Additive models with shape constraints

submitted by

Natalya Pya

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

August 2010

COPYRIGHT

Attention is drawn to the fact that copyright of this thesis rests with its author. This copy of the thesis has been supplied on the condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the thesis and no information derived from it may be published without the prior written consent of the author.

This thesis may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.

Signature of Author .....

Natalya Pya

Summary

In many practical situations when analyzing a dependence of one or more explanatory variables on a response variable it is essential to assume that the relationship of interest obeys certain shape constraints, such as monotonicity or monotonicity and convexity/concavity. In this thesis a new approach to shape preserving smoothing within generalized additive models has been developed. In contrast with previous quadratic programming based methods, the project develops intermediate rank penalized smoothers with shape constrained restrictions based on re-parameterized B-splines and penalties based on the P-spline ideas of Eilers and Marx (1996). Smoothing under monotonicity constraints and monotonicity together with convexity/concavity for univariate smooths; and smoothing of bivariate functions with monotonicity restrictions on both covariates and on only one of them are considered.

The proposed shape constrained smoothing has been incorporated into generalized additive models with a mixture of unconstrained and shape restricted smooth terms (mono-GAM). A fitting procedure for mono-GAM is developed. Since a major challenge of any flexible regression method is its implementation in a computationally efficient and stable manner, issues such as convergence, rank deficiency of the working model matrix, initialization, and others have been thoroughly dealt with. A question about the limiting posterior distribution of the model parameters is solved, which allows us to construct Bayesian confidence intervals of the mono-GAM smooth terms by means of the delta method. The performance of these confidence intervals is examined by assessing realized coverage probabilities using simulation studies.

The proposed modelling approach has been implemented in an R package `monogam`. The model setup is the same as in `mgcv(gam)` with the addition of shape constrained smooths. In order to be consistent with the unconstrained GAM, the package provides key functions similar to those associated with `mgcv(gam)`. Performance and timing comparisons of mono-GAM with other alternative methods has been undertaken. The simulation studies show that the new method has practical advantages over the alternatives considered. Applications of mono-GAM to various data sets are presented which demonstrate its ability to model many practical situations.

Acknowledgements

I would like to express my deep appreciation and gratitude to my supervisor, Professor Simon Wood. This PhD project has been continuously guided and supported by his proficiency, enthusiasm and interest. A special thank you to him for his understanding and constant inspiration. It was a real pleasure to work under his supervision.

I want to say a big thank you to Dr Matthias Schmidt at the Northwest German Forest Research Institute, Department of Forest Growth, Göttingen, for his continual interest in my project and for supplying me with a large set of forest data. It was great to have an opportunity to work with him during my PhD study and I am looking forward to working with him in future.

A great thank you to Dr Nicole Augustin for her interest in my research, and for providing me with additional material on modelling storm and forest data.

I am very grateful to Dr Anthony Robinson for fruitful discussions and for helping me with an R package installation.

A special thank you to Dr Gavin Shaddick for providing papers on modelling cancer data and supplying incinerator data.

Lots of thanks to all administrative staff of the Department of Mathematical Sciences for their hospitality and tireless everyday help.

The biggest thank you to my office mates, Adam, Fynn, Haojie, Jane, Ray, and Tanya (putting them in alphabetic order) for brightening my study. It has been enjoyable and really comfortable to share an office with them.

And the most important thank you I want to send to my mum and my daughter, Bella, just because they have been being there, unselfish, loving, supporting, giving, and much more... THANK YOU!

Contents

Chapter 1

Introduction

1.1 Motivation

A significant problem in applied statistics is to analyze the relationship between a response variable and one or more explanatory variables. Various regression models have been proposed to solve this problem in the literature. Since there are many practical situations when a parametric form of a regression function may not be easily specified, nonparametric smoothing, that is fitting a nonlinear smooth curve to noisy observations, has become very appealing in a wide range of diverse applications. A large amount of statistical literature is devoted to nonparametric smoothing techniques which are based on, for example, kernel smoothing, smoothing splines, or local polynomials.

In many studies it is natural to assume that the relationship of interest obeys certain shape restrictions. Hence, it is desirable to impose shape constraints on that relationship, since unconstrained fitting might be too flexible and give implausible results. A common requirement in many practical situations is a monotonic relationship between an explanatory variable and the response variable. For example, the growth of children over time is known to be increasing, tree height declines with altitude but increases with tree age. Dose-response curves in medicine, the cognitive development of children in social and behavioral sciences research (Bollaerts et al., 2006b), the relationships between price and quantity produced, between daily mortality and air pollution concentration (Leitenstorfer and Tutz, 2007), between body mass index and incidence of heart disease (Dunson, 2005) are other examples where a monotonic relationship is required.

Also in some research areas monotonicity should be assumed together with convexity or concavity. For example, the effect of labour input on quantity produced and the effect of temperature summed over days of the vegetation period on tree height

are assumed to be monotone increasing and concave. Though modelling a monotonic dependence between one explanatory variable and the response variable is of interest on its own, there are many areas of application where the univariate response variable is modelled as a sum of several smooth functions of explanatory variables, only some of which are assumed to be shape constrained. In particular, such problems are very common in ecological and environmental studies.

Various shape constrained smoothing techniques have been proposed in the statistical literature and some of them are briefly described in Section 1.2. Almost all the known algorithms are either complex and computationally intensive or the model smoothness selection is not fully satisfactory. Also computer codes of algorithms are not always available and it might not be straightforward to implement them. In this thesis a new approach to shape constrained smoothing is presented. The new method has been developed within the context of generalized additive models (GAM), and an R package `monogam` implementing it has been written.

1.2 Literature review

The pool adjacent violators algorithm (PAVA) which solves the problem of isotonic regression seems to be the first technique for producing a monotone regression function (Barlow et al., 1972). Kruskal (1965) suggested the isotonic regression technique for determining a monotone transformation of the response variable in linear regression. Due to the way the PAVA method is implemented, isotonic regression may not result in a smooth function. Friedman and Tibshirani (1984) instead proposed a method for exploring a scatterplot by a smooth monotonic function. Their procedure involves a combination of isotonic regression using PAVA and local averaging.

Analogous to Friedman and Tibshirani (1984) a number of authors have come up with methods which are based on two-stage procedures: unconstrained smoothing and monotonization. One of the modifications of the monotone smoothing approach proposed by Friedman and Tibshirani (1984) was developed by Mammen (1991). He applied a kernel estimator for unconstrained smoothing and used the PAVA to get a monotone fit. Asymptotic equivalency of this estimator with another one obtained by interchanging smoothing and monotonization steps has also been shown (Mammen, 1991). Other types of shape restrictions were considered by Mammen and Thomas-Agnan (1999). Mammen et al. (2001) developed a framework in which a monotone smoother of a regression function is defined as projection of an unconstrained estimator onto the constrained subset of functions. Ghosh (2007) concentrated on a binary regression model with a single monotone smooth term plus a parametric component.

The two-step fitting procedure suggested in this paper consists of unconstrained fitting using the likelihood-based approach of generalized linear mixed models and isotonizing the fitted curve using the PAVA algorithm. As an alternative to two-stage procedures Pal and Banerjee (2008) developed a direct algorithm to fit a single monotone smooth generalized regression model by a piecewise linear function. The approach of Mukerjee (1988) has the reverse procedure, firstly, it isotonizes the data and then smooths the resulted curve using a suitable kernel.

Smoothing by spline functions forms a basis for other procedures for estimating functions with monotonicity restrictions. Most methods use linearly constrained optimization in order to secure monotonicity. Ramsay (1988) introduced integrated splines which yield a monotone smoother when the spline coefficients are nonnegative. This nonnegativity can be imposed by setting linear inequality constraints in an optimization algorithm. A penalized minimization for fitting data by monotonic cubic smoothing splines based on a piecewise polynomial representation was presented in Wood (1994). In this approach the sufficient conditions for monotonicity of a cubic spline were used as linear constraints in quadratic programming. Assuming that an unknown smooth function, $f(x)$, has thrice continuous derivatives Zhang (2004) rewrote $f(x)$ in terms of its derivative, $f'(x)$, with the integrated square of the third derivative taken as a penalty. Such a representation allows imposition of non-negativeness or non-positiveness on $f'(x)$ to get the monotonicity constraint. Using full rank cubic smoothing splines $f'(x)$ is estimated directly, and $f(x)$ indirectly in this approach. In 1998, Ramsay suggested another technique for estimating a twice differentiable strictly monotonic function by solution of a homogeneous linear differential equation. The procedure is unconstrained and includes a penalty term of similar form to the cubic spline smoothing penalty, with the smoothing parameter selected by cross validation. However, the optimization algorithm considered there is computationally expensive and the technique for smoothing parameter selection is not satisfactory. Wang (2000) has extended the monotone smoothing method of Ramsay (1988, 1998) to a single smooth generalized regression model. The fitting procedure developed by Wang is a modification of iteratively reweighted least squares consisting of two steps for estimating two types of model coefficients arising from the smooth specification of Ramsay (1998). Schipper and Taylor (2008) proposed a generalized single monotone functional mixed model with constraints imposed on a smooth specification which can be fitted by maximum likelihood. An extension of monotone regression splines proposed in Ramsay (1998) to cubic monotone and convex constraints has been introduced in Meyer (2008).

Since the B-spline basis attracts a lot of interest in nonparametric smoothing, due to its flexibility and local support (de Boor, 1978, Eilers and Marx, 1996), several

B-spline monotone regression approaches have been suggested. An appealing feature of B-splines is that to obtain monotone increasing (decreasing) spline it is sufficient to guarantee a nondecreasing (nonincreasing) sequence of spline coefficients. The first method based on B-splines was proposed by Kelly and Rice (1990). The authors appeal for a nondecreasing sequence of spline coefficients as constraints in quadratic programming. However, this work selects the level of smoothness on the ad hoc basis of examining graphical displays of data. Another procedure built on a constrained linear programming algorithm and least absolute deviation fit criterion was proposed by He and Shi (1998). It is based only on quadratic B-splines, in order to obtain linear constraints for the proposed linear programming. The smoothness of the fitted function is determined by knot selection using a stepwise knot deletion process. Rousson (2008) implemented the monotone B-splines of Ramsay (1988) with a knot selection algorithm based on a sequence of F-statistics (Friedman and Silverman, 1989) which led to a least squares problem with linear inequality constraints. The recent idea of Bollaerts et al. (2006b) was to set additional asymmetric discrete penalties on n^{th}- order differences of the model coefficients in P-spline regression, in order to restrict the sign of the n^{th}-order derivative of the smooth function. The method, described further in Section 2.7.2, allows placement of different shape constraints on the fitted curve by penalizing differences reflecting n^{th}- order derivatives, but the selection of the constraint weights is not discussed and neither is the computational cost involved.

A Bayesian version of P-splines has been proposed in Lang and Brezger (2004). Following this Bayesian approach Brezger and Steiner (2008) developed monotone Bayesian P-splines by introducing indicator function to truncate the prior distribution of spline coefficients in order to achieve an ordered sequence of coefficients. This method has been implemented in the free software package `BayesX` (Brezger et al., 2005). Monotonic regression within the Bayesian framework has also been considered by Holmes and Heard (2003); Dunson and Neelon (2003), and Dunson (2005). Dunson and Neelon (2003) proposed a Bayesian approach for meeting monotonicity constraints in generalized linear models only. Their approach is based on isotonic transformation of draws from an unconstrained posterior density. A Bayesian isotonic regression for a piecewise-constant model has been introduced in Holmes and Heard (2003). The authors suggested placing a prior distribution on the number and location of change points in the model and use Markov Chain Monte Carlo simulation to sample the unconstrained model space, which then is reduced to a monotonic model space. An alternative Bayesian approach for count data, with a prior distribution specification that imposes nondecreasing constraints, has been proposed by Dunson (2005).

There are other approaches to non(semi)parametric regression which focus on con-

ditional quantile functions rather than on conditional mean functions. A few papers have developed methods to incorporate monotonicity in quantile regression (Koenker et al., 1994; Takeuchi et al., 2006; Bollaerts et al., 2006a). To meet the monotonicity or convexity constraints Koenker et al. (1994) proposed to add corresponding linear inequality constraints to an L_1 fitting problem (minimization of the sum of absolute deviations). R package `quantreg` implements an extension of this approach to additive quantile regression models. In Takeuchi et al. (2006) monotonicity was obtained by imposing additional constraints on the derivative of the smooth term in a quadratic programming method for quantile estimation. Analogous to the P-splines with additional asymmetric discrete penalties approach, proposed by Bollaerts et al. (2006b) for ordinary regression, Bollaerts et al. (2006a) suggested including asymmetric penalties in terms of absolute values of the n^{th}- order differences of the spline coefficients into the L_1- norm of quantile regression. He and Ng (1999) developed a constrained B-spline smoothing algorithm in the context of the quantile regression based on the L_1 projection ideas of He and Shi (1994, 1998). This algorithm has been implemented in an R package `cobs99` and uses only linear and quadratic splines. An improved version of this package, `cobs`, has been developed by Ng and Maechler (2007).

Alternative techniques are based on non-spline methods. For example, they include those of Antoniadis et al. (2007) who focused on a penalized wavelet regression which leads to a convex optimization problem with linear constraints. Constrained nonparametric kernel regression techniques have been considered by, for example, Hall and Huang (2001); Dette et al. (2006). Dette and Pilz (2006) also discussed and compared different monotone kernel regressors.

Besides statistics, the problem of shape constrained smooth curve representation has also aroused interest in Computer Science, Operations Research, Numerical Analysis, Management Science, Biology, etc (see, for example, Sarfraz, 2000, 2003; Matzkin, 1991; Elfving and Andersson, 1988; Dent, 1973; Demetriou, 2004a,b; Vassiliou and Demetriou, 2005; Beatson, 1982; Kopotun et al., 2008, and references therein). Various authors have worked in the area of shape-preserving interpolation, see, e.g., Andersson and Elfving (1987); Hornung (1980); Irvine et al. (1986); Sarfraz (2000, 2003); McAllister and Roulier (1981); Lahtinen (1996) among others. An algorithm which uses piecewise rational cubic functions was introduced by Sarfraz (2000), while Sarfraz (2003) suggested an alternative approach reducing to a rational quadratic interpolant. McAllister and Roulier (1981) and Lahtinen (1996) have utilized quadratic interpolation methods for shape-constrained curves. A Newton-type procedure for the interpolation problem with constraints on curve derivatives has been proposed in Andersson and Elfving (1987). Other numerical algorithms for constrained interpolation can be found

in Hornung (1980) and Irvine et al. (1986).

Other authors in the non-statistical literature have solved shape constrained approximation problem using smoothing splines, such as B-splines or others. A Newton type algorithm for obtaining convex and convex plus monotone smoothers which made use of linear B-splines was introduced in Elfving and Andersson (1988). Demetriou (1991); Vassiliou and Demetriou (2005); and Demetriou (2004b) addressed smoothing problems with different constraints by applying B-spline bases and developing quadratic programming methods. A convex programming method for piecewise convex-concave approximation was suggested by Demetriou (2004a). Schmidt and Scholz (1990) formulated an unconstrained dual problem for convex-concave smoothing using cubic splines. Other smoothing algorithms to fit constrained splines based on Jackson type estimates have been suggested in Beatson (1982) and Kopotun et al. (2008).

In spite of the diverse existing approaches to shape preserving smoothing, there is still a need for a flexible modelling approach which is able to describe practical situations, has a straightforward underlying theory for fitting, smoothness selection and interval estimation, and is implemented in a user-friendly way in a programming language standard for practical statistical analysis, such as R. The purpose of this project was to attempt to meet these requirements.

1.3 The main achievements of the thesis

The following summarizes the main achievements of the thesis.

1. A penalized smoother with monotonicity restriction based on B-splines is proposed. A penalized likelihood maximization method for fitting a generalized regression model subject to monotonicity constraint is developed, based on a Newton-Raphson method.

2. A smoothing method is proposed under other shape constraints such as, monotonicity (monotone increasing and decreasing) together with convexity/concavity for a smooth function of a single covariate. Smooths of two covariates with monotonicity constraints, where monotonicity may be assumed on only one of the covariates (single monotonicity) or on both of them (double monotonicity), is developed.

3. The proposed shape constrained smoothing is extended to generalized additive models with a mixture of unconstrained and shape constrained smooth terms (mono-GAM). A fitting procedure for a mono-GAM is developed. It is based on an outer quasi-Newton iteration to update multiple smoothing parameters, with

each step of this procedure requiring an inner, Newton based, penalized iteratively reweighted least squares scheme to obtain model coefficients. We propose an efficient way for calculating derivatives of the coefficients with respect to the smoothing parameters by extending the approach introduced in Wood (2011). Since a major challenge of any flexible regression approach is its implementation in a computationally efficient and stable manner, such issues as convergence, rank deficiency of the model matrix, initialization, and others are thoroughly discussed.

4. Bayesian confidence intervals for the shape constrained terms of mono-GAM are derived using the delta method. The performance of the proposed confidence intervals has been examined by simulation studies. The realized coverage probabilities were taken as a measure of their performance.

5. An R package, `monogam`, which implements the proposed shape constrained modelling within GAM has been written.

6. Performance and timing comparisons of mono-GAM with other alternative methods is undertaken. The simulation studies have shown that the new method has practical advantages over the considered alternatives.

7. We demonstrate the efficacy and practicality of mono-GAM in real applications. Three data sets with sample sizes ranging from 44 to 29,324 have been successfully analyzed.

1.4 The structure of the thesis

The current introduction chapter continues with a brief overview of generalized additive models (GAM) which serves as a background to the proposed mono-GAM.

In Chapter 2 smoothing under monotone increasing constraint, based on B-spline basis functions with a 'wiggliness' penalty based on the P-spline ideas of Eilers and Marx (1996) is introduced. A stable and efficient method for penalized likelihood estimation of monotone generalized regression model is developed, with a degree of model smoothness selected by direct minimization of the generalized cross validation or similar criteria. Some illustrative simulated data examples are presented in the last section of this chapter. Also in Section 2.7 we briefly discuss two alternative approaches to monotone smoothing: A constrained quadratic programming approach (Kelly and Rice, 1990; Wood, 1994) and P-spline regression with additional asymmetric penalties (Bollaerts et al., 2006b). Mono-GAM will be compared with these alternatives in Chapter 7.

In Chapter 3 penalized smoothing under other shape constraints such as, monotone decreasing constraint and mixed constraints (monotonicity plus convexity/concavity) for smooths of a single covariate is introduced. Then smoothing of bivariate functions with monotonicity restrictions on both covariates (double monotonicity) and on only one of them (single monotonicity) is developed based on tensor product smooths. Penalties for all these shape-preserving smoothers are obtained. To show the performance of the proposed smoothers several simulation examples are presented at the end of the chapter.

