773	0	8	64
A. Vir. aantal (eierl.)	5.509	6.875	0	3	32
Bedekking (fractie)	0.77	0.246	0	0.842	1
Zuurtegraad (pH)	7.008	0.549	4.9	6.965	8.1
Redox (V)	0.101	0.068	-0.087	0.107	0.285
Zuurstof (fractie)	0.56	0.286	0	0.559	1.44
EGV (mS/cm)	0.521	0.239	0.133	0.486	1.078
Temp (°C)	13.952	4.202	5.5	15.2	21
Waterdiepte (m)	0.622	0.182	0.232	0.652	1.098
Slibdiepte (m)	0.932	0.218	0.355	0.95	1.414
Slibdikte (m)	0.311	0.149	0.063	0.28	0.746

Resultaten die slechts abiotische factoren betreffen kunnen het best geproduceerd worden met behulp van de oorspronkelijk abiotische gegevens. Om deze reden bevatten Tabel 4 en Tabel 5 dezelfde informatie als de bovenstaande tabellen voor de desbetreffende dataset. Merk op dat de getoonde statistieken van overeenkomende variabelen iets kunnen verschillen, daar er bijvoorbeeld geen gegevens uit maart in de samengevoegde dataset zijn.

Tabel 4 Factoren in de abiotische gegevens en de frequenties van de bijbehorende factorniveaus.

Factor	Factorniveau:	#Observaties			
Manager	*GL*: 36	*GV*: 34	*SBBF*: 36	*SBBG*: 4	*WHA*: 62
Gebied	*GL1*: 12	*GL2*: 12	*GL3*: 12	*GV1*: 10	*GV1/2*: 2
	GV2: 10	*GV3*: 12	*SBBF1*: 12	*SBBF2*: 12	*SBBF3*: 12
	SBBG1: 4	*WHA1*: 12	*WHA2*: 12	*WHA3*: 2	*WHA4*: 12
	WHA5: 12	*WHA6*: 12			
Behandeling	*regulier*: 102	*rits*: 70			
Transect	*a*: 86	*b*: 86			
Jaar	*2015*: 32	*2016*: 30	*2017*: 56	*2018*: 54	
Maand	*maart*: 84	*september*: 16	*oktober*: 72		

Tabel 5 Gemiddelden, standaarddeviaties, minima, mediannen, en maxima van numerieke variabelen in de abiotische gegevens.

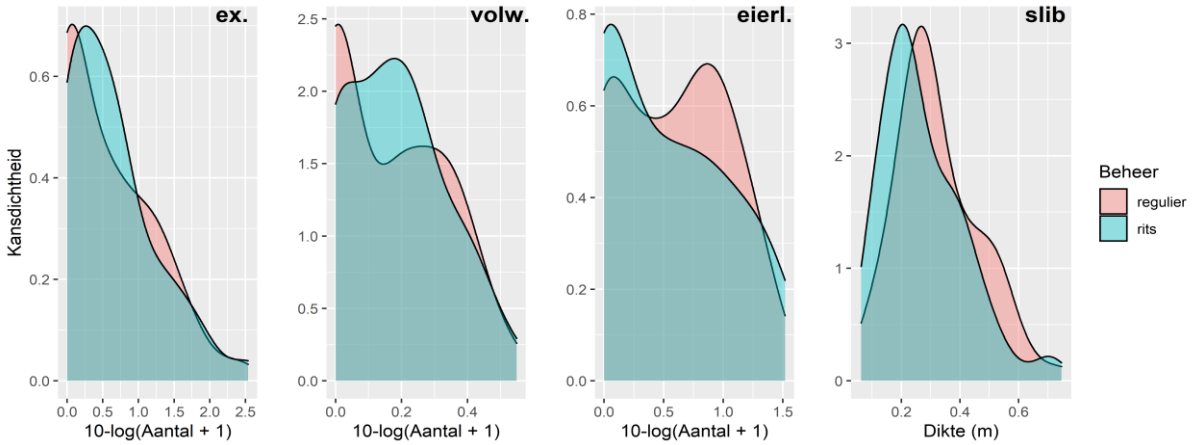
Variabele	Gemiddelde	Standaarddeviatie	Minimum	Mediaan	Maximum
Datum	-	-	9/9/2015	-	5/10/2018
Bedekking (fractie)	0.420	0.398	0.00	0.30	1.00
pH	7.06	0.549	4.90	7.10	8.25
Redox (V)	0.097	0.067	-0.087	0.104	0.285
Zuurstof (fractie)	0.654	0.317	0.000	0.645	1.460
EGV (mS/cm)	0.507	0.224	0.053	0.470	1.078
Temp (°C)	12.25	4.35	5.5	12.05	21.0
Waterdiepte (m)	0.625	0.194	0.220	0.637	1.145
Slibdiepte (m)	0.933	0.232	0.322	0.945	1.684
Slibdikte (m)	0.307	0.150	0.063	0.276	0.746

2.3.2 Spreiding en verdeling van de data

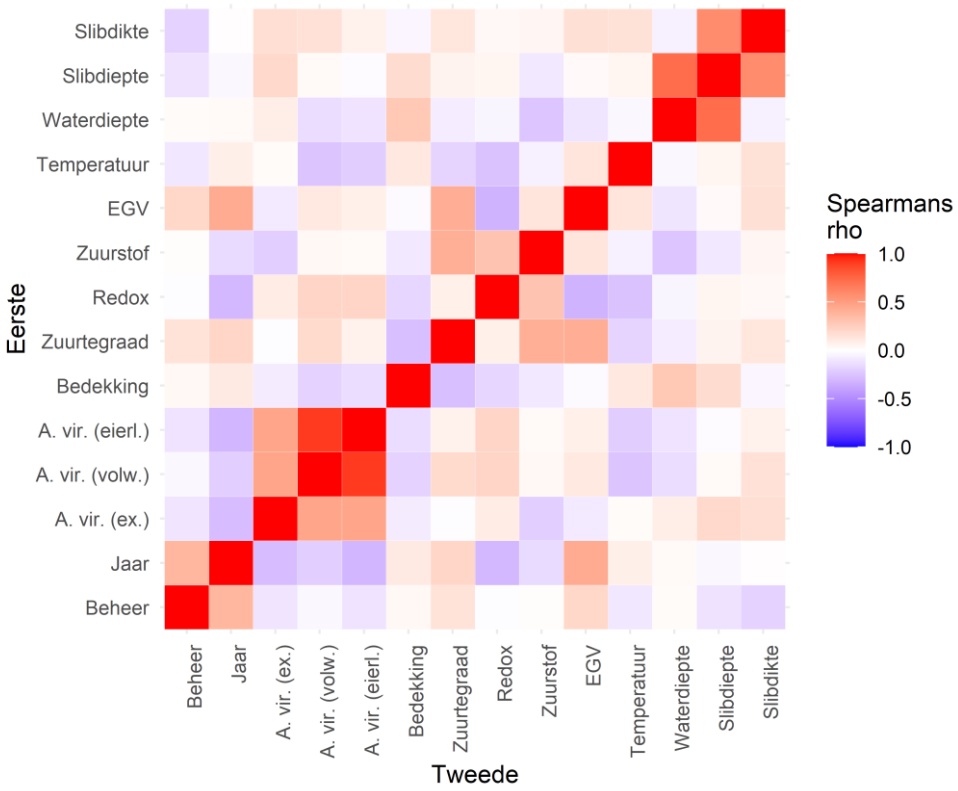
De verdelingen van de getelde aantallen libellen zijn redelijk scheef; dat wil zeggen, er zijn veel kleine aantallen geobserveerd, maar toch zijn er extreme uitschieters. Om de data goed in beeld te brengen nemen we de logaritme met grondtal 10 ($\log_{10}$) van de tellingen. Aangezien logaritmen niet gedefinieerd zijn voor het getal nul, tellen we eerst één bij de tellingen op. De oorspronkelijke getallen kunnen dus worden verkregen door getransformeerde waarden als macht in een 10-macht te nemen, en er vervolgens één vanaf te trekken (oftewel, $oorspr. = 10^{getransf.} - 1$). De resulterende empirische kansverdelingen, te vinden in de linker drie panelen van Figuur 1, bevat nog steeds scheefheid, maar deze is zichtbaar verminderd.

