University of São Paulo
“Luiz de Queiroz” College of Agriculture

GAMLSSs with applications to zero inflated and hierarquical data

Gustavo Thomas

Dissertation presented to obtain the degree of Master in Science. Area: Statistics and Agricultural Experimentation

Piracicaba
2017

Gustavo Thomas
BSc in Statistics

GAMLSSs with applications to zero inflated and hierarquical data

Advisor:

Profª Drª **CLARICE GARCIA BORGES DEMÉTRIO**

Dissertation presented to obtain the degree of Master in Science. Area: Statistics and Agricultural Experimentation

Piracicaba
2017

Dados Internacionais de Catalogação na Publicação
DIVISÃO DE BIBLIOTECA - DIBD/ESALQ/USP

Thomas, Gustavo

GAMLSSs with applications to zero inflated and hierarquical data / Gustavo Thomas. -- Piracicaba, 2017 .

75 p.

Dissertação (Mestrado) -- USP / Escola Superior de Agricultura "Luiz de Queiroz".

1. GAMLSS 2. Modelos para dados inflacionados de zeros 3. Modelos mistos 4. Software R . I. Título.

*To my parents, Silvério and Claci
and my brothers, Rodrigo and Dionei*

ACKNOWLEDGEMENTS

First of all, I would like to thank God for giving me health and strength when I needed the most during this Master's. Living alone far from your family always has tough moments and believing that there was someone somewhere above me always kept me going.

That said, I would like to say that every day away from home makes me appreciate more the place where I was born and all the love my parents and brothers gave me ever since I was born, thank you so much.

I would also like to thank my advisor, Clarice, for the wise advices and support throughout this period. Actually, I am thankful to all professors I had contact with in the department of Exact Sciences of ESALQ. They taught me a lot and I have no bad words to say about any of them.

These two years seem to have passed so fast when I think of all experiences I shared with the friends I made here. I am particularly thankful to my Masters' colleagues and to the so called "Cabeças de baleia" (Whale heads), a group of friends from the Department of Economy. I hope our friendships are lifelong.

Finally, I would like to thank CAPES for providing financial support during these 22 months. I could not have lived away from home without this funding.

“In God we trust; all others bring data.”

William Edwards Deming

CONTENTS

Resumo	7
Abstract	8
1 Introduction	9
References	9
2 Generalized additive models for location, scale and shape (GAMLSS)	11
2.1 Definition of a GAMLSS and submodels	11
2.2 Model estimation	13
2.2.1 The RS algorithm	15
2.2.2 The CG algorithm	17
2.2.3 Estimating the hyperparameters λ	19
2.3 Spline smoothing additive terms	19
2.4 Model selection	21
2.5 Diagnosis	22
2.6 Methods for comparing models	24
References	24
3 Modeling data with zero inflation and overdispersion using GAMLSSs	27
3.1 Introduction	27
3.2 Material and methods	28
3.3 Results and discussion	33
3.4 Conclusion	38
References	38
4 Analysis of a longitudinal split-plot with subsampling experiment using GAMLSSs	41
4.1 Introduction	41
4.2 Experiment description	42
4.3 Analysis	43
4.3.1 Using function <i>nlme::lme()</i>	46
4.3.2 Using function <i>gamlss::gamlss()</i>	50
4.4 Conclusions	59
References	60
5 Final considerations	65
Appendix	65

RESUMO

GAMLSSs com aplicações a dados inflacionados de zeros e hierárquicos

Os modelos lineares generalizados para locação, escala e forma (GAMLSS) desenvolvidos por *Rigby e Stasinopoulos (2005)* são uma ampla classe de modelos de regressão univariados que não pressupõem que a distribuição da variável resposta pertença à família exponencial como os modelos lineares generalizados ou aditivos generalizados, por exemplo. Além do mais, eles permitem que todos os parâmetros da distribuição da variável resposta sejam modelados explicitamente por meio de diferentes conjuntos de variáveis explanatórias. A subclasse semiparamétrica dos GAMLSS, em particular, permite que uma grande variedade de termos paramétricos e não paramétricos sejam incluídos nos preditores dos parâmetros da distribuição assumida para a variável resposta. De forma análoga aos modelos lineares generalizados, os GAMLSSs ligam os preditores aos parâmetros por meio de funções de ligação monótonas, que também podem mudar de acordo com o parâmetro a ser estimado. Esta dissertação descreve a metodologia dos modelos lineares generalizados para locação, escala e forma e apresenta duas aplicações a bancos de dados provenientes de experimentos agrícolas; explorando métodos de estimação, diagnóstico e comparação desse tipo de modelos.

Palavras-chave: GAMLSSs inflacionados de zeros; Contagem de raízes; GAMLSSs mistos; Crescimento de milho

ABSTRACT

GAMLSSs with applications to zero inflated and hierarchical data

The generalized additive models for location, scale and shape (GAMLSS) developed by RIGBY and STASINOPOULOS (2005) are a general class of univariate regression models that do not have the response distribution restricted to the exponential family as do the generalized linear and additive models, for example. In addition, they allow all the parameters of the response variable distribution to be modeled explicitly through different sets of explanatory variables. The semiparametric subclass of GAMLSS, in particular, accepts a wide range of parametric and nonparametric terms to be included in the predictors of the parameters. Similar to the generalized linear models, the GAMLSSs link predictors to parameters through monotonic link functions, which can also change for each parameter. This dissertation describes the GAMLSSs methodology and presents two applications to data sets provenient from experiments in agronomy; exploring methods of estimation, diagnosis and comparison of these models.

Keywords: Zero inflated GAMLSSs; Count of roots; Mixed GAMLSSs; Corn growth

1 INTRODUCTION

Many univariate regression models were created since the first uses of the least squares method by Legendre and Gauss in the beginning of the 19th century. The linear regression model was widely used ever since and is still an area of active research. On the second half of the last century, new robust and more flexible models started to gain popularity in the statistical community. Among these are the generalized linear models (GLMs) and the generalized additive models (GAMs), introduced by NELDER and WEDDERBURN (1972) and HASTIE and TIBSHIRANI (1990) respectively. These two types of models assume that the response variable follows an exponential family distribution with mean modeled explicitly as a function of explanatory variables. Later on, mixed versions of the GLMs and GAMs (the GLMMs and GAMMs) were developed to account for random effects. However, neither of these models are capable of modeling other parameters than the mean or the variance of the response distribution explicitly as functions of the explanatory variables.

To overcome some of these model's limitations, RIGBY and STASINOPOULOS (2005) developed the generalized additive models for location, scale and shape (GAMLSSs). This more general class of univariate regression models includes characteristics of both GLMs and GAMs in the sense that parameters are linked to predictors through known monotonic link functions which can include a wide range (and combination) of linear/non-linear and random effect/nonparametric terms. However, differently than in a GLM or GAM it allows the modeling of all response variable parameters through separate sets of explanatory variables. Often modeling the mean and variance is of interest and for large datasets higher moments of the response might also be. This is possible because in the GAMLSSs framework the exponential family assumption is relaxed to any distribution that has its probability mass/density function and first, second and cross derivatives with respect to each of the parameters computable. The model fitting is made by algorithmic procedures based on Fisher scoring or Newton-Raphson methods.

The objective of this dissertation consists on giving a brief explanation of this general class of models and exploring the features of this framework by analysing real data sets from experiments in agronomics. The text is organized in three chapters (chapters 2 to 4). Chapter 2 describes the GAMLSSs methodology giving the definition, estimation methods, selection techniques, diagnosis and comparison methods of such models. Zero inflated GAMLSSs are the focus of chapter 3, where the analysis of an experiment in horticulture is described. Finally, the 4th chapter reports the analysis of an experiment with sweet corn using a GAMLSS with random effects.

References

HASTIE, T. J. and R. J. TIBSHIRANI, 1990 *Generalized Additive Models*. CRC Press.

NELDER, J. A. and R. W. M. WEDDERBURN, 1972 Generalized Linear Models. J. R. Statist. Soc. A. **135**: 370–384.

RIGBY, R. A. and D. M. STASINOPOULOS, 2005 Generalized additive models for location, scale and shape,(with discussion). Journal of the Royal Statistical Society: Series C (Applied Statistics) **54**: 507–554.

2 GENERALIZED ADDITIVE MODELS FOR LOCATION, SCALE AND SHAPE (GAMLSS)

This chapter describes the generalized additive models for location, scale and shape of RIGBY and STASINOPOULOS (2005). Section 2.1 gives a general definition of this type of models and derives important submodels. The estimation method is explained in section 2.2, which is divided in subsections 2.2.1 and 2.2.2 where the RS and CG fitting algorithms are presented, respectively. Section 2.3 describes some of the possible GAMLSS's additive terms. The chapter has three more sections that are about model selection (section 2.4), diagnosis (section 2.5) and procedures for comparison of GAMLSSs (section 2.6).

2.1 Definition of a GAMLSS and submodels

As defined by RIGBY and STASINOPOULOS (2005), a GAMLSS is an univariate regression type model in that the parameters of the response variable distribution are modeled with functions of explanatory variables and may involve parametric and/or non-parametric terms. It assumes $\mathbf{Y}_i, i = 1, \dots, n$ to be a vector denoting an independent random variable conditional on $\boldsymbol{\theta}$ following a probability distribution $D_{\boldsymbol{\theta}}$ which might have a maximum of 4 parameters ($\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4)' = (\mu, \sigma, \nu, \tau)'$) and probability density function given by $f(\mathbf{y}_i | \boldsymbol{\theta})$. Within the GAMLSSs framework, the parameters of the distributions used are always denoted by μ, σ, ν and τ , in this order. For most of the GAMLSSs distributions the first parameter θ_1 (μ) is a location parameter associated with the mean of the distribution and θ_2 (σ) is a scale parameter associated with the distribution's variance. The remaining parameters, if available, are usually related to the shape of the distribution.

A GAMLSS can be represented in a compact form where the link functions $g_k(\cdot)$ (for $k = 1, 2, 3, 4$) link the parameters $\boldsymbol{\theta}_k$ ($n \times 1$) to predictors by

$$g_k(\boldsymbol{\theta}_k) = \boldsymbol{\eta}_k = \mathbf{X}_k \boldsymbol{\beta}_k + \sum_{j=1}^{J_k} \mathbf{Z}_{jk} \boldsymbol{\gamma}_{jk}, \quad (2.1)$$

which can be expanded to

$$\begin{aligned} g_1(\mu) &= \boldsymbol{\eta}_1 = \mathbf{X}_1 \boldsymbol{\beta}_1 + \sum_{j=1}^{J_1} \mathbf{Z}_{j1} \boldsymbol{\gamma}_{j1} \\ g_2(\sigma) &= \boldsymbol{\eta}_2 = \mathbf{X}_2 \boldsymbol{\beta}_2 + \sum_{j=1}^{J_2} \mathbf{Z}_{j2} \boldsymbol{\gamma}_{j2} \\ g_3(\nu) &= \boldsymbol{\eta}_3 = \mathbf{X}_3 \boldsymbol{\beta}_3 + \sum_{j=1}^{J_3} \mathbf{Z}_{j3} \boldsymbol{\gamma}_{j3} \\ g_4(\tau) &= \boldsymbol{\eta}_4 = \mathbf{X}_4 \boldsymbol{\beta}_4 + \sum_{j=1}^{J_4} \mathbf{Z}_{j4} \boldsymbol{\gamma}_{j4}; \end{aligned} \quad (2.2)$$

where each $\boldsymbol{\eta}_k$; $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{\nu}$ and $\boldsymbol{\tau}$ are vectors of size n . The monotonic link functions $g_k(\cdot)$ s in formulae (2.1) and (2.2) relate the distribution parameters to the respective predictors. The $\mathbf{X}_k$ s are known design matrices of order $n \times J'_k$ containing information of the covariates associated with the fixed effects $\boldsymbol{\beta}_k$ of dimension $J'_k \times 1$. The $\mathbf{Z}_{jk}$ s are known design matrices of order $n \times q_{jk}$ associated with q_{jk} -dimensional random variables $\boldsymbol{\gamma}_{jk}$, which are assumed to be distributed as $\boldsymbol{\gamma}_{jk} \sim N_{q_{jk}}(\mathbf{0}, \mathbf{G}_{jk}^{-1})$. In this notation, $\mathbf{G}_{jk}^{-1}$ is the (generalized) inverse of a $q_{jk} \times q_{jk}$ symmetric matrix $\mathbf{G}_{jk} = \mathbf{G}_{jk}^{-1}(\boldsymbol{\lambda}_{jk})$ which may depend on an extra vector of parameters $\boldsymbol{\lambda}_{jk}$ (hyper or smoothing parameters).

In definition (2.1) each predictor $\boldsymbol{\eta}_k$ is made up of a parametric component $\mathbf{X}_k\boldsymbol{\beta}_k$ and an additive component $\mathbf{Z}_{jk}\boldsymbol{\gamma}_{jk}$. However, the additive components are optional and the explanatory variables used can be different for each predictor. In fact, from this definition one can derive several submodels that arise from different model formulations. For example, if additive terms are not present in (2.1) (that is, $\mathbf{Z}_{jk}\boldsymbol{\gamma}_{jk} = 0$ for all j and k), the resulting model is the parametric GAMLSS represented as

$$\begin{aligned} g_1(\boldsymbol{\mu}) &= \boldsymbol{\eta}_1 = \mathbf{X}_1\boldsymbol{\beta}_1 \\ g_2(\boldsymbol{\sigma}) &= \boldsymbol{\eta}_2 = \mathbf{X}_2\boldsymbol{\beta}_2 \\ g_3(\boldsymbol{\nu}) &= \boldsymbol{\eta}_3 = \mathbf{X}_3\boldsymbol{\beta}_3 \\ g_4(\boldsymbol{\tau}) &= \boldsymbol{\eta}_4 = \mathbf{X}_4\boldsymbol{\beta}_4. \end{aligned} \tag{2.3}$$

This model can be extended to allow nonlinear functions, changing to

$$\begin{aligned} g_1(\boldsymbol{\mu}) &= \boldsymbol{\eta}_1 = h_1(\mathbf{X}_1, \boldsymbol{\beta}_1) \\ g_2(\boldsymbol{\sigma}) &= \boldsymbol{\eta}_2 = h_2(\mathbf{X}_2, \boldsymbol{\beta}_2) \\ g_3(\boldsymbol{\nu}) &= \boldsymbol{\eta}_3 = h_3(\mathbf{X}_3, \boldsymbol{\beta}_3) \\ g_4(\boldsymbol{\tau}) &= \boldsymbol{\eta}_4 = h_4(\mathbf{X}_4, \boldsymbol{\beta}_4), \end{aligned} \tag{2.4}$$

where $\mathbf{h}_k(\cdot)$ for $k = 1, 2, 3, 4$ represent linear or nonlinear functions of the explanatory variables $\mathbf{X}_k$ associated with the vector of fixed effects parameters $\boldsymbol{\beta}_k$. When all $\mathbf{h}_k(\cdot)$ are nonlinear (nonlinear functions are used for all covariates) in (2.4), the resulting model is called the nonlinear parametric GAMLSS. If only some are nonlinear, the model contains linear and nonlinear functions of the explanatory variables but is still a purely parametric GAMLSS.

When additive components are included, the model might also change according to which additive components are used. If the design matrices of the additive components $\mathbf{Z}_{jk}$ are assumed to be identity $n \times n$ matrices $\mathbf{I}_n$ and the random variables $\boldsymbol{\gamma}_{jk}$ change to nonparametric functions of explanatory variables $\mathbf{h}_{jk} = h_{jk}(\mathbf{x}_{jk})$ for all combinations of j and k in (2.1), the underlying model is the semiparametric additive GAMLSS given in (2.5). There are many types of additive terms supported by GAMLSSs; one of them is briefly discussed in section 2.3. A complete description is given in chapter 9 of STASINOPOULOS *et al.* (2017).

$$\begin{aligned}
g_1(\boldsymbol{\mu}) &= \boldsymbol{\eta}_1 = \mathbf{X}_1\boldsymbol{\beta}_1 + \sum_{j=1}^{J_1} h_{j1}(\mathbf{x}_{j1}) \\
g_2(\boldsymbol{\sigma}) &= \boldsymbol{\eta}_2 = \mathbf{X}_2\boldsymbol{\beta}_2 + \sum_{j=1}^{J_2} h_{j2}(\mathbf{x}_{j2}) \\
g_3(\boldsymbol{\nu}) &= \boldsymbol{\eta}_3 = \mathbf{X}_3\boldsymbol{\beta}_3 + \sum_{j=1}^{J_3} h_{j3}(\mathbf{x}_{j3}) \\
g_4(\boldsymbol{\tau}) &= \boldsymbol{\eta}_4 = \mathbf{X}_4\boldsymbol{\beta}_4 + \sum_{j=1}^{J_4} h_{j4}(\mathbf{x}_{j4}).
\end{aligned} \tag{2.5}$$

The amount of smoothing allowed by the nonparametric functions h_{jk} in (2.5) is determined by extra (smoothing) parameters λ_{jk} for each of them. This semiparametric GAMLSS can also be extended to allow nonlinear terms in the predictors, so that it becomes

$$\begin{aligned}
g_1(\boldsymbol{\mu}) &= \boldsymbol{\eta}_1 = h_1(\mathbf{X}_1, \boldsymbol{\beta}_1) + \sum_{j=1}^{J_1} h_{j1}(\mathbf{x}_{j1}) \\
g_2(\boldsymbol{\sigma}) &= \boldsymbol{\eta}_2 = h_2(\mathbf{X}_2, \boldsymbol{\beta}_2) + \sum_{j=1}^{J_2} h_{j2}(\mathbf{x}_{j2}) \\
g_3(\boldsymbol{\nu}) &= \boldsymbol{\eta}_3 = h_3(\mathbf{X}_3, \boldsymbol{\beta}_3) + \sum_{j=1}^{J_3} h_{j3}(\mathbf{x}_{j3}) \\
g_4(\boldsymbol{\tau}) &= \boldsymbol{\eta}_4 = h_4(\mathbf{X}_4, \boldsymbol{\beta}_4) + \sum_{j=1}^{J_4} h_{j4}(\mathbf{x}_{j4}).
\end{aligned} \tag{2.6}$$

Model (2.6) is the nonlinear semiparametric GAMLSS. The submodels presented in this section are a general representation of the submodels that the GAMLSSs encompass according to the terms included. A GAMLSS can contain linear, nonlinear and nonparametric functions of the explanatory variables and random effects, all at the same time and in the prediction of any or all of the response distribution parameters. Obviously this would not make sense in practice, but shows the variety of options that these models offer to anyone willing to build a regression model and learn from data.

