Semiparametric Bayesian Count Data Models

Leyre Estíbaliz Osuna Echavarría

Dissertation
at the Faculty of Mathematics, Computer Sciences and Statistics
Ludwig–Maximilians–University
Munich

1st June 2004

Semiparametric Bayesian Count Data Models

Leyre Estíbaliz Osuna Echavarría

Dissertation
at the Faculty of Mathematics, Computer Sciences and Statistics
Ludwig–Maximilians–University
Munich

1st June 2004

Leyre Estíbaliz Osuna Echavarría
Sevilla, Spain

1st Referee: Prof. Dr. Ludwig Fahrmeir
2sd Referee: PD Dr. Helmut Küchenhoff
3rd Referee: Prof. Dr. Claudia Czado
Rigorosum: 26th July 2004

Vorwort

Diese Arbeit wurde finanziell von der Deutschen Forschungsgemeinschaft gefördert, zum Teil durch den Sonderforschungsbereich 'Statistische Analyse diskreter Strukturen' (SFB 386) am Department für Statistik der Ludwig–Maximilians–Universität München und zum Teil durch ein Stipendium im Graduiertenkolleg 'Angewandte Algorithmische Mathematik' (GKAAM) am Zentrum Mathematik der Technische Universität München.

Als erstes bedanke ich mich von ganzem Herzen bei Prof. Dr. Ludwig Fahrmeir, der sich freundlicherweise angeboten hat, meine Promotion zu betreuen. Er hat mir auch die Chance gegeben, diese Arbeit in der lockeren und angenehmen Atmosphäre seines Lehrstuhls zu machen.

Mein Dank gilt auch Frau Gabriele Schnabel (immer so aufmerksam mit uns allen) und Frau Brigitte Maxa (die mir nicht nur in Sachen Uni sehr geholfen hat).

Ich möchte mich bei folgenden Kollegen bedanken: Andi Brezger, Andrea Hennerfeind, Alex Jerak, Stefan Lang, Günter Rasser, Volker Schmid und Renata Zambrzycka. Alle haben wirklich eine schwere Leistung gebracht. Es gab extrem produktive Diskussionsrunden, aus denen ich viel gelernt habe (und nicht nur über Statistik). Sie haben unendlich viel Geduld mit meinen Deutsch– oder Englisch–Fragen gehabt und waren immer hilfsbereit. Sie haben zahlreiche Aufmunterungsstunden hinter sich und mir stets gute Tipps gegeben. Andi und Stefan, danke auch für die Programmierhilfe und langen Aufklärungssitzungen in Sachen *BayesX*. Es hat echt Spaß gemacht, mit euch allen zu arbeiten!

Ich danke auch meinem Freund Torsten Loos für seine Geduld und Trost an den nicht immer fröhlichen Abenden nach stundenlangem Forschungsfrust. Und natürlich auch meinen Mitbewohnern im Geschwister–Scholl–Heim, die für die nötige Ablenkung am Abend und am Wochenende gesorgt haben. Ich will nicht meine Freunde Ana, María, David und Javi vergessen, die mich trotz der Distanz immer aufgemuntert haben.

Ich widme diese Arbeit meinen Eltern. Mit ihrer klugen Erziehungsart und vollem Ver-

trauen haben sie mir alle Türen für meine akademische Bildung geöffnet. Sie haben mich in jedem Schritt liebevoll unterstützt und dafür bin ich ihnen sehr dankbar.

Leyre Estíbaliz Osuna Echavarría

München, August 2004

Zusammenfassung

Zähldaten Modelle finden zahlreiche Anwendungen in der Praxis. Dennoch steht man oft einem oder mehreren der folgenden Probleme gegenüber, die von der Benutzung der Standard Poisson Regression abraten. Individuum spezifische unbeobachtete Heterogenität, verursacht durch nichtvorhandene Kovariablen, und/oder Exzess von Null–Beobachtungen könnten in den Daten festgestellt werden. Beide Verteilungsprobleme bewirken Abweichungen der Verteilung der Responsevariable von der klassischen Poisson Annahme. Andererseits wollen wir den Prädiktor vielleicht mit zeitlichen oder räumlichen Korrelationen und möglicherweise Effekten von stetigen Kovariablen oder Zeitskalen, vorhanden in den Daten, zusätzlich erweitern.

Hier werden semiparametrische Zähldaten Modelle entwickelt, die diese Probleme lösen können. Die Poisson Verteilung wird erweitert, um Überdispersion und/oder Exzess von Null–Beobachtungen aufzufassen. Zusätzlich werden entsprechende Komponenten in strukturierter additiver Form in den Prädiktor eingefügt. Die Modelle sind völlig Bayesianisch und Inferenz wird mit Hilfe von effizienten Markov Chain Monte Carlo (MCMC) Methoden durchgeführt. Mit Simulationsstudien wird untersucht, wie gut die verschiedenen Komponenten mit den vorliegenden Daten erkannt werden. Die Ansätze werden zum Schluß auf zwei Datensätze angewendet: auf Patentdaten und auf die Anzahl der Schäden eines großen Kfz-Datensatzes.

Abstract

Count data models have a large number of pratical applications. However there can be several problems which prevent the use of the standard Poisson regression. We may detect individual unobserved heterogeneity, caused by missing covariates, and/or excess

of zero observations in our data. Both distributional issues results in deviations of the response distribution from the classical Poisson assumption. We may in addition want to extend our predictor to model temporal or spatial correlation and possibly nonlinear effects of continuous covariates or time scales available in the data.

Here we study and develop semiparametric count data models which can solve these problems. We have extended the Poisson distribution to account for overdispersion and/or zero inflation. Additionally we have incorporated corresponding components in structured additive form into the predictor. The models are fully Bayesian and inference is carried out by computationally efficient MCMC techniques. In simulation studies, we investigate how well the different components can be identified with the data at hand. Finally, the approaches are applied to two data sets: to a patent data set and to a large data set of claim frequencies from car insurance.

Contents

1 Introduction **1**

1.1 Count data analysis 1

1.1.1 Log–linear Poisson Regression and extensions 2

1.1.2 Problems with classical count data regression 4

1.2 Overview of the thesis 7

2 Overdispersion **9**

2.1 Negative Binomial 10

2.2 Latent variables approach 12

2.2.1 Poisson–Gamma 13

2.2.2 Poisson–Inverse Gaussian 15

2.2.3 Poisson–Gaussian 16

2.3 Hierarchical centering 17

2.4 Résumé 19

3 Excess of Zero Counts **21**

3.1 Zero Inflated Models 23

3.1.1 Zero Inflated Poisson 26

3.1.2 Zero Inflated Negative Binomial 27

3.1.3 Zero Inflated-Poisson with latent variables 28

3.2 Hierarchical centering 30

3.3 Résumé 30

4 Priors and modeling of covariate effects **33**

4.1 Priors 34

4.2 Predictors 36

4.2.1 Fixed and random effects 36

4.2.2 Metrical covariates . . . 37
4.2.3 Spatial covariates . . . 41
4.3 Hierarchy of the models . . . 43
4.4 Résumé . . . 45

5 Posterior inference **47**
5.1 Posteriors . . . 48
5.1.1 Posteriors for groups A and C . . . 49
5.1.2 Posteriors for groups B and D . . . 50
5.1.3 Posterior for group E . . . 52
5.2 Full conditionals . . . 52
5.2.1 Predictor terms and their hyperparameters . . . 52
5.2.2 Model specific parameters . . . 54
5.3 Sampling Schemes . . . 61
5.3.1 Predictor terms and their hyperparameters . . . 61
5.3.2 Model specific parameters . . . 63
5.4 Algorithms . . . 68

6 Simulation studies **75**
6.1 Overdispersion . . . 76
6.1.1 Data simulation . . . 76
6.1.2 Results . . . 79
6.1.3 Résumé . . . 86
6.2 Zero inflation . . . 97
6.2.1 Data simulation . . . 97
6.2.2 Results . . . 99
6.2.3 Résumé . . . 115

7 Case studies **117**
7.1 Patent Data . . . 117
7.1.1 Data and model description . . . 119
7.1.2 Results . . . 123
7.2 Car insurance . . . 133
7.2.1 Data and model description . . . 135
7.2.2 Results . . . 139

8 Bayesian Count Data Regression with *BayesX*: A tutorial **153**
8.1 *BayesX* . . . 154
8.2 Getting started . . . 154
8.3 Dataset object . . . 156
8.4 Bayesreg object . . . 157
8.5 Post estimation commands and results . . . 161
8.6 Plots and Graph objects . . . 162
8.6.1 Post estimation plots . . . 163
8.6.2 Graph objects . . . 165

A Remarks on distributions **169**
A.1 Derivation of the Negative Binomial distribution . . . 169
A.2 General form of the inverse Gaussian distribution . . . 170
A.3 General form of the LogNormal distribution . . . 171
A.4 Zero inflation with latent variables . . . 171

B Calculation of IWLS weights **173**
B.1 NB . . . 175
B.1.1 Considered as exponential family member . . . 175
B.1.2 Direct method . . . 177
B.2 Poisson with latent variables . . . 177
B.3 ZIP . . . 178
B.4 ZIP with latent variables . . . 179
B.5 ZINB . . . 180

C MCMC **183**
C.1 Gibbs–sampling . . . 184
C.2 Metropolis–Hastings–sampling . . . 185
C.2.1 Construction of the transition kernel . . . 186
C.2.2 Justification of the transition kernel . . . 187
C.2.3 Some remarks . . . 188
C.2.4 Proposals . . . 191
C.3 Model comparison . . . 192

Chapter 1

Introduction

For count data, e.g. insurance claims frequencies, often a Poisson regression model is used. But the assumption of the Poisson distribution for the response variable is generally too restrictive in practice. Usually one has to deal with problems like overdispersion and zero inflation. In this thesis, several semiparametric approaches are introduced which take overdispersion and zero inflation of the data into account. We propose a flexible generalized regression approach, for which maximum likelihood estimation is not feasible as the likelihood does not belong to an exponential family and as the used predictor structures are very complex. We therefore define a Bayesian hierarchical model, which allows to estimate model parameters and all covariate effects simultaneously in an easy way. Because direct analyze of the posterior of the parameters will not be possible for all the models presented here, we use Markov Chain Monte Carlo (MCMC) methods for taking inference.

1.1 Count data analysis

In this section we shortly present the classical and well known Poisson regression model, which is the basis for the models presented in the next chapters. Additionally we give an overview of the problems that are related to it.

1.1.1 Log–linear Poisson Regression and extensions

Suppose we are given data $(y_i, r_i, \boldsymbol{z_i'})$, $i = 1, \ldots, n$, for each of the units under investigation. In detail, y_i is the response variable and stores the number of observed events for the i^{th} unit, $r_i > 0$ is a unit specific offset, for example time of exposure, and $\boldsymbol{z_i}$ is a column vector of covariates. Additionally suppose that the data y_i given the covariates $\boldsymbol{z_i'}$ are independently distributed for $i = 1, \ldots, n$.

To define a Generalized Linear Model (GLM) we have to make the following three assumptions. First, the response or target variable is assumed to have a distribution from the exponential family (EF). Second, given a covariate situation, we have to build a *linear predictor*, denoted by η_i for each observation. And third we have to choose a *link function* that relates the predictor η and the mean of the response variable, say μ, through $g(\mu) = \eta$.

Because of the count nature of the data presented above, the most appropriate distribution from the EF for the observation model is the Poisson distribution.

$$\begin{aligned} y_i|\boldsymbol{z_i} &\sim \text{Po}(\mu_i) \\ p(y_i|\boldsymbol{z_i}) &= \exp\{y_i \ln(\mu_i) - \mu_i - \ln(y_i!)\} \\ \mu_i &= r_i \lambda_i. \end{aligned} \tag{1.1}$$

The predictor is a linear combination of the observed covariates and some unknown parameters and is therefore called a linear predictor.

$$\eta_i = \alpha + \boldsymbol{z_i'}\boldsymbol{\beta}. \tag{1.2}$$

The mean of the response variable μ_i is related with the linear predictor through the so called link function. As μ_i has to be positive, an appropriate choice is the logarithmic function, so that we do not need further restrictions on the parameters $\boldsymbol{\beta}$. It is well known from the literature (see Fahrmeir and Tutz (2001)) that this is the natural link function for the Poisson distribution.

$$\mu_i = \exp(\log(r_i) + \eta_i). \tag{1.3}$$

The model described in this section is also called log–linear Poisson regression. To make inference, the vector $\hat{\boldsymbol{\beta}}$ that maximizes the whole likelihood of the model has to be found. In practice $\hat{\boldsymbol{\beta}}$ is the solution of the estimating equations obtained by differentiating the likelihood in terms of $\boldsymbol{\beta}$ and solving them to zero. These equations are nonlinear in $\boldsymbol{\beta}$ and iterative algorithms like Fisher scoring, Newton–Raphson or modified versions have to be applied in order to find a solution (Fahrmeir and Tutz, 2001).

Asymptotic theory related to GLMs is also applicable for the estimates $\hat{\boldsymbol{\beta}}$. These are consistent and efficient, provided that the mean and variance function of the model are correctly specified, even if the underlying data generating process is not Poisson distributed. Moreover in order to obtain consistency only the correct specification of the mean function is required. And the estimates are asymptotically normal distributed. This last property is very useful as it allows the construction of simple significance t–tests on the parameters (Fahrmeir and Tutz, 2001).

Suppose now that we are given data $(y_i, r_i, \boldsymbol{z_i'}, \boldsymbol{x_i'})$, $i = 1, \ldots, n$, for each of the units under investigation. This time $\boldsymbol{x_i}$ and $\boldsymbol{z_i}$ are column vectors of continuous and respectively categorical covariates. We suppose that the data sequences $(y_i, r_i, \boldsymbol{z_i'}, \boldsymbol{x_i'})$ are independently and identically distributed for $i = 1, \ldots, n$. The aim is to design a regression model including the information contained in the observed covariates in a more flexible way, that explains the variability in the y_i and is easy to interpret. Including nonlinear effects in the predictor to model the continuous covariates is a natural alternative to the fixed effects modeling. Generalized Additive Models (GAM) are a useful tool for this purpose.

GAMs extend standard regression in two ways. The first extension is, as in GLMs, given by the word 'generalized' and it refers to the distribution of the response variable. In classical regression it is restricted to be normally distributed. Here members of the EF are allowed as distributions for the target variable. The second extension is in the word 'additive' and concerns the terms of the predictor. In contrast to linear models or GLMs, the predictor is a sum of terms that may include linear terms and nonlinear functions of the covariates. To define a GAM we have to make the same three assumptions as

for a GLM. The distribution and the link function remain the same as given in (1.1) and (1.3). The predictor is a sum of linear combinations of the observed covariates categorical covariates z_i' and some unknown parameters, denoted by β, and some nonlinear smooth functions, denoted by $f^{(j)}$ of x_i'. This results in a semiparametric predictor

$$\eta_i = \alpha + z_i'\beta + \sum_j f^{(j)}(x_{ij}). \tag{1.4}$$

There are several approaches to estimate the smooth functions $f^{(j)}$, see Fahrmeir and Tutz (2001) and Lang (2004) for a review. In order to ensure that the estimated functions are smooth, in most of the approaches penalty terms and/or smoothing parameters are introduced for each function when maximizing the log–likelihood, to obtain the so called penalized log–likelihood.

Inference is made by maximizing the penalized log–likelihood through iterative methods, like e.g. Fisher scoring with backfitting. Presentation of these maximum likelihood estimation (MLE) methods is beyond the scope of this work. For detailed theory about GLMs and GAMs see for example McCullagh and Nelder (1989) and Hastie and Tibshirani (1990) respectively, or Fahrmeir and Tutz (2001).

1.1.2 Problems with classical count data regression

In practice, classical Poisson regression has two strong restrictions when working with practical applications. The first restriction is given by the predictor in the presence of complex covariate structures. Despite their flexibility there are data situations where even GAMs are not appropriate. For example, linear or one–dimensional smooth modeling are clearly not appropriate in the presence of some set of observed spatial covariates or group indicators, among the usual metrical and categorical variables. In the car insurance application of Section 7.2 we find metrical (driven kilometers per year) and dummy (garage) covariates, as well as group indicators (car classification) and spatial information (district). The aim is on including all these covariates in the predictor and on modeling

their effects simultaneously. In (1.5) we give an example for such an intended predictor.

$$\begin{aligned} \mu_i &= r_i \lambda_i \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \boldsymbol{z_i}'\boldsymbol{\beta} + \sum f^{(j)}(x_{ij}) + \rho_{g_i}. \end{aligned} \tag{1.5}$$

There we find linear effects represented by the term $\boldsymbol{z_i}'\boldsymbol{\beta}$. The functions $f^{(j)}$ are supposed to be one–dimensional smooth functions for one–dimensional covariates x_j and two-dimensional smooth curves if x_j are two-dimensional covariates, as for example spatial covariates are. The term ρ_{g_i} represents the group indicator effects. To overcome this problem, we present in this work Bayesian count data regression models. Bayesian regression allows for flexible predictor structures with the help of appropriate prior assumptions, according to the nature of the covariates (see Section 4.2 for a description of the priors used in this work).

The second restriction is the mean variance equality of the Poisson distribution and, in general, its lack of flexibility. Observed data sets tend to be overdispersed, which means, that the variance in the data exceeds the assumed variance of the Poisson distribution. As mentioned before, misspecification of the variance function does not affect the consistency of $\hat{\boldsymbol{\beta}}$, but leads to misspecification of the asymptotic covariance matrix of $\hat{\boldsymbol{\beta}}$. As a result we have loss of efficiency, confidence intervals or the usual tests for significance are no longer feasible. In the concrete case of overdispersion the variances of the estimates are set to be smaller as they actually are. Hence usual t–tests tend to be inflated which implies artificial statistical significance for the parameters (Cameron and Trivedi, 1998).

There are several possibilities for data to be overdispersed. We discuss them here in an informal way.

Positive contagion: This concept refers to the underlying count data generating process. Contagion denotes the dependence between the occurrence of successive events. We will talk about positive contagion when the observing of realizations of the

process increases the probability of new events. The interpretation in case of car accidents is that an individual causing an accident is more likely to produce another accident. See Cameron and Trivedi (1998) for more details.

Unobserved heterogeneity: We assume that the data generating process corresponds to a Poisson distribution. Some unobserved covariates are the source of the heterogeneity in the data and responsible for the observed overdispersion. This is a very intuitive explanation and easy to interpret when applying it to the data.

Excess of zero counts: Another departure from the Poisson distribution is an excess of observed zero counts (zero inflation) with respect to the distributional assumption. This also leads to overdispersion although the nature of the problem is not based on heterogeneity among the observations (Mullahy, 1997). We will introduce alternatives to overcome zero inflation problems in Chapter 3.

Now we briefly resume possible approaches to handle overdispersion. The approaches will be divided in three groups reflecting our belief of how overdispersion arises in the data. In the first group we renounce any distributional assumptions about the underlying distribution of the data. We choose some mean function, that relates the covariates with the mean of the data in our regression model and some variance function generally depending on the mean and a dispersion parameter.

Quasilikelihood approaches avoid making assumptions about the underlying generating process of the data (see McCullagh and Nelder (1989), Brockman and Wright (1992) and Renshaw (1994) for theory and applications to car insurance data). The motivation for these methods is that only a correct specification of the mean function is needed to guaranty consistency of the estimates in maximum likelihood estimation for exponential families. The name Poisson Quasilikelihood estimation means that the parameter estimates are defined by the first order conditions of a Poisson maximum likelihood regression but the data generating process does not need to be Poisson distributed. In practice, the mean function is chosen to be similar to the mean of the Poisson regression

in (2.1). The variance function is the product of some functional of μ and a dispersion parameter ϕ. The estimating equations for the parameters in $\boldsymbol{\eta}$ do not depend on the dispersion parameter ϕ. Hence it is handled as a nuisance parameter. Because its estimate is based on $\hat{\boldsymbol{\eta}}$, it can only be calculated at the end and so the estimation of the parameters is not simultaneous. This approach is described in Cameron and Trivedi (1998) for different variance functions. They also discuss appropriate estimators for the dispersion parameter depending on the form of the variance functions.

In the second group we suppose that the data does not follow a Poisson distribution at all. A search for alternative and more flexible count distributions that relax the strong variance assumption of the Poisson distribution is therefore necessary. As there are a multitude of ways to achieve this we treat the most common: the negative binomial. For more examples, we refer to Cameron and Trivedi (1998) or Johnson and Kotz (1969). If we have information in our data set about occurrence times of events, we could generalize the underlying waiting time distribution assumption for the Poisson and thus obtain less restrictive associated count data distributions (Winkelmann, 1995). Poisson mixtures are also an interesting alternative for a more flexible analysis of the data. We refer to Viallefont, Richardson and Green (2002), Deb and Trivedi (1997) and Aitkin (1996).

And in the third group we assume that overdispersion arises from covariates not available in the data. This approach will be presented in the next chapter in a general form as well as in three concrete cases.

1.2 Overview of the thesis

This work is structured as follows. We begin with a theoretical overview about more flexible count data distributions such as the Poisson. In Chapter 2 we present distributions that are able to account for overdispersion in the data, Chapter 3 covers distributions that are able to model zero inflation in the data.

Priors for the definition of a count data regression model with flexible predictor structures

in a Bayesian framework are given in Chapter 4. Chapter 5 describes the posteriors of the different presented regression models based on the prior assumptions of the Chapters 2 to 4. It also shows the implemented algorithms using Markov Chain Monte Carlo (MCMC) methods for estimation. A short overview of the theory on MCMC is given in Appendix C.

Chapter 6 resumes the results of the simulation studies for testing the performance of overdispersion and zero inflation models. In Chapter 7 the developed models are applied on two real data sets. Firstly, a patent data set, without spatial information, but with binary and metrical covariates. The aim is on modeling the number of forward citations for a patent depending on the given covariates. Secondly, a massive car insurance data set, with a lot of covariates containing information about the policyholders. The aim is on modeling the number of expected claims for an insured depending on the observed information.

The approaches presented in Chapters 2 and 3 have been implemented in the statistical software *BayesX*. The analyzes of Chapter 6 as well as Chapter 7 are also carried out with this program. *BayesX* is available at `http://www.stat.uni-muenchen.de/~lang/`. In Chapter 8 we present a tutorial based on the patent data to exemplify the using of *BayesX*.

Chapter 2

Overdispersion

We recall the definition of overdispersion given in the previous chapter. Given a distributional assumption in a regression model, we find overdispersion if the observed variance of the data is greater than the variance supposed by the model. As this work deals with count data, our basic model will be the classical Poisson regression, presented in Section 1.1.1. Overdispersion occurs if the variance in the data is greater than the mean. Possible sources of overdispersion in our data, possible approaches to solve this problem, and consequences of ignoring overdispersion in our model were presented in an informal way Section 1.1.2.

Recall the notational agreements of the last chapter. We are given data $(y_i, r_i, \boldsymbol{z_i'}, \boldsymbol{x_i'})$, $i = 1, \ldots, n$, with y_i the number of observed events for the i^{th} unit, $r_i > 0$ is a unit specific offset, and $\boldsymbol{z_i}$ and $\boldsymbol{x_i}$ are column vectors of categorical and continuous covariates respectively.

For later use we recall the mean structure given in (1.5) in Subsection 1.1.2:

$$\begin{aligned} \mu_i &= r_i \lambda_i \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \boldsymbol{z_i}' \boldsymbol{\beta} + \sum f^{(j)}(x_{ij}) + \rho_{g_i}. \end{aligned} \tag{2.1}$$

The scope of this chapter is to present two main approaches to account for overdispersion in the data. The first one is to substitute the Poisson by a Negative Binomial distribution, which has a more flexible variance function. This approach is presented in Section 2.1. The second approach introduces latent variables in the Poisson regression in a multiplicative way. More details about this can be found in Section 2.2. And finally, in Section 2.3 we present the hierarchical centered versions of the models of Subsections 2.2.1 and 2.2.2.

2.1 Negative Binomial

The Negative Binomial (NB) distribution has two parameters, μ_i and δ, both with strictly positive real values. We will write

$$y_i|\eta_i, \delta \quad \sim \quad NB(\mu_i, \delta). \tag{2.2}$$

Note, that we will allow μ_i to vary with the observations (denoted by the subindex i) as in (2.1) but δ will be an overall parameter in the model.

The density, mean and variance of a NB distribution are given by

$$\begin{aligned} P(y_i|\eta_i, \delta) &= \frac{\Gamma(y_i+\delta)}{\Gamma(y_i+1)\Gamma(\delta)} \left(\frac{\mu_i}{\mu_i+\delta}\right)^{y_i} \left(\frac{\delta}{\mu_i+\delta}\right)^{\delta} \\ E(y_i|\eta_i, \delta) &= \mu_i \\ V(y_i|\eta_i, \delta) &= \mu_i + \frac{\mu_i^2}{\delta} \end{aligned} \tag{2.3}$$

for all $y_i \in \boldsymbol{N} \cup \{0\}$. Comparing the first two moments with those of the Poisson distribution $Po(\mu_i)$, we see that the mean is equal and the difference appears in the second moment. As the variance is greater than the mean this distribution is able to account for overdispersion in the data with respect to the classical Poisson assumption.

The joint likelihood of the model is the product of the individual likelihoods of the units under investigation and is proportional to:

$$l(\boldsymbol{y}|\boldsymbol{\eta}, \delta) \quad = \quad \prod_{i=1}^{n} l(y_i|\eta_i, \delta)$$

$$
\begin{aligned}
\propto \quad & \exp\Big\{ \sum_{i=1}^{n} \Big(\log(\Gamma(y_i+\delta)) - \log(\Gamma(\delta)) \\
& + y_i \log(\mu_i) - y_i \log(\delta+\mu_i) + \delta \log(\delta) - \delta \log(\delta+\mu_i) \Big) \Big\} \\
= \quad & \exp\Big\{ n\Big(- \log(\Gamma(\delta)) + \delta \log(\delta) \Big) \\
& + \sum_{i=1}^{n} \Big(\log(\Gamma(y_i+\delta)) + y_i \log(\mu_i) - (y_i+\delta) \log(\delta+\mu_i) \Big) \Big\}.
\end{aligned} \tag{2.4}
$$

The terms that only depend on the data can be omited because the likelihood will only appear in quotients in the estimation algorithm of the model (see Appendix C).

The NB distribution belongs to the exponential family as long as δ is known. For a proof, see Appendix B, where we rewrite the density given in (2.3) by assuming that δ is known:

$$
\begin{aligned}
p(y_i|\mu_i,\delta) &= \frac{\Gamma(y_i+\delta)}{\Gamma(\delta)\Gamma(y_i+1)} \left(\frac{\mu_i}{\delta+\mu_i}\right)^{y_i} \left(\frac{\delta}{\delta+\mu_i}\right)^{\delta} \\
&= \exp\left\{c(y_i,\delta) + y_i\theta - b(\theta)\right\}.
\end{aligned}
$$

Here, $\theta = \log(\frac{\mu_i}{\mu_i+\delta})$ is the natural parameter, $c(.)$ depends only on δ (which is assumed to be known!) and the data, and $b(.)$ is a function that depends only on the natural parameter. If the δ–parameter is known, estimation can easily be done by maximizing the likelihood over η. But if δ is unknown, which will be the standard situation, there is no possibility to rewrite (2.3) and find such functions $b(.)$ and $c(.)$ and we are not in an exponential family framework anymore. Estimation can now be based on the maximization of the likelihood over both parameters η and δ as proposed for example by Cameron and Trivedi (1998). We will take advantage of this property of the NB distribution when implementing the regression models in the next chapters.

The NB distribution can be derived in several ways. One possibility is to consider it as the marginal distribution of the response variable of a Poisson Gamma model, as we will demonstrate in Subsection 2.2.1. Without further explanations, we annotate here that the NB distribution can also arise in the context of positive contagion or modeling of waiting times (Cameron and Trivedi, 1998; Winkelmann, 1995).

2.2 Latent variables approach

In the following, we assume that overdispersion in the model is caused by heterogeneity in the data due to unobserved covariates. The natural solution is to maintain the Poisson distribution in the observation model and to introduce independent and identically distributed unit specific latent variables ν_i. They enter the model as multiplicative random effects and should capture the effect of the unobserved covariates and make the model more flexible to account for overdispersion. In this section we give a general representation and show first consequences of this model. More details are given in the following subsections.

In a first short representation, we will write

$$\begin{aligned} y_i|\eta_i, \nu_i &\sim Po(\nu_i \mu_i) \\ \nu_i|\cdot &\sim D(\cdot) \quad i = 1, \ldots, n, \end{aligned} \tag{2.5}$$

where $D(\cdot)$ may depend on some parameters. The latent variables have to fulfill two very intuitive conditions. First, $\nu_i > 0$ ensures that the mean of y_i is properly defined. Secondly, if we want to avoid problems with the identifiability of the intercept, we should impose $E_D(\nu_i|\cdot) = 1$.

With these restrictions we can calculate the first two moments of the marginal distribution of y_i. The mean is given by

$$\begin{aligned} E(y_i|\eta_i, \cdot) &= E_D\left(E(y_i|\nu_i, \eta_i)|\cdot\right) \\ &= \mu_i E_D(\nu_i|\cdot) \\ &= \mu_i. \end{aligned} \tag{2.6}$$

There are no changes in the mean structure compared to (2.1) in the Poisson regression. The marginal variance is calculated as follows:

$$V(y_i|\eta_i, \cdot) = E_D\left(V(y_i|\nu_i, \eta_i)|\cdot\right) + V_D\left(E(y_i|\nu_i, \eta_i)|\cdot\right)$$

$$\begin{aligned} &= \mu_i E_D(\nu_i|\cdot) + \mu_i^2 V_D(\nu_i|\cdot) \\ &= \mu_i + \mu_i^2 V_D(\nu_i|\cdot). \end{aligned} \tag{2.7}$$

The variance is a second order polynomial in the mean μ_i whereby $\mu_i^2 V_D(\nu_i|\cdot)$ is always strictly positive. This implies $V(y_i|\cdot) > E(y_i|\cdot)$, so that overdispersion can be explained through this new formulation.

Note that if $V_D(\nu_i|\cdot)$ goes to zero, the distribution of ν_i is degenerated into one and hence the marginal distribution of y_i is the Poisson distribution once again. The variance of ν_i gives us a 'relative measure' of the amount of overdispersion in the data. Relative, because the marginal variance also depends on μ_i^2.

In the next subsections we present four distributions that are derived from three candidate distributions $D(\cdot)$ for the ν_i. The most commonly used is the Poisson Gamma distribution, which also induces the negative binomial distribution. Poisson Inverse Gaussian and Poisson LogNormal are more unusual but not less interesting alternatives. Kaas and Hesselager (1995) have compared the tails of the Gamma, Inverse Gaussian and LogNormal distributions with equal means and variances and obtained the order given above for increasing tails. This order can be transferred to the corresponding mixtures with Poisson distribution, resulting in Poisson–Gamma, Poisson–Inverse Gauss, and Poisson–LogNormal with increasing tails.

2.2.1 Poisson–Gamma

The Poisson–Gamma (POGA) model arises as the mixture of a Poisson and a Gamma distribution, i.e.

$$y_i|\eta_i, \nu_i \sim Po(\nu_i\, \mu_i) \tag{2.8}$$

$$\nu_i|\delta \sim G(\delta, \delta), \tag{2.9}$$

where (2.8) is the distributional assumption for the response variable in the POGA model. The $y_i, i = 1, \ldots, n$ are mutually independent and μ_i is defined as in (2.1). The conditional

density function, mean and variance of $y_i, i = 1, \ldots, n$ for all i are given by:

$$\begin{aligned} P(y_i|\eta_i, \nu_i) &= \frac{\exp(-\nu_i\mu_i)\,(\nu_i\mu_i)^{y_i}}{y_i!} \quad \text{for} \quad y_i \in I\!N \cup \{0\} \\ E(y_i|\eta_i, \nu_i) &= V(y_i|\eta_i, \nu_i) \;=\; \nu_i\,\mu_i. \end{aligned} \tag{2.10}$$

The joint likelihood of the data is the product of the individual likelihoods. Due to the same reasons as in the NB model we can again consider the whole likelihood up to a proportionality constant.

$$\begin{aligned} l(\boldsymbol{y}|\boldsymbol{\eta}, \boldsymbol{\nu}) &= \prod_{i=1}^{n} l(y_i|\eta_i, \nu_i) \\ &\propto \exp\left\{ \sum_{i=1}^{n} \Big(-\nu_i\mu_i + y_i \log(\nu_i\mu_i) \Big) \right\} \end{aligned} \tag{2.11}$$

In (2.9) we have specified a Gamma distribution as distribution for the ν_i terms. As the mean has to be one, the Gamma distribution is no longer a two–parameter distribution. Only the parameter δ is free and acts as a dispersion parameter, since it explains the variance of the ν_i's. It follows from (2.9):

$$\begin{aligned} g(\nu_i|\delta) &= \frac{\delta^\delta}{\Gamma(\delta)} \nu_i^{\delta-1} \exp(-\delta\,\nu_i) \\ E(\nu_i|\delta) &= 1 \\ V(\nu_i|\delta) &= \frac{1}{\delta}. \end{aligned}$$

For δ going to infinity, the variance of ν_i goes to zero, and we approximate the Poisson distribution. The marginal moments of the response variable are easily calculated by substituting $V(\nu_i|\delta) = \frac{1}{\delta}$ in (2.7):

$$\begin{aligned} E(y_i|\eta_i, \delta) &= \mu_i \\ V(y_i|\eta_i, \delta) &= \mu_i + \frac{\mu_i^2}{\delta}. \end{aligned} \tag{2.12}$$

Note that the mean and the variance of the NB distribution are equal to those obtained here for the marginal distribution of the POGA model. We remind the reader, that the NB model can be derived from a POGA model by marginalizing the distribution of the

response variable with respect to the multiplicative random effects. For this purpose we calculate the expectation of (2.10) respecting ν_i as is shown in Appendix A.1. This explains, why the marginal moments of the response variable of the POGA model and the moments of the NB model are identical.

2.2.2 Poisson–Inverse Gaussian

Following the same idea as with the POGA model, we now choose another appropriate distribution for the ν_i terms. This time, they are supposed to be Inverse Gaussian distributed. This distribution has heavier tails than the Gamma distribution, which may be of advantage in some applications. More details on the Inverse Gaussian distribution are given in Appendix A.2. The POIG model is given by

$$\begin{aligned} y_i|\eta_i, \nu_i &\sim Po(\nu_i \mu_i) && (2.13)\\ \nu_i|\delta &\sim IGaussian(1, \delta), && (2.14) \end{aligned}$$

with μ_i defined as in (2.1). The observational assumption in (2.13) is equal to the assumption for the POGA model given by (2.8) and therefore (2.10) is also valid here. Consequently the likelihood $l(\boldsymbol{y}|\boldsymbol{\eta}, \boldsymbol{\nu})$ is proportional to (2.11).

The mean of the prior distribution of the ν_i's has to be one to ensure that the intercept remains identifiable. With this restriction the two parameter Inverse Gaussian distribution has then only one parameter, say δ, which is, similarly as in the POGA model, a sort of scale parameter since it controls the variance of the distribution. It follows from (2.14):

$$\begin{aligned} g(\nu_i|\delta) &= \sqrt{\frac{\delta}{2\pi\nu_i^3}} \exp\left(-\frac{\delta(\nu_i-1)^2}{2\nu_i}\right)\\ E(\nu_i|\delta) &= 1\\ V(\nu_i|\delta) &= \frac{1}{\delta}. \end{aligned}$$

Note that the first two moments of the Gamma and Inverse Gaussian distribution are equal, if the mean is supposed to be one, as is the case here. This implies that also the first

two moments of the response variable conditioned only on the δ parameter are identical with those of the POGA model and therefore also with those of the NB model (see (2.3)).

2.2.3 Poisson–Gaussian

Finally we connect the standard Poisson regression with Gaussian latent variables. For this purpose we assume a LogNormal distribution for the ν_i. Kaas and Hesselager (1995) have shown, that this distribution is heavier in its tails than the Gamma and the Inverse Gaussian are for equal mean and variance. In Section A.3 we present the general form of a LogNormal distribution. Here we have to adjust the parameters to obtain $E(\nu_i|\delta) = 1$ and $V(\nu_i|\delta) = \frac{1}{\delta}$ in order to be able to compare the results with those of the other mixtures. Note that due to the mean restriction, we work with a one parameter LogNormal. Looking at A.3, it is easy to show, that for

$$\nu_i|\delta \quad \sim \quad LogN\left(-0.5\log\left(1+\frac{1}{\delta}\right), \log\left(1+\frac{1}{\delta}\right)\right) \tag{2.15}$$

we get the desired mean and variance assumptions given above. We rewrite the parameter assumption given in (2.5)

$$\begin{aligned} \nu_i\,\mu_i &= r_i\nu_i\exp(\eta_i) \\ &= r_i\exp(\eta_i+\log(\nu_i)) \\ &= r_i\exp(\eta_i+\kappa_i) \end{aligned} \tag{2.16}$$

We can now exploit the fact that $\kappa_i = \log(\nu_i)$ follows a Gaussian distribution, if ν_i is LogNormal distributed, as given in (2.15), i.e.

$$\kappa_i \quad \sim \quad N\left(-0.5\,\tau_\kappa^2, \tau_\kappa^2\right) \tag{2.17}$$

with $\tau_\kappa^2 = \log\left(1+\frac{1}{\delta}\right)$. Thus, we can convert a Poisson–LogNormal model with multiplicative random effects into a Poisson–Gaussian one with additive random effects. As we will see in the next chapter, the common prior assumption for additive random effects is a normal distribution with mean zero. The difference to (2.17) does not represent

a problem for the further regression. The mean is constant for all the κ_i and thus will be captured by the intercept α in the implementation. The following table resumes both model formulations:

Poisson–Gaussian			Poisson–LogNormal		
$y_i \vert \eta_i, \kappa_i$	$\sim$	$Po(\mu_i)$	$y_i \vert \eta_i, \nu_i$	$\sim$	$Po(\nu_i \, \mu_i)$
μ_i	$=$	$r_i \, \lambda_i$	μ_i	$=$	$r_i \, \lambda_i$
λ_i	$=$	$\exp(\eta_i + \kappa_i)$	λ_i	$=$	$\exp(\eta_i)$
$\kappa_i \vert \delta$	$\sim$	as given in (2.17)	$\nu_i \vert \delta$	$\sim$	as given in (2.15)
$E(\kappa_i \vert \delta)$	$=$	$-0.5\tau_\kappa^2$	$E(\nu_i \vert \delta)$	$=$	1
$V(\kappa_i \vert \delta)$	$=$	τ_κ^2	$V(\nu_i \vert \delta)$	$=$	$\frac{1}{\delta}$

for $i = 1, \ldots, n$. We can now interpret the model in two ways. First as a standard Poisson regression with a random effect for each unit under investigation, that is additive in the predictor. And secondly as a latent variables approach, where the terms that multiply the μ parameter of the Poisson distribution are LogNormal distributed.

Now it is clear why this model works as a connection between standard models with additive random effects in the predictor and our multiplicative models presented here. Both model formulations are equivalent. Therefore we decided to work with the first one, which is based on standard methods and already implemented in the program *BayesX*.

The main observational assumption does not have to be changed, that means, the response variable remains Poisson distributed. Due to the conditional independence of the observations given the parameters, the likelihood of the whole sample $l(\boldsymbol{y} \vert \boldsymbol{\eta}, \boldsymbol{\kappa})$ can be calculated as the product of the individual likelihoods $l(y_i \vert \eta_i, \kappa_i)$, as given in (2.11).

2.3 Hierarchical centering

In some applications a new parameterization of the latent variable models may work better than the one explained above. The idea of hierarchical centering is to omit the

intercept in the predictor and to let the mean of the multiplicative effects account for it in the model. Formally this means:

$$\begin{aligned} \nu_i \, \mu_i &= r_i \, \nu_i \, \exp\left(\alpha + f(x_i, z_i)\right) \\ &= r_i \, \nu_i \exp(\alpha) \exp(f(x_i, z_i)) \\ &= r_i \, \tilde{\nu}_i \exp(\tilde{\eta}_i) \\ &= \tilde{\nu}_i \tilde{\mu}_i \end{aligned}$$

with $\tilde{\nu}_i = \nu_i \exp(\alpha)$, $\tilde{\eta}_i = f(x_i, z_i)$ and $\tilde{\mu}_i = r_i \, \exp(\tilde{\eta}_i)$. The new model formulation is equivalent to the old one given in (2.5) in the distributional assumption for the response, but differs in the prior distribution of the ν_i:

$$\begin{aligned} y_i | \tilde{\nu}_i, \tilde{\eta}_i &\sim Po(\tilde{\nu}_i \, \tilde{\mu}_i) \\ \Big[&\sim Po(\nu_i \, \mu_i) \Big] \\ \tilde{\nu}_i | \exp(\alpha), \cdot &\sim \tilde{D}\big(\exp(\alpha), \cdot\big). \end{aligned} \tag{2.18}$$

The first two moments of the marginal distribution of the response variable are

$$\begin{aligned} E(y_i | \tilde{\eta}_i, \exp(\alpha), \cdot) &= E_{\tilde{D}} \left(E(y_i | \tilde{\nu}_i, \tilde{\eta}_i) | \exp(\alpha), \cdot \right) \\ &= E_{\tilde{D}} \left(\tilde{\nu}_i \, \tilde{\eta}_i | \exp(\alpha), \cdot \right) \\ &= \tilde{\eta}_i \, \exp(\alpha) \\ &= \mu_i \end{aligned}$$

and

$$\begin{aligned} V(y_i | \tilde{\eta}_i, \exp(\alpha), \cdot) &= E_{\tilde{D}} \left(V(y_i | \tilde{\nu}_i, \tilde{\eta}_i) | \exp(\alpha), \cdot \right) + V_{\tilde{D}} \left(E(y_i | \tilde{\nu}_i, \tilde{\eta}_i) | \exp(\alpha), \cdot \right) \\ &= \tilde{\mu}_i E_{\tilde{D}}(\tilde{\nu}_i | \exp(\alpha), \cdot) + \tilde{\mu}_i^2 V_{\tilde{D}}(\tilde{\nu}_i | \exp(\alpha), \cdot) \\ &= \tilde{\mu}_i \exp(\alpha) + \tilde{\mu}_i^2 \exp(2\,\alpha) V_D(\nu_i | \cdot) \\ &= \mu_i + \mu_i^2 V_D(\nu_i | \cdot). \end{aligned}$$

The reparameterization does not affect the marginal distribution of the response variable.

We have implemented this reparameterization for the POGA and the POIG model, which yields their hierarchical variations, the POGAH and POIGH model respectively. If $\nu_i \sim G(\delta, \delta)$, then it is well known that

$$\tilde{\nu}_i|\delta, \exp(\alpha) \sim G\left(\delta, \frac{\delta}{\exp(\alpha)}\right). \tag{2.19}$$

If $\nu_i \sim IGaussian(1, \delta)$, we obtain

$$\tilde{\nu}_i|\delta, \exp(\alpha) \sim IGaussian\big(\exp(\alpha), \exp(\alpha)\delta\big), \tag{2.20}$$

as is shown in Section A.2,

2.4 Résumé

In this chapter we have given an overview about overdispersion and developed the latent variable methods in more detail. Some comments on the models will be made here.

The first question is why do we use NB and POGA models, although we know that these models are equivalent. Actually we should take advantage of the fact that there exists a closed form of the marginal distribution of the POGA model, namely, the NB model. The theoretical advantage of the NB model is that the number of parameters to estimate in the model is much smaller as for the POGA model. Remember that we have an extra parameter per unit in the POGA model, which means n further parameters to estimate compared to the NB model. And with a massive data set, like the car insurance data set, it is an important matter to reduce computating time and resources. Finally, as we can obtain the NB distribution in several ways, we can also justify its application and interpret it in several ways. With the car insurance data set for example, using NB as the distribution arising from positive contagion will be interpreted as providing increase for the probability of producing another accident after having had one. In case we consider the NB as marginal distribution for y_i proceeding from a POGA model, we will interpret its use as accounting for missing information in the model. On the other side, the POGA

model is favorable in two aspects. First, the computations that we need to calculate for the likelihood of a NB model are quite intensive, due to the Gamma functions that are involved. And secondly, it would be a nice idea to exploit the information obtained from the estimates $\hat{v}_i$ to make further analysis of the data, and try, for example, to deduce which unobserved covariates are responsible for the overdispersion, or to group the observations in clusters in terms of the $\hat{v}_i$. We can even use the $\hat{v}_i$ in model assessment and deduce if the preliminary assumptions are satisfied. We will revisit this problem in Section 7.2.

For the POIG model we have no closed form of the marginal probabilities of y_i, but we can calculate them recursively (Dean, Lawless and Willmot, 1989). Of course it is a time–consuming process in case the observed counts are shifted to the right, that means in case they take large values.

For the POIG and the POLN models we do not have the alternative of working with the marginal model. It would be interesting to compare the behavior of the three mixture models and to analyze through simulation studies how robust the models are in case the data are not distributed as supposed.

Excess of zeros may be a consequence of positive contagion (see Zorn (1998)) and therefore appear as overdispersion in the data. In Section A.4 we prove that modeling overdispersion with latent variables also accounts for excess of zeros in the model. This proof is based on the one given in Mullahy (1997).

A final remark: The hierarchical versions POGAH and POIGH are not new models on their own, but it may be possible to improve the mixing of the chains for some parameters (specially intercept and multiplicative random effects) through the new parametrization.

More information about count data models can be found in: Winkelmann and Zimmermann (1995); Hinde and Demétrio (1998); Podlich, Faddy and Smyth (1999); Alexander, Moyeed and Stander (2000); Thurston, Wand and Wiencke (2000); Sutradhar and Jowaheer (2001); Karlis (2001) and Booth, Casella, Friedl and Hobert (2003).

Chapter 3

Excess of Zero Counts

We talk about excess of zero counts if the number of observed zero counts exceeds the number of zero counts expected by the model. Recall the notation of the last two chapters. We are given data $(y_i, r_i, \boldsymbol{z_i'}, \boldsymbol{x_i'})$, $i = 1, \ldots, n$, with y_i number of observed events for the i^{th} unit, $r_i > 0$ is a unit specific offset, and $\boldsymbol{z_i}$ and $\boldsymbol{x_i}$ are column vectors of categorical and continuous covariates respectively. We also keep the same mean structure of the last chapter given in (2.1).

$$\begin{aligned} \mu_i &= r_i \lambda_i \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \boldsymbol{z_i}' \boldsymbol{\beta} + \sum f^{(j)}(x_{ij}) + \rho_{g_i}. \end{aligned}$$

Juts as with the overdispersion case, the underlying factors that cause excess of zeros in the data can be of very different nature. In the following we describe the most usual sources in praxis.

Unobserved heterogeneity: As is informally shown in Section A.4, unobserved heterogeneity in the model implies excess of observed zero counts. A more formal proof of this assertion is given by Shaked's theorem (Shaked, 1980). Mullahy (1997) ex-

amines the implications of unobserved heterogeneity for the probability structure of count data models.

Selectivity: The observed outcomes are produced by two latent processes, a count data process and a selection process, generally independent from each other. The selection process modifies the count data process in such a way, that we can not directly observe it.

Unobserved heterogeneity as the origin of the excess of zero counts has already been discussed in the last chapter. In the following we will briefly present some possibilities to define how the selection process affects the underlying count data process under the assumption of selectivity.

The first approach reflects the belief that only the selection process determines whether we observe a zero outcome or not, independently from the underlying count data process. These models are called *hurdle models* in the literature (Winkelmann, 1998; Gurmu, 1997; Ridout, Demétrio and Hinde, 1998; Zorn, 1998). In this case the selection process ω_i can be modeled as a $0/1$ variable. If $\omega_i = 0$, then we have a zero outcome. Otherwise we observed a strictly positive count that can be modeled as a Poisson or Negative Binomial truncated at zero, for example. Note also data with too few zeros can be analyzed with hurdle models, because the probability of a zero outcome is given only by the probability of the binary variable ω_i to be 0, without further restrictions.

In contrast to hurdle models zero outcomes are not only determined by the selection process in the next approach. If $\omega_i = 0$, then we have a zero outcome. Otherwise, if $\omega_i = 1$, our outcome comes directly from the underlying count data distribution. That means, we have two types of zero outcomes: those generated by the selection variable and those generated by the count data distribution. These models are called *zero inflated* (or *with zeros*). We are going to develop them later in this chapter.

Finally, *underreporting* can also be seen as a case of selectivity (Winkelmann, 1998). The idea is to assume that not all of the produced outcomes are reported. The underlying

count data process gives the number of real occurrences y_i^{under}. The selection process is now a vector $\omega_i = (\omega_{ij})_j$ of length y_i^{under} with $0/1$ entries, where 0 indicates 'not reported' and 1, 'reported'. The selection process is assumed to be independent from the counts. Thus, the observed number of counts is given by $y_i = \sum_{j=1}^{y_i^{under}} \omega_{ij}$. The resulting marginal distribution of the y_i is a mixture of a binomial distribution and the underlying count data process.

In this chapter, only zero inflated models will be considered. We will first describe the idea in more detail and then apply the model to several underlying count data distributions.

3.1 Zero Inflated Models

As said before, in many count data applications we observe excess of zero counts. To overcome this problem zero inflated models (ZIM) introduce a latent binary variable that 'inflates' the number of zero counts expected by the observational assumption. This can be interpreted as a two step data generating process. Each observation in our data set is the result of the product of two independent processes: an underlying count data generating process and a $0/1$ 'selection' process, say y_i^{under} and ω_i respectively.

$$\begin{aligned} y_i &= \omega_i y_i^{under} \\ \omega_i &\sim Bern(1-\theta). \end{aligned} \tag{3.1}$$

We have called ω_i 'selection' variable for the purpose of interpretation. It classifies the units in our data set into those with $\omega_i = 0$, that can not produce outcomes, and those with $\omega_i = 1$, that are able to produce outcomes, but do not necessarily have to. The response variables y_i are the outcomes that we observe for each unit under investigation. With their help we can partially win information about the underlying count process and about the classification variable.

The conditional distribution of the response variable is given by the following expression,

where '$\cdot$' is placed to indicate that the count data distribution followed by y_i^{under} may (and will!) depend on further parameters.

$$P(y_i|\omega_i, \cdot) = \begin{cases} P(y_i^{under} = y_i|\cdot) & \omega_i = 1, \forall y_i \\ 0 & \omega_i = 0,\ y_i > 0 \\ 1 & \omega_i = 0,\ y_i = 0. \end{cases} \tag{3.2}$$

With the help of the indicator function $I(x) = 0$ for $x = 0$ and $I(x) = 1$ else, we can rewrite (3.2) in a more compact form as

$$P(y_i|\omega_i, \cdot) = P(y_i^{under} = y_i|\cdot)I(\omega_i) + \big(1 - I(\omega_i)\big)\big(1 - I(y_i)\big). \tag{3.3}$$

A problem in the context of car insurance is the interpretation of this conditional distribution. We could implement an algorithm similar to the algorithm for POGA, POIG or POLN to estimate the unobserved ω_i for each unit in the data set. If $\hat{\omega}_i = 0$, the i^{th} unit can not produce any outcomes, and this, applied to our car insurance data set, would mean that some policy holders can not produce any accidents. This result would be very difficult to interpret. On the other hand, we can calculate the marginal distribution of y_i with respect to the selection variables ω_i. The resulting distribution is much easier to interpret, because we get modified probabilities of the underlying count data distribution for y_i^{under} instead of results on some latent variables that may make no sense in this context. Note that in any other data set the information delivered by the ω_i estimates may be of great relevance and easy to interprete, but this is not the case here. Therefore we concentrate on the marginal distribution of $y_i|\theta, \cdot$.

As said above, the marginal distribution of the observed counts is of prime interest now. We can calculate it by combining (3.3) with the prior information given in (3.1)

$$\begin{aligned} P(y_i|\,\theta, \cdot) &= P(\omega_i = 0|\,\theta)\left\{P(y_i^{under} = y_i|\,\cdot)I(0) + \big(1 - I(0)\big)\big(1 - I(y_i)\big)\right\} \\ &+ P(\omega_i = 1|\,\theta)\left\{P(y_i^{under} = y_i|\,\cdot)I(1) + \big(1 - I(1)\big)\big(1 - I(y_i)\big)\right\} \\ &= \theta\big(1 - I(y_i)\big) + (1 - \theta)P(y_i^{under} = y_i|\,\cdot) \end{aligned} \tag{3.4}$$

We are going to analyze how zero inflation affects the first two moments of the underlying count data distribution. For the marginal mean of the response variable, we get

$$\begin{aligned} E(y_i|\,\theta,\cdot) &= E_\omega\big(E(\omega_i y_i^{under}|\omega_i,\cdot)\big) \\ &= E_\omega\big(\omega_i E(y_i^{under}|\omega_i,\cdot)\big) \\ &= E_\omega\big(\omega_i\big)E(y_i^{under}|\,\cdot) \\ &= (1-\theta)E(y_i^{under}|\,\cdot). \end{aligned} \tag{3.5}$$

This is an important result, since it shows that ignoring zero inflation in the model will lead to inconsistent estimators for the parameters, independently from the underlying count data distribution. Some more comments on this result are given in Section 3.3.