Opmerkelijk is dat er nagenoeg geen verschil in de kansverdelingen tussen de behandelingen lijkt te zijn voor de exuviae en de volwassen libellen (metingen uit 2015 achterwege latend, gezien de afwezigheid van ritsbeheer in dat jaar). Er is wel een duidelijke vermindering te zien in eierleggende vrouwtjes en de slibdikte onder het ritsbeheer.

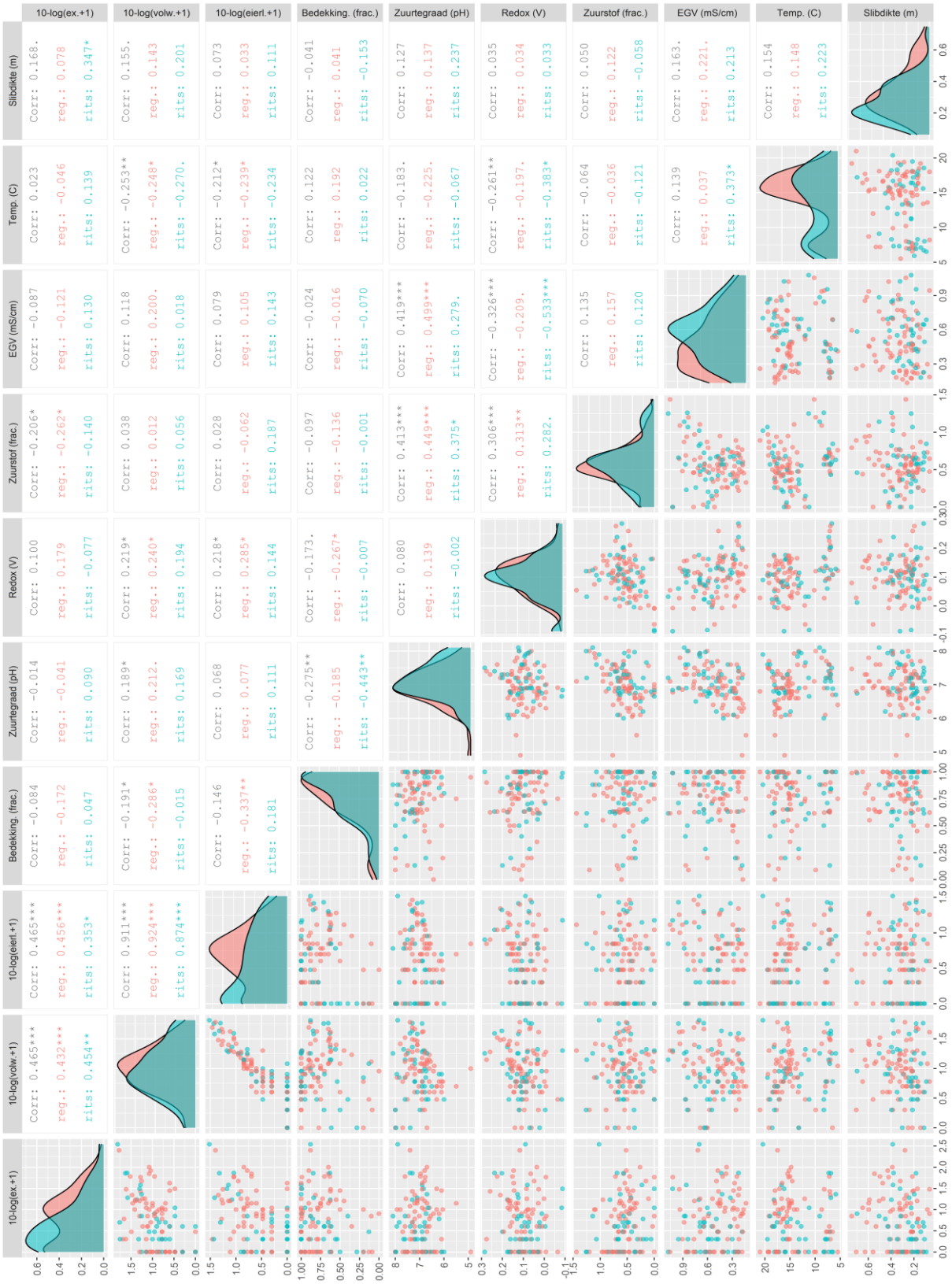
Spearmans rangcorrelatiecoëfficiënt (ook wel *Spearmans rho* genoemd) is voor paren belangrijke variabelen te vinden in Figuur 2. In tegenstelling tot de steekproef (Pearson) correlatiecoëfficiënt is Spearmans rho ook geschikt voor non-lineaire verbanden (Hollander & Wolfe, 1973). Zo is het bijvoorbeeld aannemelijk dat er een bepaalde optimale pH-waarde is in een verband tussen de zuurtegraad en de aantallen groene glazenmakers. In zo'n geval zou een kwadratisch verband bijvoorbeeld passender zijn dan een lineaire verhouding. De correlaties aanwijzingen kunnen zijn voor verbanden, maar deze zijn niet per se oorzakelijk.