2.2 Model estimation

The estimation of a GAMLSS depends on the distribution assumed for the response variable and changes whether additive terms are being used or not. For the parametric formulation in (2.3) the $\boldsymbol{\beta}_k$ s are the only parameters to be estimated, while the semiparametric GAMLSS in (2.5) requires the estimation of the $\boldsymbol{\beta}_k$ s, the $\boldsymbol{\gamma}_k$ s and the $\boldsymbol{\lambda}_k$ s, for instance. Within the R implementation of the GAMLSSs (package `gamlss`), a parametric GAMLSS is fitted by maximum likelihood estimation with respect to the fixed parameters $\boldsymbol{\beta}$. The log-likelihood function of such a model assuming independent observations is given by

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^n \log f(y_i \mid \mu_i, \sigma_i, \nu_i, \tau_i). \tag{2.7}$$

When there are additive terms in the model, either random effects or nonparametric functions, they are expressed as $h_{jk}(\mathbf{Z}_{jk}) = \mathbf{Z}_{jk}\boldsymbol{\gamma}_{jk}$, where the $\mathbf{Z}_{jk}$ $n \times q_{jk}$ matrix represents a B-spline basis design and $\boldsymbol{\gamma}_{jk}$ is a q_{jk} -dimensional vector of B-spline parameters. The representation of random effects' models as nonparametric functions is well stated in the literature, particularly well described and exemplified for many different types of nonparametric functions in WAND (2003), but can also be found in WAHBA (1990), WOOD (2006) and SPEED (2007), just to name a few. In the context of GAMLSSs, details can be found in chapter 9 of STASINOPOULOS *et al.* (2017). When the values of the smoothing parameters λ_{jk} s are fixed (this might be desired for some nonparametric functions), the fixed effects β_k s and the random effects γ_{jk} s are estimated by maximizing a penalized likelihood function given by

$$\ell_p(\boldsymbol{\theta}, \boldsymbol{\lambda}) = \ell(\boldsymbol{\theta}) - \frac{1}{2} \sum_{k=1}^4 \sum_{j=1}^{J_k} \lambda_{kj} \boldsymbol{\gamma}'_{kj} \mathbf{G}_{kj} \boldsymbol{\gamma}_{kj}. \quad (2.8)$$

In this function, $\ell(\boldsymbol{\theta})$ is the log-likelihood in (2.7) and $\mathbf{G}_{kj}$ is the symmetric $q_{jk} \times q_{jk}$ matrix defined for model (2.2) whose (generalized) inverse is the variance-covariance matrix of random effects λ_{kj} . If all additive terms are random effects (there are not any nonparametric functions), all λ s are equal to 1 in (2.8).

For maximizing either the log-likelihood function (2.7) or the penalized log-likelihood (2.8) for fixed values of the (random effect or smoothing) hyperparameters $\boldsymbol{\lambda}$, RIGBY and STASINOPOULOS (2005) introduced the RS and CG GAMLSS algorithms that are implemented in the function `gamlss::gamlss()` in R. The estimation of the smoothing parameters $\boldsymbol{\lambda}$ is briefly discussed in section 2.2.3. Both the RS and CG algorithms use the log-likelihood (2.7) of the data and its first and (optionally expected) second derivatives with respect to each of the distribution parameters. However, the CG algorithm (which is a generalization of the algorithm used by COLE and GREEN (1992)) additionally makes use of expected cross derivatives of the log-likelihood in the fitting process. When used to maximize (2.8) for given $\boldsymbol{\lambda}$, both methods lead to the maximum penalized log-likelihood estimates for the fixed and random effects (proof in Appendix C of RIGBY and STASINOPOULOS (2005)). During the fitting process, the CG and RS iterative algorithms make successive checks for convergence of the model's global deviance (GD). Particular attention should be given to the GD definition in a GAMLSS, which is different than in a GLM, for example. While a GLM deviance is defined as

$$D_{GLM} = -2 \log \left(\frac{\hat{L}_c}{\hat{L}_s} \right), \quad (2.9)$$

where $\hat{L}_c$ is the fitted likelihood of the current fitted model and $\hat{L}_s$ that of the saturated model (where in modeling μ a parameter is fitted for each observation), a GAMLSS deviance is defined as

$$D_{GAMLSS} = -2 \log(\hat{L}_c). \quad (2.10)$$

The RS and CG algorithms are explained in more details in the following subsections.

2.2.1 The RS algorithm

The RS iterative algorithm maximizes the (penalized) likelihood over each of the parameters in turn, cycling until convergence. It is faster and usually more stable than CG, being therefore the default method in the `gamlss()` function. To start the algorithm, initial values (constants) for the distribution parameter vectors μ , σ , ν and τ (assuming the response distribution has 4 parameters) are provided by default or can be given by the user. The algorithm is best described as having three nested parts: the outer iterations, the inner or GLIM iterations and the modified backfitting algorithm, each nested in the previous. These three parts of the RS algorithm are sketched in Figure 2.1 (which also applies to the CG algorithm) to aid the explanation of the algorithm process.

Figure 2.1. Sketch of the three parts/layers of the GAMLSSs algorithms.

The outer iteration starts by maximizing the (penalized) log-likelihood over the first parameter vector μ (given the latest estimates of the other parameters vectors), then over the second parameter vector σ , then ν and finally over τ . This maximization is done via the inner/GLIM and the backfitting algorithms, which are nested inside the outer iteration. After the likelihood has been maximized over each parameter vector in turns, a measure of global deviance (GD) is checked for convergence. If the GD did not converge, the outer algorithm begins another iteration.

For the maximization over each parameter vector, the outer algorithm calls the inner algorithm, which is also known as GLIM algorithm because of its similarity to an iterative reweighted least squares (IRLS). The idea of this local scoring algorithm is exactly to do repeated weighted fits to a modified response variable (using modified weights) until convergence when the maximum is reached. The modified/iterative weights vector for each of the k parameters $\boldsymbol{\theta}_k$ is given by

$$\mathbf{w}_k = -\mathbf{f}_k \circ \frac{d\boldsymbol{\theta}_k}{d\boldsymbol{\eta}_k} \circ \frac{d\boldsymbol{\theta}_k}{d\boldsymbol{\eta}_k}, \quad (2.11)$$

where $\circ$ here denotes the Hadamard element by element product, $\boldsymbol{\eta}_k = g_k(\boldsymbol{\theta}_k)$ of $n \times 1$ is the k -th parameter predictor vector and $\mathbf{f}_k$ depends on the information available for the response distribution assumed, so that

$$\mathbf{f}_k = \begin{cases} E\left(\frac{\partial^2 \ell}{\partial \boldsymbol{\theta}_k^2}\right), & \text{if this expectation exists, leading to a Fisher scoring algorithm;} \\ -\left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_k}\right) \circ \left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_k}\right) & \text{otherwise, leading to a quasi Newton-Raphson algorithm.} \end{cases} \quad (2.12)$$

The first derivative of the log-likelihood (that is, the score function) with respect to the k -th predictor is given by

$$\mathbf{u}_k = \frac{\partial \ell}{\partial \boldsymbol{\eta}_k} = \left(\frac{\partial \ell}{\partial \boldsymbol{\theta}_k}\right) \circ \left(\frac{d\boldsymbol{\theta}_k}{d\boldsymbol{\eta}_k}\right), \quad (2.13)$$

where all terms in (2.13) are vectors of size n , the number of observations. The modified response variable (also known as working variable) is given by

$$\mathbf{z}_k = \boldsymbol{\eta}_k + \mathbf{w}_k^{-1} \circ \mathbf{u}_k, \quad (2.14)$$

which is also a n -sized vector, with elements $(\eta_{k1} + w_{k1}^{-1}u_{k1}, \eta_{k2} + w_{k2}^{-1}u_{k2}, \dots, \eta_{kn} + w_{kn}^{-1}u_{kn})^T$.

These constructed vectors are used by the inner algorithm, which starts by calculating the iterative weights $\mathbf{w}_k$ and the working variable values $\mathbf{z}_k$ for the latest estimates of the parameter vectors $\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\nu}}$ and $\hat{\boldsymbol{\tau}}$. Then, the modified backfitting algorithm, which is nested within the inner iteration, is called to fit the linear explanatory variables and (if needed, the linear parts of) smoothers to the modified working variable $\mathbf{z}_k$. By that, new parameter and predictor estimates are obtained and a (new) value for the GD (at the inner iteration level) is calculated. If the GD has converged, the algorithm goes back to the outer iteration and starts the fitting of the next vector of parameters (if there is any left, otherwise checks GD at the outer iteration level to finish or start a whole new iteration). If not, the inner iteration restarts its cycle.

The modified backfitting algorithm, ‘the innermost’ part of the GAMLSS RS algorithm, is the one that properly does the estimation of the fixed effect parameters $\boldsymbol{\beta}_k$

and random effects γ_k . It is a version of the Gauss-Seidel algorithm used by HASTIE and TIBSHIRANI (1990) modified by the fact that only the linear parts of the covariates related to the penalized smoothers are used in their correspondent design matrices, which improves the algorithm convergence.

To understand how the backfitting algorithm works, suppose $\mathbf{X}_k$ represents the design matrix for the linear part of the model and that $\mathbf{Z}_{k1}$ and $\mathbf{Z}_{k2}$ represent the basis matrices of two smoothers with parameter sets γ_{k1} and γ_{k2} , respectively. With the current weights $\mathbf{w}_k$ and variable values $\mathbf{z}_k$ updated in the inner iteration and previously estimated (or initialized) values for the smoothers $\hat{\gamma}_{k1}$ and $\hat{\gamma}_{k2}$, the backfitting algorithm calculates the partial residuals $\varepsilon = \mathbf{z}_k - \mathbf{Z}_{k1}\hat{\gamma}_{k1} - \mathbf{Z}_{k2}\hat{\gamma}_{k2}$ with respect to the fixed effects parameters β_k . Using a weighted least squares (WLS) procedure, it obtains new estimates for the fixed parameters $\hat{\beta}_k$. Then, the partial residuals $\varepsilon = \mathbf{z}_k - \mathbf{X}_k\hat{\beta}_k - \mathbf{Z}_{k2}\hat{\gamma}_{k2}$ with respect to the first smoother γ_{k1} are calculated and a penalized weighted least squares (PWLS) is used to update $\hat{\gamma}_{k1}$. Analogously for the second smoother γ_{k2} , partial residuals with respect to it are obtained and with them new values for $\hat{\gamma}_{k2}$ are estimated using PWLS. If there were more smoothing terms they would be estimated in the same way, sequentially. This is not what is done within the `mgcv::gam()` function in R for fitting generalized additive models, where both linear and smoother components are fitted simultaneously. The reason why this is not done in GAMLSSs (STASINOPOULOS *et al.* (2017), section 3.2.1) is that this simultaneous modeling only works with penalized smoothers, while using backfitting it is possible to also incorporate other types of smoothers such as LOESS (CLEVELAND and DEVLIN, 1988), cubic splines and neural networks within GAMLSSs. The backfitting process is repeated until $\hat{\beta}_k$, $\hat{\gamma}_{k1}$ and $\hat{\gamma}_{k2}$ converge, then it switches to the inner iteration to check GD as explained previously.

2.2.2 The CG algorithm

The structure of the CG and RS algorithms are similar, in that both have an inner layer of iterations ‘inside’ an outer layer and a modified backfitting procedure within the inner algorithm (Figure 2.1). The main difference is that the CG uses expected cross derivatives of the log-likelihood with respect of each of the distribution parameters and therefore is able to jointly update all vectors of parameters. However, some distributions have expected cross derivatives near zero (that is, they are approximately information orthogonal), such as location and scale models and dispersion family models, for which the CG algorithm is not recommended. On the other hand, the CG algorithm is particularly useful when dealing with distributions that have highly correlated parameter estimators because for these the RS algorithm can be slower and converge to a local maximum of the log-likelihood instead of the global maximum.

The CG algorithm also uses an iterative working variable and iterative weights.

The difference is that the weights are formed by cross derivatives instead of double derivatives to the same parameter vector, so that the iterative working variable is given by

$$\mathbf{z}_k = \boldsymbol{\eta}_k + \mathbf{w}_{ks}^{-1} \circ \mathbf{u}_k \quad (2.15)$$

and the weights by

$$\mathbf{w}_{ks} = -\mathbf{f}_{ks} \circ \left(\frac{d\boldsymbol{\theta}_k}{d\boldsymbol{\eta}_k} \right) \circ \left(\frac{d\boldsymbol{\theta}_s}{d\boldsymbol{\eta}_s} \right) \quad (2.16)$$

with

$$\mathbf{f}_k = \begin{cases} -E \left(\frac{\partial^2 l}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_s} \right) \\ - \left(\frac{\partial l}{\partial \boldsymbol{\theta}_k} \right) \circ \left(\frac{\partial l}{\partial \boldsymbol{\theta}_s} \right), \end{cases} \quad (2.17)$$

where the weight vectors $\mathbf{w}_{ks}$ are calculated for all combination of parameter indexes k, s (which are 10 for a four parameter distribution). Defining and updating these iterative vectors is the main function of the outer layer/iteration of the CG algorithm, which also does a GD convergence check at the end of each complete iteration as does the RS' outer iteration.

Inside the CG's inner iteration is where a model is properly fitted for each parameter using the same modified backfitting algorithm explained in the previous section. For the fitting within the CG algorithm, however, the working variable defined in (2.15) is adjusted by different vectors defined for each parameter as

$$\begin{aligned} \mu : \mathbf{z}_1^a &= -\mathbf{w}_{11}^{-1} \circ [\mathbf{w}_{12} \circ (\boldsymbol{\eta}_2 - \boldsymbol{\eta}_2^o) + \mathbf{w}_{13} \circ (\boldsymbol{\eta}_3 - \boldsymbol{\eta}_3^o) + \mathbf{w}_{14} \circ (\boldsymbol{\eta}_4 - \boldsymbol{\eta}_4^o)] \\ \sigma : \mathbf{z}_2^a &= -\mathbf{w}_{22}^{-1} \circ [\mathbf{w}_{12} \circ (\boldsymbol{\eta}_1 - \boldsymbol{\eta}_1^o) + \mathbf{w}_{23} \circ (\boldsymbol{\eta}_3 - \boldsymbol{\eta}_3^o) + \mathbf{w}_{24} \circ (\boldsymbol{\eta}_4 - \boldsymbol{\eta}_4^o)] \\ \nu : \mathbf{z}_3^a &= -\mathbf{w}_{33}^{-1} \circ [\mathbf{w}_{13} \circ (\boldsymbol{\eta}_1 - \boldsymbol{\eta}_1^o) + \mathbf{w}_{23} \circ (\boldsymbol{\eta}_2 - \boldsymbol{\eta}_2^o) + \mathbf{w}_{34} \circ (\boldsymbol{\eta}_4 - \boldsymbol{\eta}_4^o)] \\ \tau : \mathbf{z}_4^a &= -\mathbf{w}_{44}^{-1} \circ [\mathbf{w}_{14} \circ (\boldsymbol{\eta}_1 - \boldsymbol{\eta}_1^o) + \mathbf{w}_{24} \circ (\boldsymbol{\eta}_2 - \boldsymbol{\eta}_2^o) + \mathbf{w}_{34} \circ (\boldsymbol{\eta}_3 - \boldsymbol{\eta}_3^o)]. \end{aligned} \quad (2.18)$$

These vectors $\mathbf{z}_k^a$ for $k = 1, 2, 3, 4$ are weighted combinations of differences in the corresponding predictors multiplied by the inverse of the respective parameter weight vectors, which are added to the working variable $\mathbf{z}_k$ in (2.15) to create a new working variable $\mathbf{z}_k'$ for each parameter, that is,

$$\mathbf{z}_k' = \mathbf{z}_k + \mathbf{z}_k^a. \quad (2.19)$$

With these new working variables, the inner algorithm calls the modified backfitting algorithm to fit a model for each of the distribution parameters in turn and updates the respective predictor vector. In other words, it starts by fitting a model for $\boldsymbol{\mu}$ using $\mathbf{z}_1' = \mathbf{z}_1 + \mathbf{z}_1^a$ and updating $\boldsymbol{\eta}_1$, then for $\boldsymbol{\sigma}$ and updates $\boldsymbol{\eta}_2$, etc; up to the model for $\boldsymbol{\tau}$, applying the modified backfitting process for each fit. Once all the predictor vectors are updated, the inner algorithm updates the global deviance value and checks if it has converged. If not, the inner algorithm restarts its cycle, otherwise the process goes back to the outer layer and does another GD convergence check to decide whether to finish the

CG algorithm or to start a whole new iteration.