For the marginal variance we get

$$\begin{aligned} V(y_i|\,\theta,\cdot) &= E_\omega\big(V(\omega_i y_i^{under}|\omega_i,\cdot)\big) + V_\omega\big(E(\omega_i y_i^{under}|\omega_i,\cdot)\big) \\ &= E_\omega\big(\omega_i^2 V(y_i^{under}|\omega_i,\cdot)\big) + V_\omega\big(\omega_i E(y_i^{under}|\omega_i,\cdot)\big) \\ &= (V_\omega(\omega_i) + E_\omega^2(\omega_i))V(y_i^{under}|\,\cdot) + V_\omega(\omega_i)E^2(y_i^{under}|\,\cdot) \\ &= (\theta(1-\theta) + (1-\theta)^2)V(y_i^{under}|\,\cdot) + \theta(1-\theta)E^2(y_i^{under}|\,\cdot) \\ &= (1-\theta)V(y_i^{under}|\,\cdot) + \theta(1-\theta)E^2(y_i^{under}|\,\cdot). \end{aligned} \tag{3.6}$$

Due to its complicated form, it is not easy to make comparisons between the variances of the underlying count data distribution and the zero inflated model for the general case. Hence we will discuss each case separately in the following subsections, where we specify concrete distributions for y_i^{under}.

In the next subsections we are going to present several alternatives for the underlying count data distribution and the corresponding zero inflated versions. Of these alternatives, the mixture with Poisson and the mixture with Negative Binomial are the most commonly used in the literature. But all the results of the previous chapter provide new models to test.

3.1.1 Zero Inflated Poisson

The zero inflated Poisson model (ZIP) is a ZIM with an underlying Poisson distribution. We will denote it by

$$y_i|\,\eta_i,\theta \quad\sim\quad ZIP(\eta_i,\theta). \tag{3.7}$$

The density distribution can be derived from (3.4) by inserting the density of a Poisson distribution in $P(y_i^{under} = y_i|\cdot)$ and its given by

$$P(y_i|\,\eta_i,\theta) \quad = \quad \theta\big(1-I(y_i)\big) + (1-\theta)\frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!}. \tag{3.8}$$

The mean of the ZIP model is the mean of the Poisson multiplied by $(1-\theta)$ as given in (3.5)

$$E(y_i|\,\eta_i,\theta) \quad = \quad (1-\theta)\mu_i \tag{3.9}$$

We could not draw great conclusions for the variance of ZIM in the general case. But now, with the ZIP model, we see that with the help of (3.6) and (3.9) it takes the form

$$\begin{aligned} V(y_i|\,\eta_i,\theta) &= (1-\theta)\mu_i + \theta(1-\theta)\mu_i^2 \\ &= E(y_i|\,\eta_i,\theta) + E^2(y_i|\,\eta_i,\theta)\frac{\theta}{1-\theta}. \end{aligned} \tag{3.10}$$

The variance is a squared polynomial in the mean μ_i. This is a nice result, since it shows that ZIP models are able to account for overdispersion. With the same arguments as in Section 2.2, we see that $V(y_i|\,\eta_i,\theta) > E(y_i|\,\eta_i,\theta)$ and $\frac{\theta}{1-\theta}$ plays the role of the δ parameter in the models for unobserved heterogeneity.

For the later implementation of the model we need the whole likelihood of the data under the ZIP distributional assumption. Taking the product of (3.8) over all units we get

$$\begin{aligned} l(\boldsymbol{y}|\boldsymbol{\eta},\theta) &= \prod_{i=1}^{n} P(y_i|\,\eta_i,\theta) \\ &= \exp\left\{\sum_{i=0}^{n}\log\left(\theta(1-I(y_i)) + (1-\theta)\frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!}\right)\right\} \end{aligned}$$

$$
\begin{aligned}
&= \exp\left\{ \sum_{y_i=0} \log\left(\theta + (1-\theta)\exp(-\mu_i)\right) \right. \\
&+ \left. \sum_{y_i\neq 0} \log\left((1-\theta)\frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!} \right) \right\} \\
&\propto \exp\left\{ \sum_{y_i=0} \log\left(\theta + (1-\theta)\exp(-\mu_i)\right) \right. \\
&+ \left. Z_0 \log(1-\theta) + \sum_{y_i\neq 0} \left(-\mu_i + y_i \log(\mu_i)\right) \right\},
\end{aligned}
\tag{3.11}
$$

where Z_0 represents the number of units with strictly positive response. Note that we consider the likelihood only up to a proportionality constant for the same reasons as indicated in the previous chapter.

3.1.2 Zero Inflated Negative Binomial

The second most commonly used model in the literature is the zero inflated negative binomial (ZINB). It comes from a zero inflation on a negative binomial distribution. We will represent this model by

$$
y_i|\,\eta_i, \theta, \delta \;\sim\; ZINB(\eta_i, \theta, \delta). \tag{3.12}
$$

Note that the ZINB distribution depends on two more parameters together with those in the predictor. The density of a ZINB model is given by

$$
P(y_i|\,\eta_i, \theta, \delta) = \theta\big(1 - I(y_i)\big) + (1-\theta)\frac{\Gamma(y_i+\delta)}{\Gamma(y_i+1)\Gamma(\delta)} \left(\frac{\delta}{\delta+\mu_i}\right)^{\delta} \left(\frac{\mu_i}{\delta+\mu_i}\right)^{y_i}. \tag{3.13}
$$

The mean structure of the negative binomial distribution is equal to the one of the Poisson distribution. Thus the mean of the ZINB is also given by (3.9). We can proceed with the variance similarly as before. Then we get from (3.6) and (3.9)

$$
\begin{aligned}
V(y_i|\,\eta_i, \theta, \delta) &= (1-\theta)\left(\mu_i + \frac{\mu_i^2}{\delta}\right) + \theta(1-\theta)\mu_i^2 \\
&= (1-\theta)\mu_i + \mu_i^2(1-\theta)\left(\frac{1}{\delta} + \theta\right)
\end{aligned}
$$

$$= \quad E(y_i|\,\eta_i,\theta) + E^2(y_i|\,\eta_i,\theta)\left(\frac{\theta}{1-\theta} + \frac{1}{(1-\theta)\delta}\right). \tag{3.14}$$

Consequently, the ZINB offers a more flexible way to model the variability of the data, using the two parameters θ and δ. We see that there is some similarity to the variance structure of the ZIP model given in (3.10). The coefficient of the squared mean is extended by the $\frac{1}{\delta(1-\theta)}$ term. This could provide some reference point to compare the results from a ZIP and a ZINB in an informal way, and maybe to decide if the ZIP is enough to explain the variability in the data (if this term is small), or if a ZINB works better (otherwise).

The likelihood of the ZINB model is also calculated by the product of (3.13) over all units. After some calculations we get:

$$\begin{aligned} l(\boldsymbol{y}|\boldsymbol{\eta},\theta,\delta) &= \prod_{i=1}^{n} P(y_i|\,\eta_i,\theta,\delta) \\ &\propto \exp\left\{\sum_{y_i=0}\log\left(\theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right)\right. \\ &+ Z_0\Big(\log(1-\theta) - \log(\Gamma(\delta)) + \delta\log(\delta)\Big) \\ &+ \left.\sum_{y_i\neq 0}\Big(\log(\Gamma(y_i+\delta)) + y_i\log(\mu_i) - (y_i+\delta)\log(\delta+\mu_i)\Big)\right\}, \end{aligned} \tag{3.15}$$

where Z_0 is the number of units with strictly positive response. We see from the form of the likelihood that the calculating time for this model is highly influenced by the loggamma functions, which require a great computational effort.

3.1.3 Zero Inflated-Poisson with latent variables

In this subsection we present jointly the zero inflated models derived from a POGA, POIG or POLN assumption for the underlying count data distribution.

The first two models are the zero inflated POGA (ZIPGA) and the zero inflated POIG (ZIPIG). The common structure at the first hierarchical level is given by

$$y_i|\,\eta_i,\theta,\nu_i \quad\sim\quad ZIP(\eta_i,\theta,\nu_i), \tag{3.16}$$

where the difference is given by the prior assumptions for the ν_i, Gamma (2.9) and Inverse Gaussian (2.14) for ZIPGA or ZIPIG respectively. The density is given by

$$P(y_i|\,\eta_i,\theta,\nu_i) \;=\; \theta\big(1-I(y_i)\big)+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}. \tag{3.17}$$

The mean and variance of the response variable are given by

$$E(y_i|\,\eta_i,\theta,\nu_i) \;=\; (1-\theta)\nu_i\,\mu_i \tag{3.18}$$

$$V(y_i|\,\eta_i,\theta,\nu_i) \;=\; E(y_i|\,\eta_i,\theta,\nu_i)+E^2(y_i|\,\eta_i,\theta,\nu_i)\left(\frac{\theta}{1-\theta}\right). \tag{3.19}$$

Note that they have the same structure as with the ZIP model. So $\frac{\theta}{1-\theta}$ could also be interpreted as a sort of dispersion parameter. But it is not the only source of extra variability. We have to consider that the inclusion of the ν_i terms also influences the flexibility of the model to account for heterogeneity.

For both models the likelihood is the product of (3.17) over all observations in our data, and it is, up to a proportionality constant, given by

$$\begin{aligned} l(\boldsymbol{y}|\boldsymbol{\eta},\theta,\boldsymbol{\nu}) &= \prod_{i=1}^{n} P(y_i|\,\eta_i,\theta) \\ &= \exp\left\{\sum_{i=0}^{n}\log\left(\theta(1-I(y_i))+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}\right)\right\} \\ &\propto \exp\left\{\sum_{y_i=0}\log\left(\theta+(1-\theta)\exp(-\nu_i\,\mu_i)\right)\right. \\ &+ \left. Z_0\log(1-\theta)+\sum_{y_i\neq 0}\Big(-\nu_i\,\mu_i+y_i(\log(\mu_i)+\log(\nu_i))\Big)\right\}, \end{aligned} \tag{3.20}$$

with Z_0 defined as above.

The zero inflated POLN model (ZIPLN) can be implemented in a similar way as the ZIP. The κ_i in (2.17) are an additive part of the predictor and they do not destroy the basic model structure given in Subsection 3.1.1. Hence it is not necessary to rewrite the hole model once again.

3.2 Hierarchical centering

Applying the same idea as in Section 2.3, we reparameterize the models ZIPGA and ZIPIG by moving the intercept from the predictor to the ν_i terms. We then obtain the priors given in (2.19) and (2.20) for the new $\tilde{\nu}_i$ and the modified predictor $\tilde{\eta}_i$ without intercept term respectively. The resulting models will be called ZIPGAH and ZIPIGH. Since the only differences to the ZIPGA and ZIPIG models lie in the prior of the $\tilde{\nu}_i$ and presence or absence of an intercept in the model, the density and likelihood presented in the last section are valid for the new models.

Note that these models could also be obtained by applying zero inflation to the POGAH and POIGH. Both procedures are equivalent.

The ZIPGAH and ZIPIGH are not of great relevance, because they are not new models on their own but rather reparameterized versions of the ZIPGA and ZIPIG models. Nevertheless it may be interesting to test them on data, where their nonhierarchical equivalents do not work properly.

3.3 Résumé

The focus of this chapter was to present excess of zeros in count data and the zero inflation as a solution to this problem. As we have seen, zero inflated models are able to account for some amount of overdispersion in the data.

In this work we have modeled the selection variables ω_i independently from any observed covariates. A desirable extension is to include a second predictor in the model, linked to the θ parameter through a logit function. In this case we will have a parameter θ_i for each unit in the model, see Lambert (1992).

Another important item is the inconsistency of estimates when ignoring the presence of zero inflation. Note that in our modeling the mean of the underlying count data distribution is transformed by the factor $(1 - \theta)$, as given in (3.5). As long as θ is equal for

all units in the data set, this factor will only affect the estimation of the intercept, but not the estimation of other parameters in the predictor. We can directly compare the estimation results for the rest of the terms in the predictor from the 'normal' model with those obtained from the zero inflated model. But if we introduce a second predictor in the model, the mean of the underlying count data distribution will be multiplied by $(1 - \theta_i)$, affecting all effects in the predictor and not only the intercept. That makes comparison more difficult, but is of course a very interesting point that could be considered in future research.

After analysing the results obtained by applying ZIM to the car insurance data (see Section 7.2), it turns out that it could also be of interest to implement and apply hurdle models to this data.

Underreporting may be a very interesting approach to the car insurance data, in particular due to its elemental interpretation: Maybe some low costs for car body damages are not reported to the company by the policy holders so as to avoid higher premiums in the coming years. That would mean, that not all accidents are reported and it would explain the large amount of zero counts in the data.

For more literature about excess of zeros, see Lee, Stevenson, Wang and Yau (2002), Ridout, Hinde and Demétrio (2001), Agarwal, Gelfand and Citron-Pousty (2002) and Wikle and Anderson (2003, to appear)

Chapter 4

Priors and modeling of covariate effects

In a Bayesian regression framework the parameters in the model are supposed to follow some underlying distributions, called priors. To complete the exposition of our Bayesian generalized regression models, we have to discuss these priors. They should account for available information and reflect our prior knowledge about the parameters. Often, these priors will depend on further parameters, called hyperparameters. This reflects the hierarchical structure of the model. Of course it is possible and sometimes desirable to put also prior distributions on these hyperparameters. We are then talking about hyper-priors. We may distinguish between priors for the covariates in the predictor and priors for the model specific parameters. The first group imposes structures on the covariates and thus is an important and active part of the model construction. Priors in second group may be more determined by the nature of the parameter. In any case it is always recommended to take care and not to put too much information on the prior.

The scope of this chapter is to complete the formulation of the models. In Chapters 2 and 3 we have described several possibilities to model a count response variable, that are more flexible as the common Poisson assumption. Now we first present priors for the

model specific parameters, and then priors for the parameters and functions in the predictor. We will deal with the first type of priors in Section 4.1. The predictor is common for all model formulations presented here and depends on the data situation or, more precisely, on the covariate situation given by the data. We give an overview of possible terms in Section 4.2, where we will present the elements which form the predictor and their priors to complete the Bayesian formulation of the models. In Section 4.3 we give a graphical representation of the hierarchy of the models, that will help us to factorize the posterior in the next chapter.

4.1 Priors

This section will make some comments on the choice of the prior distributions for the model specific parameters. The terms in the predictor and their priors will be explained in the next subsection. The individual specific random effects are already commented on each model in the preceding subsections in case they are present. So the only two parameters that have to be considered here are the scale parameter δ and the zero inflation parameter θ.

The parameter δ has common properties for all the models where it is present, so the assumptions below are also valid for all of them (see the comments about the Poisson–Normal model below). Because $\delta > 0$, only distributions with positive domain are appropriate priors. If we do not have any knowledge about the parameter, which will be the normal situation, a proper distribution is the best option. More precisely we choose a gamma distribution

$$\delta \quad \sim \quad G(a, b) \tag{4.1}$$

with density

$$g(\delta) \quad = \quad \frac{b^a}{\Gamma(a)} \delta^{a-1} e^{-b\,\delta}$$

$$\begin{aligned} E(\delta) &= \frac{a}{b} \\ V(\delta) &= \frac{a}{b^2}\,. \end{aligned} \tag{4.2}$$

The parameters a and b can be fixed for the model. Standard values in the literature are for example $a = 1$ and $b = 0.005$. For a fully Bayesian approach, they can be considered as hyperparameters and some hyperpriors should then be introduced in the model. In this work we are going to treat only b as a hyperparameter. Reasons for this decision are given in Subsection 5.3.2, when posteriors are analyzed in more detail. As hyperprior we choose once again a gamma distribution

$$b \sim G(\alpha_1, \alpha_2) \tag{4.3}$$

with $\alpha_1 = 1$ and $\alpha_2 = 0.005$ because the hyperparameter b can only take positive values and we obtain a known form for the full conditional distribution of b easy to work with, as explained in Chapter 5. In the following we will write the distribution and the moments of δ given in (4.2) conditional on b. In case of $\nu_i = \exp(\kappa_i)$, δ is not modeled directly, but $\tau_\kappa^2 = \log(1 + \frac{1}{\delta})$. The implemented hyperprior is an inverse gamma distribution $\tau_\kappa^2 \sim IG(a, b)$ with fixed $a = 1$ and $b = 0.005$. As we will see in Subsection (4.2.1), this is the standard choice of hyperprior for the variance of random effects. Note that the inverse gamma defined here does not have a properly defined mean nor a variance.

For the parameter θ the prior assumption must also respect its nature. Since θ indicates a probability, only values between 0 and 1 are allowed. The prior we have chosen is a uniform distribution over the interval $[0, 1]$. We will write

$$\theta \sim U[0, 1] \tag{4.4}$$

with density $g(\theta) = 1$, mean $E(\theta) = 0.5$ and variance $V(\theta) = \frac{1}{12}$. A Beta distribution may be also a good alternative for the zero inflation parameter. However, our opinion is that the next extension of this model in further work should not be concerned with the investigation of prior distributions for θ, but the extension to two predictors to include covariates in the modeling of θ.

4.2 Predictors

After we have presented the different observation models we are going to work with, it is time to take a look at the predictor. As said before, it mainly depends on the type of covariates that we have observed, and on how we want to model their effects on the response variable. First we make some comments on the covariate types, fix the notation for the predictor and then describe which priors are appropriate in each case.

The covariates are in general either discrete or metrical. Discrete covariates may be dummy variables (0/1) or categorical. In the latter case they may be interpreted as some group indicator and handled as a random effect. If these group indicators refer to some spatial information, then we talk about spatial covariates. Metrical covariates may be characterized by some timescale (e.g. car age in the application of Chapter 7) or another metrical quantity (e.g. number of driven km per year also in Chapter 7).

Now we fix the notation for this section. Suppose we have data $(y_i, r_i, \boldsymbol{x_i}', \boldsymbol{z_i}')$ for $i = 1, \ldots, n$, where y_i is the response variable for the unit i, r_i is an unit specific offset, metrical and spatial covariates are given in $\boldsymbol{x_i}' = (x_{i1}, \ldots, x_{iP})$ and further covariates in the vector $\boldsymbol{z_i}' = (z_{i1}, \ldots, z_{iQ})$. The additive predictor is then given by

$$\eta_i = \alpha + \boldsymbol{z_i}'\boldsymbol{\beta} + \sum f^{(j)}(x_{ij}) + \rho_{g_i} \tag{4.5}$$

for $i = 1, \ldots, n$. In (4.5) α is an intercept, common to all units, $\boldsymbol{\beta}$ is the vector of length Q of parameters for fixed effects, $f^{(j)}$ are unknown smooth functions, and ρ_g are random effects for the groups $g = 1, \ldots, G$. The prior distribution for the parameters should reflect the information available about the covariate.

4.2.1 Fixed and random effects

In this subsection we define some priors for fixed and random effects. For the first case the usual choice are diffuse priors which do not give any information about the effects.

$$p(\beta_q) \propto constant, \tag{4.6}$$

where the priors are supposed to be independent for each $q = 1, \ldots, Q$. Independent flat Gaussian priors are also a good choice for this case.

For the random effects we will take Gaussian independent priors

$$\rho_g \quad \sim \quad N(0, \tau_\rho^2), \tag{4.7}$$

for $g = 1, \ldots, G$. The parameter τ_ρ^2 is a hyperparameter, and we will assume an improper inverse gamma hyperprior with parameters $a = 1$ and $b = 0.005$ (as a standard option).

4.2.2 Metrical covariates

Suppose first that x is a metrical covariate with a vector of ordered equidistant observed values $(x_{(1)}, \ldots, x_{(m)}, \ldots, x_{(M)})$. Then we will denote the vector of function evaluations on these values of x by $\boldsymbol{f} = (f(x_{(1)}), \ldots, f(x_{(m)}), \ldots, f(x_{(M)}))$. To simplify the notation let $\boldsymbol{f} = (f_1 \ldots f_m \ldots f_M)$. In this situation it is natural to suppose that function evaluations of two consecutive values of x can not extremely differ, that means, $\boldsymbol{f}$ is a smooth function. To implement this intuitive assumption we distinguish two approaches. Note that both approaches can be presented in a unified matrix notation.

Random walk of first or second order

The easiest form of representing this intuitive approach is to penalize the differences between two consecutive values of x. Formally

$$\begin{aligned} f_1 \quad &\propto \quad constant \\ f_m - f_{m-1} \quad &\sim \quad N(0, \tau_f^2), \end{aligned} \tag{4.8}$$

for $m = 2, \ldots M$. This is called a first order random walk prior, in short RW1. By taking second differences into account, we get

$$f_1, f_2 \quad \propto \quad constant$$

$$\begin{aligned}(f_m - f_{m-1}) - (f_{m-1} - f_{m-2}) &= \\ f_m - 2f_{m-1} + f_{m-2} &\sim N(0, \tau_f^2),\end{aligned} \tag{4.9}$$

for $m = 3, \ldots, M$. This approach is called a second order random walk, in short RW2. A RW1 penalizes too big jumps between f_{m-1} and f_m. On the other hand a RW2 penalizes deviations from the linear trend. Therefore a RW2 imposes a smoother function $\boldsymbol{f}$ than a RW1 does. The influence of this penalty is controlled through the parameter τ_f^2 in both cases. The bigger the variance of the normal distribution, the rougher is the function $\boldsymbol{f}$. The hyperprior for τ_f^2 is an improper inverse gamma distribution, chosen in a similar way as the hyperprior for τ_ρ^2 given in Subsection 4.2.1.

We now rewrite the formulations above in matrix notation. The joint distribution of $\boldsymbol{f}$ in the RW1 case can be factorized as follows:

$$\begin{aligned}p(\boldsymbol{f}) &= p(f_M|f_{M-1}) \ldots p(f_m|f_{m-1}) \ldots p(f_2|f_1)p(f_1) \\ &\propto \exp\left(\frac{1}{2\tau_f^2} \sum_{m=2}^{M} (f_m - f_{m-1})^2\right) \\ &\propto \exp\left(\frac{1}{2\tau_f^2} \boldsymbol{f}' K_{RW1} \boldsymbol{f}\right) \\ &\sim N\left(0, \tau_f^2 \widetilde{K_{RW1}}\right),\end{aligned}$$

where $\frac{1}{\tau_f^2} K_{RW1}$ is the precision matrix of the Gaussian distribution. The so called *penalty matrix* K_{RW1} is given by:

$$K_{RW1} = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}$$

The same procedure applied to the RW2 gives the joint density

$$p(\boldsymbol{f}) = p(f_M|f_{M-1}, f_{M-2}) \ldots p(f_m|f_{m-1}, f_{m-2}) \ldots p(f_3|f_2, f_1)p(f_2)p(f_1)$$

$$\begin{aligned} &\propto \exp\left(\frac{1}{2\tau_f^2}\sum_{m=3}^{M}(f_m - 2f_{m-1} + f_{m-2})^2\right) \\ &\propto \exp\left(\frac{1}{2\tau_f^2}\boldsymbol{f}' K_{RW2}\boldsymbol{f}\right) \\ &\sim N\left(0, \tau_f^2 \widetilde{K_{RW2}}\right). \end{aligned}$$

In this case K_{RW2} is given by:

$$K_{RW2} = \begin{pmatrix} 1 & -2 & 1 & & & & & & \\ -2 & 5 & -4 & 1 & & & & & \\ 1 & -4 & 6 & -4 & 1 & & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & & 1 & -4 & 6 & -4 & 1 \\ & & & & & 1 & -4 & 5 & -2 \\ & & & & & & 1 & -2 & 1 \end{pmatrix}$$

Note that neither K_{RW1} nor K_{RW2} have full rank. Hence their inverse does not exist and what we have defined is a partially improper prior. Both are sparse diagonal matrices which is a good property for efficient implementation.

If the observed values of the covariate x are not equidistant, both RW1 and RW2 can be reformulated by including appropriate weights in the penalties (see Fahrmeir and Lang (2001*a*) and Knorr-Held (1997)).

Bayesian P–Splines

In the following a nonparametric approach for estimation of smooth functions is briefly introduced. For a complete review on Bayesian P–splines see Lang and Brezger (2004) and Biller (2000) for Bayesian spline regression. The main idea is to represent the unknown smooth function $f(\cdot)$ as a linear combination of some known basis functions, say $B_1(\cdot), \ldots, B_S(\cdot)$:

$$f(\cdot) = \sum_{s=1}^{S} \gamma_s B_s(\cdot) \tag{4.10}$$

Among all the possible basis functions we are going to focus our attention on B–splines. First we have to fix the degree of the splines, say d, and $K + 2d$ equally spaced knots as follows

$$\zeta_1 < \ldots < \zeta_{1+d} = x_{(1)} < \ldots < \zeta_k < \ldots < \zeta_{K+d} = x_{(M)} < \ldots < \zeta_{K+2d}$$

Now the basis consists of $S = K + d$ splines of degree d, denoted by B_s^d. Each spline is nonzero on a compact domain over $2 + d$ knots. Figure 4.1 gives an example for B–splines of degree 3.

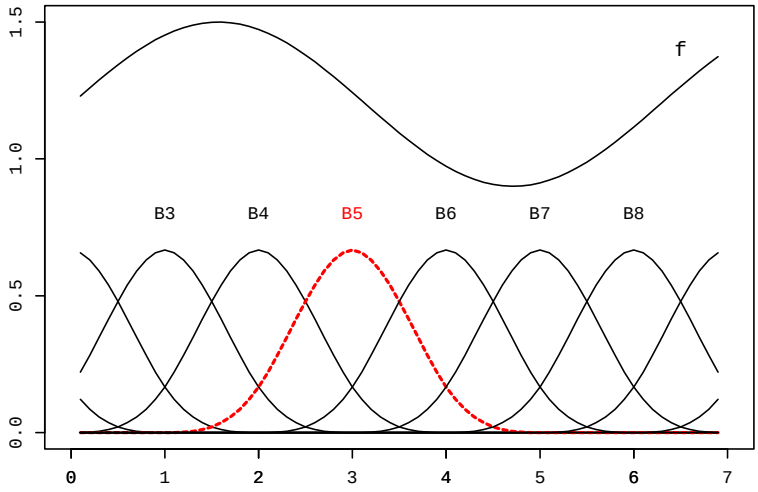

Figure 4.1: B–Splines basis of degree 3

It also gives a graphical description of the spline estimation idea: the smooth function f should be represented as a linear combination of the splines basis functions. For this purpose we have to estimate the coefficient vector $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_s, \ldots, \gamma_S)$ for (4.10). Note that if no further restriction is made, we may get a rather rough estimate for f. In maximum likelihood approaches a penalty term depending on a so called smoothing parameter is added to the likelihood of the model (see for example Eilers and Marx (1996) and Hastie and Tibshirani (1990)). As we are working in a Bayesian framework, we can control the roughness of the estimation by imposing an appropriate prior on the coefficients $\boldsymbol{\gamma}$. The two possibilities we consider here are a RW1 or RW2 as defined before

in (4.8) and (4.9). Thus we can take advantage of the Bayesian approach and avoid the calculation of an optimal smoothing parameter.

For implementation it is interesting to show that the P–spline approach can be written in matrix notation. Define $\boldsymbol{f}$ as in the RW situation and

$$B = \begin{pmatrix} B_{11} & \dots & B_{1S} \\ \vdots & B_{ms} & \vdots \\ B_{M1} & \dots & B_{MS} \end{pmatrix}$$

with $B_{ms} = B_s(x_{(m)})$. For simplicity we have omitted the degree of the basis functions. As the splines have only positive values on a compact interval and are zero elsewhere, the matrix B has some sort of band structure that can be used to improve the computational implementation. With these reformulations we get:

$$\boldsymbol{f} = B\boldsymbol{\gamma}.$$

The prior for $\boldsymbol{\gamma}$ is

$$\boldsymbol{\gamma} \sim N\left(0, \tau_\gamma^2 \widetilde{K_{RW}}\right), \tag{4.11}$$

where $\frac{1}{\tau_\gamma^2} K_{RW}$ is the precision matrix of the Gaussian distribution. For the penalty matrix K_{RW} we have to set K_{RW1}, if we have chosen a RW1 prior for the coefficients, or K_{RW2}, if we have chosen a RW2.

4.2.3 Spatial covariates

The data situation with spatial covariates can be resumed as follows. Let z be a covariate with spatial information for each unit in the dataset. Usually, z is an index for the regions1 to R of a geographical map, so that $z_i \in \{1, \dots, R\}$ for $i = 1, \dots, n$. Let Ω_r be the set of neighbors of region r, N_r the number of elements in Ω_r, and define $\boldsymbol{f} = (f_1, \dots, f_r, \dots, f_R)$ as the vector of effects of each region.

The aim is to find a prior that reflects the natural assumption, that effects of neighboring regions should be similar. The candidate we are going to consider here is a Gaussian intrinsic autoregression prior given by:

$$f_r|\boldsymbol{f}_{-r} \quad \sim \quad N\left(\frac{1}{N_r}\sum_{s\in\Omega_r} f_s, \frac{1}{N_r}\tau_f^2\right). \tag{4.12}$$

It says that the effect in region r has to be 'similar' to the mean of the effects of its neighbors. The amount of 'similarity' is controlled by the variance. It depends on the number of neighbors and may be problematic at the boundary regions, where there may be only few neighbors for some regions. It also depends on the parameter τ_f^2. In a fully Bayesian analysis, an inverse gamma prior is assigned to this hyperparameter.

We use (4.12) to write the joint density of $\boldsymbol{f}$ in matrix form similar to the random walks as

$$p(\boldsymbol{f}) \quad \propto \quad \exp\left(-\frac{1}{2\tau_f^2}\boldsymbol{f}'K_f\boldsymbol{f}\right). \tag{4.13}$$

The matrix K_f has N_r entries for the elements in the main diagonal, $K_{rs} = 1$ if regions r and s are neighbors and zero elsewhere. By definition K_f is a sparse band matrix. By reordering the regions an optimal form for K_f with minimal bandwidth can be found, which helps to improve the efficiency of computations. Note that the matrix K_f has no inverse because it is not of full rank. So the Gaussian distribution defined here is not proper.

Often an unstructured spatial term is also introduced, which should model extreme deviations from the imposed structured spatial prior. This term is considered as a random effect per region and is implemented as given in (4.7) at the beginning of this subsection.

Notice that both terms are based on the same covariate, namely the regional information. Despite this fact they are at least at the prior level identifiable, due to the different prior assumptions.

4.3 Hierarchy of the models

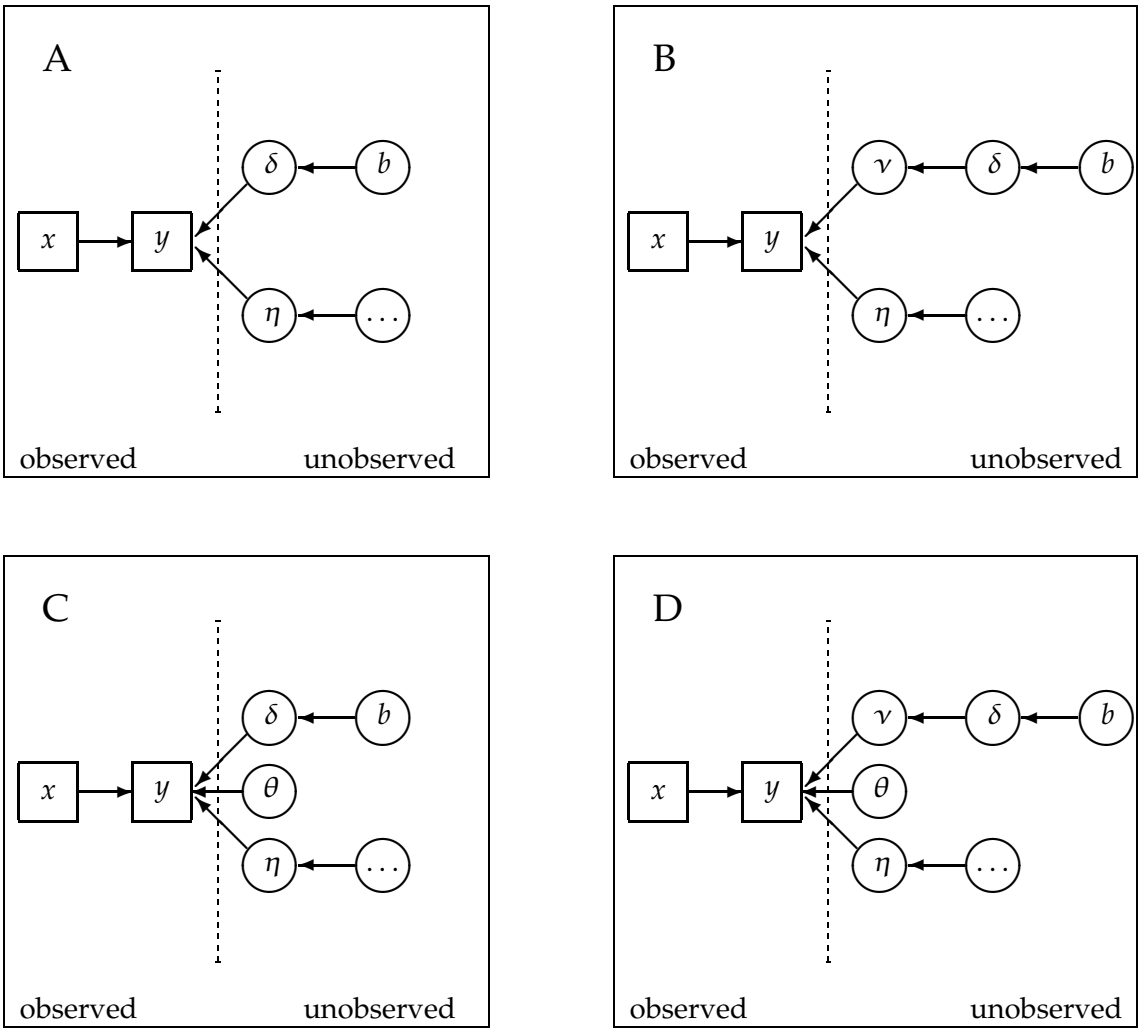

Figure 4.2: Representation of the hierarchy for the models with (right) or without (left) latent variables and with (bottom) or without (top) zero inflation

In Figure 4.2, we find a graphical representation of the different hierarchies of the models. This is important for the next chapter because the factorization of the posterior of the parameters depends on this structure. The difference is mainly given by the presence or absence of the latent variables or zero inflation in the model. In Figure 4.2, the first

row represents models with no zero inflation, and the second row models with zero inflation. Furthermore, the first column gives the models without latent variables, and the second the models where latent variables are present. So we can classify our models in to four groups. The first group only consists of the NB model, since it is the only model where no latent variables and no zero inflation are present. Its hierarchical structure is given by Picture A in Figure 4.2. There we see that the influence of the scale parameter δ directly affects the response variable. The dots in both pictures represent further hyperparameters which may also be implemented for the priors of the elements in $\boldsymbol{\eta}$. Picture B represents the second group and there we find the POGA, POIG and POLN models. The parameter δ appears a level beneath in the model hierarchy compared to Picture A. Now the individual specific random effects directly influence the response variable and they depend on δ. The third group contains the ZINB model and is given in Picture C. Its hierarchical structure is similar to the one in Picture A, but we find a new parameter θ in the model, that directly affects the response variable. The last group, given in Picture D, comprises the models ZIPGA, ZIPIG and ZIPLN, where latent variables and zero inflation are present.

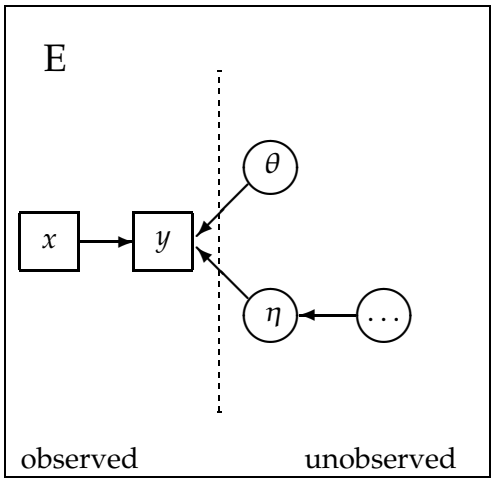

Figure 4.3: Representation of the hierarchy for the ZIP model

Note that the ZIP model can not be represented by any of the given pictures because of the absence of a dispersion parameter. Thus we add a fifth group E and give the graphical design of this model separately in Figure 4.3.

4.4 Résumé

In this chapter we have shown a small part of the potential flexibility of hierarchical Bayesian approaches for modeling data with complex covariate structure. We have seen that priors have to respect the underlying nature of the parameters and at the same time can force them to satisfy restrictions that may be desirable for some covariates.

We have to remark that the aim of a prior is to incorporate information in the model about the parameters, but it is important not to choose too informative priors, if we do not know how the parameter is actually distributed. The amount of 'information' of a prior is generally controlled through its hyperparameters and can be determined by examining the first moments or the shape of the density function, for example.

For $\boldsymbol{\gamma}$, $\boldsymbol{f}$ and $\boldsymbol{\rho}$ an unified form for their priors is possible using a matrix representation. Note that with

$$p(\boldsymbol{v}|\tau_v^2) \quad \propto \quad \exp\left\{-\frac{1}{2\,\tau_v^2}\boldsymbol{v}'K_v\boldsymbol{v}\right\} \tag{4.14}$$

we can represent all three priors (4.7), (4.11) and (4.13) by taking $\boldsymbol{v} \in \{\boldsymbol{\rho}, \boldsymbol{\gamma}, \boldsymbol{f}\}$, the penalty matrix $K_v \in \{I_G, K_\gamma, K_f\}$ and the hyperparameter $\tau_v^2 \in \{\tau_\rho^2, \tau_\gamma^2, \tau_f^2\}$ respectively. This fact is very useful for the posterior inference in the next Chapter.

Another important fact for an unified representation is given by the matrix notation of the predictor. Note that in the presence of some fixed effects parametrized by $\boldsymbol{\beta}$, a metrical covariate x with parameter vector $\boldsymbol{\gamma}$ for the spline, a random effect represented by $\boldsymbol{\rho}$ and spatial information modeled by $\boldsymbol{f}$, we will have the predictor

$$\boldsymbol{\eta} \quad = \quad X_\beta\boldsymbol{\beta} + X_\gamma\boldsymbol{\gamma} + X_\rho\boldsymbol{\rho} + X_f\boldsymbol{f}. \tag{4.15}$$

For the fixed effects, X_β is a nxQ matrix, where Q is the number of discrete covariates, with entries

$$(X_\beta)_{iq} \quad = \quad z_{iq}. \tag{4.16}$$

From (4.10) we can derive X_γ as a nxS matrix (S = number of basis functions for the Spline) with entries given by

$$(X_\gamma)_{is} \quad = \quad B_s(x_i), \tag{4.17}$$

which means that the i^{th} row of X_γ are the values of the basis functions on the observed value of x for the i^{th} unit.

X_ρ is a nxG $0/1$ incidence matrix with

$$(X_\rho)_{ig} \quad = \quad \begin{cases} 1 & g_i = g \\ 0 & \text{otherwise.} \end{cases} \tag{4.18}$$

Remember that G is the observed number of different categories for the discrete covariate defining the random effect ρ.

X_f is also nxR $0/1$ incidence matrix with following entries

$$(X_f)_{ir} \quad = \quad \begin{cases} 1 & r_i = r \\ 0 & \text{otherwise.} \end{cases} \tag{4.19}$$

We will use this unified representation in the next chapter, where we present the algorithms for posterior inference on our models.

Chapter 5

Posterior inference

In Bayesian regression inference is based on the analysis of the posterior distribution of the parameters given the data. In general this high dimensional posterior will not have a known closed form but rather a complicated high dimensional density only known up to the proportionality constant, which makes direct inference almost impossible. Markov Chain Monte Carlo (MCMC) Methods are sophisticated techniques that have been developed to resolve this problem. A brief overview about MCMC theory and some bibliographic information are given in Appendix C.

For practical applications of MCMC methods we proceed as follows. First we build the joint posterior of the parameters in the model. Then we derive the full conditional distribution for each natural group of parameters, that is the conditional distribution of this group of parameters given the data and all the other parameters in the model. If the full conditional is proportional to a known distribution, we can apply Gibbs–sampling (see Section C.1 in Appendix C). If not, then Metropolis–Hastings (M–H) sampling has to be implemented (Section C.2) and we need to find a so called proposal distribution for the algorithm.

In Section 5.1 we first calculate the joint posterior distributions for the models. Afterwards we derive the full conditionals for each parameter block from these posteriors

and, if Gibbs–sampling is not possible, appropriate proposal distributions are given in Section 5.3. The last section gives an overview of the sampling algorithms.

5.1 Posteriors

Once we have the model structure and priors for the parameters, we can calculate the joint posterior distribution of the parameters in the model given the data. In a short form:

$$\begin{aligned} \pi(\xi|y) &= \frac{l(y|\xi)P(\xi)}{P(y)} \\ &\propto l(y|\xi)P(\xi), \end{aligned}$$

with ξ denoting the parameters in the model and $P(\xi)$ their prior distribution. The likelihood of the data given the parameters is $l(y|\xi)$ and $P(y)$ is the marginal distribution of the data.

For the calculations of the posteriors in this work we must differentiate between the five groups represented in Figures 4.2 and 4.3, because the hierarchical structure determines the form of factorizing the joint distribution. The five possibilities were: First the group A (NB model), without latent variables and zero inflation but with overdispersion parameter. Second, group B (POGA, POIG and POLN models) with latent variables but still without zero inflation. Third the group C (ZINB model) without latent variables, but with overdispersion and zero inflation. Fourth, group D (ZIPGA, ZIPIG and ZIPLN models) with latent variables and zero inflation. And last, group E, the fifth group, where we have the ZIP model without overdispersion, but with zero inflation.

For convenience, we will classify the groups in three blocks. The first block is presented in Subsection 5.1.1 and contains the hierarchical groups A and C. In Subsection 5.1.2 we calculate the posteriors for the second block of groups, namely B and D. And the third block contains the hierarchical group E and is shown in Subsection 5.1.3.

For the following we remember that $\boldsymbol{\beta}$ denotes the parameter vector of fixed effects, $\boldsymbol{\gamma}$ the vector of coefficients for the splines, $\boldsymbol{f}$ the structured spatial effects and $\boldsymbol{\rho}$ some random effects in the model. Please note that without loss of generality all together can be shortly represented by $\boldsymbol{\eta}$ (see (4.5)). If present, $\boldsymbol{\nu}$ or $\boldsymbol{\kappa}$ refer to the unit specific latent variables, δ to the dispersion parameter and θ to the zero inflation parameter.

5.1.1 Posteriors for groups A and C

Under reasonable conditional independence assumptions the posterior distribution for the NB model (group A) is given by:

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \delta, b | \, \boldsymbol{y}) \quad &\propto \quad l(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \delta) \\
&\quad P(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \delta, b) \\
&= \quad l(\boldsymbol{y}|\boldsymbol{\eta}, \delta) \, p(\boldsymbol{\beta}) \, p(\boldsymbol{\gamma}|\tau_\gamma^2) \, g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2) \, g(\tau_f^2) \, p(\boldsymbol{\rho}|\tau_\rho^2) \, g(\tau_\rho^2) \\
&\quad g(\delta|b) \, g(b).
\end{aligned} \tag{5.1}$$

A similar result holds for the ZINB model (group C), where in addition we have the zero inflation parameter θ.

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \delta, b, \theta | \, \boldsymbol{y}) \quad &\propto \quad l(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \delta, \theta) \\
&\quad P(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \delta, b, \theta) \\
&= \quad l(\boldsymbol{y}|\boldsymbol{\eta}, \delta, \theta) \, p(\boldsymbol{\beta}) \, p(\boldsymbol{\gamma}|\tau_\gamma^2) \, g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2) \, g(\tau_f^2) \, p(\boldsymbol{\rho}|\tau_\rho^2) \, g(\tau_\rho^2) \\
&\quad g(\delta|b) \, g(b) \, g(\theta).
\end{aligned} \tag{5.2}$$

All the factors in these products of distributions are presented in Chapters 2, 3 and 4. The likelihood of the model $l(\boldsymbol{y}|\boldsymbol{\eta}, \delta)$ is in the first case the density of a Negative Binomial distribution and is given in (2.4). Or in case of the ZINB model, $l(\boldsymbol{y}|\boldsymbol{\eta}, \delta, \theta)$ is as given

in (3.15). For $g(\delta|b)$ and $g(b)$ two Gamma distributions were chosen as respectively explained in (4.2) and (4.3). If we have zero inflation, $g(\theta)$ is defined in (4.4). For the fixed effects we assume a diffuse prior $p(\boldsymbol{\beta}) \propto constant$ as said in (4.6), and for the random effects $p(\boldsymbol{\rho}|\tau_\rho^2)$ as in (4.7). The prior $p(\boldsymbol{\gamma}|\tau_\gamma^2)$ of the coefficients for the P–splines is given in (4.11). The structured spatial effects prior $p(\boldsymbol{f}|\tau_f^2)$ is the GMRF described in (4.13). Finally, all the $g(\tau_\cdot^2)$ are distributed as $IG(a,b)$, with hyperparameters a, b. Standard choices are $a = 1, b = 0.005$ or $a = b = 0.001$; with the latter choice the IG prior is nearer to Jeffrey's noninformative prior.

5.1.2 Posteriors for groups B and D

Due to the notational remarks made in Subsection 2.2.3 about implementation of the POLN model, we need two posterior forms for the latent variables cases: one for the POGA and POIG models and another one, similar in interpretation but in some notational aspects different, for the POLN model. Of course we get the same classification for the corresponding zero inflated versions.

In the presence of latent variables $\boldsymbol{\nu}$ without zero inflation and under conditional independence assumptions the posterior generally looks like:

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \boldsymbol{\nu}, \delta, b|\, \boldsymbol{y}) &\propto l(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \boldsymbol{\nu}) \\
&\quad P(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \boldsymbol{\nu}, \delta, b) \\
&= l(\boldsymbol{y}|\boldsymbol{\eta}, \boldsymbol{\nu}) p(\boldsymbol{\beta}) p(\boldsymbol{\gamma}|\tau_\gamma^2) g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2) g(\tau_f^2) p(\boldsymbol{\rho}|\tau_\rho^2) g(\tau_\rho^2) \\
&\quad g(\boldsymbol{\nu}|\delta) g(\delta|b) g(b).
\end{aligned} \tag{5.3}$$

Adding zero inflation to the model, we obtain

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \boldsymbol{\nu}, \delta, b, \theta|\, \boldsymbol{y}) &\propto l(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \boldsymbol{\nu}, \theta) \\
&\quad P(\boldsymbol{\beta}, \boldsymbol{\gamma}, \tau_\gamma^2, \boldsymbol{f}, \tau_f^2, \boldsymbol{\rho}, \tau_\rho^2, \boldsymbol{\nu}, \delta, b, \theta)
\end{aligned}$$

$$
\begin{aligned}
&= l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\nu},\theta)p(\boldsymbol{\beta})p(\boldsymbol{\gamma}|\tau_\gamma^2)g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2)g(\tau_f^2)p(\boldsymbol{\rho}|\tau_\rho^2)g(\tau_\rho^2) \\
&\quad g(\boldsymbol{\nu}|\delta)g(\delta|b)g(b)g(\theta).
\end{aligned} \tag{5.4}
$$

As before, all the factors in (5.3) and (5.4) are already explained in the last chapter. They essentially remain the same as in (5.1) and (5.2), with some slight differences. The likelihood terms are now $l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\nu})$, which is a Poisson distribution common for all the latent variables models, given in (2.11), and $l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\nu},\theta)$, a zero inflated Poisson distribution, given in (3.20). For the prior distribution denoted by $g(\boldsymbol{\nu}|\delta) = \prod_{i=1}^{n} g(\nu_i|\delta)$ we can choose a Gamma prior as given in (2.9) or an Inverse Gaussian as in (2.14).

For the POLN model, the posterior is calculated in a similar way:

$$
\begin{aligned}
\pi(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2,\boldsymbol{\kappa},\tau_\kappa^2|\,\boldsymbol{y}) &\propto l(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{f},\boldsymbol{\rho},\boldsymbol{\kappa}) \\
&\quad P(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2,\boldsymbol{\kappa},\tau_\kappa^2) \\
&= l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\kappa})p(\boldsymbol{\beta})p(\boldsymbol{\gamma}|\tau_\gamma^2)g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2)g(\tau_f^2)p(\boldsymbol{\rho}|\tau_\rho^2)g(\tau_\rho^2) \\
&\quad g(\boldsymbol{\kappa}|\tau_\kappa^2)g(\tau_\kappa^2).
\end{aligned} \tag{5.5}
$$

For the ZIPLN, where zero inflation is included in the model, we have

$$
\begin{aligned}
\pi(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2,\boldsymbol{\kappa},\tau_\kappa^2,\theta|\,\boldsymbol{y}) &\propto l(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{f},\boldsymbol{\rho},\boldsymbol{\kappa},\theta) \\
&\quad P(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2,\boldsymbol{\kappa},\tau_\kappa^2,\theta) \\
&= l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\kappa},\theta)p(\boldsymbol{\beta})p(\boldsymbol{\gamma}|\tau_\gamma^2)g(\tau_\gamma^2) \\
&\quad p(\boldsymbol{f}|\tau_f^2)g(\tau_f^2)p(\boldsymbol{\rho}|\tau_\rho^2)g(\tau_\rho^2) \\
&\quad g(\boldsymbol{\kappa}|\tau_\kappa^2)g(\tau_\kappa^2)g(\theta).
\end{aligned} \tag{5.6}
$$

In these cases, the likelihood $l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\kappa})$ is the product of the individual likelihood contributions, defined in the table of Subsection 2.2.3 as Poisson densities, and $l(\boldsymbol{y}|\boldsymbol{\eta},\boldsymbol{\kappa},\theta)$ is the term given in (3.11). The vector of parameters $\boldsymbol{\kappa}$ is handled as a common vector of

random effects for each unit, and therefore $g(\boldsymbol{\kappa}|\tau_\kappa^2)$ and $g(\tau_\kappa^2)$ are specified by (4.7) and by a $IG(a,b)$ distribution, respectively. The rest of the factors remains as explained in Subsection 5.1.1.

5.1.3 Posterior for group E

Finally, we present here the posterior for the basic ZIP, which can be easily derived from the hierarchy given in Figure 4.3.

$$\begin{aligned}\pi(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2\theta|\,\boldsymbol{y}) &\propto l(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{f},\boldsymbol{\rho},\theta)\\ &\quad P(\boldsymbol{\beta},\boldsymbol{\gamma},\tau_\gamma^2,\boldsymbol{f},\tau_f^2,\boldsymbol{\rho},\tau_\rho^2,\theta)\\ &= l(\boldsymbol{y}|\boldsymbol{\eta},\theta)p(\boldsymbol{\beta})p(\boldsymbol{\gamma}|\tau_\gamma^2)g(\tau_\gamma^2)\\ &\quad p(\boldsymbol{f}|\tau_f^2)g(\tau_f^2)p(\boldsymbol{\rho}|\tau_\rho^2)g(\tau_\rho^2)\\ &\quad g(\theta).\end{aligned} \tag{5.7}$$

The products used here are the same as in Subsection 5.1.1. The difference is given by the likelihood term $l(\boldsymbol{y}|\boldsymbol{\eta},\theta)$, which is defined as in (3.11).

5.2 Full conditionals

It is clear that none of the possible posteriors described before has a 'nice' closed form, from which we could directly draw samples for inference. Therefore we must proceed with the analysis of the full conditionals of blocks of parameters as explained in this section.

5.2.1 Predictor terms and their hyperparameters

This part of the calculation of the full conditionals is well known in the standard literature. The elements in the predictor are common for all models presented here. The

difference in the full conditionals calculated from (5.1) to (5.7) for these terms is only given by the likelihood factor, which is the product of Poisson, Negative Binomial, zero inflated Poisson or zero inflated Negative Binomial densities. This likelihood term will be represented in this Subsection jointly for all the models by $l(\boldsymbol{y}|\boldsymbol{\eta}, \cdot)$, where '$\cdot$' represents

- δ or δ, θ in a NB or ZINB model
- $\boldsymbol{\nu}$ or $\boldsymbol{\nu}, \theta$ for a POGA, POIG or ZIPGA, ZIPIG models
- $\boldsymbol{\kappa}$ or $\boldsymbol{\kappa}, \theta$ for the POLN or ZIPLN formulation
- only θ for the ZIP model.

Consequently we are only going to write down the full conditionals for each block of parameters in a general form, which is valid for all the models.

Let us begin with the block $\boldsymbol{\beta}$. Its full conditional is given by

$$\begin{aligned} \pi(\boldsymbol{\beta}|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \cdot)\, p(\boldsymbol{\beta}) \\ &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \cdot), \end{aligned} \tag{5.8}$$

resulting from $p(\boldsymbol{\beta}) \propto \textit{constant}$ that the full conditional of $\boldsymbol{\beta}$ is proportional to the likelihood of the model.

Using the unified form for the priors of $\boldsymbol{\gamma}$, $\boldsymbol{f}$ and $\boldsymbol{\rho}$ given in (4.14) we can represent their full conditionals in a compact way as product of the joint likelihood and the prior,

$$\begin{aligned} \pi(\boldsymbol{v}|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \cdot)\, p(\boldsymbol{v}|\tau_v^2) \\ &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \cdot)\, \exp\left\{-\frac{1}{2\,\tau_v^2}\boldsymbol{v}' K_v \boldsymbol{v}\right\} \end{aligned} \tag{5.9}$$

for $\boldsymbol{v} \in \{\boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}\}$, the penalty matrix $K_v \in \{K_\gamma, K_f, I_G\}$ and the hyperparameter $\tau_v^2 \in \{\tau_\gamma^2, \tau_f^2, \tau_\rho^2\}$ respectively.