Chapter 4 generalizes the proposed approach to generalized additive models with shape constraints on some terms (mono-GAM). For simplicity of presentation, the discussion starts with an additive model with monotonicity constraint imposed only on one smooth term, and only B-spline bases are used for representation of unconstrained smooth terms. This is extended to a more general structure of mono-GAM, which can incorporate any available penalized regression spline basis to represent each unconstrained term, including multivariate terms, and includes bivariate terms with monotonicity constraints. The fitting procedure of a mono-GAM is thoroughly discussed. Some simulated examples are given in the last section of this chapter, to illustrate the performance of mono-GAM.

In Chapter 5 a question about the limiting distribution of the model parameters is solved, which allows construction of Bayesian confidence intervals of the mono-GAM smooth terms by means of the delta method. The performance of the proposed confidence intervals is examined by assessing realized coverage probabilities of the proposed intervals using simulation studies.

Chapter 6 describes the design and usage of the R package `monogam` which implements the proposed generalized additive modelling with monotonicity restrictions on some smooth terms. The model setup is similar to `mgcv(gam)` with the addition of shape constrained smooths. In order to be consistent with the unconstrained GAM the package provides similar key functions to `mgcv`, which are demonstrated on some simulation examples.

A more extensive simulation study is presented in Chapter 7, to illustrate the performance of mono-GAM. Comparison with unconstrained GAM, the quadratic programming approach to shape preserving smoothing (Wood, 1994), and constrained P-splines regression (Bollaerts et al., 2006b) is undertaken here. Simulated examples on univariate single smooth term models, bivariate single smooth models, and additive models with a mixture of unconstrained and monotone smooth terms are considered for evaluation of the performance of the four different approaches, and for timing comparisons.

Chapter 8 presents applications of mono-GAM to various data sets. In the first example a small sample of 44 spatial data is analyzed. The goal of this analysis is to investigate whether proximity to municipal incinerators in Great Britain increases the risk of stomach cancer (Shaddick et al., 2007). The second application uses data from the National Morbidity, Mortality, and Air Pollution Study (Peng and Welty, 2004). The relationship between daily death rate in Chicago and air pollution levels is investigated. There are about 5000 daily measurements in the second data set. Modelling these data assumes that death rate increases with increase in levels of ozone, sulphur dioxide, and levels of particular matter. The third example studies a prediction of tree height as a function of tree diameter and additional tree-stand-level parameters. The large cross-sectional sample of these data (29 324 tree observations) is from the Northwest German Forest Research Institute, Department of Forest Growth, Göttingen, Germany, and was kindly made available by Dr. Matthias Schmidt. The height-diameter model introduced in this chapter includes strictly parametric model components and both monotonic and unconstrained smooth terms.

Throughout the thesis matrices and vectors are boldfaced.

1.5 GAM overview

Unconstrained generalized additive models (GAM) (Hastie and Tibshirani, 1986, 1990; Wood, 2006a) are used extensively in practical applications for modelling nonlinear relationships between a response variable and multiple covariates. As a new approach to shape constrained smoothing, proposed in this thesis, is merged with a GAM framework, a brief overview of GAM is needed for later reference.

Suppose, Y_i, $i = 1, \ldots, n$, are independent observations of a response variable from an exponential family distribution and $x_{1i}, x_{2i}, \ldots, x_{pi}$ are possible explanatory variables. Generalized additive modelling suggests that the mean value of Y_i is linked to an additive, possibly nonlinear, effect of explanatory variables through a known link function. The model may be written as follows

$$g(\mu_i) = \mathbf{X}_i^* \boldsymbol{\delta} + \sum_{j=1}^{p} f_j(x_{ji}), \tag{1.1}$$

where g is a known smooth monotone link function, $\mu_i = \mathrm{E}(Y_i)$, $\mathbf{X}_i^*$ is the i^{th} row of a model matrix for any strictly parametric effects, with corresponding unknown vector of parameters $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_{q_0})^T$, and $f_j(x_{ji})$ are smooth unknown functions of the covariates, x_{ji} may be a vector quantity. The right-hand side of (1.1) is called the linear predictor and is usually denoted as $g(\mu_i) = \eta_i$.

Under the model it is assumed that Y_i are independent random variables from the exponential family of distributions with the probability density function

$$f_{\theta_i}(y_i) = \exp\left[\{y_i\theta_i - b_i(\theta_i)\}/a_i(\phi) + c_i(y_i, \phi)\right], \tag{1.2}$$

where a_i, b_i, and c_i are arbitrary functions, ϕ an arbitrary 'scale' parameter, and θ_i a 'canonical parameter' of the distribution related to $\eta(\mathbf{x})$ via the relationship $\mathrm{E}(Y_i) = b'(\theta_i)$ (see Wood, 2006a). The scale parameter ϕ is assumed to be constant for all observations.

In comparison with parametric regression, model (1.1) allows for much more flexibility in building the relationship between the response and explanatory variables. Many practical situations exist where strictly parametric model specifications do not provide an appropriate fit to the data. However, if when fitting a parametric model we have one task which is to estimate unknown model coefficients, two additional tasks arise in a nonparametric regression such as GAM: how to represent smooth functions f_i, and how to choose their smoothness.

Different approaches to fitting a GAM have been developed:

- The 'backfitting' technique,
- The generalized smoothing spline approach,
- The penalized regression smoothing spline approach,
- The Bayesian P-spline approach to GAM.

The first method was proposed by Hastie and Tibshirani (1986, 1990) where GAM originated. Their 'backfitting' technique is an iterative procedure of smoothing partial residuals in order to estimate each smooth model component. Hastie and Tibshirani (2000) also proposed a Bayesian backfitting procedure to GAM. The generalized smoothing spline approach (Wahba, 1990; Gu, 2002) is another alternative for GAM estimation. The underlying theory of this method is not as straightforward to understand as it is for penalized regression smoothing splines. The theory of penalized regression smoothers and their practical applications are given in Wood (2006a). The Bayesian framework to GAM has been also developed (e.g. Fahrmeir et al., 2004; Lang and Brezger, 2004). While this approach has fully Bayesian inference and uses Markov chain Monte Carlo techniques, the representation of GAM components has been built on penalized smoother ideas.

In this project the penalized regression spline approach is employed which can be split into three stages: i) representation of smooth model terms using penalized regression splines, ii) model coefficient estimation by penalized log likelihood maximization,

and iii) smoothness selection by minimization of the generalized cross validation score (or similar criteria). This framework for unconstrained GAM has been developed and thoroughly discussed in Wood (2006a). The rest of this section will briefly outline the basic ideas of this approach.

To solve the first problem of representing the smooth functions in (1.1), various penalized regression smoothers are available, such as cubic regression splines and P-splines, for representing smooths of a single covariate; or thin plate regression splines and tensor product smooths for smooth of several covariates. The idea is to specify a basis for each function and choose an appropriate set of basis functions, B_{jk}, so that the j^{th} smooth function can be represented as

$$f_j(x_j) = \sum_{k=1}^{q_j} B_{jk}(x_j)\beta_{jk},$$

where β_{jk} are coefficients to be estimated, and q_j is a number of basis functions. In vector-matrix notation each smooth term may be written as

$$\mathbf{f}_j = \mathbf{X}_j\boldsymbol{\beta}_j,$$

where $\mathbf{f}_j$ is a vector with $\mathbf{f}_{ji} = f_j(x_{ji})$, $\boldsymbol{\beta}_j = (\beta_{j1}, \beta_{j2}, \ldots, \beta_{jq_j})^T$, and the i^{th} row of the model matrix $\mathbf{X}_j$ is $\mathbf{X}_{ji} = \left\{B_{j1}(x_{ji}), B_{j2}(x_{ji}), \ldots, B_{jq_j}(x_{ji})\right\}$.

The model (1.1) is usually not identifiable. In order to deal with the identifiability problem a 'centering' constraint (Wood, 2006a) may be imposed on each smooth. This problem will also be discussed in Section 4.1.1. Having solved the identifiability problem, the model can be written as

$$g(\mu_i) = \mathbf{X}_i\boldsymbol{\beta}, \tag{1.3}$$

where $\boldsymbol{\beta}^T = \left[\boldsymbol{\delta}^T, \boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T, \ldots, \boldsymbol{\beta}_p^T\right]$, and $\mathbf{X} = [\mathbf{X}^* : \mathbf{X}_1 : \mathbf{X}_2 : \ldots : \mathbf{X}_p]$ is the $n \times q$ model matrix with q less than n.

Since the model (1.3) is represented now as a generalized linear model (GLM), $\boldsymbol{\beta}$ can be estimated by likelihood maximization, which in practice is solved by iteratively re-weighted least squares. However, before coming to the details of coefficient estimation, we should solve the problem of controlling the smoothness of each smooth term. If too many basis functions are taken for f_j representation (q_j is large), then the model may be overfitted, and conversely, too small number of q_j may result in underfitting. That is why a 'wiggliness' measure, or *penalty*, is used for each smooth term so that the model coefficients are estimated by *penalized* likelihood maximization. The penalty may be

expressed, for example, in terms of the integrated squared second-order derivative of the smooth. Most commonly the penalty is expressed in a quadratic form of the full coefficient vector. For example, the penalty for the j^{th} smooth may be written as $\boldsymbol{\beta}^T\mathbf{S}_j\boldsymbol{\beta}$, where $\mathbf{S}_j$ is a matrix with some known elements.

After setting the penalties for each smooth function, the penalized log likelihood function can be defined as

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\boldsymbol{\beta}^T\mathbf{S}\boldsymbol{\beta}, \tag{1.4}$$

where $l(\boldsymbol{\beta})$ is the log likelihood of the model, $\mathbf{S} = \sum_{j=1}^{p}\lambda_j\mathbf{S}_j$, and λ_j are smoothing parameters which now control the model smoothness (the fraction of $1/2$ is taken for a convenient representation of the derivatives). Given the values λ_j, $\boldsymbol{\beta}$ is estimated by maximizing the penalized log likelihood $l_p(\boldsymbol{\beta})$.

Therefore, we now have two questions, how to maximize (1.4) w.r.t. $\boldsymbol{\beta}$, and how to choose λ_j. Firstly, suppose that the values of λ_j are given and consider the problem of $\boldsymbol{\beta}$ estimation.

To find $\boldsymbol{\beta}$, a score vector $\mathbf{u}_p(\boldsymbol{\beta})$, a vector of the first order derivatives of $l_p(\boldsymbol{\beta})$ w.r.t. $\boldsymbol{\beta}$, should be equated to zero. The equations $\mathbf{u}_p(\boldsymbol{\beta}) = \mathbf{0}$ are non-linear and generally have no analytical solution, so some numerical methods should be applied. In practice, a penalized iteratively re-weighed least squares (P-IRLS) scheme based on Fisher scoring is used to solve these equations. If $\boldsymbol{\beta}^{[k]}$ is the current estimate of $\boldsymbol{\beta}$, then the next Fisher scoring estimate is

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathcal{I}(\boldsymbol{\beta}^{[k]})^{-1}\mathbf{u}_p(\boldsymbol{\beta}^{[k]}), \tag{1.5}$$

where $\mathcal{I}(\boldsymbol{\beta}) = -\mathrm{E}(\mathbf{H}(\boldsymbol{\beta}))$ is the Fisher information matrix, and $\mathbf{H}(\boldsymbol{\beta})$ is the Hessian of the penalized log likelihood function which is not difficult to derive from (1.4).

After substituting the analytical expressions of $\mathcal{I}(\boldsymbol{\beta})$ and $\mathbf{u}_p(\boldsymbol{\beta})$ into (1.5) and applying simple mathematical operations, (1.5) becomes a penalized weighted least squares equation. That is, $\boldsymbol{\beta}^{[k+1]}$ minimizes the following penalized weighted sum of squares

$$\| \sqrt{\mathbf{W}^{[k]}} \left(\mathbf{z}^{[k]} - \mathbf{X}\boldsymbol{\beta} \right) \|^2 + \boldsymbol{\beta}^T\mathbf{S}\boldsymbol{\beta}, \tag{1.6}$$

where $\mathbf{z}^{[k]}$ is a vector of pseudodata with $z_i^{[k]} = g'(\mu_i^{[k]})(y_i - \mu_i^{[k]}) + \mathbf{X}_i\boldsymbol{\beta}^{[k]}$, and $\mathbf{W}^{[k]}$ is a diagonal matrix with diagonal elements

$$w_i^{[k]} = \frac{1}{V(\mu_i^{[k]})g'(\mu_i^{[k]})^2},$$

where $V(\mu_i)\phi = \mathrm{var}(Y_i)$.

Therefore, given the smoothing parameters $\boldsymbol{\lambda}$, to obtain the maximum penalized likelihood estimates, $\hat{\boldsymbol{\beta}}$ the following Fisher scoring algorithm is iterated to convergence:

1. Set initial values: $\mu_i^{[0]} = y_i$, $\eta_i^{[0]} = g(\mu_i^{[0]})$, and set $k = 0$.
2. Evaluate $\mathbf{z}^{[k]}$ and $\mathbf{W}^{[k]}$ using the current values of $\boldsymbol{\mu}^{[k]}$ and $\boldsymbol{\eta}^{[k]}$.
3. Minimize (1.6) w.r.t $\boldsymbol{\beta}$ to find $\boldsymbol{\beta}^{[k+1]}$, and therefore $\boldsymbol{\eta}^{[k+1]} = \mathbf{X}_i\boldsymbol{\beta}^{[k+1]}$ and $\boldsymbol{\mu}^{[k+1]}$. Increment k.
4. Repeat steps 2 and 3 until convergence.

This algorithm is referred to as a P-IRLS scheme (Wood, 2006a). It should be noted that the canonical link function is $g(\mu_i) = b'^{-1}(\mu_i)$ and under the canonical link, the canonical parameter of the exponential family distribution equals the linear predictor, $\theta_i = \eta_i = g(\mu_i)$. Then, it is not difficult to show that $\mathcal{I}(\boldsymbol{\beta}) = -\mathbf{H}(\boldsymbol{\beta})$ and hence, for the canonical link, Fisher scoring and Newton-Raphson method are equivalent.

In order to select values of the smoothing parameters, λ_j, a separate criterion, $\mathcal{V}(\boldsymbol{\lambda})$, expressed as a function of $\boldsymbol{\lambda}$, can be directly optimized. Various criteria have been developed in the literature. One possibility is to minimize criteria based on model prediction error ideas. Such a criterion for the model with a known scale parameter, ϕ, is the Un-Biased Risk Estimator (UBRE) (Craven and Wahba, 1979; Wood, 2006a; Mallows, 1973). For unknown ϕ, the generalized cross validation score (GCV) may be used (Craven and Wahba, 1979; Hastie and Tibshirani, 1990). Another alternative for λ_j selection is to optimize a likelihood based criteria such as maximum marginal likelihood (Anderssen and Bloomfield, 1974) or restricted maximum likelihood (Wahba, 1985; Wood, 2011). Comprehensive discussion and references on smoothness selection criteria are given in (Wood, 2011).

Given an appropriate $\mathcal{V}(\boldsymbol{\lambda})$, a computational method for its optimization should be developed. There are two ways of implementing $\boldsymbol{\lambda}$ estimation. One alternative known as a 'performance oriented iteration' is to update $\hat{\boldsymbol{\lambda}}$ at each P-IRLS step. The main problem with this approach is divergence or cycling of the P-IRLS scheme (Gu, 2002; Wood, 2004, 2006a). Another alternative is based on nested or outer iterations (Wood, 2008, 2011). In this case each step of the $\mathcal{V}(\boldsymbol{\lambda})$ optimization scheme requires inner P-IRLS iterations to convergence to find $\hat{\boldsymbol{\beta}}$ for the current $\hat{\boldsymbol{\lambda}}$. Usually in practice, a Newton or quasi-Newton method is used for $\mathcal{V}(\boldsymbol{\lambda})$ optimization. In this project an outer quasi-Newton iteration is used for minimizing UBRE/GCV to update $\hat{\boldsymbol{\lambda}}$, and each step of this procedure will require an inner Newton-Raphson based P-IRLS to obtain $\hat{\boldsymbol{\beta}}$, given $\hat{\boldsymbol{\lambda}}$ (Section 4.3).

Chapter 2

Single smooth models with monotonicity constraint

This chapter introduces smoothing under monotone increasing constraint based on B-spline basis functions with a 'wiggliness' penalty based on the P-spline ideas of Eilers and Marx (1996). A stable and efficient method for penalized likelihood estimation of a monotone generalized regression model is developed, with the degree of model smoothness selected by direct minimization of the generalized cross validation score or similar criteria. Some illustrative simulated data examples are presented in the last section of this chapter.

2.1 Monotonic P-splines

As was mentioned in the introduction, various techniques of nonparametric monotone smoothing have been developed for a single term Gaussian model. However, models, for example, with binary or count response variables, which assume Binomial or Poisson distributions, are less-discussed in the literature, despite the fact that there are many ecological, economical, and social problems where these models may be applied. In this section a single smooth generalized monotone regression model will be set up using B-spline basis functions. The method will be extended to the GAM context in Chapter 4.

Consider a single smooth generalized regression model

$$g(\mu_i) = f(x_i), \quad i = 1, \ldots, n, \tag{2.1}$$

where

$$\mu_i = \mathrm{E}(Y_i), \quad Y_i \sim \text{some exponential family distribution},$$

Y_i are independent response variables, x_i is a covariate, $f(x_i)$ is a smooth function that satisfies a monotonicity constraint

$$f(x_i) > f(x_j) \text{ if } x_i > x_j$$

(for the monotone increasing function) [1], and g is a known smooth monotonic 'link function'. For simplicity of presentation, only the monotone increasing case will be considered in this chapter. Smoothing under other shape constraints will be discussed in the next chapter. The probability density function of Y_i, in canonical form, can be written as

$$f_{\theta_i}(y_i) = \exp\left[\{y_i\theta_i - b_i(\theta_i)\}/a_i(\phi) + c_i(y_i, \phi)\right], \tag{2.2}$$

where a_i, b_i, and c_i are arbitrary functions, ϕ an arbitrary 'scale' parameter, and θ_i a 'canonical parameter' of the distribution related to $f(x)$ via the relationship $\mathrm{E}(Y_i) = b'(\theta_i)$ (see Wood, 2006a). While the functions a_i, b_i, and c_i may vary with i, the scale parameter ϕ is assumed to be constant for all observations.