2.2.3 Estimating the hyperparameters λ

For the fitting of multiple smoothing terms within a GAMLSS, there are an external and an internal method (or a global and a local method, respectively) to the GAMLSS fitting algorithm. RIGBY and STASINOPOULOS (2014) argue that the internal one is much faster and often produces similar results to the external. The internal method fits each of the hyperparameters sequentially in cycles until convergence within the RS algorithm. This method can also be seen as a generalisation of the procedure given by LEE *et al.* (2006), since it is a penalized quasi-likelihood (PQL) method applied locally to each parameter.

Apart from the internal/external methods, there are a couple of strategies to be chosen from for the estimation of the smoothing parameters. The current strategies available are (restricted) maximum likelihood (ML/Restricted ML), generalized Akaike information criterion (GAIC) and (generalized) cross validation (CV/GCV). Information about the estimation procedures using each of them is given in section 3.4 of STASINOPOULOS *et al.* (2017). The next section is about one of the additive terms that can be used within GAMLSSs and its connection to random effects' models.

2.3 Spline smoothing additive terms

Generally speaking, a spline smoother is approximately a convolution or a weighted moving average smoothing where the width of convolution varies across the data sample. As in other non-parametric methods, spline smoothing uses a roughness penalty to measure the rapid local fluctuation of a curve. One of the most convenient of such penalties is given by the integrated square of the second derivative of the fitted curve. For this penalty, REINSCH (1967) showed that the curve that minimizes the penalized residual sum of squares is a cubic polynomial in each interval of (ordered) design points (t_i, t_{i+1}) with the first two derivative continuous at each of the design points t_i and is linear at $(-\infty, t_1)$ and (t_n, ∞) . One of the interesting consequences of these properties is that the four coefficients of each interval's cubic polynomial are found by solving a band-limited linear system of size n (the sample size), which is a very quick and computationally cheap process (DE BOOR, 1977).

Within a GAMLSS's formula, either univariate smoothers (non-linear main effects of one explanatory variable) or multivariate smoothers (non-linear interaction effects of more than one explanatory variable) can be included. For each of these two categories, smoothers can be further split according to the use of quadratic penalties on the fitted smooth model parameters (penalized ones).

One of the univariate penalized smoothers that uses quadratic penalties is a P-spline. In the `gamlss` package in R, the `pb()` function stands for Penalized B-splines and is an implementation of the P-splines defined by EILERS and MARX (1996). These are piecewise polynomials defined by a combination of B-spline basis functions in the explanatory variables, where the coefficients of the basis functions are penalized to guarantee sufficient smoothness. Among the attractive properties of P-splines can be cited:

1. P-splines show no boundary effects. Many kernel estimators spread their fitted curves outside the domain of the data, usually bending toward zero. A P-spline density smoother is not troubled by that, but it is important to specify the domain of the data correctly.
2. P-splines fit polynomial data exactly. For any polynomial relationship between the explanatory and response variables of order k , a P-spline of order $k + 1$ or higher will exactly fit the data. This follows from the fact that P-splines are composed of B-splines, which themselves have this characteristic (DE BOOR, 1977).
3. For strong smoothing (large values of λ), the limit of a P-spline fit is a polynomial.

Univariate penalized smoothers such as P-splines might be thought as the solution of minimising a quantity Q with respect to the smoothing vector of parameters $\boldsymbol{\gamma}$ in

$$Q = (\mathbf{y} - \mathbf{Z}\boldsymbol{\gamma})^T \mathbf{W}(\mathbf{y} - \mathbf{Z}\boldsymbol{\gamma}) + \lambda \boldsymbol{\gamma}^T \mathbf{G}\boldsymbol{\gamma}, \quad (2.20)$$

which has solution

$$\hat{\boldsymbol{\gamma}} = (\mathbf{Z}^T \mathbf{W} \mathbf{Z} + \lambda \mathbf{G})^{-1} \mathbf{Z}^T \mathbf{W} \mathbf{y}. \quad (2.21)$$

Other types of smoothers are produced by modifying the basis $\mathbf{Z}'$ s and the penalty matrices $\mathbf{G}'$ s. A comprehensive description of all types of smoothers supported by GAMLSSs is given in chapter 9 of STASINOPOULOS *et al.* (2017).

Another interesting property of P-splines is their connection to random effects. To make this connection in details a knowledge of extended or hierarquical likelihood is required but will not be entered here. Instead, a quick overview can be gained by considering the model

$$\mathbf{y} = \mathbf{Z}\boldsymbol{\gamma} + \mathbf{e}, \quad (2.22)$$

where

$$\begin{aligned} \mathbf{e} &\sim N(\mathbf{0}, \sigma_e^2 \mathbf{W}^{-1}) \\ \boldsymbol{\gamma} &\sim N(\mathbf{0}, \sigma_b^2 \mathbf{G}^{-1}). \end{aligned} \quad (2.23)$$

By manipulating the extended/hierarquical log-likelihood of this model (which is proportional to the posterior density of $\boldsymbol{\gamma}$) and using $\lambda = \sigma_e^2 / \sigma_b^2$ one would arrive at the equation for Q in (2.20). This means that the penalized least squares estimator of $\boldsymbol{\gamma}$ obtained by (2.21) minimizing (2.20) is the same as the posterior mode or best

linear unbiased estimator (this last one is also known as maximum a posteriori estimator) that would be obtained with model (2.22). Furthermore, this shows that the smoothing parameter is a ratio between two variances, the error variance divided by the random effect variance. These two variances (and consequently the smoothing parameter) can be estimated by maximum likelihood or restricted maximum likelihood, as explained in section 2.2 of PINHEIRO and BATES (2000).

2.4 Model selection

The procedure of selecting a GAMLSS for a specific set of data consists of comparing models with different combinations of the distribution for the response variable, sets of link functions, predictor terms for each predictor and sets of hyperparameters.

i) Selecting a distribution for the response variable

The distribution selection can be made by comparing models with different distributions using the generalized Akaike criterion (GAIC) (AKAIKE, 1983). This criterion penalizes overfitting since it is defined as the sum of the fitted deviance plus a fixed penalty for each effective degree of freedom (HASTIE and TIBSHIRANI, 1990) used in the model. STASINOPOULOS *et al.* (2017) suggest that a value of the penalty between 2.5 and 4 works well, but the sensitivity of the fitted model can be investigated by changing the penalty weights. Another strategie is to compare plots of fitted distributions where constants were fitted to each of the parameters (using function `gamlss::histDist()`). Once the distribution is selected (the one with the smallest GAIC), inadequacies in the model globally or within ranges of the explanatory variables (one or two at time) can be detected with worm plots (function `gamlss::wp()` in R), which are detrended QQ-plots introduced by VAN BUUREN and FREDRIKS (2001).

ii) Selecting suitable link functions

The appropriate link function to be used depends on the range of the parameters of the response variable distribution. For example, a normal response distribution has its mean (μ) varying over the real line and deviation (σ) greater or equal to zero, so an identity for μ and a log link for σ would be sensible choices. However, there might be situations where more than one link function can be suitable for a specific parameter. In such cases, different link functions can be compared directly using the global deviance and the best one will result in the lowest deviance.

iii) Establishing the best parameter predictors

There are several strategies that can be employed to find the best predictor for a parameter of the response variable distribution. The usual forward, backward and stepwise procedures are appealing and popularly used. Within the `gamlss` package, the function `stepGAICall.A()` uses such a procedure and the generalized Akaike as the criterion for selection. For a fixed response distribution with set of parameters $(\mu, \sigma, \nu, \tau)^T$,

this function's strategie can be described in the following sequential steps:

1. Uses a forward GAIC selection procedure to select an appropriate model for μ , with σ , ν and τ fitted as constants;
2. Given the model for μ obtained in (1) and for ν and τ fitted as constants, use a forward selection procedure to select an appropriate model for σ ;
3. Given the models for μ and σ obtained in (1) and (2) respectively and with τ fitted as constant, use a forward selection procedure to select an appropriate model for ν ;
4. Given the models for μ , σ and ν obtained in (1), (2) and (3) respectively, use a forward selection procedure to select an appropriate model for τ ;
5. Given the models for μ , σ and τ obtained in (1), (2) and (4) respectively, use a backward selection procedure to select appropriate model for ν ;
6. Given the models for μ , ν and τ obtained in (1), (5) and (4) respectively, use a backward selection procedure to select appropriate model for σ ;
7. Given the models for σ , ν and τ obtained in (6), (5) and (4) respectively, use a backward selection procedure to select an appropriate model for μ and then stop.

The final model will contain different subsets of terms (not necessarily the same) for each of the parameters (μ , σ , ν and τ in the example). One thing to take into account is that different terms for the same predictor may affect the associated parameter distinctly. This is the case specially with semiparametric models, where a single continous variable has a linear effect while a smooth function of it affects the correspondent parameter non-linearly, for example. Furthermore, basing this selection entirely on a stepwise procedure is usually not the best option; it is always recommended to take into account the researcher or an expert's opinion.

iv) Fitting the smoothing parameters

Finally, the smoothing parameters can be either fixed or estimated from the data. The usual procedure when fixing the smoothing parameters is to fix the effective degrees of freedom for smoothing (HASTIE and TIBSHIRANI (1990)). As commented in section 2.4, smoothing parameters are estimated by GCV, GAIC or ML methods in procedures that is internal or external to the GAMLSS algorithm, explained in details in section 3.4 of STASINOPOULOS *et al.* (2017).

2.5 Diagnosis

One of the most classical tools for diagnosing regression models is residual analysis. However, differently from ordinary least squares (OLS) regression that uses raw residuals and from GLMs that use deviance or Pearson residuals, GAMLSSs use normalized (randomized) quantile residuals introduced by DUNN and SMYTH (1996). A

normalized quantile residual can be defined as

$$\hat{r}_i = \Phi^{-1}(\hat{u}_i), \quad (2.24)$$

where $\Phi^{-1}(\cdot)$ is the inverse cumulative distribution function (cdf) of a standard normal distribution and $\hat{u}_i$ is a quantile residual. Consider y as an observation from a continuous response variable Y and $u = F(y|\boldsymbol{\theta})$ and $\hat{u} = F(y|\hat{\boldsymbol{\theta}})$ as the model and fitted cumulative distribution functions (cdf's) respectively. Let U be a random variable defined as $U = F_Y(Y) = P(Y \leq y)$, i.e., equal to the cdf of Y . Then U has cdf $F_U(u)$ given by

$$F_U(u) = P(U \leq u) = P(F_Y(Y) \leq u) = P(Y \leq F_Y^{-1}(u)) = F_Y(F_Y^{-1}(u)) = u, \quad (2.25)$$

so that

$$F_U(u) = \begin{cases} 0 & u < 0 \\ u & 0 \leq u \leq 1 \\ 1 & u > 1, \end{cases} \quad (2.26)$$

which is the cdf of a $\text{Uniform}(0, 1)$ random variable. This is known as the probability integral transform and it guarantees that, if the model is correctly specified, the quantile residual u has the uniform distribution between 0 and 1 and consequently r follows a standard normal distribution. For the discrete case a generalisation of the same idea is made and a randomisation of the quantile residuals $\hat{u}$ is required because of the granularity resulted from the mass points (details are given in section 3 of DUNN and SMYTH (1996)). The normalized quantile residuals are preferred for GAMLSSs because of their feature to follow a standard normal distribution when the assumed model is correct, whatever the distribution of the response variable.

The evaluation of the adequacy of the quantile residuals to a standard normal distribution is made by the usual procedures of normality testing and graphical evaluation. For example, within **R**, the residuals of a fitted **gamlss** object can be assessed using the function **plot()** (analogue to **lm** and **glm** objects), which produces plots of the (quantile) residuals against fitted values of the μ parameter, against an index or specific covariate, a kernel density estimate and the traditional Q-Q normal plot (that is, it produces a 2 by 2 matrix of plots). Apart from this function, other **R** functions such as the Q statistics function **Q.stats()**, the function **rqres.plot()** (which does repeated randomization of the residuals for discrete response variables) and the worm plot function **wpl()** can be used for the residual analysis of a GAMLSSs. The interpretations of the most common shapes obtained in a wormplot are that a vertical shift, a slope, a parabola and a S shape indicate a misfit in the mean, variance, skewness and kurtosis of the residuals, respectively.

2.6 Methods for comparing models

At the end of the model selection procedure one may end up with two or more models satisfactorily well fitted to choose from. To evaluate the fitting of two nested GAMLSSs M_0 and M_1 (M_0 is a subclass of M_1) with global deviances GD_0 and GD_1 and degrees of freedom df_0 and df_1 respectively, the following (generalized likelihood ratio) test statistic can be used:

$$\Lambda = GD_0 - GD_1. \quad (2.27)$$

Under the hypothesis that the correct model is M_0 , Λ has an asymptotic X^2 -square distribution with degrees of freedom $df = df_0 - df_1$ (given that the regularity conditions are satisfied), so that M_1 would be preferred over M_0 whenever Λ proves to be higher than the quantile of the Chi-Square distribution with df degrees of freedom and a chosen level of significance. To compare non-nested GAMLSSs the available procedures/criteria that can be used are cross validation, splitted data evaluation (for large datasets) and also the GAIC.

References

- AKAIKE, H., 1983 Information measures and model selection. *Bulletin of the International Statistical Institute* **50**: 277–291.
- CLEVELAND, S. and S. J. DEVLIN, 1988 Locally Weighted Regression : An Approach to Regression Analysis by Local Fitting. *Journal of the American Statistical Association* **83**: 596–610.
- COLE, T. J. and P. J. GREEN, 1992 Smoothing Reference Centile Curves - the Lms Method and Penalized Likelihood. *Statistics in Medicine* **11**: 1305–1319.
- DE BOOR, C., 1977 Package for calculating B-Splines. *Journal on Numerical Analysis* **36**: 338–347.
- DUNN, P. K. and G. K. SMYTH, 1996 Randomized Quantile Residuals. *Journal of Computational and Graphical Statistics* **5**: 236–244.
- EILERS, P. H. C. and B. D. MARX, 1996 Flexible smoothing with B -splines and penalties. *Statistical Science* **11**: 89–121.
- HASTIE, T. J. and R. J. TIBSHIRANI, 1990 *Generalized Additive Models*. CRC Press.
- LEE, Y., J. A. NELDER, and Y. PAWITAN, 2006 *Generalized Linear Models with Random Effects: Unified Analysis via H-likelihood*. CRC Press.
- PINHEIRO, J. C. and D. M. BATES, 2000 *Mixed-Effects Models in S and S-PLUS*. Springer.

- REINSCH, C. H., 1967 Smoothing by spline functions. *Numerische Mathematik* **16**: 177–183.
- RIGBY, R. A. and D. M. STASINOPOULOS, 2014 Automatic smoothing parameter selection in GAMLSS with an application to centile estimation. *Statistical methods in medical research* **23**: 318–332.
- SPEED, T. P., 2007 Comment on 'That BLUP Is a Good Thing : The Estimation of Random Effects' (by G. K. Robinson). *Statistical Science* **6**: 15–32.
- STASINOPOULOS, M., R. RIGBY, G. HELLER, V. VOUDOURIS, and F. D. BASTIANI, 2017 *Flexible Regression and Smoothing The GAMLSS packages in R*. CRC Press.
- VAN BUUREN, S. and M. FREDRIKS, 2001 Worm plot: A simple diagnostic device for modelling growth reference curves. *Statistics in Medicine* **20**: 1259–1277.
- WAHBA, G., 1990 Spline model for observational data. *Society for industrial and applied mathematics* .
- WAND, M. P., 2003 Smoothing and mixed models. *Computational Statistics* **18**: 223–249.
- WOOD, S. N., 2006 *Generalized Additive Models: An Introduction with R*. CRC Press.