The joint posterior distribution only depends on the hyperparameter τ_v^2 through its prior and the prior for $\boldsymbol{v}$. The first distribution is the same for τ_γ^2, τ_f^2 and τ_ρ^2, and for the second

we have a unified representation for all three values in (4.14). It is clear that we can also write down a general form for the full conditional, valid for all $\tau_v^2 = \tau_\gamma^2, \tau_f^2, \tau_\rho^2$ and given by

$$\begin{aligned}
\pi(\tau_v^2|\ldots) &\propto p(\boldsymbol{v}|\tau_v^2)g(\tau_v^2) \\
&\propto \exp\left\{-\frac{\text{rank}(K_v)}{2}\ln(\tau_v^2) - \frac{1}{2\,\tau_v^2}\boldsymbol{v}'K_v\boldsymbol{v} - (a+1)\ln(\tau_v^2) - \frac{b}{\tau_v^2}\right\} \\
&= \exp\left\{-(a + \frac{\text{rank}(K_v)}{2} + 1)\ln(\tau_v^2) - \frac{1}{\tau_v^2}(\frac{1}{2}\boldsymbol{v}'K_v\boldsymbol{v} + b)\right\} \\
&\propto IG\left(a + \frac{1}{2}\text{rank}(K_v), \frac{1}{2}\boldsymbol{v}'K_v\boldsymbol{v} + b\right). \qquad (5.10)
\end{aligned}$$

As said in Subsections 2.2.3 and 3.1.3, we are going to work with the POLN and ZI-PLN models by assuming that the random effects $\boldsymbol{\kappa}$ are added into the predictor and are normally distributed. In fact they are nothing else as common random effects defined in Subsection 4.2.1. Therefore, although they are model 'specific' parameters, we make some remarks about their full conditional here and not in the next section as for the other models.

Obviously, if $\boldsymbol{\kappa}$ is a vector of random effects the full conditional can be also represented by (5.9) by setting $\boldsymbol{v} = \boldsymbol{\kappa}$, $K_v = I_n$ and $\tau_v^2 = \tau_\kappa^2$.

The same generalization holds for the full conditional of τ_κ^2, which is also given by (5.10).

5.2.2 Model specific parameters

Because the following parameters are specific for the models and can not be found in the standard literature, this subsection describes in more detail how to calculate the full conditionals. First we indicate for each parameter which terms of the general joint posterior build the full conditional and then we move on to the concrete models.

We begin with the vector $\boldsymbol{\nu}$. Its components are supposed to be independent a priori. Hence we can sample and update each component separately. Of course, to calculate the full conditionals for $\boldsymbol{\nu}$ only the posteriors given in (5.3) and (5.4) are relevant. Hence

we distinguish between the full conditional in a general case for a single component i with (from (5.4)) or without (from (5.3)) zero inflation. We eliminate all factors from the posteriors, that do not depend on ν_i and obtain following expressions

$$\pi(\nu_i|\ldots) \propto l(y_i|\eta_i,\nu_i,\theta)g(\nu_i|\delta) \tag{5.11}$$

$$\pi(\nu_i|\ldots) \propto l(y_i|\eta_i,\nu_i)g(\nu_i|\delta), \tag{5.12}$$

respectively. Now we analyze the concrete models, first with zero inflation, namely ZIPGA and ZIPIG models, and second without, for POGA and POIG.

For the ZIPGA model the likelihood $l(y_i|\eta_i,\nu_i,\theta)$ is given in (3.17) and the prior $g(\nu_i|\delta)$ in (2.9). Putting these expressions together in the product (5.11), we get:

$$\begin{aligned}
\pi(\nu_i|\ldots) &\propto l(y_i|\eta_i,\nu_i,\theta)g(\nu_i|\delta) \\
&= \left\{\theta\big(1-I(y_i)\big)+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}\right\} \\
&\quad \frac{\delta^\delta}{\Gamma(\delta)}\nu_i^{\delta-1}\exp(-\delta\,\nu_i) \\
&\propto \left\{\theta\big(1-I(y_i)\big)+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}\right\} \\
&\quad \exp\left\{(\delta-1)\log(\nu_i)-\delta\,\nu_i\right\}.
\end{aligned} \tag{5.13}$$

Without great effort we see that, due to the complicated form of the likelihood, this expression can not be rewritten to be proportional to any known distribution from which we could easily take samples. Thus we have to implement a M–H step for the update.

In a similar way, we obtain the result for the ZIPIG model. The full conditional is the same as before, but with an Inverse Gaussian distribution for the ν_i.

$$\begin{aligned}
\pi(\nu_i|\ldots) &\propto l(y_i|\eta_i,\nu_i,\theta)g(\nu_i|\delta) \\
&= \left\{\theta\big(1-I(y_i)\big)+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}\right\} \\
&\quad \sqrt{\frac{\delta}{2\,\pi\,\nu_i^3}}\exp\left\{-\delta\frac{(\nu_i-1)^2}{2\,\nu_i}\right\} \\
&\propto \left\{\theta\big(1-I(y_i)\big)+(1-\theta)\frac{\exp(-\nu_i\,\mu_i)(\nu_i\,\mu_i)^{y_i}}{y_i!}\right\}
\end{aligned}$$

$$\exp\left\{-\frac{3}{2}\ln(\nu_i) - \delta\frac{(\nu_i-1)^2}{2\,\nu_i}\right\}. \tag{5.14}$$

There is no closed form for this expression and M–H update step will be presented in the next section.

We now concentrate on the full conditionals for ν_i in the models without zero inflation POGA and POLN.

In a POGA formulation the likelihood term $l(y_i|\eta_i,\nu_i)$ comes from a Poisson distribution with parameter $\nu_i\,\mu_i$ (see (2.11)), and $g(\nu_i|\delta)$ is Gamma distributed. Including this information in (5.12) gives the full conditional for ν_i:

$$\begin{aligned}
\pi(\nu_i|\ldots) &\propto l(y_i|\eta_i,\nu_i)g(\nu_i|\delta)\\
&= \frac{\exp\{-\nu_i\,\mu_i\}(\nu_i\,\mu_i)^{y_i}}{y_i!}\frac{\delta^\delta}{\Gamma(\delta)}\nu_i^{\delta-1}\exp\{-\delta\,\nu_i\}\\
&\propto \exp\{-\nu_i\,\mu_i + y_i\ln(\nu_i) + (\delta-1)\ln(\nu_i) - \delta\,\nu_i\}\\
&\propto \exp\{(y_i+\delta-1)\ln(\nu_i) - (\mu_i+\delta)\nu_i\}\\
&\sim G(y_i+\delta,\mu_i+\delta)
\end{aligned} \tag{5.15}$$

In this particular case, the full conditional is proportional to a Gamma distribution with parameters $y_i+\delta$ and $\mu_i+\delta$. Therefore, Gibbs sampling can be used to update the ν_i's in a POGA formulation.

For a POIG model the procedure is the same. The likelihood term in (5.12) is similar as before, with the difference given only by the factor $g(\nu_i|\delta)$, which is now Inverse Gaussian distributed. After substituting these terms the full conditional is:

$$\begin{aligned}
\pi(\nu_i|\ldots) &\propto l(y_i|\eta_i,\nu_i)g(\nu_i|\delta)\\
&= \frac{\exp\{-\nu_i\,\mu_i\}(\nu_i\,\mu_i)^{y_i}}{y_i!}\sqrt{\frac{\delta}{2\,\pi\,\nu_i^3}}\exp\left\{-\delta\frac{(\nu_i-1)^2}{2\,\nu_i}\right\}\\
&\propto \exp\left\{-\nu_i\,\mu_i + y_i\ln(\nu_i) - \frac{3}{2}\ln(\nu_i) - \delta\frac{(\nu_i-1)^2}{2\,\nu_i}\right\}
\end{aligned} \tag{5.16}$$

This time we are not able to find a known distribution that is proportional to this full conditional. Hence Gibbs sampling is not possible and a M–H step must be implemented.

The parameter δ is present in the NB, POGA, POIG models and their zero inflated versions. We first present the full conditional in the hierarchical groups A and C (NB and ZINB respectively) and finally in the groups B (POGA and POIG) and D (ZIPGA and ZIPIG).

We can calculate the full conditional for δ in the NB formulation by eliminating all factors that do not depend on it from the joint posterior given in (5.1) as follows:

$$\begin{aligned}
\pi(\delta|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta},\delta)\, g(\delta|b) \\
&\propto \exp\Big\{ n\Big(\delta\,\log(\delta) - \log\Gamma(\delta)\Big) \\
&\quad + \sum_{i=1}^{n}\Big(-\log\Gamma(y_i+\delta) - (y_i+\delta)\log(\delta+\mu_i)\Big) \\
&\quad + (a-1)\log(\delta) - b\,\delta\Big\}.
\end{aligned} \tag{5.17}$$

For its zero inflated version ZINB we have to proceed in a similar way but using the posterior given in (5.2) instead:

$$\begin{aligned}
\pi(\delta|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta},\delta,\theta)\, g(\delta|b) \\
&\propto \exp\Bigg\{ \sum_{y_i=0}\log\left(\theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) \\
&+ Z_0\Big(\log(1-\theta) - \log(\Gamma(\delta)) + \delta\log(\delta)\Big) \\
&+ \sum_{y_i\neq 0}\Big(\log(\Gamma(y_i+\delta)) + y_i\log(\mu_i) - (y_i+\delta)\log(\delta+\mu_i)\Big) \\
&+ (a-1)\log(\delta) - b\delta\Bigg\} \\
&\propto \exp\Bigg\{ \sum_{y_i=0}\log\left(\theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) + Z_0\Big(\delta\log(\delta) - \log(\Gamma(\delta))\Big) \\
&+ \sum_{y_i\neq 0}\Big(\log(\Gamma(y_i+\delta)) - (y_i+\delta)\log(\delta+\mu_i)\Big) + (a-1)\log(\delta) - b\delta\Bigg\}.
\end{aligned} \tag{5.18}$$

Neither (5.17) nor (5.18) are proportional to any distribution from which it is easy to sample. They both have a rather complicated form, that will increase computation time.

An appropriate M–H step must be implemented in both cases following the explanations in the next section.

The general form of the full conditional for δ in a model formulation of the groups B and D is

$$\pi(\delta|\ldots) \quad \propto \quad g(\boldsymbol{\nu}|\delta)\, g(\delta|b). \tag{5.19}$$

The factor $g(\delta|b)$ is given in (4.2) and common for all four models. The difference is given by the factor $g(\boldsymbol{\nu}|\delta)$ and this is the same for POGA and ZIPGA (a Gamma distribution) and for POIG ans ZIPIG (an Inverse Gaussian distribution).

We begin with the POGA and ZIPGA formulations. As said before, $g(\boldsymbol{\nu}|\delta)$ comes from a Gamma distribution and the full conditional is proportional to

$$\begin{aligned}
\pi(\delta|\ldots) &\propto g(\boldsymbol{\nu}|\delta)\, g(\delta|b) \\
&= \prod_{i=1}^{n} \left\{ \frac{\delta^{\delta}}{\Gamma(\delta)} \nu_i^{\delta-1} \exp\{-\delta\,\nu_i\} \right\} \frac{b^a}{\Gamma(a)} \delta^{a-1} \exp\{-b\,\delta\} \\
&\propto \exp\left\{ \sum_{i=1}^{n} \Big(\delta \log(\delta) - \log\Gamma(\delta) + (\delta-1)\log(\nu_i) - \delta\,\nu_i \Big) \right. \\
&\quad \left. + (a-1)\log(\delta) - b\,\delta \right\} \\
&\propto \exp\left\{ n\Big(\delta\log(\delta) - \log\Gamma(\delta)\Big) + \delta \sum_{i=1}^{n} \Big(\log(\nu_i) - \nu_i \Big) \right. \\
&\quad \left. + (a-1)\log(\delta) - b\,\delta \right\}
\end{aligned} \tag{5.20}$$

It is not possible to find an appropriate distribution proportional to this full conditional, and we need the help of the M–H algorithm.

Finally, in the last two formulations (POIG and ZIPOIG) $\boldsymbol{\nu}|\delta$ is Inverse Gaussian distributed. From substituting it appropriately in (5.19) we get:

$$\begin{aligned}
\pi(\delta|\ldots) &\propto g(\boldsymbol{\nu}|\delta)\, g(\delta|b) \\
&= \prod_{i=1}^{n} \left\{ \sqrt{\frac{\delta}{2\,\pi\,\nu_i^3}} \exp\left\{ -\delta \frac{(\nu_i - 1)^2}{2\,\nu_i} \right\} \right\} \frac{b^a}{\Gamma(a)} \delta^{a-1} \exp\{-b\,\delta\}
\end{aligned}$$

$$\begin{aligned}
&\propto \exp\left\{\sum_{i=1}^{n}\left(\frac{1}{2}\log(\delta)-\delta\frac{(\nu_i-1)^2}{2\,\nu_i}\right)+(a-1)\log(\delta)-b\,\delta\right\}\\
&= \exp\left\{\left(\frac{n}{2}+a-1\right)\log(\delta)-\delta\left(\sum_{i=1}^{n}\frac{(\nu_i-1)^2}{2\,\nu_i}+b\right)\right\}\\
&\sim G\left(\frac{n}{2}+a,\sum_{i=1}^{n}\frac{(\nu_i-1)^2}{2\,\nu_i}+b\right). \qquad (5.21)
\end{aligned}$$

Hence a simple Gibbs step can be used to update δ in the POIG and ZIPIG models. If we take a look at this full conditional it is also clear why we just sample b as hyperparameter and not both a and b: a is not relevant because in normal cases $\frac{n}{2}$ will be much larger than a. But the sum $\sum_{i=1}^{n}\frac{(\nu_i-1)^2}{2\,\nu_i}$ can be very close to 0 when the ν_i are all close to 1 and they are supposed to have mean 1 a priori. So the parameter b may play an important role in the full conditional of δ.

For the other formulations we could not find such an argumentation by analyzing the full conditionals that justify sampling only for b. Nevertheless we decided to proceed in a similar way to unify the model formulations. The full conditional for b is similar in all the cases because it only depends on the prior for b and the prior for δ, and they remain the same for all models.

$$\begin{aligned}
\pi(b|\ldots) &\propto g(\delta|b)g(b)\\
&= \frac{b^a}{\Gamma(a)}\delta^{a-1}\exp\{-b\,\delta\}\frac{\alpha_2^{\alpha_1}}{\Gamma(\alpha_1)}b^{\alpha_1-1}\exp\{-\alpha_2\,b\}\\
&\propto \exp\{a\log(b)-b\,\delta+(\alpha_1-1)\log(b)-\alpha_2\,b\}\\
&\propto \exp\{(a+\alpha_1-1)\log(b)-(\delta+\alpha_2)\,b\}\\
&\propto G(a+\alpha_1,\delta+\alpha_2). \qquad (5.22)
\end{aligned}$$

Finally, we concentrate on the zero inflation parameter θ. First, we recover the notation of Section 3.1 and give a general structure for the full conditional of θ in zero inflated models.

$$\pi(\theta|\ldots) \propto \prod_{i=1}^{n}P(y_i|\cdot,\theta)g(\theta)$$

$$
\begin{aligned}
&= \prod_{y_i=0} \Big\{\theta + (1-\theta)P(y_i^{under} = 0|\cdot)\Big\} \prod_{y_i\neq 0} \Big\{(1-\theta)P(y_i^{under} = y_i|\cdot)\Big\} \\
&\propto \exp\left\{\sum_{y_i=0} \log\Big(\theta + (1-\theta)P(y_i^{under} = 0|\cdot)\Big) + Z_0 \log(1-\theta)\right\}, \quad (5.23)
\end{aligned}
$$

with $g(\theta) = 1$ as given in (4.4), and Z_0 as the number of nonzero counts in the data. Remember that with y_i we have denoted the observed count data outcome and with y_i^{under} the underlying count data process. For the last one we have chosen several options: Poisson, Poisson with latent variables and Negative Binomial.

Now it is easy to calculate the full conditional of θ in the different models. We only need to replace $P(y_i^{under} = 0|\cdot)$ by the corresponding count data distribution. We first examine the full conditional for the hierarchical group C (ZINB model), then for the group D (ZIPGA and ZIPIG models), and finally for the group E (ZIP model) separately.

In a ZINB model, the underlying count data distribution is a Negative Binomial. From (5.23) we get

$$
\begin{aligned}
\pi(\theta|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \delta, \theta) g(\theta) \\
&\propto \exp\left\{\sum_{y_i=0} \log\left(\theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) + Z_0 \log(1-\theta)\right\}. \quad (5.24)
\end{aligned}
$$

For the models in group D we need the probability of zero counts from a Poisson distribution with mean $\nu_i\mu_i$

$$
\begin{aligned}
\pi(\theta|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \boldsymbol{\nu}, \theta) g(\theta) \\
&\propto \exp\left\{\sum_{y_i=0} \log\Big(\theta + (1-\theta)\exp(-\nu_i\mu_i)\Big) + Z_0 \log(1-\theta)\right\}. \quad (5.25)
\end{aligned}
$$

Next we calculate the full conditional for θ in a ZIP model. Similar as before, we need the probability of zero counts of a Poisson distribution, but this time with mean given by μ_i.

$$
\begin{aligned}
\pi(\theta|\ldots) &\propto l(\boldsymbol{y}|\boldsymbol{\eta}, \theta) g(\theta) \\
&\propto \exp\left\{\sum_{y_i=0} \log\Big(\theta + (1-\theta)\exp(-\mu_i)\Big) + Z_0 \log(1-\theta)\right\}. \quad (5.26)
\end{aligned}
$$

These full conditionals have a rather complicated functional form in θ. In the next Section the corresponding M–H update steps will be presented.

5.3 Sampling Schemes

In this section we use the results obtained in Section 5.2 and the theory of Appendix C to find convenient update steps. Some general comments have already been made about how to proceed, but nothing has been said about the choice of proposal distributions, when needed. In the following we use θ^* to denote the proposed value for the parameter θ in the update step.

5.3.1 Predictor terms and their hyperparameters

Gamerman's IWLS proposals (Gamerman, 1997*a*) combine the likelihood and the prior information to approximate the full conditional of the parameters. They allow the update of parameters in a M–H step without any tuning. Thus convergence and mixing behavior of the chains using IWLS proposals is very satisfactory. For more information about IWLS proposals see Subsection C.2.4 in Appendix C.

We can give a general form of the IWLS proposals used for the update of the terms in the predictor almost irrespectively of the model we have chosen. For notational convenience we will use $\boldsymbol{v}$ to denote one of the parameter vectors $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{f}$ or $\boldsymbol{\rho}$. The proposed value $\boldsymbol{v}$ will be drawn from a multivariate normal distribution as follows:

$$\boldsymbol{v}^* \sim N(m(\boldsymbol{v}), \widetilde{M(\boldsymbol{v})}), \tag{5.27}$$

$M(\boldsymbol{v})$ is meant to be a precision matrix defined as

$$\begin{aligned} M(\boldsymbol{v}) &= F^{\mathrm{E}}(\boldsymbol{v}) + \frac{1}{\tau_v^2} K_v \\ &= X_v' W(\boldsymbol{v}) X_v + \frac{1}{\tau_v^2} K_v, \end{aligned}$$

where, $F^{\mathrm{E}}(\boldsymbol{v})$ is the corresponding block of the expected Fisher information matrix as given in (B.2) in Appendix B. For $\boldsymbol{v}$=$\boldsymbol{\gamma}$, $\boldsymbol{\rho}$ or $\boldsymbol{f}$, τ_v^2 and K_v are the elements of the prior for $\boldsymbol{v}$ as given in Subsections 4.2.2 and 4.2.3. Note that for $\boldsymbol{\beta}$ we have chosen a flat prior and therefore if $\boldsymbol{v}$=$\boldsymbol{\beta}$ then the form of $M(\boldsymbol{v})$ simplifies to $F^{\mathrm{E}}(\boldsymbol{v})$. The first equality is explained in Appendix B where we analyze the general form of the Fisher information matrix in our models. The components X_v represent the design matrices given in (4.16), (4.17), (4.18) and (4.19) for $\boldsymbol{v}$ equal $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\rho}$ or $\boldsymbol{f}$ respectively. And $W(\boldsymbol{v}) = \mathrm{diag}(w_i(v))$ is a diagonal weight matrix with entries given in Table 5.1 for the different model formulations.

The mean vector $m(\boldsymbol{v})$ is given by

$$\begin{aligned} m(\boldsymbol{v}) &= M(\boldsymbol{v})^{-1}\left(S(\boldsymbol{v}) + F^{\mathrm{E}}(\boldsymbol{v})\boldsymbol{v}\right) \\ &= \left(X_v' W(\boldsymbol{v}) X_v + \frac{1}{\tau_v^2} K_v\right)^{-1} \left(S(\boldsymbol{v}) + X_v' W(\boldsymbol{v}) X_v \boldsymbol{v}\right) \end{aligned}$$

noting that if $\boldsymbol{v} = \boldsymbol{\beta}$, we have a simplified form for $M(\boldsymbol{v})$. In this expression, $S(\boldsymbol{v})$ represents the score vector of the models, whose components are defined as $\sum_{i=1}^{n} \frac{\partial l_i}{\partial v_j}$ and are also given in Appendix B for all the models considered in this work.

From a computational point of view this proposal demands a great effort. Each step requires the sampling from a multivariate Gaussian distribution and for the densities $q(\boldsymbol{v}^* \to \boldsymbol{v})$ the calculating of the determinants of the $M(\boldsymbol{v})$ matrices are needed. This disadvantage is compensated by fast convergence and good mixing behavior of the obtained chains. Furthermore, with IWLS proposals we do not need any tuning for the variance of the proposal.

As the covariance matrix and the mean of the proposal depend on the current values through the weights the quotient

$$\frac{q(\boldsymbol{v}^* \to \boldsymbol{v})}{q(\boldsymbol{v} \to \boldsymbol{v}^*)}$$

does not simplify to one. The acceptance probability for the block $\boldsymbol{v}$, given in (C.5) for the general case, is

$$\alpha(\boldsymbol{v}, \boldsymbol{v}^*) = \min\left\{\frac{\pi(\boldsymbol{v}^*|\ldots) q(\boldsymbol{v}^* \to \boldsymbol{v})}{\pi(\boldsymbol{v}|\ldots) q(\boldsymbol{v} \to \boldsymbol{v}^*)}, 1\right\}, \tag{5.28}$$

	$y_i = 0$	$y \neq 0$
$w_i^{\text{NB}}(\boldsymbol{v})$	$\frac{\delta\mu_i}{\delta+\mu_i}$	$\frac{\delta\mu_i}{\delta+\mu_i}$
$w_i^{\text{PO}^*}(\boldsymbol{v})$	$\nu_i\mu_i$	$\nu_i\mu_i$
$w_i^{\text{ZIP}}(\boldsymbol{v})$	μ_i	$(1-\theta)\exp(-\mu_i)\mu_i\frac{\exp(l_i)-\mu_i\theta}{\exp(2l_i)}$
$w_i^{\text{ZIP}^*}(\boldsymbol{v})$	$\nu_i\mu_i$	$(1-\theta)\exp(-\nu_i\mu_i)\nu_i\mu_i\frac{\exp(l_i)-\nu_i\mu_i\theta}{\exp(2l_i)}$
$w_i^{\text{ZINB}}(\boldsymbol{v})$	$\frac{\delta\mu_i}{\delta+\mu_i}$	$(1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta+2}\mu_i\frac{\exp(l_i)-\mu_i\theta}{\exp(2l_i)}$

Table 5.1: Weights for the IWLS proposals

with $\pi(\boldsymbol{v}|\ldots)$ given by (5.8) if we are updating the fixed effects in the model and by (5.9) otherwise. The acceptance probabilities are typically quite high for this proposal.

Updating τ_v^2 is straightforward with a Gibbs step. The full conditional in (5.10) is proportional to an inverse gamma, and sampling proceeds as follows:

$$\tau_v^{2^*} \sim IG\left(a+\frac{1}{2}\text{rank}(K_v), \frac{1}{2}\boldsymbol{v}'K_v\boldsymbol{v}+b\right). \tag{5.29}$$

At this point we make some comments about the update of $\boldsymbol{\kappa}$, the vector of unit specific random effects in the POLN model, and its hyperparameter τ_κ^2. As it has a random effects prior, $\boldsymbol{\kappa}$ can be sampled following the same scheme as explained above. The design matrix X_κ is a nxn matrix and equal the identity matrix, so that we can write $X_\kappa = I_n$. For τ_κ^2 the Gibbs sampling step given in (5.29) is also valid.

5.3.2 Model specific parameters

The update of the parameter vector $\boldsymbol{\nu}$ is done by updating each component separately. Depending on the model definition we will use three different update schemes for these

parameters.

The first one is a Gibbs sampling step and is used for the POGA model. The full conditional for the i^{th} component of the vector is given by (5.15). Therefore to update ν_i we sample the new value

$$\nu_i^* \quad \sim \quad G(y_i + \delta, \mu_i + \delta) \tag{5.30}$$

and accept it as the next stage in the chain for ν_i.

The second and third update schemes for ν_i are used for the POIG and ZIPGA, ZIPIG respectively. They are both M–H update steps with uniform proposals. These are of course restricted to deliver only strictly positive values. The update steps differ from each other in the way of calculating the central point of the uniform proposal.

For the POIG model the full conditional for ν_i is given by (5.16), and it is not proportional to any known distribution. The aim is to find an appropriate proposal that improves the convergence of the chain. We decided to implement an uniform proposal with the maximum of the full conditional ν_i^{max} as central point. This value is calculated as follows. First, it is well known that maximizing the full conditional $\pi(\nu_i|\ldots)$ is equivalent to maximizing its logarithm. To find ν_i^{max} we differentiate $f(x) = \log(\pi(x|\ldots))$ with respect to x and calculate the zeros of $f'(x)$. We begin with the derivative of $f(x)$. Note that working with proportionalities does not affect the calculation of the maximum, because by differentiating the proportionality constants will disappear.

$$\begin{aligned} f(x) &= -x\,\mu_i + \left(y_i - \frac{3}{2}\right)\ln(x) - \delta\frac{(x-1)^2}{2\,x} \\ f'(x) &= -\mu_i + \left(y_i - \frac{3}{2}\right)\frac{1}{x} - \frac{\delta}{2}\left(1 - \frac{1}{x^2}\right) \\ &= -\left(\mu_i + \frac{\delta}{2}\right) + \left(y_i - \frac{3}{2}\right)\frac{1}{x} + \frac{\delta}{2x^2} \\ &= 0 \end{aligned}$$

Because $x > 0$, it is equivalent to find the zeros of the following second order polynomial

$$g(x) \quad = \quad \left(\mu_i + \frac{\delta}{2}\right)x^2 - \left(y_i - \frac{3}{2}\right)x - \frac{\delta}{2}$$

$$= \quad 0.$$

It is easy to see that the polynomial always cuts the axis of abscissae two times, independently of the values of y_i, δ and μ_i. Thus we find two solutions for this equation, but only the positive one is admissible for our problem. This solution is given by

$$\nu_i^{max} \quad = \quad \frac{y_i - 1.5 + \sqrt{(y_i - 1.5)^2 + \delta(2\mu_i + \delta)}}{2\mu_i + \delta} \tag{5.31}$$

The uniform proposal with central point in ν_i^{max} is then given by

$$\nu_i^* \quad \sim \quad U\left(\max\{\nu_i^{max} - p_i, 0\}, \nu_i^{max} + p_i\right), \tag{5.32}$$

to ensure that the proposed values are all positive. In (5.32), p_i is a sort of tuning parameter, that controls the acceptance rate of ν_i. The parameter p_i is chosen adaptively in the burnin phase in order to achieve a final rate between 30% and 60% (as explained in Section C.2). After each 100^{th} iteration in the burnin phase the acceptance rate for ν_i is calculated. Is this rate below 30%, the value of p_i is reduced, and if the rate is above 60%, p_i is incremented. The proposal density is given by

$$q(\nu_i \to \nu_i^*) \quad = \quad \frac{1}{\nu_i^{max} + p_i - \max\{\nu_i^{max} - p_i, 0\}}. \tag{5.33}$$

Since $q(\cdot)$ does not depend on ν_i and ν_i^*, we always have $q(\nu_i \to \nu_i^*) = q(\nu_i^* \to \nu_i)$, and herewith the acceptance probability for each ν_i^* simplifies to

$$\alpha(\nu_i, \nu_i^*) \quad = \quad \min\left\{\frac{\pi(\nu_i^*|\ldots)}{\pi(\nu_i|\ldots)}, 1\right\} \tag{5.34}$$

with $\pi(\nu_i|\ldots)$ from (5.16). This proposal has two main advantages. It is easy to implement and fast in the calculations. And the proposed ν_i^* values make sense because they have the current maximum of the full conditional as reference point, which improves the convergence of the chain.

Now we present the proposal for ν_i if we work with ZIPGA or ZIPIG models. As we said before, we have also chosen an uniform proposal, but in this case we can not easily

calculate the maximum of the full conditional and we prefer to fix the current value ν_i as the central point. Formally the proposal is given by

$$\nu_i^* \sim U\left(\max\{\nu_i - p_i, 0\}, \nu_i + p_i\right). \tag{5.35}$$

The parameters p_i's play the same role as explained before. The proposal density is

$$q(\nu_i \to \nu_i^*) = \frac{1}{\nu_i + p_i - \max\{\nu_i - p_i, 0\}}. \tag{5.36}$$

Note that in this case $q(\nu_i \to \nu_i^*) = q(\nu_i^* \to \nu_i)$ holds only if both ν_i and ν_i^* are greater that p_i. Otherwise we can not simplify the quotient in the acceptance probability and in general it will be

$$\alpha(\nu_i, \nu_i^*) = \min\left\{\frac{\pi(\nu_i^*|\ldots)q(\nu_i^* \to \nu_i)}{\pi(\nu_i|\ldots)q(\nu_i \to \nu_i^*)}, 1\right\} \tag{5.37}$$

This proposal is easy to implement but convergence may be slightly slower.

For the scale parameters δ we have two sorts of full conditionals. Those for the NB, ZINB, POGA and ZIPGA, were we have no closed form, and those for the POIG and ZIPIG models, were a closed form is found.

The full conditionals in the first group are calculated in (5.17), (5.18), and (5.20) respectively. For all these models a M–H step is necessary and thus we need a proposal for δ. We have implemented two options. Following the same idea as for ν_i, the first option is an uniform proposal. The construction of this proposal is similar in all the steps to the one presented in (5.35) and (5.36), but substituting ν_i and ν_i^* by δ and δ^* respectively and with the corresponding tuning parameter p_δ.

$$\delta^* \sim U\left(\max\{\delta - p_\delta, 0\}, \delta + p_\delta\right). \tag{5.38}$$

The second proposal is based on a gamma distribution. The parameters of this gamma proposal are functions of δ and p_δ so that its mean is the actual value δ and the variance is given by p_δ. Similar as for the ν_i's, p_δ is a sort of tuning parameter, which controls the

acceptance rate for δ and again is chosen adaptively in the burn in phase. The proposal distribution is

$$\begin{aligned} \delta^* &\sim G\left(\frac{\delta^2}{p_\delta}, \frac{\delta}{p_\delta}\right) \\ E(\delta^*) &= \delta \\ V(\delta^*) &= p_\delta \\ q(\delta \to \delta^*) &= \frac{(\frac{\delta}{p_\delta})^{\frac{\delta^2}{p_\delta}}}{\Gamma(\frac{\delta^2}{p_\delta})} \delta^{*\frac{\delta^2}{p_\delta}-1} \exp\{-\frac{\delta}{p_\delta}\delta^*\} \end{aligned} \tag{5.39}$$

The acceptance probability for both proposal options is the same, because in general $q(\delta \to \delta^*) = q(\delta^* \to \delta)$ does not hold for any of the options, so that the quotient does not simplify.

$$\alpha(\delta, \delta^*) = \min\left\{\frac{\pi(\delta^*|\ldots)q(\delta^* \to \delta)}{\pi(\delta|\ldots)q(\delta \to \delta^*)}, 1\right\} \tag{5.40}$$

with $\pi(\delta|\ldots)$ from (5.17) for the NB model, (5.18) for the ZINB model, or (5.20) for the POGA and ZIPGA models, and $q(\delta \to \delta^*)$ from (5.38) or (5.39).

The difference between the presented proposals for δ is not of great importance for the results, as we could expect from the M–H algorithm. The first ones works quite good and is fast in the computations. The second one respects the nature of δ as positive parameter but may lead two computational problems if the values for δ are quite close to zero and due to the gamma functions requires a greater computational effort. Therefore we have mostly worked with the first option.

In a POIG and ZIPIG formulations the full conditional for δ is proportional to a gamma distribution as given in (5.21). A Gibbs step is implemented by drawing

$$\delta^* \sim G\left(\frac{n}{2} + a, \sum_{i=1}^{n} \frac{(\nu_i - 1)^2}{2\,\nu_i} + b\right). \tag{5.41}$$

The update step for the parameter b is common for all model formulations. To update b we refer to its full conditional given in (5.22). It is proportional to a gamma distribution

and we use this fact to implement a Gibbs sampling for b through

$$b^* \quad \sim \quad G(a+\alpha_1, \delta+\alpha_2). \tag{5.42}$$

Finally, we analyze the update step for the zero inflation parameter θ. From the form of the full conditionals given in (5.24) through (5.26) we know that a M–H step is needed. The proposal distribution will be the same for all the zero inflated models. It have to respect the probability nature of θ, that means, only proposed values between zero and one have the chance to be accepted. We have implemented an uniform proposal that, with the help of some restrictions, overcomes this matter. Formally, we will sample the proposed values from

$$\theta^* \quad \sim \quad U\left(\max\{\theta - p_\theta, 0\}, \min\{\theta + p_\theta, 1\}\right), \tag{5.43}$$

with

$$q(\theta \to \theta^*) \quad = \quad \frac{1}{\min\{\theta + p_\theta, 1\} - \max\{\theta - p_\theta, 0\}}. \tag{5.44}$$

Note that $q(\theta \to \theta^*) = q(\theta^* \to \theta)$ only holds when $1 - p_\theta < \theta, \theta^* < p_\theta$. Hence, in general the acceptance probability does not simplify and remains

$$\alpha(\theta, \theta^*) \quad = \quad \min\left\{\frac{\pi(\theta^*|\ldots)q(\theta^* \to \theta)}{\pi(\theta|\ldots)q(\theta \to \theta^*)}, 1\right\} \tag{5.45}$$

with $\pi(\theta|\ldots)$ as given in a general form in (5.23) and $q(\theta^* \to \theta)$ from (5.44).

5.4 Algorithms

We summarize with an overview of the sampling algorithms for each of the nine models. To simplify the representation we always refer to the proposals and acceptance probabilities given in the last sections.

An important matter for convergence of the chain are the starting values. For the terms in the predictor the starting values are the posterior mode estimates (Brezger and Lang, 2003).

For the parameter vector $\boldsymbol{\nu}$ we take a vector of length n with 1 in all the entries, that is their prior mean.

For the parameter δ we have experimented with the starting value

$$\delta^{(0)} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n}(y_i - \hat{\mu}_i) - \sum_{i=1}^{n} y_i} \tag{5.46}$$

(Cameron and Trivedi, 1998) with $\hat{\mu}_i$ obtained from the posterior mode estimation, but we did not obtained satisfactory results. Clearly, in (5.46) it is not always sure that $\delta > 0$. So we have set $\delta^{(0)} = 0.1$ for all runs.

For θ we have chosen

$$\theta^{(0)} = \frac{n - Z_0}{n}$$

with Z_0 the number of nonzero counts in the data. Hence, $\theta^{(0)}$ is the proportion of zero counts in the data. Of course this starting value will be better for a large mean of the underlying count data distribution than for a small one. However convergence of the chain seems to remain unaffected by this fact.

In the following we set the number of iterations to J and use j to denote each of them. The number of items in our data set is n, that is at the same time the length of the vector of multiplicative random effects $\boldsymbol{\nu}$. We suppose that in our model there are some fixed effects, a linear covariate modeled through a P–spline, a random effect and we have geographical information. Of course in real data applications we may have more than one of these types, but extension to this case is straightforward.

Note that the update steps for the terms in the predictor are similar in its algorithmic structure for all the models. Thus we are only going to describe them for the NB model and then refer to them for the other models. The sampling step of b remains the same with a Gibbs step for all the models.

We begin with the NB model. Its sampling algorithm is given below.

NB model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_{\gamma}^{2(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_{f}^{2(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_{\rho}^{2(0)}$, $\delta^{(0)}$, $b^{(0)}$ and set $j = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{f}$ and $\boldsymbol{\rho}$ with **M--H step**
   (a) Sample $v^* \sim N(\cdot, \cdot)$ as in (5.27)
   (b) $v^{(j+1)} = v^*$ with probability $\alpha\left(v^{(j)}, v^*\right)$ given by (5.28), otherwise let $v^{(j+1)} = v^{(j)}$
   (c) If $v \neq \boldsymbol{\beta}$: update τ_v^2 with **Gibbs step**
   Sample $\tau_v^{2(j+1)} \sim IG(\cdot, \cdot)$ as in (5.29)
4. Update δ with **M--H step**
   (a) Sample $\delta^* \sim U(\cdot, \cdot)$ as in(5.38) or $\delta^* \sim G(\cdot, \cdot)$ as in (5.39)
   (b) $\delta^{(j+1)} = \delta^*$ with probability $\alpha\left(\delta^{(j)}, \delta^*\right)$ given by (5.40) otherwise let $\delta^{(j+1)} = \delta^{(j)}$
5. Update b with **Gibbs step**
   Sample $b^{(j+1)} \sim G(\cdot, \cdot)$ as in (5.42)
6. Go to 2. till $j = J$

Note that we give two possibilities for the update of δ, because both are described above, but only one is used in the practice.

For the ZINB model the algorithm is similar as for the NB model. The difference is given by the initialization and introduction of the update step for the zero inflation parameter θ. The position where we introduce it is not relevant for the algorithm.

ZINB model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_{\gamma}^{2(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_{f}^{2(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_{\rho}^{2(0)}$, $\delta^{(0)}$, $b^{(0)}$, $\theta^{(0)}$ and set $j = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{f}$ and $\boldsymbol{\rho}$ with **M--H step**: like NB model.
4. Update δ with **M--H step**: like NB model.
5. Update b with **Gibbs step**: like NB model.

6. Update θ with **M--H step**
   (a) Sample $\theta^* \sim U(\cdot,\cdot)$ as in (5.43)
   (b) $\theta^{(j+1)} = \theta^*$ with probability $\alpha\left(\theta^{(j)},\theta^*\right)$ given by (5.45)
7. Go to 2. till $j = J$

For the POGA model we must incorporate the sampling of the components for the parameter vector $\boldsymbol{\nu}$. This is done through a Gibbs step for each ν_i, as explained in the last section.

POGA model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_\gamma^{2\,(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_f^{2\,(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_\rho^{2\,(0)}$, $\boldsymbol{\nu}^{(0)}$, $\delta^{(0)}$, $b^{(0)}$ and set $j, i = 0$
2. Set $j = j+1$
3. Update $v = \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}$ with **M--H step**: like NB model
4. Update $\boldsymbol{\nu}$ with **Gibbs step**
   (a) Set $i = i+1$
   (b) Sample $\nu_i^{(j+1)} \sim G(\cdot,\cdot)$ as in (5.30)
   (c) Go to 5.a if $i < n$. Otherwise set $i = 0$
5. Update δ with **M--H step**: like NB model
6. Update b with **Gibbs step**: like NB model
7. Go to 2. till $j = J$

For the update of the ν_i in the ZIPGA model we can not use a Gibbs step similar as in the POGA model, so a M–H step is needed. In addition we also have the initialization and update step of the zero inflation parameter θ.

ZIPGA model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_\gamma^{2\,(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_f^{2\,(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_\rho^{2\,(0)}$, $\boldsymbol{\nu}^{(0)}$, $\delta^{(0)}$, $b^{(0)}$, $\theta^{(0)}$ and set $j, i = 0$
2. Set $j = j+1$

3. Update $v = \beta, \gamma, f, \rho$ with **M--H step**: like NB model
4. Update ν with **Gibbs step**
   (a) Set $i = i + 1$
   (b) Sample $\nu_i^{(j+1)} \sim G(\cdot, \cdot)$ as in (5.35)
   (c) $\nu_i^{(j+1)} = \nu_i^*$ with probability $\alpha\left(\nu_i^{(j)}, \nu_i^*\right)$ given by (5.37) otherwise let $\nu_i^{(j+1)} = \nu_i^{(j)}$
   (d) Go to 4.a if $i < n$. Otherwise set $i = 0$
5. Update δ with **M--H step**: like NB model
6. Update b with **Gibbs step**: like NB model
7. Update θ with **M--H step**: like ZINB model
8. Go to 2. till $j = J$

The algorithm for the POIG model differs from the POGA one in the sampling method for ν and δ. This time we loose the Gibbs sampling for ν and use a componentwise M–H step to update the components ν_i based on the maximum of the full conditional. On the other side we can take advantage of a Gibbs step for the sampling δ.

POIG model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_\gamma^{2(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_f^{2(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_\rho^{2(0)}$, $\boldsymbol{\nu}^{(0)}$, $\delta^{(0)}$, $b^{(0)}$ and set $j, i = 0$
2. Set $j = j + 1$
3. Update $v = \beta, \gamma, f, \rho$ with **M--H step**: like NB model
4. Update ν with **M--H step**
   (a) Set $i = i + 1$
   (b) Sample $\nu_i^* \sim G(\cdot, \cdot)$ as in (5.32)
   (c) $\nu_i^{(j+1)} = \nu_i^*$ with probability $\alpha\left(\nu_i^{(j)}, \nu_i^*\right)$ given by (5.34) otherwise let $\nu_i^{(j+1)} = \nu_i^{(j)}$
   (d) Go to 4.a if $i < n$. Otherwise set $i = 0$
5. Update δ with **Gibbs step**
   Sample $\delta^{(j+1)} \sim G(\cdot, \cdot)$ as in (5.41)

6. Update b with **Gibbs step**: like NB model
7. Go to 2. till $j = J$

For the ZIPIG we change the update step for the ν_i parameters. Now we can not easily calculate the maximum of the full conditionals, and therefore we use a similar M–H step as in the ZIPGA model, taking the current value as central point for the proposal. Additionally, we introduce the zero inflation parameter θ in the algorithm, with a similar M–H update step as for the ZINB.

ZIPIG model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_\gamma^{2\,(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_f^{2\,(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_\rho^{2\,(0)}$, $\boldsymbol{\nu}^{(0)}$, $\delta^{(0)}$, $b^{(0)}$, $\theta^{(0)}$ and set $j, i = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}, \boldsymbol{\gamma}, f, \boldsymbol{\rho}$ with **M--H step**: like NB model
4. Update $\boldsymbol{\nu}$ with **M--H step**: like ZIPGA model
5. Update δ with **Gibbs step**: like POIG model
6. Update b with **Gibbs step**: like NB model
7. Update θ with **M--H step**: like ZINB model
8. Go to 2. till $j = J$

In the ZIP model we do not have overdispersion terms but the zero inflation parameter remains in the model. The algorithm is given by:

ZIP model

1. Initialize $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\gamma}^{(0)}$, $\tau_\gamma^{2\,(0)}$, $\boldsymbol{f}^{(0)}$, $\tau_f^{2\,(0)}$, $\boldsymbol{\rho}^{(0)}$, $\tau_\rho^{2\,(0)}$, $\theta^{(0)}$ and set $j = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}, \boldsymbol{\gamma}, f, \boldsymbol{\rho}$, with **M--H step**: like NB model
4. Update θ with **M--H step**: like ZINB model

5. Go to 2. till $j = J$

We consider now the sampling algorithm for the POLN model. As explained in Section 5.2 the vector $\boldsymbol{\kappa}$ can be sampled analogously as the terms $\boldsymbol{\gamma}$, $\boldsymbol{f}$ and $\boldsymbol{\rho}$. Thus we can simplify the representation of the algorithm and put all these terms together in the fourth step.

POLN model

1. Initialize $\boldsymbol{\beta}^{(\mathbf{0})}$, $\boldsymbol{\gamma}^{(\mathbf{0})}$, $\tau_{\gamma}^{2^{(0)}}$, $\boldsymbol{f}^{(\mathbf{0})}$, $\tau_{f}^{2^{(0)}}$, $\boldsymbol{\rho}^{(\mathbf{0})}$, $\tau_{\rho}^{2^{(0)}}$, $\boldsymbol{\kappa}^{(\mathbf{0})}$, $\tau_{\kappa}^{2^{(0)}}$ and set $j = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \boldsymbol{\kappa}$ with **M--H step**: like NB model
4. Go to 2. till $j = J$

Finally, if we are working a ZIPLN model, we can consider it as an extension of a ZIP with random effects for each item. Therefore, we extend the algorithm of the ZIP to have a new vector $\boldsymbol{\kappa}$ which is sampled analogously as in the POLN model, as explained before.

ZIPLN model

1. Initialize $\boldsymbol{\beta}^{(\mathbf{0})}$, $\boldsymbol{\gamma}^{(\mathbf{0})}$, $\tau_{\gamma}^{2^{(0)}}$, $\boldsymbol{f}^{(\mathbf{0})}$, $\tau_{f}^{2^{(0)}}$, $\boldsymbol{\rho}^{(\mathbf{0})}$, $\tau_{\rho}^{2^{(0)}}$, $\boldsymbol{\kappa}^{(\mathbf{0})}$, $\tau_{\kappa}^{2^{(0)}}$, θ and set $j = 0$
2. Set $j = j + 1$
3. Update $v = \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{f}, \boldsymbol{\rho}, \boldsymbol{\kappa}$ with **M--H step**: like NB model
4. Update θ with **M--H step**: like ZINB model
5. Go to 2. till $j = J$

Chapter 6

Simulation studies

The aim of this study is to explore the performance of the proposed methodology for complex predictor structures, similar to those which will be used in the real data application in the next chapter. In particular, we will investigate how well different components in the predictor can be identified and separated from each other.

As a goodness of fit measure for single components in the predictor, we use their relative mean square errors (MSE). If f is an effect on x with K different values, its MSE_f is defined as

$$\text{MSE}_f = \sqrt{\frac{\sum_{k=1}^{K}(\hat{f}_k - f_k)^2}{\sum_{k=1}^{K} f_k^2}}$$

with $\hat{f}_k$ the estimated value for f_k.

First we test the models on data that fulfill the model assumptions. Then we investigate how robust these models are if the data generating process is not the same as supposed by the model.

6.1 Overdispersion

In this section, the proposed models for overdispersion are tested in the presence of complicated predictor structures like the ones we are going to find in a real data situation. Therefore, the simulation study is conducted with a covariate situation similar to the structure of the car insurance data. The overdispersion component should be recognized as well as the individual specific random effects, when they are present.

6.1.1 Data simulation

We generate data sets from POGA and POIG models with $\mu_i = \nu_i \exp(\eta_i)$, where ν_i are Gamma and Inverse Gaussian distributed respectively, and from POLN model with $\mu_i = \nu_i \exp(\eta_i) = \exp(\eta_i + \kappa_i)$, where κ_i are Gaussian random effects. The predictor η_i is the same for all three models and is defined by

$$\eta_i = o_i + \alpha + \beta z_i + \sin(x_i) + \rho_{g_i} + f_{str}(s_i) + f_{unstr}(s_i), \tag{6.1}$$

for $i = 1, ..., 1920$. The offsets o_i are obtained by i.i.d. sampling from a uniform distribution on the interval [3,6]. The values z_i are obtained as i.i.d. samples from a binary random variable $z \sim B(1; 0.5)$. The intercept and slope are $\alpha = -5$ and $\beta = 0.5$.

The realizations of the metrical covariate x are the 26 knots of an equidistant grid on the interval [-3,3]. The observations x_i, $i = 1, ..., 1920$, are generated by systematically repeating these 26 values until 1920 observations are reached. The nonlinear effect $f(x)$ of x is assumed to be a sine–curve $f(x) = \sin(x)$.

The covariate ρ represents a group indicator, as for the covariate type class of car in our car insurance application. It has 7 levels $g = 1, ..., 7$, with 7 equidistant effects

$$\rho_{(1)} = -0.3, \quad \rho_{(2)} = -0.2, \quad ..., \quad \rho_{(6)} = 0.2, \quad \rho_{(7)} = 0.3.$$

The observations ρ_{g_i}, $i = 1, ..., 1920$, are generated as a random sample from these values.

The structured spatial effects $f_{str}(s_i)$ are evaluations of the function

$$\begin{aligned} f_{str} : \mathbb{R}^2 &\longrightarrow \mathbb{R} \\ s = (u, v) &\longmapsto c_0 \sin(5\, u\, v) - c_1 \end{aligned}$$

at the coordinates $s_i = (u_i, v_i), i = 1, ..., 96$, of the standardized centroids of the 96 districts in Bavaria. The normalizing constants c_0 and c_1 are chosen so that the function values are centered about 0 and have approximate empirical variance 0.25. These structured spatial effects $f_{str}(s)$, $s = 1, ..., 96$, are visualized in Figure 6.12. For each district i, we assign $f_{str}(s_i)$ to 20 observations.

To generate the unstructured effects $f_{unstr}(s_i)$, we draw $f_{unstr}(s)$, $s = 1, ..., 96$, as an i.i.d. sample from $N(0, \tau^2)$. Then these values are assigned to the same 20 observations per district as in the case of structured spatial effects. To investigate the impact of unstructured effects, we generate data for three values

$$\tau^2 = 0, \quad \tau^2 = 0.01, \quad \tau^2 = 0.25$$

of the variance τ^2, corresponding to no, small and large unstructured effects. For $\tau^2 = 0.25$, the unstructured effects have the same variability as the structured effects. A particular reason for this choice is that we want to see whether f_{str} and f_{unstr} can be separately identified in the sum

$$f_{spat} = f_{str} + f_{unstr}$$

of total spatial effects.

The random effects ν_i, $i = 1, ..., 1920$, for the POGA and POIG model are obtained as i.i.d. samples from a $G(\delta, \delta)$ or a $IGaussian(1, \delta)$ distribution respectively with

$$\begin{aligned} E(\nu_i) &= 1 \\ V(\nu_i) &= \frac{1}{\delta} \end{aligned}$$

in both cases. In a similar way as for f_{unstr}, we generate data for

$$\delta = 0.5, \quad \delta = 1, \quad \delta = 2$$

corresponding to large, medium and small individual specific effects. Random effects in the POLN model are obtained as i.i.d. samples

$$\kappa_i = \log(\nu_i) \sim N\left(-\frac{\tau_\kappa^2}{2}, \tau_\kappa^2\right)$$

with

$$\tau_\kappa^2 = \log\left(1+\frac{1}{\delta}\right),$$

leading to

$$\tau_\kappa^2 = 1.098, \quad \tau_\kappa^2 = 0.6931, \quad \tau_\kappa^2 = 0.4055.$$

Then the log–normal effects have

$$\begin{aligned} E(\nu_i) &= 1 \\ V(\nu_i) &= \frac{1}{\delta} \end{aligned}$$

just as the Gamma or Inverse Gaussian random effects. Combining the possible values of the variance τ^2 of the unstructured spatial effects with those of the scale parameter δ, we obtain data for 9 different NB, POGA, POIG and POLN models. For the discussion of simulation results, we denote them by $M(\tau^2;\delta)$. For example, $M(0;1)$ is a (NB, POGA, POIG or POLN) model without $(\tau^2 = 0)$ unstructured spatial effects and individual random effects with medium $(\delta = 1)$ variability, and $M(0.25;2)$ is a model with high variability $(\tau^2 = 0.25)$ of unstructured spatial effects and low $(\delta = 2)$ variability of individual random effects. With this simulation design, we can assess the impact of the relative magnitude of spatial and individual random effects on estimation of the various components.

For each model, we generate counts

$$\{y_i^{(r)},\ i = 1, ..., 1920\},$$

for simulation runs $r = 1, ..., R = 100$. For each simulation run r, we calculate posterior means, standard deviations, quantiles and the DIC criterion (see Section C.3). From $R =$

100 simulation runs, we obtain then measures such as: overall empirical bias, MSE, box plots etc. for the estimates of all unknown parameters and functions.

6.1.2 Results

This subsection consists of two blocks. In the first block we will show the results for the POIG and POIGH models and in the second one the results for the NB, POGA and POLN models. The reason for this partition is that the results for the POIG model are not as satisfactory as we expected, and we try to improve them by introducing the POIGH model. The other models work quite well and hence we present and compare their results together at the end of this subsection.

POIG

As said before, the POIG model did not fit the simulated data good enough. In the following we present some results and some attempts to improve these results. We can say in advance that the efforts did not lead to any significant improvements.

1. The POIG model failed in the estimation of both δ and $\boldsymbol{\nu}$. The fact that the failure affects both parameter blocks is a natural consequence of the hierarchical structure of the model. The main problem is that δ is always overestimated and therefore $\boldsymbol{\nu}$ has not enough variability on the prior assumption to reach the original values.