Figuur 1 Empirische kansverdelingen van getelde aantallen groene glazenmakers (linker drie panelen) en slibdikte (rechter paneel) voor regulier en ritsbeheer. De tellingen zijn getransformeerd ($\log_{10}(aantal + 1)$) voor de duidelijkheid. Merk op dat de waarden op de y-as verschillen, omdat de oppervlakten onder de grafieken gelijk zijn aan 1. Een hoge kansdichtheid bij een waarde op de horizontale as geeft aan dat deze waarde vaak geobserveerd is. De gegevens uit 2015 zijn niet betrokken bij de berekening van de figuren, aangezien er in dat jaar alleen regulier beheer is toegepast. Op deze manier kan een eerlijke vergelijking gemaakt worden tussen de verschillende beheren. We zien dat er geen overtuigend verschil is bij de exuviae en de volwassen libellen. Aan de andere kant lijken we wel een vermindering te kunnen waarnemen onder het ritsbeheer bij de eierleggende vrouwtjes en de slibdikte.



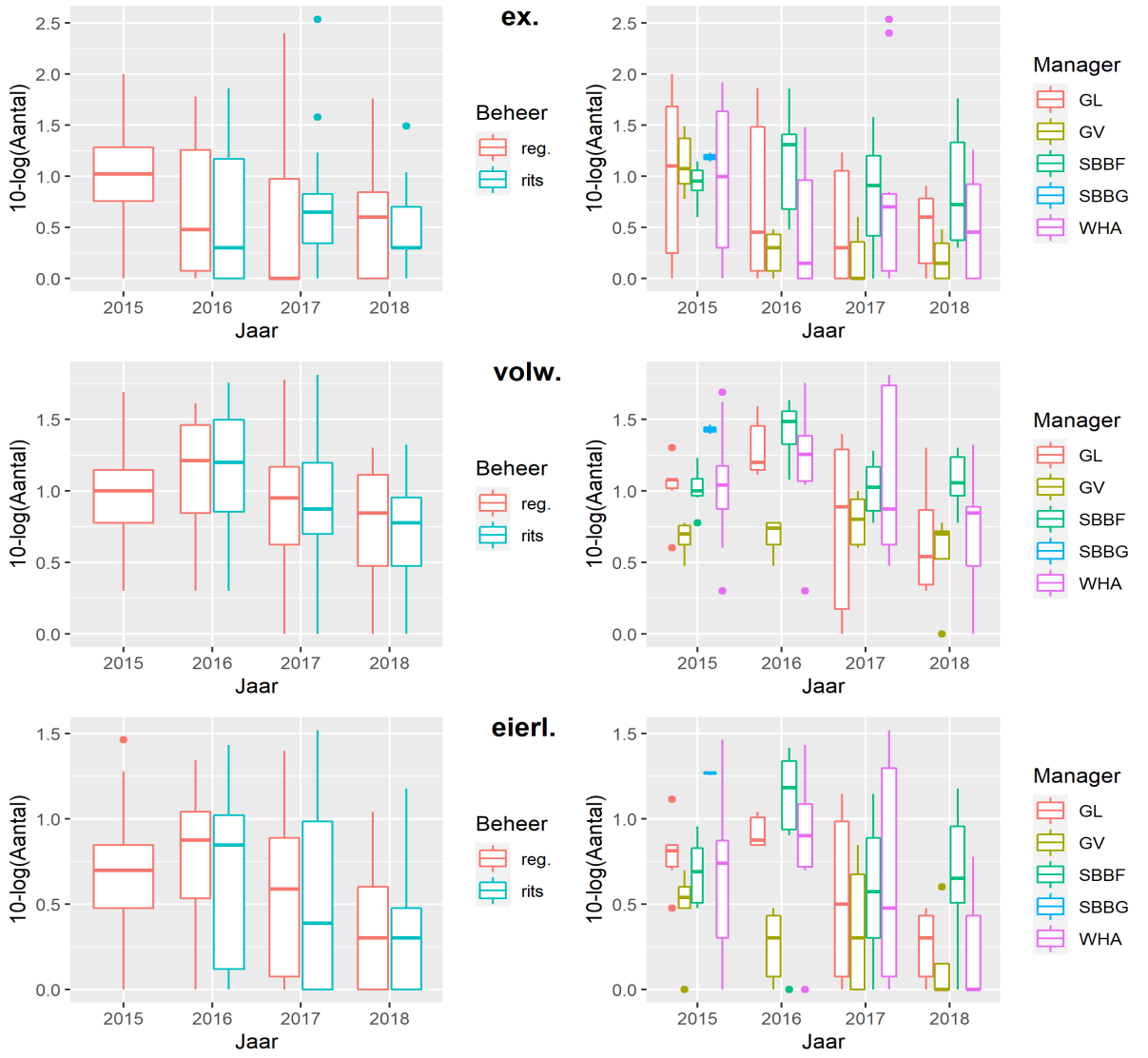
Figuur 2 (Spearmans rang)correlatiecoëfficiënten voor paren belangrijke variabelen. De kleur duidt volgens de legenda de waarde van de correlatiecoëfficiënt aan. Er zijn statistisch significante correlaties tussen beheer en aantallen exuivae (-0.115), slibdiepte (-0.123), en slibdikte (-0.192), tussen aantallen exuivae en volwassenen (0.465), eierleggende vrouwtjes (0.465), EGV (-0.087), en slibdikte (0.168), tussen volwassenen en eierleggende vrouwjes (0.911), bedekkingsgraad (-0.191), redox (0.219), temperatuur (-0.253), en waterdiepte (-0.146), en tussen eierleggende vrouwtjes en bedekkinsgraad (-0.146), redox (0.218), temperatuur (-0.212), en waterdiepte (-0.119) (onder andere).



Figuur 3 Spreiding (onderdriehoek) en Spearmans rho (correlatie, bovendriehoek) voor verschillende beheren van belangrijke variabelen in verhouding tot elkaar, gekleurd op beheer. De afbeelding is een kwartslag geroteerd tegen de klok in. Op de diagonaal is de verdeling van elke variabele geschetst voor beide beheren. Merk op dat de aantallen groene glazenmakers zijn getransformeerd ($\log_{10}(aantal + 1)$) als voorheen; dit betekent dat op de horizontale as de getallen 0, 0.5, 1, 1.5, 2.0 en 2.5 grofweg de tellingen 0, 2, 10, 30, 100 en 315 aangeven.

In Figuur 3 zijn puntgrafieken te zien voor paren belangrijke variabelen, en empirische kansverdelingen van elke betrokken variabele voor beide beheren. De aantallen groene glazenmakers zijn wederom getransformeerd op de eerder genoemde wijze. In de bovendriehoek staan Spearmans rangcorrelatiecoëfficiënten, voor de gehele dataset en voor de beheren afzonderlijk.