3 MODELING DATA WITH ZERO INFLATION AND OVERDISPERSION USING GAMLSS

Abstract

Count data with high number of zeros are found in many areas, specially in biology. Statistical models to analyze such data started to be developed in the 80s and are still a topic of active research. This type of models usually assume a response distribution that belongs to the exponential family of distributions and therefore the analysis is performed under the generalized linear models' framework. However, the generalized additive models for location, scale and shape (GAMLSSs) represent a more general class of univariate models that can also be used for zero inflated modeling. Therefore, the analysis of a data set with excess of zeros and overdispersion already analyzed in the literature is described using GAMLSSs.

Keywords: Zero inflated models; GAMLSSs; *Trajan* data set

3.1 Introduction

When analysing count data, it is common to encounter a higher number of zero responses than predicted by the Poisson distribution. In a biological context, this is often a result of individuals that respond in two distinct ways to the effect of interest. While one group of individuals is not able to respond and these will always give zeros (also known as structural zeros), another might respond or not (whose occasional zeros are known as sampling zeros). This situation can be described by a two component mixture model with zero-inflation.

The literature of zero-inflated models dates back to the 80s with MULLAHY (1986), who was the first to define a two-part model. A couple of years later, LAMBERT (1992) introduced the zero-inflated Poisson (ZIP) model with an application to manufacturing defects. These ideas originated a whole class of new models such as the zero-inflated binomial (ZIB) model, the zero-inflated negative binomial (ZINB) model (GREENE, 1994), zero altered (ZA) models (HEILBRON, 1994), mixed versions of the ZIP and ZIB (HALL, 2000), etc.

RIDOUT *et al.* (1998) make a review of the literature on ZIP and other closely related zero-inflated models for unbounded counts and present a comparison of these models for an horticulture experiment (*Trajan* data) using a generalized linear model (GLM) (NELDER and WEDDERBURN, 1972) approach. On that occasion, they found that a ZINB model provided the best fit over traditional Poisson, negative binomial and ZIP models, comparing different predictors for each distribution parameter.

This work describes the modeling of the *Trajan* data using ZIP, ZINB, ZIPIG (zero-inflated Poisson inverse Gaussian) and ZIBNB (zero-inflated beta negative binomial) models using the generalized additive models for location, scale and shape (GAMLSSs) (RIGBY and STASINOPOULOS, 2005) framework instead of the GLM approach these authors used. Specific GAMLSSs tools are used for model comparison and model adequacy checking.

3.2 Material and methods

The data recorded in the horticulture experiment were the number of roots produced by 270 micropropagated shoots of the columnar apple cultivar *Trajan*. All shoots were kept under identical conditions apart from concentrations of a hormone and photoperiod durations inside growth cabinets. Two growth cabinets were used, one with 8 and the other with 16 hours of photoperiod. Jars with shoots were placed at random in one of the two cabinets. In addition, four concentrations of the cytokinin BAP hormone (2.2, 4.4, 8.8 and $17.6\mu\text{M}$) were used, so that the experiment had a 4×2 factorial design of treatments. The objective of the study was to evaluate the effects of the two factors (concentration of cytokinin and photoperiod duration) on the number of roots produced. The full description of the experiment can be found in MARIN *et al.* (1993).

The key feature of this data set is that almost half of the shoots under the 16 hours' photoperiod did not root, as can be seen from Table 3.1 and Figure 3.1. This means that a zero inflated model that accounts for this excesses of zeros is needed. RIDOUT *et al.* (1998) pointed out that the data are also overdispersed, which can also be seen from the two bottom lines of mean and variance in Table 3.1. The variances of the number of roots generated in the cabinet with 16 hours of photoperiod were more than 3 times higher than the corresponding means for all BAP concentrations, while for the other treatment combinations this ratio is much smaller.

In this paper, the *Trajan* data set is analyzed using GAMLSSs. The GAMLSSs allow any distribution that has its probability density function and first, second and cross derivatives with respect to each of the parameters computable to be used and allow all parameters (up to four) this distribution to be modeled explicitly with predictors. This type of regression models generalize not only the GLMs but also the generalized additive models (GAMs) of HASTIE and TIBSHIRANI (1990) in the sense that parameters are linked to predictors through known monotonic link functions which can include a wide range of linear/non-linear as well as random effect/nonparametric terms. In this application, though, only linear functions of the predictors were used, so that the underlying model is a parametric GAMLSS.

As a general definition, a parametric GAMLSS assumes an independent random variable $Y_i, i = 1, \dots, n$ following a probability distribution $\mathbf{D}_{\boldsymbol{\theta}}$. The parameters

Table 3.1. Frequency table of the *Trajan* data set.

BAP (μM)	Photoperiod							
	8 hours				16 hours			
	2.2	4.4	8.8	17.6	2.2	4.4	8.8	17.6
Number of roots								
0	0	0	0	2	15	16	12	19
1	3	0	0	0	0	2	3	2
2	2	3	1	0	2	1	2	2
3	3	0	2	2	2	1	1	4
4	6	1	4	2	1	2	2	3
5	3	0	4	5	2	1	2	1
6	2	3	4	5	1	2	3	4
7	2	7	4	4	0	0	1	3
8	3	3	7	8	1	1	0	0
9	1	5	5	3	3	0	2	2
10	2	3	4	4	1	3	0	0
11	1	4	1	4	1	0	1	0
12	0	0	2	0	1	1	1	0
> 12	13, 17	13	14, 14	14				
Number of shoots	30	30	40	40	30	30	30	40
Mean	5.8	7.8	7.5	7.2	3.3	2.7	3.1	2.5
Variance	14.1	7.6	8.5	8.8	16.6	14.8	13.5	8.5

Figure 3.1. Barplots of the number of roots per each photoperiod duration.

$\theta=(\theta_1, \theta_2, \theta_3, \theta_4)'=(\mu, \sigma, \nu, \tau)'$ (up to four) of this distribution are linked to linear predictors by link functions the same way as in generalized models. However, in the GAMLSS framework link functions and predictors can be defined for all distribution parameters, not only for the mean of the distribution (the first parameter). For example, in this analysis the mean number of roots generated (represented by the μ parameter) could be modeled by the main effects of photoperiod and BAP concentration factors and the zero inflation parameter (represented by a second parameter, σ) could be allowed to depend on photoperiod duration. The correspondent model would be represented as

$$\begin{aligned} \mathbf{Y} &\sim \mathbf{D}(\boldsymbol{\mu}, \boldsymbol{\sigma}) \\ g_1(\boldsymbol{\mu}) &= \mathbf{X}_1\boldsymbol{\rho} + \mathbf{X}_2\boldsymbol{\beta} \\ g_2(\boldsymbol{\sigma}) &= \mathbf{X}_1\boldsymbol{\rho}, \end{aligned} \tag{3.1}$$

where $\mathbf{X}_1$ and $\mathbf{X}_2$ denote the design matrices of the photoperiod durations $\boldsymbol{\rho}$ and BAP concentrations $\boldsymbol{\beta}$, respectively.

Distribution parameters are conventionally denoted in order by μ , σ , ν and τ in the GAMLSS framework. For most of them, the first parameter μ is the location parameter associated with the mean of the distribution and σ is the scale parameter associated with the variance. The remaining parameters are commonly related to the shape of the distribution. The GAMLSSs allow all response distribution parameters to be estimated (by maximum likelihood), whereas GLM's only allows the first two parameters to be modeled as functions of explanatory variables (the second implicitly).

The selection procedure of a parametric GAMLSS consists of comparing models with different combinations of distributions for the response variable and predictor terms for each parameter predictor. This means that a suitable zero inflated distribution will have to be selected to build a GAMLSS for the number of roots in the *Trajan* micropropagation experiment. Zero adjusted models can also handle higher frequencies of zeros in the response, but since they are two-part models (also known in economy as *hurdle* models) they do not have a good interpretation for this biological process. RIDOUT *et al.* (1998) used the Poisson, negative binomial, zero-inflated Poisson and zero-inflated negative binomial models. They used the first two of them more as a base for comparison, since the excess of zeros was clear. The zero-inflated Poisson accounted for that, but the authors found that the chosen model had to handle some overdispersion as well, so that the ZINB model gave the best fit in their analysis.

To aid the procedure of choosing a suitable population distribution for a GAMLSS, the `gamlss` package in R provides the `histDist()` function. This function produces an histogram of the data set and overimposes a model fit using a response distribution to be tested. It gives just an initial idea of the distribution suitability to the data set, since the fit generated is entirely based on the response variable's histogram, not taking into account any covariate contribution. Figure 3.2 shows the plots produced with six distri-

butions using the `histDist()` function for the *Trajan* data set. The distribution fits are represented by the vertical lines with open circles at the top. It can be seen that both the Poisson and negative binomial models (PO and NB) do not account for the zero inflation, even though the second adjusts better for higher root counts. Between the ZIP, ZINB, ZIPIG and ZIBNB the only visible difference is that the ZIP superestimates the number of roots a little more than the others, but all four appear to fit the data reasonably well. Therefore, these four represent good candidates to build a GAMLSS for the roots data set and are better described in what follows.

The ZIP distribution used in this analysis is a reparameterized version of the original ZIP of LAMBERT (1992) where the first parameter μ is the mean of the distribution. This facilitates the model's interpretation. Analogous the original ZIP, this modified version of the ZIP assumes that the response variable has a high frequency of zeros (represented by the zero inflation parameter σ) and follows a Poisson process with constant mean μ . Its probability mass function (pmf) is given by

Figure 3.2. Histograms of the number of roots with selected fitted distributions.

$$P(Y = y|\mu, \sigma) = \begin{cases} \sigma + (1 - \sigma)e^{-\left(\frac{\mu}{1-\sigma}\right)}, & \text{if } y = 0 \\ (1 - \sigma)\frac{\mu^y}{y!(1-\sigma)^y}e^{-\left(\frac{\mu}{1-\sigma}\right)}, & \text{if } y = 1, 2, \dots \end{cases} \quad (3.2)$$

The ZINB distribution is similar in form to the ZIP, but it inflates the probability of a zero count by one parameter (ν) and has another (σ) to adjust the variance independently of the mean. The pmf for the ZINB distribution is equal to

$$P(Y = y|\mu, \sigma, \nu) = \begin{cases} \nu + (1 - \nu)P(Y = y|\mu, \sigma), & \text{if } y = 0 \\ (1 - \nu)P(Y = y|\mu, \sigma), & \text{if } y = 1, 2, \dots, \end{cases} \quad (3.3)$$

where

$$P(Y = y|\mu, \sigma) = \frac{\Gamma(y + \frac{1}{\sigma})}{\Gamma(\frac{1}{\sigma})\Gamma(y + 1)} \left(\frac{\sigma\mu}{1 + \sigma\mu} \right)^y \left(\frac{1}{1 + \sigma\mu} \right)^{\frac{1}{\sigma}} \quad (3.4)$$

for $y = 0, 1, 2, \dots$, $\mu \geq 0$, $\sigma \geq 0$ and $0 \leq \nu \leq 1$. In fact, equation (3.4) is (one of the parameterizations of) the probability mass function of the negative binomial distribution.

The Poisson inverse Gaussian is another distribution that has potential for modeling highly dispersed count data due to its flexibility. It is a continuously mixed Poisson distribution that allows for even higher skewness (longer upper tail) than the negative binomial. The zero inflated version of it has pmf given by

$$P(Y = y|\mu, \sigma, \nu) = \begin{cases} \nu + (1 - \nu)P(Y = y|\mu, \sigma), & \text{if } y = 0 \\ (1 - \nu)P(Y = y|\mu, \sigma), & \text{if } y = 1, 2, \dots, \end{cases} \quad (3.5)$$

where

$$P(Y = y|\mu, \sigma) = \frac{\mu^y e^{1/\sigma} K_{y-1/2}(\alpha)}{(\alpha\sigma)^y y!} \left(\frac{2\alpha}{\pi} \right)^{\frac{1}{2}}. \quad (3.6)$$

In (3.6), $\alpha^2 = \frac{1}{\sigma^2} + \frac{2\mu}{\sigma}$, $K_\lambda(t) = \frac{1}{2} \int_0^\infty x^{\lambda-1} \exp[-\frac{1}{2}t(x + x^{-1})]dx$ (which is known as the modified Bessel function of third kind) and both (3.5) and (3.6) are valid only for $y = 0, 1, 2, \dots$, $\mu \geq 0$ and $\sigma \geq 0$.

Finally, the pmf of the ZIBNB distribution is

$$P(Y = y|\mu, \sigma, \nu) = \begin{cases} \tau + (1 - \nu)P(Y = y|\mu, \sigma, \nu), & \text{if } y = 0 \\ (1 - \tau)P(Y = y|\mu, \sigma, \nu), & \text{if } y = 1, 2, \dots, \end{cases} \quad (3.7)$$

where

$$P(Y = y|\mu, \sigma, \nu) = \frac{\Gamma(y + \frac{1}{\nu})B(y + \frac{\mu\nu}{\sigma}, \frac{1}{\sigma} + \frac{1}{\nu} + 1)}{\Gamma(y + 1)B(\frac{\mu\nu}{\sigma}, \frac{1}{\sigma} + 1)} \quad (3.8)$$

is one form of the beta negative binomial function. The first parameter μ is the mean counts of the supposed independent Bernoulli trials, whose probability of success is assumed to follow a Beta(σ , ν) distribution, while τ is the zero inflation parameter. A comparison of model fits using the GAMLSS framework with these four distributions for the roots data set is made in the next section.

3.3 Results and discussion

An exploratory analysis for the *Trajan* data set is summarized graphically in Figures 3.3 and 3.4. From the boxplots by treatment shown in Figure 3.3, there is a positive effect of the 8h photoperiod duration on the number of roots, but only mild effects of the different concentration levels. The interaction plot in Figure 3.4 gives some evidence of interaction between the factors since the mean number of roots (represented by red and blue lines for each photoperiod duration) change in different directions when the BAP concentration increases from 2.2 to 8.8 μM . In addition, the frequency of a specific number of roots for each treatment combination can be seen by the size of the blue and red circles in this graph. For example, the big blue circles in the bottom highlight the high frequency of unrooted shoots in the 16h cabinet.

Figure 3.3. Boxplots of each treatment combination for the number of roots.

The stepwise procedure described in section 2.4 was used to find the best predictors for each distribution parameters using AIC criterion. Once the best predictors for each model were established, AIC and BIC model's criteria were computed and Table 3.3 shows the resulting comparison.

The stepwise procedure described in section 2.4 was used to find the best predictors for each distribution parameters using AIC criterion. Once the best predictors for each model were established, AIC and BIC model's criteria were computed and Table 3.2 shows the resulting comparison.

Figure 3.4. Interaction plot of the factors for the number of roots.

Table 3.2. AIC and BIC of selected models.

Model	AIC	BIC
ZIP	1350.961	1361.757
ZINB	1245.623	1267.214
ZIPIG	1259.782	1274.175
ZIBNB	1250.955	1272.545

As found by RIDOUT *et al.* (1998), the ZINB model is the one that results in the lowest AIC and BIC for the *Trajan* data set. However, the criteria values shown for the ZIP and ZINB model in Table 3.2 are lower than the values they found, because the deviance of GLMs and GAMLSSs are defined differently. The deviance of a GLM is defined as

$$D_{GLM} = -2 \log \left(\frac{\hat{L}_c}{\hat{L}_s} \right), \quad (3.9)$$

where $\hat{L}_c$ is the fitted likelihood of the current fitted model and $\hat{L}_s$ that of the saturated model (where in modeling μ a parameter is fitted for each observation); while the GAMLSS's deviance is defined as

$$D_{GAMLSS} = -2 \log(\hat{L}_c). \quad (3.10)$$

Therefore, a GLM is not directly comparable with a GAMLSS, although one could compare (using AIC/BIC) a GAMLSS where only the mean of the distribution is

modeled with a linear predictor and the second parameter estimated as a constant to one where both are allowed to depend on covariates. However, despite the fact that the response distribution for the best model found under GAMLSS was the same found under the analysis by RIDOUT *et al.* (1998), the predictors found were different as is discussed after the residual analysis.

To diagnose the fitted ZINB GAMLSS, worm plots and a classical Normal Q-Q plot of the normalized (randomized) quantile residuals are reproduced in Figures 3.5, 3.6 and 3.7, respectively.

Figure 3.5. Worm plot for ZINB model.

Figure 3.5 is a general worm plot of the ZINB model's residuals. In this case it is a single detrended QQ-plot that highlights possible departures from normality. Figure 3.6 shows one worm plot for each level of the factor photoperiod duration. This one highlights failures of the model within each level of the explanatory variable. From both Figures and also from the Normal Q-Q plot in Figure 3.7 it can be seen that the residuals are within the 95% confidence bands given by the traced parable curves and with residual shape lines roughly horizontal for the worm plots and residuals mostly on the Q-Q line for the normal Q-Q plot. Therefore, it is concluded that there is no visual departure from normality and the residuals are well distributed.