   In Figure 6.1 we show box plots for the estimated posterior mean values for δ from the different simulated models $M(\cdot;\cdot)$. In the optimal case, the first group of three box plots should be placed around the reference line $\delta = 0.5$, the second one around the line $\delta = 1$ and the third one around $\delta = 2$. But here we have a different situation. The model has difficulties to find the overdispersion in the data for all tested values of δ delivering in all the cases larger posterior mean estimates than the orginal values.

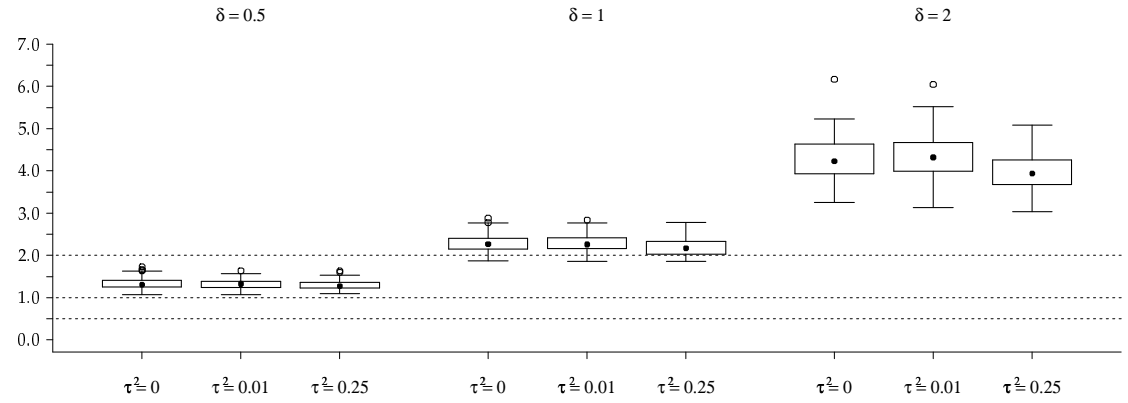

Figure 6.1: Box Plots for posterior means of δ of the simulation results for each POIG model. Plotted are also the reference lines $\delta = 0.5, 1$ and 2.

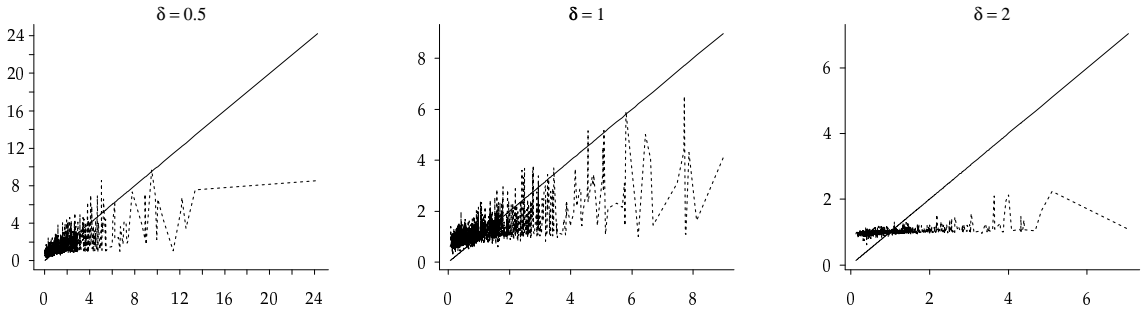

Figure 6.2: Diagonal plots for $\boldsymbol{\nu}$ (true versus estimated effects) obtained for selected models from M(0.25; $\cdot$).

It is evident how overestimation of the scale parameter affects the individual specific random effects. They are not able to jump to the original value and remain near to 1, their prior mean. See for example Figure 6.2.

The conclusion is that δ is always overestimated, which means that not all the overdispersion in the data is recognized by the model.

We have tried to improve these results by implementing other proposal distributions for $\boldsymbol{\nu}$ that may draw better candidates for the chain, and by implementing a

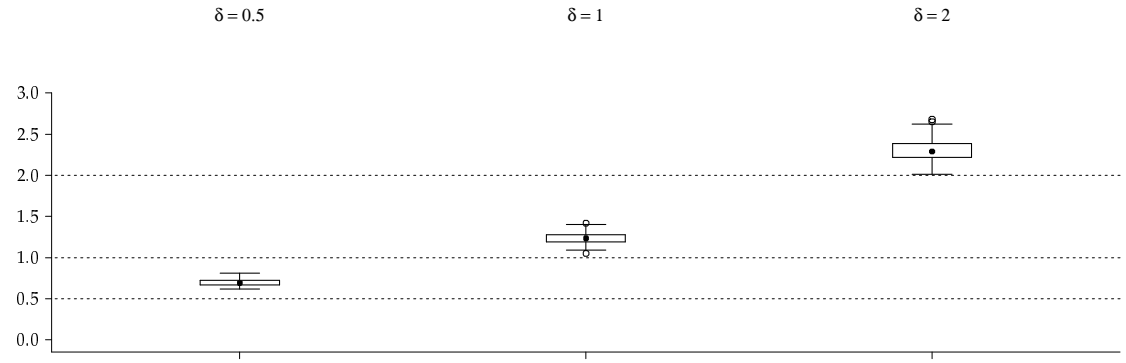

Figure 6.3: Box Plots for posterior means of δ of the simulation results with simple linear predictor for each POIG model. Plotted are also the reference lines $\delta = 0.5, 1$ and 2.

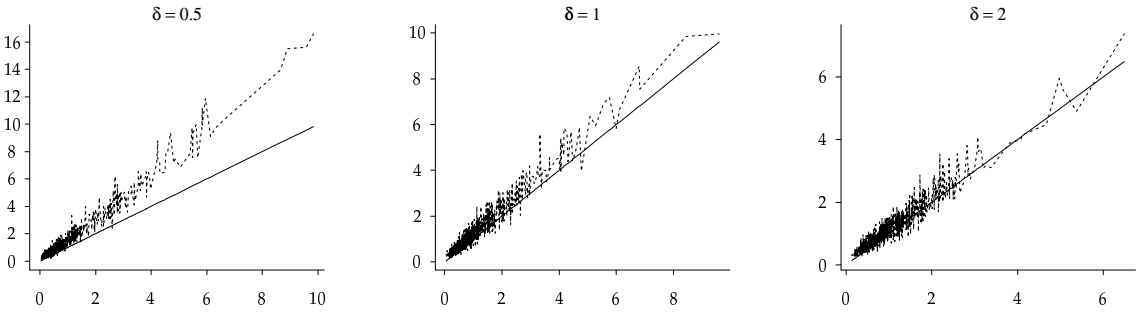

Figure 6.4: Diagonal plots for $\boldsymbol{\nu}$ (true versus estimated effects) obtained from the POIG model for selected models with simple linear predictor.

M–H step for δ (instead of the Gibbs step), but we have found no differences in the results.

2. If we take a look at the bibliography referring to POIG models in Chapter 2, we see that none of the papers uses such complicated predictor structures in the models, as we did. Hence we have simulated three new data sets, all of them with the same simple linear predictor

$$\eta_i \quad = \quad \alpha + \beta z_i, \tag{6.2}$$

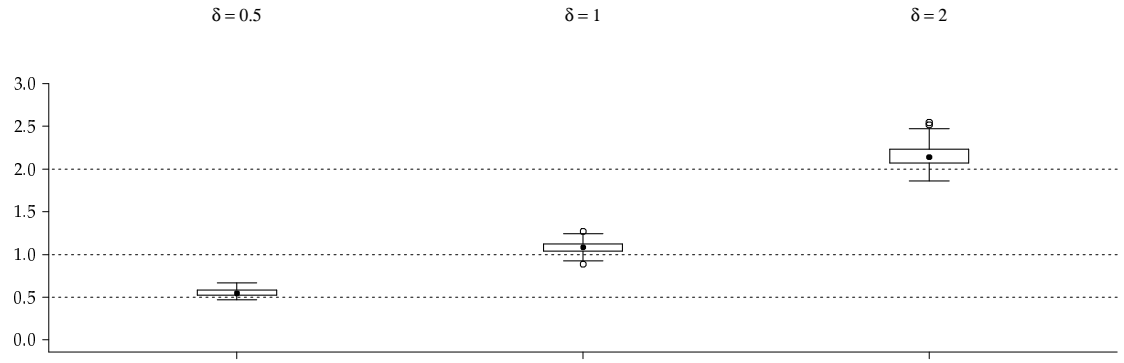

Figure 6.5: Box Plots for the posterior means of δ of the simulation results with simple linear predictor for each POIGH model. Plotted are also the reference lines $\delta = 0.5, 1, 2$.

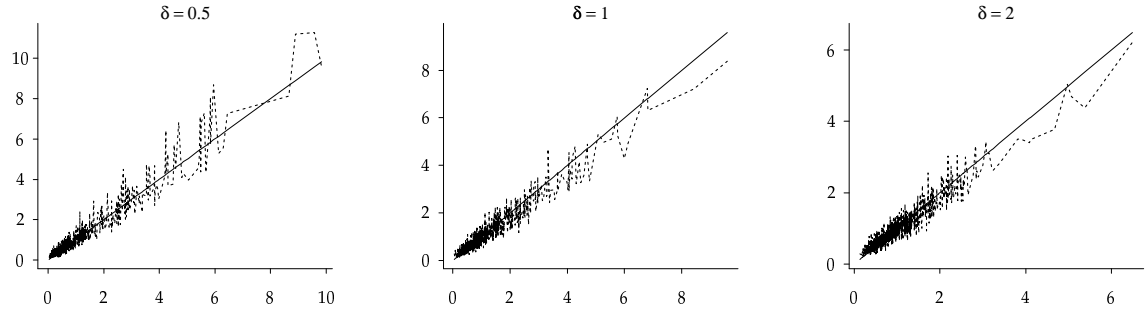

Figure 6.6: Diagonal plots for $\boldsymbol{\nu}$ (true versus estimated effects) obtained from the POIGH model for selected models with simple linear predictor.

where $\alpha = 3$, $\beta = 0.5$ and $z_i \sim B(1, 0.5)$. The dispersion parameter takes the values $\delta = 0.5, 1$ or 2. The results obtained from the POIG model applied to these data sets are much better than the results for data with a more complicated predictor structure presented before.

In Figure 6.3 we see box plots for the estimated posterior mean values for δ and the reference lines for $\delta = 0.5$, $\delta = 1$ and $\delta = 2$. We can see here that the box plots are better placed compared to Figure 6.1, although they are still not optimal at all.

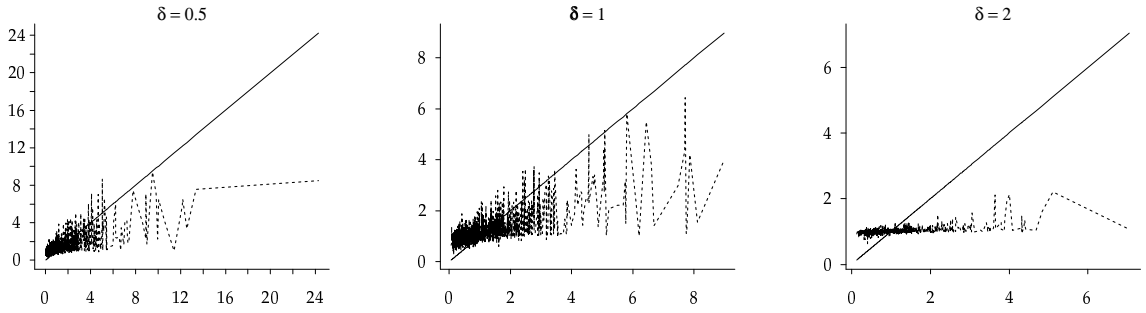

Figure 6.7: Diagonal plots for $\boldsymbol{\nu}$ (true versus estimated effects) obtained from the POIGH model for selected models $M(0.25; \cdot)$.

We can also confirm improvements in Figure 6.4. The diagonal plots for selected models show that the larger δ, the better the posterior mean estimates for $\boldsymbol{\nu}$.

3. A new idea is to reparameterize the POIG model by modifying its hierarchical structure, as explained in Section 2.3. The results for the POIGH model with a simple linear predictor are presented in Figures 6.5 and 6.6. The first figure shows a substantial improvement in the box plots for δ. All of them indicate some bias, implying a small overestimation for δ, but the bias is much smaller compared to the POIG case in Figure 6.3. In the second figure we have an absolutely better alignment of the estimates for $\boldsymbol{\nu}$ to the diagonal line, which means an important improvement with respect to the POIG model. Particularly for the small δ a better performance is evident.

4. Finally, we assert the performance of the POIGH model on data with complicated predictor structure. We have tried the POIGH model on the $M(\cdot, \cdot)$, described at the beginning of this chapter. Surprisingly the results are even worse as those of the POIG model. We had numerical problems with running the algorithm of most of the models. After some trials, we found out that for large and medium δ the POIGH model was generally not able to achieve convergence for the dispersion

parameter. So we can not show the figure corresponding to 6.1 for the POIGH model. In Figure 6.7 we have the plot for the POIGH model equivalent to Figure 6.2. As we can see comparing both plots, there is no correction at all for the results using POIGH instead of POIG if the predictor structure of the data is complex and not only linear. And as said before, in general the estimates are even worse as those of the POIG model.

Other models

Some important results arise from this simulation study with the NB, POGA, POGAH and POLN models applied to the $M(\cdot;\cdot)$ data.

1. The POGA model and its hierarchical version POGAH do not differ in their results applied to the same data sets. Therefore we will show only results from the POGA model.

2. Results for NB and POGA models applied to the same data sets are virtually indistinguishable. Therefore, if one is not interested in the latent individual random effects, a NB model may be preferable. Also, computation time and storage requirements may be an issue, depending on the sample size. The computation time for POGA is lower than for the NB, due to the gamma functions that have to be implemented for the latter model. However, this difference obviously decreases while increasing the number of observations in the data set, because in consequence this also increases the number of parameters to be estimated in the POGA model.

3. Unknown fixed effects α, β and the scale parameter δ are estimated very well, regardless of the specific model. This is illustrated for a sample of models in Table 6.1. Note that for the POLN model we have other reference values as for the NB and POGA ones. Remember that if $\delta = 0.5, 1$ or 2 then $\tau_\kappa^2 = 1.0986, 0.6931$ or 0.4055 respectively. Note also that the intercept α is inflated for the results of the POLN

	$M(0.01;1)$		$M(0.25;1)$		$M(0.25;2)$	
NB						
α	-5.02	(0.0543)	-5.008	(0.0551)	-5.001	(0.0473)
β	0.511	(0.0706)	0.518	(0.0716)	0.504	(0.0591)
δ	1.028	(0.0764)	0.979	(0.0726)	2.013	(0.1846)
POGA						
α	-5.019	(0.0543)	-5.009	(0.0558)	-5.002	(0.0475)
β	0.512	(0.0705)	0.515	(0.0719)	0.502	(0.0591)
δ	1.029	(0.0771)	0.981	(0.0718)	2.013	(0.1841)
POLN						
α	-5.352	(0.101)	-5.341	(0.1014)	-5.209	(0.0999)
β	0.499	(0.0662)	0.507	(0.0666)	0.502	(0.0571)
τ_κ^2	0.687	(0.0567)	0.693	(0.0574)	0.400	(0.0391)

Table 6.1: Posterior means and standard deviations (in brackets) for selected models.

model. The reason for this fact is that the Gaussian distributed random effects $\boldsymbol{\kappa}$ have not mean 0 but $-0.5\,\tau_\kappa^2$ and therefore this term is integrated in the intercept during estimation, as we see in the table.

4. Estimating the nonlinear sine curve $f(x) = \sin(x)$, see Figures 6.8 and 6.9, works also very well for all the models. A reason for this obviously quite stable identification of both fixed effects and the nonlinear effect of the metrical covariate x is that the priors are rather different from the priors for the remaining effects, which supports separation from the latter ones.

5. The effects ρ_g of the group indicator g can still be estimated quite well, but they seem to be more sensitive to the specific model. Figure 6.10 displays box plots of mean square errors for the 9 models. It seems that variation of the scale parameters

has some impact, while results are comparably insensitive to variations in dispersion of unstructured spatial effects. Figure 6.11 shows true effects (dot lines) and averaged posterior mean effects for selected models together with pointwise 10%– and 90%–posterior credible intervals. We can observe a shrinkage effect towards zero which becomes larger for smaller δ, i.e. larger individual random effects. Comparing POGA with POLN, the lastest seems to fit better the random effect ρ_g.

6. Separation of structured and unstructured spatial effects is generally very unreliable. In particular, unstructured spatial effects are always underestimated, partly to a large extent. Obviously their influence is already captured by structured spatial and by individual effects. This can be particularly well recognized in the 'diagonal plots' of Figures 6.12 to 6.15 where true and estimated unstructured random effects are plotted against each other. Ideally, the scatter plots should be near to the diagonal, but they are almost horizontal for the unstructured effects! For models with no $(\tau^2 = 0)$ or small $(\tau^2 = 0.01)$ unstructured effects, the structured spatial effects are still recovered satisfactorily (Figures 6.12 and 6.13). For models $M(0.25, \cdot)$, where variability of structured and unstructured effects is the same, most of unstructured spatial variability is captured by overestimating structured spatial effects, see Figure 6.14, 6.15 and 6.16.

   However, as Figure 6.17 shows, it makes always sense to include structured and unstructured effects, because the sum

$$f_{spat} = f_{str} + f_{unstr} \tag{6.3}$$

   has always the lowest MSE. Of course, then only the total spatial effects f_{spat} can be interpreted.

6.1.3 Résumé

We give a short overview of the consequences drawn from this simulation study.

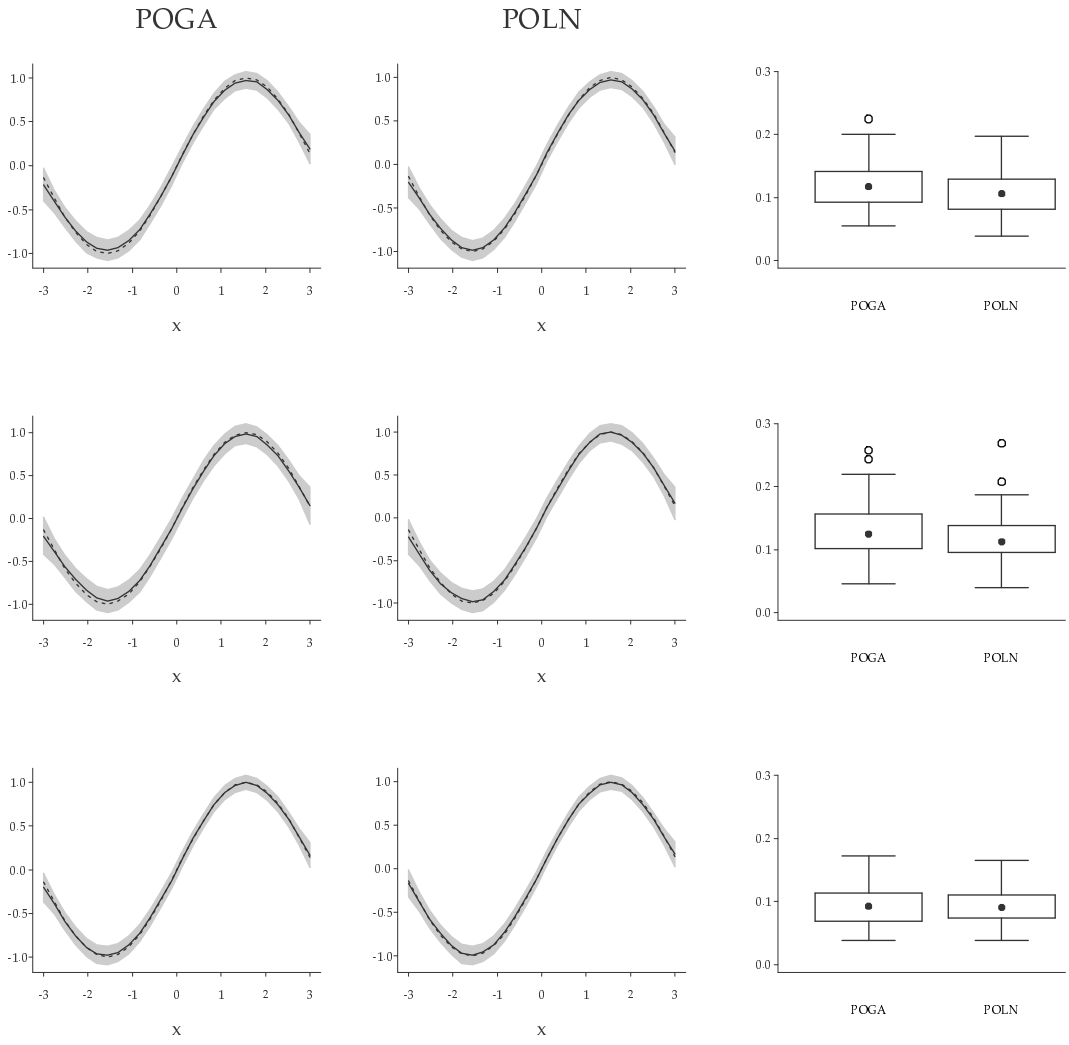

Figure 6.8: Average posterior mean estimates with pointwise 80% credible interval and MSE Box Plots for the nonlinear sine–term of models $M(0.01;1)$ (top), $M(0;0.5)$ (center) and $M(0.01;2)$ (bottom).

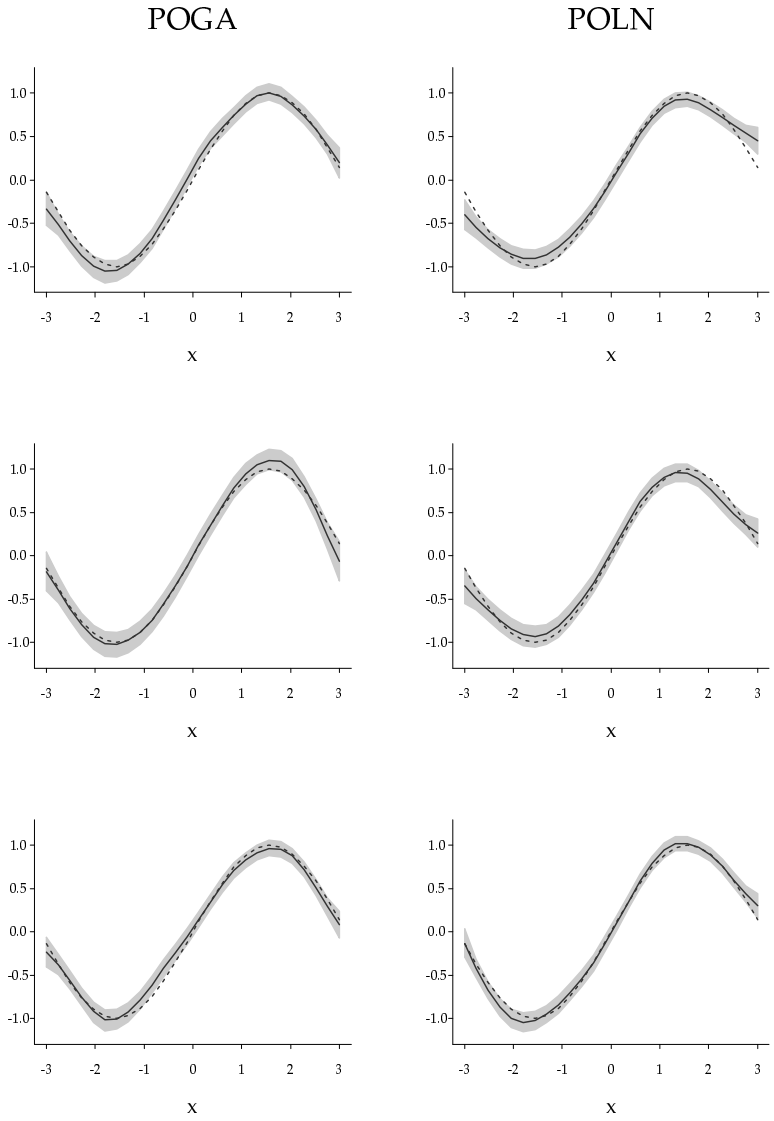

Figure 6.9: Selected estimates with pointwise 80% credible interval for the nonlinear sine-term of models $M(0.01;1)$ (top), $M(0;0.5)$ (center) and $M(0.01;2)$ (bottom).

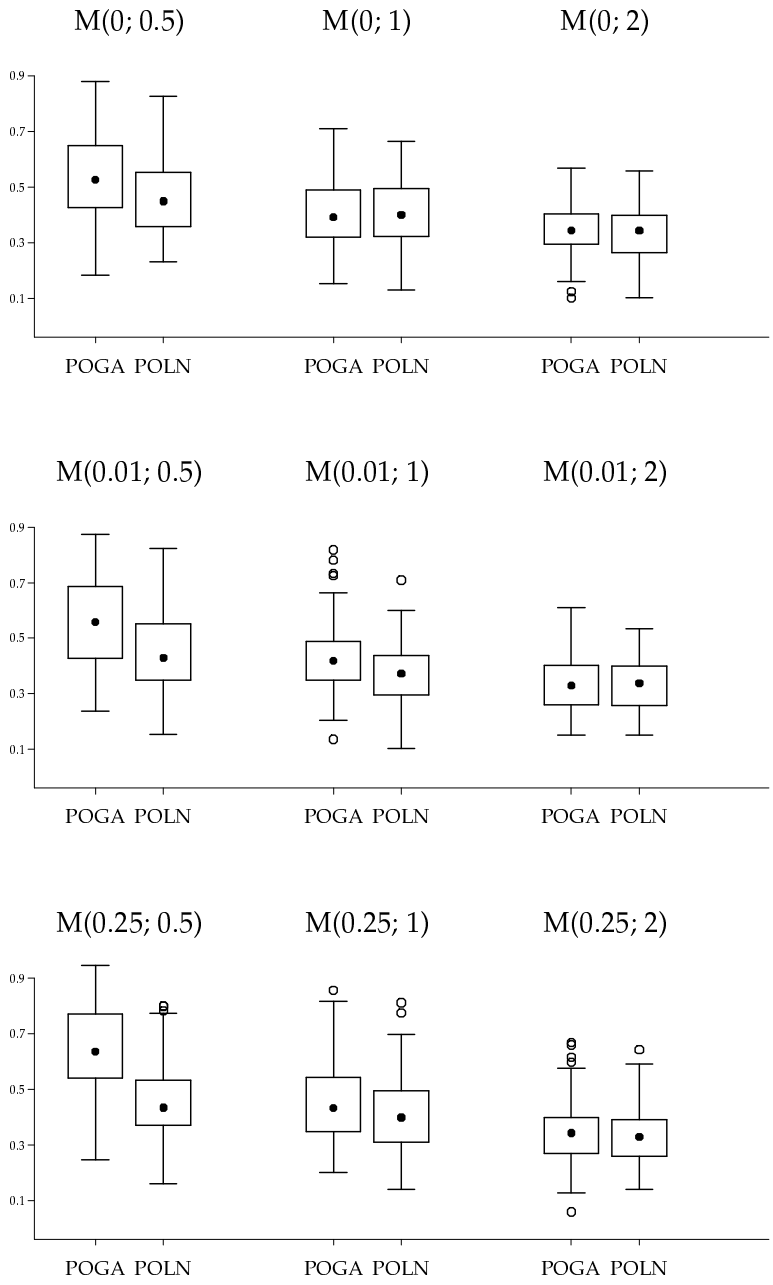

Figure 6.10: MSE Box Plots for posterior mean estimates of group indicator effects.

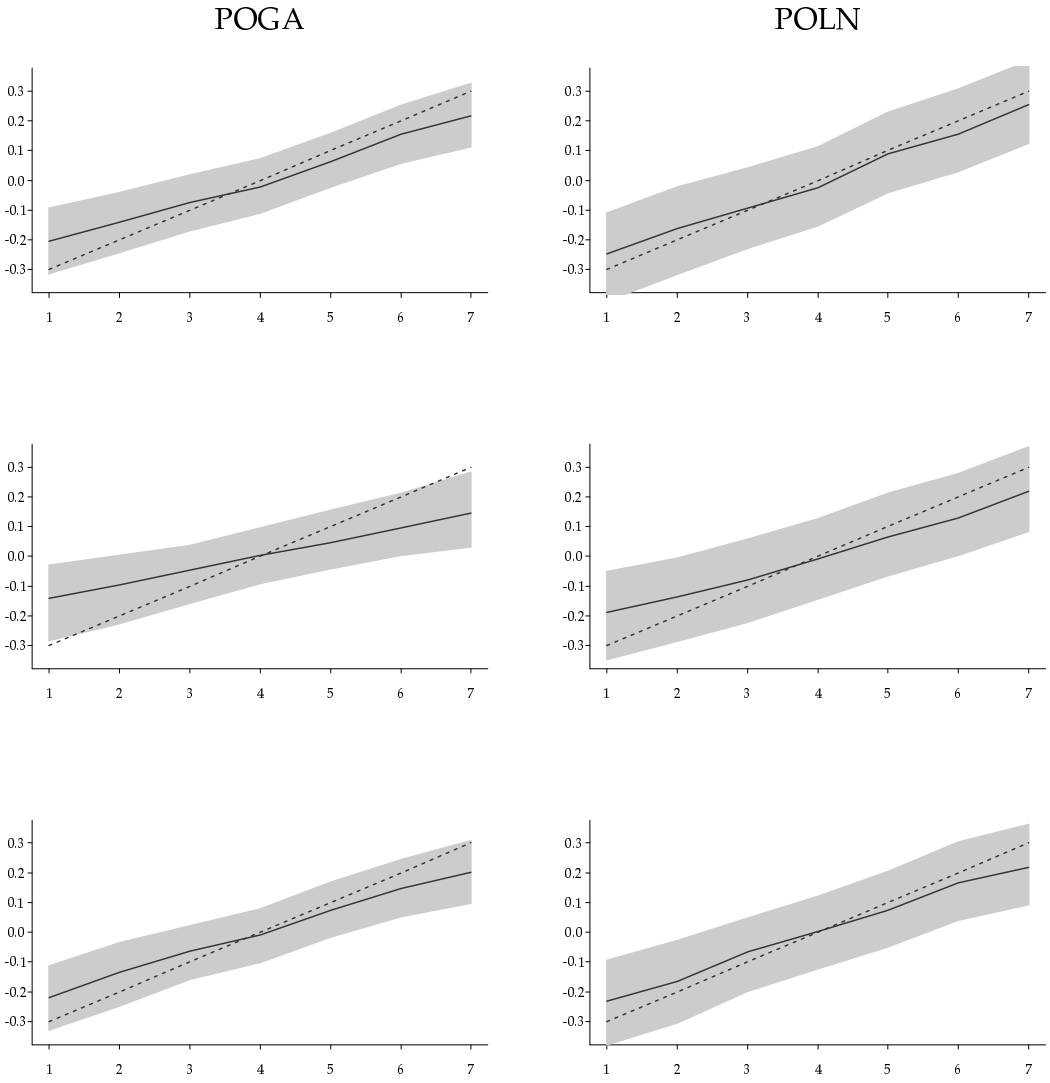

Figure 6.11: True effects and average posterior means of group indicator effects with pointwise 80% credible interval for selected models: $M(0.01; 1)$, $M(0.01; 0.5)$ and $M(0.25; 1)$ from the top to the bottom respectively.

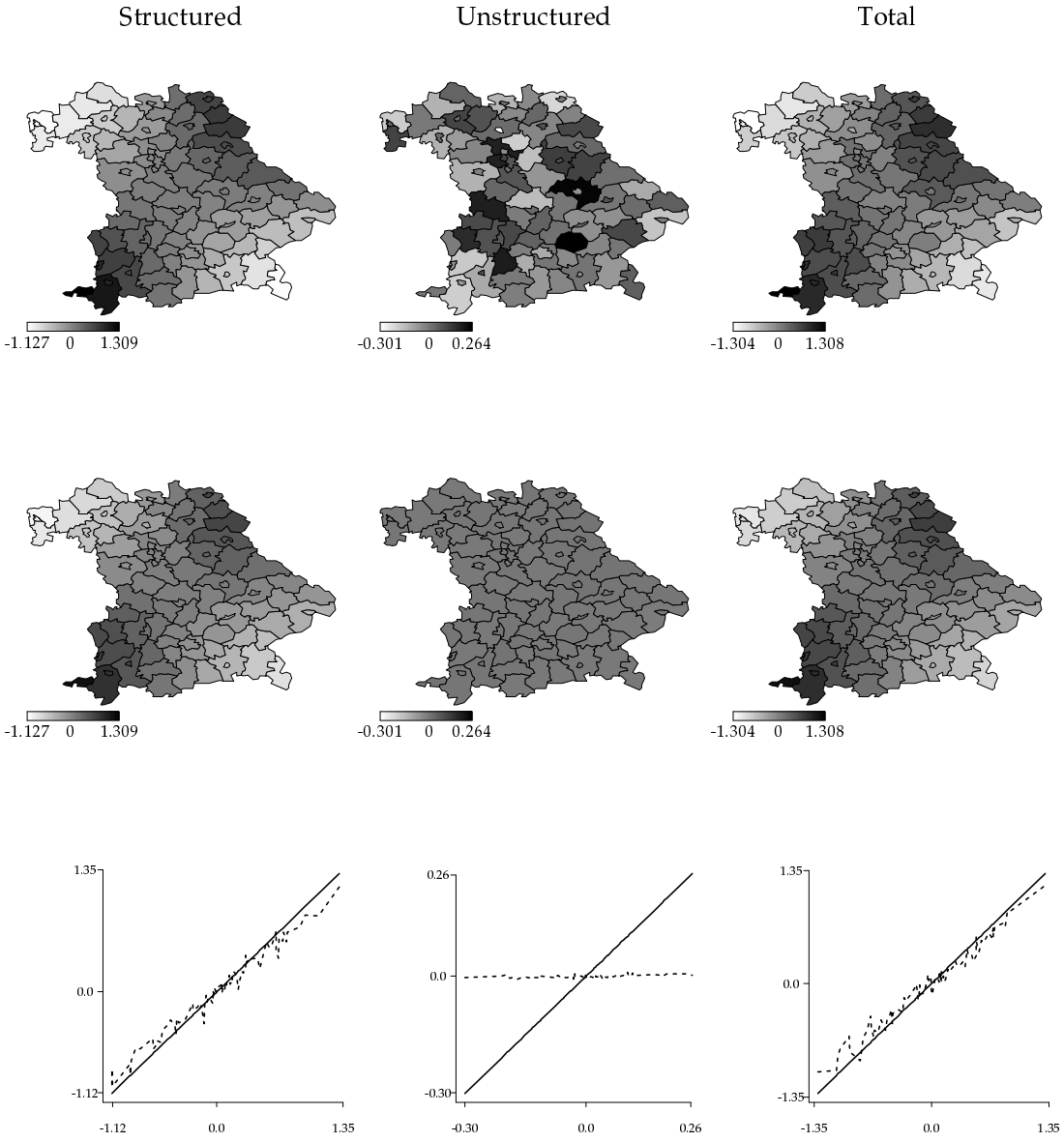

Figure 6.12: True (first row) and estimated (second row) structured, unstructured and total spatial effects together with diagonal plots (last row, true versus estimated effects) obtained for the POGA model $M(0.01; 1)$.

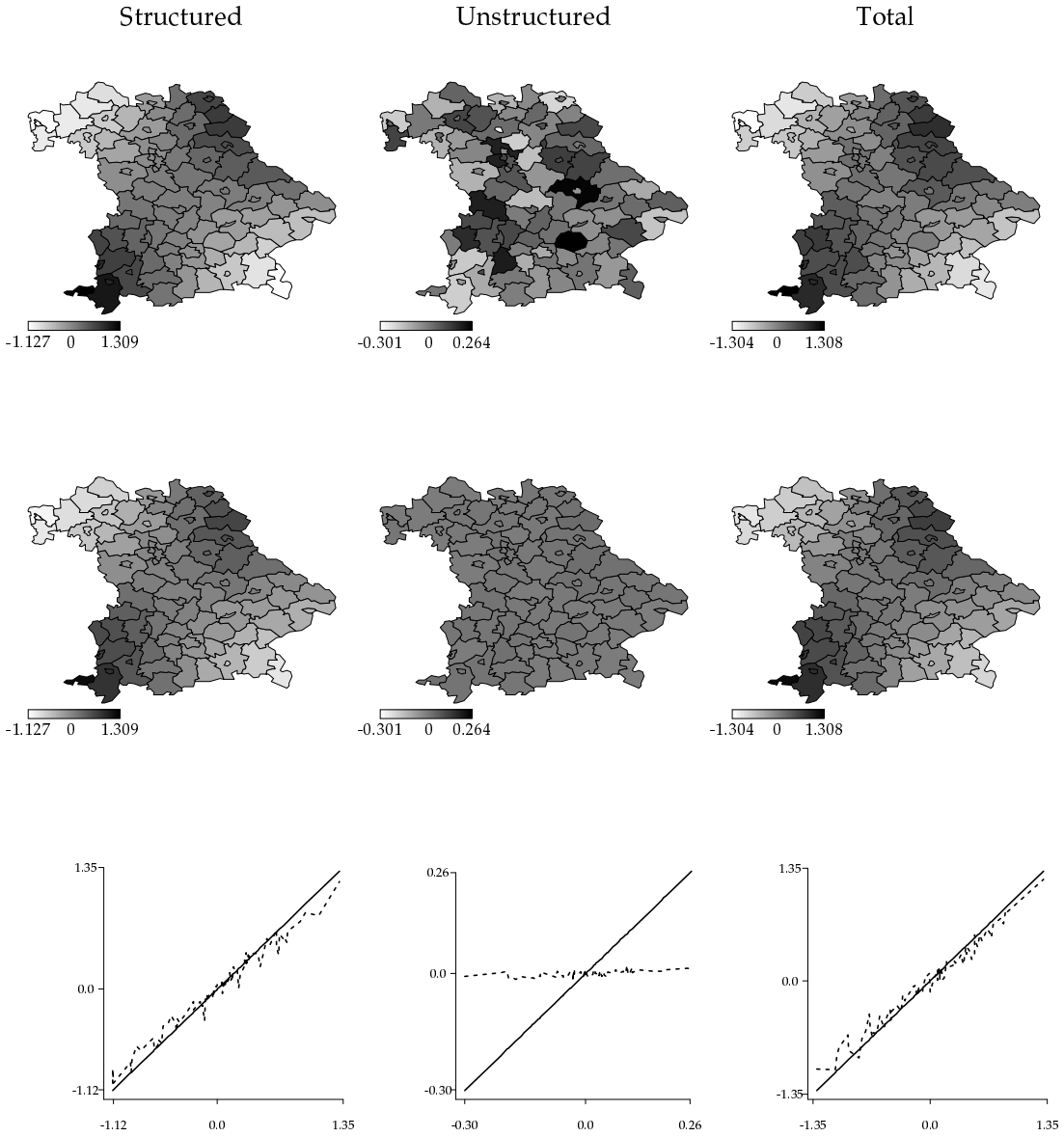

Figure 6.13: True (first row) and estimated (second row) structured, unstructured and total spatial effects together with diagonal plots (last row, true versus estimated effects) obtained for the POLN model $M(0.01; 1)$.

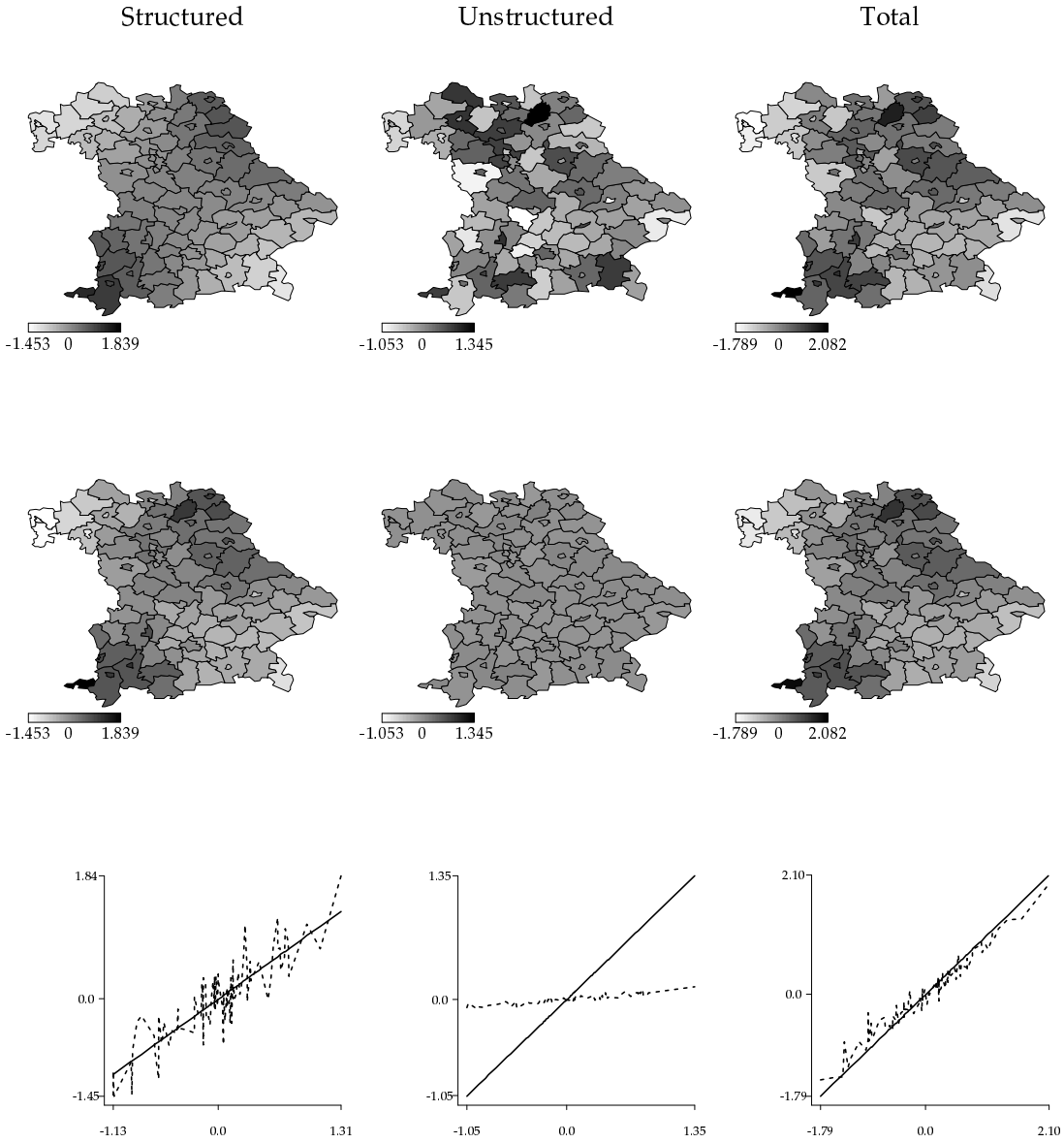

Figure 6.14: True (first row) and estimated (second row) structured, unstructured and total spatial effects together with diagonal plots (last row, true versus estimated effects) obtained for the POGA model $M(0.25; 1)$.

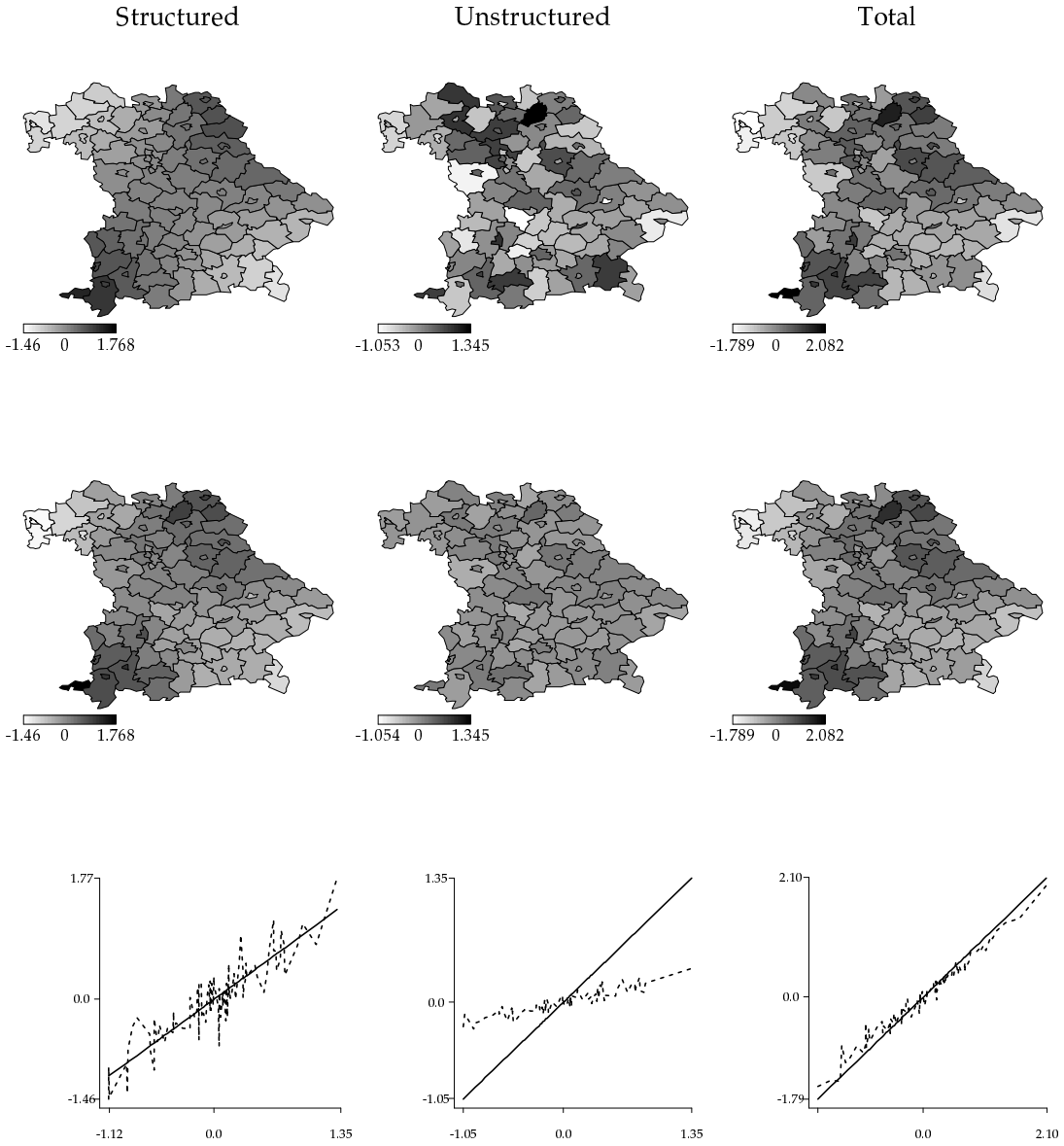

Figure 6.15: True (first row) and estimated (second row) structured, unstructured and total spatial effects together with diagonal plots (last row, true versus estimated effects) obtained for the POLN model $M(0.25; 1)$.

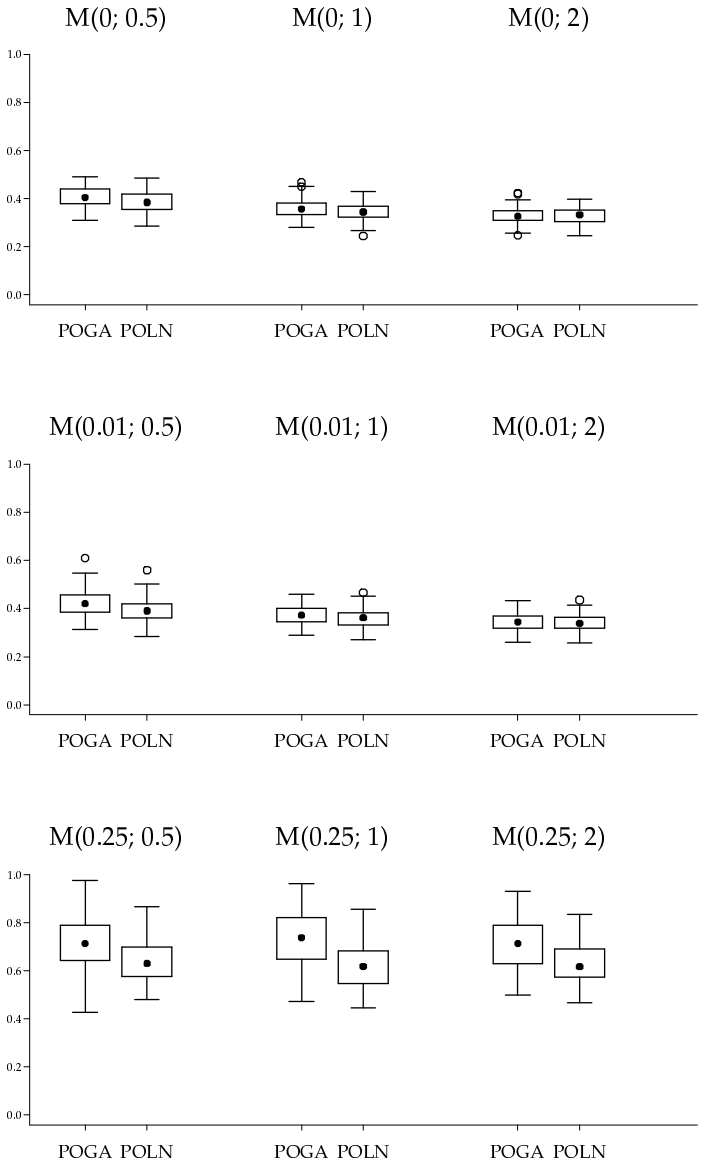

Figure 6.16: MSE Box Plots for posterior mean estimates of structured spatial effects.

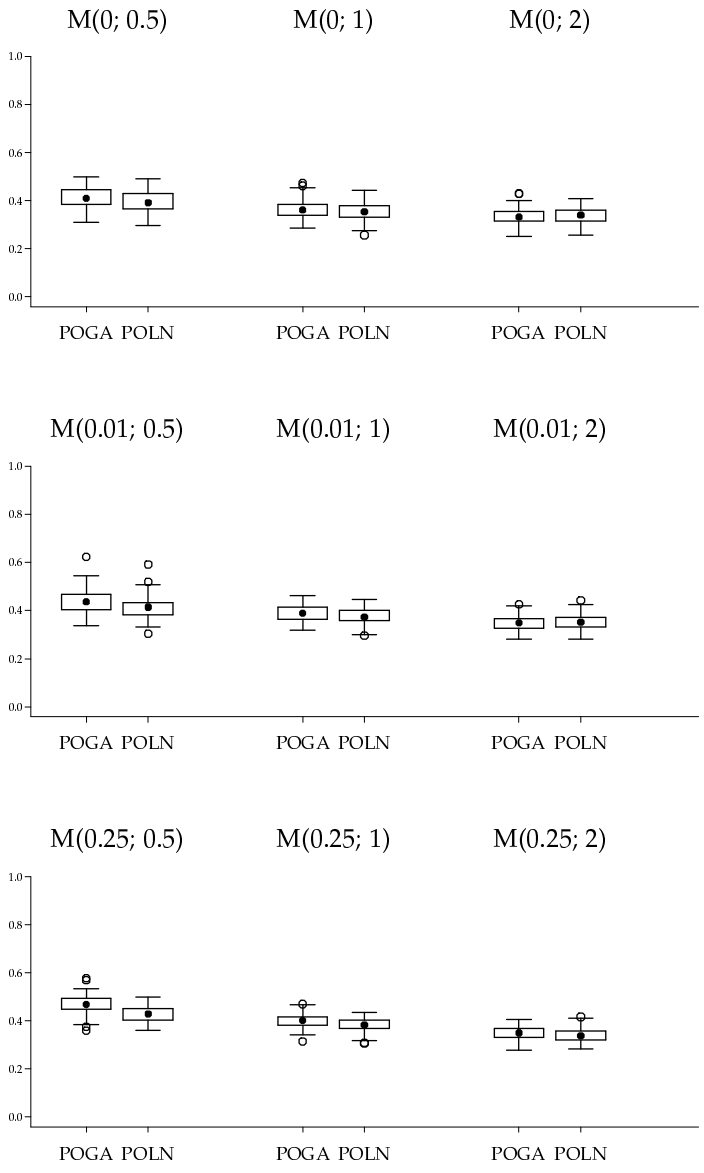

Figure 6.17: MSE Box Plots for posterior mean estimates of total spatial effects.

- After some preliminary analysis we found out that the performance of the POIG model decreases considerably by adding random or nonparametric terms in the predictor. The scale parameter is overestimated in the presence of random components and nonparametric terms.
- The hierarchical version of POIGH satisfactorily improves results only for simple linear predictor structures. For more complex predictor structures there is no improvement.
- The performance of POGA, NB and POLN models is quite satisfactorily on all presented data sets. Only following problem was found.
- The separation of the regional effects in a structured and an unstructured part as assumed by the predictor was not possible. The latter effects were not recognized and absorbed by the structured ones.

6.2 Zero inflation

In this second part of the simulation study we concentrate on models with zero inflation, with and without overdispersion. We keep the complicated covariate structures in the predictor. Zero inflation and overdispersion components should be recognized and properly separated. As a final test, the models should find out, applied to the adequate data, which source of variation the data contains. Is there only zero inflation, only overdispersion or both?