De puntgrafieken kunnen een indicatie geven of er een verband tussen twee variabelen is, en hoe dit eruitziet. Als de variabelen in kwestie in werkelijkheid ongerelateerd zijn verwachten we een ovaalvormige 'wolk' observaties. Dit is bijvoorbeeld het geval bij elektrisch geleidingsvermogen en slibdikte, maar juist niet bij zuurtegraad en zuurstof; hogere pH waarden zijn vaak in verband met hoge zuurstof fracties (zie ook de correlatie in bijbehorende Figuur 2). De empirische kansverdelingen laten zien met welke relatieve frequenties verschillende waarden voorkomen in beide beheren. Zo is te zien dat de bedekkingsgraad over het algemeen maar weinig verschilt per beheer, terwijl elektrisch geleidingsvermogen daarentegen duidelijk vaker lagere waarden aanneemt onder het reguliere beheer.

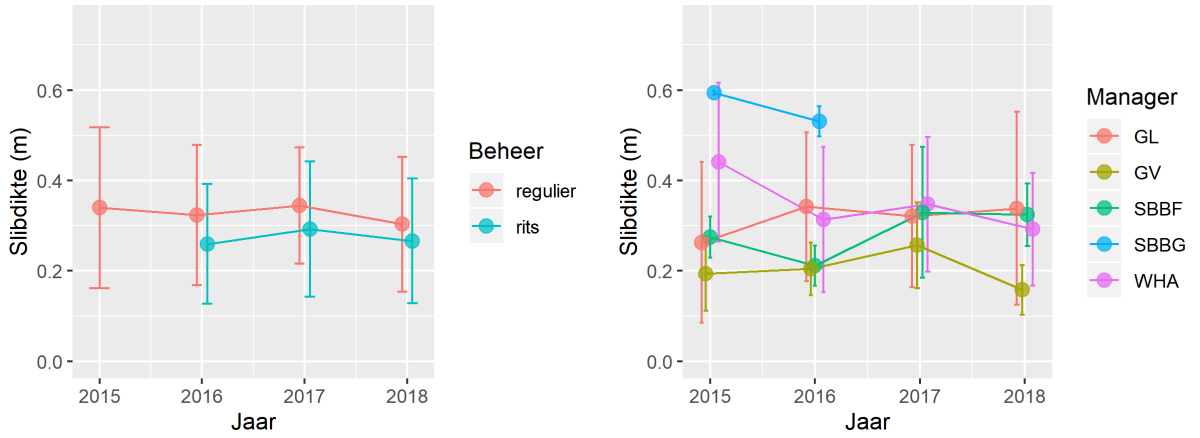


Figuur 4 Boxplots van getelde aantallen groene glazenmakers per behandeling (links) en per manager (rechts), en corresponderende boxplots (onder). Uitschieters, gedefinieerd als observaties verder dan 1.5 maal de interkwartielafstand onder of boven respectievelijk het eerste of het derde kwartiel, zijn als losse punten weergegeven. Merk op dat de aantallen groene glazenmakers zijn getransformeerd ($\log_{10}(aantal + 1)$); dit betekent dat de waarden 0, 0.5, 1, 1.5, 2.0 en 2.5 grofweg de tellingen 0, 2, 10, 30, 100 en 315 aangeven.

2.3.3 Vergelijking van gemiddelden en medianen voor belangrijke variabelen

Met betrekking tot de onderzoeksvraag is het nuttig de centra en de spreidingen (hier de medianen en interkwartielafstanden) van de aantallen groene glazenmakers te vergelijken voor beide beheren in verschillende jaren. De grafieken zijn te vinden links in Figuur 4; rechts zien we voor de compleetheid ook boxplots van de aantallen voor elke manage. De verschillende spreidingen in acht nemende aan de linkerzijde vinden we geen overtuigende aanleiding voor significante verschillen. Dit geldt niet voor de verschillen in aantallen ten opzichte van de managers; zo is bijvoorbeeld in 2015 en 2016 een nadrukkelijk verschil waarneembaar tussen GL en GV.

Naar aanleiding van de eerder genoemde negatieve correlatie tussen beheer en slibdikte zijn ook de gemiddelden plusminus standaarddeviaties hiervoor te zien in verschillende jaren links in Figuur 5. Omdat er minder scheefheid in de verdeling van slibdikten zit, cf. Figuur 3, zijn boxplots hier niet noodzakelijk. Opnieuw tonen we ook de metingen voor verschillende managers voor een compleet beeld. Deze grafieken zijn gemaakt met enkel de abiotische gegevens om ongewenste kopieën (een set metingen kan aan maar liefst zes getelde aantallen groene glazenmakers gekoppeld zijn) niet mee te nemen in de berekeningen, en zodat data uit maart erbij betrokken wordt. De gemiddelden zijn consequent lager voor de rits-behandeling vergelijken met de reguliere. Hetzelfde geldt voor, bijvoorbeeld, GV vergeleken met WHA.



Figuur 5 Jaargemiddelden van slibdikten plusminus standaarddeviaties per beheer (links) en per manager (rechts).

2.4 Methodiek

In Paragraaf 2.5 wordt (onder andere) het effect van het beheer geanalyseerd in verschillende regressie-modellen. Deze subparagraaf dient ertoe deze modellen uit te lichten en te voorzien van enige intuïtie. De inhoud is met name belangrijk voor de compleetheid, dus de paragraaf kan in principe overgeslagen worden. De belangrijkste eigenschappen zijn dat de modellen rekening houden met het nonnegatieve karakter van de tellingen en de slibdikten, evenals de herhaalde metingen.

2.4.1 Regressiemodel voor aantallen Aeshna viridis

Zoals eerder genoemd, dient er rekening gehouden te worden met de nonnegativiteit van de aantallen groene glazenmakers, en dat deze waarden uitsluitend gehele getallen zijn. Een veelgebruikte techniek is om de natuurlijke logaritmen te nemen van de verwachte aantallen en vervolgens een lineair verband met de verklarende variabelen (dat wil zeggen, het beheer en de verstorende factoren) te zoeken. Het gebruik van verwachte aantallen maakt ook ongehele getallen als voorspellingen zinnig.

De simpelste manier is om aan te nemen dat de aantallen een Poisson verdeling volgen, oftewel

$$\mathbb{P}(aantal_i = y_i \mid \mu_i) = \frac{\mu_i^{y_i}\, \mathrm{e}^{-\mu_i}}{y_i!}, \qquad y_i = 0, 1, 2, \ldots$$

We zullen deze kansverdeling gebruiken voor de getelde aantallen volwassenen. Deze distributie heeft echter als eigenschap dat het gemiddelde gelijk is aan de variantie. Gezien dit niet het geval is voor de getelde aantallen exuviae en eierleggende vrouwtjes, is het gepaster om aan te nemen dat ze een geometrische verdeling aanhouden. Met deze verdeling kunnen we de hogere mate van spreiding beter modelleren. Dit betekent dat de kans op het observeren van aantal y_i voor observatie i gegeven is als

$$\mathbb{P}(aantal_i = y_i \mid \mu_i) = \frac{\mu_i^{y_i}}{(\mu_i + 1)^{y_i+1}}, \qquad y_i = 0, 1, 2, \ldots$$