To estimate the smooth function $f(x)$ in (2.1) a penalized regression spline basis can be used. In this project the B-spline basis functions are used to represent an unknown monotone smooth function. The B-splines are very attractive due to their smooth interpolation property, flexibility - splines of different orders can be represented, and their local support - the B-spline basis takes positive values between $(m+3)$ adjacent knots, where $(m+1)$ is the order of the B-spline, and zero values otherwise. Given a sequence of evenly spaced knots, $k_1 < k_2 < \ldots < k_{q+m+2}$, where q is a number of basis functions and the spline should be evaluated within the interval $[k_{m+2}, k_{q+1}]$, an $(m+1)^{th}$ order B-spline can be represented as (see De Boor, 1978; Wood, 2006a)

$$f(x) = \sum_{j=1}^{q} B_j^m(x)\gamma_j,$$

where

$$B_j^m(x) = \frac{x - k_j}{k_{j+m+1} - k_j} B_j^{m-1}(x) + \frac{k_{j+m+2} - x}{k_{j+m+2} - k_{j+1}} B_{j+1}^{m-1}(x), \quad j = 1, \ldots, q, \tag{2.3}$$

$$B_j^{-1}(x) = \begin{cases} 1, & k_j \le x \le k_{j+1} \\ 0, & \text{otherwise} \end{cases}$$

and γ_j are unknown parameters. The B-spline of order $(m+1)$ is made up of $(m+$

[1] Since the main purpose of the paper is to develop an efficient computational method of the monotone smoothing, the difference between the strict signs of inequalities ($>$, $<$) and not strict ($\ge$, $\le$) is not meaningful

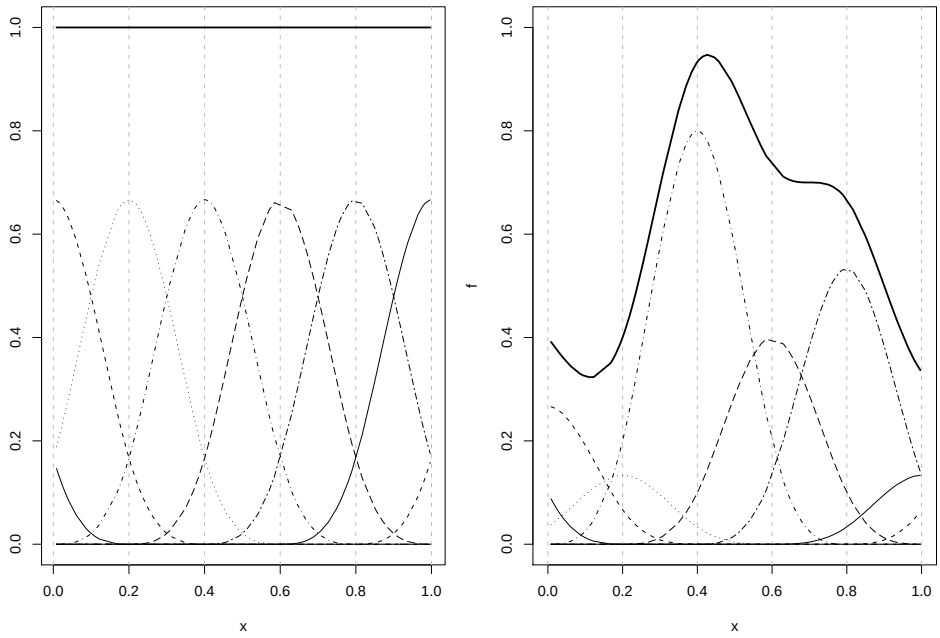


Figure 2-1: Spline regression using the 3^{rd} order B-spline bases.

2) polynomial sections of the $(m+1)^{th}$ order, joined together so that the spline is continuous to m^{th} derivative. And for any value x within $[k_{m+2}, k_{q+1}]$,

$$\sum_{j=1}^{q} B_j^m(x) = 1.$$

It is possible to space knots unevenly but the interpretation of the penalties, which are described at the end of this section, is less clear in such a case.

The left panels of Figures 2-1 and 2-2 illustrate eight B-splines basis functions of the third and forth orders correspondingly. The grey vertical dashed lines show the knot locations. In the right panels the B-spline basis functions multiplied by the corresponding coefficients are illustrated by thin curves. The splines obtained by the summation of the basis functions multiplied by the coefficients are represented by the thick solid lines.

Following De Boor (1978) the first order derivative of the B-spline with equally spaced knots is

$$f'(x) = \frac{1}{h} \sum_{j=2}^{q} B_j^{m-1}(x) \triangle^1 \gamma_j,$$

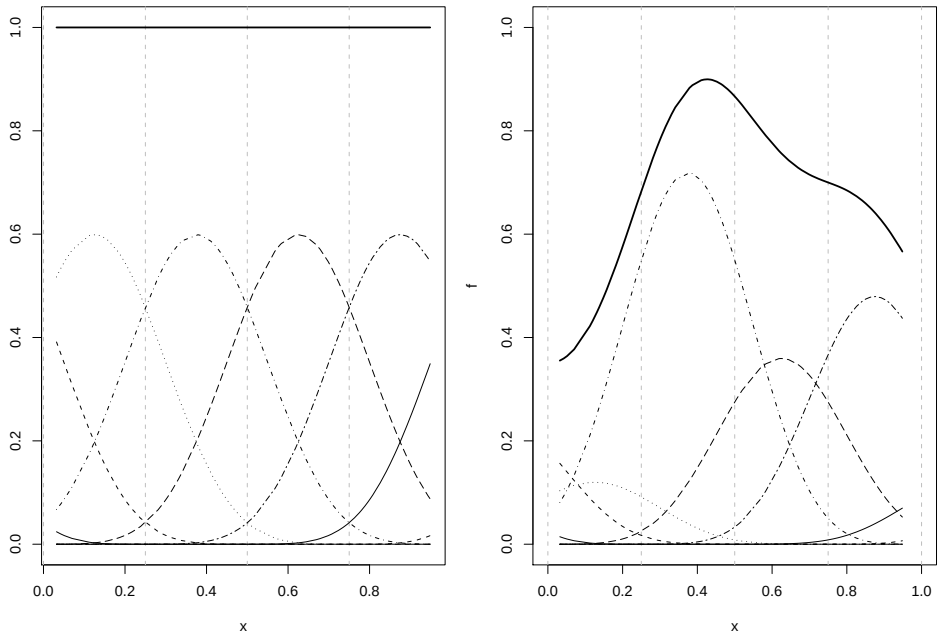


Figure 2-2: Spline regression using the 4^{th} order B-spline bases.

where $\triangle^1\gamma_j$ is the first order difference of the model parameters, and h is the distance between two adjacent knots. Since all B-spline basis functions are nonnegative by definition, a sufficient condition for $f'(x) > 0$ is

$$\triangle^1\gamma_j = \gamma_j - \gamma_{j-1} > 0 \ . \tag{2.4}$$

Therefore, an increasing sequence of all model parameters $\gamma_j,\ \ j = 1, \ldots, q$, will produce a monotonically increasing function. It should be mentioned that the monotonic smoothers of Kelly and Rice (1990), Bollaerts et al (2006), and Leitenstorfer and Tutz (2007) were developed on the same concept.

In this project, to achieve (2.4), the constrained model coefficients, γ_j, are redefined as

$$\gamma_1 = \beta_1, \ \ \gamma_j = \beta_1 + \sum_{i=2}^{j} \exp(\beta_i), \ \ j = 2, \ldots, q, \tag{2.5}$$

where the β_i's are unknown unconstrained parameters. In Figure 2-3 one can see the representation of the monotone increasing smooth curves using eight B-spline basis functions of the third (on the left panel of the figure) and fourth (on the right panel of the figure) orders.

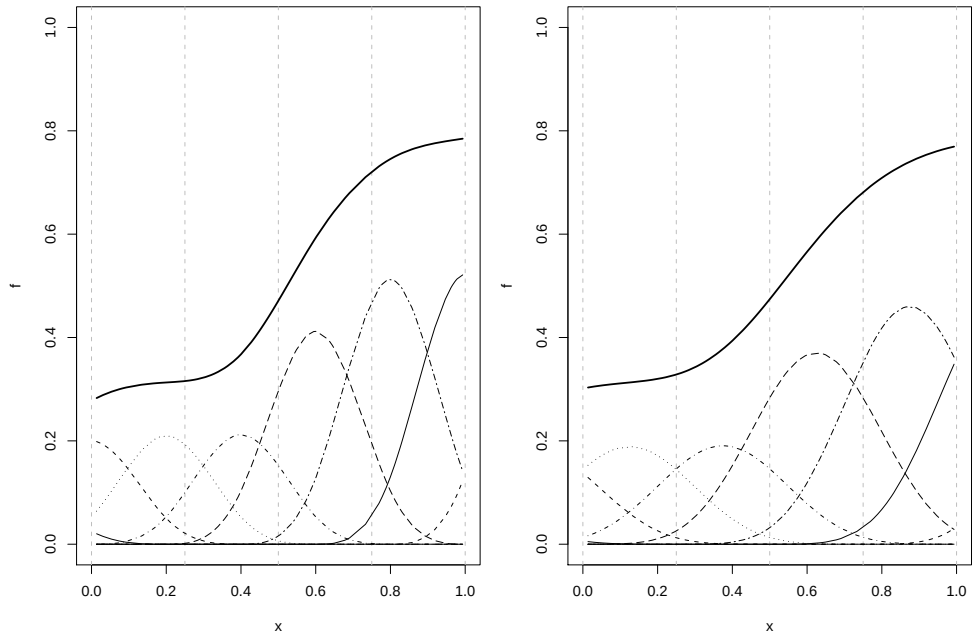


Figure 2-3: Illustration of the monotone increasing smooth curves using the third (left panel) and fourth (right panel) order B-spline bases.

Then the corresponding monotone smooth generalized model may be written as

$$g(\mu_i) = \mathbf{X}_i \boldsymbol{\Sigma} \tilde{\boldsymbol{\beta}}, \tag{2.6}$$

where $\eta_i = \mathbf{X}_i \boldsymbol{\Sigma} \tilde{\boldsymbol{\beta}}$ is called the linear predictor,

$$\mathbf{X}_i = \left\{ B_1^m(x_i), B_2^m(x_i), ..., B_q^m(x_i) \right\}$$

is the i^{th} row of the model matrix $\mathbf{X}$,

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix} \tag{2.7}$$

is a $q \times q$ matrix, and

$$\tilde{\boldsymbol{\beta}} = (\beta_1, \exp(\beta_2), \exp(\beta_3), ..., \exp(\beta_q))^T. \tag{2.8}$$

Throughout this thesis, $\tilde{\boldsymbol{\beta}}$ is referred to as a vector of the model parameters (or coefficients), while $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_q)^T$ is referred to as a vector of the unconstrained working model parameters or just a vector of the working parameters (coefficients).

Given $\mathbf{y}$, a vector of observations of the mutually independent response variables Y_i, maximum likelihood estimation of $\boldsymbol{\beta}$ is possible. The log likelihood function of $\boldsymbol{\beta}$ to be optimized is

$$l(\boldsymbol{\beta}) = \log \prod_{i=1}^{n} f_{\theta_i}(y_i) = \sum_{i=1}^{n} \log f_{\theta_i}(y_i),$$

where the probability density function $f_{\theta_i}(y_i)$ was defined in (2.2). However, when using intermediate rank smoothers, such as B-splines, a question about the degree of model smoothness arises. The degree of smoothing here is controlled by the basis dimension, q. To overcome the problem of overfitting with too many B-splines or underfitting when there is insufficient number of basis functions, B-splines with a 'wiggliness' penalty based on the P-spline ideas of Eilers and Marx (1996) may be used with a 'generous' number of basis functions.

P-splines are intermediate rank penalized smoothers with a k^{th}- order difference penalty applied directly to the working parameters β_j. For the monotone P-splines the first-order difference penalty starting with the second working parameter is used

$$P = \sum_{j=2}^{q-1} (\beta_{j+1} - \beta_j)^2 = \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta}, \tag{2.9}$$

where

$$\mathbf{S} = \begin{pmatrix} 0 & 0 & 0 & 0 & . & . \\ 0 & 1 & -1 & 0 & . & . \\ 0 & -1 & 2 & -1 & . & . \\ 0 & 0 & -1 & 2 & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \end{pmatrix}. \tag{2.10}$$

Such a penalization is intuitively sensible, since by keeping unconstrained parameters close to each other (starting with the second one), we have similar increments in the model coefficients, γ_j (see (2.5)), and the resulted B-spline becomes a linear function. On the other hand, having a generous number of basis functions, q, and making no restrictions on β_j, we get an un-penalized curve with the greatest 'wiggliness' possible for the given construction. Hence, rather than estimating $\boldsymbol{\beta}$ by maximizing $l(\boldsymbol{\beta})$, it can

be estimated by maximizing the penalized log likelihood function

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\lambda \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta}, \tag{2.11}$$

where λ is called as a *smoothing parameter* since it balances the trade off between the model fit and model smoothness. Thus, by varying the values of the smoothing parameter between $\lambda = 0$ and $\lambda \to \infty$ the estimate for $f(x)$ changes from a straight line fit to an un-penalized estimate. Since the first coefficient, $\gamma_1 = \beta_1$, is the model intercept, the penalization is started from β_2. Figure 2-4 illustrates how smoothness of the monotone fit of the simulated data changes for five different values of the smoothing parameter, $\lambda_1 = 1e-4$, $\lambda_2 = 0.005$, $\lambda_3 = 0.01$, $\lambda_4 = 0.1$, and $\lambda_5 = 100$. Twenty five B-spline basis functions of the third order were used for this example.

So, given λ, the penalized regression spline fitting problem is to maximize the penalized log likelihood function (2.11) with respect to $\boldsymbol{\beta}$ (the constant 1/2 is included for later convenience).

The estimation of the smoothing parameter will be discussed in Section 2.6. For the next three sections λ is treated as known and the problem of $\boldsymbol{\beta}$ estimation by penalized likelihood maximization is considered.

2.2 A Newton method for penalized likelihood estimation of monotone smooth generalized regression models

This section describes a Newton (Newton-Raphson) method for maximizing the penalized likelihood of a monotone smooth generalized regression model. As was mentioned in the introduction, for the unconstrained GAM the P-IRLS based on Fisher scoring, rather than a Newton method, is usually used for the penalized likelihood estimation. However, initial investigations on a proposed monotone model showed that Fisher scoring may require step length reductions at the end of the iterative procedure and converge very slowly. This is possibly due to the non-linearity of the objective in the working model coefficients and the presence of the non-canonical link function. The same problem may arise while fitting the common unconstrained GAM with a non-canonical link function (Wood, 2011). So, a full Newton method is applied to optimize the penalized log-likelihood function (2.11) in order to obtain $\hat{\boldsymbol{\beta}}$. The details of the fitting procedure described below may look complex but it is based on simple basic ideas.

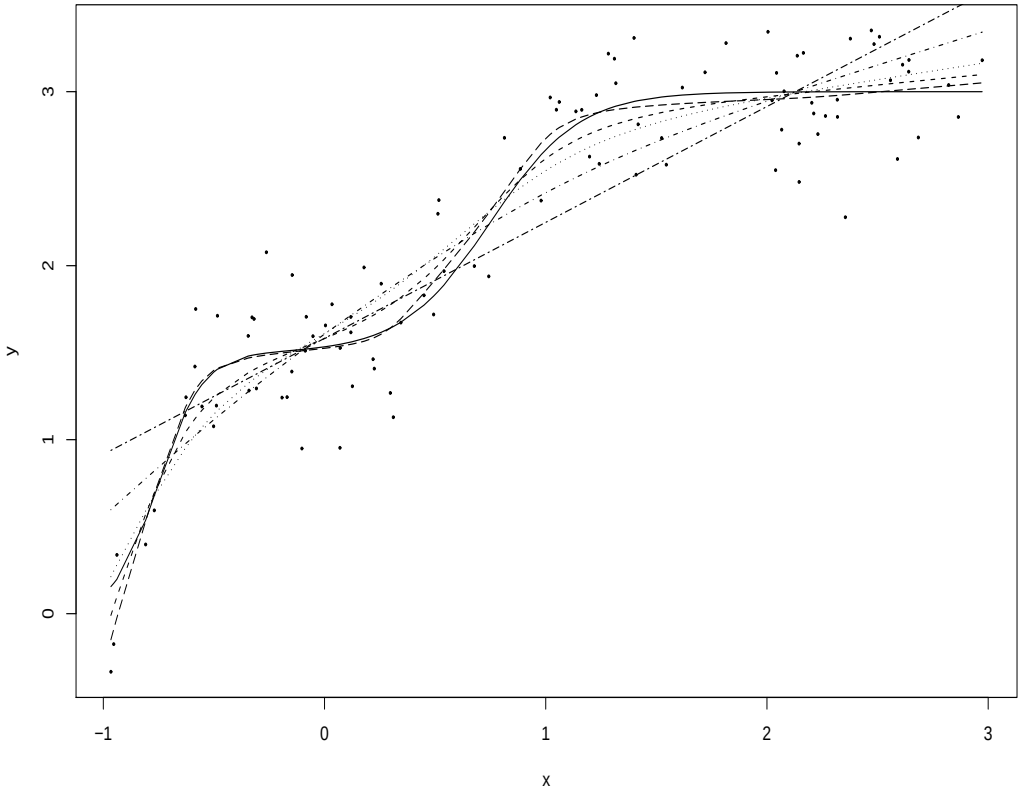


Figure 2-4: Illustration of the monotone P-splines for five values of the smoothing parameter: $\lambda_1 = 1e-4$ (long dashed curve), $\lambda_2 = 0.005$ (short dashed curve), $\lambda_3 = 0.01$ (dotted curve), $\lambda_4 = 0.1$ (dot-dashed curve), and $\lambda_5 = 100$ (two dashed curve). The true curve is represented as a solid line and dots are the simulated data.