6.2.1 Data simulation

We generate data sets following the models ZIP, ZIPGA, ZIPIG and ZIPLN. We carry out the generation of the data in three steps, according to the definition of zero inflation given in (3.1) of Section 3.1. In the first step we generate values from the underlying count data distribution process $\boldsymbol{y}^{under}$. Then we generate the $\boldsymbol{w}$ $0/1$–vectors for a given θ. In the

third step we multiply the entries from $\boldsymbol{w}$ with those from $\boldsymbol{y}^{under}$ to obtain the true vector of observed responses $\boldsymbol{y}$.

The first step depends on the distributional assumption for the underlying count data process and is therefore different for every data set. We describe each of them separately. We take advantage of the simulated data from the last section. This saves computation time and simplifies comparison, if desired. With the estimation experience of the last section and because here the focus is on the separation of δ and θ, we only use the data sets with $\tau^2 = 0.01$.

For the ZIP data set, we exploit the created vector of linear predictors $\boldsymbol{\eta}$ for $\tau^2 = 0.01$ from the last section, which, has length 1920. With its help we generate 100 replications from a Poisson distribution with mean $\exp(\eta_i)$. Let us denote each replication with $\{y_i^{(r)},\ i = 1, ..., 1920\}^{under}$ for $r = 1, \ldots, 100$, where *under* still denotes the underlying process.

For the ZIPGA, ZIPIG and ZIPLN we can even use the generated replications from the corresponding POGA, POIG and POLN with $\tau^2 = 0.01$. Note that, in contrast to the ZIP data, we have three groups of replications for each model here, as we take three values for $\delta = 0.5, 1$, and 2 to draw them. We use the same notation for the response vectors as in the ZIP case: $\{y_i^{(r)},\ i = 1, ..., 1920\}^{under}$ for $r = 1, \ldots, 100$.

In addition, we have to generate binary vectors with 0/1 entries $\boldsymbol{w}^{(r)}$ for each vector of responses $\boldsymbol{y}^{under}$ of the last step. All the entries are $Bern(1-\theta)$ distributed. For θ we use three probability values θ =0.2, 0.5, and 0.8 in order to examine how many information we can loose through the selection process and nevertheless obtain good estimation results.

In the last step the observed response observations $\boldsymbol{y}^{(r)}$ are calculated as a product of $\boldsymbol{w}^{(r)}$ and $\boldsymbol{y}^{(r)\,under}$ for $r = 1, \ldots, 100$ and each underlying count data distribution. We combine the possible values of the zero inflation parameter θ with those for the scale parameter δ and get data for 6 different ZINB, ZIPGA, ZIPIG and ZIPLN models.

For the ZIP we have only two data sets. We obtain

$$\{y_i^{(r)},\, i = 1, ..., 1920\} \qquad r = 1, \ldots, 100.$$

All four models have the same predictor, given by

$$\eta_i \;=\; o_i + \alpha + \beta z_i + \sin(x_i) + \rho_{g_i} + f_{str}(s_i) + f_{unstr}(s_i), \tag{6.4}$$

for $i = 1, ..., 1920$. For more information about the individual terms we refer to Subsection 6.1.1.

For the discussion of simulation results, we denote the generated data sets by $M(\theta; \delta)$ or $M(\theta)$. For example, $M(0.5; 1)$ is a (ZINB, ZIPGA, ZIPIG or ZIPLN) model with 50% zero inflation $(\theta = 0.5)$ and individual random effects with medium $(\delta = 1)$ variability. $M(0.2)$ is a ZIP model with a low zero inflation $(\theta = 0.2)$. With this simulation design, we can assess the impact of the relative magnitude of zero inflation and individual random effects on estimation of the various components.

For each simulation run r, we calculate posterior means, standard deviations, quantiles and the DIC criterion. From $R = 100$ simulation runs, we obtain overall empirical bias, MSE, box plots etc. for the estimates of all unknown parameters and functions.

6.2.2 Results

In the following the results and conclusions for the simulation study on zero inflated data are presented. Already in the first runs we have seen that $\theta = 0.8$ does not work well at all. From an interpretational point of view, it would mean that we have lost about 80% of the information in the data. This was too much to keep the models work properly and we restrict the exposition to the models with $\theta = 0.2$ and $\theta = 0.5$. This is presented in several blocks. The first block concentrates on the ZIP. As it does not have an overdispersion parameter it is not easy to compare its estimation results with those of the other models. The second block briefly gives some comments about the results of the

ZIPLN model. Finally, the third block present the main findings of the ZINB, ZIPGA and ZIPIG models jointly.

ZIP

Here we summarize the results of the simulation for the ZIP model. The aim is to check, how the model fits, and how the goodness of fit varies with the predetermined values of θ. According to our findings, we can divide this block of results into two groups, depending on the sensitivity with respect to the zero inflation parameter. The first one sums up the results for the fixed effects and the zero inflation parameter, and the second one the rest of the terms in the predictor.

1. The results of the simulations for the fixed effects α, β and the zero inflation parameter are summarized in Table 6.2. We see that for both of them the ZIP works very well independently of the proportion of zero counts we have in the model.

ZIP				
	$M(0.5)$		$M(0.2)$	
α	-5.003005	(0.1001695)	-4.99105	(0.0920597)
β	0.5051805	(0.0617621)	0.4958	(0.0460211)
θ	0.498956	(0.0188491)	0.203506	(0.0174884)

Table 6.2: Posterior means and standard deviations (in brackets) for the fixed effects and the zero inflation in the ZIP model.

2. The estimation of the other effects in the predictor seems to be more sensitive to the value of θ than the estimation of the fixed effects or θ itself. First, we consider the nonlinear curve $f(x) = \sin(x)$. Although both average posterior means for the splines have a very good shape, the MSE box plot in Figure 6.18 reveals that the fit of model $M(0.2)$ is on average better than the fit of model $M(0.5)$.

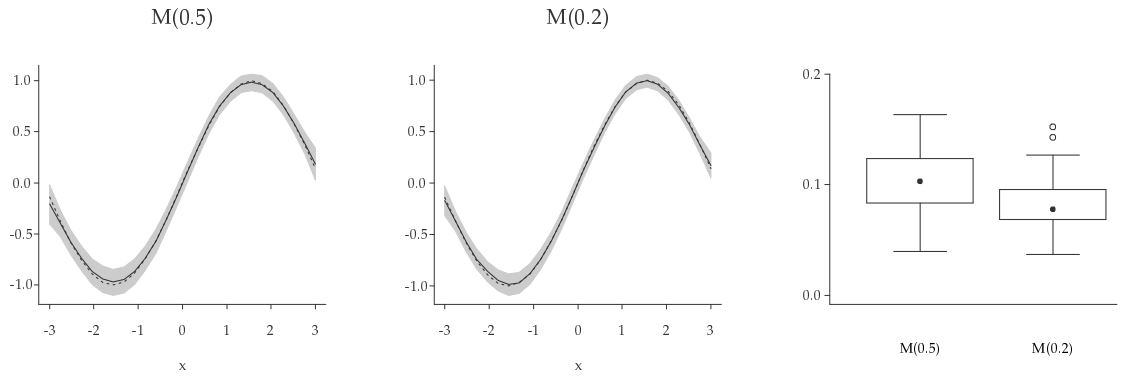

Figure 6.18: Average posterior mean estimates with pointwise 80% credible interval and MSE Box Plots for the nonlinear sine-effect of models $M(0.5)$ and $M(0.2)$.

Looking at Figure 6.19 we can draw the same conclusion as for the fit of nonlinear terms. There we show posterior mean estimates with pointwise 80% credible intervals for the group indicator effects. In the two left plots we see that the average fit is quite good for both models, but in the MSE box plot on the right, we see that the corresponding box for the $M(0.2)$ is placed in a lower position in the plot than for the $M(0.5)$. The finding is that the ZIP fits the group indicator effects better for $M(0.2)$ as for $M(0.5)$.

The same conclusion can be drawn from Figure 6.20, showing plots for the spatial effects. In the first column we find the diagonal plots for the average estimates of the structured spatial effect (first row), of the unstructured spatial effects (second row), and of the total spatial effects (third row) corresponding to the $M(0.5)$ model, in the second column we see the equivalent plots for the $M(0.2)$ model, and in the third column the box plots for the posterior mean estimates of the different spatial effects (structured: top; unstructured: middle; total: bottom) for both $M(0.5)$ and $M(0.2)$. The structured as well as the total spatial estimates are fitted very well, but we recognize the same identification problem for the unstructured spatial effects as in overdispersion models, so that in practical applications only the sum of both

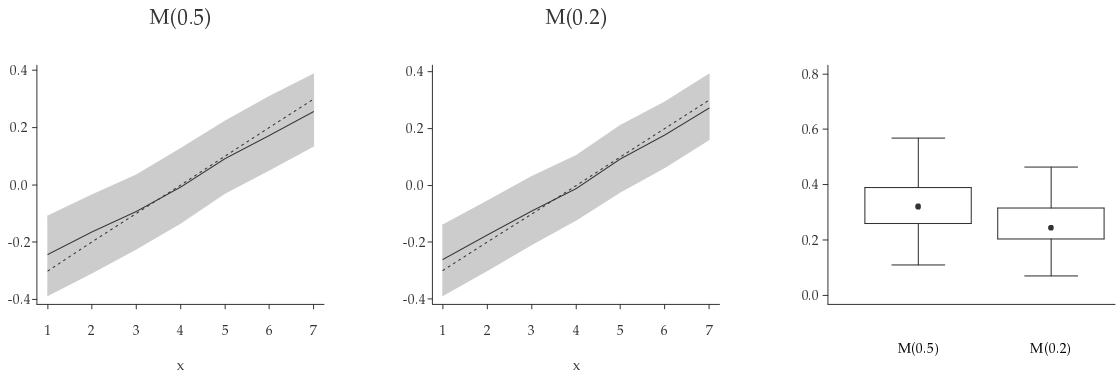

Figure 6.19: Average posterior mean estimates with pointwise 80% credible intervals and MSE Box Plots for the group indicator effect of models $M(0.5)$ and $M(0.2)$.

effects (total spatial effect) can be interpreted. From the box plots on the right side we deduce that the ZIP fits better for $M(0.2)$.

That the results for $M(0.2)$ are better than those for $M(0.5)$ is a logical result: The higher the proportion of zeros through increasing θ, the more information is lost about the generating process of the data depending on the covariates. So the fit for complex structures in the predictor becomes worse when θ increases.

ZIPLN

Recall, that the ZIPLN model is equivalent in its implementation to a ZIP model with gaussian distributed random effects for each observed unit in our data set. Although the results of the simulation study for the ZIP were very satisfactory, we could not carry out a similar study for the ZIPLN. There were numerical problems, that we could not solve. In Table 6.3 we present a part of the results obtained by running the ZIPLN model on the first replication of our six simulated data sets $M(\theta, \tau_\kappa^2)$, with $\theta = 0.2, 0.5$ and $\tau_\kappa^2 = 1.098, 0.6931, 0.4055$.

We see that none of the θ or δ parameters are estimated properly. For the zero inflation

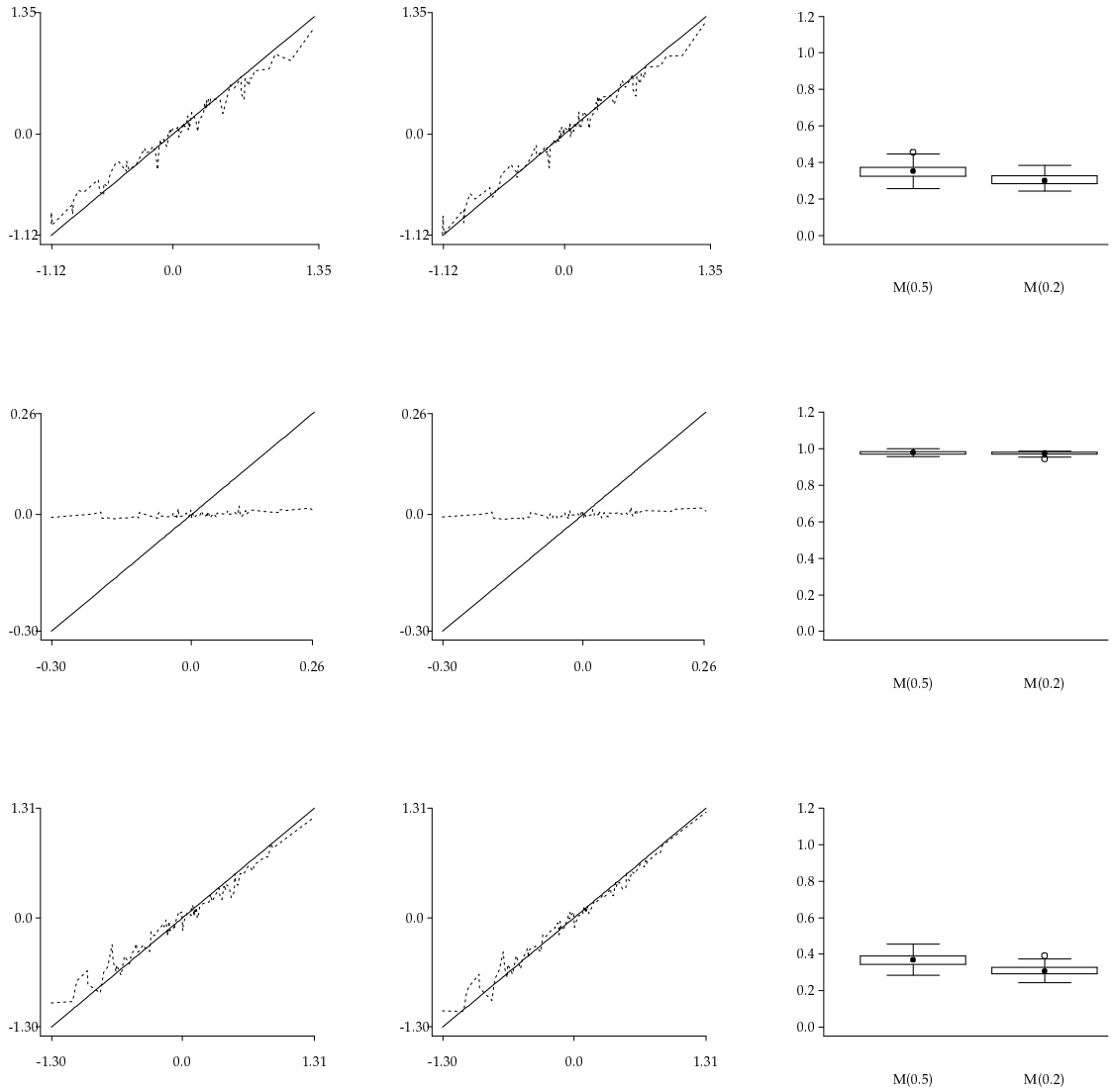

Figure 6.20: Diagonal plots of average posterior mean estimates and MSE Box Plots for models $M(0.5)$ and $M(0.2)$.

	$M(0.2;1.098)$		$M(0.2;0.6931)$		$M(0.2;0.4055)$	
τ_κ^2	0.501349	(0.0471057)	0.325239	(0.0354098)	0.166588	(0.0242277)
θ	0.998736	(0.00126418)	0.998605	(0.00140245)	0.99835	(0.00160501)
	$M(0.5;1.098)$		$M(0.5;0.6931)$		$M(0.5;0.4055)$	
τ_κ^2	0.482469	(0.0607495)	0.340732	(0.047905)	0.168726	(0.0303825)
θ	0.999026	(0.00102598)	0.998862	(0.00115092)	0.99888	(0.00107578)

Table 6.3: Posterior means and standard deviations (in brackets) for selected models.

parameter θ the model does not recognize its original value at all and delivers cases estimates arround the value 1 in all the cases. This causes numerical instability of the log–likelihood, that moves in a range of extremely small negative values, due to the term $\log(1-\theta)$. The estimation results for other parameters are very unreliable as well, see for example overdispersion parameter τ_κ^2 in Table 6.3. We know that both parameters are somehow related and see that for overestimated θ parameter τ_κ^2 is underestimated.

Other models

Finally, we present the results for ZINB, ZIPGA and ZIPIG models jointly.

1. We observed slight differences between the estimates of ZINB and the ZIPGA models, and hence we present both separately, when needed.

2. Table 6.4 shows the results for the ZINB, ZIPGA and ZIPIG models applied to $M(0.2;1)$, $M(0.5;1)$ and $M(0.5;0.5)$. The first two data sets are presented to compare how an increase of θ influences the estimates while keeping δ fixed. The third data set $M(0.5;0.5)$ is the most extreme case of information loss and overdispersion we have presented and is therefore interesting to proof the performance of the models on difficult situations. Analyzing Table 6.4, we see that in general fixed effects as well as overdispersion and zero inflation parameters are recovered very

well by the ZINB and ZIPGA models, may be with the exception of the last data set $M(0.5; 0.5)$. The ZINB model seems to recover overdispersion and zero inflation parameters a little bit better than the ZIPGA model.

A surprise comes up with the ZIPIG model, wich works excellent on all tested data sets, even in the worst case $M(0.5; 0.5)$. Remember that in Subsection 6.1.2 the results for the POIG model were not satisfactory at all. The introduction of the zero inflation parameter seems to improve its performance considerably. Table 6.4 also shows that the quality of the estimates decreases with increasing θ, which we know is equivalent to increase information loss.

3. Looking at Figure 6.21 we get the general impression that estimating the sine curve works very well for all the models. Figure 6.22 reveals that the quality of the estimation for the nonparametric terms depends also strongly on the value of θ used for the simulation, as it was the case with the fixed effects estimates. Increasing the zero inflation parameter from $\theta = 0.2$ to $\theta = 0.5$ worsens the estimates and places the box plots in a higher position on the plot. We also see a sensibility of the box plots with respect to the overdispersion parameter. Increasing the overdispersion in the model (setting a smaller δ) pushes the box plots upwards. Another interesting fact is that ZINB, ZIPGA and ZIPIG do not displays differences in the estimation of the sine curve.

4. Figures 6.23 and 6.24 summarize the results for the effects ρ_g of the group indicator g. Both figures clearly show that the quality of the results varies strongly depending on the values of the overdispersion and zero inflation parameters. The last row of Figure 6.23 corresponds to the model $M(0.2; 2)$, which has the lowest zero inflation and the lowest overdispersion among all the models. We see that the alignment of the black and dotted line are much better than in the other rows. Figure 6.24 confirms the first consequences drawn from Figure 6.23 and their extension to all the models. The variation of overdispersion and zero inflation has a great impact on

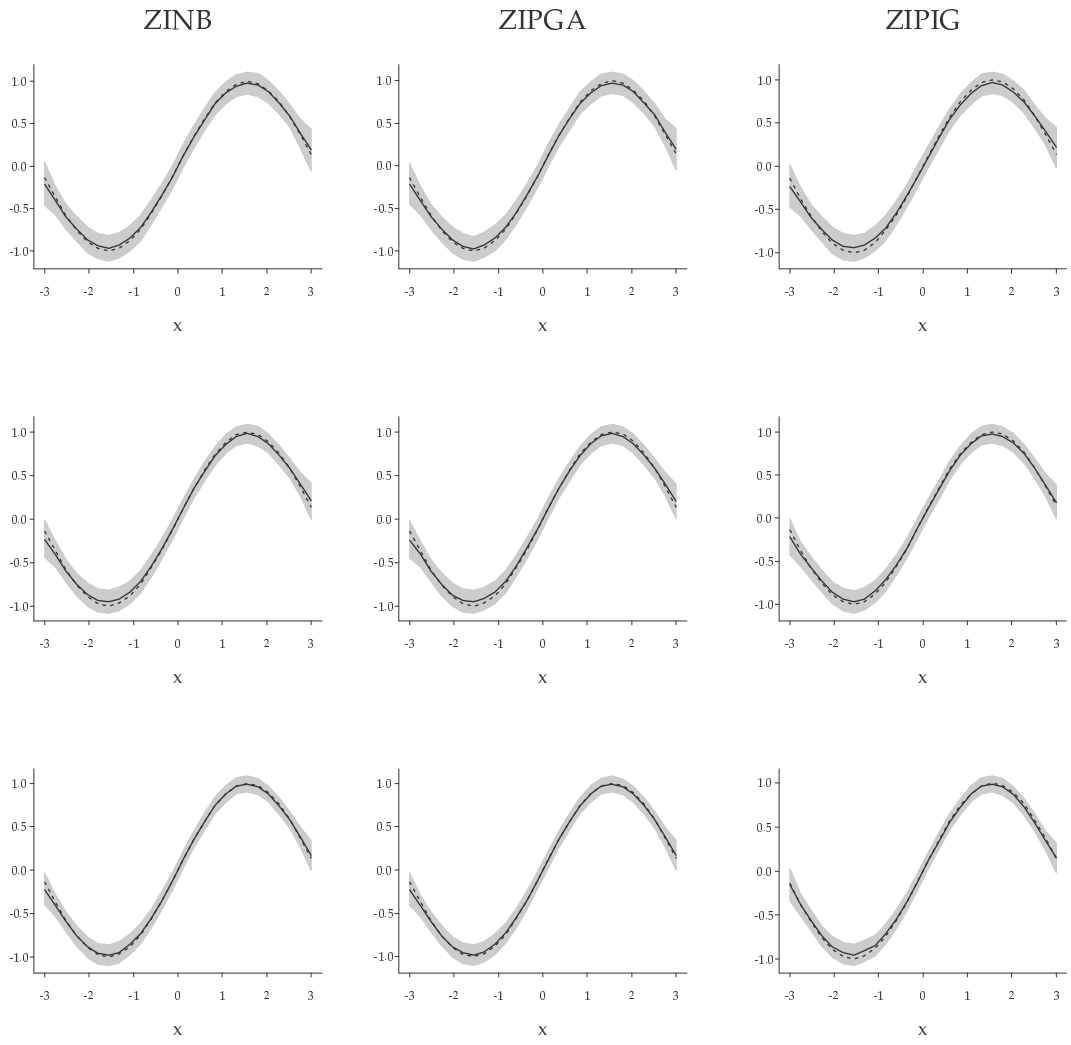

Figure 6.21: Average posterior mean estimates with pointwise 80% credible interval for the nonlinear sine–term of models $M(0.2; 0.5)$ (top), $M(0.2; 1)$ (center) and $M(0.2; 2)$ (bottom).

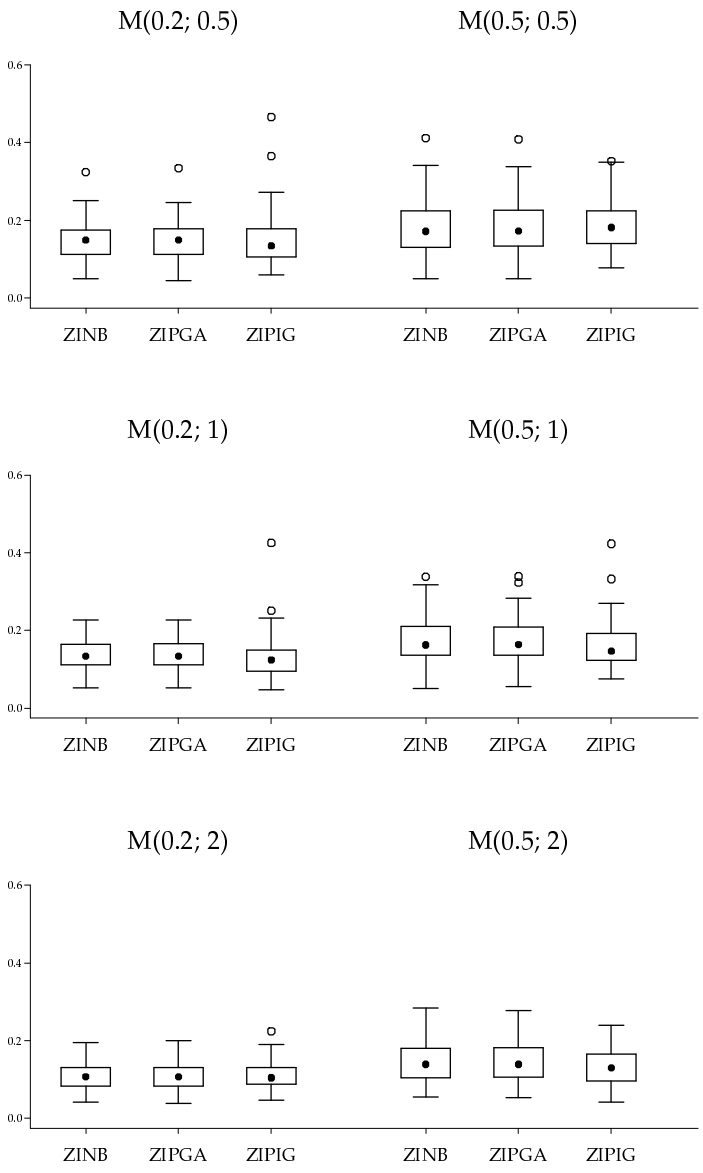

Figure 6.22: MSE Box Plots for posterior mean estimates of the nonlinear sine–term.

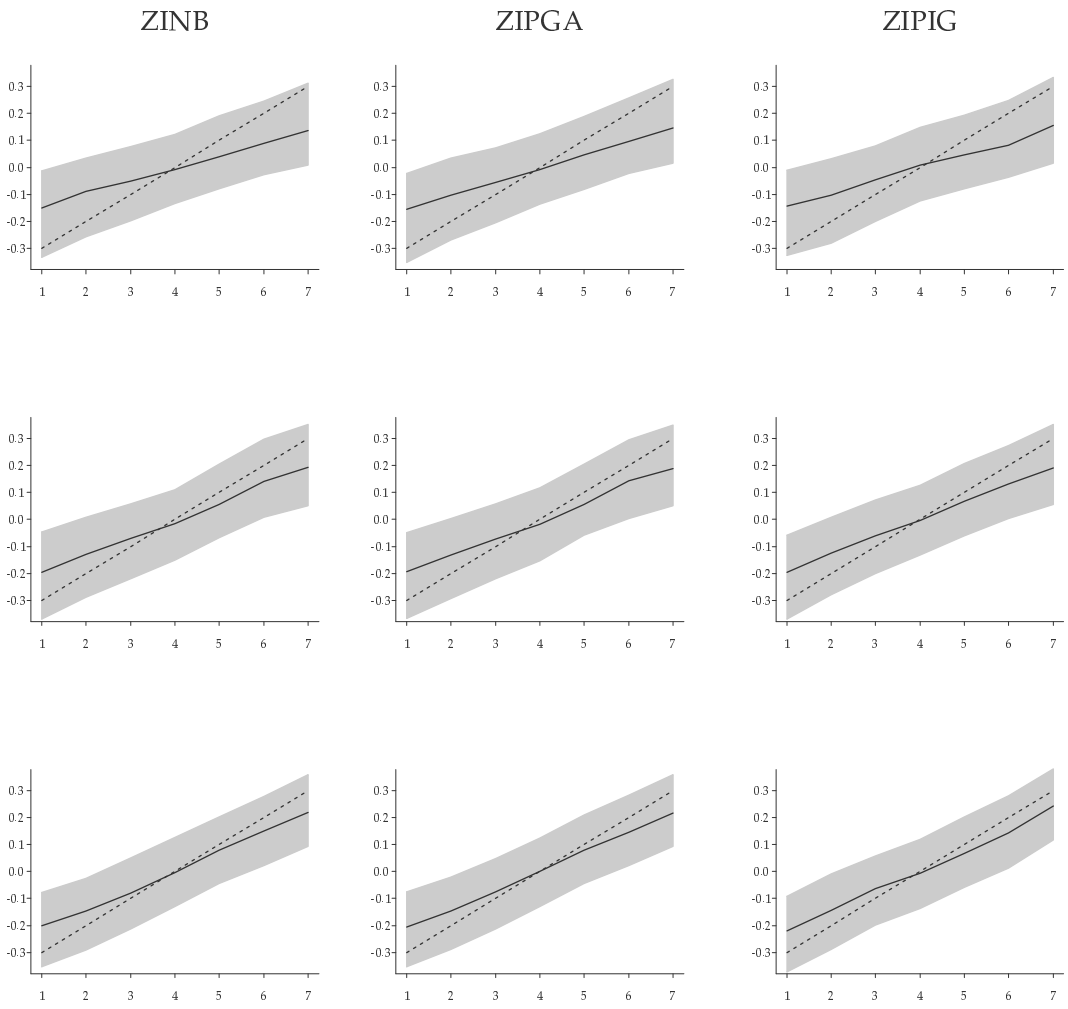

Figure 6.23: Average posterior mean estimates with pointwise 80% credible interval for the group indator effects of models $M(0.2; 0.5)$ (top), $M(0.2; 1)$ (center) and $M(0.2; 2)$ (bottom).

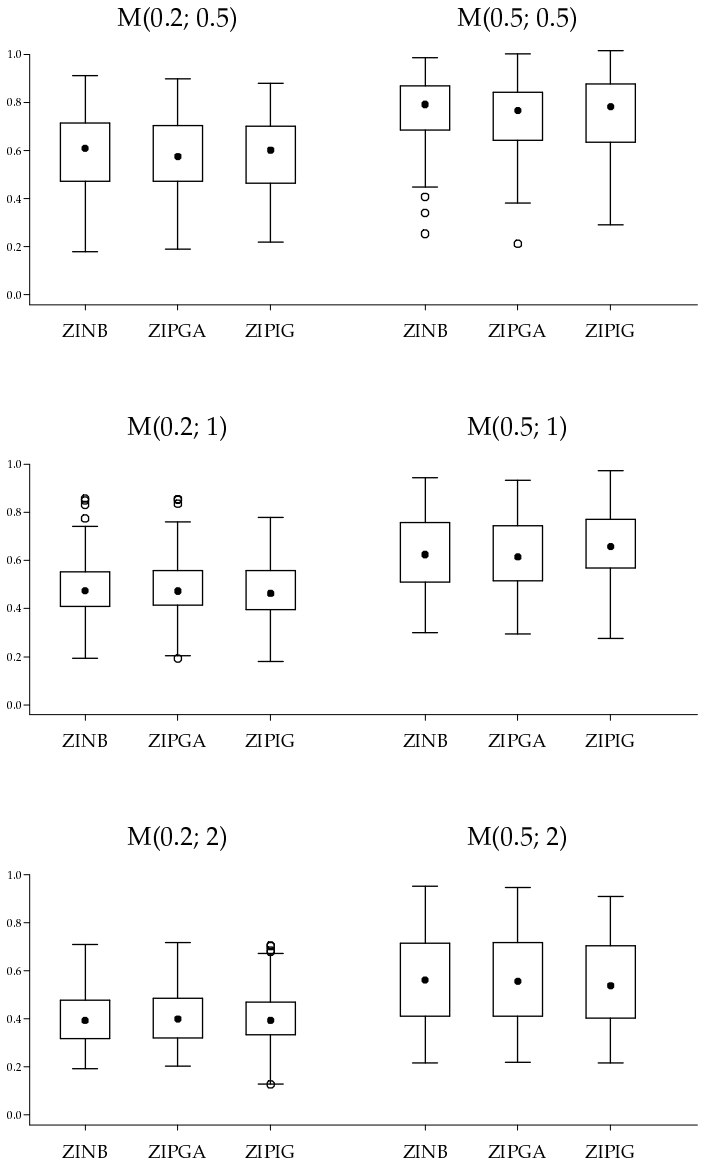

Figure 6.24: MSE Box Plots for posterior mean estimates of group indicator effects.

	$M(0.2;1)$		$M(0.5;1)$		$M(0.5;0.5)$	
ZINB						
α	-5.0363	(0.1145)	-5.0549	(0.1453)	-5.0973	(0.2153)
β	0.5032	(0.0827)	0.5107	(0.1103)	0.4915	(0.1347)
δ	0.9966	(0.1742)	0.9659	(0.2336)	0.4321	(0.1527)
θ	0.1895	(0.0484)	0.4755	(0.0496)	0.4439	(0.1127)
ZIPGA						
α	-5.0235	(0.1115)	-5.0150	(0.1269)	-4.8305	(0.1334)
β	0.5072	(0.0821)	0.5061	(0.1085)	0.4821	(0.1276)
δ	1.0190	(0.1629)	1.0759	(0.2108)	0.7720	(0.1084)
θ	0.1961	(0.0414)	0.4959	(0.0350)	0.5758	(0.0300)
ZIPIG						
α	-5.0060	(0.1011)	-5.0231	(0.1178)	-5.0089	(0.1357)
β	0.4989	(0.0815)	0.5173	(0.1082)	0.4923	(0.1308)
δ	1.0083	(0.1454)	1.0015	(0.1863)	0.5026	(0.1040)
θ	0.1935	(0.0284)	0.4911	(0.0281)	0.5003	(0.0356)

Table 6.4: Posterior means and standard deviations (in brackets) for selected models.

the fit of the random effects. In contrast to the estimation of nonparametric terms, there does not seem to be a difference in the quality of fit between the ZIPIG model and the ZINB and ZIPGA models.

5. Finally, we present the estimation results for the spatial term. Remember that we have split the total spatial effect in two further effects: a structured and an unstructured. Figures 6.25 and 6.26 show that the separation of structured and unstructured effects is also very unreliable in zero inflated models. The unstructured spatial component is integrated in the structured one, so that at least the sum of both

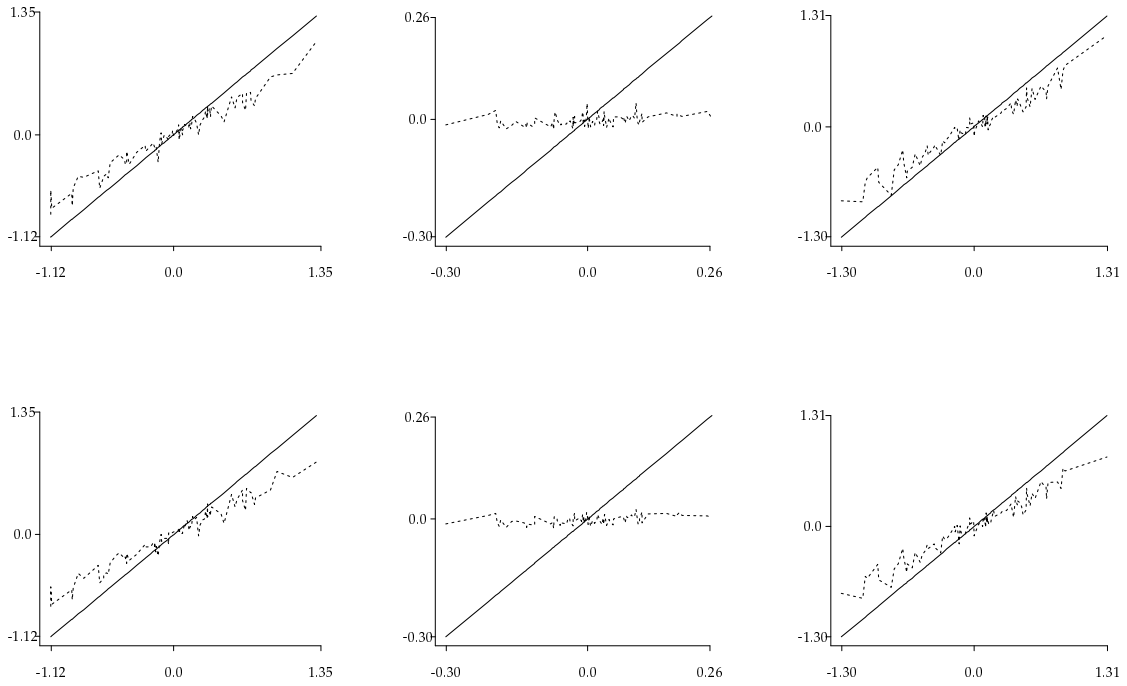

Figure 6.25: Diagonal plots of average posterior mean estimates of structured (first column), unstructured (second column), and total (third column) spatial effects for ZIPGA (first row) and ZIPIG (second row) models on $M(0.5; 0.5)$.

remains a good estimator for the total spatial effect of the regions. Comparing the two figures, sensitivity of the models with respect of overdispersion and zero inflation becomes clear. The first figure presents results from ZIPGA and ZIPIG models on data with high overdispersion and high zero inflation ($M(0.5; 0.5)$). The second, on data with low overdisperion and low zero inflation ($M(0.2, 2)$). The dotted lines match the diagonal black line in the second figure much better.

Figures 6.27 and 6.28 show box plots for structured and total spatial effects, respectively. Both figures reflect the impact of amount of overdisperion and zero inflation on the box plots. With increasing θ (from left to right) the box plots are pushed upwards. The interpretation: increasing zero inflation worsens the fit. With in-

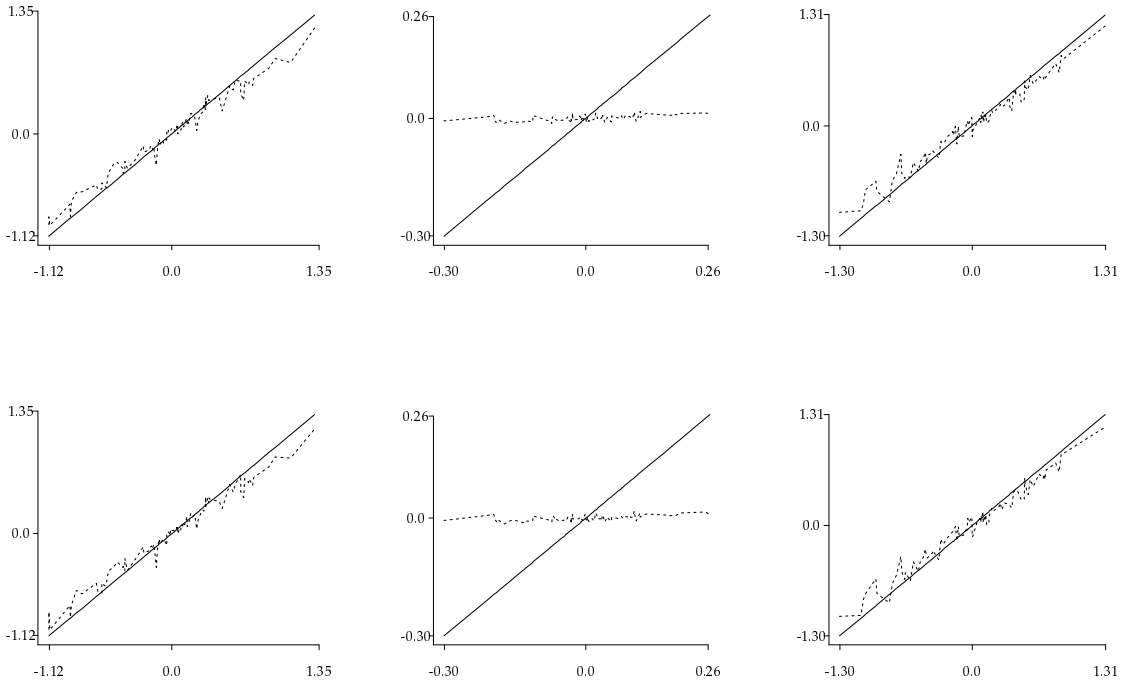

Figure 6.26: Diagonal plots of average posterior mean estimates of structured (first column), unstructured (second column), and total (third column) spatial effects for ZIPGA (first row) and ZIPIG (second row) models on $M(0.2; 2)$.

creasing δ (from the top to the bottom) the box plots are pushed downwards. The interpretation: decreasing overdispersion betters the fit.

From the box plots we can also see that the estimation results for spatial effects in ZINB, ZIPGA and ZIPIG models are quite similar.

6. As explained in the Chapters 2 and 3, overdispersion implies excess of zeros in the observed data and zero inflation implies overdispersion in the data. The question is how reliable the models are in discovering the right source for zero inflation and/or overdispersion in the data. The ZIPGA model is able to estimate both overdispersion and zero inflation parameters and it has provided good results in this simulation study. Therefore we check its reliability by applying a ZIPGA model to four

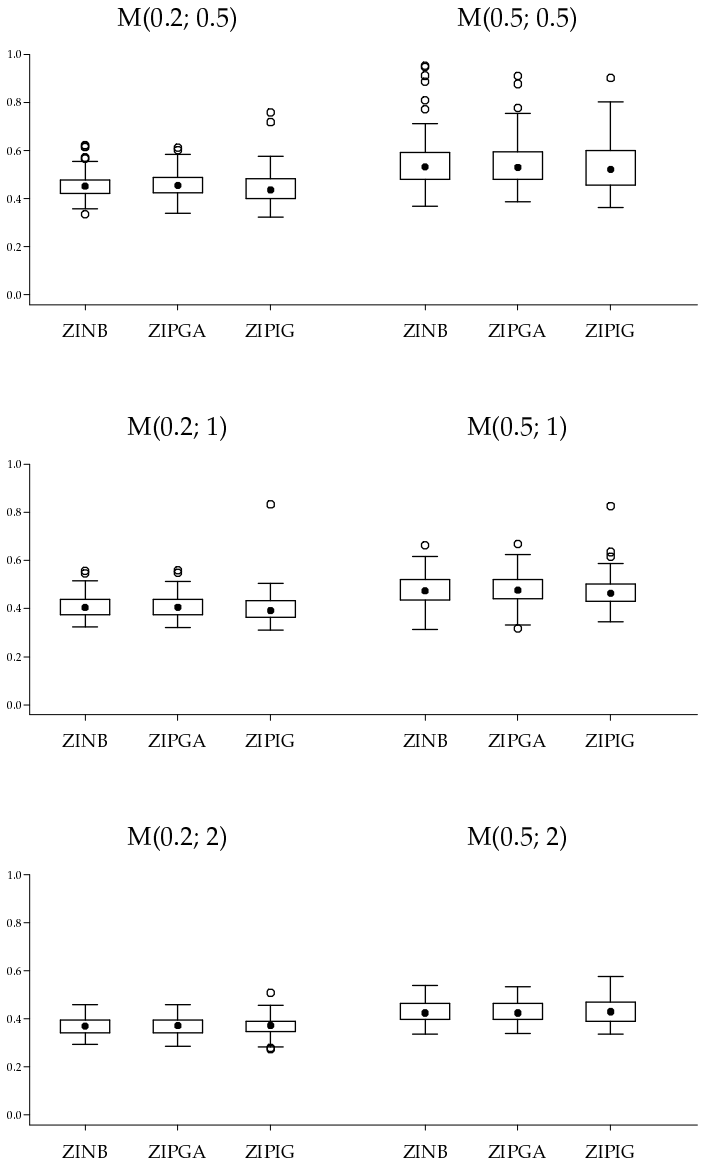

Figure 6.27: MSE Box Plots for posterior mean estimates of structured spatial effects.

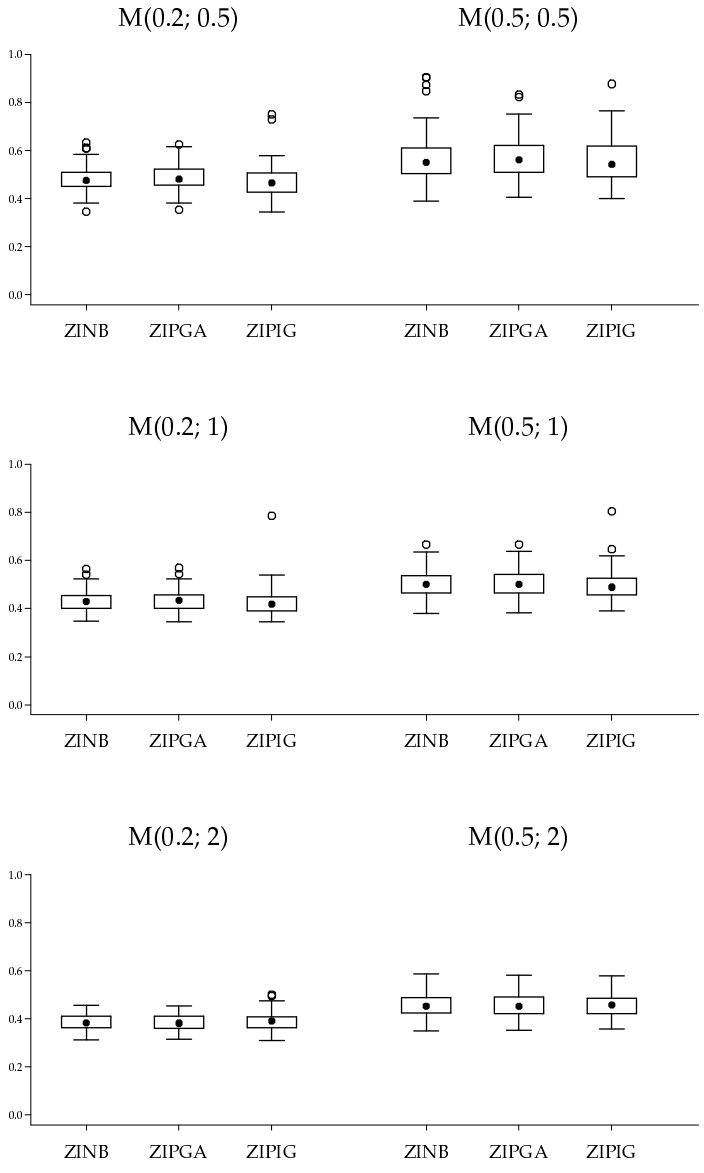

Figure 6.28: MSE Box Plots for posterior mean estimates of total spatial effects.

data sets with different generating processes. Each one is the first replication of the data sets described during this chapter. The first data set (PO) is Poisson distributed and is the first step in a ZIP generating process, before we multiply the data by the $0/1$ vector. This data has no overdispersion and no zero inflation. The second data set (ZIP) is zero inflated Poisson distributed and extracted from $M(0.5)$; thus it has zero inflation with $\theta = 0.5$ but no overdispersion. The third data set (POGA) is the first replication from $M(0.01, 1)$, generated in Subsection 6.1.1. It has overdispersion with $\delta = 1$ but no zero inflation. And finally, the fourth data set (ZIPGA) shows both zero inflation with $\theta = 0.5$ and overdisperion with $\delta = 1$ and is the first replication from $M(0.5, 1)$ generated in Subsection 6.2.1.

In Table 6.5 we list the results from applying the ZIPGA model to the four data sets. Remember that in our notation a large value for δ is a signal of no overdispersion in the data, and a small value for θ indicates no zero inflation. We see that the ZIPGA model is able to recognize what is actually hidden in the data. For the PO data, estimation results show no overdispersion and no zero inflation, in accordance with the data. On ZIP data, the estimated posterior mean for δ is 130.326, which is a sign for no overdispersion. On the other hand, the estimate for θ is 0.508053, which is very close to the real value 0.5. Applied to POGA data, the ZIPGA model finds no trace of zero inflation (with an estimate for θ near zero) and estimates the scale parameter δ by approximately 1, its real value. For ZIPGA data, both overdispersion and zero inflation are well recognized.

6.2.3 Résumé

In the following a summary of the results presented in the last subsection is given.

- The performance of ZIP models was very satisfactory.
- The results achived by ZINB, ZIPGA and ZIPIG are also in general quite correct.

ZIPGA model					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
on PO data					
δ	130.326	100.259	31.0548	87.557	366.721
θ	0.00652863	0.00570259	0.000225573	0.00505018	0.0213395
on ZIP data					
δ	111.305	133.924	20.8465	43.564	417.572
θ	0.508053	0.0198114	0.468088	0.50795	0.548033
on POGA data					
δ	1.0342	0.109076	0.854555	1.02085	1.26776
θ	0.0470683	0.0276712	0.00336344	0.0447656	0.105396
on ZIPGA data					
δ	1.14479	0.206438	0.78946	1.12776	1.56934
θ	0.54088	0.0311651	0.473972	0.544119	0.5945

Table 6.5: Results for the zero inflation and overdispersion parameters in the ZIPGA model

- All the models show sensitivity problems with the amount of zero inflation and overdispersion.
- As expected, the separation of spatial effects in structured and unstructured effects was not possible.
- The ZIPLN model did not achieve the desired results. The estimation of θ was not appropriate at all and hence other parameters could not be estimated as desired.

Chapter 7

Case studies

In this chapter we apply the developed models to two real data sets. In Section 7.1 we will work with a patent data set, also used in Jerak and Wagner (2003). We apply Bayesian generalized additive mixed models for count data using some of the distributions presented in Chapters 2 and 3. The data contains metrical and binary covariates and we can build a semiparametric predictor structure as explained in Chapter 4. The second data set is described in Section 7.2. It is a massive car insurance data set, that has been analyzed previously in Fahrmeir, Lang and Spies (2003) using a Bayesian generalized geoadditive Poisson regression. It contains a lot of covariates, among others also geographical information. In this work we apply a Bayesian generalized geoadditive mixed count data regression to capture possible overdispersion or zero inflation in the data.

In both sections we first present and describe the data, as well as the models we are going to apply. Second, we describe part of the results and draw some conclusions.

7.1 Patent Data

Analysis of patent data has a long tradition in economic research. The number of patents can be seen as a sort of measure for the innovative activity or inventiveness and hence

somehow for scientific development. To apply for a patent, the inventor must cite all related already existing patents where his new patent is based on. All the citations included in new patents that refer to an already existing patent are called *forward citations*, and can be understood as a good indicator for the patent's social and monetary value, or, in other words, as an indicator for its quality.

First we will apply a classical Poisson (PO) regression model, where the data given the covariates are supposed to be Poisson distributed. Second we use further regression models that allow for overdispersion in the data, namely NB, POGA, POIG and POLN regression models, and compare the results. In the NB model, the data given the covariates are supposed to follow a negative binomial distribution. In the POGA, POIG and POLN models, the data are supposed to be Poisson distributed given the covariates and random effects. The difference to the classical Poisson regression (PO) is that in addition to the given covariates we also estimate a vector of individual specific random effects, that is supposed to have i.i.d. components with gamma (POGA), inverse Gaussian (POIG) or LogNormal (POLN) prior respectively. Actually, the NB and POGA formulations are equivalent from a theoretical point of view, but depending on the situation it may be more interesting to use the NB (algorithms converge faster) or the POGA model (provides more information). All three models are described in Chapter 2. Finally, we try the ZIPGA model (see Chapter 3) on the patent data. The aim is to detect whether there is zero inflation together with overdispersion in the data or not on the basis of the estimates for θ and δ from the ZIPGA model.

For a more detailed description of the patent data and its institutional background we refer to Jerak and Wagner (2003). In their work, they apply a Bayesian semiparametric binary regression model for the event 'opposition or not'. Here we consider this variable as a binary effect in our predictor. Guo and Trivedi (2002) apply (among others) a NB and a POIG regression models to two cross–sectional long-tailed patent data sets to account for overdispersion. They model the number of patents applications of companies depending on the research and development (R&D) spendings among other covariates,

but only with linear predictors. In the next subsection we describe the patent data used here (see also Jerak and Wagner (2003)) and the models that we have applied. Afterwards we present the results.

7.1.1 Data and model description

We will analyze the dependence between the number of forward citations ($forwcits$) for a patent and the variables given in Table 7.1, based on 4805 observations. Before we proceed with the analysis, we first take a look at the raw data.

Metrical covariates	
gryear	Grant year
nstat	Number of designated states
claims	Number of EPO claims
Binary covariates (1 = Yes / 0 = No)	
biopharm	Patent from biotech/pharma sector
ustwin	US twin exists
cntry_us	Holder of the patent from US
cntry_ch_de_gb	Patentholder from Switzerland, Germany or Great Britain
accexam	Accelerated exam requested
accsrch	Accelerated search requested
pct	Patent Cooperation Treaty (PCT) application filed
opp	Opposition(s)

Table 7.1: Variables in the patent data set.

The response variable $forwcits$ has mean 1.6289 and variance 7.3541, i.e. the variance exceeds the mean by far. Its minimum is 0 and its maximum is 40. About 46% of the observations are zero and 95% are smaller or equal 6, which is a sign for long tails. These

facts are an indication for possible overdispersion. As the number of zero counts is large it may make sense to explore zero inflation in the data as well.

The data set contains three metrical variables. First, we have the grant year for the patent (*gryear*), with the values 1980 to 1997. Second, the number of designated states in Europe (*nstat*), which is a sort of territorial measure, ranging from 1 to 17. Third, the number of patent claims (*claims*), which define and set boundaries to the invention, and may be considered as a measure for the patent value. The variable *claims* assumes the values 1 to 50. In Figure 7.1 we show some plots of the distribution of $forwcits$ for the different observed values of the metrical covariates. The plots show pointwise mean (black line), and 5% and 95% quantiles (grey area) of $forwcits$ for each different observed value of the metrical covariates. In the first plot, the black line indicating the pointwise mean of $forwcits$ seems to decrease for increasing values of *gryear*. In the second, the black line has an increasing trend with respect to *nstat*. Finally, in the third plot we observe a more or less increasing trend till $claims = 40$.

In Table 7.1, we also have 8 dummy covariates. In Figure 7.2 the distribution of $forwcits$ within the two values (0 or 1) of each covariate is given. The most important fact we observe in this plot is that in the category 1 of the covariate *pct* there are extremely few values different of zero for $forwcits$ compared with the number of zero observations in this category. This can lead to problems in the estimation because of the almost missing variability of the data within this level. Nevertheless we include the variable in the model and will carefully observe the results. Changing from the value 0 to the value 1 in the covariates *accsrch*, *biopharm*, and opp seems to have a positive impact on the response variable $forwcits$. On the other hand, changing from 0 to 1 in *accexam* seems to have a negative impact. By the covariates *cntry_us*, *cntry_ch_de_gb*, and *ustwin* we do not find any visible behavior pattern.