The fitted linear predictors found for each of the ZINB parameters are

$$\log(\hat{\mu}) = 1.9725 \times photo_8 + 1.6954 \times photo_{16}, \quad (3.11)$$

Figure 3.6. Worm plots for each photoperiod duration.

Figure 3.7. Normal Q-Q plot for ZINB model.

$$\log(\hat{\sigma}) = -3.1158 \times photo_8 - 1.8286 \times photo_{16} \quad (3.12)$$

and

$$\text{logit}(\hat{\nu}) = -4.3808 \times \text{photo}_8 - 0.1351 \times \text{photo}_{16}. \quad (3.13)$$

To facilitate the interpretation of these predictors, the `gamlss` package provides the function `term.plot()`. This function produces plots (shown in Figure 3.8) of parameter estimates (in the link function scale) for each covariate in the predictor of each parameter of the population distribution assumed. Point estimates are represented by the horizontal lines in Figure 3.8 and the shaded areas correspond to the estimates' standard errors.

Figure 3.8. Termplots for the parameters μ , σ and ν of the ZINB distribution fitted.

From the left termplot of Figure 3.8 and predictor (3.11) there is evidence of positive and different effects of the photoperiod durations in the mean number of roots, because both term estimates are positive and their standard errors do not intersect neither contain the zero value. By the same reason, the termplot for σ shows little evidence that the overdispersion is different for the two groups (since σ represents the overdispersion parameter of the NB distribution). This means that a model with only the intercept for the σ parameter would give a similar fit. Finally, from the termplot for photoperiod in the predictor of the zero-inflation parameter ν (on the right), it can be seen that the

16h photoperiod has a much higher effect than the lower duration, as expected once the zero-inflation was observed mainly in that group.

3.4 Conclusion

This work described the analysis of a zero-inflated count data set provenient from a real experiment. Using specific GAMLSS's tools, the best response distributions were chosen and predictors for the parameters were obtained. Worm plots facilitated the residual diagnosis in the model globally and within ranges of the explanatory variable photoperiod. The interpretation of the model's results was aided by termplots, which enhanced the effects of the covariate on each response distribution parameter. The best GAMLSS fitted for the number of roots was a ZINB model with the photoperiod duration as a significant factor for both the mean and the zero-inflation parameters. This analysis showed that the GAMLSSs are a competitive and flexible approach to model complex data structures, such as ones with excesses of zeros and overdispersion.

The results obtained in this analysis agree with what was found by RIDOUT *et al.* (1998) and support the conclusion of MARIN *et al.* (1993) that in *Trajan* there is little effect of BAP concentration. Marin also adds that this absense of hormone influence happens in other apple varieties as well, in contrast to conventional varieties where BAP has a strong deleterious effect.

References

- GREENE, W. H., 1994 Accounting for excess zeros and sample selection in Poisson and Negative Binomial regression models. NYU Working Paper No. EC-94-10 **9**: 265–265.
- HALL, D. B., 2000 Zero-Inflated Poisson and Binomial Regression with Random Effects : A Case Study. *Biometrics* **56**: 1030–1039.
- HASTIE, T. J. and R. J. TIBSHIRANI, 1990 *Generalized Additive Models*. CRC Press.
- HEILBRON, D. C., 1994 Zero-altered and other Regression Models for Count Data with Added Zeros. *Biometrical Journal* **36**: 531–547.
- LAMBERT, D., 1992 Zero-inflated Poisson regression, with an application to defects in manufacturing. *Technometrics* **34**: 1–14.
- MARIN, J. A., O. P. JONES, and H. W. C., 1993 Micropropagation of columnar apple trees. *Journal of Horticultural Science* **68**: 289–297.
- MULLAHY, J., 1986 Specification and testing of some modified count data models. *Journal of Econometrics* **33**: 341–365.

- NELDER, J. A. and R. W. M. WEDDERBURN, 1972 Generalized Linear Models. J. R. Statist. Soc. A. **135**: 370–384.
- RIDOUT, M., C. G. DEMETRIO, and J. HINDE, 1998 Models for count data with many zeros. International Biometric Conference pp. 179–192.
- RIGBY, R. A. and D. M. STASINOPOULOS, 2005 Generalized additive models for location, scale and shape,(with discussion). Journal of the Royal Statistical Society: Series C (Applied Statistics) **54**: 507–554.

4 ANALYSIS OF A LONGITUDINAL SPLIT-PLOT WITH SUBSAMPLING EXPERIMENT USING GAMLSSS

Abstract

The standard procedures for analysing multilevel or grouped data are by (non)linear and generalized mixed models. However, the generalized additive models for location, scale and shape (GAMLSSs) also allow different types of random effects to be included in the model formulation. Even though already popular in many areas of research, this type of models are not routinely used for mixed model purposes. Therefore, this paper describes the analysis of an experiment with plants' growth using GAMLSSs and compares it to a linear mixed model approach.

Keywords: linear mixed models; mixed GAMLSSs; grouped data; sweet corn growth

4.1 Introduction

Statistical analysis of experiments using linear mixed effects models date back at least to JACKSON (1939), who defined a model with one factor as having fixed effects and another as a sample from a normally distributed random variable. Since then, mixed modeling has become a major area of statistical research, including work on computation of maximum likelihood estimates, non-linear mixed effect models, missing data in mixed effects models, and Bayesian estimation of mixed effects models. Mixed models are applied in many disciplines where multiple correlated measurements are made on each unit of interest. A brief history of linear mixed models can be found in WEST *et al.* (2014).

Mixed models for non normal data were more recently developed, though. SCHALL (1991) was one of the first to present the estimation of generalized linear models with random effects. Usually referred to as generalized linear mixed models (GLMMs), this type of regression models were popularized by the works of BRESLOW and CLAYTON (1993), MCCULLOCH and NEUHAUS (2001) and BOLKER *et al.* (2009), to name a few.

Linear, nonlinear and generalized mixed models are currently the standard approaches for analysing experiments with grouped data. REES *et al.* (1999), BUCKLEY *et al.* (2003), SCHLENKER and ROBERTS (2009), STRATHE *et al.* (2010) and PAINE *et al.* (2012) are some examples. These types of models restrict the number of possible distributions to be assumed for the response variable to the ones belonging to the exponential family of distributions. Furthermore, they only allow the mean parameter of the response distribution to be modeled explicitly with functions of the available covariates.

The generalized additive models for location, scale and shape (GAMLSSs) introduced by RIGBY and STASINOPOULOS (2005) allow any computable probability distribution to be assumed for the response variable as well as to model all its parameters

(up to four). In fact, there are more than 80 distributions available in the current implementation of the `gamlss.dist` package in R and anyone can create a new one. While GAMLSSs are already well established in the literature for centile growth curve estimation (frequently used with this purpose by the WHO - World Health Organization) and have been gaining popularity in other areas as well (such as industrial (BARAJAS *et al.*, 2015), medical (PETTERLE and FORMIGUIERI, 2014), financial (GILCHRIST *et al.*, 2009), forestry (HUDSON *et al.*, 2009), etc), no applications where GAMLSSs were used for mixed modeling purposes have been found. Since the `gamlss::gamlss()` (this means R package::function within package) function allows an interface to be made with the well known `nlme::lme()` function, it can be used for repeated measurements, multilevel modeling, random intercept and slopes, etc, among other mixed model fit purposes. The present work aims to illustrate how this GAMLSSs' methodology can be used to analyze a multilevel sweet corn experiment with repeated measures comparing it to a classical mixed model approach.

4.2 Experiment description

The sweet corn experiment analyzed was carried out from March to May of 2015 in greenhouses in the Instituto Federal Goiano (IF Goiano) in Goiás State, Brazil. Water stress in sweet corn plants is difficult to monitor due to climatic events such as El Niño (LI *et al.*, 2011), leading to considerable losses. Therefore, this study aimed to test the hypothesis that potassium silicate applications could elicit the resistance of these plants, reducing the environmental stress effects caused by the lack of water. The experimental design consisted of a split-plot with subsampling. The area inside the greenhouse was divided in four blocks, four plots within each block, four subplots within each plot and fourteen sweet corn plants were planted in each subplot. From these fourteen plants, four were selected per subplot. The sweet corn plants represented the subsamples in each subplot. Four levels (15, 30, 45 and 60 kPa) of soil water tensions were randomized to plots and four doses of potassium silicate (0, 6, 12 and 24 L/ha) were randomized to subplots. The heights of the plants were measured (in the same four plants in each subplot at all times) at 30, 45, 60, 75 and 90 days after seeding. The objective of the experiment was to evaluate the relationship between sweet corn plant age and its development under induced water stress and leaf potassium silicate applications.

Initially, an exploratory analysis of the data set was conducted. Histograms of the height measurements at each time of observation are displayed in Figure 4.1. The histograms highlight that the height distributions are roughly symmetric at all times but with increasing variability. For a visual assessment of treatment effects, growth plots of the 256 plants measured in the experiment were produced and are displayed by treatment (combination of soil tension and silicate dose levels) in Figure 4.2. Each line on each

Figure 4.1. Histograms of height per time.

plot of this figure links the 5 height measurements (5 dots) of a plant under a specific treatment (displayed at the top of each plot). There does not seem to be large differences between plant's development across treatments, since on average the plants show similar growth patterns and reach around 300 centimeters of height at 90 days after seeding. Furthermore, the growth pattern reveals that the highest height increase occurs between 45 and 60 days after seeding. This nonlinear pattern can be better seen in Figure 4.3, which shows the mean growth profiles for each combination of soil tension and potassium silicate levels (16 treatment lines). In accordance with the Figure 4.2, this one shows that there are not strong treatment effects but there is a mild triple interaction effect of both factors and time, since there are some crossings of the mean treatment lines in the right side of the graph.

4.3 Analysis

A common procedure for the analysis of multilevel experiments is to define random effects for all terms justified by the randomization (PIEPHO *et al.* (2003), BRIEN and DEMETRIO (2009)). For a classical split-plot experiment, this would mean one random effect for mainplots and one for subplots, which corresponds to a two-level model in the notation of PINHEIRO and BATES (2000). Since in this sweet corn experiment there are also subsamples (the sweet corn plants) within each subplot, a 3rd level is characterized. Therefore, the random effects part of the model was formulated with a random effect for main plots, one for subplots and another for subsamples. The fixed part of the model contains the main effects of blocks (controlling a soil fertility gradient), soil tensions, potassium silicate doses, the interaction of the last two and an appropriate function to

Figure 4.2. Plant's profile growth per treatment.

model the effect of time, possibly interacting with the factors as well. A mixed model for this experiment is given by

$$y_{ijklm} = \mu + \beta_i + \alpha_j + \epsilon_{ij} + \lambda_k + \alpha\lambda_{jk} + \varepsilon_{ijk} + \xi_{ijkl} + f(\tau_m) + \zeta_{ijklm}, \quad (4.1)$$

where y_{ijklm} is the m -th height measurement of the l -th plant located at the i -th block, j -th plot and k -th subplot (i, j, k and l vary from 1 to 4 and m from 1 to 5); μ is a constant common to all observations; β_i is the fixed effect of block i ; α_j is the fixed effect of the j -th soil tension level; ϵ_{ij} is the error associated to plot ij ; λ_k is the effect of the k -th potassium silicate level; $\alpha\lambda_{jk}$ is the interaction of soil tension and potassium silicate at levels j and k , respectively; ε_{ijk} is the error associated with the subplot ijk ; ξ_{ijkl} is the error associated with the subsample (plant) $ijkl$; $f(\tau_m)$ is a function of time (days when measurements were taken) and ζ_{ijklm} is the 'residual error' associated with the m -th height measurement on plant $ijkl$.

The first three error terms in Equation (4.1) are assumed to be random variables with distributions $\epsilon_{ij} \sim \mathcal{N}(0, \sigma_P^2)$, $\varepsilon_{ijk} \sim \mathcal{N}(0, \sigma_{SP}^2)$ and $\xi_{ijkl} \sim \mathcal{N}(0, \sigma_{SS}^2)$. They are also

Figure 4.3. Average growth lines of each treatment combination.

supposed to be independent among themselves and independent of the residual errors, which are assumed to follow $\zeta_{ijklm} \sim \mathcal{N}(0, \sigma^2)$.

In matrix notation, this mixed model is written as

$$\mathbf{Y} = \mathbf{X}_G\boldsymbol{\mu} + \mathbf{X}_B\boldsymbol{\beta} + \mathbf{X}_A\boldsymbol{\alpha} + \mathbf{X}_L\boldsymbol{\lambda} + \mathbf{X}_{AL}\boldsymbol{\alpha}\boldsymbol{\lambda} + \mathbf{Z}_{SP}\boldsymbol{\varepsilon} + \mathbf{X}_T\boldsymbol{\tau} + \mathbf{Z}_P\boldsymbol{\epsilon} + \mathbf{Z}_{SS}\boldsymbol{\xi} + \boldsymbol{\zeta}, \quad (4.2)$$

where the $\mathbf{X}$'s and $\mathbf{Z}$'s correspond to design matrices for the fixed and random effects, respectively. The mean of the response vector is given by $E[\mathbf{Y}] = \mathbf{X}_G\boldsymbol{\mu} + \mathbf{X}_B\boldsymbol{\beta} + \mathbf{X}_A\boldsymbol{\alpha} + \mathbf{X}_L\boldsymbol{\lambda} + \mathbf{X}_{AL}\boldsymbol{\alpha}\boldsymbol{\lambda} + \mathbf{X}_T\boldsymbol{\tau}$ and the variance-covariance matrix by $\text{Var}[\mathbf{Y}] = \sigma_P^2\mathbf{Z}_P\mathbf{Z}_P' + \sigma_{SP}^2\mathbf{Z}_{SP}\mathbf{Z}_{SP}' + \sigma_{SS}^2\mathbf{Z}_{SS}\mathbf{Z}_{SS}' + \sigma^2\mathbf{I}_{256}$. The variance-covariance structure resulting from this matrix does not correspond to any known structure, therefore an unstructured or general symmetric positive-definite matrix was initially assumed for $\text{Var}[\mathbf{Y}]$ in the analysis.

Even though Figures 4.2 and 4.3 gave a visual idea of the time effect, the number of possible functions to be tested for $f(\tau_m)$ is high. Furthermore, from Figure 4.3 there is some evidence of an interaction effect between time and the treatment factors. In this analysis, linear polynomial functions were used to model the effect of time. Two analyses were performed separately using the `nlme::lme()` and `gamlss::gamlss()` functions. These strategies are described in the following sections.

4.3.1 Using function `nlme::lme()`

The `nlme::lme()` function is among the most popular ones to fit linear mixed-effects models in R. The function was implemented under the general framework of LINDSTROM and BATES (1988), but some of its features were originally documented by different authors. For example, the model formulation was first described in LAIRD and WARE (1982), the variance-covariance parameterizations were described in PINHEIRO and BATES (1996) and the use of the variance functions implemented in the function were first presented by DAVIDIAN and GILTINAN (1995). However, the most complete reference of the function and the `nlme` package as a whole is given by PINHEIRO and BATES (2000). The authors give an easy-going introduction to linear and nonlinear mixed models through analyses of real datasets using the `nlme` library, explaining the capabilities of the library's functions and theoretical aspects along the way.

This section describes the analysis of the sweet corn experiment using linear mixed effects models (LMMs) fitted by maximum likelihood using the `nlme::lme()` function. The first mixed model fitted to the sweet corn data differed from (4.1) only in that it had the triple interaction of time with potassium silicate and soil tension factors and the double interactions of time with each of the factors along with the main effect of time. Following the methodology described in PINHEIRO and BATES (2000) section 2.4, the significance of fixed effects' terms in the model was assessed by conditional F-tests using sequential sum of squares. Table 4.1 shows the F-tests results for the fixed terms in the model in the order they were included. The bottom line of this table shows that the triple interaction tested was highly significant. Since time is a quantitative covariate, this means that the plants' growth trend differed significantly using at least 2 different combinations of soil tension and potassium silicate levels, that is, there are significantly different growth trends in plants that received different treatments.

Table 4.1. F-tests of fixed effects terms in the classical LMM.

Term	Degrees of freedom	F-value	p-value
intercept	1	5,716.53	<.0001
block	3	3.20	0.0623
time	1	19,669.19	<.0001
tension	3	0.52	0.6716
silicate	3	0.18	0.9113
time:tension	3	3.05	0.0279
time:silicate	3	0.88	0.4510
tension:silicate	9	1.23	0.2749
time:tension:silicate	9	2.56	0.0065

A graph of the standardized residuals plotted against the fitted values of this model is shown in Figure 4.4. From this plot five groups of residuals are identifiable, corresponding to each moment of height measurement. At 30 days the plant's height

varied from 0 to approximately 40cm, at 45 days it varied from around 50 until 100cm and so on until the last measurements from around 220cm onwards. This plot also highlights at least 3 problems with the model fitted. Two of them are the size of the residuals (some of them are below -4) and the heterogeneity of variance among the groups of observations (days of measurement). The third is that the model did a very poor job at predicting plant heights below 100cm, specially the ones taken at 45 days. This happens because this linear mixed model is assuming a constant growth rate during the entire period of 90 days, which is not consistent with Figure 4.3. The growth rate between 30 and 45 days was definitely lower than between 45 and 60 days, which caused the model to overestimate the heights at 45 days resulting in the group of negative residuals in Figure 4.4.