The penalized log likelihood function to be maximized w.r.t. $\boldsymbol{\beta}$ is

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\lambda \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta},$$

where from (2.2) the log likelihood of $\boldsymbol{\beta}$ is

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[\{ y_i \theta_i - b_i(\theta_i) \} / a_i(\phi) + c_i(\phi, y_i) \right]. \tag{2.12}$$

The distribution parameters θ_i depend on the working model parameters β_j via the link between the mean of Y_i and θ_i, $\mathrm{E}(Y_i) = b_i'(\theta_i)$. Recall that the smoothing parameter λ is considered to be fixed while estimating $\boldsymbol{\beta}$. Consider only cases where $a_i(\phi) = \phi/\omega_i$, and ω_i is a known constant, which usually equals 1. Almost all probability distributions of interest from the exponential family are covered by such a limitation. Then

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[\omega_i \{ y_i \theta_i - b_i(\theta_i) \} / \phi + c_i(\phi, y_i) \right]$$

and the first order derivative of $l(\boldsymbol{\beta})$ w.r.t. β_j is

$$\frac{\partial l_p}{\partial \beta_j} = \frac{1}{\phi} \sum_{i=1}^{n} \omega_i \left\{ y_i \frac{\partial \theta_i}{\partial \beta_j} - b_i'(\theta_i) \frac{\partial \theta_i}{\partial \beta_j} \right\} - \lambda \mathbf{S}_j \boldsymbol{\beta},$$

where (for this chapter only) $\mathbf{S}_j$ is the j^{th} row of the matrix $\mathbf{S}$, and

$$\frac{\partial \theta_i}{\partial \beta_j} = \frac{\partial \theta_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \beta_j}.$$

Taking the first order derivatives from the both sides of the linking equation $\mathrm{E}(Y_i) = b_i'(\theta_i)$, we get

$$\frac{\partial \mu_i}{\partial \theta_i} = b_i''(\theta_i) \Rightarrow \frac{\partial \theta_i}{\partial \mu_i} = \frac{1}{b_i''(\theta_i)},$$

$$\frac{\partial l_p}{\partial \beta_j} = \frac{1}{\phi} \sum_{i=1}^{n} \frac{\{ y_i - b_i'(\theta_i) \}}{b_i''(\theta_i)/\omega_i} \frac{\partial \mu_i}{\partial \beta_j} - \lambda \mathbf{S}_j \boldsymbol{\beta}. \tag{2.13}$$

Since

$$g(\mu_i) = \mathbf{X}_i \boldsymbol{\Sigma} \tilde{\boldsymbol{\beta}}, \;\; i = 1, \ldots, n,$$

then

$$g'(\mu_i) \frac{\partial \mu_i}{\partial \beta_1} = [\mathbf{X}\boldsymbol{\Sigma}]_{i1}, \;\; g'(\mu_i) \frac{\partial \mu_i}{\partial \beta_j} = [\mathbf{X}\boldsymbol{\Sigma}]_{ij} \exp(\beta_j), \;\; \text{for } \; j = 2, \ldots, q.$$

Hence

$$\frac{\partial \mu_i}{\partial \beta_1} = \frac{[\mathbf{X\Sigma}]_{i1}}{g'(\mu_i)}, \;\; \frac{\partial \mu_i}{\partial \beta_j} = \frac{[\mathbf{X\Sigma}]_{ij} \exp(\beta_j)}{g'(\mu_i)}, \;\; \text{for } \; j = 2, \ldots, q.$$

Another key point of the exponential family concerns the variance

$$\text{var}(Y_i) = b_i''(\theta_i) a_i(\phi) = b_i''(\theta_i)\phi/\omega_i,$$

which is represented in the theory of GLMs in terms of μ_i as $\text{var}(Y_i) = V(\mu_i)\phi$, where $V(\mu_i) = b_i''(\theta_i)/\omega_i$.

Let $\mathbf{G}$ and $\mathbf{W}_1$ be $n \times n$ diagonal matrices with the diagonal elements $G_i = g'(\mu_i)$ and

$$w_{1i} = \frac{\omega_i}{V(\mu_i) g'^2(\mu_i)},$$

and let $\mathbf{C}$ be a $q \times q$ diagonal matrix with

$$\text{diag}(\mathbf{C}) = (1, \exp(\beta_2), \ldots, \exp(\beta_q)).$$

Then a penalized score vector may be written as

$$\mathbf{u}_p(\boldsymbol{\beta}) = \frac{\partial l_p}{\partial \boldsymbol{\beta}} = \frac{1}{\phi}(\mathbf{X\Sigma C})^T \mathbf{W}_1 \mathbf{G}(\mathbf{y} - \boldsymbol{\mu}) - \lambda \mathbf{S}\boldsymbol{\beta}. \tag{2.14}$$

To find the working model parameters estimates, $\hat{\boldsymbol{\beta}}$, one needs to solve $\mathbf{u}_p(\boldsymbol{\beta}) = \mathbf{0}$. These equations are non-linear and have no analytical solution, so some numerical methods should be applied. In the case of unconstrained GAM the penalized iteratively reweighed least squares (P-IRLS) scheme based on Fisher scoring is used to solve these equations (see Introduction).

To proceed the Hessian of the log-likelihood function is derived from (2.14)

$$\mathbf{H}(\boldsymbol{\beta}) = \left[\frac{\partial^2 l_p}{\partial \boldsymbol{\beta}_j \partial \boldsymbol{\beta}_k}\right] = -\frac{1}{\phi}(\mathbf{X\Sigma C})^T \mathbf{W} \mathbf{X\Sigma C} + \frac{1}{\phi}\mathbf{E} - \lambda \mathbf{S}, \tag{2.15}$$

where $\mathbf{W}$ is a diagonal matrix with

$$w_i = \frac{\omega_i \alpha_i}{V(\mu_i) g'^2(\mu_i)}, \;\; \text{and} \;\; \alpha_i = 1 + (y_i - \mu_i)\left\{\frac{V'(\mu_i)}{V(\mu_i)} + \frac{g''(\mu_i)}{g'(\mu_i)}\right\}, \tag{2.16}$$

$\mathbf{E}$ is a $q \times q$ diagonal matrix with

$$E_1 = 0 \;\; \text{and} \;\; E_j = \sum_{i=1}^{n} \frac{\omega_i \, [\mathbf{X\Sigma C}]_{ij}}{V(\mu_i) g'(\mu_i)}(y_i - \mu_i), \;\; j = 2, \ldots, q. \tag{2.17}$$

Note that for the model with a canonical link function, the second term of α_i is equal to zero, since in this case

$$V'(\mu_i)/V(\mu_i) + g''(\mu_i)/g'(\mu_i) = 0.$$

Therefore, $\alpha_i = 1$ and the matrices $\mathbf{W}_1$ and $\mathbf{W}$ are identical.

So, using the Newton method, if $\boldsymbol{\beta}^{[k]}$ is the current estimate of $\boldsymbol{\beta}$, then the next estimate is

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \left\{(\mathbf{X}\boldsymbol{\Sigma}\mathbf{C}^{[k]})^T\mathbf{W}^{[k]}\mathbf{X}\boldsymbol{\Sigma}\mathbf{C}^{[k]} - \mathbf{E}^{[k]} + \lambda\mathbf{S}\right\}^{-1}\left\{(\mathbf{X}\boldsymbol{\Sigma}\mathbf{C}^{[k]})^T\mathbf{W}_1^{[k]}\mathbf{G}^{[k]}(\mathbf{y} - \boldsymbol{\mu}^{[k]}) - \lambda\mathbf{S}\boldsymbol{\beta}^{[k]}\right\}, \tag{2.18}$$

where the scale parameter ϕ is absorbed into the smoothing parameter λ.

To use (2.18) directly for $\boldsymbol{\beta}$ estimation is not efficient since explicit formation of the Hessian would square the condition number of the working model matrix, $\sqrt{\mathbf{W}}\mathbf{X}\boldsymbol{\Sigma}\mathbf{C}$ (Golub and van Loan, 1996). The condition number is the ratio of the largest to the smallest eigenvalues which allows determination of whether a system is ill-conditioned (large condition number) and it is desirable to develop a solution method that keeps the condition number as low as possible. Before considering efficient and stable evaluation of $\hat{\boldsymbol{\beta}}$, it should be noted that the Hessian matrix also appears in an expression for the effective degrees of freedom (edf) of the fitted model (used later for the smoothing parameter selection).

2.3 Degrees of freedom

An un-penalized model would have as many degrees of freedom as the number of unconstrained model parameters. However, the use of penalties decreases the number of degrees of freedom so that a model with $\lambda \to \infty$ would have the degrees of freedom near 1. Using the concept of the divergence of the maximum likelihood estimator, the effective degrees of freedom of the penalized fit can be found as (Meyer and Woodroofe, 2000; Wood, 2001)

$$\tau = \mathrm{div}(\hat{\boldsymbol{\mu}}) = \sum_{i=1}^{n} \frac{\partial}{\partial y_i}\hat{\mu}_i(\mathbf{y}).$$

Substituting (2.18) into (2.6) and taking first-order derivatives with respect to y_i, we get

$$\frac{\partial\hat{\mu}_i}{\partial y_i} = \left[\mathbf{X}\boldsymbol{\Sigma}\mathbf{C}\left\{(\mathbf{X}\boldsymbol{\Sigma}\mathbf{C})^T\mathbf{W}\mathbf{X}\boldsymbol{\Sigma}\mathbf{C} - \mathbf{E} + \lambda\mathbf{S}\right\}^{-1}(\mathbf{X}\boldsymbol{\Sigma}\mathbf{C})^T\mathbf{W}_1\right]_{ii},$$

where the right-hand-side of this expression is the i^{th} diagonal element of the matrix written in the square brackets.

Therefore,

$$\tau = \text{tr}(\mathbf{F}), \tag{2.19}$$

where

$$\mathbf{F} = \left\{(\mathbf{X\Sigma C})^T\mathbf{W}\mathbf{X\Sigma C} - \mathbf{E} + \lambda\mathbf{S}\right\}^{-1}(\mathbf{X\Sigma C})^T\mathbf{W}_1\mathbf{X\Sigma C}$$

and the matrices $\mathbf{W}$, $\mathbf{W}_1$, $\mathbf{C}$, and $\mathbf{E}$ are evaluated at convergence. Note that $\mathbf{F}$ is the expected Hessian of $l(\boldsymbol{\beta})$, pre-multiplied by the inverse of the Hessian of $l_p(\boldsymbol{\beta})$.

2.4 Stable and efficient evaluation of $\hat{\boldsymbol{\beta}}$ and τ

This section proposes an efficient and stable method for the evaluation of the working parameter estimates, $\hat{\boldsymbol{\beta}}$. In the case of the unconstrained model (Wood, 2006a) a stable solution for $\hat{\boldsymbol{\beta}}$ is based on a QR decomposition of $\sqrt{\mathbf{W}}\mathbf{X}$ augmented with $\sqrt{\lambda}\mathbf{B}$, where $\mathbf{B}^T\mathbf{B} = \mathbf{S}$. The same approach can be applied here for the monotone model, i.e. use a QR decomposition of the augmented $\sqrt{\mathbf{W}}\mathbf{X\Sigma C}$. However, the values of $\mathbf{W}$ can be negative when a non-canonical link function is assumed, so firstly, the issue with these negative weights has to be handled.

The approach applied here is similar to that given in Section 3.3 of Wood (2011). Let $|\mathbf{W}|$ denote a diagonal matrix with the elements $|w_i|$, and $\mathbf{W}^-$ be a diagonal matrix with

$$w_i^- = \begin{cases} 0, & \text{if } w_i \geq 0 \\ -w_i, & \text{otherwise.} \end{cases}$$

Then

$$(\mathbf{X\Sigma C})^T\mathbf{W}\mathbf{X\Sigma C} = (\mathbf{X\Sigma C})^T|\mathbf{W}|\mathbf{X\Sigma C} - 2(\mathbf{X\Sigma C})^T\mathbf{W}^-\mathbf{X\Sigma C}.$$

Now the QR decomposition may be used for the augmented matrix,

$$\begin{bmatrix} \sqrt{|\mathbf{W}|}\mathbf{X\Sigma C} \\ \sqrt{\lambda}\mathbf{B} \end{bmatrix} = \mathbf{QR}, \tag{2.20}$$

and $\sqrt{|\mathbf{W}|}\mathbf{X\Sigma C} = \mathbf{Q}_1\mathbf{R}$, where $\mathbf{Q}_1$ is the first n rows of $\mathbf{Q}$.

Therefore

$$\begin{aligned}(\mathbf{X\Sigma C})^T\mathbf{W}\mathbf{X\Sigma C} + \lambda\mathbf{S} - \mathbf{E} &= \mathbf{R}^T\mathbf{R} - 2(\mathbf{X\Sigma C})^T\mathbf{W}^-\mathbf{X\Sigma C} - \mathbf{E} \\ &= \mathbf{R}^T\left(\mathbf{I} - 2\mathbf{R}^{-T}(\mathbf{X\Sigma C})^T\mathbf{W}^-\mathbf{X\Sigma C}\mathbf{R}^{-1} - \mathbf{R}^{-T}\mathbf{E}\mathbf{R}^{-1}\right)\mathbf{R} \\ &= \mathbf{R}^T\left(\mathbf{I} - 2\mathbf{Q}_1^T\mathbf{I}^-\mathbf{Q}_1 - \mathbf{R}^{-T}\mathbf{E}\mathbf{R}^{-1}\right)\mathbf{R},\end{aligned}$$

where $\mathbf{I}^-$ is an $n \times n$ diagonal matrix with

$$I_i^- = \begin{cases} 0, & \text{if } w_i \geq 0 \\ 1, & \text{otherwise.} \end{cases}$$

The eigen-decomposition

$$2\mathbf{Q}_1^T \mathbf{I}^- \mathbf{Q}_1 + \mathbf{R}^{-T}\mathbf{E}\mathbf{R}^{-1} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

gives

$$(\mathbf{X\Sigma C})^T \mathbf{W} \mathbf{X\Sigma C} + \lambda \mathbf{S} - \mathbf{E} = \mathbf{R}^T \left(\mathbf{I} - \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \right) \mathbf{R} = \mathbf{R}^T \mathbf{U} \left(\mathbf{I} - \mathbf{\Lambda} \right) \mathbf{U}^T \mathbf{R}.$$

Defining

$$\tilde{z}_i = \begin{cases} (y_i - \mu_i) g'(\mu_i)/\alpha_i, & \text{if } w_i \geq 0 \\ -(y_i - \mu_i) g'(\mu_i)/\alpha_i, & \text{otherwise,} \end{cases}$$

then

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathbf{R}^{-1}\mathbf{U}(\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^T\mathbf{Q}_1^T \sqrt{|\mathbf{W}|}\tilde{\mathbf{z}} - \lambda \mathbf{R}^{-1}\mathbf{U}(\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^T\mathbf{R}^{-T}\mathbf{S}\boldsymbol{\beta}^{[k]}. \quad (2.21)$$

By denoting

$$\mathbf{P} = \mathbf{R}^{-1}\mathbf{U}(\mathbf{I} - \mathbf{\Lambda})^{-1/2} \text{ and } \mathbf{K} = \mathbf{Q}_1\mathbf{U}(\mathbf{I} - \mathbf{\Lambda})^{-1/2} \quad (2.22)$$

(2.21) may be written as

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathbf{P}\mathbf{K}^T \sqrt{|\mathbf{W}|}\tilde{\mathbf{z}} - \lambda \mathbf{P}\mathbf{P}^T\mathbf{S}\boldsymbol{\beta}^{[k]}. \quad (2.23)$$

The last expression has roughly the square root of the condition number of (2.18) for the unpenalized likelihood maximization problem, since the condition number of $\mathbf{R}^{-1}$ equals the condition number of $\sqrt{|\mathbf{W}|}\mathbf{X\Sigma C}$.

Note that in case of the canonical link function $\alpha_i = 1$ for any i, and therefore, $|\mathbf{W}| = \mathbf{W}$.

Now, given the value of the smoothing parameter, λ, the following Newton algorithm should be iterated to convergence, and at convergence $\hat{\boldsymbol{\beta}}$ maximizes the penalized log likelihood function:

1. Set initial values $\boldsymbol{\beta}^{[0]}$ of $\hat{\boldsymbol{\beta}}$ and set $k = 0$.
2. Evaluate $\boldsymbol{\mu}^{[k]}$, $\tilde{\mathbf{z}}^{[k]}$, $|\mathbf{W}|^{[k]}$, $\mathbf{P}^{[k]}$, and $\mathbf{K}^{[k]}$, at the current values of $\boldsymbol{\beta}^{[k]}$.
3. Calculate $\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathbf{P}^{[k]}\mathbf{K}^{[k]T} \sqrt{|\mathbf{W}|^{[k]}}\tilde{\mathbf{z}}^{[k]} - \lambda \mathbf{P}^{[k]}\mathbf{P}^{[k]T}\mathbf{S}\boldsymbol{\beta}^{[k]}$. Increment k.
4. Repeat steps 2 and 3 until convergence.

The penalized deviance may be used for the convergence test.

Using the approach and notations of this section, the effective degrees of freedom (2.19) can also be obtained in a stable manner. If $\mathbb{G} = (\mathbf{X\Sigma C})^T\mathbf{WX\Sigma C} - \mathbf{E} + \lambda\mathbf{S}$, and introducing $n \times n$ diagonal matrices $\mathbf{I}^+$ with

$$I_i^+ = \begin{cases} 1, & \text{if } w_i \geq 0 \\ -1, & \text{otherwise,} \end{cases}$$

and $\mathbf{L} = \text{diag}(\ldots, 1/\alpha_i, \ldots)$, then the expression for the effective degrees of freedom (2.19) becomes

$$\begin{aligned} \text{tr}(\mathbf{F}) &= \text{tr}\left(\sqrt{|\mathbf{W}|}\mathbf{X\Sigma C}\mathbb{G}^{-1}(\mathbf{X\Sigma C})^T\sqrt{|\mathbf{W}|}\mathbf{LI}^+\right) \\ &= \text{tr}(\mathbf{KK}^T\mathbf{LI}^+). \end{aligned} \tag{2.24}$$

The next section addresses several implementational issues for the above algorithm, such as initialization of the model parameters, column rank deficiency of the model matrix, and choice of the basis dimension.

2.5 Some optimization issues

2.5.1 Initialization

To start the iteration one needs to set $\boldsymbol{\beta}^{[0]}$ so that the initial fitted curve goes through the data. It is suggested to take $\boldsymbol{\mu}^{[0]} = \mathbf{y}$, $\boldsymbol{\eta}^{[0]} = g(\mathbf{y})$, as initial values, and given, the value of the smoothing parameter, to solve the following penalized least squares problem

$$\|\boldsymbol{\eta}^{[0]} - \mathbf{X\Sigma}\tilde{\boldsymbol{\beta}}\|^2 + \lambda\tilde{\boldsymbol{\beta}}^T\mathbf{S}\tilde{\boldsymbol{\beta}},$$

using the quadratic programming approach (see Section 2.7.1) with linear inequality constraints, $\tilde{\beta}_j > 0,\ \ j = 2, \ldots, q$. Then $\beta_1^{[0]} = \tilde{\beta}_1^{[0]}$, $\beta_j^{[0]} = \log(\tilde{\beta}_j^{[0]}),\ \ j = 2, \ldots, q$. This can be implemented using the `pcls()` function from the R package `mgcv`.