Before we present the results, we give some commenst about the models we have applied. The number of forward citations was analyzed with structured additive NB, POGA, POIG, and POLN regression. We additionally apply ZIP, ZIPGA and ZIPIG to check

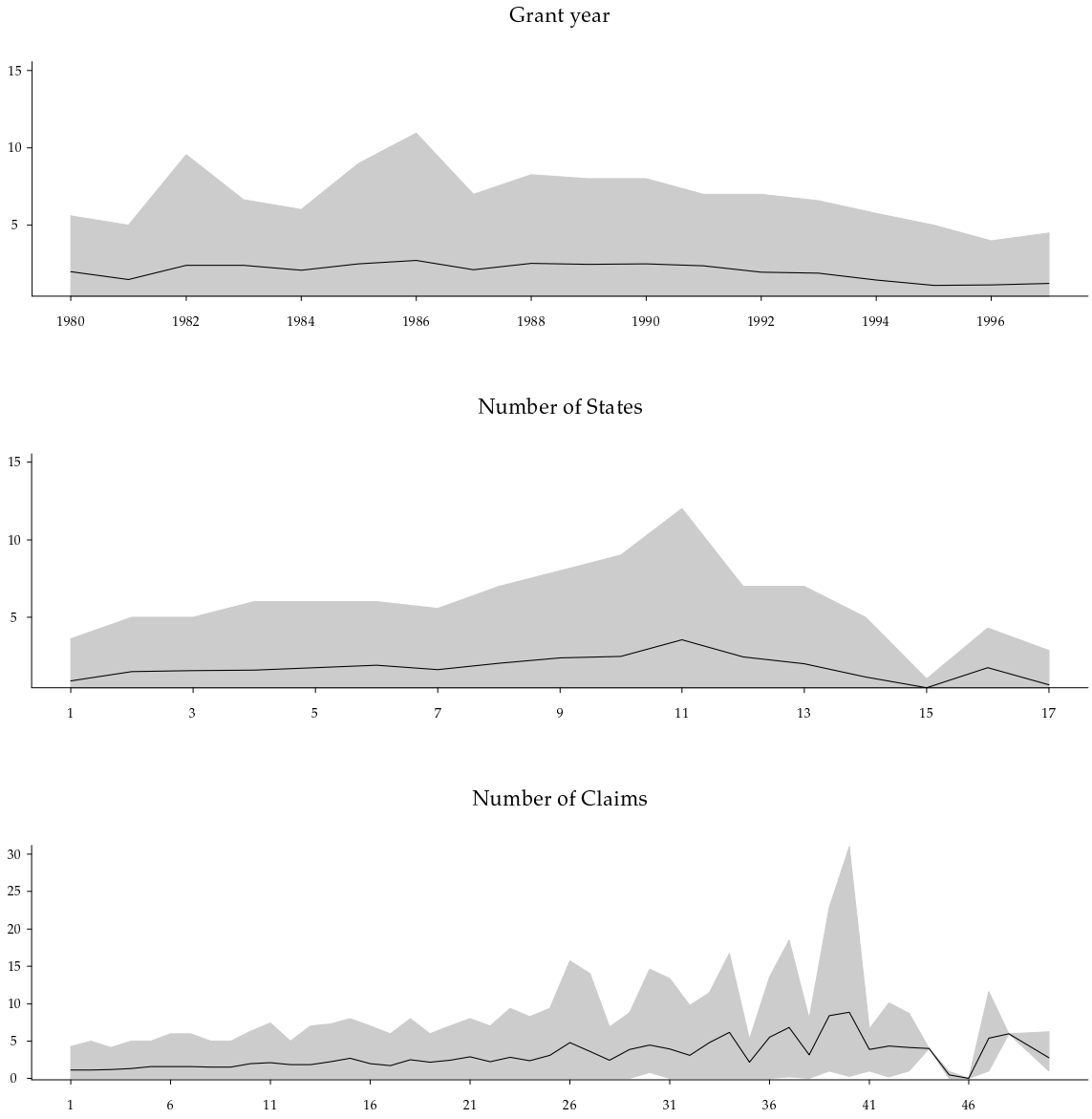

Figure 7.1: Plots for *forwcits* versus *gryear* (top), *nstat* (center) and *claims* (bottom). Given are pointwise mean (black line), and 5% and 95% quantiles (grey area) of *forwcits* for each different observed value of the metrical covariates.

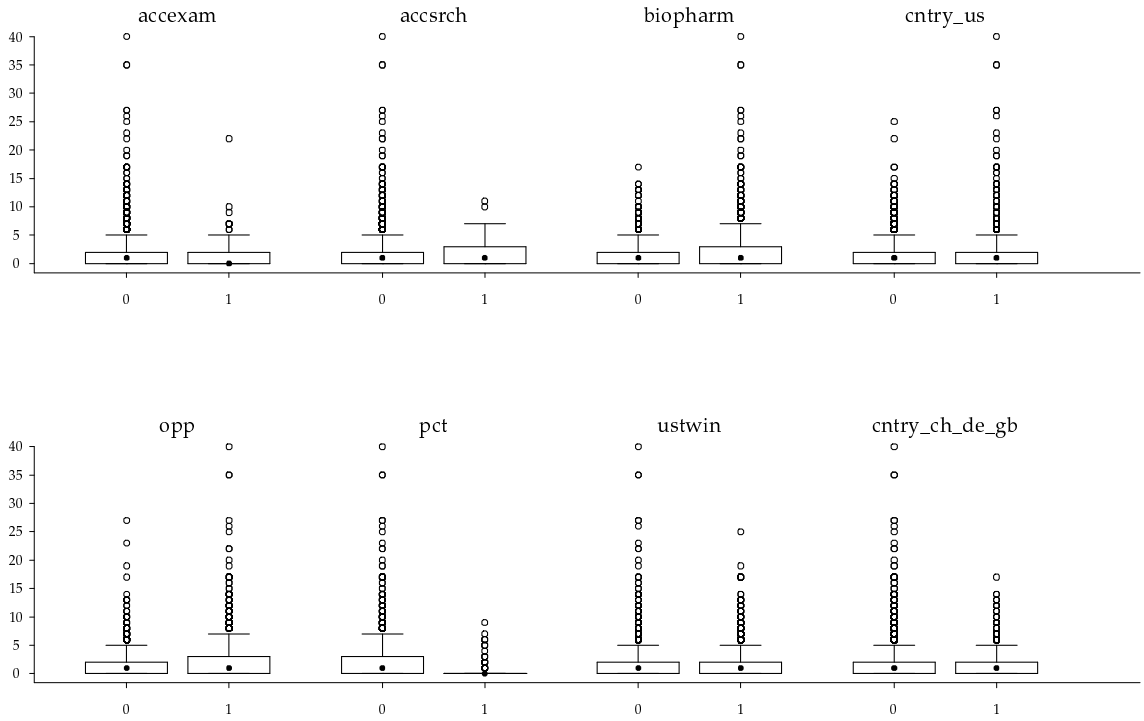

Figure 7.2: Box plots for $forwcits$ within the different categories of the binary covariates.

for zero inflation in the model. All the binary covariates are modeled as fixed effects with diffuse priors as given in (4.6). We include nonparametric terms in our regression to study nonlinear dependencies between the response variable ($forwcits$) and the metrical covariates $gryear$, $nstat$ and $claims$. All estimated models have the same predictor structure given by

$$\eta_i = z_i'\beta + f_1(gryear_i) + f_2(nstat_i) + f_3(claims_i) \tag{7.1}$$

$$\eta_i = z_i'\beta + f_1(gryear_i) + f_2(nstat_i) + f_3(claims_i) + \kappa_i, \tag{7.2}$$

where the second row is used for the POLN model. The vector z_i contains the binary covariates and also an intercept term, and f_j are cubic P–splines with 14, 14 and 20 knots for $j = 1, 2, 3$ respectively. We have choosen 20 knots for $claims$ because it has 50 different observed values in contrast to the 17 different values of $gryear$ and $nstat$.

All estimations are based on 5000 iterations and a burn in period of 1000 to ensure convergence. Each 4^{th} iteration value was stored to reduce the dependence of the chains, so that we have a sample of size 1000 for each parameter. The sampling paths show convergence for these values and the autocorrelations are satisfactory.

In the next subsection we present the results of the models. We first give some relevant conclusions for a preliminary analysis with the POGA model, concerning the covariate pct. Finally, present the results in more detail for the rest of the models.

7.1.2 Results

Preliminary analysis

In a first step we present the results obtained from a POGA model on our patent data. However, we found problems with the covariate pct, and recall the remarks made in the last subsection about the distribution of $forwcits$ within the two categories of pct. We take a look at Table 7.2 and Figure 7.3.

Figure 7.3 presents an interesting problem. The top panel gives a single point plot for the means of the posteriors for the ν_i terms in the given data order. We observe some sort of structure of the points on this plot. To make this structure clearer we ordered the data twice. First by $forwcits = 0$ or otherwise and second by $pct = 0$ or otherwise. The first ordering process provides two logical and clearly different regions on the bottom plot: on the left, the ν_i corresponding to patents with $forwcits_i = 0$ displaying a sort of 'broken line' form, and on the right for the rest, having a more or less 'cloudy' form. Accordingly

		Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
POGA	pct	-2.9867	0.1148	-3.2171	-2.9875	-2.7656
POLN	pct	-3.1601	0.1200	-3.4088	-3.1580	-2.9363

Table 7.2: Results for pct from the POGA and POLN models

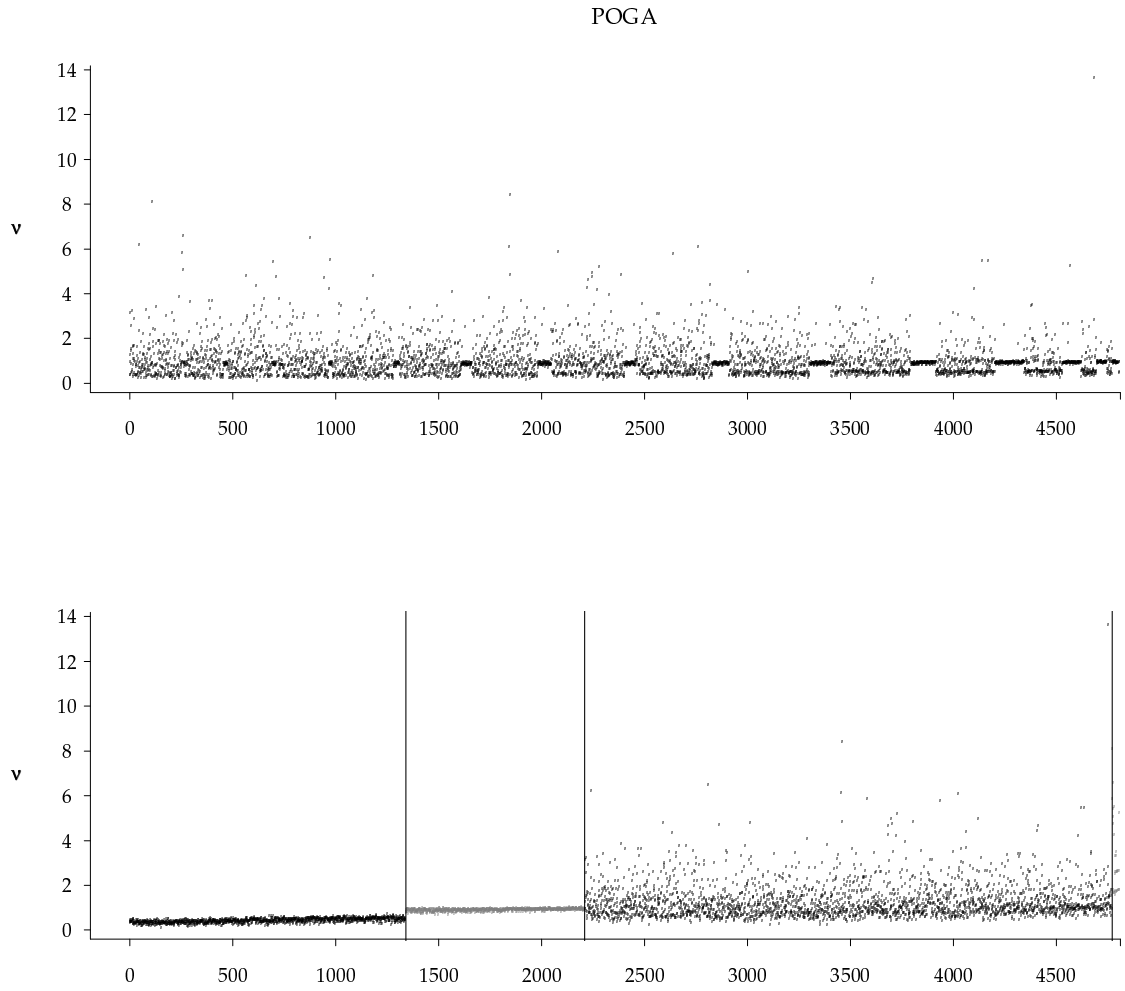

Figure 7.3: Mean of the posterior distribution for the multiplicative random effects in the POGA model. Top: in the given data order. Bottom: in the following order, $forwcits = 0$ and $pct = 0$, $forwcits = 0$ and $pct = 1$, $forwcits \neq 0$ and $pct = 0$, $forwcits \neq 0$ and $pct = 1$

with the data, the region on the left has smaller values (for the patents without forward citations) than the region on the right (for the patents with forward citations). The second ordering process is highlighted by the colors black (for patents with $pct = 0$) and grey (for patents with $pct = 1$). This second ordering causes the jumps in the by the first

ordering originated regions and is more visible in the left region because of the small number of patents with $forwcits \neq 0$ and $pct = 1$. As we can see in the plot, the values corresponding to $pct = 1$ are larger as those for $pct = 0$ within the same $forwcits$–group. This is a useful feature of models with latent variables: It allows us to explore the results for the ν_i terms and try to discover some new facts about the given data or possible missing covariates. The question now is what are the reasons for the observed patterns. It seems to be an identification problem due to the lack of variability of the response variable within the category 1 for pct. We have examined the same plot for the POIG and POLN model and found similar patterns. The estimated posterior mean for pct has similar values for all the models, but in Table 7.2 we only give the results for the POGA and POLN model. The posterior mean estimates are negative and therefore the terms $\nu_i \exp(\eta_i)$ are smaller for patents with $pct_i = 1$. But on the other side, exactly these patents get a slightly larger ν_i term as those with $pct = 0$, what makes $\nu_i \exp(\eta_i)$ increase.

The solution to overcome this problem is to split the data set into two parts. The first one for observations with $pct = 0$ is the data set that we will use in the remaining of this section for further analysis. The second data set contains all observations with $pct = 1$. Doing so, we have now 3900 observations in our data, corresponding to $pct = 0$. The mean of the response variable $forwcits$ is 1.9831 and its variance 8.2941. Its minimum and its maximum remain 0 and 40 respectively. With about 35% of the observations being zero and 95% being smaller or equal 7, the hypothesis of overdispersion persists.

Final models

From the results of our models on the patent data set with $pct = 0$ we obtain some main conclusions. We present them divided into three blocks. The first block contains the results concerning the models PO, NB, POGA and POLN. The second block is refered to the results from the ZIPGA model. Finally, the third block present the conclusions of the POIG and POIGH models.

First block: PO, NB, POGA and POLN

The results from the NB and POGA models are quite similar, as it was to be expected from the theory and confirmed by the simulation results of the last chapter.

Estimates for fixed effects from PO, NB and POGA models are very similar. We therefore only show the posterior mean estimations for POGA and POLN in Table 7.3. Both tables show some noticeable differences.

The first one is the posterior mean for the intercept. We should remember that the priori assumptions for the POLN were in some way different as for the POGA model. Actually every κ_i should have a $N(-0.5\tau_\kappa^2, \tau_\kappa^2)$ prior. But in our practical implementation their prior is $N(0, \tau_\kappa^2)$. The $-0.5\tau_\kappa^2$ term is equal for all κ_i and is therefore included in the intercept. Now we can adjust the posterior mean of the intercept in the POLN model by adding $0.5\widehat{\tau_\kappa^2} = 0.36955$ and see that it takes a similar value as in the POGA model.

To compare the posterior means of δ and τ_κ^2 we have the priori relationship $\delta = \frac{1}{\exp(\tau_\kappa^2)-1}$. Using the stored sampled values for τ_κ^2 and calculating the mean, we get 0.9170, which is smaller and not very close to $\delta = 1.2014$. This result is somehow a contradiction with the heavier tails of the LN distribution compared to those of the Gamma distribution. One could expect that the POLN distribution is able to capture the same amount of overdispersion in the data with a larger dispersion parameter as the POGA does.

From Table 7.3 we also see that zero is included in the credible intervals of the covariates *ustwin*, *accexam* and *accsrch*, so none of them has a significant effect.

Figure 7.4 shows that the observed problem with individual specific random effects has been eliminated for this model. The plots do not show any suspect pattern.

For the POLN model, we have transformed the estimated posterior means of the κ_i through the prior relationship $\nu_i = \exp(\kappa_i)$ in order to compare results with the POGA model. The patterns of their plot are similar to those presented here. The values range between 0.19 and 36.61 in the POLN model, and 0.12 and 13.11 in the POGA model, due to the smaller overdispersion parameter in the POLN than its equivalent in the POGA

POGA					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
const	0.6710	0.0820	0.5017	0.6729	0.8276
biopharm	0.2260	0.0575	0.1178	0.2250	0.3388
ustwin	-0.0625	0.0428	-0.1421	-0.0640	0.0227
accexam	-0.0994	0.1275	-0.3484	-0.1075	0.1428
accsrch	0.1028	0.1311	-0.1416	0.1019	0.3625
cntry_us	0.1568	0.0464	0.0667	0.1556	0.2473
cntry_ch_de_gb	-0.1957	0.0531	-0.3043	-0.1962	-0.0894
opp	0.4372	0.0405	0.3597	0.4377	0.5156
δ	1.2014	0.0474	1.1130	1.1986	1.3026
POLN					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
const	0.2949	0.0824	0.1204	0.2999	0.4492
biopharm	0.2058	0.0584	0.0990	0.2018	0.32323
ustwin	-0.0497	0.0440	-0.1369	-0.0484	0.0396
accexam	-0.0593	0.1288	-0.3107	-0.0563	0.1906
accsrch	0.1242	0.1482	-0.1773	0.1268	0.4061
cntry_us	0.1155	0.0489	0.0251	0.1160	0.2124
cntry_ch_de_gb	-0.2114	0.0537	-0.3176	-0.2129	-0.1051
opp	0.4690	0.0445	0.3800	0.4692	0.5513
τ_κ^2	0.7391	0.0351	0.6718	0.7377	0.8093

Table 7.3: Results for fixed effects and dispersion parameter from the POGA and POLN models

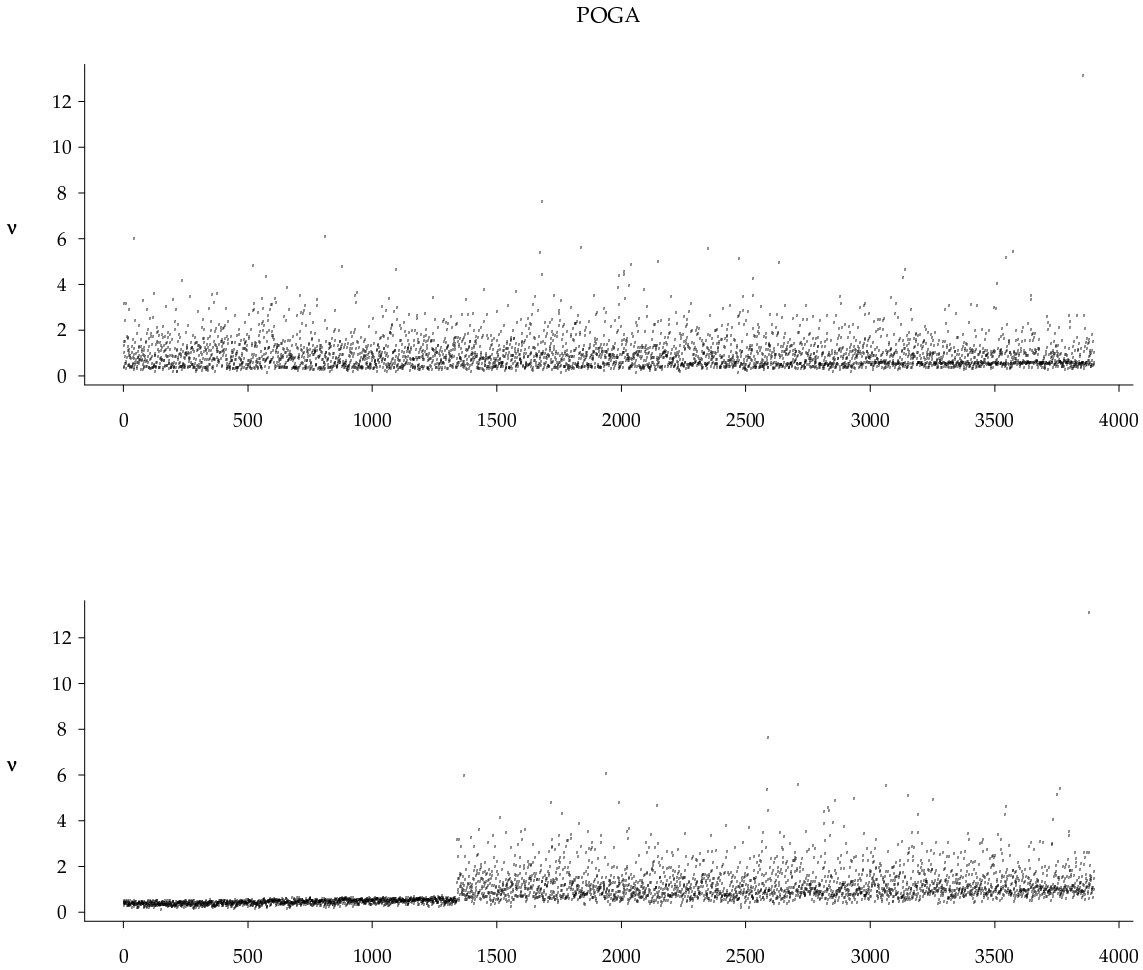

Figure 7.4: Mean of the posterior distribution for the multiplicative random effects in the POGA model. Top: in the given data order. Bottom: ordered by $forwcits = 0$ or $forwcits \neq 0$

one.

For the nonparametric terms, only the PO model shows relevant differences with the other models NB, POGA and POLN, that provide very similar results for the P–splines. Hence we only present the results for the NB and the PO models in Figure 7.5.

The credible intervals are constructed by computing the lower and upper posterior quan-

tiles corresponding to the respective nominal level, namely 2.5% and 97.5% quantiles for a nominal level of 95%. Note that the functions are centered about zero.

We observe that the estimated functions for the NB model are much smoother than those for the PO model.

The effect of *gryear* remains almost constant until approximate by 1987 and then begins to decrease. For *nstat* the effect also remains near constant for the first 11 values. And then it decreases when *nstat* goes toward 17. The estimated effect of *claims* is almost linear except at the end, which might result from the sparse data in the large categories, so the use of a spline is not necessary.

In Figure 7.6 we compare the results for $\hat{\nu}_i\hat{\mu}_i$ from POGA and POLN. As a consequence of the smaller estimated overdispersion parameter for the POLN model, the latter one seems to fit better, especially for large values of $forwcits$.

ZIPGA					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
θ	0.005293	0.004507	0.000165	0.004248	0.016614
δ	1.2285	0.0514	1.1350	1.2266	1.3378

Table 7.4: Results for the zero inflation and overdispersion parameters in the ZIPGA model

Second block: ZIPGA

We have also experimented with zero inflated models on the patent data. Table 7.4 gives the results for the zero inflation and overdispersion parameters. The estimated posterior mean for θ is almost zero. Hence we can conclude that there is no zero inflation in the model. Note that the estimated value for δ is very similar to that given in Table 7.3 for the POGA model.

It would be interesting to compare the obtained results with those of another statistical software. For this purpose we have used the zero inflated negative binomial regression

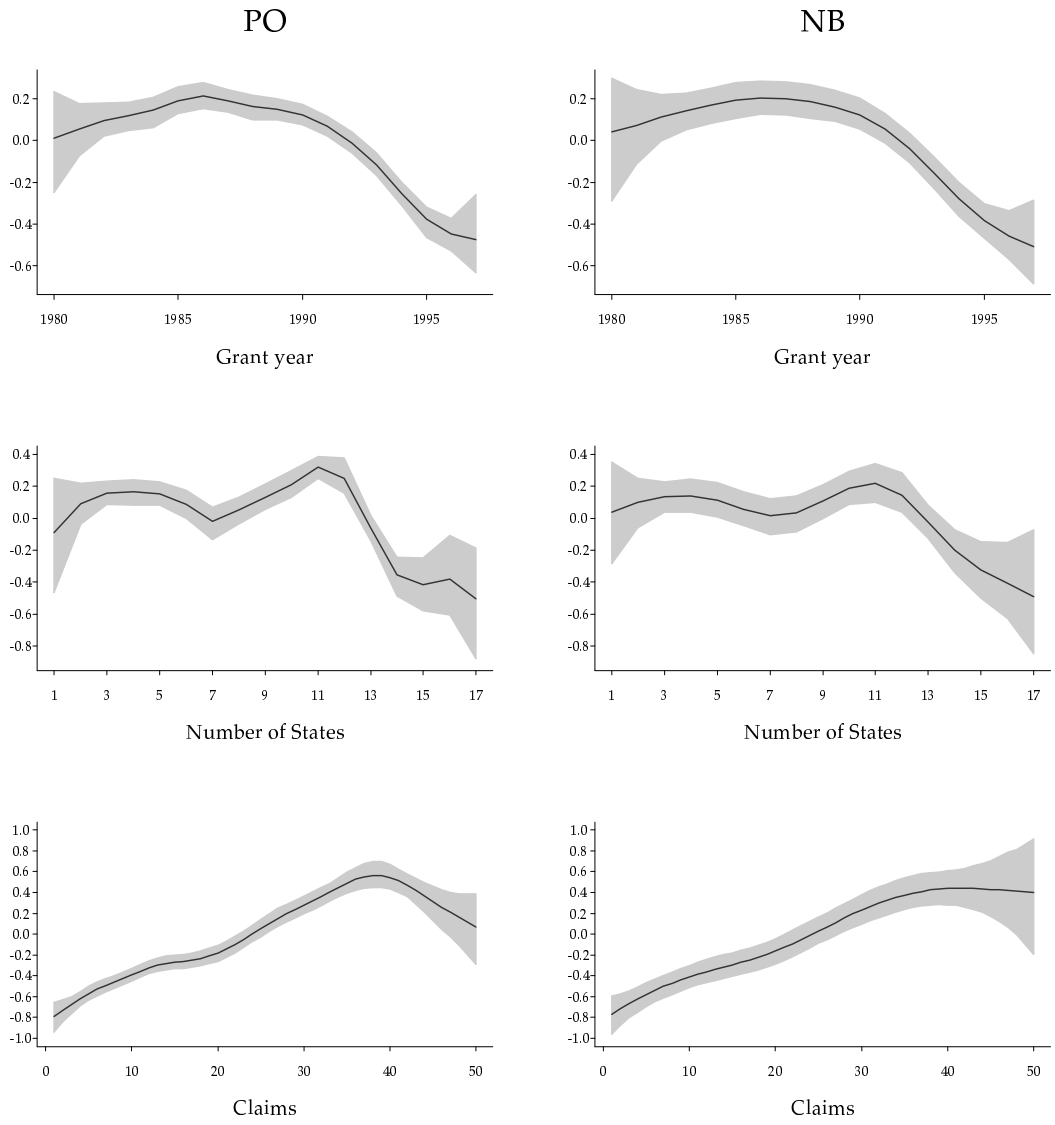

Figure 7.5: Estimated P–splines for the nonparametric terms in the PO and NB models together with pointwise 95% confidence intervals.

of Intercooled Stata 7.0. Unfortunately we can only specify a linear predictor, but we can take advantage of the robustness of the models with respect to the terms in the predictor. We have introduced the estimated posterior mean vector $\widehat{\eta}_i$ for the η_i terms from the NB

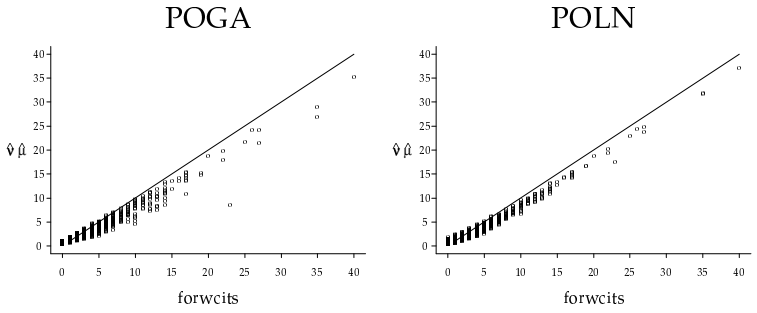

Figure 7.6: Response variable $forwcits$ versus posterior mean estimations for μ from NB, POGA and POLN models

model as a fixed effect ($linpred$) in the predictor. We expect to get the estimate of the intercept about zero, the coefficient for the $\widehat{\eta}_i$ vector about 1 and similar values as those given in Table 7.4 for the parameters θ and δ. Table 7.5 gives a summary of the results obtained with Stata.

```
Zero-inflated negative binomial regression        Number of obs   =       3900
                                                  Nonzero obs     =       2559
                                                  Zero obs        =       1341

Inflation model = logit                           LR chi2(1)      =     697.12
Log likelihood  = -7061.339                       Prob > chi2     =     0.0000

------------------------------------------------------------------------------
    forwcits |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
forwcits     |
     linpred |   1.013013   .0377229    26.85   0.000     .9390771    1.086948
       _cons |  -.0091485   .0310178    -0.29   0.768    -.0699422    .0516452
-------------+----------------------------------------------------------------
inflate      |
       _cons |  -27.74988   87782.29    -0.00   1.000    -172077.9    172022.4
-------------+----------------------------------------------------------------
    /lnalpha |   -.202745   .0431269    -4.70   0.000    -.2872721   -.1182179
-------------+----------------------------------------------------------------
       alpha |   .8164864   .0352125                      .7503076    .8885024
------------------------------------------------------------------------------
```

Table 7.5: Results for the zero inflation negative binomial regression model of Stata

The coefficient for the $linpred$ is very close to 1 and the estimated intercept has no significance in the model because it is almost zero. We transform $alpha$ and the inflate term

($inflate$) in order to compare them with the corresponding δ and θ from our model. The expression $\delta = \frac{1}{alpha}$ links both overdispersion parameters. We obtain 1.2285 for our ZIPGA model and 1.2248 for the ZINB of the Stata software. For the zero inflation parameters the linking function is $\theta = \frac{\exp(inflate)}{\exp(inflate)+1}$. Setting the value for $inflate$ we obtain 8.879331e^{-013} for the Stata model and the given 0.005293 for the ZIPGA model. Here the estimated parameters are not as close as for the overdispersion case. But both results point out that there is no indication of zero inflation in the data.

Third block: POIG and POIGH

We have also applied the POIG and POIGH models to the data. A consequence from the simulation study of the last chapter was that in the presence of nonparametric terms in the predictor using of POIGH does not improve the quality of the results compared with the POIG model. As expected the behavior of both models here is similar, both with inflated dispersion parameter when comparing it with the estimate from the NB or POGA models.

Summary

We conclude this section with a brief summary of our findings:

- The NB, POGA and POLN models could clearly identify overdispersion in the data. So they are preferable to a classical PO model.
- NB, POGA and POLN show similar estimation results for the predictor terms, which confirms the robustness of the models with respect to the underlying distribution for the multiplicative random effects.
- It also seems reasonable to include nonparametric terms in the predictor, as shown through the form of the estimated nonparametric effects for the metrical covariates.
- Concerning the estimation of the overdispersion parameter, the POGA and the NB model give similar results. The POLN model seems to fit better with a smaller δ.

- No indication of zero inflation could be found by running a ZIPGA model on the data.

As a last conclusion, it is always recommended to run a POGA or POLN model on the data and analyze the estimates for the individual specific random effects. With their help, we can identify outliers in the data or may discover specific patterns for some units.

7.2 Car insurance

Two main quantities are needed by a company to fix the premium for a policyholder: the estimated claim risk and estimated amount of loss per claim. These must be calculated very carefully to guarantee the competitive position of the insurance company on the market. If the insurer charges too large premiums, the policyholders will change their insurance company. But on the other side the firm has to keep profitable.

In this work we are going to concentrate on the modeling of claim frequencies and calculate the expected claim risk for each of the policyholders in the portfolio of a German insurance company. The response variable in the analysis is clearly of a count nature. We will apply both models for overdispersion and for zero inflation, presented in Chapters 2 and 3 respectively.

In the literature a large amount of papers concerned with car insurance analysis can be found. Dionne and Vanasse (1989) present the Poisson and the Negative Binomial regression with a linear predictor and use this regression to develop a bonus malus system on an individual basis. Tremblay (1992) also uses bonus malus system, but this time without covariates and based on a Poisson model, whose parameter is inverse Gaussian distributed. Schlüter, Deely and Nicholson (1997) fit a Bayesian Negative Binomial regression on the cumulated number of claims over 35 sites in Auckland, New Zealand. They do not include any spatial correlation or further covariates in the model. Jørgensen and Paes de Souza (1994) present a Tweedie's compound Poisson model to fit simultane-

ously claim frequency and claim severity. Their regression model has three parameters: a mean, a dispersion and a shape parameter. They only implemented parametric modeling of the covariates for the mean parameter. Smyth and Jørgensen (2002) extended the model presented in Jørgensen and Paes de Souza (1994) to include covariates in the modeling of the dispersion parameter. Brockman and Wright (1992) and Renshaw (1994) use Generalized Linear Models for the modeling of the claim frequency based on rating factors. The former article gives also a large overview in calculating premium rates for car insurance. Both indicate extensions for the response distribution as well as for the estimation methods to account for overdispersion. Boskov and Verrall (1994) have applied a spatial effect decomposition in structured (with a Markov Random Field, MRF) and unstructured (random effects) effect for a Poisson regression without further covariates and used Bayesian methods to make inference. Brouhns, Denuit, Masuy and Verrall (2002) have extended the Boskov and Verrall model by introducing a previous step. First, they fit a classical GLM Poisson regression, without spatial information. Afterwards, the results from the first model are included as an offset in the second step as described above. It is interesting that in their application they are not able to find significant unstructured spatial effects, which would be consistent with our simulation results.

Dimakos and Frigessi (2002) model claim frequency and claim severity through a hierarchical Bayesian model. They include the geographical information of the data set as a MRF, without estimating its hyperparameter in the model (ad hoc procedure), and as independent random effects per region, separately, but not both effects together.

In this section, we will base our analysis of the car insurance data set on the work of Fahrmeir, Lang and Spies (2003). They have implemented a hierarchical Bayesian regression with a Poisson assumption for the response variable. They included fixed and random effects as well as nonparametric effects (modeled through P–Splines) and spatial information (split up in structured and unstructured effects) in what is called a geoadditive model. Our expansion of the model is based on the generalization of the response distribution to account for overdispersion or/and zero inflation, as presented in Chap-

ters 2 and 3. In the following, we present the data and the models in some detail, and afterwards the obtained results and the conclusions we can draw from them.

7.2.1 Data and model description

We apply structured count data regression models to a data set of 200681 individual claim frequencies of a sample of policyholders with full comprehensive car insurance for one year. Among others, the covariates given in Table 7.6 were included in the predictor. The aim is to analyze the dependency of the number of claims *claims*, as response variable, on these covariates. To make the data source anonymous, some additional covariates used for the analysis are not described in the paper.

The covariate driven kilometers per year (*km*) is a metrical variable. It is conceivable that increasing the number of driven kilometers also increases the probability of having accidents. The covariate car classification (*car*) is an ordinal covariate indicating the potential risk of a car type, from low to high and *bonus* reflects how long an insured car has been driven without accident until now in increasing order.

The car insurance data set that we are going to analyze is not free of problems. For the response variable *claims* we have over 96% zero observations, its mean is 0.03987, its variance 0.04133948 and its maximum 4. The maximum is observed only three times. That means, we have not too much variation in the data to discover effects.

In Figure 7.7 we have three plots of the mean (black line) and the 5% and 95% quantiles (grey area) of *claims* within the different observed values of the metrical covariates. These plots reflect the main problem of the car insurance data commented above. The mean of the response variable within the categories is very small.

The same information is reproduced in Figure 7.8. There we have box plots for *claims* within the two categories of the three binary covariates, noting that all the principal quantities in all the six box plots (upper extreme, upper quartile, median, lower quartile, and lower extreme) are zero.

The response variable exhibits an extremely large proportion of zeros. On the other hand, the fact that the maximum of *claims* is 4 and that its mean is quite low (0.03987) but smaller than its variance leads to the conclusion that both overdispersion and zero inflation should be studied here.

Claim frequencies were analyzed with structured additive NB, POGA, POIG and POLN regression. We use some of the results obtained in Fahrmeir, Lang and Spies (2003) about the PO model for comparison. ZIP, ZINB and ZIPGA models were also tested to check for zero inflation. The predictor is defined for all the models by

$$\eta_i = lndur_i + z_i'\beta + f_1(km_i) + f_2(bonus_i) + car_i + f_{spat}(district_i) + \ldots \tag{7.3}$$

Offset	
lndur	logarithmed duration of the policy (in days)
Metrical covariates	
km	kilometers driven per year in thousands
car	car classification, measured by $G = 31$ scores from 10-40
bonus	no–claims bonus, defined by 27 classes from 0–25
others	
Binary covariates (yes = 1, no = -1)	
garage	garage available
tariff	civil servants and coequal professionals/others
ownpart	deductible
others	
Spatial covariate	
district	district in Germany ('Zulassungsbezirk' resp. 'Landkreis'), with $S = 438$ districts

Table 7.6: Some of the variables in the car insurance data set.

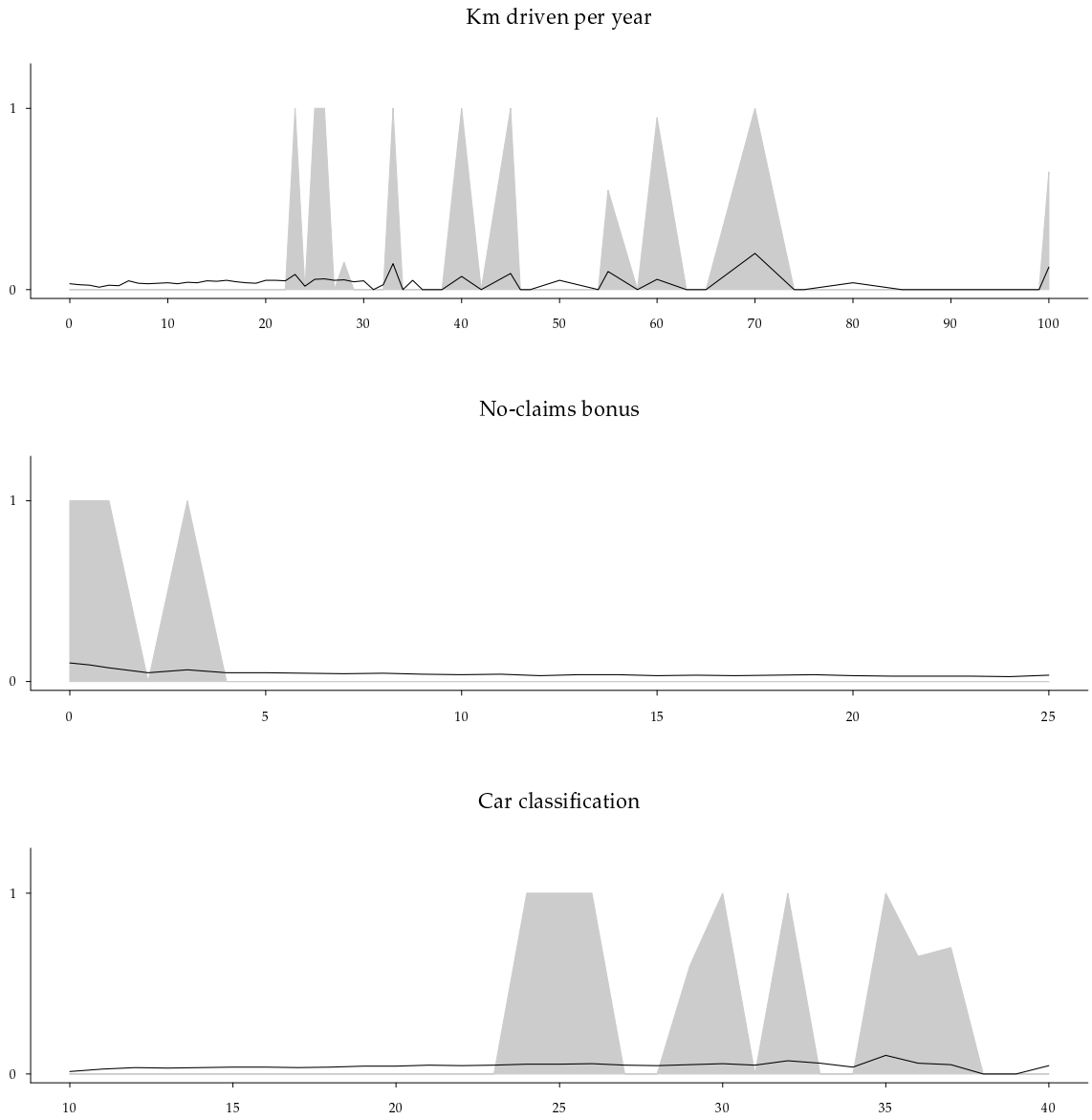

Figure 7.7: Plots for *km* (top), *bonus* (center) and *car* (bottom) versus *claims*.

$$\eta_i \quad = \quad lndur_i + z_i'\beta + f_1(km_i) + f_2(bonus_i) + car_i + f_{spat}(district_i) + \kappa_i + \ldots. \quad (7.4)$$

The second row is used if the model is POLN. The dots indicate that the predictor comprises additional metrical and binary covariates not shown for reasons of confidentiality.

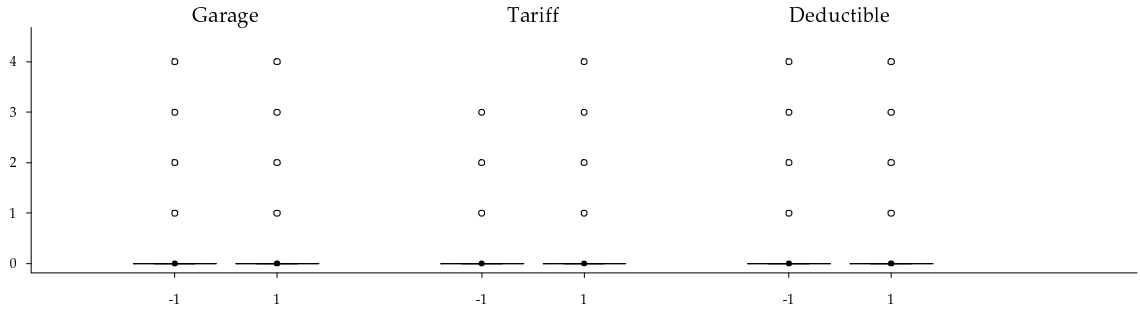

Figure 7.8: Plots for the mean (black line) and the 5% and 95% quantiles (grey area) of *claims* within the different categories of the binary covariates.

The spatial effect $f_{spat}(district)$ is further split up into the sum of structured and unstructured effects, i.e.

$$f_{spat}(district) = f_{str}(district) + f_{unstr}(district).$$

The vector z_i contains the categorical covariates and an intercept term *const*. The effects f_1 and f_2 of the metrical covariates are modeled by cubic P–splines, each of them with 20 knots. The effect *car* of car classification and the unstructured spatial effect f_{unstr} are treated as i.i.d. random effects, and for the structured spatial effect f_{str} a Markov random field prior is used.

All estimations were executed with 30000 iterations and a burn in period of 5000 to ensure convergence. Each 25^{th} iteration was stored to reduce the dependence of the chains, so that we have a sample of size 1000 for each parameter to make inference. For the POGA and POLN models the mixing of the chains for these inputs was not satisfactory enough, so we rerun the programs with 75000 iterations, 5000 burnin and a thinning of 70.

The posterior estimates presented in the following are calculated as the empirical corresponding values from the stored chains of the posterior distributions.

7.2.2 Results

In the following we summarize the results from the different models on the car insurance data set. The results are presented in three blocks. The first block contains the PO, NB, POGA and POLN models. The second block summarizes the results for the zero inflated models (ZIM) ZIPGA and ZINB. And the third block gives some comments about the results of the POIG and POIGH models.

First block: PO, NB, POGA and POLN

POGA and NB model are very close in their results. The main difference between both models is given by the posterior distribution of the scale parameter. For the NB model the sampling path shows convergence and the posterior mean estimate is 1.4275, while the sampling path in the POGA model is far from convergence, even for the 75000 iterations case, and the posterior mean estimate is 1.3312. This divergence may be due to the large number of parameters, where the estimation of δ in the POGA model is based on. But both values for the posterior mean estimate of δ are consistent with the hypothesis of overdispersion.

We take a look at Table 7.7. There we find a summary for the fixed effects and the dispersion parameters for the NB and POLN models. The immediate conclusion is that the results are very robust, independent of the model we are using. For the difference between the intercepts we must argue as explained in Section 7.1. The estimated posterior mean for the POLN model should be corrected by adding $0.5\widehat{\tau_\kappa^2} = 0.226$ before we can compare it with the value resulting from the NB model.

As we know, the priori relationship between δ and τ_κ^2 is given by $\delta = \frac{1}{\exp(\tau_\kappa^2)-1}$. Plugging the corresponding sample values for τ_κ^2 into this formula and calculating the mean, we obtain 1.8033. This value is larger than the estimate 1.4275 for δ in the NB model.

In this application the POLN model behaves completely different as in the patent data application of Subsection 7.1,which could be a consequence of the difference in the ranges for the response variables between both data sets.

NB					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
const	-8.2047	0.1865	-8.5581	-8.2014	-7.8594
garage	-0.0263	0.0142	-0.0534	-0.0262	0.0009
tariff	0.0209	0.0145	-0.0066	0.0210	0.0489
ownpart	-0.0304	0.014	-0.0587	-0.0303	-0.0025
δ	1.4275	0.2080	1.1029	1.4009	1.8993
POLN					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
const	-8.4352	0.1932	-8.8447	-8.4285	-8.0670
garage	-0.0251	0.0146	-0.0539	-0.0246	0.0039
tariff	0.0210	0.0139	-0.0056	0.0202	0.0502
ownpart	-0.0305	0.0142	-0.0569	-0.0304	-0.0011
τ_κ^2	0.4521	0.0662	0.3031	0.4586	0.5731

Table 7.7: Results for fixed effects and dispersion parameter from the NB and POLN models

From Table 7.7 we also see that the covariates *garage* and *tariff* are not significant because zero is included in their credible intervals. To have a policy with cost sharing seems to reduce the risk of reporting a claim.

The posterior mean estimates of the functions f_1 and f_2 and of the random effect for the car classification variable, together with 95% pointwise credible bands are displayed in Figure 7.9 for the PO and NB. The credible intervals are constructed in a similar way as described in the patent data application. The functions are centered about zero. In contrast to the patent data application in the last section, we do not see relevant differences between the results for the nonparametric terms from the PO and the rest of the models.

The effect of kilometers driven per year shows a distinct, almost linear increase until

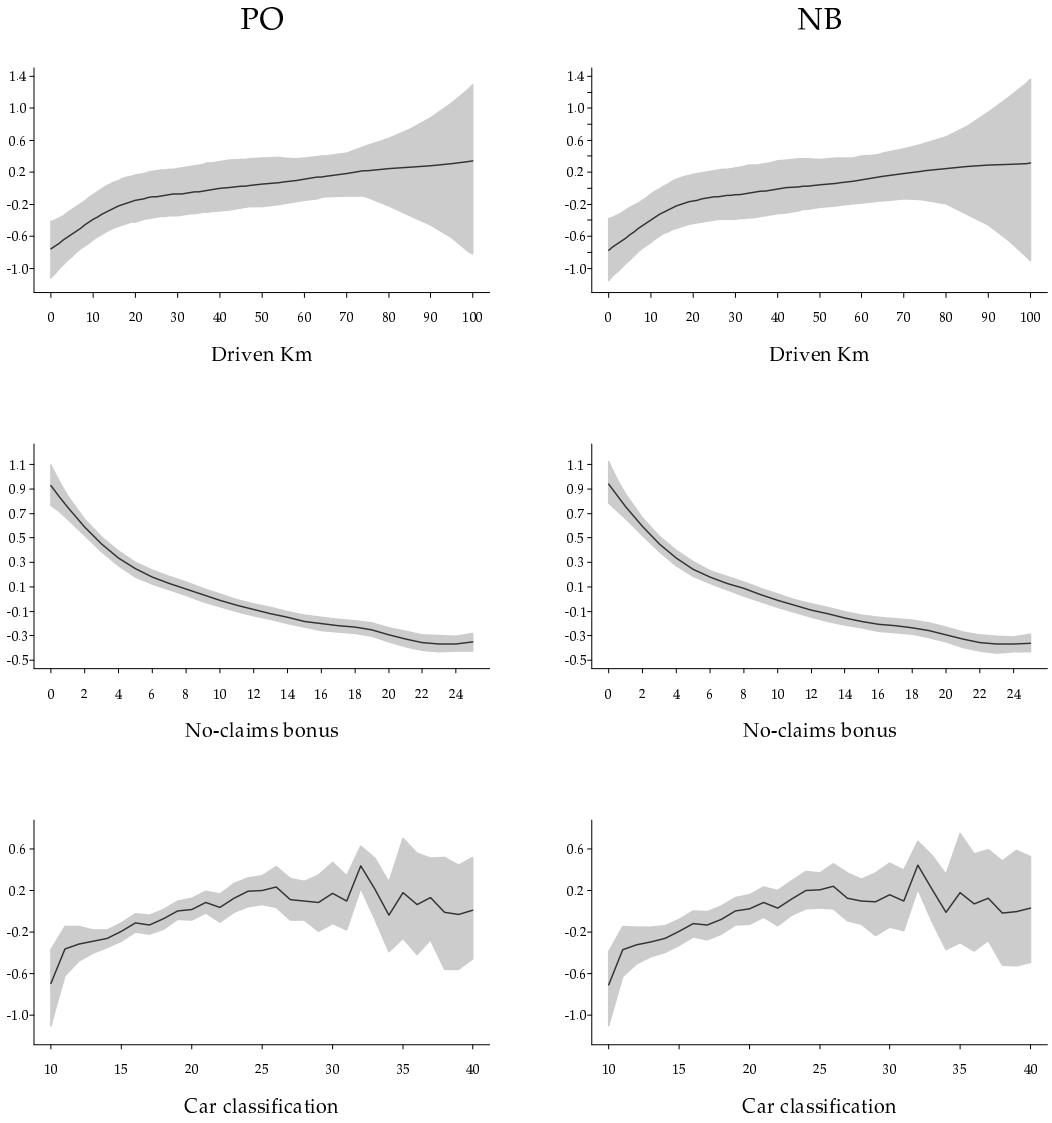

Figure 7.9: Estimated P–splines (black lines) for the nonparametric terms *km*, *bonus*, and *car* in the PO and NB models together with pointwise 95% credible intervals (grey areas)

about 20 000 km/year. Thereafter, the increase becomes much smaller. Looking at the credible bands, even a constant effect cannot be rejected. A possible explanation is that these frequently used cars are driven by experienced persons and, probably, to a larger

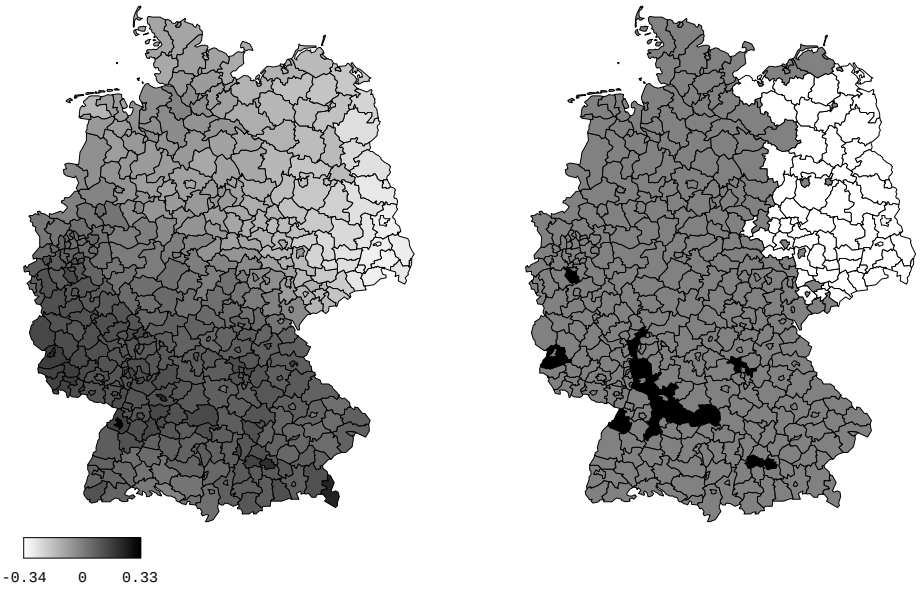

Figure 7.10: Structured spatial effect for POLN. The left panel shows the posterior mean, the left panel displays posterior probabilities based on nominal levels of 95%. White colored regions correspond to strictly negative credible intervals and black colored regions to strictly positive intervals. Districts with credible intervals containing zero are colored in grey.

extent on a freeway than others.