Hier is μ_i de verwachtingswaarde van het i^{de} aantal en ϕ een parameter die later geschat wordt. Een mogelijk model dat rekening houdt met nonnegatieve tellingen is daarmee

$$\mu_i = \exp\left(\beta_0 + \beta_1 \cdot \text{ritsbeheer} + \boldsymbol{\beta} \cdot \textbf{controlevariabelen}\right), \qquad i = 1, \ldots, n$$

waar n het aantal observaties is en de subscripten i weggelaten zijn aan de rechterkant van het vergelijkingsteken voor de leesbaarheid. Hier is $\boldsymbol{\beta} \cdot \textbf{controlevariabelen} = \beta_2 \cdot \text{controle-1} + \beta_3 \cdot \text{controle-2} + \ldots$, en de controlevariabelen zijn bijvoorbeeld de beheerder of de krabbenscheerbedekking. Ook is het mogelijk deze term in zijn geheel weg te laten; er zijn dan slechts drie te schatten parameters, namelijk β_0, β_1 en ϕ. Een dergelijk model is een (vorm van een) *Generalized Linear Model* (Dobson & Barnett, 2008). De parameters worden geschat door de aannemelijkheid van alle gegevens tezamen,

$$L(\phi, \beta_0, \beta_1, \boldsymbol{\beta} \mid y_1, \ldots, y\_n) = \mathbb{P}(aantal_1 = y_1 \mid \mu_1, \phi) \cdot \cdots \cdot \mathbb{P}(aantal_n = y_n \mid \mu_n, \phi),$$

te maximaliseren ten opzichte van $\phi, \beta_0, \beta_1, \boldsymbol{\beta}$. Dit gebeurt in de praktijk door middel van een toegespitste numerieke benadering.

De onderlinge afhankelijkheid van observaties op hetzelfde transect kan een probleem vormen bij het schatten van het bovengenoemde model. Een oplossing is het bijvoegen van zogeheten willekeurige effecten, waarmee we de gegevens opdelen in groepen waarbinnen ze onafhankelijk zouden moeten zijn. Dit betekent dat we een willekeurig effect per locatie toevoegen. Het nieuwe model wordt dus

$$\mu_i = \exp(\beta_0 + \beta_1 \cdot \text{ritsbeheer} + \boldsymbol{\beta} \cdot \textbf{controlevariabelen} + b_1 \cdot \text{locatie(GL1)} + \cdots + b_{17} \cdot \text{locatie(WHA6)},$$

voor $i = 1, \dots, n$. Hier is elke locatie(LOC) een dummyvariabele die gelijk is aan 1 als observatie i waargenomen is op locatie LOC, en anders aan 0. De parameters $b_1, \dots, b_{17}$ zijn de willekeurige effecten, die onafhankelijk zijn en een normale verdeling volgen met gemiddelde 0 en (nader te bepalen) standaarddeviatie τ. De parameters $\beta_0, \beta_1, \boldsymbol{\beta}$, en τ worden geschat door, nadat er geïntegreerd is m.b.t. de willekeurige effecten, de aannemelijkheid te maximaliseren. Een regressiemodel zoals hier beschreven met zowel willekeurige als onwillekeurige effecten is een vorm van een *Generalized Linear Mixed Model* (Zuur, Ieno, Walker, Saveliev, & Smith, 2009).

2.4.2 Regressiemodel voor slibdikte

Daar de slibdikte een continue variabele is dient hiervoor een ander model toegepast te worden. Een gamma kansdichtheidsfunctie is in dit geval gepaster. Voor een waarde y_i voor slibdikte i is deze gegeven als

$$f(y_i \mid \mu_i, \sigma) = \frac{1}{\Gamma(\sigma^{-2})\,(\mu_i \sigma^2)^{\sigma^2}} y_i^{\sigma^{-2}-1} e^{-\frac{y_i}{\mu_i \sigma^2}}, \qquad -\infty < y_i < \infty,$$

waar $\sigma > 0$ gelijktijdig met de rest van de parameters geschat wordt. Hier is $\Gamma(\cdot)$ de Gamma functie, de (verschoven) interpolatie van de faculteit. Ook hier nemen we een model

$$\mu_i = \exp(\beta_0 + \beta_1 \cdot \text{ritsbeheer} + \boldsymbol{\beta} \cdot \textbf{controlevariabelen} + b_1 \cdot \text{locatie(GL1)} + \cdots + b_{17} \cdot \text{locatie(WHA6)},$$

voor μ_i, waar $\beta_0, \beta_1, \boldsymbol{\beta}$, **controlevariabelen**, en $b_1, \dots, b_{17}$ zijn gedefinieerd als voorheen. De parameters worden op dezelfde manier geschat door de geïntegreerde waarschijnlijkheid te maximaliseren.

2.5 Formele analyse

In deze subparagraaf schatten we de modellen beschreven in Subparagraaf 2.4 met behulp van de gegevens, en presenteren we de geschatte parameters en hun significanties. Deze worden hier kort besproken; minder oppervlakkige interpretaties worden uitgesteld tot Subparagraaf 2.6. Verder verifiëren we dat de gevonden modellen goed op de data passen.

De regressieresultaten zijn te vinden in Tabel 6. De betrokken covariabelen zijn gekozen door een heuristische methode die stapsgewijs de minst waardevolle variabelen vanaf het maximale model verwijderd. Vervolgens zijn interacties met de overgebleven variabelen toegevoegd en is het proces herhaald. Voor de exuviae hoorde beheer niet bij het maximale model, omdat er dan geen convergentie tot een oplossing plaatsvond. Voor de volwassen libellen leverde de toevoeging van interacties helaas geen passend model op. De reden om geen grondigere methode te gebruiken om eventuele interacties daarvoor te ontdekken is om overfitting tegen te gaan.

De marginale effecten op de gemiddelde aantallen kunnen verkregen worden door de gegeven parameterschatting te exponentiëren ($vermenigvuldigingsfactor = \exp(parameter)$). De effecten zijn statistisch significant (dit nemen we hier als: gelijk aan nul met een kans kleiner dan of gelijk aan 0.05) als nul geen deel uitmaakt van het bijbehorende 95% betrouwbaarheidsinterval. Deze effecten zijn voorzien van een asterisk. Neem bijvoorbeeld de parameter voor de bedekkinsgraad (3.45) in de eerste kolom van Tabel 6; deze impliceert dat er gemiddeld $\exp(3.45) = 31.5$ maal zoveel exuivae zijn als de krabbenscheer bedekking met één toeneemt. Aangezien deze fractie zich altijd tussen nul en één bevindt, kunnen we hier alleen het verschil uit opmaken tussen volledige bedekking door en afwezigheid van krabbenscheer. Als we het effect van een toename ongelijk aan één willen vinden vermenigvuldigen we eerst de parameter met deze toename ($vermenigv. = \exp(toename \times param.)$). Daarmee wordt een toename van 10% in krabbenscheer bedekking dus geassocieerd met gemiddeld $\exp(0.1 \times 3.45) = 1.41$ maal zoveel exuviae. Voor een afname nemen we simpelweg een negatieve toename. Merk op dat 'effect' hier niet noodzakelijkerwijs een oorzakelijk verband aanduidt; de betekenis die eraan verbonden wordt in deze uitleg ligt dichter bij 'associatie'.