2.5.2 Stability

When dealing with a complex smoothing regression model, numerical instability may arise due to column rank deficiency of the Hessian of the log likelihood function (2.15) whose inverse should be calculated to obtain a Newton estimate of $\boldsymbol{\beta}$ (2.18). This may cause problems with parameter estimation. To detect the rank deficiency of the fitting problem one may use the QR decomposition (2.20) with pivoting. The rank deficiency problem can be dealt with by deleting the redundant columns of $\mathbf{Q}$ and corresponding

rows and columns of $\mathbf{R}$. Then

$$\mathbf{P} = \begin{bmatrix} \mathbf{R}^{-1}\mathbf{U}(\mathbf{I} - \boldsymbol{\Lambda})^{-1/2} \\ \mathbf{0} \end{bmatrix},$$

where zeroes stand for unidentifiable parameters.

Another way of dealing with the ill-conditioning is to form the singular value decomposition instead of the QR decomposition.

$$\begin{bmatrix} \sqrt{|\mathbf{W}|}\mathbf{X}\boldsymbol{\Sigma}\mathbf{C} \\ \sqrt{\lambda}\mathbf{B} \end{bmatrix} = \tilde{\mathbf{U}}\mathbf{D}\mathbf{V}^T, \tag{2.25}$$

where the diagonal matrix of the singular values, $\mathbf{D}$, reveals this deficiency (Golub and van Loan, 1996; Wood, 2006a). By setting $\mathbf{Q} = \tilde{\mathbf{U}}$ and $\mathbf{R} = \mathbf{D}\mathbf{V}^T$, the expression for $\mathbf{P}$ and $\mathbf{K}$ will be the same as in (2.22), but $\mathbf{R}^{-1} = \mathbf{V}\mathbf{D}^-$, where $\mathbf{D}^-$ is a diagonal matrix with $d_j^- = 1/d_j$ (inverse singular values), $j = 1, \ldots, q$, or $d_j^- = 0$ if d_j is 'too small'. Then the expression for the working parameter estimates may be written in the same way as in (2.23)

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathbf{P}\mathbf{K}^T\sqrt{|\mathbf{W}|}\tilde{\mathbf{z}} - \lambda\mathbf{P}\mathbf{P}^T\mathbf{S}\boldsymbol{\beta}^{[k]}.$$

For the Newton method there is a requirement that $\mathbb{G} = (\mathbf{X}\boldsymbol{\Sigma}\mathbf{C})^T\mathbf{W}\mathbf{X}\boldsymbol{\Sigma}\mathbf{C} - \mathbf{E} + \lambda\mathbf{S}$ is a positive semi-definite matrix, so that $\Lambda_j \leq 1$. The requirement might not be met for some steps of the iterative procedure. To avoid indefiniteness problem in the Newton iteration, a Fisher scoring should be substituted if $\Lambda_j > 1$ for any j. In this situation one should set $\alpha_i = 1$, $i = 1, \ldots, n$, so that $w_i \geq 0$ for any i, while $\mathbf{P} = \begin{bmatrix} \mathbf{R}^{-1} \\ \mathbf{0} \end{bmatrix}$ and $\mathbf{K} = \mathbf{Q}_1$. Then $\boldsymbol{\beta}^{[k+1]} = \mathbf{P}\mathbf{K}^T\sqrt{\mathbf{W}}\mathbf{z}$, where $\mathbf{z} = \mathbf{G}(\mathbf{y} - \boldsymbol{\mu}) + \mathbf{X}\boldsymbol{\Sigma}\mathbf{C}\boldsymbol{\beta}^{[k]}$. For the singular value decomposition

$$\boldsymbol{\beta}^{[k+1]} = \mathbf{V}\mathbf{D}^{-1}\tilde{\mathbf{U}}_1^T\sqrt{\mathbf{W}}\mathbf{z}.$$

One more consideration should be mentioned here. For a sufficiently non-linear monotonic smooth function, $f(x)$, divergence of the proposed iterative scheme may occasionally occur. Reduction of the parameter step length taken will stabilize fitting in this circumstance.

2.5.3 Basis dimension

Another question for any smoothing by penalized regression splines concerns the choice of the basis dimension, q. Using low rank penalized smoothers allows reduced computational effort in comparison with full rank smoothers. Moreover, the choice of the basis

dimension is not crucial, since the smoothing parameter controls the actual effective degrees of freedom. Therefore, it is required to set only an upper bound on the model flexibility by choosing q. The recommendation for the proposed mono-GAM would be as following: at first, to use an unconstrained GAM to decide on the number of basis functions, and then to chose q for mono-GAM at least as much as that for unconstrained case. If not running the unconstrained case first, based on general results from a series of simulation studies, it is suggested to use q at least 15–20. The monotonic terms require higher q because the monotonicity constraint "uses up" degrees of freedom.

2.6 Smoothing parameter selection

For estimating working model coefficients, $\boldsymbol{\beta}$, by penalized likelihood maximization the smoothing parameter, λ, should be given, so this section describes how the smoothing parameter can be estimated for a single smooth model with monotonicity constraint. When the scale parameter ϕ is known, λ can be estimated by minimizing the Un-Biased Risk Estimator (UBRE) (Craven and Wahba, 1979; Wood, 2006a), which is also Mallows' C_p (Mallows, 1973)

$$\mathcal{V}_u = \frac{1}{n} D(\hat{\boldsymbol{\beta}}) - \phi + \frac{2}{n}\phi\gamma\tau, \tag{2.26}$$

where $\gamma \geq 1$ is an ad hoc tuning parameter which can be used to force smoother models. $D(\hat{\boldsymbol{\beta}})$ is the model deviance

$$D(\hat{\boldsymbol{\beta}}) = 2(l_{max} - l(\hat{\boldsymbol{\beta}}))\phi, \tag{2.27}$$

with l_{max} denoting the maximum likelihood of the saturated model with one parameter for every observation.

If the scale parameter is unknown, the generalized cross validation (GCV) can be minimized with respect to the smoothing parameter λ (Craven and Wahba, 1979; Hastie and Tibshirani, 1990)

$$\mathcal{V}_g = \frac{nD(\hat{\boldsymbol{\beta}})}{(n - \gamma\tau)^2}. \tag{2.28}$$

Since there is no direct analytical method for minimizing (2.28) and (2.26), some numerical method should be developed to optimize it. For the model with a single smoothing parameter, which is the case considered in this chapter, the simplest method is to apply the direct grid search for the GCV/UBRE optimal smoothing parameter, which means that the model fitting algorithm must be iterated for each value of the smoothing parameter from the grid. Multiple smoothing parameter selection for an additive model

will be discussed in Section 4.3.

2.7 Other approaches to monotone smoothing

As previously stated, B-spline basis functions are very attractive in nonparametric smoothing due to their flexibility, local support, and an useful property of spline coefficients. So, several other B-spline monotone regression approaches have been developed. Almost all of them require an increasing sequence of spline coefficients to impose the monotonicity constraint. This section briefly describes the two main competitive approaches: constrained quadratic programming (Kelly and Rice, 1990; Wood, 1994) and P-spline regression with additional asymmetric penalties (Bollaerts et al., 2006b).

2.7.1 Quadratic programming

In Section 2.1 it was shown that a sufficient condition for a monotone increasing spline function is that the sequence of the B-spline coefficients should be of increasing size. So, to achieve monotonicity one may set up an increasing coefficient size constraint as linear inequality constraints in quadratic programming. Firstly, consider a Gaussian regression model with monotonicity constraint:

$$y_i = f(x_i) + \epsilon_i, \quad i = 1, \ldots, n,$$

where the ϵ_i are i.i.d. normally distributed random variables with parameters $(0, \sigma^2)$, $f(x)$ is an unknown smooth monotone increasing function. To estimate such a model we can use a B-spline basis for the smooth function with the degree of smoothness controlled by second-order difference penalties (Eilers and Marx, 1996), then the model can be written as

$$y_i = \mathbf{X}_i \boldsymbol{\beta} + \epsilon_i,$$

$\mathbf{X}_i$ is the i^{th} row of the model matrix consisting of B-spline basis functions (2.3) and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_q)^T$. Then a penalized likelihood for the model can be defined as

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\lambda \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta},$$

where $l(\boldsymbol{\beta})$ is a Gaussian log likelihood and the penalty matrix $\mathbf{S} = \mathbf{P}^T\mathbf{P}$,

$$\mathbf{P} = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & . & . \\ 0 & 1 & -2 & 1 & 0 & . & . \\ 0 & 0 & 1 & -2 & 1 & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \end{pmatrix}.$$

It is known that the penalized likelihood for the unconstrained model can be maximized by penalized least squares (e.g., Wood, 2006a); i.e. by minimization of

$$S_p = \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda\boldsymbol{\beta}^T\mathbf{S}\boldsymbol{\beta} \tag{2.29}$$

with respect to $\boldsymbol{\beta}$. The monotonicity condition of an increasing sequence of the model coefficients can be written as linear inequality constraints

$$\mathbf{A}\boldsymbol{\beta} > \mathbf{0}, \tag{2.30}$$

where

$$\mathbf{A} = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & . \\ 0 & -1 & 1 & 0 & 0 & . \\ 0 & 0 & -1 & 1 & 0 & . \\ . & . & . & . & . & . \end{pmatrix},$$

and $\mathbf{0}$ is a vector of zeros of dimension $q - 1$. Then, given the smoothing parameter λ, the model coefficients $\boldsymbol{\beta}$ can be estimated by minimizing (2.29) subject to (2.30) which is a quadratic programming problem. The solution for this problem can be obtained using the algorithm given in, for example, Gill et al. (1981) or Nocedal and Wright (2006). The `mgcv` package provides an R routine `pcls()`, which can be used to solve this problem.

For the generalized regression model (2.1)

$$g(\mu_i) = f(x_i),$$

the monotonicity constraint may be achieved by setting the quadratic programming problem within a P-IRLS loop. The following scheme is applied to fit the model:

1. Set initial values for $\boldsymbol{\mu}^{[0]}$.
2. Given the current $\boldsymbol{\mu}^{[k]}$, evaluate the weights $w_i^{[k]} = \omega_i / \left(V(\mu_i^{[k]})g'^2(\mu_i^{[k]})\right)$ and pseudodata $z_i^{[k]} = g'(\mu_i^{[k]})(y_i - \mu_i^{[k]}) + \mathbf{X}_i\boldsymbol{\beta}^{[k]}$.

(Notice that to start the iteration one does not need the initial values of $\boldsymbol{\beta}^{[0]}$, but rather $\mathbf{X}_i\boldsymbol{\beta}^{[0]}$ which equals $\boldsymbol{\eta}^{[0]} = g(\boldsymbol{\mu}^{[0]})$.)

3. Minimize the following quadratic programming problem w.r.t. $\boldsymbol{\beta}$ to find $\boldsymbol{\beta}^{[k+1]}$:

$$\min \quad \left\| \sqrt{\mathbf{W}^{[k]}} \left(\mathbf{z}^{[k]} - \mathbf{X}\boldsymbol{\beta} \right) \right\|^2 + \lambda \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta}, \tag{2.31}$$

$$\text{subject to} \quad \mathbf{A}\boldsymbol{\beta} > \mathbf{0}.$$

Calculate the linear predictor $\boldsymbol{\eta}^{[k+1]} = \mathbf{X}\boldsymbol{\beta}^{[k+1]}$ and fitted values $\mu_i^{[k+1]} = g^{-1}(\eta_i^{[k+1]})$.

4. Repeat steps 2 and 3 until convergence.

For the above procedure of $\boldsymbol{\beta}$ estimation the smoothing parameter λ should be given. Since it is plausible that the degree of smoothness for $f(x)$ will be similar for both unconstrained and monotonic fits, λ can be chosen via GCV or UBRE from the unconstrained model fit.

2.7.2 P-splines with additional asymmetric penalties

Bollaerts et al. (2006b) proposed another way of achieving an increasing order of the spline coefficients to meet the monotonicity constraint. The idea is to use the second-order differences of the coefficients as a 'wiggliness' penalty as before and to set additional asymmetric discrete penalties on the first-order differences of the coefficients in order to secure an increasing sequence. So, the penalized least squares for the single monotone smooth generalized regression model (2.1) may be represented as the following:

$$S_p = \left\| \sqrt{\mathbf{W}} \left(\mathbf{z} - \mathbf{X}\boldsymbol{\beta} \right) \right\|^2 + \lambda \sum_{j=3}^{q} \left(\triangle^2 \beta_j \right)^2 + k \sum_{j=2}^{q} v_j(\boldsymbol{\beta}) (\triangle^1 \beta_j)^2, \tag{2.32}$$

where the second term represents the 'wiggliness' penalty with the smoothing parameter λ, $\triangle^2 \beta_j = \beta_j - 2\beta_{j-1} + \beta_{j-2}$. The third item is a penalty reflecting the monotonicity constraint where

$$v_j(\boldsymbol{\beta}) = \begin{cases} 0, & \text{if} \quad \beta_j - \beta_{j-1} \geq 0 \\ 1, & \text{otherwise,} \end{cases}$$

$\triangle^1 \beta_j = \beta_j - \beta_{j-1}$, and k is a user-defined constant parameter which is suggested to be chosen sufficiently high, say 10^6, to ensure that the monotonicity assumption will be satisfied.

Hence, to estimate $\boldsymbol{\beta}$ we can use the P-IRLS scheme as in the quadratic programming approach, but instead of minimizing a quadratic programming problem (2.31), minimize (2.32) with respect to $\boldsymbol{\beta}$ at the current values of $\mathbf{z}^{[k]}$ and $\mathbf{w}^{[k]}$ to obtain the next $\boldsymbol{\beta}^{[k+1]}$. Replacing $\triangle^1\beta_j$ by the $n^{th}-$order differences allows to restrict the sign of the $n^{th}-$order derivative of $f(x)$, therefore, it is possible to set other shape constraints on the smooth function $f(x)$. It should be noted that Bollaerts et al. (2006b) do not suggest an efficient method of smoothing parameter selection.

Chapter 7 presents some simulation studies on performance of the proposed monotonic P-splines in comparison with the above mentioned approaches.

2.8 Illustrative simulations

To illustrate the performance of the proposed modelling approach with parameters estimation by the Newton based method and smoothness selection by direct minimization of the GCV/UBRE scores, some simulated data examples are presented here. In this section simulated examples on a single smooth monotone generalized regression model with a response variable Y_i that is Gaussian, Poisson, or Gamma are considered:

$$g(\mu_i) = f(x_i), \;\; i = 1, \ldots, n,$$

where $\mathrm{E}(Y_i) = \mu_i$. For the Gamma model the non-canonical log link function is taken and for the others the link function is canonical.

Example 1.1: In the first example consider a Gaussian model $y_i = f(x_i) + \epsilon_i$, where $f(x)$ is a monotonic function and ϵ_i are i.i.d. $N(\mu = 0, \;\; \sigma^2)$ random variables.

The following function was investigated

$$f(x) = \exp(4x) / \left\{1 + \exp(4x)\right\}.$$

One hundred values of the covariate, x_i, were simulated from a uniform distribution on $[-1, 3]$, and the function $f(x)$ was applied to these covariate values to obtain the true response mean. The response data, y_i, were simulated from the normal distribution with that mean at each of three noise levels, $\sigma = 0.05, \;\; 0.1$, and 0.2.

Each of three data sets was modeled using $q = 15$ basis functions of the proposed monotonic P-spline, and a cubic spline ($m = 2$) was used in each case. The smoothing parameter was selected by direct minimization of the GCV score (2.28) with $\gamma = 1$. The GCV optimal models are shown in Figure 2-5. The effective degrees of freedom, τ, of the monotone and unconstrained fits and the corresponding minimal GCV scores are given in Table 2.1.

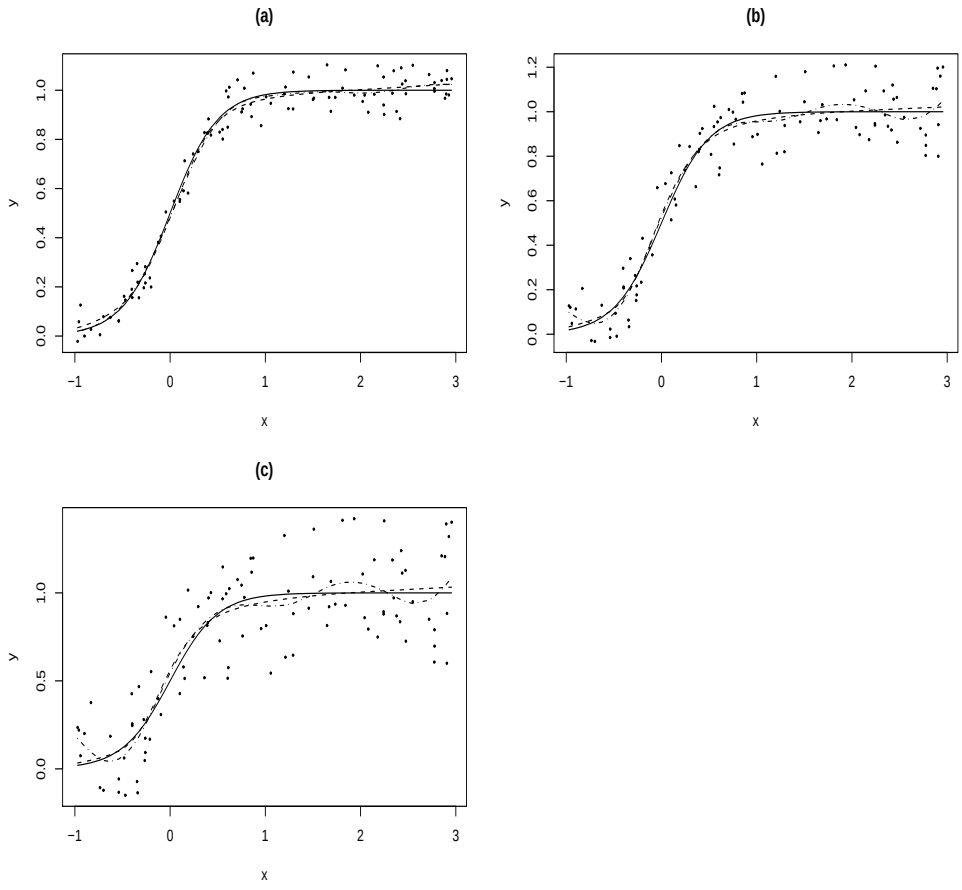


Figure 2-5: The best fits of one term Gaussian models with monotonicity constraints (dashed line), the true function (solid line), the unconstrained fit (dot dashed line), and simulated data points. (a) $\sigma = 0.05$, (b) $\sigma = 0.1$, (c) $\sigma = 0.2$.