The form of the effect for the covariate *bonus* confirms the classification of the insurance company. Clearly, the effect decreases for increasing value of *bonus*, what means that for cars, which have not reported a claim for a long period, the risk decreases.

Because the covariate car classification was considered as a group indicator with a random effects assumption, the estimated function looks considerably rougher than the other function. It shows an increasing trend until about category 33 that is coherent with the intended definition of the groups. The decreasing trend of the posterior mean line and the wider credible bands after this category may be due to sparse data in these last categories.

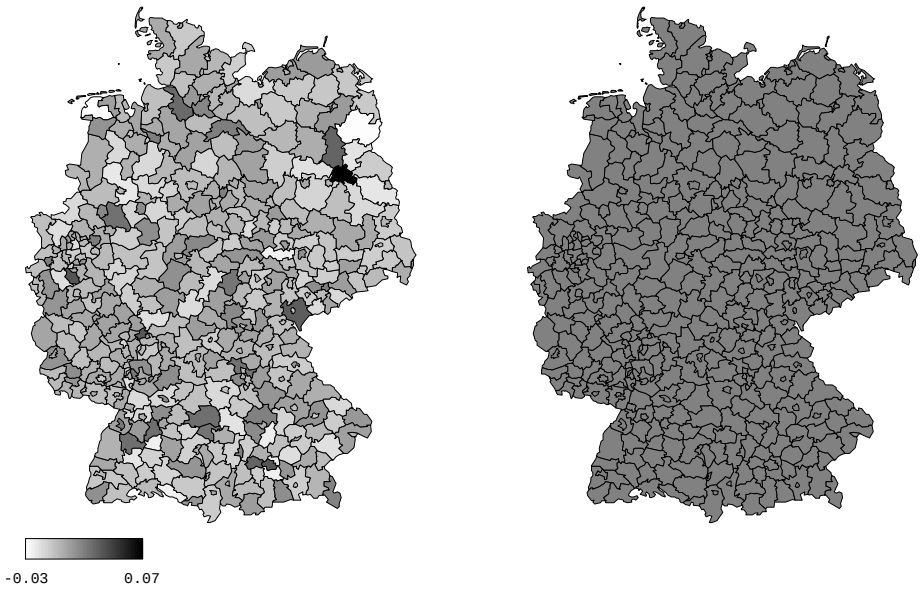

Figure 7.11: Unstructured spatial effect for POLN. The left panel shows the posterior mean, the right panel displays posterior probabilities based on nominal levels of 95%. White colored regions correspond to strictly negative credible intervals and black colored regions to strictly positive intervals. Districts with credible intervals containing zero are colored in grey.

Let us now turn to the geographical, district-specific effects. In Figures 7.10-7.12 we have displayed the results for the POLN model only, as there are no great differences between the models. The left map of Figure 7.10 shows the posterior means for the structured effects f_{str} displaying a smooth but very clear regional pattern: there is a clear decline from southwest to northeast. This is confirmed by the 95% 'significance maps' in the right map of Figure 7.10. White colored regions correspond to strictly negative credible intervals (i.e. a 'significant negative effect') and black colored regions to strictly positive credible intervals (i.e. a 'significant positive effect'). Districts with credible intervals containing zero are colored in grey. The left map in Figure 7.11 shows the posterior means of the unstructured effects f_{unstr}. We cannot observe any typical pattern in this plot, and

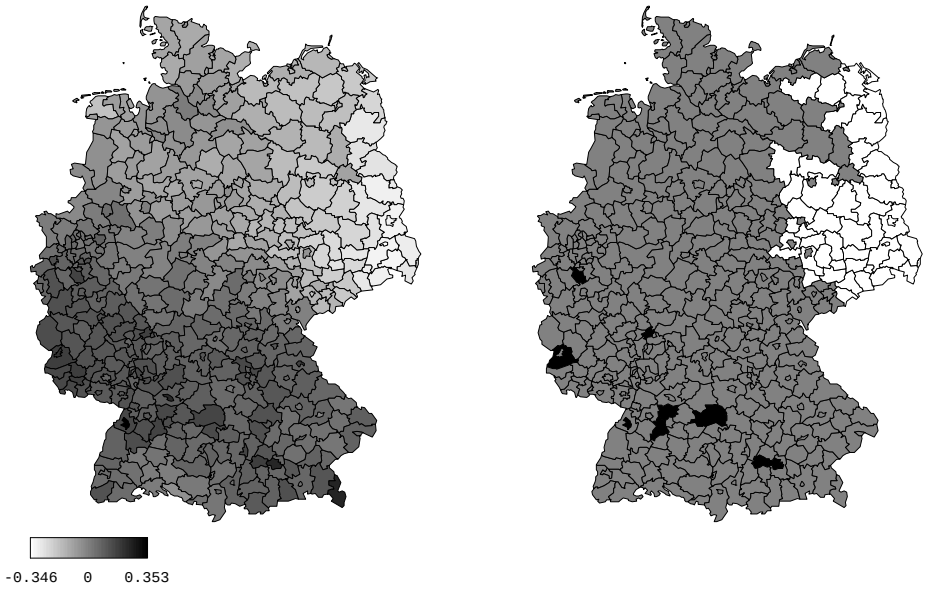

Figure 7.12: Sum of the structured and the unstructured spatial effect for POLN. The left panel shows the posterior mean, the right panel displays posterior probabilities based on nominal levels of 95%. White colored regions correspond to strictly negative credible intervals and black colored regions to strictly positive intervals. Districts with credible intervals containing zero are colored in grey.

accordant with the results of the simulation study in Chapter 6, the unstructured, local effects are much smaller than the corresponding structured effects. This is confirmed by the significance map in the right part: no district has significant effect for a nominal level of 90%. The maps for the sum f_{spat} of structured and unstructured effects in Figure 7.12 resemble the maps in Figure 7.10, but are less smooth. Table 7.8 gives a summary of the significant positive/negative or non significant effects for the total geographical effect in the PO, NB, POGA and POLN models. There we see that particularly the last three models mostly agree in the classification.

Figure 7.13 displays box plots for the estimated posterior means of the multiplicative random effects within the different observed values for *claims* from 0 to 4 for the POGA

	sign. negative	non sign.	sign. positive
PO	45	383	10
POGA	46	384	8
NB	46	384	8
POLN	46	383	9

Table 7.8: Number regions with negative significant, no significant and positive significant total geographical effect in the different models.

and POLN models. For the latter one, the values have been transformed in order to compare them with the POGA results. The transformation is based on their prior relation, given by $\nu_i = \exp(\kappa_i)$. For $claims = 4$ we have only plotted the estimated posterior means for the three observed values. The pattern is clearly shown. The ν_i build clusters associated with the value of the response variable. This is an undesirable effect, because this is a sign for insufficient explanation of the response variable through the covariates. The estimated values for μ_i are not large enough to fit observed responses greater than zero, and the ν_i have to account for this lack of approximation, as we see in the well defined jumps between the box plots for increasing response value.

We suggest two interpretations for this behavior. The first one is related to the lack of information available in the model. The used covariates can explain only a very small part of the response variable and the main explanation relays on the individual specific random effects, which by their definition should account for unavailable information in the model. In this case there is no much statistical work to do. It would be important to consider, which covariates could be strongly related to the number of claims and to collect new information.

The second interpretation is related to the immense amount of zero observations in the data set. We may have too little variation in the response, dominated by zero responses, to extract the information contained in the covariates. One solution to this problem is to

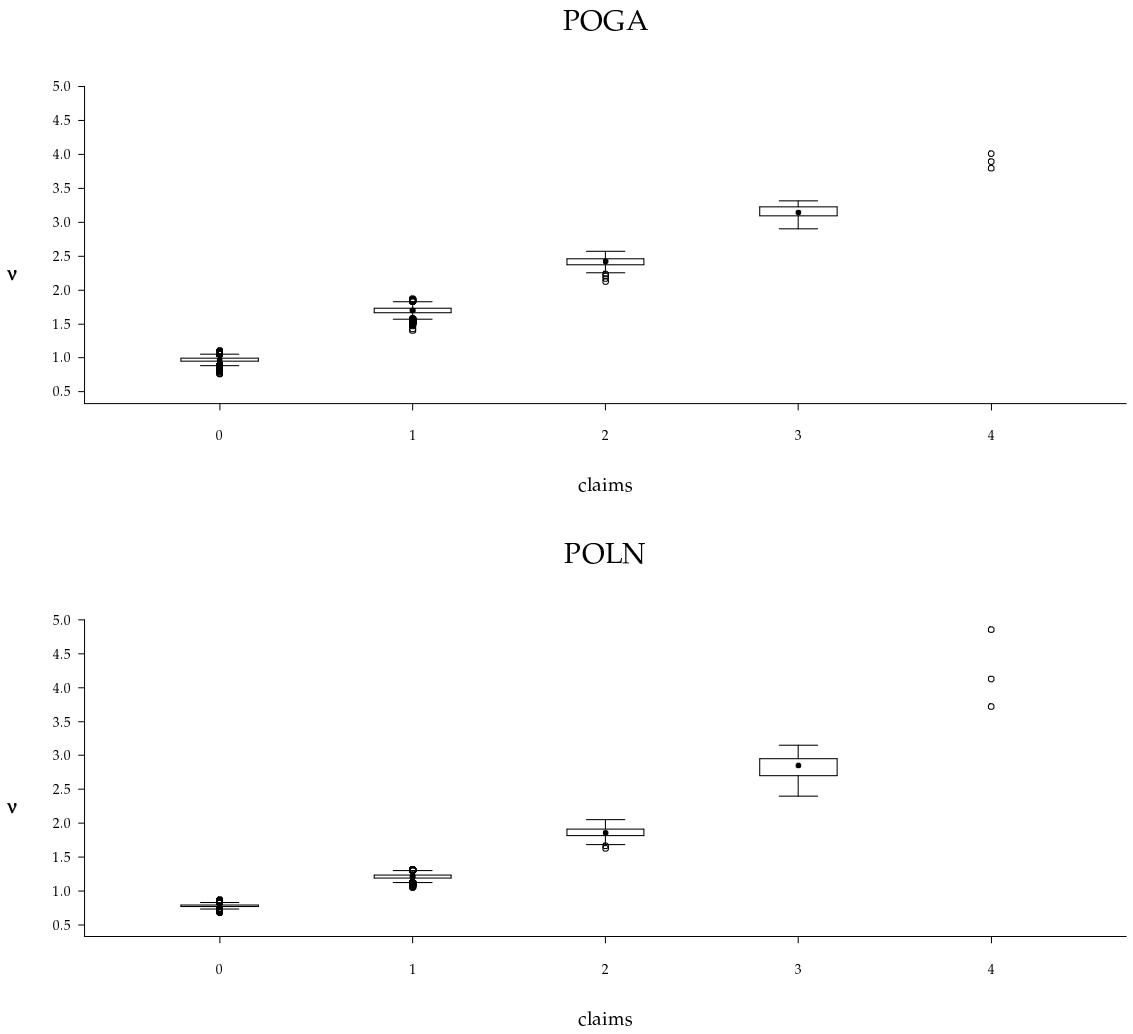

Figure 7.13: Box plots for the estimated posterior means of the individual specific random effects of the POGA (top) and POLN (bottom) models split by the response variable.

test zero inflated models on the data. These could account for a differentiating modeling for zero and non zero observations.

Second block: ZIPGA and ZINB

From an interpretational point of view, ZIMs are attractive in car insurance applications. We can differentiate between two kind of zero observations: The first class consists on

those zeros, where actually no accidents have been produced, which corresponds to the situation where the underlying count data process is zero, independently of the value of the latent selection process. The second class consists on those claims caused e.g. by small car body damages that were not reported to preserve the no–claim bonus of the insured. In this case, the underlying count data process is not zero, but the selection process.

We have tested both models ZINB and ZIPGA. As the results for both are very similar, we present the results for the ZIPGA model only.

In Table 7.9 we give the results for the fixed effects, zero inflation and overdispersion parameter for the ZIPGA model applied to the car insurance data set. As we see, there are only minor differences between the results exposed in Table 7.7. Note the discrepancy in the intercepts: We have already explained this for the POLN case. Now, for the ZIPGA model, we must recall that in zero inflated models the marginal mean assumption is different from the one in overdispersion models. In the former we have $E(y_i|\cdot) = (1-\theta)\mu_i$ and in the later we have $E(y_i|\cdot) = \mu_i$. Because our modeling implies a constant θ for all the observations, the factor that precedes μ_i will be compensated with the intercept by $\log(1-\theta) = -0.072809$. So adjusting the intercept of the ZIPGA model by adding -0.07280889 bring us closer to the estimated posterior mean of the intercept in the NB model. The rest of the fixed effects remains quite unaltered, which is once again a proof of the robustness of the estimation for the predictor, independently of the chosen model.

The estimation of the P–splines and the random effect term results in very similar plots as presented in Figure 7.9, which also holds for the results of the geographical terms.

In Table 7.9 we also see the estimated posterior means for θ and δ. The value for θ is very small, but even large enough to increase the value of δ with its presence in the model, when comparing it with the value of the overdispersion parameters in Table 7.7. That means, running the model under the assumption of zero inflation decreases the estimated overdispersion parameter. In ideal case, the box plots in the Figures 7.13 and 7.14 should be placed around the base line 1. In Figure 7.14 we can see that the range of the estimated

ZIPGA					
	Mean	STD	2.5%-Quant.	Median	97.5%-Quant.
const	-8.1420	0.2102	-8.5765	-8.1381	-7.7271
garage	-0.0254	0.0150	-0.0545	-0.0257	0.0040
tariff	0.0211	0.0146	-0.0074	0.0208	0.0500
ownpart	-0.0309	0.0141	-0.0590	-0.0308	-0.0016
θ	0.0702	0.0472	0.0038	0.0636	0.1810
δ	1.7244	0.1829	1.4511	1.6834	2.0856

Table 7.9: Results for the fixed effects, zero inflation and overdispersion parameters in the ZIPGA model

posterior means for the random effects shrinks to about 3. But we still have the jumps between the classes defined by the response variable and the model is far away from the optimal case.

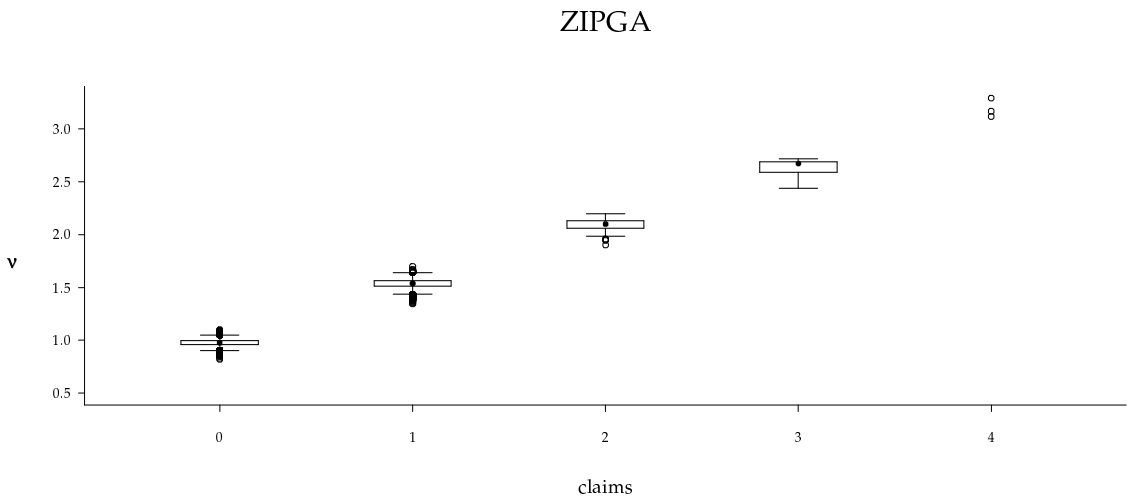

Figure 7.14: Box plots for the estimated posterior means of the individual specific random effects of the ZIPGA model split by the response variable.

Now we would like to compare our results with those of other statistical software. For this purpose we have run some tests using Intercooled Stata 7.0. With this software we can calculate generalized regression models with a zero inflated Poisson or zero inflated negative binomial response distribution. The problem is that only linear terms are allowed in the predictor. So we can not estimate exactly the same model as with *BayesX*. What we have done is to run a zero inflated negative binomial regression model only with a linear effect in predictor ($linpred$), given by the vector of estimates $\widehat{\eta}_i$ from the NB model, and a constant intercept.

```
Zero-inflated negative binomial regression        Number of obs   =     200681
                                                  Nonzero obs     =       7719
                                                  Zero obs        =     192962

Inflation model = logit                           LR chi2(1)      =    1084.54
Log likelihood  = -33395.85                       Prob > chi2     =     0.0000

------------------------------------------------------------------------------
          sh |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
sh           |
     linpred |    .544745     .01809    30.11   0.000     .5092892    .5802007
       _cons |  -1.415536   .0589386   -24.02   0.000    -1.531054   -1.300019
-------------+----------------------------------------------------------------
inflate      |
       _cons |   -9.81249   48.78934    -0.20   0.841    -105.4378    85.81286
-------------+----------------------------------------------------------------
    /lnalpha |  -.4613818   .1514733    -3.05   0.002    -.7582641   -.1644996
-------------+----------------------------------------------------------------
       alpha |   .6304119   .0954906                      .468479    .8483181
------------------------------------------------------------------------------
```

Table 7.10: Results for the zero inflation negative binomial regression model of Stata

As we observed in the output of Table 7.10, the estimated coefficient for *linpred* is 0.544745 and thus far away from 1. And the estimate for the intercept is not zero, as we could expect, but has a negative value. This is somehow surprising, because if the information in the covariates is not enough to explain the response variable (as the multiplicative random effects from the ZIPGA model insinuate), then we would not expect, that the new estimated predictor adopts smaller values than the old one.

We recall the formula that related δ and θ with the parameters $alpha$ and $inflate$. For the

former $\delta = 1/alpha$. Plugging in the corresponding value we get 1.5863 for the overdispersion parameter from the Stata model and 1.7244 for the ZIPGA model. For the zero inflation parameter it holds $\theta = \frac{\exp(inflate)}{\exp(inflate)+1}$. Consequently we have 0.00005476032 for the Stata model and 0.0702 for the ZIPGA one. There are significant differences between the results of both models. This gives further evidence for the insufficiency of the ZIPGA model for this car insurance data set.

Third block: POIG and POIGH

As expected, POIG and POIGH are not able to find overdispersion in the data. POIG could not be applied for the planned 25000 iterations. A test run with 2000 iterations shows that the posterior estimate for δ moves around 2.39729e+06. This extreme large value causes numerical problems in the program and produces its crash. The POIGH model shows in this aspect a slightly but not relevant improvement. The program also did not run for the desired 25000 iterations, but with a shorter run of 2000 we obtained 207980 as posterior mean estimate for δ, which is anyway a smaller value as the one obtained from the POIG model.

Summary

We conclude with a short overview of the presented results.

- First, none of the models is optimal for the car insurance data set. This may be due to the structure of the data, with a great disproportion of zero counts versus nonzero counts.

- Second, we can say that NB, POGA, ZIPGA and POLN models could clearly identify overdispersion in the data. So they are preferable to a classical PO model.

- All of them show similar estimation results for the predictor, for the fixed effects, for the random effects, for the P–Splines, as well as for the geographical covariate. This gives evidence for the robustness of the models with respect to the underlying distribution of the multiplicative random effects.

- The inclusion of nonparametric terms in the predictor seems to be reasonable, as it is shown through the form of the estimated nonparametric effects for the metrical covariates. Also the spatial effects account for significant differences between the regions.

- For the estimation of the overdispersion parameter we find some discrepancies. The POGA and the NB model give similar estimates, but not the same. The POLN model seems to fit better with a smaller δ.

- By introducing zero inflation in the model (ZIPGA) we do not get a large value for the θ parameter, but it is enough to alter the estimate of δ and make it larger. So we can not ensure that there is almost no zero inflation in the model.

- Comparing the results with those of the Stata software confirms us, that even the more complete model presented here (ZIPGA) is not good enough for this car insurance data set. A possible alternative for the modeling of this car insurance data set are underreporting models (see Winkelmann (1996)).

We have experienced that it is always recommended to run a ZIPGA, POGA or POLN model on the data and analyze the estimates for the individual specific random effects. Together with the advantages stated at the end of the last section, they are also very helpful in model assertion.

An interesting further development for the presented overdispersed and zero inflated models would be the implementation of possible dependences of overdispersion and zero inflation parameters on covariates. This may improve the estimation results, bring more flexibility in to the models and could be interesting for the interpretation of the results.

Chapter 8

Bayesian Count Data Regression with *BayesX*: A tutorial

The focus of this chapter is on showing how count data can be analyzed in *BayesX*. For this purpose, we describe how to estimate some of the regression models discussed in Section 7.1 to analyze the patent data. All the models presented in this work are implemented in this program. Three semiparametric regression models are selected. First, we apply a classical Poisson (PO) regression model, where the data given the covariates are supposed to be Poisson distributed. Second, we estimate a regression model that allow for overdispersion in the data, namely, a Poisson–Gamma (POGA) regression. In this model the data are supposed to be Poisson distributed given the covariates. The difference to the Poisson regression is that in addition to the given covariates we also estimate a vector of individual specific random effects, that is supposed to have i.i.d. components with gamma prior. Finally, the data will be tested for zero inflation with the help of a Zero Inflated Poisson–Gamma (ZIPGA) model. This is an extension of the POGA model, where a zero inflation parameter is introduced. For a more detailed explanation of the models we refer to Chapters 2 and 3.

This chapter is structured as follows. Section 8.1 presents the software package *BayesX*.

A description of the general use of *BayesX* and some comments about its structure are given in Section 8.2. Section 8.3 describes how to handle and manipulate data sets with the program. The main aspects on count data regression are shown in Section 8.4. In the last sections we describe the methods that are implemented in *BayesX* to plot and analyze regression results.

8.1 *BayesX*

BayesX is a software tool for performing complex Bayesian inference. Among other features *BayesX* supports Bayesian semiparametric regression based on Markov Chain Monte Carlo (MCMC) simulation techniques, handling and manipulation of data sets, and visualizing data. More information about further features available in the program can be found in the manual (Brezger, Kneib and Lang, 2003). The full Bayesian approach is described in detail in this work and in Fahrmeir and Lang (2001*a*), Fahrmeir and Lang (2001*b*), Lang and Brezger (2004) and Brezger and Lang (2003). Details about the estimation techniques for the empirical Bayes approach can be found in Fahrmeir, Kneib and Lang (2003). Survival models are treated in Hennerfeind, Brezger and Fahrmeir (2003) and Fahrmeir and Hennerfeind (2003). Count data regression is covered in Fahrmeir and Osuna Echavarría (2003). *BayesX* is available at `http://www.stat.uni-muenchen.de/~lang/bayesx/bayesx.html`.

8.2 Getting started

After having started *BayesX*, a main window with four sub–windows appears on the screen. In the *command window* we enter and execute commands. A command will be executed by pressing the return key. The *review window* enable easy access to past commands. Click on the desired command and it will appear in the *command window*. There one can modify and/or execute it. The *object browser* displays all objects currently avail-

able. This window has two sub–windows. In the left one the different object types supported by *BayesX* are shown. By selecting one type in this left window with a mouse click, a list of the available objects of this type is displayed. In the *output window* commands and results are displayed. It may be desirable to save the *output window* contents in a a file. To do this we open a so called *log–file*.

```
> logopen using d:\patent\results\patent.log
```

After opening a *log–file*, all commands entered and all program output appearing on the screen will be saved in this file. If the file already exists, *BayesX* will be append the new contents to those in the old file. If we want to replace the old file, then we have to add the option `replace` as follows

```
> logopen, replace using d:\patent\results\patent.log
```

Having finished the estimation we may close the *log–file* by typing `logclose`. Note, that the *log–file* is closed automatically when exiting *BayesX*. If a *log–file* was not opened at the beginning of the session but we are interested in storing the contents of the *output window*, we will always be asked by leaving out *BayesX* to save the output.

BayesX is object oriented although the concept is limited, i.e. inheritance and other concepts of object oriented languages like C++ or S–plus are not supported. For every object type a number of object-specific methods may be applied to a particular object. For estimating Bayesian regression models we need at least a *dataset object* to incorporate, handle and manipulate data, a *bayesreg object* to estimate semiparametric regression models, and a *graph object* to visualize part of the results with some plots. The syntax for generating a new object in *BayesX* is

```
> objecttype objectname
```

where *objecttype* is the type of the object, e.g. `dataset`, and *objectname* is the name to be given to the new object.

8.3 Dataset object

First we create a *dataset object*. This is done by typing

```
> dataset patent
```

in the *command window*, where `patent` is the name of the *dataset object*. Several methods are available for *dataset objects*.

We first read the data using method `infile`. It allows us to read the data from the file *patent.raw* into `patent`, for example with

```
> patent.infile using d:\patent\data\patent.raw
```

This command supposes that the variable names are given in the first row of the file, as is the case in *patent.raw*. Otherwise we would have to write the names of the variables right after `infile`.

If our data set has more than 10.000 observations it is recommended to set the option `maxobs` to the number of rows in our data. This option allows *BayesX* to allocate enough memory to store all the data.

We can take a look at our data by executing `describe`.

```
> patent.describe
```

Note that the variable *cntry_ch_de_gb* given in Table 7.1 does not exist in the read data, but only three dummies *cntry_ch*, *cntry_de* and *cntry_gb*. Using method `generate` we can create this variable and add it to the *dataset object* `patent`. The command for this operation is

```
> patent.generate cntry_ch_de_gb = cntry_ch + cntry_de + cntry_gb
```

We can examine any continuous covariate from our data with the help of `descriptive`.

```
> patent.descriptive claims

   Variable   Obs       Mean   Median        Std  Min   Max
     claims  4805  12.326535       10   8.1304757   1    50
```

Or may be we want to obtain the frequency table of a dummy covariate. In this case method `tabulate` is appropriate.

```
> patent.tabulate pct
Variable: pct

          Value       Obs          Freq           Cum
              0      3900        0.8117        0.8117
              1       905        0.1883             1
```

In Section 7.1 we run a preliminary model and discovered that it is better to drop the observations in our data corresponding to $pct = 1$. Following output

```
> patent.tabulate pct if forwcits!=0

Variable: pct

          Value       Obs          Freq           Cum
              0      2559        0.9865        0.9865
              1        35       0.01349             1
```

confirms this decision. Dropping these observations is an easy matter in *BayesX*.

```
> patent.drop if pct = 1
```

Because we do not need the variables *cntry_ch*, *cntry_de*, and *cntry_gb* any more, we may delete them from the data.

```
> patent.drop cntry_ch cntry_de cntry_gb
```

Note that method `drop` allows to delete variables as well as observations from our data set.

We can save the modified data set in the file *patent.dat* using `outfile`.

```
> patent.outfile, replace header using d:\patent\data\patent.dat
```

8.4 Bayesreg object

We want to estimate three regression models for the patent data, a Poisson, a POGA and a ZIPGA regression. To fit these models we need to fix some regression specific

elements and some options for the MCMC estimating algorithm. The only difference between the specifications of these models is the determination of the distribution family for the response variable. The rest (predictor, priors for the predictor terms and MCMC options) are similar. For the PO model the distributional assumption for the response is the classical $y_i|\ldots \sim Po(\mu_i)$. The POGA model is characterized by the assumption $y_i|\ldots \sim Po(\nu_i\,\mu_i)$, where $\nu_i|\delta \sim G(\delta,\delta)$ are independent individual specific random effects. The ZIPGA model will be represented by $y_i|\ldots \sim ZIPGA(\mu_i,\nu_i,\theta)$, with μ_i being the mean, ν_i the multiplicative random effects with the same prior assumption as for the POGA model, and θ the zero inflation parameter.

To estimate the regression models we have to create three *bayesreg objects* which we name `po, poga` and `zipga`:

```
> bayesreg po
> bayesreg poga
> bayesreg zipga
```

By default estimation results are written to the subdirectory `output` of the installation directory. In this case the default filenames are composed of the name of the *bayesreg object* and the type of the specific file. However, it is usually more convenient to store the results in a user–specified directory. To define this directory we use method `outfile` for *bayesreg objects*:

```
> po.outfile = d:\patent\results\po\po
> poga.outfile = d:\patent\results\poga\poga
> zipga.outfile = d:\patent\results\zipga\zipga
```

Note, that `outfile` does not only specify a directory but also a base filename (the characters 'po', 'poga' and 'zipga' in our example) and that it may of course be different from the name of the bayesreg object. Therefore executing the second command above leads to storage of the results in the directory `'d:\papent\results\poga\'` and all filenames start with the characters 'poga'.

Now we estimate our semiparametric regression models using the `regress` method. We list below the commands for each model.

```
> po.regress forwcits = biopharm + accexam + accsrch + cntry_us + opp
+ cntry_ch_de_gb + ustwin + claims(psplinerw2)
+ gryear(psplinerw2, nrknots=14) + nstat(psplinerw2, nrknots=14),
iterations=5000 step=4 burnin=1000 family=poisson predict using patent


> poga.regress forwcits = biopharm + accexam + accsrch + cntry_us + opp
+ cntry_ch_de_gb + ustwin + claims(psplinerw2)
+ gryear(psplinerw2, nrknots=14) + nstat(psplinerw2, nrknots=14),
iterations=5000 step=4 burnin=1000 family=nbinomial distopt=poga
predict using patent

> zipga.regress forwcits =biopharm + accexam + accsrch + cntry_us + opp
+ cntry_ch_de_gb + ustwin + claims(psplinerw2)
+ gryear(psplinerw2, nrknots=14) + nstat(psplinerw2, nrknots=14),
iterations=5000 step=4 burnin=1000 family=zip zipdistopt=zipga
predict using patent
```

These models are explained at the end of Subsection 7.1.1. The syntax for all three models is similar. The difference relies only on the `family` option, as we will explain bellow.

The first lines of this commands define the response variable together with the predictor and the priors for each term. The semiparametric predictor is common for all three models and is given by:

$$\begin{aligned}\eta &= \gamma_0 + f_1(claims) + f_2(gryear) + f_3(nstat) \\ &\quad + \gamma_1\, biopharm + \gamma_2\, accexam + \gamma_3\, accsrch + \gamma_4\, cntry\_us \\ &\quad + \gamma_5\, opp + \gamma_6\, cntry\_ch\_de\_gb + \gamma_7\, ustwin\end{aligned}$$

The three continuous covariates of Table 7.1 are assumed to have a possibly nonlinear effect on the response variable (*forwcits*) and are therefore modeled by P–splines (with second order random walk prior). With the option `nrknots=14` the number of knots is set to 14 for *gryear* and *nstat*. For *claims* the P–spline has 20 knots, which is the default value. The reasons for this choice are given in Subsection 7.1.1. The remaining variables are dummies and modeled as linear effects.

The options `iterations`, `burnin` and `step` define properties of the MCMC-algorithm that is used to estimate the model. The total number of MCMC iterations is given by

`iterations` while the number of burnin iterations is given by `burnin`. Therefore we obtain a sample of 4000 random numbers with the above specifications of these options. Since, in general, these random numbers are correlated we do not use all of them but thin out the Markov chain by the thinning parameter `step`. Specifying `step=4` as above forces *BayesX* to store only every 4th sampled parameter which leads to a random sample of length 1000 for every parameter in our example. If the option `predict` is specified, samples of the deviance, the effective number of parameters p_D, and the deviance information criteria DIC of the model are computed, see Spiegelhalter, Best, Carlin and van der Linde (2002). In addition, estimates for the predictor and the expectation of every observation are obtained.

The option `family` specifies the distribution family for the response variable. There are several possibilities implemented in *BayesX* for this option and we refer to the manual (Brezger et al., 2003) for more details. As we are interested in analizing count data, three families are of main interest here: the Poisson, the overdispersion and the zero inflation families. For a Poisson regression model with loglink we have to set `family=poisson`, as we did in the first command. Other link functions than the loglink are not supported by *BayesX*.

For a regression model with overdispersion we set the family for the response distribution to `family = nbinomial`. The link function is also here the logarithm, for the same reasons as for the Poisson distribution. For this family we have two further options. The option `aresp`, which can be used to set the hyperparameter a of the gamma prior for the scale parameter to the desired value. Note that only positive values are allowed. The default is `aresp = 1`. The second option is `distopt`. It allows to work directly with a negative binomial density (`distopt = nb`) for the NB model or indirectly with a Poisson-Gamma mixture (`distopt = poga`) for the POGA model. Setting `distopt = poig` allows us to work with a Poisson–Inverse Gaussian mixture.

Zero inflated models are also implemented in a family in *BayesX*. To use them we have to set `family = zip`. Within this family we have the option `zipdistopt`, that offers four

alternatives for the distribution of the response variable. The first one assumes a zero inflated Poisson distribution for the response variable and is given by `zipdistopt=zip`. The second alternative allows us to change to a zero inflated negative binomial distribution (`zipdistopt=zinb`). Zero inflation with latent variables is also possible by setting `zipdistopt=zipga`, for a zero inflated Poisson–Gamma formulation, or alternatively `zipdistopt=zipig`, for a zero inflated Poisson–Inverse Gauss.

BayesX calculates the posterior mean and median, the posterior 2.5%, 10%, 90% and 97.5% quantiles, and the corresponding 95% and 80% posterior probabilities of the estimated effects. The nominal levels of the posterior quantiles may be changed by the user using the options `level1` and `level2`. For example specifying `level1=99` and `level2=70` in the option list of the `regress` command leads to the computation of 0.5%, 15%, 85% and 99.5% quantiles of the posterior. The defaults are `level1=95` and `level2=80`.

8.5 Post estimation commands and results

After estimation, results for each effect are written to an external ASCII file, together with the information written in the *output window*. These files contain the posterior mean and median, and the indicated posterior quantiles. In addition to the files for the different effects two files with endings `.tex` and `.ps` are created and stored in the outfile directory. The *tex file* contains a summary of the estimation results which may be compiled using LaTeX, the *ps file* contains figures of the nonparametric effects.

For the POGA model four more files are additionally created, when comparing results with the PO model. Two of them contain the estimation results for the multiplicative random effects and the sampling paths of ten of them, if the number of observations is larger than 500 (as is the case here) and of all them otherwise. The other two store estimation results and sampling path for the scale parameter. Compared to the POGA model, we find two additional files in the results for the ZIPGA model, which store the summarized results and the sampling paths for the zero inflated parameter.

To save memory, the sampling paths of the other estimated parameters are only stored temporarily by default. If we want to store them, we have to execute the `getsample` command

```
> po.getsample
> poga.getsample
> zipga.getsample
```

which stores the sampled parameters in ASCII files. To avoid too large files, the samples are typically partitioned into several files. It is always recommended to take a look at some of the sampling paths of the parameters to ensure convergence is achieved. With the method `autocor` *BayesX* calculates and stores autocorrelation functions of all sampled parameters in a file named *autocor.raw*.

```
> po.autocor
> poga.autocor
> zipga.autocor
```

As we will see in the next section, this file is also created when plotting autocorrelations with the command `plotautocor`.

8.6 Plots and Graph objects

BayesX provides three possibilities to visualize estimation results:

- As mentioned in the previous section, certain results are automatically visualized by *BayesX* and stored in *ps files*.
- Post estimation plots of *bayesreg objects* allow to visualize results after having executed a `regress` command.
- *Graph objects* may be used to produce graphics using the ASCII files containing the estimation results. In principle *graph objects* allow the visualization of any content

of a *dataset object*. *Graph files* are also used in the batch file containing the commands to reproduce the automatically generated graphics.

In Subsection 8.6.1 we explain how to create and modify post estimation plots in *BayesX*. In Subsection 8.6.2 some comments about *graph objects* are given. We refer the reader to the Manual (Brezger et al., 2003) for more information.

8.6.1 Post estimation plots

After having executed a `regress` command simple plots for nonparametric effects can be produced. Through executing the commands

```
> poga.plotnonp 1
> poga.plotnonp 3
> poga.plotnonp 5
```

we obtain plots for the covariates *claims*, *gryear* and *nstat* respectively. The graphs produced for this commands appears in an *Object-Viewer window* and are shown in Figure 8.1. By default these plots contain the posterior mean and pointwise credible intervals according to the levels specified in the `regress` command. So by default the plot includes pointwise 80% and 95% credible intervals.

BayesX enables the user to customize this basic graph style. For example, we may only want to plot one of the confidence intervals. The options `levels=1` or `levels=2` produce plots with the 95% or the 80% confidence intervals respectively.

Sometimes it may be convenient to give a title to the graph or to indicate what is plotted in the x- or y-axis. With the options `title`, `xlab` and `ylab` is this matter is easy to solve. Following options can be used to modify axis labels and tick marks. `xlimtop` gives the upper and `xlimbottom` the lower limit for the x-axis in the graph. `xstep` gives the distance between tick marks in the x-axis. Of course `ylimtop`, `ylimbottom` and `ystep` are equivalent expressions for the y-axis.

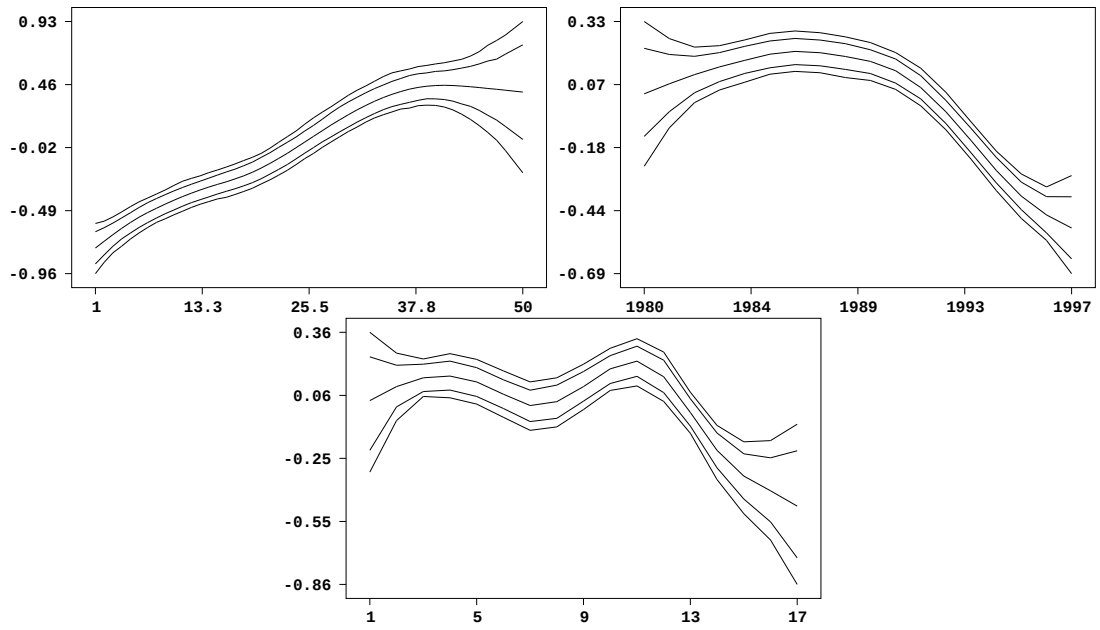

Figure 8.1: Effect of the number of EPO claims, grant year and number of designated states together with pointwise 80% and 95% credible intervals for the model POGA.

If we want to store a plot we may either do this by using the dialog that appears on closing the *Object-Viewer window* or by using the `outfile` option. Again specifying `replace` allows *BayesX* to overwrite an existing file. Note, that if the option `outfile` is specified, *BayesX* does not display the graph on the screen.

The usage of this options is illustrated in Figure 8.2, for which we have executed following command. The plot is stored in the file `d:\patent\results\po\po_nstat.ps`.

```
> po.plotnonp 5, levels=2 xstep=2 ylimtop=0.6 ystep=0.2 ylimbottom=-1.0
  xlab="nstat" ylab="f_nstat" title="Poisson:Nr. of designated States"
  replace outfile=d:\patent\results\po\po_nstat.ps
```

Another method for *bayesreg objects* is the function `plotautocor`. It computes and displays the autocorrelation functions for all estimated parameters with `maxlag` specifying

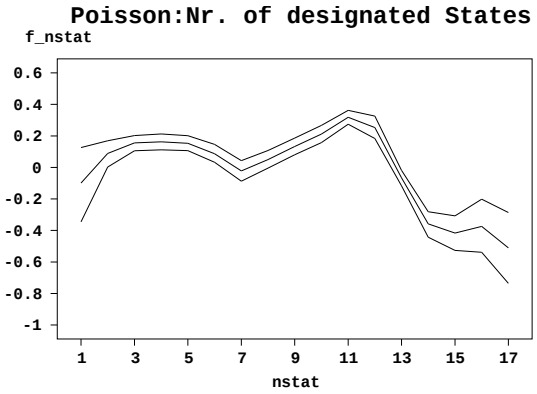

Figure 8.2: Effect of number of designated states together with pointwise 80% credible intervals for the PO model.

the maximum lag number.

```
> po.plotautocor, maxlag=250
> nb.plotautocor, maxlag=250
> poga.plotautocor, maxlag=250
```

Figure 8.3 shows the autocorrelation plot corresponding to the scale parameter in the POGA model.

Note, that executing the `plotautocor` command also stores the computed autocorrelation functions in a file named `autocor.raw` in the output directory of the *bayesreg object*. From this plot we can see that the autocorrelations for the scale parameter are not so good. In such a case it would be recommended to run the POGA model once again with a larger number of `iterations` and a larger number for `step`.

8.6.2 Graph objects

Graph objects are used to visualize data and estimations results. These objects enable us to create equivalents to the post estimation plots of the last subsection from estimation results of past regression analyzes. We can also visualize sampling paths for parameters,

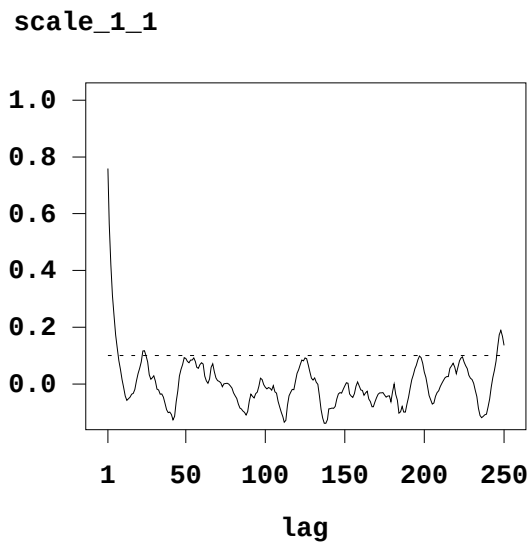

Figure 8.3: Autocorrelation function for the scale parameter in the POGA model

draw maps if geographical information is available or create scatterplots from some given data. In this tutorial we introduce the methods `plotsample` and `plot`.

To create a *graph object* we execute

```
> graph g
```

Now we need a new *dataset object* to store the data we want to plot.

```
> dataset d
```

After having executed the method `getsample` on a *bayesreg object*, `plotsample` can be used to visualize the sampling paths for the parameters. The plots of Figure 8.4 have been created by the following code:

```
> d.infile using d:\patent\results\zipga\zipga_scale_sample.raw
> g.plotsample using d
> d.infile using d:\patent\results\zipga\zipga_theta_sample.raw
> g.plotsample using d
```

Of course we may save these plots in files using the options `outfile` and `replace` as already mentioned in the last subsection. No further options are allowed fot this method.

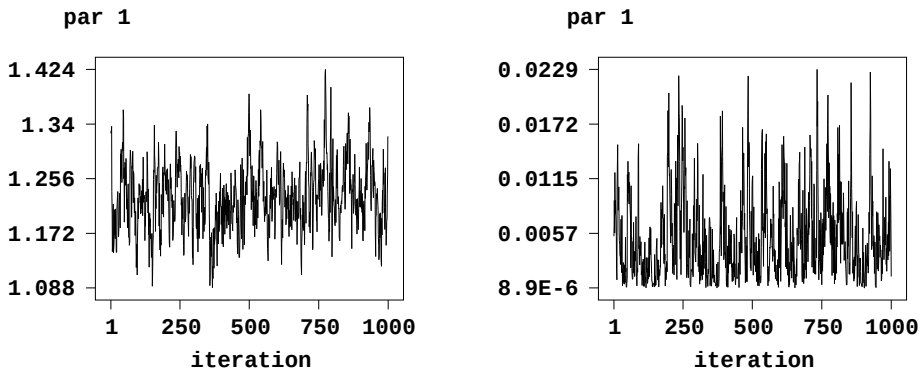

Figure 8.4: Sampling paths for the scale parameter (left) and the zero inflation parameter (right) of the ZIPGA model

Other method that we can apply to a *graph object* is the method `plot`. It is used to draw scatterplots between two or more variables. All options described in Subsection 8.6.1 for the method `plotnonp` are also valid here among others. For example we can use the option `connect` to specify how points in scatterplot are connected. To see the four implemented specifications we refer to the *BayesX* manual. For the plot in Figure 8.5 we used `connect=p`, which means that points are not connected. The commands to obtain this figure are indicated below.

```
> d.infile using d:\patent\results\poga\poga_nu.res
> g.plot nu pmean, outfile=d:\patent\results\poga\nu_scatter.ps
connect=p replace xlab="Index" ylab="posterior mean"
title="Multiplicative random effects" using d
```

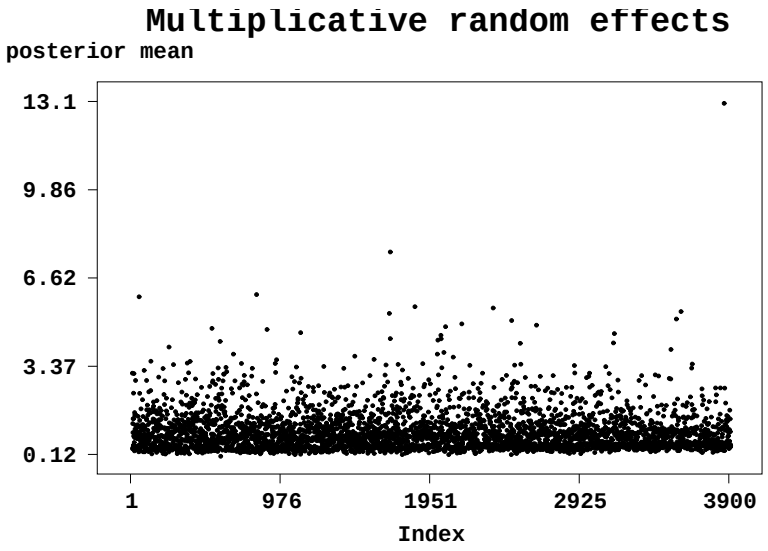

Figure 8.5: Scatterplot for the estimated posterior mean of the multiplicative random effects in the model POGA.

Appendix A

Remarks on distributions

A.1 Derivation of the Negative Binomial distribution

For simplicity, we leave out the subscript in this section to show how the Negative Binomial distribution is derived through integration from the mixed Poisson–Gamma distribution. Consider the variables

$$\begin{aligned} y|\nu,\mu &\sim Po(\nu\,\mu) \qquad \text{and} \\ \nu|\delta &\sim G(\delta,\delta) \end{aligned}$$

with densities

$$P(y|\nu,\mu) = \frac{\exp(-\nu\,\mu)(\nu\,\mu)^y}{y!} \quad \text{for} \quad y \in \boldsymbol{N} \cup \{0\}$$

and

$$g(\nu|\delta) = \frac{\delta^\delta}{\Gamma(\delta)}\nu^{\delta-1}\exp(-\delta\,\nu) \quad \text{for} \quad \nu > 0$$

respectively. Then the dependency of y on ν can be eliminated by taking the expectation of $P(y|\nu,\,\mu)$ over ν as follows

$$P(y|\mu) = E_\nu(P(y|\nu,\,\mu))$$

$$
\begin{aligned}
&= \int P(y|\nu,\mu)g(\nu)d\nu \\
&= \int_0^\infty \frac{\exp(-\nu\,\mu)(\nu\,\mu)^y}{y!}\frac{\delta^\delta}{\Gamma(\delta)}\nu^{\delta-1}\exp(-\delta\,\nu)d\nu \\
&= \frac{\mu^y\,\delta^\delta}{y!\Gamma(\delta)}\int_0^\infty \exp(-(\mu+\delta)\,\nu)\nu^{y+\delta-1}d\nu \\
&= \frac{\mu^y\,\delta^\delta}{\Gamma(y+1)\,\Gamma(\delta)}\frac{\Gamma(y+\delta)}{(\mu+\delta)^{y+\delta}} \\
&= \frac{\Gamma(y+\delta)}{\Gamma(y+1)\,\Gamma(\delta)}\frac{\mu^y\,\delta^\delta}{(\mu+\delta)^{(y+\delta)}} \\
&= \frac{\Gamma(y+\delta)}{\Gamma(y+1)\,\Gamma(\delta)}\left(\frac{\mu}{\mu+\delta}\right)^y\left(\frac{\delta}{\mu+\delta}\right)^\delta \quad \text{for} \quad y\in \mathbb{N}\cup\{0\}
\end{aligned}
$$

A.2 General form of the inverse Gaussian distribution

Is X an inverse Gaussian distributed variable with parameters $\mu > 0$ and $\delta > 0$, $X \sim IGauss(\mu,\delta)$, then its density is of the form

$$
g(x) = \sqrt{\frac{\delta}{2\pi\,x^3}}\exp\left(-\frac{\delta(x-\mu)^2}{2\,x\,\mu^2}\right) \qquad x\in\mathbb{R}^+.
$$

The mean and variance are

$$
\begin{aligned}
E(X) &= \mu \\
V(X) &= \frac{\mu^3}{\delta}.
\end{aligned}
$$

Studying the distribution of $Y = aX$ if $X \sim IGauss(\mu,\delta)$ and $a > 0$:

$$
\begin{aligned}
g(y) &= \sqrt{\frac{\delta}{2\pi\,\left(\frac{y}{a}\right)^3}}\exp\left(-\frac{\delta(\frac{y}{a}-\mu)^2}{2\,\frac{y}{a}\,\mu^2}\right)\frac{1}{a} \\
&= \sqrt{\frac{\delta a^3}{2\pi\,y^3\,a^2}}\exp\left(-\frac{\delta\,a\,(y-a\mu)^2}{2\,y\,a^2\mu^2}\right) \\
&= \sqrt{\frac{\delta a}{2\pi\,y^3}}\exp\left(-\frac{\delta\,a\,(y-a\mu)^2}{2\,y\,(a\mu)^2}\right)
\end{aligned}
$$

It turns out that $Y \sim IGauss(a\mu, a\delta)$.

A.3 General form of the LogNormal distribution

If Y is $N(\mu, \sigma^2)$ distributed, then $X = \exp(Y)$ is a LogNormal distributed variable with parameters $\mu \in \mathbb{R}$ and $\sigma > 0$, denoted $X \sim LogN(\mu, \sigma^2)$. Its density is of the form

$$g(x) = \sqrt{\frac{1}{2\pi\,\sigma^2\,x^2}} \exp\left(-\frac{1}{2\,\sigma^2}\left(\log(x)-\mu\right)^2\right)$$

for $x \in \mathbb{R}^+$. The mean and variance are given by

$$\begin{aligned} E(X) &= \exp\left(\mu + \frac{\sigma^2}{2}\right) \\ V(X) &= \exp\left(2\mu+\sigma^2\right)\left(\exp\left(\sigma^2\right)-1\right). \end{aligned}$$

A.4 Zero inflation with latent variables

We will show that the probability of zero counts for the marginal distribution of a mixture distribution as given by (2.5) exceeds the probability for zero counts in the Poisson distribution. Formally, we proof:

$$P(y_i = 0) < P(y_i = 0|\cdot) = E(P(y_i = 0|\nu_i)|\cdot) \tag{A.1}$$

Define

$$\begin{aligned} f(\nu_i) &= P(y_i|\nu_i) \\ &= \frac{\exp(-\mu_i\nu_i)(\mu_i\nu_i)^{y_i}}{y_i!} \end{aligned}$$

A Taylor series approximation for $f(\nu_i)$ around $\nu_i = 1$ gives

$$f(\nu_i) = f(1) + f'(1)(\nu_i - 1) + \frac{1}{2}f''(\xi)(\nu_i-1)^2 \tag{A.2}$$

with ξ between ν_i and 1. Furthermore it holds

$$\begin{aligned} f'(\nu_i) &= \frac{\exp(-\mu_i\nu_i)(\mu_i\nu_i)^{y_i}}{y_i!}\left(-\mu_i + \frac{y_i}{\nu_i}\right) \\ &= f(\nu_i)\left(-\mu_i + \frac{y_i}{\nu_i}\right) \end{aligned}$$

and

$$\begin{aligned} f''(\nu_i) &= f'(\nu_i)\left(-\mu_i + \frac{y_i}{\nu_i}\right) + f(\nu_i)\left(-\frac{y_i}{\nu_i^2}\right) \\ &= f(\nu_i)\left(\left(-\mu_i + \frac{y_i}{\nu_i}\right)^2 - \left(-\frac{y_i}{\nu_i^2}\right)\right) \end{aligned}$$

Taking expectations on ν_i in (A.2) and considering that $E(\nu_i) = 1$, we obtain

$$\begin{aligned} E(f(\nu_i)) &= f(1) + f'(1)E(\nu_i - 1) + \frac{1}{2}f''(\xi)V(\nu_i) \\ &= f(1) + \frac{1}{2}V(\nu_i)f(\xi)\left(\frac{y_i}{\xi^2} + \left(\frac{y_i}{\xi} - \mu_i\right)^2\right) \end{aligned}$$

For $y_i = 0$ and rewriting f in its original form we have

$$\begin{aligned} P(y_i = 0|\cdot) &= P(y_i = 0|\nu_i = 1) + \frac{1}{2}V(\nu_i)f(\xi)\mu_i^2 \\ &= P(y_i = 0) + \frac{1}{2}V(\nu_i)f(\xi)\mu_i^2 \end{aligned}$$

with $\frac{1}{2}V(\nu_i)f(\xi)\mu_i^2$ always being positive. Hence (A.1) can be asserted.