Figure 4.4. Residuals versus fitted values graph for the classical linear mixed model.

To account for the different growth rates of the sweet corn plants, a quadratic and a cubic function of time were added to the fixed part of the model. Table 4.2 shows the related F-tests' results for these terms along with the triple interaction of the linear effect of time with both factors. It is concluded that both polynomial effects are needed in the model and their inclusion increased the contribution of the triple interaction to the model.

The correspondent graph of standardized residuals versus fitted values is given in Figure 4.5. One of the issues appears to be diagnosed, that is, the residuals related to the height measurements taken at 45 days after seeding are more equally distributed around zero. However, there are still many residuals with high absolute values and the increase

Table 4.2. F-tests for the triple interaction and polynomial functions of time in the classical LMM.

Term	Degrees of freedom	F-value	p-value
time:tension:silicate	9	3.750	0.0001
time ²	1	69.15	<.0001
time ³	1	180.53	<.0001

in the residuals' variability with time is evident.

Figure 4.5. Residuals versus fitted values graph for the polynomial linear mixed model.

Since the triple interaction of the linear effect of time with potassium silicate and soil tension levels is significant, one could think of testing the triple interactions of the factors with the quadratic and cubic effects of time as well. However, when these were included in the model neither of them was found to be significant using F-tests. Because the model with these two additional triple interactions is nested within the one with simple quadratic and cubic functions of time, another way of testing the contribution of the additional triple interactions is by means of a likelihood ratio test (LRT). The likelihood ratio test result between these two models shown in Table 4.3 also favours the simpler one. The increase in the logarithm of the likelihood is too little to compensate for the number of degrees of freedom these terms require to be estimated.

Therefore, the best model so far contains three nested levels of random effects (for plots, subplots and subsamples/plants), a fixed effect term of blocks, a triple interaction of soil tension and potassium silicate levels with the linear effect of time (and all

Table 4.3. Likelihood ratio test result of nested LMMs.

Model	Degrees of freedom	loglikelihood	LRT Statistic	p-value
M1	41	-5676.09		
M2	71	-5660.39	31.40	0.40

correspondent two-level interactions and main effects) and the quadratic and cubic effects of time. This model does not account for the heterogeneity of variances though, as can be seen in Figure 4.5. Even though there are three levels of random effects in the model, each of them accounts for the correlations among groups of observations defined by the physical arrangement of the experiment, which does not allow the model to estimate specific variances for groups of heights taken on the same day. Such a (more general) model can be obtained using a function to model the variance structure of the within-time errors using the time covariate. Following the notation of DAVIDIAN and GILTINAN (1995), the general variance function for the multilevel mixed model (4.1) is expressed as

$$\text{Var}(\zeta_{ijklm} | \epsilon_{ij}, \epsilon_{ijk}, \xi_{ijkl}) = \sigma^2 g^2(\mu_{ijklm}, \mathbf{v}_{ijklm}, \boldsymbol{\delta}), \quad (4.3)$$

where $\mu_{ijklm} = E[y_{ijklm} | \epsilon_{ij}, \epsilon_{ijk}, \xi_{ijkl}]$ (the expected plant height given the random effects or BLUP - best linear unbiased predictor), $\mathbf{v}_{ijklm}$ is a vector of variance covariates, $\boldsymbol{\delta}$ is the vector of variance parameters and $g(\cdot)$ is the variance function, assumed continuous in $\boldsymbol{\delta}$. The power variance function was used in this analysis because of its flexibility for modeling monotonic heteroscedasticity when the covariate used is bounded away from zero (which is the case with time). Using the time covariate, the power variance function is given by

$$\text{Var}(\zeta_{ijklm}) = \sigma^2 \text{time}_{ijklm}^{2\delta}. \quad (4.4)$$

Modifying the model with this variance function results in an heteroscedastic model that improves the fit over the previous homoscedastic model for the plants' growth data. This can be seen by the likelihood ratio test result in Table 4.4, which shows that there is a highly significant increase in the log-likelihood associated with the inclusion of the variance function. Furthermore, the residuals of the heteroscedastic model have now similar variances within days of observation, as shown in Figure 4.6. The estimated fixed effects changed very little by the use of the variance function.

Table 4.4. LRT result of the heteroscedastic model over the homoscedastic model.

Model	Degrees of freedom	loglikelihood	LRT Statistic	p-value
Homoscedastic	41	-5676.09		
Heteroscedastic	42	-5268.10	815.97	<.0001

The plants' heights predicted by the heteroscedastic model along with the observed ones are depicted in Figure 4.7 by treatment. The traced lines denote the growths

Figure 4.6. Graph of standardized residuals versus fitted values for the heteroscedastic mixed model.

predicted by the model and the solid are the observed ones. The predicted growths look reasonable since for most treatments they are in the middle of the plant's growth lines observed. A striking feature in this graph is the lack of variation among the predicted lines, though. Given that the model includes random intercepts for plots, subplots and plants a higher distinction among the predicted lines' intercepts would be expected.

A visual evaluation of the random intercepts predicted by the final model is presented in Figure 4.8. There does not seem to be large deviations from the normality assumption by the graphs. However, the magnitude of the estimated intercepts is in the scale of hundreds of millions of units, that is, pretty much zero. This explains why the predicted plant growth lines in Figure 4.7 are basically superimposed. Using likelihood ratio tests to test the contribution of one random effect at time using nested models all three gave significant contributions to the model, though.

4.3.2 Using function *gamlss::gamlss()*

The first implementation of the GAMLSS in the late 90's was written in GLIM4 (FRANCIS *et al.* (1993)), which is nowadays almost an extinct software. The first R implementation of the *gamlss* function occurred in the beginning of this century (AKANTZILIOU *et al.*, 2002). In 2006 the first of a series of R packages was released in CRAN (the Comprehensive R Archive Network), the *gamlss.dist*, that contained the initial

Figure 4.7. Observed and predicted heights with the final model.

distributions available to be used with the `gamlss()` function implemented at that time. Currently, there are eleven `gamlss()` packages available, each related to a specific aspect of this general type of regression models. This analysis was made with the base `gamlss` package version 5.0-2 using probability distributions available in the `gamlss.dist` package of the same version.

A search on Google Scholar on 30th of September, 2017 for the word “gamlss” retrieved 2.360 articles. GAMLSSs have already been applied to many scientific areas using parametric (linear and nonlinear) and semiparametric models that contain parametric and nonparametric smoothing terms, but no applications where GAMLSSs were used for mixed modeling purposes were found. The only related paper found was a comparison of GAMLSSs with hierarchical generalized models by MANCO *et al.* (2011), who found by simulations that both strategies showed similar results.

Therefore, this analysis aims to illustrate how mixed models can be fitted using the GAMLSSs’ methodology. The idea of comparing it with a classical and still largely used method is to highlight the advantages and disadvantages of this approach. Basically, a GAMLSS with random effects is a generalization of both linear and generalized linear mixed models by many reasons. The most important ones are that the exponential family

Figure 4.8. Random effects predicted by the final model.

assumption for the distribution of the response is relaxed and up to four parameters of the response distribution are possible to be modeled by functions of covariates, including random effects for as many of them as needed. The form of the normal mixed GAMLSS analogous to (4.2) is given by

$$\begin{aligned}
 \mathbf{Y} | \boldsymbol{\epsilon}, \boldsymbol{\varepsilon}, \boldsymbol{\xi} &\sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}) \\
 g(\boldsymbol{\mu}) &= \mathbf{X}_G \boldsymbol{\mu} + \mathbf{X}_B \boldsymbol{\beta} + \mathbf{X}_A \boldsymbol{\alpha} + \mathbf{X}_L \boldsymbol{\lambda} + \mathbf{X}_{AL} \boldsymbol{\alpha} \boldsymbol{\lambda} + \mathbf{X}_T \boldsymbol{\tau} + \mathbf{Z}_{SP} \boldsymbol{\varepsilon} + \mathbf{Z}_P \boldsymbol{\epsilon} + \mathbf{Z}_{SS} \boldsymbol{\xi} \quad (4.5) \\
 g(\boldsymbol{\sigma}) &= \phi,
 \end{aligned}$$

where ϕ is the maximum likelihood estimate of the standard deviation of the plant's height in the scale of the link function $g(\cdot)$ used.

The `gamlss` library contains the functions `gamlssNP()`, `random()` and `re()` for fitting random effects within a GAMLSS. However, only the last of them allows random

effects to be estimated at different levels (multilevel modeling), which is required for the analysis of the sweet corn experiment. The fitting procedure of a mixed GAMLSS with `re()` uses a local (internal to the GAMLSS fitting algorithm) normal approximation to the model's likelihood, known in the literature as penalized quasi likelihood (BRESLOW and CLAYTON (1993)). The model's fitted values derive from the joint likelihood function of the response variable and the random effects vectors, while inference is based on the conditional likelihood of the response given the random effects. Details of the fitting procedure can be found in chapters 3 and 10 of STASINOPOULOS *et al.* (2017).

The form of defining a mixed GAMLSS in `gamlss()` using `re()` internally is exactly the same as with `lme()`. This makes the fitting of mixed GAMLSSs with `re()` straightforward to users of the `nlme` package. Once the mixed GAMLSS is fitted and stored in an R object, the function `getSmo()` can be used to transform it to an `lme` object, so that all methods used with mixed models fitted via `lme()` become available, such as analysis of variance via the `anova()` function and random effects estimates by level via `ranef()`. The only tool of `lme()` that is not available via `re()` is to define variance functions for the within group heteroscedastic structure, which was used in the previous analysis. This is because the philosophy of GAMLSSs is to handle problems at the observational level (such as overdispersion and heterogeneity of variances) using more general distributions for the response variable instead of trying to overcome them with less flexible ones.

Apart from having the facilities of `lme()` through the interace with `re()`, the `gamlss()` function also provides its own functionalities for fitting and assessing regression models. Among the functions that were useful in this analysis were the `wp()` and `plot()` functions. The `wp()` function provides single or multiple worm plots for `gamlss` fitted objects. Worm plots are detrended QQ-plots introduced by VAN BUUREN and FREDRIKS (2001) in order to find intervals of an explanatory variable where the model does not adequately fit the data. The well known `plot()` function of R has a personalized version in the `gamlss` library. For a `gamlss` fitted object, it produces a two by two matrix of plots, that contains a plot of the residuals against fitted values of the μ parameter, against an index of specific covariate, a kernel density estimate of the residuals and a QQ-normal plot of the residuals. It is important to state that GAMLSSs use normalized quantile residuals of DUNN and SMYTH (1996) (henceforth referred to simply as residuals), which are more general than other types of residuals in the sense that they always have a standard normal distribution when the assumed model is correct, whatever the distribution of the response variable.

Initially, a mixed model with the same random and fixed effects terms used in the final model in the previous analysis was fitted, also assuming normal distribution (NO) for the conditional response variable. F-tests' results for selected fixed effects terms in this model are shown in Table 4.5. They are exactly the same as Table 4.2. A graph of the residuals for this model obtained with the personalized `plot()` function is shown

in Figure 4.9. The top left plot in this figure is equal to Figure 4.5. The plot on the right of it shows the residuals against the days when the plant's heights were taken, highlighting the already known heterogeneity of variances by time. The two bottom plots show that the normality assumption does not seem too bad for this model. However, the worm plot presented in Figure 4.10 is a little more enlightening about it. If the normality assumption is not to be rejected, at least 95% of the quantile residuals should stay between the two elliptic curves (they are approximate point-wise 95% confidence intervals for the residuals), which is clearly not the case. Another feature of worm plots is that different shapes indicate different inadequacies in the model. The worm plot in Figure 4.10 has an S-shape form with left bent down, which indicates that the tails of the fitted distribution are too light (the meanings of all worm plots' shapes are given in section 12.4 of STASINOPOULOS *et al.* (2017)).

Table 4.5. F-tests for selected terms of the mixed GAMLSS with normal distribution.

Term	Degrees of freedom	F-value	p-value
time:tension:silicate	9	3.75	0.0001
time ²	1	69.15	<.0001
time ³	1	180.52	<.0001

Figure 4.9. Plots of residuals for mixed model with NO distribution.

A more general alternative distribution that allows for heavier tails than the normal is the generalized gamma (GG). The GG is a three-parameter distribution with

Figure 4.10. Worm plot of residuals for mixed model with NO distribution.

mean equal to the first parameter μ and variance equal to $\mu^2\sigma^2$, a function of the first two parameters (σ is the second parameter). The third parameter ν is related to the skewness of the distribution. The default link functions for each parameter were used when modeling, which are the log function for μ and σ and the identity for ν . A mixed GAMLSS was fitted using the GG distribution with the same model formula for fixed and random effects as before. Using F-tests the results found were very similar to the previously obtained ones with the normal distribution. This reinforces that there is a triple interaction effect of the linear function of time with potassium silicate and soil tension levels and a cubic polynomial function of time is needed. The model also estimated the σ and ν parameters as significantly different from zero. The two by two graph of the residuals for this model is presented in Figure 4.11. From the residuals vs fitted values plot can be seen that the heterogeneity of variances has been addressed by the use of the GG distribution. The plot of residuals vs time confirms that the residuals' variability is similar for all days of measurements. Both bottom plots agree in that there is no deviation from the normality assumption for this model overall. The worm plot for this model (Figure 4.12) confirms that by displaying the solid fitted curve of the residual points basically on the x-axis and with all points inside the two elliptic curves.

This overall worm plot is equal to the normal QQ-plot in Figure 4.13 detrended by subtracting the line. It highlights deviations of the observed residuals to their approximate expected values represented by the horizontal dotted line. However, there might be failures in the model within ranges of the explanatory variables yet. These are minor inadequacies in the model but their diagnose is still recommended, specially for the most important explanatory variables in the analysis (STASINOPOULOS *et al.* (2017), page 428). This is

Figure 4.11. Plots of residuals for mixed model with GG distribution.

Figure 4.12. Plots of residuals for mixed model with GG distribution.

possible to be check for GAMLSSs with multiple worm plots by intervals of the explanatory variables (or by levels of factors). When applied to the levels of soil tension and potassium silicate factors in the mixed GG model no misfits were found. For the time covariate five worm plots were generated with non-overlapping intervals of equal number of observations and are displayed in Figure 4.13. The upper panel gives the interval ranges (each day

of observation in this case) and the worm plots below are read along rows from bottom left to top right, made with the residuals on each interval in order. For example, the bottom right worm plot was made with residuals of heights taken at 60 days (the third time interval). Contrasting with the previous worm plot shown, in this figure all worm plots appear to have problems. Each worm plot contains a thin solid line that is a cubic fit on the residuals. When fitting multiple worm plots with `wp()` it is possible to get the coefficients of these cubic fits, which help understanding the inadequacies in the worm plots. The estimated coefficients (intercept $\widehat{\beta}_0$, linear $\widehat{\beta}_1$, quadratic $\widehat{\beta}_2$ and cubic $\widehat{\beta}_3$) of the cubic fits in these worm plots are given in Table 4.6.

Figure 4.13. Worm plots by time for mixed model with GG distribution.

VAN BUUREN and FREDRIKS (2001) classify the absolute values of these cubic fits' coefficients. For the intercept and the linear coefficient they established that estimates above 0.1 are considered misfits that (according to BENARD and BOS-LEVENBACH (1953) and VAN ZWET (1964)) indicate high differences in the mean and variation of the theoretical model residuals and the fitted residuals. Similarly, they considered threshold values for quadratic and cubic absolute estimates as 0.05 and 0.03 and argue that they suggest inadequacies in the residuals' skewness and kurtosis, respectively. Applying these rules to the values on Table 4.6 it can be concluded that there are inadequacies in the mean of the residuals at all times (since the first column has all estimates in absolute values above 0.1), in the variance for some days and that there are not misfits in the skewness and kurtosis of the residuals. This residual evaluation can also be related directly to the height predictions obtained with this model, shown in Figure 4.14 in black dotted lines over the observed ones. For example, the highly negative intercept for the cubic fit at 45

Table 4.6. Coefficients of cubic fits in worm plots of GG mixed model.

Time	$\widehat{\beta}_0$	$\widehat{\beta}_1$	$\widehat{\beta}_2$	$\widehat{\beta}_3$
30 days	0.1806	-0.0291	-0.0251	-0.0135
45 days	-0.4605	-0.1162	-0.0163	-0.0008
60 days	0.5964	0.0790	-0.0454	0.0039
75 days	-0.4880	-0.1002	0.0116	-0.0092
90 days	0.2662	-0.3170	-0.0187	0.0215

days in Table 4.6 means that the mean of the fitted distribution at that time was too high compared to the observed one, generating negative residuals in the bottom middle worm plot. This also translates to superestimated plants' heights on average at that time, as can be seen in Figure 4.14.