Table 2.1: The effective degrees of freedom and minimal GCV scores for the one-dimensional Gaussian model

	$\sigma = 0.05$		$\sigma = 0.1$		$\sigma = 0.2$	
	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$
Monotone model	4.68	$3.596 \cdot 10^{-3}$	4.18	0.01311	3.70	0.05223
Unconstrained model	8.19	$3.801 \cdot 10^{-3}$	8.54	0.01330	8.13	0.05306

The figure shows the best fits to the constrained model (solid line), the true monotonic function (dashed line), the unconstrained fit (long dash line), and the simulated observations. It should be noticed that unconstrained GAM does not reproduce the monotone curve on the plateau regions of the function for all three noise levels, emphasizing the advantage of the monotone smooth.

From the table one may see that the effective degree of freedom of the monotone fit is less than that of the unconstrained fit for all three values of σ. This is in accordance with the visual appearance of the smoothness.

Example 1.2: Consider a Poisson model with log link function,

$$\log(\mu_i) = f(x_i), \;\; i = 1, \ldots, n,$$

where $\mu_i = \mathrm{E}(Y_i)$, $Y_i \sim \mathrm{Pois}\left[\exp\left\{f(x_i)\right\}\right]$.

The test function was as in the previous Gaussian model but with an additional constant d to control the strength of the signal

$$f(x_i) = d \times \exp(4x_i)/\left\{1 + \exp(4x_i)\right\}.$$

The x_i were drawn from a Unif$(-1, 3)$ distribution, $n = 100$. The values of d were taken as 2, 3, and 4.

The data set was modeled using the cubic P-spline of the dimension $q = 15$, fitted by penalized likelihood maximization with λ chosen by UBRE. Figure 2-6 illustrates the optimal fitted curve of the estimated mean values, $\hat{\mu}_i = \exp\{\hat{f}(x_i)\}$, as a dashed line, the true curve of mean values, $\mu_i = \exp\left\{f(x_i)\right\}$, as a solid line, the unconstrained curve as a dot dashed line, and the simulated points for three values of d. For this example the true curve of mean values μ_i should also be monotone increasing, but the unconstrained model does not reflect this fact.

Table 2.2 shows the effective degrees of freedom of the fits and the minimal UBRE scores.

Example 1.3: In the third example a Gamma model with a non-canonical log link

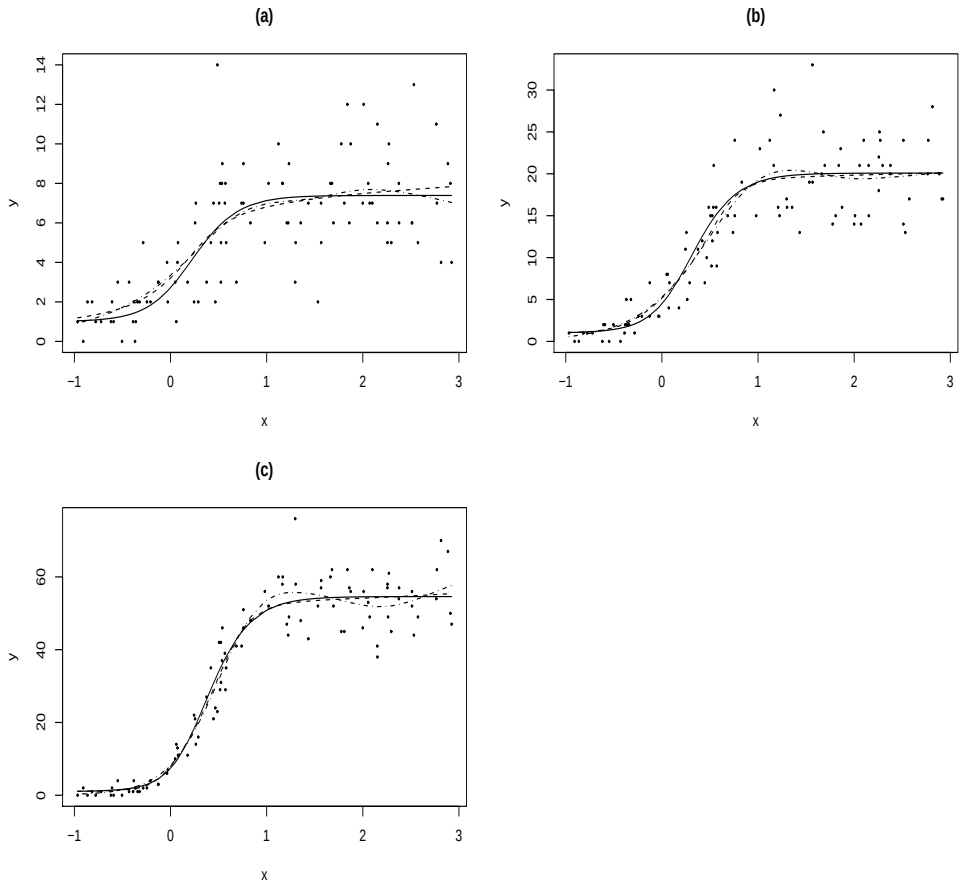


Figure 2-6: The best fits of one term Poisson models with monotonicity constraints (dashed line), the true function (solid line), the unconstrained fit (dot dashed line), and simulated data points. (a) $d = 2$, (b) $d = 3$, (c) $d = 4$.

Table 2.2: The effective degrees of freedom and minimal UBRE scores for the one-dimensional Poisson regression

	$d=2$		$d=3$		$d=4$	
	τ	$\mathcal{V}_u$	τ	$\mathcal{V}_u$	τ	$\mathcal{V}_u$
Monotone model	3.30	-0.0843	4.78	0.1039	4.87	0.1351
Unconstrained model	4.80	-0.0612	12.99	0.1467	6.55	0.1586

function is fitted

$$\log(\mu_i) = f(x_i), \;\; i = 1, \ldots, n,$$

where $Y_i \sim$ Gamma $[\nu = 1, \;\; \theta = \exp\{f(x_i)\}]^2$, with a shape parameter ν and a scale parameter θ.

$$f(x_i) = d \times \exp\left\{5((x_i - 1) + 0.05)\right\} / \left[1 + \exp\left\{5((x_i - 1) + 0.01)\right\}\right],$$

Let $x_i \sim \text{Unif}(0, 4)$, $n = 200$. Three values of the signal strength were taken, $d = 1.5$, 2, and 3.5. For both the monotone P-spline and the common unconstrained P-splines, $q = 20$ basis functions were used with $m = 2$. The optimal fitted curves by the GCV score minimization are illustrated in Figure 2-7 with τ and the GCV score presented in Table 2.3.

Table 2.3: The effective degrees of freedom and minimal GCV scores for the one-dimensional Gamma regression

	$d=1.5$		$d=2$		$d=3.5$	
	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$
Monotone model	3.99	1.11622	4.22	1.11799	4.66	1.12198
Unconstrained model	5.76	1.11794	6.49	1.12083	7.57	1.12694

As in the previous examples the unconstrained GAM exhibits non-monotonic fluctuations, especially on the plateau region of the function. The GCV/UBRE scores of the monotone models of the examples are less than the GCV/UBRE scores of the unconstrained fit for all three levels of the signal strength. This is probably due to the very small or even negligible increment in the parameters of the monotone fit on the plateau regions of the function which leads to tiny contribution to the effective degrees of freedom (edf) from those parameters. Hence, the smaller value of the overall

[2]The probability density function of a gamma distributed random variable Y is $f(y; \nu, \theta) = y^{\nu-1} \frac{\exp(-y/\theta)}{\theta^\nu \Gamma(\nu)}$ for y, ν, $\theta > 0$. $\text{E}(Y) = \nu\theta$, $\text{var}(Y) = \nu\theta^2$.

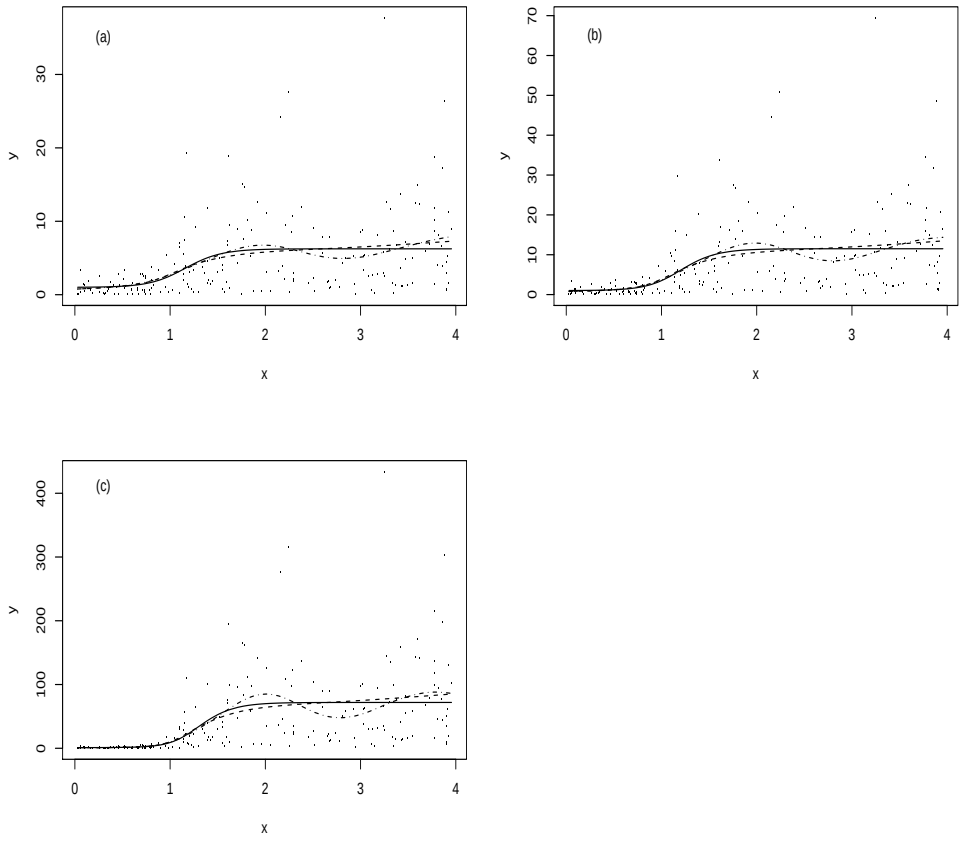


Figure 2-7: The best fits of one term Gamma models with monotonicity constraints (dashed line), the true function (solid line), the unconstrained fit (dot dashed line), and simulated data points. (a) $d = 1.5$, (b) $d = 2$, (c) $d = 3.5$.

degrees of freedom of the fitted monotone model in comparison with the edf of the unconstrained model lowers the GCV/UBRE scores of the monotone fit.

The above simulated examples are given only for illustration of the mono-GAM performance. A more extensive simulation study will be introduced in Chapter 7.

Chapter 3

Extensions to other shape preserving smoothing and bivariate monotonicity

In the previous chapter only models with a monotone increasing smooth term have been considered. This chapter will introduce univariate smoothing under other shape constraints such as, a monotone decreasing constraint and monotonicity together with convexity/concavity. Based on tensor product smooths, smoothing of bivariate functions with monotonicity restrictions on both covariates (double monotonicity) and on only one of them (single monotonicity) will also be developed. Penalties for these shape - preserving smoothers will be obtained. To show the performance of the proposed smoothers, several simulation examples will be presented in the last section of this chapter.

3.1 Monotone decreasing smoothing

Consider the same one-smooth model as in Section 2.1

$$g(\mu_i) = f(x_i), \;\; i = 1, \ldots, n, \tag{3.1}$$

but now the smooth function $f(x_i)$ is assumed to be monotone decreasing, that is

$$f(x_i) < f(x_j) \;\text{ if }\; x_i > x_j.$$

Using the arguments given in Section 2.1, we may conclude that a sufficient condi-

tion for $f'(x_i) < 0$ is

$$\triangle^1 \gamma_j = \gamma_j - \gamma_{j-1} < 0,$$

where the γ_j are unknown parameters of the B-spline, $j = 1, \ldots, q$.

Therefore, the sequence of all model parameters γ_j, should be decreasing. To satisfy this condition the model coefficients for the monotone decreasing constraint are defined in terms of working coefficients, $\boldsymbol{\beta}$, as follows,

$$\gamma_1 = \beta_1, \;\; \gamma_j = \beta_1 - \sum_{i=2}^{j} \exp(\beta_i), \;\; j = 2, \ldots, q.$$

If the matrix $\boldsymbol{\Sigma}$ is

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & -1 & 0 & \ldots & 0 \\ 1 & -1 & -1 & \ldots & 0 \\ . & . & . & . & . \\ 1 & -1 & -1 & \ldots & -1 \end{pmatrix} \tag{3.2}$$

for the decreasing case, then the model (3.1) will be written as for the monotone increasing smooth term model

$$g(\mu_i) = \mathbf{X}_i \boldsymbol{\Sigma} \tilde{\boldsymbol{\beta}},$$

where

$$\boldsymbol{\beta} = (\beta_1, \exp(\beta_2), \exp(\beta_3), ..., \exp(\beta_q))^T.$$

Figure 3.1 illustrates the monotone decreasing smooths obtained by using eight B-splines basis functions of the third and fourth orders.

To control function 'wiggliness' the same penalty as for the monotone increasing smooth can be applied here.

3.2 P-splines with mixed constraints

In some research areas a monotonicity constraint may be assumed, together with convexity or concavity. For instance, in forest research it is assumed that a tree's height depends on an aridity index which is calculated as a fraction of the precipitation sum per year over mean temperature per year plus 10. This dependence is expected to be monotone increasing and concave. Meyer (2008) considered smoothing of yield as a function of planting density of onions where the relationship was supposed to be decreasing and convex. For references and different approaches to shape constrained smoothing, see, for example, Meyer (2008), Ng and Maechler (2007), Turlach (2005),

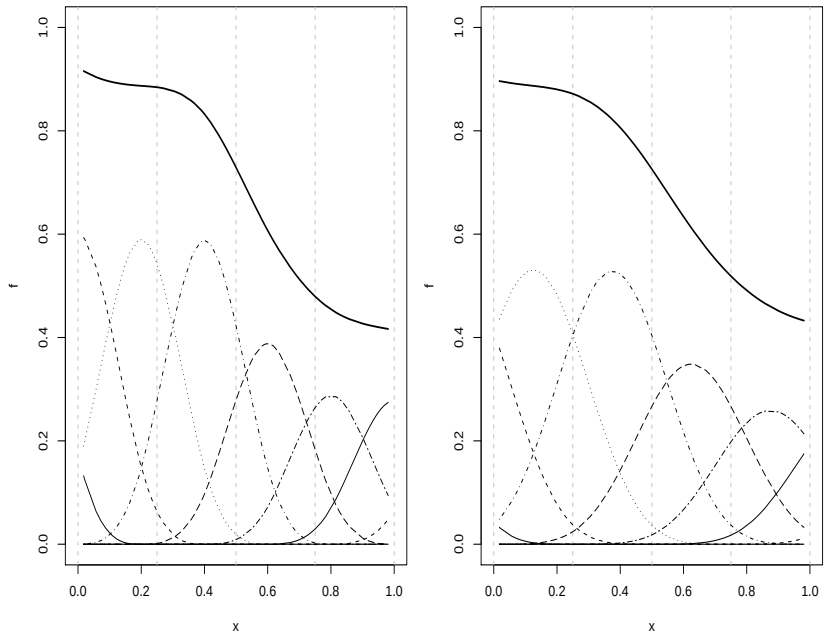


Figure 3-1: Illustration of monotone decreasing smooth curves using third (left panel) and fourth (right panel) order B-spline bases.

Elfving and Andersson (1988). In this section mixed constrained P-splines are presented which in fact differ from the monotone P-splines only in the representation of the matrix $\mathbf{\Sigma}$ and a small change in the penalty matrix.

Consider again the single smooth model

$$g(\mu_i) = f(x_i), \;\; i = 1, \ldots, n, \tag{3.3}$$

but now assume that the smooth function $f(x_i)$ is monotone increasing (or decreasing) and convex (or concave). Using P-splines it is possible to re-parameterize the model coefficients in such a way that sufficient conditions for monotonicity and convexity are satisfied.

From De Boor (1978) the second order derivative of the B-spline with equally spaced knots is

$$f''(x_i) = \frac{1}{h^2} \sum_{j=3}^{q} B_j^{m-2}(x_i) \triangle^2 \gamma_j,$$

where $\triangle^2 \gamma_j$ is the second order difference of the model parameters. Then a sufficient

condition for $f''(x_i) > 0$ (or < 0) is

$$\triangle^2 \gamma_j = \gamma_j - 2\gamma_{j-1} + \gamma_{j-2} > 0 \ (< 0).$$

Therefore, to construct P-splines with mixed constraints the following two conditions should be satisfied simultaneously:

$$\triangle^1 \gamma_j > 0 \ (< 0) \ \text{ and } \ \triangle^2 \gamma_j > 0 \ (< 0), \ \ j = 1, \ldots, q. \tag{3.4}$$

To achieve (3.4) the model parameters $\boldsymbol{\gamma}$ are parameterized such that $\boldsymbol{\gamma} = \boldsymbol{\Sigma}\tilde{\boldsymbol{\beta}}$, where $\tilde{\boldsymbol{\beta}} = (\beta_1, \exp(\beta_2), \exp(\beta_3), ..., \exp(\beta_q))^T$ and $\boldsymbol{\Sigma}$ is a $q \times q$ matrix with the following elements for four different types of the mixed constraints:

1. For a monotone increasing and convex smooth:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & 0 & \ldots & 0 \\ 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 2 & 1 & 0 & \ldots & 0 \\ 1 & 3 & 2 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & q-1 & q-2 & q-3 & \ldots & 1 \end{pmatrix}$$

2. For a monotone increasing and concave smooth:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & \ldots & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 & \ldots & 2 & 2 & 1 \\ 1 & 3 & 3 & 3 & \ldots & 3 & 2 & 1 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & q-1 & q-2 & q-3 & \ldots & 3 & 2 & 1 \end{pmatrix}$$

3. For a monotone decreasing and convex smooth:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 1 & -1 & -1 & -1 & \ldots & -1 & -1 & -1 \\ 1 & -2 & -2 & -2 & \ldots & -2 & -2 & -1 \\ 1 & -3 & -3 & -3 & \ldots & -3 & -2 & -1 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & -(q-1) & -(q-2) & -(q-3) & \ldots & -3 & -2 & -1 \end{pmatrix}$$

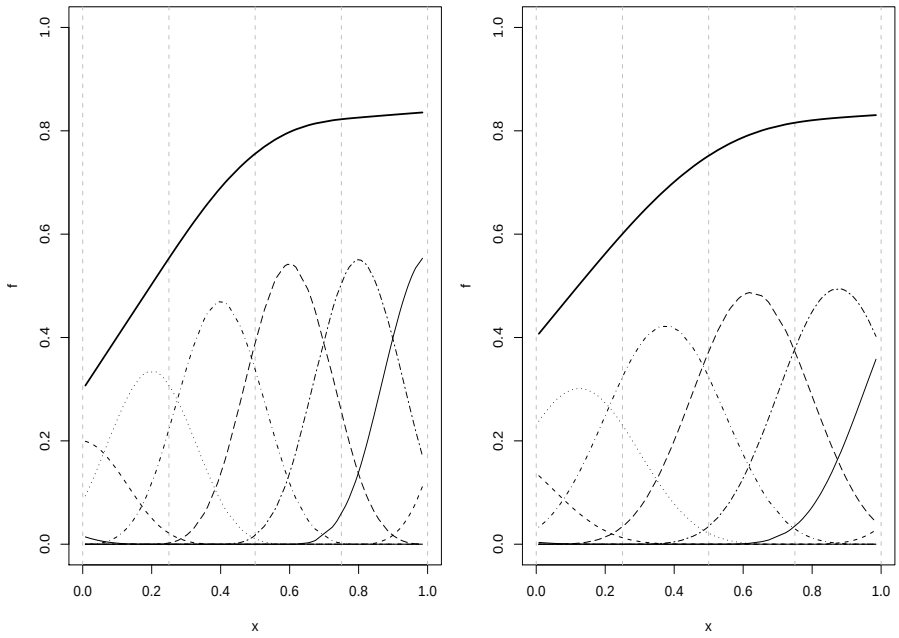


Figure 3-2: Illustration of monotone increasing and concave smooth curves using third (left panel) and fourth (right panel) order B-spline bases.