Merk op dat het effect van het ritsbeheer in geen van de modellen voor de groene glazenmakers statistisch significant is. Dit houdt in dat we met behulp van deze regressieresultaten voor de beschikbare data niet kunnen concluderen dat het beheer en de gemiddelde aantallen libellen direct gerelateerd zijn. Dit is niet het geval voor slibdikte, daar zien we dat het ritsbeheer geassocieerd wordt met een grofweg 16.5% ($\exp(-0.18) \approx 0.835$).

Als we niet statistisch significante effecten achterwege laten, zien we verder dat de jaren 2016 en 2017, het zuurstofniveau, en de slibdikte de gemiddelde aantallen exuivae negatief beïnvloeden. Wegens de interactie tussen het jaar en het EGV is er een negatief effect in 2015 en 2016 (de respectievelijke geassocieerde vermenigvuldigingen bedragen $\exp(0.1 \times -7.61) = 0.47$ en $\exp\big(0.1 \times (-7.61 + 5.80)\big) = 0.83$ per 100 µS/cm toename). Dit effect is echter positief in 2017 en 2018 (vergelijkbare berekeningen duiden op respectievelijke toenamen van 16.8% en 26.6%).

Op de gemiddelde aantallen volwassen libellen hebben volgens Tabel 6 het jaar 2016, de bedekkingsgraad, de waterdiepte, en de temperatuur een positief effect. Het jaar 2018, het zuurstofniveau, het EGV, en de slibdikte hebben daarentegen een negatieve invloed. Voor de eierleggende vrouwtjes lijkt slechts het jaar 2018 geassocieerd te zijn met een aantoonbare vermindering van de gemiddelde getelde aantallen. Naast het beheer zijn er voor slibdikte geen andere statistisch significante parameterschattingen voor andere covariabelen.

Enkele willekeurige locatie-effecten zijn statistisch significant, zoals te zien in Tabel 7. Zo zien we dat er een aanzienlijk hogere aantallen exuviae geassocieerd zijn met GL1, SBBG1, en WHA6, terwijl de aantallen naar verwachting lager zijn in GL2 en GV2. Voor volwassenen zijn de aantallen hoger in GL1, WHA4, en WHA6, maar lager in WHA1, WHA3, en WHA5. De locaties SBBF3, WHA4 en WHA6 zijn gerelateerd aan hogere gemiddelde aantallen eierleggende vrouwtjes, en anderzijds verwachten we lagere aantallen in GV2, WHA1, WHA3, en WHA5. De schatting is dat er meer slib aanwezig is in GL1 en WHA3, maar minder in WHA4. De variantie van de willekeurige effecten is relatief hoog (5.58) voor de exuviae en relatief laag (0.08) voor de slibdikte.

Tabel 6 Resultaten van het regresseren van aantallen groene glazenmakers en slibdikte op beheer en covariabelen. Parameterschatting geven het effect aan op de natuurlijke logaritme van de getelde aantallen als er sprake is van het corresponderende factorniveau, of als de bijbehorende variabele met één stijgt (welke van toepassing is), en zijn voorzien van 95% betrouwbaarheidsintervallen (met 95% kans licht de echte, onbekende waarde in dit interval). Een parameterschatting is statistisch significant als nul zich niet in dit interval bevindt; dit is aangeduid met een *. De observatie onder regulier beheer, beheerd door GL, in het jaar 2015 met de overige variabelen gelijk aan nul dient als referentie.

	Afhankelijke variabele:			
	Exuviae	Volwassenen	Eierleggende vrouwtjes	Slibdikte
(Constante)	4.84 * [1.09; 8.38]	0.60 [-0.82; 1.86]	1.54 * [1.00; 2.11]	-1.14 * [-1.47; -0.83]
Beheer (rits)		0.05 [-0.09; 0.23]	-0.04 [-0.56; 0.48]	-0.18 * [-0.31; -0.02]
Manager (GV)		-1.21 [-2.74; 0.47]		-0.42 [-0.96; 0.03]
Manager (SBBF)		0.18 [-1.28; 1.95]		-0.01 [-0.47; 0.45]
Manager (SBBG)		1.64 [-0.06; 3.93]		0.61 [-0.23; 1.48]
Manager (WHA)		0.10 [-1.32; 1.62]		0.14 [-0.29; 0.59]
Jaar (2016)	-2.06 [-5.65; 0.67]	1.17 * [0.70; 1.56]	0.26 [-0.48; 0.99]	
Jaar (2017)	-5.67 * [-8.24; -2.22]	0.11 [-0.08; 0.34]	-0.18 [-0.86; 0.57]	
Jaar (2018)	-7.10 * [-11.11; -3.64]	-0.51 * [-0.77; -0.22]	-1.30 * [-1.96; -0.53]	
Zuurstof (frac.)	-3.11 * [-5.07; -1.37]	-0.53 * [-0.84; -0.05]		
EGV (mS/cm)	-7.61 * [-11.06; -3.40]	-0.61 * [-1.15; -0.19]		
Waterdiepte (m)		0.81 * [0.13; 1.52]		
Bedekkinsgraad (frac.)	3.45 * [1.60; 5.08]	1.21 * [0.83; 1.66]		
Slibdikte (m)	-3.33 * [-5.98; -0.08]	-0.91 * [-1.48; -0.18]		
Temperatuur (C)	0.05 [-0.16; 0.25]	0.07 * [0.02; 0.10]		
Year (2016) * EGV	5.80 * [0.30; 11.47]			
Year (2017) * EGV	9.16 * [3.12; 13.77]			
Year (2018) * EGV	9.97 * [4.11; 16.22]			
Behandeling (rits) * EGV	-3.10 [-6.06; 1.24]			
Behandeling (rits) * Temperatuur	0.15 [-0.02; 0.29]			
AIC	696.35	850.48	606.38	-235.24
Log waarschijnlijkheid	-333.18	-409.24	-297.19	125.62
Aantal observations	114	114	114	172
Aantal locaties	17	17	17	17
Willekeurige effecten variantie (Locatie)	5.58	0.77	0.76	0.08

Tabel 7 Geschatte willekeurige locatie-effecten voor de modellen in Tabel 6, voorzien van 95% betrouwbaarheidsintervallen.