Appendix B

Calculation of IWLS weights

In this Chapter we are going to analyze the form of the expected Fisher information matrix F^{E} for our models. We have stated in Subsection 5.2.1 that the Fisher Information matrix has the general form $F^{\mathrm{E}} = X'WX$. The aim is to prove this assertion and to calculate the weights matrix W needed for the implementation of the IWLS proposals used in Subsection 5.2.1. For this purpose some notation has to be introduced. Recall the notation for the predictor given in (4.15):

$$\begin{aligned}
\boldsymbol{\eta} &= X_\beta \boldsymbol{\beta} + X_\gamma \boldsymbol{\gamma} + X_\rho \boldsymbol{\rho} + X_f \boldsymbol{f} \\
&= \left(X_\beta, X_\gamma, X_\rho, X_f\right) \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\gamma} \\ \boldsymbol{\rho} \\ \boldsymbol{f} \end{pmatrix} \\
&= X\Psi
\end{aligned}$$

With $K = Q + S + G + R$, X is a nxK matrix and Ψ a Kx1 vector. The observed Fisher information matrix $F^{\mathrm{E}}(\Psi)$ is a KxK matrix defined as

$$F^{\mathrm{E}}(\Psi) = -E\left(\sum_i \frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}\right)_{jk} \tag{B.1}$$

where l_i is the loglikelihood of observation i in our data. We will see from our calculations, that a general form for $F^{\mathrm{E}}(\Psi)$ common for all the models can be found, namely

$$\begin{aligned} F^{\mathrm{E}}(\Psi) &= X'WX \\ &= \begin{pmatrix} X'_\beta W X_\beta & \cdots & \cdots & \cdots \\ \cdots & X'_\gamma W X_\gamma & \cdots & \cdots \\ \cdots & \cdots & X'_\rho W X_\rho & \cdots \\ \cdots & \cdots & \cdots & X'_f W X_f \end{pmatrix} \\ &= \begin{pmatrix} F^{\mathrm{E}}(\beta) & \cdots & \cdots & \cdots \\ \cdots & F^{\mathrm{E}}(\gamma) & \cdots & \cdots \\ \cdots & \cdots & F^{\mathrm{E}}(\rho) & \cdots \\ \cdots & \cdots & \cdots & F^{\mathrm{E}}(f) \end{pmatrix} \end{aligned} \tag{B.2}$$

Note that only the diagonal blocks given in (B.2) will be relevant in our implementation. This form permits a fast implementation of the algorithm. The X matrix is constant along all the iterations. And the W matrix is a nxn diagonal matrix $W = \mathrm{diag}(w_1, \ldots, w_n)$. In an exponential family framework, the form of the weights w_i is well known

$$w_i = \left(b''(\theta_i) \left(g'(\mu_i) \right)^2 \right)^{-1} \tag{B.3}$$

with the usual notation for exponential families: θ natural parameter, $g(\cdot)$ link function, and $b(\cdot)$ depending only on the natural parameter. But here we do not have underlying exponential family distribution in general. So we have to calculate the weights by the direct way: differentiating the loglikelihood twice. As known from Chapter 3, in a zero inflated model we have different forms for the likelihood of a zero response or otherwise. Therefore we have to distinguish between the two possibilities when calculating the weights.

In the next subsections we calculate $E\left(\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}\right)$ for our distributions and obtain

$$-E\left(\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}\right) = x_{ij} x_{ik} w_i \tag{B.4}$$

for all the models. This justifies the form $F^{\mathrm{E}}(\Psi) = X'WX$. Note that when $\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}$ does not depend on y_i, then $E\left(\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}\right) = \frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}$ and we can omit taking expectations. Another conclusion from the results obtained in the following sections is that w_i always depends on μ_i and hence on Ψ. That is the reason why we write $F^{\mathrm{E}}(\Psi)$ and $w_i(\Psi)$.

B.1 NB

For a known scale parameter the Negative Binomial distribution is an exponential family member. Hence two possibilities to calculate the weights exist. The exponential family properties can be exploited or a direct calculation is made. We present both approaches.

B.1.1 Considered as exponential family member

We first rewrite the density of an observation in an exponential family form, supposing that the δ parameter is known.

$$\begin{aligned} p(y_i|\ldots,\delta) &= \frac{\Gamma(y_i+\delta)}{\Gamma(\delta)\Gamma(y_i+1)}\left(\frac{\mu_i}{\delta+\mu_i}\right)^{y_i}\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta} \\ &= \exp\left\{\ln\Gamma(y_i+\delta)-\ln\Gamma(\delta)-\ln\Gamma(y_i+1)\right. \\ &\quad \left.+y_i\ln\left(\frac{\mu_i}{\delta+\mu_i}\right)+\delta\ln\left(\frac{\delta}{\delta+\mu_i}\right)\right\} \\ &= \exp\left\{c(y_i,\delta)+y_i\theta_i-\delta\ln(\delta+\mu_i)\right\} \end{aligned}$$

The obtained link function $g(\cdot)$ and the natural parameter θ_i from the expression below are given by

$$\begin{aligned} \mu_i &= h(\eta_i) = \exp(\eta_i) \\ \eta_i &= g(\mu_i) = \ln(\mu_i) \\ \theta_i &= \ln\left(\frac{\mu_i}{\delta+\mu_i}\right) \\ \mu_i &= \frac{\delta\exp(\theta_i)}{1-\exp(\theta_i)} \end{aligned}$$

From the last equality above we see that the logarithm is not the natural link function of the Negative Binomial distribution. Despite this fact we prefer it because of its simple form and because it ensures that μ_i is positive. Now we calculate the elements of the product given in (B.3), namely $g'(\mu_i)$ and $b''(\theta_i)$:

$$\begin{aligned}
g'(\mu_i) &= \frac{1}{\mu_i} \\
b(\theta_i) &= \delta \ln(\delta + \mu_i) \\
&= \delta \ln\left(\delta + \frac{\delta \exp(\theta_i)}{1 - \exp(\theta_i)}\right) \\
&= \delta \ln(\delta) + \delta \ln\left(\frac{1 - \exp(\theta_i) + \exp(\theta_i)}{1 - \exp(\theta_i)}\right) \\
&= \delta \ln(\delta) + \delta \ln\left(\frac{1}{1 - \exp(\theta_i)}\right) \\
&= \delta \ln(\delta) - \delta \ln\left(1 - \exp(\theta_i)\right) \\
b'(\theta_i) &= -\delta \frac{1}{1 - \exp(\theta_i)}(-\exp(\theta_i)) = \frac{\delta \exp(\theta_i)}{1 - \exp(\theta_i)} \\
b''(\theta_i) &= \delta \frac{\exp(\theta_i)(1 - \exp(\theta_i)) - \exp(\theta_i)(-\exp(\theta_i))}{(1 - \exp(\theta_i))^2} \\
&= \frac{\delta \exp(\theta_i)}{(1 - \exp(\theta_i))^2} \\
&= \delta \frac{\frac{\mu_i}{\delta + \mu_i}}{(1 - \frac{\mu_i}{\delta + \mu_i})^2} \\
&= \frac{\delta \mu_i (\delta + \mu_i)^2}{(\delta + \mu_i)\delta^2} \\
&= \frac{\mu_i(\delta + \mu_i)}{\delta} \\
w_i^{\mathrm{NB}}(\Psi) &= \frac{\delta}{\mu_i(\delta + \mu_i)} \mu_i^2 \\
&= \frac{\delta \mu_i}{\delta + \mu_i}
\end{aligned} \tag{B.5}$$

B.1.2 Direct method

Here the weights for the Negative Binomial are calculated directly, namely differentiating the loglikelihood of an observation twice.

$$\begin{aligned}
l_i &= \log\left(\frac{\Gamma(y_i+\delta)}{\Gamma(y_i+1)\Gamma(\delta)}\left(\frac{\mu_i}{\delta+\mu_i}\right)^{y_i}\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) \\
&= \log\left(\Gamma(y_i+\delta)\right) - \log\left(\Gamma(y_i+1)\right) - \log\left(\Gamma(\delta)\right) \\
&+ y_i\log(\mu_i) + \delta\log(\delta) - (y_i+\delta)\log(\delta+\mu_i) \\
\frac{\partial l_i}{\partial \Psi_j} &= y_i x_{ij} - (y_i+\delta)\frac{\mu_i x_{ij}}{\delta+\mu_i} \\
&= x_{ij}\left(y_i - (y_i+\delta)\frac{\mu_i}{\delta+\mu_i}\right) \\
\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j} &= -x_{ij}(y_i+\delta)\frac{\mu_i x_{ij}(\delta+\mu_i) - \mu_i\mu_i x_{ij}}{(\delta+\mu_i)^2} \\
&= -x_{ij}x_{ik}\mu_i(y_i+\delta)\frac{\delta+\mu_i-\mu_i}{(\delta+\mu_i)^2} \\
&= -x_{ij}x_{ik}\mu_i(y_i+\delta)\frac{\delta}{(\delta+\mu_i)^2}
\end{aligned}$$

Because $\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}$ depends on the response we have to take expectations.

$$\begin{aligned}
\left(F^{\mathrm{E}}(\Psi)\right)_{jk} &= -\sum_{i=1}^{n} E\left(\frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j}\right) \\
&= \sum_{i=1}^{n} x_{ij}x_{ik}\frac{\delta\mu_i}{(\delta+\mu_i)} \\
w_i^{\mathrm{NB}}(\Psi) &= \frac{\delta\mu_i}{\delta+\mu_i}
\end{aligned} \tag{B.6}$$

B.2 Poisson with latent variables

For the POGA and POIG models (shorthand denoted PO* in the formulas below) we could exploit the fact that the response is Poisson distributed with parameter $\nu_i\mu_i$. Results for the Poisson distribution are well known. We have preferred to present few cal-

culations required to obtain the result.

$$\begin{aligned} l_i &= \log\left(\frac{\exp(-\nu_i\mu_i)(\nu_i\mu_i)^{y_i}}{y_i!}\right) \\ &\quad -\nu_i\mu_i + y_i\log(\nu_i) + y_i\log(\mu_i) - \log(y_i!) \\ \frac{\partial l_i}{\partial \Psi_j} &= -\nu_i\mu_i x_{ij} + y_i x_{ij} \\ &= x_{ij}(y_i - \nu_i\mu_i) \\ \frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j} &= -x_{ij}x_{ik}\nu_i\mu_i \\ w_i^{\mathrm{PO}^*}(\Psi) &= \nu_i\mu_i. \end{aligned} \tag{B.7}$$

B.3 ZIP

As explained before, we have to calculate the weights for zero or nonzero observations separately, because of the different likelihood forms. We begin with the weights for the zero observations.

$\boxed{y_i = 0}$

$$\begin{aligned} l_i &= \log\left(\theta + (1-\theta)\exp(-\mu_i)\right) \\ \exp(l_i) &= \theta + (1-\theta)\exp(-\mu_i) \\ \frac{\partial l_i}{\partial \Psi_j} &= -x_{ij}(1-\theta)\frac{\mu_i\exp(-\mu_i)}{\theta + (1-\theta)\exp(-\mu_i)} \\ \frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j} &= -x_{ij}(1-\theta)\frac{1}{\left(\theta + (1-\theta)\exp(-\mu_i)\right)^2} \\ &\quad \left(x_{ik}\mu_i\exp(-\mu_i) - \mu_i\exp(-\mu_i)\mu_i x_{ik}\right) - \mu_i\exp(-\mu_i)\left(-(1-\theta)\exp(-\mu_i)\mu_i x_{ik}\right) \\ &= -x_{ij}x_{ik}(1-\theta)\exp(-\mu_i)\mu_i\frac{(1-\mu_i)\exp(l_i) + (1-\theta)\exp(-\mu_i)\mu_i}{\exp(2l_i)} \\ &= -x_{ij}x_{ik}(1-\theta)\exp(-\mu_i)\mu_i\frac{(1-\mu_i)\exp(l_i) + \mu_i(\exp(l_i) - \theta)}{\exp(2l_i)} \\ &= -x_{ij}x_{ik}(1-\theta)\exp(-\mu_i)\mu_i\frac{\exp(l_i) - \mu_i\theta}{\exp(2l_i)} \\ w_i^{\mathrm{ZIP}}(\Psi) &= (1-\theta)\exp(-\mu_i)\mu_i\frac{\exp(l_i) - \mu_i\theta}{\exp(2l_i)} \end{aligned} \tag{B.8}$$

Now the weights for the nonzero observations. The term $\log(1-\theta)$ disappears with the differentiation, thus the calculations here are actually equivalent with those for a Poisson model.

$\boxed{y_i \neq 0}$

$$\begin{aligned} l_i &= \log\left((1-\theta)\frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!}\right) \\ &= \log(1-\theta) - \mu_i + y_i\log(\mu_i) - \log(y_i!) \\ \frac{\partial l_i}{\partial \Psi_j} &= -\mu_i x_{ij} + y x_{ij} \\ &= x_{ij}(y_i - \mu_i) \\ \frac{\partial^2 l_i}{\partial \Psi_k \partial \Psi_j} &= -x_{ij}x_{ik}\mu_i \\ w_i^{\text{ZIP}}(\Psi) &= \mu_i \end{aligned} \tag{B.9}$$

B.4 ZIP with latent variables

We will denote the weights with $w_i^{\text{ZIP}^*}(\Psi)$ leading to $w_i^{\text{ZIPGA}}(\Psi)$ and $w_i^{\text{ZIPIG}}(\Psi)$ for ZIPGA and ZIPIG respectively. For these models we also have to distinguish between zero or nonzero observations, but we are going to exploit the last results and avoid the calculations. As the ν_i are only a factor of μ_i independent of Ψ, we can take the weights for the ZIP model and complete them by multiplying μ_i by ν_i as follows.

$\boxed{y_i = 0}$

$$\begin{aligned} l_i &= \log\left(\theta + (1-\theta)\exp(-\nu_i\mu_i)\right) \\ w_i^{\text{ZIP}^*}(\Psi) &= (1-\theta)\exp(-\nu_i\mu_i)\nu_i\mu_i\frac{\exp(l_i) - \nu_i\mu_i\theta}{\exp(2l_i)} \end{aligned} \tag{B.10}$$

$\boxed{y_i \neq 0}$

$$\begin{aligned} l_i &= \log\left((1-\theta)\frac{\exp(-\nu_i\mu_i)(\nu_i\mu_i)^{y_i}}{y_i!}\right) \\ w_i^{\text{ZIP}^*}(\Psi) &= \nu_i\mu_i \end{aligned} \tag{B.11}$$

B.5 ZINB

The weights for the ZINB model are obtained following the same scheme as before except that the calculations are a little more complicated. For the zero observations we have:

$\boxed{y_i = 0}$

$$
\begin{aligned}
l_i &= \log\left(\theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) \\
&= \log\left(\theta(\delta+\mu_i)^{\delta} + (1-\theta)\delta^{\delta}\right) - \delta\log(\delta+\mu_i) \\
\exp(l_i) &= \theta + (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta} \\
\frac{\partial l_i}{\partial \Psi_j} &= \frac{\delta\theta\mu_i x_{ij}(\delta+\mu_i)^{\delta-1}}{\theta(\delta+\mu_i)^{\delta} + (1-\theta)\delta^{\delta}} - \frac{\delta\mu_i x_{ij}}{\delta+\mu_i} \\
&= x_{ij}\delta\mu_i \frac{\theta(\delta+\mu_i)^{\delta} - \theta(\delta+\mu_i)^{\delta} - (1-\theta)\delta^{\delta}}{\theta(\delta+\mu_i)^{\delta+1} + (1-\theta)\delta^{\delta}(\delta+\mu_i)} \\
&= -x_{ij}(1-\theta)\delta^{\delta+1}\frac{\mu_i}{\theta(\delta+\mu_i)^{\delta+1} + (1-\theta)\delta^{\delta}(\delta+\mu_i)} \\
\frac{\partial^2 l_i}{\partial\Psi_k\partial\Psi_j} &= -x_{ij}(1-\theta)\delta^{\delta+1}\frac{1}{(\delta+\mu_i)^2\left((\delta+\mu_i)^{\delta}\theta + (1-\theta)\delta^{\delta}\right)^2} \\
&\quad \Big\{\mu_i x_{ik}\left(\theta(\delta+\mu_i)^{\delta+1} + (1-\theta)\delta^{\delta}(\delta+\mu_i)\right) \\
&\quad -\mu_i\left((\delta+1)\theta\mu_i x_{ij}(\delta+\mu_i)^{\delta} + (1-\theta)\delta^{\delta}\mu_i x_{ij}\right)\Big\} \\
&= -x_{ij}x_{ik}(1-\theta)\delta^{\delta+1}\mu_i\frac{1}{(\delta+\mu_i)^2\exp(2l_i)(\delta+\mu_i)^{2\delta}} \\
&\quad \Big\{(\delta+\mu_i)\exp(l_i)(\delta+\mu_i)^{\delta} - \mu_i\left(\theta\delta(\delta+\mu_i)^{\delta} + \theta(\delta+\mu_i)^{\delta} + (1-\theta)\delta^{\delta}\right)\Big\} \\
&= -x_{ij}x_{ik}(1-\theta)\delta^{\delta+1}\mu_i\frac{1}{\exp(2l_i)(\delta+\mu_i)^{2+2\delta}} \\
&\quad (\delta+\mu_i)^{\delta+1}\exp(l_i) - \mu_i\left(\theta\delta(\delta+\mu_i)^{\delta} + \exp(l_i)(\delta+\mu_i)^{\delta}\right) \\
&= -x_{ij}x_{ik}(1-\theta)\delta^{\delta+1}\mu_i\frac{(\delta+\mu_i)\exp(l_i) - \mu_i\exp(l_i) - \mu_i\theta\delta}{\exp(2l_i)(\delta+\mu_i)^{2+\delta}} \\
&= -x_{ij}x_{ik}(1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta+2}\mu_i\frac{\exp(l_i) - \mu_i\theta}{\exp(2l_i)} \\
w_i^{\text{ZINB}}(\Psi) &= (1-\theta)\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta+2}\mu_i\frac{\exp(l_i) - \mu_i\theta}{\exp(2l_i)} \qquad \text{(B.12)}
\end{aligned}
$$

For nonzero observations we can use the results obtained for the NB model. The term $\log(1-\theta)$ does not depend on Ψ and therefore disappears in the first differentiation with respect to Ψ. The rest of the calculations are then similar as for the NB model and we can take the results obtained there.

$\boxed{y_i \neq 0}$

$$\begin{aligned} l_i &= \log\left((1-\theta)\frac{\Gamma(y_i+\delta)}{\Gamma(y_i+1)\Gamma(\delta)}\left(\frac{\mu_i}{\delta+\mu_i}\right)^{y_i}\left(\frac{\delta}{\delta+\mu_i}\right)^{\delta}\right) \\ w_i^{\text{ZINB}}(\Psi) &= \frac{\delta\mu_i}{\delta+\mu_i} \end{aligned} \tag{B.13}$$

Appendix C

MCMC

It is not the aim of this work to present Markov Chain Monte Carlo (MCMC) methods in detail. Therefore only a brief overview is given here. For more detailed information see Casella and George (1992), Chib and Greenberg (1995), Gamerman (1997*b*), Gelman, Carlin, Stern and Rubin (1995), and Spiegelhalter et al. (2002).

A motivation for MCMC theory is the following: In some applications we may have a target distribution with density $\pi(\theta)$ which is numerically intractable. Note that with θ we may refer to a parameter or parameter vector. This is the usual situation for the posterior distribution in Bayesian statistics. This posterior is generally a high dimensional distribution obtained through

$$\begin{aligned} \pi(\theta|y) &= \frac{p(y|\theta)p(\theta)}{p(y)} \\ &= \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta} \\ &\propto p(y|\theta)p(\theta) \end{aligned}$$

The problem arises with the high dimensional integral $\int p(y|\theta)p(\theta)d\theta$, which is not directly calculable in most of the cases. In a usual Bayesian analysis we have a high dimensional posterior distribution known up to a constant from which information about the parameters has to be obtained. MCMC methods allow us to obtain a sample of this

posterior. So the advantage compared to other estimation methods is that we achieve pointwise estimators with some confidence intervals for the parameters in our model and also a whole sample as large as desired from their posterior distribution. When pointwise estimators are needed, empirical equivalents of them can be calculated from the sample with the desired accuracy. Furthermore, quantiles or inclusive the density trough nonparametric estimation techniques can be computed.

In the following two sections we present the main sampling methods: Gibbs–sampling and the Metropolis–Hastings–algorithm. In the last section we given some comments about model selection.

C.1 Gibbs–sampling

Suppose we are given a multidimensional density, say $\pi(\boldsymbol{\theta}) = \pi(\theta_1, \ldots, \theta_p)$. Each of the full conditional distributions of this density, in the following denoted by $\pi_i(\theta_i|\boldsymbol{\theta_{-i}}) = \pi_i(\theta_i|\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_p)$, is of a well known form and can be sampled from. Suppose we are interested in one or more of its marginal distributions, say $\pi_i(\theta_i)$, which are given by

$$\pi_i(\theta_i) \quad = \quad \int \pi(\theta_1, \ldots, \theta_p) d\theta_1, \ldots d\theta_{i-1} d\theta_{i+1} \ldots d\theta_p \tag{C.1}$$

for $i = 1, \ldots, p$ or in π self. These usually high dimensional integrals are very complicated in most of the cases and difficult to solve. Gibbs–sampling (see Casella and George (1992)) allows us to indirectly get a sample from the marginal distribution $\pi_i(\theta_i)$ and thus avoids the calculation of (C.1). The **algorithm** can be resumed as follows

1. Initialize $\boldsymbol{\theta}_{\mathbf{-1}}^{(\mathbf{0})}$ and set $i = 0$ and $j = 0$
2. Set $j = j + 1$
3. Set $i = i + 1$
4. Sample $\theta_i^{(j)} \sim \pi_i(\theta_i|\theta_1^{(j+1)}, \ldots, \theta_{i-1}^{(j+1)}, \theta_{i+1}^{(j)}, \ldots, \theta_p^{(j)})$

5. If $i = p$, set $i = 0$ and go to 2.

   Otherwise go to 3.

The algorithm ends when j arrives at a predetermined value, say J, which gives the desired length of the sample. As the values of the j^{th} iteration only depend on the values of the last iteration, Gibbs–sampling provides a homogeneous Markov chain. For each component θ_i the transition kernels are given in the 4^{th} step of the algorithm and their stationary distributions are the corresponding marginal distributions π_i. The transition kernel of the whole chain is the product of the individual component transition kernels

$$p(\boldsymbol{\theta}^{(j)}, \boldsymbol{\theta}^{(j+1)}) \;=\; \prod_{i=1}^{p} \pi_i(\theta_i^{(j+1)} | \theta_1^{(j+1)}, \ldots, \theta_{i-1}^{(j+1)}, \theta_{i+1}^{(j)}, \ldots, \theta_p^{(j)}),$$

and it has π as its stationary distribution. The first J_b iterations are called **burn in** and are not taken into account for later inference to ensure that convergence is achieved. In the following and in order to avoid notational complications we are going to assume that $\theta_i^{(1)}$ is the first value of the chain for π_i after the burnin phase. A typical output from this algorithm is then given by

$$\begin{array}{ccccc} \theta_1^{(1)} & \ldots & \theta_i^{(1)} & \ldots & \theta_p^{(1)} \\ \vdots & & \vdots & & \vdots \\ \theta_1^{(j)} & \ldots & \theta_i^{(j)} & \ldots & \theta_p^{(j)} \\ \vdots & & \vdots & & \vdots \\ \theta_1^{(J)} & \ldots & \theta_i^{(J)} & \ldots & \theta_p^{(J)}. \end{array}$$

So under convergence the i^{th} column of this matrix represents a sample from $\pi_i(\theta_i)$ for each $i = 1, \ldots, p$, and the j^{th} row of the matrix is a draw from π for $j = 1, \ldots, J$.

C.2 Metropolis–Hastings–sampling

The starting point for the Metropolis–Hastings–algorithm (M–H) is a target density $\pi(\boldsymbol{\theta})$ known up to the normalizing constant from which no direct sampling is possible. In

a more general framework as Gibbs–sampling, not all the full conditionals $\pi_i(\theta_i|\boldsymbol{\theta}_{-i}) = \pi_i(\theta_i|\theta_1,\ldots,\theta_{i-1},\theta_{i+1},\ldots,\theta_p)$ have to be completely known or can be sampled from. The M–H–algorithm describes how to iteratively obtain a sample from $\pi(\boldsymbol{\theta})$ by generating a Markov chain $\boldsymbol{\theta}^{(0)},\ldots,\boldsymbol{\theta}^{(j)},\ldots$ whose stationary distribution coincides with the target distribution π. Just as every Markov chain it can be characterized through its transition kernel $p(\boldsymbol{\theta},\boldsymbol{\theta}^*)$. First we are going to describe how to appropriate build the transition kernel of this chain and then justify this choice.

C.2.1 Construction of the transition kernel

Two distributions are necessary to construct the kernel. These are a so called *proposal distribution* $q(\boldsymbol{\theta}\rightarrow\boldsymbol{\theta}^*)$ and an *acceptance probability* $\alpha(\boldsymbol{\theta},\boldsymbol{\theta}^*)$. By each step of the algorithm a $\boldsymbol{\theta}^*$ value is drawn from the proposal distribution. This $\boldsymbol{\theta}^*$ can be seen as a sort of *candidate* to the next stage of the chain. For the moment, no restrictions are made for the choice of the proposal distribution, but some comments about it will be given bellow in subsections C.2.2 and C.2.4. Whether this candidate becomes a next stage in the chain or not is stored through the acceptance probability defined by

$$\alpha(\boldsymbol{\theta},\boldsymbol{\theta}^*) \quad = \quad \min\left\{1, \frac{\pi(\boldsymbol{\theta}^*)\, q(\boldsymbol{\theta}^* \rightarrow \boldsymbol{\theta})}{\pi(\boldsymbol{\theta})\, q(\boldsymbol{\theta} \rightarrow \boldsymbol{\theta}^*)}\right\} \tag{C.2}$$

Note that the normalizing constant for $\pi(\boldsymbol{\theta})$ is not needed because it only appears in a quotient. This fact is very important since this normalizing constant is often unknown for the posterior distributions in Bayesian analysis as mentioned at the beginning of this appendix. So the algorithm works quite easy: just take a value for $\boldsymbol{\theta}^*$ from q and accept it with probability α as the new stage of the chain.

Algorithm

1. Initialize $\boldsymbol{\theta}^{(0)}$ and set $j = 0$
2. Set $j = j + 1$
3. Sample $\boldsymbol{\theta}\sim q(\boldsymbol{\theta}^{(j)} \rightarrow \boldsymbol{\theta}^*)$

4. Accept $\boldsymbol{\theta}^{(j+1)}=\boldsymbol{\theta}^*$ with probability $\alpha(\boldsymbol{\theta}^{(j)},\boldsymbol{\theta}^*)$,
   otherwise let $\boldsymbol{\theta}^{(j+1)}=\boldsymbol{\theta}^{(j)}$

The algorithm stops when the chain has the predetermined length J. Following the same notation as in Section C.1, the first J_b iterations are the so called **burn in** and are not used for further analyses to ensure convergence is achieved. So the output of the M–H–algorithm is some vector $\boldsymbol{\theta}^{(0)},\ldots,\boldsymbol{\theta}^{(j)},\ldots,\boldsymbol{\theta}^{(J)}$, whose components can be interpreted as drawn from π and therefore having the same dimension.

Now lets take a look on the transition kernel

$$\begin{aligned} p(\boldsymbol{\theta}, A) &= \int_A q(\boldsymbol{\theta} \to \boldsymbol{x})\alpha(\boldsymbol{\theta}, \boldsymbol{x})d\boldsymbol{x} \\ &+ I_A(\boldsymbol{\theta}) \left[1 - \int q(\boldsymbol{\theta} \to \boldsymbol{x})\alpha(\boldsymbol{\theta}, \boldsymbol{x})d\boldsymbol{x}\right] \end{aligned} \tag{C.3}$$

where A is a subset of the parameter space. If the proposal is a discrete distribution, then we have sums instead of integrals. The first line in (C.3) is the probability that the chain moves from $\boldsymbol{\theta}$ to any point in A. The second line gives the probability that the chain remains in $\boldsymbol{\theta}$ if it is a point in A. So $p(\boldsymbol{\theta}, A)$ is the probability to get from $\boldsymbol{\theta}$ to A.

C.2.2 Justification of the transition kernel

It is now time to explain, why this transition kernel in (C.3) has exactly the target distribution $\pi(\boldsymbol{\theta})$ as stationary distribution. From the general Markov chain theory it is known that an irreducible and aperiodic chain has a stationary distribution. And if in addition the reversibility condition holds

$$\pi(\boldsymbol{\theta})\, p(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = \pi(\boldsymbol{\theta}^*)\, p(\boldsymbol{\theta}^*, \boldsymbol{\theta}) \tag{C.4}$$

for every $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ from the support of π, then π is the stationary distribution of the chain. Irreducible means that one can get from every $\boldsymbol{\theta}$ to every $\boldsymbol{\theta}^*$ in the support of the chain in a finite number of steps. And aperiodicity says that the number of moves to get from

θ to θ^* are not required to be a multiple of some integer. Both properties are ensured if the support of the proposal distribution covers or is at least equal to the support of the target distribution. They hold also for uniform proposals with center in the current point and finite window width. Hence irreducibility and aperiodicity are guaranteed through an appropriate choice of the proposal distribution.

To demonstrate that the reversibility condition given in (C.4) holds, note first that the transition kernel defined in (C.3) can also be written as

$$p(\theta, \theta^*) = \begin{cases} q(\theta \to \theta^*)\alpha(\theta, \theta^*) & \text{if} \quad \theta \neq \theta^* \\ 1 - \int q(\theta \to \theta^*)\alpha(\theta, \theta^*)d\theta^* & \text{if} \quad \theta = \theta^* \end{cases}$$

In this form it is easily shown that the reversibility condition holds. For $\theta = \theta^*$ it is obvious and for the case $\theta \neq \theta^*$ just very few lines are needed:

$$\begin{aligned} \pi(\theta)\, p(\theta, \theta^*) &= \pi(\theta)\, q(\theta \to \theta^*)\, \alpha(\theta, \theta^*) \\ &= \min\left\{1, \frac{\pi(\theta^*)\, q(\theta^* \to \theta)}{\pi(\theta)\, q(\theta \to \theta^*)}\right\} \pi(\theta)\, q(\theta \to \theta^*) \\ &= \min\left\{\pi(\theta)\, q(\theta \to \theta^*), \pi(\theta^*)\, q(\theta^* \to \theta)\right\} \\ &= \min\left\{1, \frac{\pi(\theta)\, q(\theta \to \theta^*)}{\pi(\theta^*)\, q(\theta^* \to \theta)}\right\} \pi(\theta^*)\, q(\theta^* \to \theta) \\ &= \pi(\theta^*)\, q(\theta^* \to \theta)\, \alpha(\theta^*, \theta) \\ &= \pi(\theta^*)\, p(\theta^*, \theta) \end{aligned}$$

The choice of this special form for the acceptance probability is hence justified. We have thus shown that the transition kernel $p(\theta, \theta^*)$ has an invariant distribution and this distribution is $\pi(\theta)$.

C.2.3 Some remarks

Convergence behavior

Although it is theoretically proved that the iterations of the transition kernel converge to the target distribution, convergence behavior must be controlled for each particular

analysis. This should be done at least with the help of two tools: the *acceptance rate* and some graphical analysis. The acceptance rate is defined as the proportion of accepted changes of stage in the chain. A high acceptance rate means that too many of the proposed values are accepted. Or equivalently, that the proposed values are very close to the current ones. So the support of the target density will be covered very slowly because of the small steps, and hence the chain will have a poor mixing behavior. On the other hand, if the acceptance rate is too low, then the chain does not change often enough the states because the proposed values may fall in low probability zones of the support of the target distribution, far away from the current value. As a consequence slow convergence and poor mixing are achieved. After these considerations it is clear that the variance of the proposal distribution plays an important role in controlling this acceptance rate. Hence it can be considered as a sort of tuning parameter. For an optimal mixing the acceptance rate should be as a rule of thumb between 30% and 60%. A complement to the acceptance rate, a graphical monitoring is always strongly recommended. Some plots of the sampling paths and analysis of the autocorrelation functions will provide evidence whether the chain shows good convergence behavior or not. If the autocorrelations are too high then some *lag* should be introduced, that means, only the value of each iteration multiple of l after the burn in phase will be taken as a stage in the chain. The question how many iterations should be calculated cannot be answered.

Block move and hybrid algorithms

Suppose we are given the target distribution $\pi(\boldsymbol{\theta})$ as explained in the beginning of this section C.2 and in the easiest case that we can divide the parameter vector $\boldsymbol{\theta}$ in two components $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$. Suppose also that for a fixed $\boldsymbol{\theta_2}$ there exists a transition kernel $p_1(\boldsymbol{\theta_1}, \boldsymbol{\theta_1^*}|\boldsymbol{\theta_2})$ which has as invariant distribution $\pi(\boldsymbol{\theta_1}|\boldsymbol{\theta_2})$. Analogous the same holds for $\boldsymbol{\theta_2}$ for fixed $\boldsymbol{\theta_1}$ and a transition kernel p_2. Under these conditions, the product of the transition kernels $p_1(\boldsymbol{\theta_1}, \boldsymbol{\theta_1^*}|\boldsymbol{\theta_2})$ and $p_2(\boldsymbol{\theta_2}, \boldsymbol{\theta_2^*}|\boldsymbol{\theta_1})$ has $\pi(\boldsymbol{\theta_1}, \boldsymbol{\theta_2})$ as invariant distribution. For the proof of this result see for example Chib and Greenberg (1995) or Gamerman (1997*b*). The practical advantage of this situation is that we can divide $\boldsymbol{\theta}$ into blocks,

say $(\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_D})$ where the block size do not need to be the same for every $\boldsymbol{\theta_d}$, and alternatively run a M–H–step over all components in each iteration. The order in which we run the components may be random or fixed in each iteration. Note that the block size can also be 1; this would mean that some single parameters are updated alone. The acceptance probability for the d^{th} block in a general case is now given by

$$\alpha_d(\boldsymbol{\theta_d}, \boldsymbol{\theta_d^*}) \quad = \quad \min\left\{1, \frac{\pi(\boldsymbol{\theta_d^*}|\boldsymbol{\theta_{-d}})\, q(\boldsymbol{\theta_d^*} \rightarrow \boldsymbol{\theta_d})}{\pi(\boldsymbol{\theta_d}|\boldsymbol{\theta_{-d}})\, q(\boldsymbol{\theta_d} \rightarrow \boldsymbol{\theta_d^*})}\right\} \tag{C.5}$$

To better visualize how the **blockwise algorithm** works it is written down for the simple case of a fixed order for the blocks.

1. Initialize $\boldsymbol{\theta^{(0)}}$ and set $d = 0$ and $j = 0$
2. Set $j = j + 1$
3. Set $d = d + 1$
4. Sample $\boldsymbol{\theta_d} \sim q(\boldsymbol{\theta_d^{(j)}} \rightarrow \boldsymbol{\theta_d^*})$
5. Accept $\boldsymbol{\theta_d^{(j+1)}} = \boldsymbol{\theta_d^*}$ with probability $\alpha(\boldsymbol{\theta_d^{(j)}}, \boldsymbol{\theta_d^*})$,
   otherwise let $\boldsymbol{\theta_d^{(j+1)}} = \boldsymbol{\theta_d^{(j)}}$
6. If $d = D$ set $d = 0$ and go to 2
   otherwise go to 3.

The same comments about length of the chain or lag for the samples are valid here. A consequence of this result is that Gibbs–sampling can be seen as a special case of the M–H–algorithm. If $\pi(\boldsymbol{\theta})$ is a target distribution appropriate for Gibbs–sampling, we choose the distributions given by $q(\theta_i \rightarrow \theta_i^*) = \pi_i(\theta_i^*|\boldsymbol{\theta_{-i}})$ as proposals for the components of $\boldsymbol{\theta}$. By putting these proposals in (C.5) it is clear that the α_i are always 1, that is all the proposed candidates are accepted. As a further but very important extension we remark that for all blocks different sampling methods or different proposals may be used, so called **hybrid algorithms**. This makes the whole implementation of the algorithm more efficient because the best possibility in fit and in velocity of convergence can be taken for each block.

C.2.4 Proposals

The choice of the proposal distribution is arbitrary up to certain mild restrictions. Some general characteristics for a distribution to be an appropriate proposal are somehow intuitive and not very restrictive. First it must have an easy form to sample from. The support of the target distribution should be covered by the support of the proposal, although in general an uniform distribution with finite window width and center on the current value also works good. Of advantage is also that the tails of the proposal dominate the tails of the target distribution to ensure that every point of the support of the target distribution is visited often enough. Whether these points are accepted or not is of course controlled through the acceptance probability. Finally, the variance of the proposal is a tuning parameter controlling the acceptance rate of the chain. There are some common possibilities for these proposals, and some of them are presented below. For the notation, the subindexes are omitted and the θ can be parameter vectors, blocks of any size or just one parameter.

Independent proposal: The proposal distribution just gives a value for θ^* independent of the current value θ, that means $q(\theta \to \theta^*) = q(\theta^*)$. In this case the probability to accept θ^* as the next value in the chain is given by

$$\alpha(\theta, \theta^*) \quad = \quad \min\left\{1, \frac{\pi(\theta^*)\, q(\theta)}{\pi(\theta)\, q(\theta^*)}\right\}.$$

Random Walk proposal: The proposed value θ^* is sampled from a normal distribution with mean θ and variance p. In this case, the proposal is proportional to $(\theta - \theta^*)^2$ for both $q(\theta \to \theta^*)$ and $q(\theta^* \to \theta)$. In other words, the proposal density is symmetric with respect to θ and θ^*. Therefore the $\alpha(\theta, \theta^*)$–quotient simplifies as follows

$$\alpha(\theta, \theta^*) \quad = \quad \min\left\{1, \frac{\pi(\theta^*)}{\pi(\theta)}\right\}.$$

Conditional prior proposal: Suppose that θ represents a block of parameters with prior given by $\pi(\theta)$. In general, the prior will have a Gaussian form, so that $\pi(\theta_i|\theta_{-i})$

are again Gaussian distributed for $i = 1, \ldots, p$. The idea of using conditional prior proposals is to sample θ_i^* from $q(\theta_i \rightarrow \theta_i^*) = \pi(\theta_i|\theta_{-i})$ (Knorr-Held, 1999; Knorr-Held, 1997). The main advantage is celerity of computations, due to the sampling from a Gaussian distribution and the form of the acceptance rate that simplifies to

$$\alpha(\theta_i, \theta_i^*) \quad = \quad \min\left\{1, \frac{p(y|\theta_i^*)}{p(y|\theta_i)}\right\}.$$

IWLS proposal: The idea in Gamerman (1997*a*) is to take into account the prior information, as well as the likelihood structure of the model. The proposal distribution is then a single iterative step by combining the prior and the iterativly weighted least squares of the common likelihood approaches. The algorithm has the advantage that the variance of the proposal is given by the covariance matrix of the IWLS algorithm and the one of the prior distribution, and thus it has not to be tuned by the user.

C.3 Model comparison

The Deviance Information Criterion (DIC) is a common choice for comparison of complex hierarchical Bayesian models. It was developed by Spiegelhalter et al. (2002). For notational simplicity, we denote the set of parameters in the model with $M = \{\mu, \theta\}$, where μ is the mean and θ the rest of parameters in the model. The DIC is defined as

$$\text{DIC} \quad = \quad \overline{D(M)} + p_D.$$

There, $D(M)$ is the deviance of the model and is given by $D(M) = -2\log L(y|M)$ in the unstandardized case or by $D(M) = -2\log L(y|M) + 2\log L(y|\mu = y, \theta)$ for the saturated case. The deviance $D(M)$ is a function of the parameters in the model and can be calculated in each iteration step. Hence, $\overline{D(M)}$ is the posterior mean of the stored deviance samples. The term $p_D = \overline{D(M)} - D(\overline{M})$ is the effective number of parameters in the model and can be interpreted as a sort of complexity measure of the model. Note that $D(\overline{M})$ is the deviance function applied on the posterior estimates of the parameters.

In their work Spiegelhalter et al. (2002) analyzes the theoretical properties of the DIC criterion only on exponential families. In our work, we are far away from an exponential family frame work. Nevertheless both versions of the DIC are standard calculated for each model. But after preliminary analysis we detect an important instability of the calculated DICs (even with negative values for the estimated p_D) and we cannot use this criterion to make assertion about model selection or comparison.

Bibliography

Agarwal, D. K., Gelfand, A. E. and Citron-Pousty, S. (2002), 'Zero–inflated models with application to spatial count data', *Environmental and Ecological Statistics* **9**, 341–355.

Aitkin, M. (1996), 'A general maximum likelihood analysis of overdispersion in generalized linear models', *Statistics and Computing* **6**, 251–262.

Alexander, N., Moyeed, R. and Stander, J. (2000), 'Spatial modelling of individual–level parasite counts using the negative binomial distribution', *Biostatistics* **1**(4), 453–463.

Biller, C. (2000), Bayesianische Ansätze zur nonparametrischen Regression, PhD thesis, Ludwig Maximilian Universität, München.

Booth, J., Casella, G., Friedl, H. and Hobert, J. (2003), 'Negative binomial loglinear mixed models', *Statistical Modelling* **3**, 179–191.

Boskov, M. and Verrall, R. (1994), 'Premium rating by geographical area using spatial models', *ASTIN Bulletin* **24**, 131–143.

Brezger, A., Kneib, T. and Lang, S. (2003), 'BayesX manual'.

Brezger, A. and Lang, S. (2003), 'Generalized structured additive regression based on Bayesian P–splines', *SFB 386 Discussion Paper, Department of Statistics, University of Munich* **DP 321**.

Brockman, M. and Wright, M. (1992), 'Statistical motor rating: making effective use of your data', *Journal of the Institute of Actuaries* **119**, 457–543.

Brouhns, N., Denuit, M., Masuy, B. and Verrall, R. (2002), 'Ratemaking by geographical area in the Boskov and Verrall model : a case study using belgian car insurance data', *Actu–L* **2**, 3–28.

Cameron, A. and Trivedi, P. (1998), *Regression Analysis of Count Data*, Cambrigde University Press, New York.

Casella, G. and George, E. (1992), 'Explaining Gibbs sampler', *American Statistical Association* **46**(3), 167–174.

Chib, S. and Greenberg, E. (1995), 'Understanding the Metropolis–Hastings algorithm', *American Statistical Association* **49**(4), 327–335.

Dean, C., Lawless, J. and Willmot, G. (1989), 'A mixed Poisson–inverse–Gaussian regression model', *The Canadian Journal of Statistics* **17**(2), 171–181.

Deb, P. and Trivedi, P. (1997), 'Demand for medical care by the elderly: a finite mixture approach', *Journal of Applied Econometrics* **12**, 313–336.

Dimakos, X. K. and Frigessi, A. (2002), 'Bayesian premium rating with latent structure', *Scandinavian Actuarial Journal* **3**, 162–184.

Dionne, G. and Vanasse, C. (1989), 'A generalization of automobile insurance rating models: the negative binomial distribution with a regression component', *ASTIN Bulletin* **19**, 199–212.

Eilers, P. H. and Marx, B. D. (1996), 'Flexible smoothing using B–splines and penalized likelihood (with comments and rejoinder)', *Statistical Science* **11**(2), 89–121.

Fahrmeir, L. and Hennerfeind, A. (2003), 'Nonparametric Bayesian hazard rate models based on penalized splines', *SFB 386 Discussion paper, Department of Statistics, University of Munich* **DP 361**.

Fahrmeir, L., Kneib, T. and Lang, S. (2003), 'Penalized structured additive regression for space-time data: A Bayesian perspective', *Statistica Sinica (under revision). Available from* `www.stat.uni-muenchen.de/~kneib/papers.html`.

Fahrmeir, L. and Lang, S. (2001*a*), 'Bayesian inference for generalized additive mixed models based on Markov random field priors', *Applied Statistics, (JRSS C)* **50**, 201–220.

Fahrmeir, L. and Lang, S. (2001*b*), 'Bayesian semiparametric regression analysis of multicategorical time-space data', *Annals of the Institute of Statistical Mathematics* **53**, 10–30.

Fahrmeir, L., Lang, S. and Spies, F. (2003), 'Generalized geoadditive models for insurance claims data', *Bltter der Deutschen Gesellschaft fr Versicherungsmathematik* **26**, 7–23.

Fahrmeir, L. and Osuna Echavarría, L. (2003), 'Structured count data regression', *SFB 386 Discussion Paper, Department of Statistics, University of Munich* **DP 334**.

Fahrmeir, L. and Tutz, G. (2001), *Multivariate statistical modelling based on generalized linear models, 4^{th} ed.*, Springer Series in Statistics, New York.

Gamerman, D. (1997*a*), 'Efficient sampling from the posterior distribution in generalized linear mixed models', *Statistics and Computing* **7**, 57–68.

Gamerman, D. (1997*b*), *Markov Chain Mote Carlo, Stochastic simulation for Bayesian inference*, Chapman and Hall, London.

Gelman, A., Carlin, J., Stern, H. and Rubin, D. (1995), *Bayesian Data Analysis*, Chapman and Hall, London.

Guo, J. Q. and Trivedi, P. K. (2002), 'Flexible parametrics models for long–tailed patent count distributions', *Oxford Bulletin of Economics and Statistics* **64**, 63–82.

Gurmu, S. (1997), 'Semiparametric estimation of hurdle regression models with an application to medicaid utilization', *Journal of Applied Econometrics* **12**, 225–242.

Hastie, T. and Tibshirani, R. (1990), *Generalized additive models*, Chapman and Hall, London.

Hennerfeind, A., Brezger, A. and Fahrmeir, L. (2003), 'Geoadditive survival models', *SFB Discussion paper, Department of Statistics, University of Munich* **DP 333**.

Hinde, J. and Demétrio, C. G. B. (1998), 'Overdispersion: Models and estimation', *Computational Statistics and Data Analysis* **27**, 151–170.

Jerak, A. and Wagner, S. (2003), 'Modeling probabilities of patent oppositions in a Bayesian semiparametric regression framework', *SFB 386 Discussion Paper, Department of Statistics, University of Munich* **DP 323**.

Johnson, L. N. and Kotz, S. (1969), *Discrete Distributions*, John Wiley and Sons, New York.

Jørgensen, B. and Paes de Souza, M. C. (1994), 'Fitting Tweedie's compound Poisson model to insurance claims data', *Scandinavian Actuarial Journal* **1**, 69–93.

Kaas, R. and Hesselager, O. (1995), 'Ordering claim size distributions and mixed Poisson probabilities', *Insurance: Mathematics and Economics* **17**, 193–201.

Karlis, D. (2001), 'A general EM approach for maximum likelihood estimation in mixed Poisson regression models', *Statistical Modelling* **1**, 305–318.

Knorr-Held, L. (1997), Hierarchical Modelling of Discrete Longitudinal Data, PhD thesis, Ludwig Maximilian Universität, München.

Knorr-Held, L. (1999), 'Conditional prior proposals in dynamic models', *Scandinavian Journal of Statistics* **26**, 129–144.

Lambert, D. (1992), 'Zero inflated Poisson regression with an application to defects in manufacturing', *Technometrics* **34**, 1–14.

Lang, S. (2004), Structured additive regression: models, inference and applications, Postdoctoral lecture qualification, Ludwig Maximilian Universität, München.

Lang, S. and Brezger, A. (2004), 'Bayesian P–splines', *Journal of Computational and Graphical Statistics* **13**, 183–212.

Lee, A. H., Stevenson, M. R., Wang, K. and Yau, K. K. W. (2002), 'Modeling young driver motor vehicle crashes: data with extra zeros', *Accident Analysis and Prevention* **34**, 515–521.

McCullagh, P. and Nelder, J. (1989), *Generalized linear models, 2nd ed.*, Chapman and Hall, London.

Mullahy, J. (1997), 'Heterogeneity, excess zeros, and the structure of count data models', *Journal of applied econometrics* **12**, 337–350.

Podlich, H., Faddy, M. and Smyth, G. (1999), 'Semi–parametric extended Poisson process models', *Research Report, Department of Statistics and Demography, University of Southern Denmark* .

Renshaw, A. E. (1994), 'Modelling the claims process in the presence of covariates', *ASTIN Bulletin* **24**, 265–285.

Ridout, M., Demétrio, C. and Hinde, J. (1998), 'Models for count data with many zeros', *Proceedings of XIXth International Biometric Society Conference,IBC98, Cape Town* pp. 179–192.

Ridout, M., Hinde, J. and Demétrio, C. (2001), 'A score test for testing a zero–inflated Poisson regression model against zero–inflated negative binomial alternatives', *Biometrics* **57**, 219–223.

Schlüter, P. J., Deely, J. J. and Nicholson, A. J. (1997), 'Ranking and selecting motor vehicle accidents sites by using a hierarchical Bayesian model', *The Statistician* **46**(3), 293–316.

Shaked, M. (1980), 'On mixtures from exponential families', *Journal of the Royal Statistical Society, Series B* **42**, 192–198.

Smyth, G. K. and Jørgensen, B. (2002), 'Fitting Tweedie's compound Poisson model to insurance claims data: dispersion modelling', *ASTIN Bulletin* **32**, 143–157.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P. and van der Linde, A. (2002), 'Bayesian measures of model complexity and fit', *Journal of the Royal Statistical Society, Series B* **64**(3), 1–34.

Sutradhar, B. C. and Jowaheer, V. (2001), 'Log normal versus gamma random effects in a familial longitudinal Poisson mixed model', *unpublished manuscript* .

Thurston, S. W., Wand, M. P. and Wiencke, J. K. (2000), 'Negative binomial additive models', *Biometrics* **56**, 139–144.

Tremblay, L. (1992), 'Using the Poisson inverse Gaussian in bonus–malus systems', *ASTIN Bulletin* **22**, 97–106.

Viallefont, V., Richardson, S. and Green, P. J. (2002), 'Bayesian analysis of Poisson mixtures', *Journal of Nonparametric Statistics* **14**(1–2), 181–202.

Wikle, C. K. and Anderson, C. J. (2003, to appear), 'Climatiological analysis of tornado report counts using a hierarchical Bayesian spatio–temporal model', *Journal of Geophysical Research* .

Winkelmann, R. (1995), 'Duration dependence and dispersion in count models', *Journal of Business and Economic Statistics* **13**(4), 467–474.

Winkelmann, R. (1996), 'Markov Chain Monte Carlo analysis of underreported count data with an application to worker absenteeism', *Empirical Economics* **21**(4), 575–587.

Winkelmann, R. (1998), 'Count data models with selectivity', *Econometric Reviews* **17**, 339–359.

Winkelmann, R. and Zimmermann, K. F. (1995), 'Recent developments in count data modelling: theory and application', *Journal of Economic Surveys* **9**(1), 1–24.

Zorn, C. (1998), 'Evaluating zero–inflated and hurdle Poisson specifications', *Sociological Methods and Research* **26**, 368–400.

Lebenslauf

Leyre Estíbaliz Osuna Echavarría

geboren am 01. Dezember 1975 in Sevilla (Spanien)

Schulausbildung

09/1981–06/1989	Grundschule in Córdoba (Spanien)
09/1989–07/1993	Gymnasium in Córdoba (Spanien)

Studium und Promotion

09/1993–07/1998	Studium im Fach Mathematik mit Fachrichtung Statistik an der Universidad Complutense de Madrid
10/1996–11/1997	Studienaustausch mit der TU München im Rahmen des ERASMUS–Programmes
Seit 10/1999	Promotion an der LMU München

Arbeitserfahrung

01/1999–07/1999	Studentische Hilfskraft im Statistischen Beratungslabor (STABLAB)
09/1999–01/2001	Wissenschaftliche Mitarbeiterin im SFB 386
02/2001–01/2004	Stipendiatin im Graduiertenkolleg Angewandte Algorithmische Mathematik an der TU München

München, 5. August 2004