4. For a monotone decreasing and concave smooth:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 1 & -1 & 0 & 0 & \dots & 0 \\ 1 & -2 & -1 & 0 & \dots & 0 \\ 1 & -3 & -2 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & -(q-1) & -(q-2) & -(q-3) & \dots & -1 \end{pmatrix}$$

As in the monotone case, the model (3.3) can be now written as

$$g(\mu_i) = \mathbf{X}_i \boldsymbol{\Sigma} \tilde{\boldsymbol{\beta}}. \tag{3.5}$$

Figures 3-2 and 3-3 show smooths with mixed constraints which were constructed using B-splines of orders 3 and 4 with $q = 8$ for both cases.

To control the degree of model smoothness, penalties based on the first-order differences of the adjacent model coefficients are used. But for the P-splines with mixed constraints such penalties should be started from the third working coefficient, β_3,

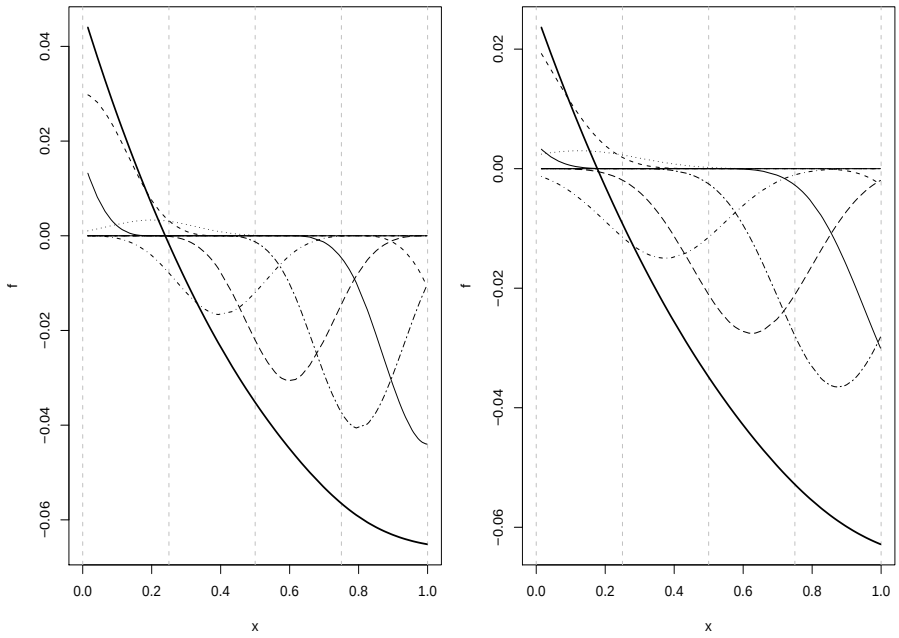


Figure 3-3: Illustration of monotone decreasing and convex smooth curves using third (left panel) and fourth (right panel) order B-spline bases.

since the second working coefficient is responsible for the slope of the fitted curve. By allowing β_1 and β_2 to vary while keeping other parameters close to each other, such a penalization will lead to a quadratic function when $\lambda \to \infty$, which is proved in the next subsection. Therefore, the following penalty is used for the mixed constrained model:

$$P = \sum_{j=3}^{q-1} (\beta_{j+1} - \beta_j)^2 = \boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta}, \tag{3.6}$$

where

$$\mathbf{S} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & . & . \\ 0 & 0 & 0 & 0 & 0 & . & . \\ 0 & 0 & 1 & -1 & 0 & . & . \\ 0 & 0 & -1 & 2 & -1 & . & . \\ 0 & 0 & 0 & -1 & 2 & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \end{pmatrix}.$$

Since the difference between the mixed constraint model and the monotone model of Section 2.1 is only in the representation of the matrix $\boldsymbol{\Sigma}$, (3.5) can be estimated by

the same method used for the monotone P-splines.

Form of mixed constraint P-splines when $\lambda \to \infty$

This section proves that the penalty (3.6), used for smooths with mixed constraints, produces a quadratic function when the smoothing parameter λ tends to infinity. Consider first the case of mixed constraints, i.e. monotone increasing plus convexity. The mixed constrained P-spline of this type can be written as

$$f(x_i) = \mathbf{X}_i \boldsymbol{\gamma},$$

where $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_q)^T$,

$$\begin{aligned}
&\gamma_1 = \beta_1 \\
&\gamma_2 = \beta_1 + \exp(\beta_2) \\
&\gamma_3 = \beta_1 + 2\exp(\beta_2) + \exp(\beta_3) \\
&\gamma_4 = \beta_1 + 3\exp(\beta_2) + 2\exp(\beta_3) + \exp(\beta_4) \\
&\ldots \\
&\gamma_q = \beta_1 + (q-1)\exp(\beta_2) + (q-2)\exp(\beta_3) + \ldots + \exp(\beta_q).
\end{aligned}$$

When $\lambda \to \infty$ the penalty (3.6) keeps the model parameters close to each other starting with the third one, β_3. In particular it restricts the values as follows, $\beta_k \to \beta_3, \ \forall k \geq 4$, so that, in the limit

$$\begin{aligned}
&\gamma_1 = \beta_1 \\
&\gamma_2 = \beta_1 + \exp(\beta_2) \\
&\gamma_3 = \beta_1 + 2\exp(\beta_2) + \exp(\beta_3) \\
&\gamma_4 = \beta_1 + 3\exp(\beta_2) + 3\exp(\beta_3) \\
&\gamma_5 = \beta_1 + 4\exp(\beta_2) + 6\exp(\beta_3) \\
&\gamma_6 = \beta_1 + 5\exp(\beta_2) + 10\exp(\beta_3) \\
&\ldots.
\end{aligned}$$

In this case, as was shown in Section 2.1, the first order derivative of the B-spline with equally spaced knots is

$$f'(x_i) = \frac{1}{h} \sum_{j=2}^{q} B_j^{m-1}(x_i) \triangle^1 \gamma_j,$$

where the first order differences of the model parameters, $\triangle^1\gamma_j$, have the values below

$$\triangle^1\gamma_2 = \exp(\beta_2),\ \ \triangle^1\gamma_3 = \exp(\beta_2) + \exp(\beta_3),\ \ \triangle^1\gamma_4 = \exp(\beta_2) + 2\exp(\beta_3), \ldots,$$
$$\triangle^1\gamma_q = \exp(\beta_2) + (q-2)\exp(\beta_3).$$

Similarly the second order derivative of $f(x_i)$ is

$$f''(x_i) = \frac{1}{h^2}\sum_{j=3}^{q} B_j^{m-2}(x_i)\triangle^2\gamma_j,$$

with $\triangle^2\gamma_3 = \triangle^2\gamma_4 = \ldots = \triangle^2\gamma_q = \exp(\beta_3)$. By induction the next derivative will be equal to zero. Therefore, the second order derivative, $f''(x_i)$, is equal to a constant, from which it follows that $f'(x_i)$ is a linear function, and hence the mixed constrained P-splines represents a quadratic function when the smoothing parameter goes to infinity.

The same approach can be used for the smooths with the other mixed constraints. Moreover, it is also not difficult to prove that the penalty applied for the monotone P-splines results in a straight line when $\lambda \to \infty$, although this can also be seen by inspection.

Figures 3-4 - 3-7 provide illustrations of how the curves change with λ for each type of the mixed constraints. For all four cases, twenty five B-spline basis functions of the third order were used.

3.3 Double monotonicity for smooths of two covariates

In the previous sections shape constrained penalized regression smoothers based on univariate P-splines have been introduced for smooth functions of a single covariate. Using the concept of tensor product spline bases (De Boor, 1978; Wood, 2006a) it is not difficult to build up smooths of two covariates under monotonicity constraint, where monotonicity may be assumed on only one of the covariates (single monotonicity) or on both of them (double monotonicity).

In the first section tensor products of two monotonic P-splines will be developed in order to achieve double monotonicity along both directions. Single monotonicity, that is monotonicity only along one direction, will be introduced in the second subsection.

3.3.1 Tensor product with monotonic P-splines

Consider the single smooth term model

$$g(\mu_i) = f(x_{1i}, x_{2i}),\ \ i = 1, \ldots, n, \tag{3.7}$$

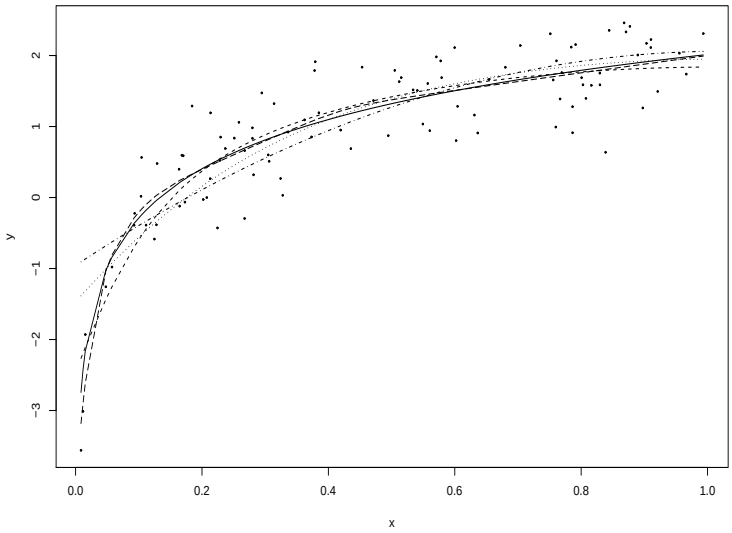


Figure 3-4: Illustration of the monotone increasing and concave smooth curves for four values of the smoothing parameter: $\lambda_1 = 1e-7$ (long dashed curve), $\lambda_2 = 1e-4$ (short dashed curve), $\lambda_3 = 5e-4$ (dotted curve), and $\lambda_4 = 10$ (dot-dashed curve). The true curve is represented as a solid line and dots are the simulated data.

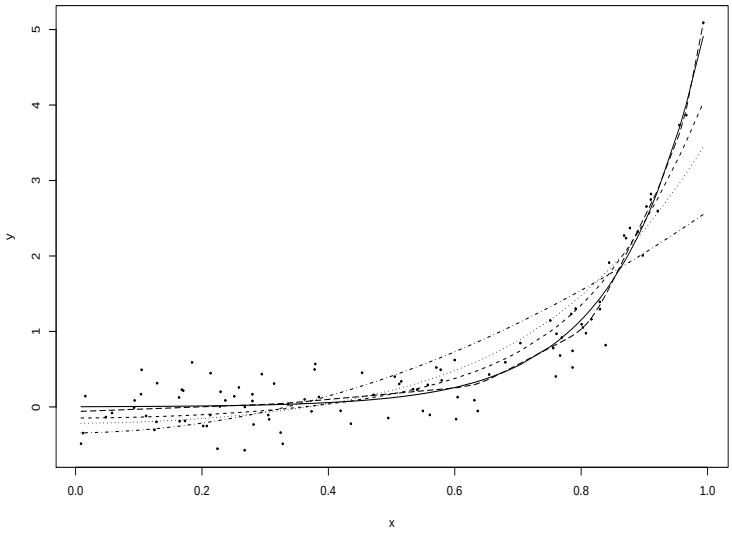


Figure 3-5: Illustration of the monotone increasing and convex smooth curves for four values of the smoothing parameter: $\lambda_1 = 1e-9$ (long dashed curve), $\lambda_2 = 5e-4$ (short dashed curve), $\lambda_3 = 1e-3$ (dotted curve), and $\lambda_4 = 10$ (dot-dashed curve). The true curve is represented as a solid line and dots are the simulated data.

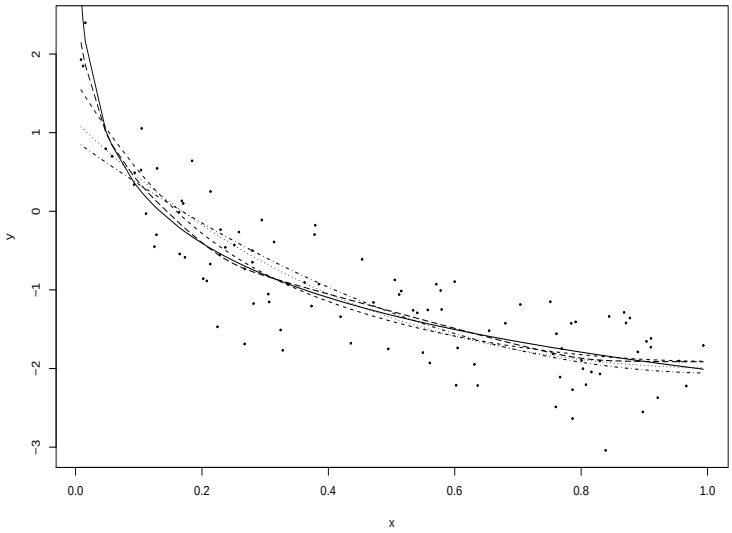


Figure 3-6: Illustration of the monotone decreasing and convex smooth curves for four values of the smoothing parameter: $\lambda_1 = 1e-7$ (long dashed curve), $\lambda_2 = 1e-4$ (short dashed curve), $\lambda_3 = 5e-4$ (dotted curve), and $\lambda_4 = 10$ (dot-dashed curve). The true curve is represented as a solid line and dots are the simulated data.

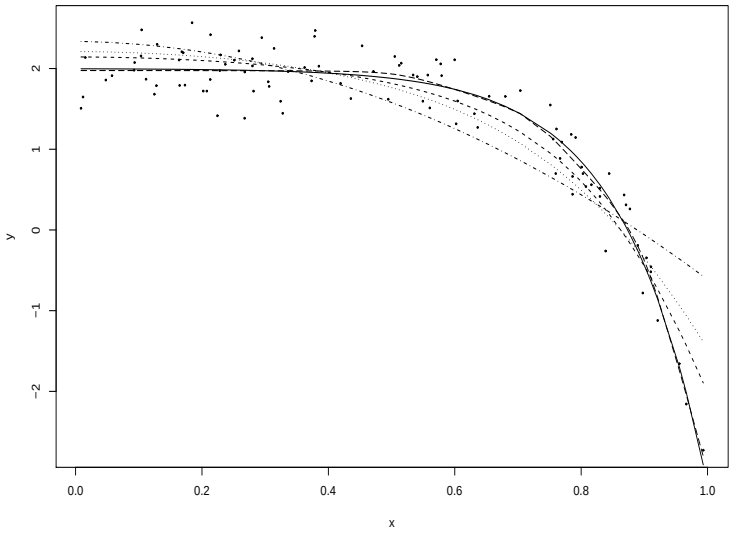


Figure 3-7: Illustration of the monotone decreasing and concave smooth curves for four values of the smoothing parameter: $\lambda_1 = 1e-7$ (long dashed curve), $\lambda_2 = 5e-4$ (short dashed curve), $\lambda_3 = 1e-3$ (dotted curve), and $\lambda_4 = 10$ (dot-dashed curve). The true curve is represented as a solid line and dots are the simulated data.

where the unknown function f now depends on two covariates x_1 and x_2, and is subject to double monotonicity constraints.

Consider two $(m+1)^{th}$ order B-splines with basis dimensions q_1 and q_2, as described in Section 2.1 for representing two smooth functions, each of single covariates

$$\mathtt{f}_1(x_{1i}) = \sum_{j=1}^{q_1} B_j^m(x_{1i})\alpha_j, \;\; \mathtt{f}_2(x_{2i}) = \sum_{k=1}^{q_2} B_k^m(x_{2i})\gamma_k,$$

where $B_j^m(x_1)$ and $B_k^m(x_k)$ are B-spline basis functions, and α_j and γ_k are parameters. Then, to represent the smooth function of two covariates, parameters α_j can be expressed as the B-spline of the second covariate (Wood, 2006a), hence

$$f(x_{1i}, x_{2i}) = \sum_{j=1}^{q_1}\sum_{k=1}^{q_2} B_{jk}^m(x_{1i}, x_{2i})\beta_{jk},$$

with $B_{jk}^m(x_{1i}, x_{2i}) = B_j^m(x_{1i}) \cdot B_k^m(x_{2i})$.