	Afhankelijke variabele:			
	Exuviae	Volwassenen	Eierleggende vrouwtjes	Slibdikte
GL1	2.70 *	1.02 *	0.49	0.47 *
	[0.98; 8.64]	[0.62; 3.48]	[-0.74; 2.24]	[0.53; 1.44]
GL2	-5.15 *	-0.83	-0.15	-0.26
	[-15.41; -5.36]	[-2.88; 0.05]	[-2.08; 0.95]	[-1.04; 0.06]
GL3	-0.20	-0.20	-0.10	-0.20
	[-5.66; 2.43]	[-2.13; 1.11]	[-1.78; 1.25]	[-0.96; 0.05]
GV1	-2.43 *	-0.42	-0.89	-0.16
	[-9.81; -0.52]	[-2.16; 0.67]	[-3.24; 0.01]	[-0.81; 0.11]
GV1/2	-0.02	0.60	0.09	-0.14
	[-3.70; 2.62]	[-0.60; 2.01]	[-1.27; 0.99]	[-0.64; 0.27]
GV2	-2.39 *	-0.43	-1.02 *	0.14
	[-8.58; -0.29]	[-2.24; 0.32]	[-3.66; -0.23]	[-0.20; 0.76]
GV3	-1.51	0.25	-0.44	0.15
	[-7.31; 2.09]	[-0.68; 1.75]	[-2.63; 0.58]	[-0.16; 0.84]
SBBF1	-0.51	-0.11	0.54	0.22
	[-6.14; 3.50]	[-1.50; 1.43]	[-0.41; 2.76]	[-0.19; 0.81]
SBBF2	-0.03	0.12	-0.10	-0.17
	[-5.50; 3.91]	[-1.44; 1.53]	[-2.01; 0.89]	[-0.78; 0.30]
SBBF3	1.08	-0.01	1.10 *	-0.05
	[-2.41; 6.68]	[-1.54; 1.13]	[0.08; 3.78]	[-0.54; 0.35]
SBBG1	2.93 *	0.00	0.85 *	0.00
	[1.36; 8.29]	[-0.00; 0.00]	[0.44; 2.99]	[-0.00; 0.00]
WHA1	-0.65	-0.98 *	-0.76 *	0.14
	[-6.50; 3.14]	[-3.60; -0.27]	[-3.17; -0.37]	[-0.15; 0.78]
WHA2	-0.01	0.19	0.12	-0.19
	[-4.28; 4.32]	[-1.15; 1.69]	[-1.49; 1.94]	[-0.79; 0.07]
WHA3	-0.34	-1.00 *	-0.79 *	0.24 *
	[-5.42; 3.62]	[-3.47; -0.66]	[-3.12; -0.32]	[0.14; 0.85]
WHA4	1.12	1.50 *	1.23 *	-0.35 *
	[-2.04; 5.60]	[1.19; 4.60]	[0.84; 4.21]	[-1.08; -0.14]
WHA5	1.43	-0.79 *	-0.96 *	-0.11
	[-1.45; 6.73]	[-3.27; -0.11]	[-3.73; -0.60]	[-0.85; 0.13]
WHA6	3.99 *	1.08 *	0.77 *	0.27
	[4.04; 11.09]	[0.89; 3.52]	[0.04; 3.40]	[-0.03; 1.02]
AIC	696.35	850.48	606.38	-235.24
Log waarschijnlijkheid	-333.18	-409.24	-297.19	125.62
Aantal observaties	114	11	114	172
Aantal locaties	17	17	17	17
Willekeurige effecten variantie (Locatie)	5.58	0.77	0.76	0.08

We beoordelen of de modellen goed op de data passen door willekeurige kwantielresiduen te vergelijken met een uniforme verdeling op het eenheidsinterval (Hartig, 2017). Met een Kolmogorov-Smirnov test komen we zo op de voor de modellen in eerste tot en met de laatste kolom in Tabel 6 op de respectievelijke p-waarden 0.91, 0.97, 0.17, en 0.34 uit. Deze waarden (allen groter dan 0.05) duiden aan dat de geobserveerde data met voldoende kans gegenereerd zou kunnen zijn door de gevonden modellen, welke daarmee als passend gezien kunnen worden. Het is dus aannemelijk dat de parameters in Tabel 6 en Tabel 7 een redelijk beeld van de werkelijkheid geven.

2.6 Conclusies en vooruitzicht

De resultaten in Paragraaf 2.5 laten zien dat er geen waarneembaar direct effect is van het beheer op aantallen groene glazenmakers. Verlaagde aantallen exuviae lijken geassocieerd te zijn met hogere waarden van het zuurstofniveau en de slibdikte. De krabbenscheer bedekking heeft naar schatting een positieve invloed op de aantallen. Er is een interactie tussen EGV en het jaar, waardoor EGV een negatief effect heeft in 2015 en 2016, maar een positief effect in 2017 en 2018. Het is mogelijk dat er een wisselwerking is, of dat de verlaagde aantallen exuviae het verhoogde EGV teweegbrengen.

De aantallen volwassen libellen zijn aanzienlijk hoger geschat voor 2016; dit is tevens te zien in de data. Opnieuw hebben het zuurstofniveau en de slibdikte een negatief effect, en hetzelfde geldt in geval voor EGV. Anders dan bij de exuviae hebben de waterdiepte en temperatuur een positief geschat effect dat statistisch significant is. Op de aantallen eierleggende vrouwtjes lijkt enkel het jaar 2018 een (negatief) effect te hebben.

Er is een zichtbaar negatief effect van ritsbeheer op slibdikte ontdekt, en dit verband is mogelijk oorzakelijk. Om precies te zijn ritsbeheer geassocieerd is met een vermindering van 16.5%. Voor het gevonden model zijn geen andere parameters statistisch significant, suggererende dat de slibdikte alleen bepaald wordt door het beheer. Het negatieve effect van ritsbeheer op slibdikte kan merkbare gevolgen hebben. Aannemende dat het doel is de slibdikte beneden een bepaald peil te houden, kan dit resultaat betekenen dat met het ritsbeheer een kleiner oppervlak aan krabbenscheer verwijderd hoeft te worden, of dat de verwijderingen minder frequent hoeven te zijn. Gezien de samenhang van populatiegrootte van de groene glazenmakers met de aanwezigheid van krabbenscheer biedt dit wellicht een mogelijkheid om, naast het zorgen voor minder onderhoudskosten, het welzijn van de soort te verbeteren.

De verhouding tussen de tellingen exuviae en volwassen libellen is mogelijk bestudeerbaar in een model met een tweedimensionale respons. Met die opzet kan de correlatie gemodelleerd worden in termen van de beschikbare covariabelen. Een mogelijkheid is een Bayesiaans model, gebaseerd op een tweedimensionale geometrische kansverdeling dat minder gevoelig zou moeten zijn voor extreme uitschieters in de data (van Oppen, 2020). De huidige resultaten lijken erop te wijzen dat de correlatie tussen de twee telmethoden positief is. Bovendien is deze 25% lager geschat bij minimale ten opzichte van gemiddelde tot maximale slibdikte. Wat betreft de andere covariabelen is geen statistisch significante verandering gevonden. Het is momenteel echter nog niet zeker of de huidige procedure om de parameterverdelingen te schatten volledige betrouwbaar is, dus tot die tijd kunnen deze indicaties nog niet als sterk onderbouwd gezien worden.