Figure 4.14. Plants' heights observed and predicted with GG mixed model.

STASINOPOULOS *et al.* (2017) comment that in general it will not be possible to build a model without areas of misfits for all the covariates included. Even though very sensitive, this detailed residual analysis gives valuable information about the model fitting according to ranges of the explanatory variables. Furthermore, it shows where specific failures in the model are and what needs to be done to minimize them. In this analysis, the problem with the mean has to do with the flexibility of the chosen function for the time effect. Probably a nonlinear or a smoothing function of time would account better for the unequal growth rates. However, since there are no problems with the residual analysis by neither of the factors, it is unlikely that the significant interaction effect would change by modifying the time function.

Finally, the predicted random effects for the GG mixed GAMLSS are shown in Figure 4.15. Evaluations of the normality assumption might be unreliable due to the number of observations they take into account, but overall there does not seem to be issues with normality. Differently than in the previous analysis, here the magnitude of the random effects vary considerably by level and are much higher in the plots and plants levels than the correspondent ones obtained by the linear mixed model. This is also noticeable from the variability of the height predictions in Figures 4.7 and 4.14.

Figure 4.15. Estimated random effects with the GG mixed model by level.

4.4 Conclusions

In this paper a comparison of two methodologies for analysing a multilevel mixed model was described. In the first, that used the `nlme::lme()` function, the heterogeneity of variances was handled by defining a power variance function for the residuals by the

time covariate. The final model contained three levels of random effects whose values were predicted in the scale of 1×10^{-8} , resulting in similar predictions for plants that received the same treatment. The significant effects obtained with this model were a triple interaction of soil tension and potassium silicate factors with the linear effect of time, apart from significant effects of the quadratic and cubic functions of time. The second analysis used the GAMLSSs methodology with random effects to model the plant growths. The initial model fitted with `lme()` assuming normality for the residuals was discarded and a more flexible one with the generalized gamma distribution was used instead. A detailed residual analysis using worm plots was conducted and inadequacies in the model by ranges of the time covariate were found, even though the overall normality assumption was not rejected. The estimated random effects varied in magnitude by levels which resulted in more variable predictions of the plants' growths. The GG mixed GAMLSS suggested the same fixed effects terms as the linear mixed model.

The approaches used to deal with inadequacies in the model were different in the two analysis. The heteroscedasticity was accommodated in the second analysis using a more flexible distribution for the response variable, while in the linear mixed model a more general form for the variance matrix of the residuals was defined. However, in the linear mixed model a detailed residual analysis using worm plots by time was not made. In fact, checking the normality assumption with a shapiro-wilk test for the final mixed model resulted in a p-value in the order of 1×10^{-8} , while for the mixed GAMLSS with GG distribution the same test gave 0.91 as the descriptive level. The AIC criterion for the LMM is 10620,21 and for the GG GAMLSS is 10425,14, also favouring the second. Furthermore, the `re()` interface of `gamlss()` allows the same facilities of `lme()` as well as a wide range of response distributions to be used and the possibility to model other parameters than the mean explicitly with covariates.

The result of the experiment was not dependent on the method of analysis used, though. The conclusion is there are significantly different growth trends in plants that were on soil with different tension levels and received different doses of potassium silicate. However, which treatment combinations differ among themselves and why would have to be further explored, since the graph of average growth lines per treatment in Figure 4.3 is not clear about it. Furthermore, another modeling approach should be investigated to reduce inadequacies in the model with respect to the time covariate. Possible alternatives could be to use a logistic nonlinear function modeling this functions' parameters with random and fixed effects or to smooth the time effect with a nonparametric function such as a cubic spline.

References

AKANTZILIOTOU, C., R. A. RIGBY, and D. M. STASINOPOULOS, 2002 The R imple-

- mentation of generalized additive models for location, scale and shape. In *Statistical modelling in Society: Proceedings of the 17th International Workshop on statistical modelling*, pp. 75–83, Statistical Modelling Society.
- BARAJAS, F. H., M. TORRES, L. ARTEAGA, and C. CASTRO, 2015 GAMLSS models applied in the treatment of agro-industrial waste. *Comunicaciones en Estadística* **8**: 245.
- BENARD, A. and E. C. BOS-LEVENBACH, 1953 Het uitzetten van waarnemingen op waarschijnlijkheidspapier. *Statistica Neerlandica* **7**: 163–173.
- BOLKER, B. M., M. E. BROOKS, C. J. CLARK, S. W. GEANGE, J. R. POULSEN, M. H. H. STEVENS, and J.-S. S. WHITE, 2009 Generalized linear mixed models: a practical guide for ecology and evolution. *Trends in ecology & evolution* **24**: 127–135.
- BRESLOW, N. E. and D. G. CLAYTON, 1993 Approximate Inference in Generalized Linear Mixed Models. *Journal of the American Statistical Association* **88**: 9–25.
- BRIEN, C. J. and C. G. DEMETRIO, 2009 Formulating Mixed Models for Experiments, Including Longitudinal Experiments. *Journal of Agricultural, Biological, and Environmental Statistics* **14**: 253–280.
- BUCKLEY, Y. M., D. T. BRIESE, and M. REES, 2003 Demography and management of the invasive plant species *Hypericum perforatum*. I. Using multi-level mixed-effects models for characterizing growth, survival and fecundity in a long-term data set. *Journal of Applied Ecology* **40**: 481–493.
- DAVIDIAN, M. and D. M. GILTINAN, 1995 *Nonlinear models for repeated measurement data*, volume 62. CRC press.
- DUNN, P. K. and G. K. SMYTH, 1996 Randomized Quantile Residuals. *Journal of Computational and Graphical Statistics* **5**: 236–244.
- FRANCIS, B., M. GREEN, and C. PAYNE, 1993 *The GLIM system: release 4 manual*. Clarendon Press.
- GILCHRIST, R., A. KAMARA, and J. RUDGE, 2009 an Insurance Type Model for the Health Cost of Cold Housing: an Application of Gamlss. *REVSTAT–Statistical Journal* **7**: 55–66.
- HUDSON, I. L., S. W. KIM, and M. R. KEATLEY, 2009 Climatic influences on the flowering phenology of four Eucalypts: a GAMLSS approach. In *18th World IMACS / MODSIM Congress*.

- JACKSON, W. B., 1939 Reliability of mental tests. *British Journal of Psychology* **29**: 267–287.
- LAIRD, N. M. and J. H. WARE, 1982 Random-effects models for longitudinal data. *Biometrics* pp. 963–974.
- LI, W., P. ZHANG, J. YE, L. LI, and P. A. BAKER, 2011 Impact of two different types of El Niño events on the Amazon climate and ecosystem productivity. *Journal of Plant Ecology* **4**: 91–99.
- LINDSTROM, M. J. and D. M. BATES, 1988 Newton—Raphson and EM algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association* **83**: 1014–1022.
- MANCO, O. C. U., F. H. BARAJAS, and V. GIAMPAOLI, 2011 A study to compare HGLM and GAMLSS in mixed linear models. *IWSM 2011* p. 622.
- MCCULLOCH, C. E. and J. M. NEUHAUS, 2001 *Generalized linear mixed models*. Wiley Online Library.
- PAINE, C. E. T., T. R. MARTHEWS, D. R. VOGT, D. PURVES, M. REES, A. HECTOR, and L. A. TURNBULL, 2012 How to fit nonlinear plant growth models and calculate growth rates: An update for ecologists. *Methods in Ecology and Evolution* **3**: 245–256.
- PETTERLE, R. R. and L. FORMIGUIERI, 2014 Modelos aditivos generalizados para locação, escala e forma na análise do número de lesões corticais em pacientes com esclerose múltipla. Technical report, Universidade Federal do Paraná.
- PIEPHO, H. P., A. BUCHSE, and K. EMRICH, 2003 A Hitchhiker 's Guide to Mixed Models for Randomized Experiments. *Journal of Agronomy & Crop Science* pp. 310–323.
- PINHEIRO, J. C. and D. M. BATES, 1996 Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing* **6**: 289–296.
- PINHEIRO, J. C. and D. M. BATES, 2000 *Mixed-Effects Models in S and S-PLUS*. Springer.
- REES, M., A. SHEPPARD, D. BRIESE, and M. MANGEL, 1999 Evolution of size-dependent flowering in *Onopordum illyricum*: a quantitative assessment of the role of stochastic selection pressures. *The American Naturalist* **154**: 628–651.
- RIGBY, R. A. and D. M. STASINOPOULOS, 2005 Generalized additive models for location, scale and shape,(with discussion). *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **54**: 507–554.

- SCHALL, R., 1991 Estimation in generalized linear models with random effects. *Biometrika* **78**: 719–727.
- SCHLENKER, W. and M. J. ROBERTS, 2009 Nonlinear temperature effects indicate severe damages to U.S. crop yields under climate change. *Proceedings of the National Academy of Sciences* **106**: 15594–15598.
- STASINOPOULOS, M., R. RIGBY, G. HELLER, V. VOUDOURIS, and F. D. BASTIANI, 2017 *Flexible Regression and Smoothing The GAMLSS packages in R*. CRC Press.
- STRATHE, A. B., A. DANFÆR, H. SØRENSEN, and E. KEBREAB, 2010 A multilevel nonlinear mixed-effects approach to model growth in pigs. *Journal of Animal Science* **88**: 638–649.
- VAN BUUREN, S. and M. FREDRIKS, 2001 Worm plot: A simple diagnostic device for modelling growth reference curves. *Statistics in Medicine* **20**: 1259–1277.
- VAN ZWET, W. R., 1964 *Convex transformations of random variables*, volume 7. Mathematisch centrum.
- WEST, B. T., K. B. WELCH, and A. T. GALECKI, 2014 *Linear mixed models: a practical guide using statistical software*. CRC Press.

5 FINAL CONSIDERATIONS

The aim of this dissertation was to understand and explore the generalized additive models for location, scale and shape. This started to be accomplished by studying this type of models for more than a year and analyzing real data sets in this dissertation. It was tough in the beginning because there were many papers published using this methodology of analysis but very few explaining its theory. In fact, the first book about GAMLSSs was published by the authors in April of this year (2017), which was bought readily and received about two months later.

Mainly based in the original papers and materials found on line about the topic, the first chapter of this dissertation was written about the methodology of GAMLSSs. This increased my knowledge about the capabilities of this general class of univariate models and what differentiates it to other approaches. It became clear that the high number of options it gives to practitioners comes with the extra cost of knowing all of them and how they work within this framework.

Some knowledge of GAMLSSs in practice was gained by the two analysis in the second and third chapters. The count data with excess of zeros was my first analysis using the GAMLSSs' methodology. The data set had already been analyzed in a previous paper by my supervisor and other authors using a GLM's approach. Although simple, this analysis made me study the different options of models for zero inflated data. The sweet corn experiment, conversely, had a complicated design with three levels of nested factors. It's analysis enhanced my understanding of what modeling correlated data means and why it is important.

The generalized additive models for location, scale and shape encompass most of the univariate regression models created in the history of regression analysis. It took decades for the authors to develop it and certainly would not be possible to fully understand it's theory with the time of a Master's degree. GAMLSSs are nowadays widely used and in constant improvement. For example, one important extension (`R` package `gamboostLSS`) has been made by Mayr *et al.* (2012)¹ that allows modeling of high dimensional data. Therefore, there is still lots to be learned about regression analysis using GAMLSSs, but I am excited to have given the first step.

¹Mayr, A., Fenske, N., Hofner, B., Kneib, T., & Schmid, M. (2012). Generalized additive models for location, scale and shape for high dimensional data—a flexible approach based on boosting. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3), 403-427.

APPENDIX

Appendix A: Roots analysis

```
# Roots analysis with GAMLSSs
rm(list=ls())
pac <- c("ggplot2", "grid", "gtable", "gamlss")
lapply(pac, library, character.only = TRUE)
d1 <- data.frame(read.csv2(file = "data.csv", h = T))
d1 <- transform(d1, photo = as.factor(photo), bap = as.factor(bap))
levels(d1$photo) <- c("8h", "16h")
levels(d1$bap) <- c("2.2\u03BCM", "4.4\u03BCM", "8.8\u03BCM", "17.6\u03BCM")
str(d1)

# Exploratory analysis
hist(d1$roots)
table(d1$roots)
# Splitted histogram of number of roots for each photoperiod duration
p <- ggplot(data=d1, aes(roots)) +
  geom_bar() +
  facet_grid(~photo) +
  labs(x = "Number of roots", y = "Frequency") +
  theme(axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 12),
        strip.text = element_text(size = 12))
label = "Photoperiod duration"
z <- ggplotGrob(p)
posT <- subset(z$layout, grepl("strip-t", name), select = t:r)
height <- z$heights[min(posT$t)] # Height of current top strips
z <- gtable_add_rows(z, height, min(posT$t)-1)
stripT <- gTree(name = "Strip_top", children = gList(
  rectGrob(gp = gpar(col = NA, fill = "grey85")),
  textGrob(label, gp = gpar(fontsize = 12))))
z <- gtable_add_grob(z, stripT, t=min(posT$t), l=min(posT$l), r=max(posT$r))
z <- gtable_add_rows(z, unit(1/5, "line"), min(posT$t) )
grid.newpage()
grid.draw(z)

# Boxplots by treatments
ggplot(d1, aes(x = photo, y = roots, fill = bap2)) +
  geom_boxplot() +
  scale_fill_discrete(name = "BAP \u2013 concentration") +
  labs(x = "Treatment combination", y = "Number of roots", color = "Bap") +
  theme(axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 12),
```



```

strip.text = element_text(size = 12),
legend.title = element_text(size = 12),
legend.text = element_text(size = 12) )
# Interaction plot of the two factors
ggplot(d1, aes(x = bap, y = roots, color = photo) ) +
geom_count() +
stat_summary(aes(y = roots, group = photo), fun.y = mean, geom = "line") +
labs(x = "BAP□concentration", y = "Number□of□roots", colour = "Photoperiod",
size = "Frequency") +
scale_y_continuous(limits = c(0, 18), breaks = c(seq(0, 18, length.out = 10) ) ) +
theme(axis.title.x = element_text(size = 12),
axis.title.y = element_text(size = 12),
axis.text.x = element_text(size = 12),
axis.text.y = element_text(size = 12),
strip.text = element_text(size = 12),
legend.title = element_text(size = 12),
legend.text = element_text(size = 12) )

# Analysis – Modeling with GAMLSSs – Initial inspection of ZI distributions
newpar <- par(mfrow = c(2,3), cex = .6, cex.axis = 1.3, cex.main = 1.3)
histDist(y=d1$roots, family=PO, ylim=c(0,0.3), main="PO", par=newpar)
histDist(y=d1$roots, family=NBI, ylim=c(0,0.3), main="NB", par=newpar)
histDist(y=d1$roots, family=ZIP2, ylim=c(0,0.3), main="ZIP", par=newpar)
histDist(y=d1$roots, family=ZINBI, ylim=c(0,0.3), main="ZINB", par=newpar)
histDist(y=d1$roots, family=ZIPIG, ylim=c(0,0.3), main="ZIPIG", par=newpar)
histDist(y=d1$roots, family=ZIBNB, ylim=c(0,0.3), main="ZIBNB", par=newpar)
# ZIP2 dist
m1 <- gamlss(roots ~ 1, d = d1, family = ZIP2)
m1 <- stepGAICAll.A(m1, scope = list(lower = ~1, upper = ~photo*bap), d = d1)
summary(m1)
drop1(m1)
drop1(m1, what = "sigma")
plot(m1)
rqres.plot(m1) # Randomizes residual plot for discrete response variables
wp(m1, ylim.all = 1) # Overall worm plot
wp(m1, xvar = ~photo)
# ZINB dist
m2 <- gamlss(roots ~ 1, d = d1, family = ZINBI)
m2 <- stepGAICAll.A(m2, scope = list(lower = ~1, upper = ~photo*bap), d = d1)
summary(m2)
drop1(m2)
plot(m2)
rqres.plot(m2)
wp(m2)
legend(x = .96, y = 1.3, legend = "8□h", col = "red", border = F, bty = "n")
legend(x = -1.73, y = 1.3, legend = "16□h", col = "red", border = F, bty = "n")
wp(m2, xvar = ~photo)

```

```

# ZIPIG dist (3 parameters - nu is the zero inflation par)
m3 <- gamlss(roots ~ 1, d = d1, family = ZIPIG)
m3 <- stepGAICAll.A(m3, scope = list(lower = ~1, upper = ~photo*bap), d = d1)
summary(m3)
drop1(m3)
plot(m3)
rqres.plot(m3)
wp(m3)
wp(m3, xvar = ~photo)
# ZIBNB dist (4 parameters - mu = mean, tau is the zero inflation par)
m4 <- gamlss(roots ~ 1, d = d1, family = ZIBNB)
m4 <- stepGAICAll.A(m3, scope = list(lower = ~1, upper = ~photo*bap), d = d1)
plot(m4)
rqres.plot(m4)
wp(m4)
wp(m4, xvar = ~photo)
# Comparing models with different distributions
GAIC(m1, m2, m3, m4) # AIC
GAIC(m1, m2, m3, m4, k = log(length(d1$roots))) # BIC

```

Appendix B: Sweet corn analysis

```

# Corn analysis GAMLSS
rm(list=ls())
pac <- c("gamlss", "ggplot2", "grid", "gtable", "nlme", "dplyr", "directlabels",
"gamlss.nl", "nlstools", "gtools", "tidyr", "lattice")
library(pac, library, character.only = TRUE) # Load all packages
d1 <- read.csv2(file = "dados_plantas_ord.csv", header = T, sep=",")
d1 <- transform(d1, block=factor(block), plot=factor(plot), tens=factor(tens),
subplot=factor(subplot), sil=factor(sil), treat=factor(treat),
plant=factor(plant), time2=factor(time))
levels(d1$tens) <- c("15kPa", "30kPa", "45kPa", "60kPa")
levels(d1$sil) <- c("0L/hec", "6L/hec", "12L/hec", "24L/hec")
levels(d1$time2) <- c("Day_30", "Day_45", "Day_60", "Day_75", "Day_90")
levels(d1$treat) <- c("15kPa, 0L/hec", "15kPa, 6L/hec",
"15kPa, 12L/hec", "15kPa, 24L/hec",
"30kPa, 0L/hec", "30kPa, 6L/hec",
"30kPa, 12L/hec", "30kPa, 24L/hec",
"45kPa, 0L/hec", "45kPa, 6L/hec",
"45kPa, 12L/hec", "45kPa, 24L/hec",
"60kPa, 0L/hec", "60kPa, 6L/hec",
"60kPa, 12L/hec", "60kPa, 24L/hec")
str(d1)

# Exploratory analysis - height
summary(d1$height)
hist(d1$height) # Does not inform anything since plants grow with time...