Bibliografie

Dobson, A. J., & Barnett, A. (2008). *An introduction to generalized linear models.* CRC press.

Hartig, F. (2017). DHARMa: residual diagnostics for hierarchical (multi-level/mixed) regression models. R package. Vienna, Austria: CRAN. https://CRAN.R-project.org/package=DHARMa.

Hollander, M., & Wolfe, D. A. (1973). A distribution-free test for indepedence based on ranks (Spearman). In *Nonparametric Statistical Methods* (pp. 427--440). New York: Wiley.

van Oppen, Y. (2020, Augustus). Statistical analysis of the aeshna viridis (green hawker) populations in the northern Netherlands. *Masterthesis*. Groningen, Nederland: Rijksuniversiteit Groningen.

Zuur, A., Ieno, E. N., Walker, N., Saveliev, A. A., & Smith, G. M. (2009). *Mixed effects models and extensions in ecology with R.* Springer Science & Business Media.

3. Appendix

3.1 R code voor statistische analyse

Deze subparagraaf is bedoeld om de gebruikte **R** code in essentie weer te geven voor de reproduceerbaarheid van de resultaten. **R** is een softwarepakket en programmeertaal toegespitst op statistiek en data-analyse. De volgende functies en stukken code nemen aan dat de `data.frame` objecten `abiotic` en `df` (te laden vanuit de bijgevoegde bestanden `abiotic.csv` en `dragonflies.csv`) beschikbaar zijn, die respectievelijk de samengevoegde en de abiotische gegevens bevatten. Als deel van een Engelstalige masterthesis zijn de functies, de variabelen en de `data.frames` gegeven in het Engels. Om de lengte van deze subsectie te beperken worden alleen coderegels getoond die gebruikt zijn voor berekeningen; bijna alle figuren zijn te reproduceren met behulp van functies uit de bibliotheek `ggplot2`. Figuur 3 is gemaakt met de functie `ggpairs()` uit de bibliotheek `GGally`. De tekst in de coderegels hieronder na een #-teken dient als commentaar en wordt niet uitgevoerd.

3.1.1 Spearmans rangcorrelatiecoëfficiënten

Spearmans rangcorrelatiecoëfficiënten en bijbehorende significanties voor paren belangrijke variabelen kunnen berekend worden met de volgende regels **R** code. Om de beheren erbij te betrekken dienen deze eerst naar getallen geconverteerd te worden (de volgende secties nemen aan dat deze bewerking vervolgens ongedaan gemaakt is).

```
library(Hmisc)                                  # p-values for Spearman's rho

                                                # import data
dragonflies <- read.csv("dragonflies.csv", stringsAsFactors = T)

                                                # make numeric
dragonflies$treatment <- as.numeric(dragonflies$treatment)

                                                # compute correlation coefficients and p-values
corcoefs  <- cor(dragonflies[-(1:3)], method = "spearman")
pvalues <- rcorr(corcoefs)$P
```

3.1.2 Selectie, regressie en kwaliteit van de fitting

In het volgende script worden de covariabelen geselecteerd voor de aantallen groene glazenmakers en de slibdikte. De bijbehorende parameters worden gelijktijdig geschat. Tevens wordt de kwaliteit van de fitting beoordeeld door willekeurige kwantielresiduen te vergelijken met de uniforme verdeling op het eenheidsinterval met behulp van een Kolmogorov-Smirnov test.

```
library(buildmer)                               # model selection
library(DHARMa)                                 # goodness-of-fit using randomized quantile residuals

                                                # import data
abiotic     <- read.csv("abiotic.csv",     stringsAsFactors = T)
dragonflies <- read.csv("dragonflies.csv", stringsAsFactors = T)

abiotic$year     <- as.factor(abiotic$year) # convert years to factors
dragonflies$year <- as.factor(dragonflies$year)

                                                # model selection
exuviae <- buildglmmTMB(A_vir_ex ~ treatment + (year + emers_frac + redox_V + O2_frac
                                                + EC_mS_cm + temp_C + (1 | area))^2,
                        include = ~ treatment, data = dragonflies, family = nbinom2,
                        map=list(betad=factor(NA)), start=list(betad=log(1)))

adults <- buildglmmTMB(A_vir_ad ~ treatment + manager + year + emers_frac + pH + redox_V
                                          + O2_frac + EC_mS_cm + temp_C + water_depth_m
                                          + sludge_thickness_m + (1 | area),
                       include = ~ treatment, data = totals, family = poisson)
```

```
egglaying <- buildglmmTMB(A_vir_eggl ~ treatment + manager + year + emers_frac + pH + redox_V
                                       + O2_frac + EC_mS_cm + temp_C + water_depth_m
                                       + sludge_thickness_m + (1 | area),
                          include = ~ treatment, data = dragonflies, family = nbinom2,
                          map=list(betad=factor(NA)), start=list(betad=log(1)))

sludge <- buildglmmTMB(sludge_thickness_m ~ treatment + manager + year + emers_frac + pH
                                            + redox_V + O2_frac + EC_mS_cm + temp_C
                                            + water_depth_m + (1 | area),
                       include = ~ treatment, data = abiotic, family = Gamma(link = "log"))

summary(  exuviae@model)                    # summaries
summary(   adults@model)
summary(egglaying@model)
summary(   sludge@model)
                                            # goodness-of-fit tests
ks.test(simulateResiduals(  exuviae@model, n = 2000, use.u = T)$scaledResiduals, punif, 0, 1)
ks.test(simulateResiduals(   adults@model, n = 2000, use.u = T)$scaledResiduals, punif, 0, 1)
ks.test(simulateResiduals(egglaying@model, n = 2000, use.u = T)$scaledResiduals, punif, 0, 1)
ks.test(simulateResiduals(s   ludge@model, n = 2000, use.u = T)$scaledResiduals, punif, 0, 1)
```

3.1.3 Bootstrap 95% betrouwbaarheidsintervallen

De betrouwbaarheidsintervallen in Tabel 6 en Tabel 7 zijn berekend met behulp van bootstrapping. De functie `basicbootCI()` berekent het basis-bootstrapinterval (95% betrouwbaarheid) voor de paramater in het `model` met index `parindex`.

```
library(boot)                               # bootstrapping

basicbootCI <- function(model, parindex) {  # generate bootstrap sample
    sample <- bootMer(model, function(mod) mod$fit$parfull[parindex], nsim = 100)

                                            # compute basic bootstrap 95% CI
    return(boot.ci(sample, type = "basic", index = idx)$basic[4:5])
}
```