```

```

plot(height ~ block, data = d1) # Very similar heights by blocks
tapply(d1$height, d1$block, mean)
plot(height ~ tens, data = d1) # Very similar heights by tension levels
tapply(d1$height, d1$tens, mean)
plot(height ~ sil, data = d1) # Very similar heights by silicate levels
tapply(d1$height, d1$sil, mean)
plot(height ~ time, data = d1) # Very similar heights by silicate levels
tapply(d1$height, d1$time, min)
tapply(d1$height, d1$time, max)
# Personalized graphs
# Histograms at each observation day
ggplot(data=d1, aes(height)) +
geom_histogram(bins=40) +
facet_grid(~time2) +
labs(x="Height□(cm)", y="Frequency") +
theme(axis.title.x = element_text(size=12),
axis.title.y = element_text(size=12),
axis.text.x = element_text(size=12),
axis.text.y = element_text(size=12),
strip.text = element_text(size=12))
# Plant's growth by treatments
ggplot(d1, aes(x=time, y=height)) +
geom_line(aes(col=plant)) +
facet_wrap(~treat) +
geom_point() +
labs(x="Time□(days)", y="Plant's□height□(cm)", colour="Plants") +
scale_x_continuous(breaks = c(30,45,60,75,90)) +
theme(axis.title.x = element_text(size=6),
axis.title.y = element_text(size=6),
axis.text.x = element_text(size=6),
axis.text.y = element_text(size=6),
strip.text.x = element_text(size=6),
legend.position="none")
# Plant's growth
ggplot(d1, aes(x=time, y=height)) +
geom_line(aes(col=plant)) +
geom_point() +
labs(x="Time□(days)", y="Plant's□height□(cm)") +
scale_x_continuous(breaks = c(30,45,60,75,90)) +
theme(axis.title.x = element_text(size=8),
axis.title.y = element_text(size=8),
axis.text.x = element_text(size=8),
axis.text.y = element_text(size=8),
legend.position = "none")
# Plant's mean growth by treatment
ggplot(d1, aes(x = time, y = height)) +
geom_point() +

```

```

stat_summary(fun.y = mean, geom="line", lwd=1, aes(group=treat, col=treat) ) +
labs(x = "Time□(days)", y = "Plant's□height□(cm)", colour = "Treatments") +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size=8),
axis.title.y = element_text(size=8),
axis.text.x = element_text(size=6),
axis.text.y = element_text(size=6),
legend.text = element_text(size=6),
legend.title= element_text(size=8),
legend.key.size = unit(0.5, "cm"))
# Mean plant's growth by plots
ggplot(d1, aes( x = time, y = height) ) +
geom_point() +
stat_summary(fun.y=mean, geom="line", lwd=1, aes(group=plot, col=plot) ) +
labs(x = "Time□(days)", y = "Plant's□height□(cm)", colour = "Plot") +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size=8),
axis.title.y = element_text(size=8),
axis.text.x = element_text(size=6),
axis.text.y = element_text(size=6),
legend.text = element_text(size=6),
legend.title= element_text(size=8),
legend.key.size = unit(0.5, "cm"))
# Mean plant's growth by subplots
ggplot(d1, aes( x = time, y = height) ) +
geom_point() +
stat_summary(fun.y = mean, geom = "line", lwd=1, aes(group=subplot, col=subplot) ) +
labs(x = "Time□(days)", y = "Plant's□height□(cm)", colour = "Subplot") +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size=8),
axis.title.y = element_text(size=8),
axis.text.x = element_text(size=6),
axis.text.y = element_text(size=6),
legend.text = element_text(size=6),
legend.title= element_text(size=8),
legend.key.size = unit(0.5, "cm"))

# Growth analysis
# Analysis with nlme
# Testing fixed effects in lme()
# For testing terms (factors) use anova() without "-1", because
# "-1" puts the intercept variation to the first term making it significant.
# Note also that the order of terms matter because they 'arent orthogonal!
# About orthogonal polynomials: one could use them whenever there is no particular
# interest on comparing or interpreting the parameter estimated values, because
# using them makes their estimation uncorrelated with other sources of variation
# Analysis begins here

```

```

# Model only with triple (linear) interaction
mn1 <- lme(height ~ block + tens*sil*I(time),
random = ~1|plot/subplot/plant, method = "ML", data = d1)
anova(mn1) # There is a significant triple interaction effect!
plot(mn1) # Linear models does not capture the nonlinearity, maybe a cubic would
# Model with cubic term for time and triple (linear) interaction
mn2 <- update( mn1, ~. + I(time^2) + I(time^3))
anova(mn2) # Interaction of sil*tens with time (linear) significant
plot(mn2) # Residuals of 2nd group much smaller, but still heteroscedastic
# Model with triple (linear) interactions for time, time^2 and time^3
mn3 <- update( mn1, ~. + tens*sil*I(time^2) + tens*sil*I(time^3))
anova(mn3) # Interactions of sil*tens with time^2 and time^3 NOT significant
# Therefore, we stay with model mn2
# Modeling heteroscedasticity
# Modeling heteroscedasticity of model mn2 with time
mn4 <- update(mn2, weights = varPower(form = ~ time)) # Pg 177 of P&B (2000)
anova(mn4) # Significance of terms has not changed
anova(mn2, mn4) # Heteroscedastic model improved the fit
plot(mn4) # Variance of residuals is more similar among groups
plot(mn4, resid(., type = "p") ~ fitted(.)|time2,
abline=0, ylim = c(-5,5), xlim = c(-50,350) ) # Same graph split by time
plot(mn4, height ~fitted(.), abline = c(1,1) ) # Response vs predicted, very good
qqnorm(mn4,~resid(.)) # However, the QQ-plot has a jump in the middle
shapiro.test(resid(mn4) ) # In fact, Shapiro-Wilk test rejects normality
AIC(mn4)
BIC(mn4)
# Growth predictions with this final model for each plant by treatment
d1.2 <- data.frame( cbind( d1, prev = predict(mn4) ) ) # Adding plant predictions
ggplot(d1.2, aes( x = time, y = height, col = treat ) ) +
facet_wrap(~treat, ncol = 4) +
geom_line(aes( group = plant ) ) +
geom_line(linetype = "longdash", col = "black", lwd = .5,
aes(y = prev, group = plant, col = "Pred") ) +
labs(x = "Time", y = "Height", colour = "Treatments") +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size = 12),
axis.title.y = element_text(size = 12),
axis.text.x = element_text(size = 10),
axis.text.y = element_text(size = 10),
legend.position = "none")
# Checking final mixed model assumptions – Pg 174 P&B
# Assumption 1 – Residuals within groups ~N(0, sigma^2)
boxplot(resid(mn4) ~ d1$plot) # Boxplot of residuals within plots, ok
boxplot(resid(mn4) ~ d1$subplot) # Boxplot of residuals within subplots, ok
boxplot(resid(mn4) ~ d1$plant) # Boxplot of residuals within plants, ok
# Assumption 2 – Random effects ~N(0,V)
qqnorm(unlist(ranef(mn2)$plot))

```

```

qqline(unlist(ranef(mn2)$plot))
# Normality of ranef at plots' level questionable, too few observations
qqnorm(unlist(ranef(mn2)$subplot))
qqline(unlist(ranef(mn2)$subplot))
# Normality assumption of random effects at subplots' level better
qqnorm(unlist(ranef(mn2)$plant))
qqline(unlist(ranef(mn2)$plant))
# Normality assumption of ranef at plants' level questionable by bottom tail
# Density plots of predicted random effects
aux <- plyr::ldply(ranef(mn4))
ggplot(aux, aes(x = '(Intercept)')) +
geom_density() +
geom_rug() +
facet_wrap(~ .id, scales = "free_y") +
theme(axis.title.x = element_text(size = 15),
axis.title.y = element_text(size = 15),
axis.text.x = element_text(size = 13),
axis.text.y = element_text(size = 13),
strip.text.x = element_text(size = 13))
# Or using package lattice
aux <- plyr::ldply(list("Random effects" = plyr::ldply(ranef(mn4))), .id = "model")
densityplot(~ '(Intercept)' | model + .id,
auto.key = TRUE,
scales = "free",
data = aux)

# Modeling with gamlss()
# drop1() can be used to do LRT of terms of GAMLSSs fitted with ML estimation
# summary() does Wald tests (less reliable, pg 125 'GAMLSS book) of single coef's
# But gamlss() has 2 ways of dealing with random effects based on PQL,
# apart from the gamlssNP() function which can also be used (limited capabilities).
# The 2 ways are:
# i) Use function 'random()' as argument in the gamlss() function to fit random
# intercept models for one or more random effects in the model.
# ii) Use function 're()' as argument in the gamlss() function to fit random
# intercept and slope models, multilevel and repeated measures, which is needed
# Model only with triple (linear) interaction
mg1 <- gamlss(height ~ re(fixed = ~ block + tens*sil*I(time)),
random = ~1|plot/subplot/plant, msMaxIter = 99), data = d1)
anova(getSmo(mg1)) # Same results as with lme(), triple interaction significant
plot(getSmo(mg1)) # Clearly this linear model does not capture the nonlinearity,
# maybe a cubic would. This plot is centered at x = fitted = 0!
# Model with cubic term for time and triple (linear) interaction
mg2 <- update(mg1, ~re(fixed = ~block + tens*sil*I(time) + I(time^2) + I(time^3),
random = ~1|plot/subplot/plant))
anova(getSmo(mg2)) # Interaction of sil*tens with time (linear) significant
plot(getSmo(mg2)) # Residuals of 2nd group much smaller, but stil heteroscedastic

```

```

plot(mg2) # GAMLSS 2 by 2 plot of residuals
wp(mg2, ylim.all = 1.3) # Worm plot not ok
x <- wp(mg2, xvar = ~time, ylim.worm = 1, n.inter = 5) # Worm plots by time
x # Cubic 'fits coefficients
# Changing graphical options of 'GAMLSS plot
newpar <- par(mfrow = c(2,2), mar = par("mar") + c(0,1,0,0), cex = .4,
cex.lab = 2, cex.axis = 1.5, cex.main = 2)
plot(mg2, xvar = d1$time, par = newpar)
# Graph of predictions with NO model for each plant in its respective treatment
d1.no <- data.frame( cbind( d1, prev = predict(mg2, type = "response") ) )
ggplot(d1.no, aes( x = time, y = height, col = treat ) ) +
geom_line(aes( group = plant ) ) +
geom_line(linetype = "longdash", col = "black", lwd = .08,
aes(y = prev, group = plant, col = "Pred") ) +
labs(x = "Time", y = "Height", colour = "Treatments") +
facet_wrap(~treat) +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size = 13),
axis.title.y = element_text(size = 13),
axis.text.x = element_text(size = 12),
axis.text.y = element_text(size = 12),
legend.text = element_text(size = 12),
legend.title = element_text(size = 12),
legend.position = "none" )

# GAMLSS with GG dist
histDist(height, family="GG", data = d1, ylim = c(0,0.005),
main="Hist of height with fitted GG dist", xlab="Height")
# Getting to know GG dist (could be done better with gamlss.demo package)
n <- 1e5
y <- rGG(n, mu=2, sigma=.5, nu=1)
x <- seq(0, 50, length.out = 1e4)
hist(y, prob = T, ylim = c(0,1), border = F, main = "", cex.axis = 1.4, cex.lab=1.4)
lines(density(y), lwd = 4)
legend(x=5,y=1,legend="mu=2, sigma=.5, nu=1", fill="black", text.font=4,cex=1.1)
lines(density(rGG(n, mu=3, sigma=1, nu=1)), col = "blue", lwd = 4)
legend(x=5, y=.8, legend = "mu=3, sigma=1, nu=.9", fill="blue", text.font=4,cex=1.1)
lines(density(rGG(n, mu=1, sigma=.7, nu=1)), col = "red", lwd = 4)
legend(x=5, y=.6, legend = "mu=1, sigma=.7, nu=1", fill="red", text.font=4,cex=1.1)
lines(density(rGG(n, mu=3, sigma=.2, nu=3)), col = "green", lwd = 4)
legend(x=5, y=.4, legend = "mu=3, sigma=.2, nu=3", fill="green", text.font=4,cex=1.1)
# Modeling only the mean parameter, same predictor as normal model
mg.gg <- gamlss(height ~re(fixed = ~block + tens*sil*I(time) + I(time^2) +
I(time^3), random = ~1|plot/subplot/plant, opt = "optim" ),
family = GG, data = d1, method = mixed(2,99),
glm.trace = T, bf.trace = T) # Took me 1.57 mins
anova(getSmo(mg.gg)) # Same conclusions as with normal model

```

```

summary(mg.gg) # 'GAMLSS object summary, sigma and nu intercepts significant
plot(mg.gg, xvar = d1$time) # 2 by 2 graph with time as the index in top right plot
GAIC(mg.gg) # Equivalent to AIC when no k value is defined because the default is 2
wp(mg.gg)
# Worm plots
# 2 by 2 wormplots are very sensitive
x1 <- wp(mg.gg, xvar = ~tens + sil, ylim.worm = 1, n.inter = 4)
x1
x2 <- wp(mg.gg, xvar = ~tens, ylim.worm = 1) # Worm plots by tens (4, one per level)
x2 # Cubic 'fits coefficients
x3 <- wp(mg.gg, xvar = ~sil, ylim.worm = 1) # Worm plots by sil (4, one per level)
x3 # Cubic 'fits coefficients
x4 <- wp(mg2, xvar=~time, ylim.worm=1, n.inter=5) # Worm plots by time (5 intervals)
x4 # Cubic 'fits coefficients
# Graph of predicted curves with GG model for each plant in its respective treatment
d1.gg <- data.frame( cbind( d1, prev = predict(mg.gg, type = "response") ) )
ggplot(d1.gg, aes( x = time, y = height, col = treat ) ) +
geom_line(aes( group = plant ) ) +
geom_line(linetype = "longdash", col = "black", lwd = .3,
aes(y = prev, group = plant, col = "Pred") ) +
labs(x = "Time", y = "Height", colour = "Treatments") +
facet_wrap(~treat) +
scale_x_continuous(breaks = c(30,45,60,75,90) ) +
theme(axis.title.x = element_text(size = 12),
axis.title.y = element_text(size = 12),
axis.text.x = element_text(size = 12),
axis.text.y = element_text(size = 12),
legend.text = element_text(size = 12),
legend.title = element_text(size = 12),
legend.position = "none" )
# Predicted random effects with GG model
aux <- plyr::ldply(ranef(getSmo(mg.gg))) # ggplot2 package
ggplot(aux, aes(x = '(Intercept)')) +
geom_density() +
geom_rug() +
facet_wrap(~.id, scales = "free") +
theme(axis.title.x = element_text(size = 15),
axis.title.y = element_text(size = 15),
axis.text.x = element_text(size = 13),
axis.text.y = element_text(size = 13),
strip.text.x = element_text(size = 13))
aux <- plyr::ldply(list("Random effects" = plyr::ldply(ranef(getSmo(mg.gg))),
.id = "model") # lattice package
densityplot(~'(Intercept)' | model + .id,
auto.key = TRUE,
scales = "free",
data = aux)

```