Using the matrix-vector notations the univariate smooth functions can be written as $\mathtt{f}_1(x_{1i}) = \mathbf{X}_{1i}\boldsymbol{\alpha}$ and $\mathtt{f}_2(x_{2i}) = \mathbf{X}_{2i}\boldsymbol{\gamma}$, where $\mathbf{X}_{1i}$ and $\mathbf{X}_{2i}$ are the i^{th} rows of model matrices, consisting of evaluated B-spline basis functions, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{q_1})^T$, and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_{q_2})^T$. Therefore, by denoting the model matrix of the smooth of two covariates as $\mathbf{X}$ we get

$$\mathbf{X}_i = \mathbf{X}_{1i} \otimes \mathbf{X}_{2i},$$

where $\otimes$ denotes a Kronecker product, that is the i^{th} row of the bivariate model matrix is the Kronecker product of two rows of univariate model matrices, so

$$f(x_{1i}, x_{2i}) = \mathbf{X}_i\boldsymbol{\beta},$$

and the vector of model parameters may be expressed in the following order

$$\boldsymbol{\beta} = (\beta_{11}, \beta_{12}, \ldots, \beta_{1q_2}, \beta_{21}, \ldots, \beta_{2q_2}, \ldots, \beta_{q_1q_2})^T.$$

For equally spaced knot locations for both covariates the first order derivative of the bivariate B-spline with respect to the first covariate is

$$\frac{\partial f(x_{1i}, x_{2i})}{\partial x_{1i}} = \frac{1}{h_1}\sum_{j=2}^{q_1}\sum_{k=1}^{q_2} B_k^m(x_{2i})B_j^{m-1}(x_{1i})\triangle_1^1\beta_{jk},$$

where h_1 is the distance between two adjacent knots of the first covariate, $\triangle_1^1\beta_{jk} = \beta_{jk} - \beta_{(j-1),k}$ is the first order difference of the model parameters with respect to the

first index only. Similarly, the first order derivative with respect to x_{2i} will be

$$\frac{\partial f(x_{1i}, x_{2i})}{\partial x_{2i}} = \frac{1}{h_2} \sum_{j=1}^{q_1} \sum_{k=2}^{q_2} B_j^m(x_{1i}) B_k^{m-1}(x_{2i}) \triangle_2^1 \beta_{jk},$$

where h_2 is the distance between two adjacent knots of the second covariate, $\triangle_2^1 \beta_{jk} = \beta_{jk} - \beta_{j,(k-1)}$ is the first order difference of the model parameters with respect to the second index.

Therefore, sufficient condition for $\frac{\partial f(x_{1i}, x_{2i})}{\partial x_{1i}} > 0$ is

$$\triangle_1^1 \beta_{jk} > 0,$$

and sufficient condition for $\frac{\partial f(x_{1i}, x_{2i})}{\partial x_{2i}} > 0$ is

$$\triangle_2^1 \beta_{jk} > 0.$$

In order to achieve these conditions for double monotonicity the following re-parametrizations of the model parameters are proposed:

1. For the double monotone increasing bivariate function (monotone increasing with respect to both covariates):

   Let

   $$\mathbf{\Sigma}_1 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix} \tag{3.8}$$

   be $q_1 \times q_1$ matrix, and $\mathbf{\Sigma}_2$ is as (3.8) but of the dimension $q_2 \times q_2$. Then for the bivariate B-spline with monotonicity constraint

   $$\mathbf{\Sigma} = \mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2, \tag{3.9}$$

   and model (3.7) can be written as

   $$g(\mu_i) = \mathbf{X}_i \mathbf{\Sigma} \tilde{\beta},$$

   where

   $$\tilde{\beta} = (\beta_{11}, \exp(\beta_{12}), \exp(\beta_{13}), \dots, \exp(\beta_{1q_2}), \exp(\beta_{21}), \dots, \exp(\beta_{2q_2}), \dots, \exp(\beta_{q_1 q_2}))^T \tag{3.10}$$

2. For the double monotone decreasing bivariate function:

   Let $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ be as above and let $\boldsymbol{\Sigma}' = -\boldsymbol{\Sigma}_1 \otimes \boldsymbol{\Sigma}_2$. Then

$$\boldsymbol{\Sigma} = \boldsymbol{\Sigma}'\mathtt{L},$$

   where $\mathtt{L}$ is a diagonal matrix with

$$\mathtt{L}_{jj} = \begin{cases} -1, & \text{if } j = 1 \\ 1, & \text{otherwise,} \end{cases}$$

   that is $\boldsymbol{\Sigma}$ is a matrix $\boldsymbol{\Sigma}'$ with the first column replaced by the column of one's.

   All the rest remains the same as in the double monotone increasing case.

3.3.2 Single monotonicity along only one direction

Consider at first the single monotonicity of the bivariate function along the x_1 direction. As in the previous case the univariate marginal smooth functions are constructed using the B-spline basis functions. Hence, the bivariate smooth function is represented as

$$f(x_{1i}, x_{2i}) = \sum_{j=1}^{q_1} \sum_{k=1}^{q_2} B_j^m(x_{1i}) B_k^m(x_{2i}) \beta_{jk} = \mathbf{X}_i \boldsymbol{\beta}.$$

The i^{th} row of the bivariate model matrix is $\mathbf{X}_i = \mathbf{X}_{1i} \otimes \mathbf{X}_{2i}$, where

$$\mathbf{X}_{1i} = \left\{ B_1^m(x_{1i}), B_2^m(x_{1i}), \ldots, B_{q_1}^m(x_{1i}) \right\},$$

$$\mathbf{X}_{2i} = \left\{ B_1^m(x_{2i}), B_2^m(x_{2i}), \ldots, B_{q_2}^m(x_{2i}) \right\}.$$

To satisfy single monotonicity along only x_1 the first order derivative with respect to x_1 should be considered

$$\frac{\partial f(x_{1i}, x_{2i})}{\partial x_{1i}} = \frac{1}{h_1} \sum_{j=2}^{q_1} \sum_{k=1}^{q_2} B_k^m(x_{2i}) B_j^{m-1}(x_{1i}) \triangle_1^1 \beta_{jk},$$

and obviously, the condition for a single monotone increasing bivariate function is therefore,

$$\triangle_1^1 \beta_{jk} > 0.$$

Similarly, it is easy to see that the sufficient condition for the single monotonicity along the second covariate x_2 is $\triangle_2^1 \beta_{jk} > 0$.

To satisfy the conditions for monotone increase or decrease ($\triangle_l^1 \beta_{jk} < 0, l = 1, 2$) in each direction, the following four re-parameterizations are developed:

1. For the single monotone increasing bivariate function along the x_1 direction:

   Let $\mathbf{\Sigma}_1$ be the same as in (3.8) and $\mathbf{I}_2$ be an identity matrix of size q_2, then

   $$\mathbf{\Sigma} = \mathbf{\Sigma}_1 \otimes \mathbf{I}_2, \tag{3.11}$$

   and

   $$\tilde{\beta} = (\beta_{11}, \beta_{12}, \ldots, \beta_{1q_2}, \exp(\beta_{21}), \exp(\beta_{22}), \ldots, \exp(\beta_{2q_2}), \ldots, \exp(\beta_{q_1 1}), \ldots, \exp(\beta_{q_1 q_2}))^T.$$

2. For the single monotone decreasing bivariate function along the x_1 direction:

   The re-parametrization is the same as above except for the representation of the matrix $\mathbf{\Sigma}_1$ which is now the $q_1 \times q_1$ matrix

   $$\mathbf{\Sigma}_1 = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ 1 & -1 & 0 & \ldots & 0 \\ 1 & -1 & -1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & -1 & -1 & \ldots & -1 \end{pmatrix}. \tag{3.12}$$

3. For the single monotone increasing bivariate function along the x_2 direction:

   Let $\mathbf{I}_1$ be an identity matrix of size q_1, and let $\mathbf{\Sigma}_2$ be a $q_2 \times q_2$ matrix of the type (3.8). Then

   $$\mathbf{\Sigma} = \mathbf{I}_1 \otimes \mathbf{\Sigma}_2, \tag{3.13}$$

   and

   $$\tilde{\beta} = (\beta_{11}, \exp(\beta_{12}), \ldots, \exp(\beta_{1q_2}), \beta_{21}, \exp(\beta_{22}), \ldots, \exp(\beta_{2q_2}), \ldots, \beta_{q_1 1}, \exp(\beta_{q_1 2}), \ldots, \exp(\beta_{q_1 q_2}))^T.$$

4. For the single monotone decreasing bivariate function along the x_2 direction:

   Everything is as in 3 except that $\mathbf{\Sigma}_2$ is a $q_2 \times q_2$ matrix of type (3.12).

Then the generalized regression model for all four considered situations is of the usual form

$$g(\mu_i) = \mathbf{X}_i \mathbf{\Sigma} \tilde{\beta}, \tag{3.14}$$

where $\boldsymbol{\Sigma}$ and $\tilde{\boldsymbol{\beta}}$ have the corresponding representations according to the required shape constraint. After construction of the monotonic tensor product smooth, the next step is to estimate the model parameters. As previously, to overcome the issues with basis dimensions selection, the penalized log likelihood function will be maximized when fitting the model,

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\lambda_1 \boldsymbol{\beta}^T \mathbf{S}_1 \boldsymbol{\beta} - \frac{1}{2}\lambda_2 \boldsymbol{\beta}^T \mathbf{S}_2 \boldsymbol{\beta},$$

but now the penalization works in both directions of the two covariates separately. λ_1 and λ_2 are the two smoothing parameters for penalization in each direction. The following section will develop the penalty matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ for the double and single monotonicity cases.

3.3.3 Penalties for double and single monotonicity

For double monotonicity the penalties may be written as the following:

$$\mathcal{P} = \lambda_1 \sum_{j=2}^{q_1-1} \sum_{k=1}^{q_2} (\triangle_1^1 \beta_{jk})^2 + \lambda_2 \sum_{j=1}^{q_1} \sum_{k=2}^{q_2-1} (\triangle_2^1 \beta_{jk})^2, \tag{3.15}$$

where

$$\triangle_1^1 \beta_{jk} = \beta_{(j+1),k} - \beta_{jk}, \;\; \triangle_2^1 \beta_{jk} = \beta_{j,(k+1)} - \beta_{jk}.$$

In matrix notation

$$\mathcal{P} = \lambda_1 \boldsymbol{\beta}^T \mathbf{S}_1 \boldsymbol{\beta} + \lambda_2 \boldsymbol{\beta}^T \mathbf{S}_2 \boldsymbol{\beta},$$

where $\mathbf{S}_1 = \mathbf{P}_1^T \mathbf{P}_1$ and $\mathbf{S}_2 = \mathbf{P}_2^T \mathbf{P}_2$,

$$\mathbf{P}_1 = \mathbf{P}_{m1} \otimes \mathbf{I}_2 \;\text{ and }\; \mathbf{P}_2 = \mathbf{I}_1 \otimes \mathbf{P}_{m2},$$

where

$$\mathbf{P}_{mj} = \begin{pmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ 0 & 0 & -1 & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 \end{pmatrix}, \tag{3.16}$$

$j = 1, 2$, are $(q_1 - 1) \times q_1$ and $(q_2 - 1) \times q_2$ matrices for $j = 1$ and 2 correspondingly, and $\mathbf{I}_j$ are identity matrices of sizes q_1 and q_2 respectively.

The penalties for single monotonicity along x_1 is

$$\mathcal{P} = \lambda_1 \boldsymbol{\beta}^T \mathbf{S}_1 \boldsymbol{\beta} + \lambda_2 \boldsymbol{\beta}^T \tilde{\mathbf{S}}_2 \boldsymbol{\beta}, \tag{3.17}$$

where $\mathbf{S}_1$ is defined as above. The degree of smoothness in the unconstrained direction can be controlled by the second-order difference penalties applied to the non-exponentiated working parameters $\beta_{11}, \ldots, \beta_{1q_2}$, and by the first-order difference penalties for the rest of the working parameters,

$$\boldsymbol{\beta}^T \tilde{\mathbf{S}}_2 \boldsymbol{\beta} = \sum_{k=1}^{q_2-2} \left(\triangle_2^2 \beta_{1k}\right)^2 + \sum_{j=2}^{q_1} \sum_{k=1}^{q_2-1} (\triangle_2^1 \beta_{jk})^2,$$

where $\triangle_2^2 \beta_{1k} = \beta_{1,(k+2)} - 2\beta_{1,(k+1)} + \beta_{1k}$. The second-order difference penalties are applied to $\beta_{11}, \ldots, \beta_{1q_2}$, since they achieve the same purpose as the first-order difference penalties for the exponentiated parameters, in that both penalizations result in straight lines when $\lambda \to \infty$.

$\tilde{\mathbf{S}}_2$ can be represented as $\tilde{\mathbf{S}}_2 = \tilde{\mathbf{P}}_2^T \tilde{\mathbf{P}}_2$, where

$$\tilde{\mathbf{P}}_2 = \begin{pmatrix} \mathbf{P}_{u2} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{s2} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_{s2} & \ldots & \mathbf{0} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{P}_{s2} \end{pmatrix},$$

$\mathbf{P}_{u2}$ and $\mathbf{P}_{s2}$ are $(q_2 - 2) \times q_2$ and $(q_2 - 1) \times q_2$ matrices respectively of the following type

$$\mathbf{P}_{u2} = \begin{pmatrix} 1 & -2 & 1 & 0 & \ldots & 0 \\ 0 & 1 & -2 & 1 & \ldots & 0 \\ 0 & 0 & 1 & -2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & \ldots & 1 \end{pmatrix}, \quad \mathbf{P}_{s2} = \begin{pmatrix} -1 & 1 & 0 & \ldots & 0 \\ 0 & -1 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 1 \end{pmatrix},$$

and $\mathbf{0}$ are null matrices of the corresponding dimensions.

Tensor product penalties for the single monotonicity restriction on the second covariate can be obtained easily in a similar way. As for the univariate cases, the penalties will keep the parameter estimates close to each other, resulting in similar increments in the model coefficients of marginal smooths. When $\lambda_j \to \infty$ such penalization results in straight lines for marginal curves.

Given the values of λ_1 and λ_2, to fit the model (3.14) the penalized log likelihood function $l_p(\boldsymbol{\beta})$ can be maximized using the approach described in Chapter 1.

3.4 Simulations

To illustrate the performance of the proposed shape-preserving smoothers, several simulation examples are presented in this section.

Example 1. Consider a single smooth term Gaussian model with mixed constraint restrictions, $y_i = f(x_i) + \epsilon_i$, $i = 1, \ldots, n$, where ϵ_i are i.i.d. random variables following $N(\mu = 0, \sigma^2)$. Two mixed constraint functions were investigated

$$f_1(x_1) = \log(x_1), \quad f_2(x_2) = (x_2 - 3)^6,$$

where $f_1(x_1)$ is subject to a monotone increase plus concavity constraint and $f_2(x_2)$ is monotone decreasing and convex.

One hundred values of the covariates, x_1 and x_2, were simulated from the uniform distribution on $[1, 100]$ and $[-1, 2]$ respectively, and the true values of the functions were calculated. Three noise levels were considered, $\sigma = 0.05$, 0.10, and 0.20, to generate the values of the response variable y_i. To model the data, P-splines with a monotone increase plus concavity constraint for the first function, and P-splines with a monotone decrease and convexity constraint for the second one were used, $q = 15$ in both cases. The models were fitted by penalized likelihood maximization with the smoothing parameter selected by GCV. The optimal models are shown in Figure 3-8. The effective degrees of freedom of the mixed constrained and unconstrained fits and the minimal GCV scores are given in Tables 3.1 and 3.2. The apparent observed pattern in τ is due to the results presented being from a single realization of the single data set.

Table 3.1: The effective degrees of freedom and minimal GCV scores for the one-dimensional Gaussian model with monotone increase and concavity constraint, $f_1(x_1)$.

	$\sigma = 0.05$		$\sigma = 0.1$		$\sigma = 0.2$	
	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$
Monotone model	3.722	$3.628 \cdot 10^{-3}$	5.00	$7.608 \cdot 10^{-3}$	3.25	0.03794
Unconstrained model	10.06	$3.932 \cdot 10^{-3}$	10.38	$7.515 \cdot 10^{-3}$	4.19	0.037834

From the figures one may note that the unconstrained models do not reflect the shape constraints for both functions for all three levels of noise. As in the monotone case the shape constrained fits are smoother than the unconstrained models and the effective degrees of freedom of the latter are more than that of the constrained fits.

Example 2. The next example considers a single term binomial model subject to a

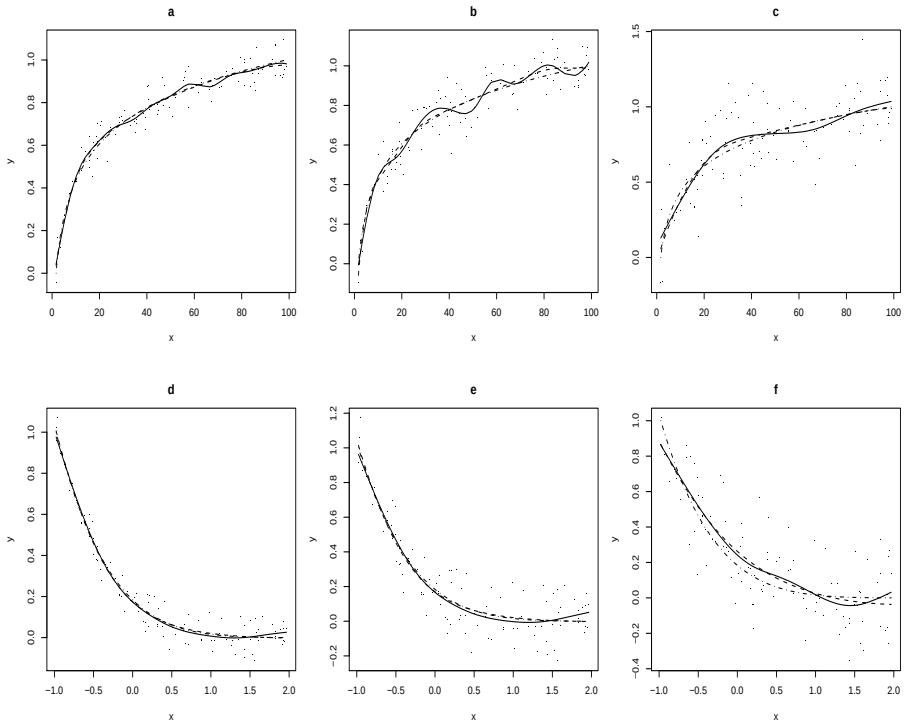


Figure 3-8: (a)-(c) Illustration of single term Gaussian models subject to a monotone increase and concavity constraint for three noise levels. (a) f_1, $\sigma = 0.05$, (b) f_1, $\sigma = 0.10$, (c) f_1, $\sigma = 0.20$. (d)-(f) Illustration of the single term Gaussian models subject to a monotone decrease and convexity constraint for three noise levels. (d) f_2, $\sigma = 0.05$, (e) f_2, $\sigma = 0.10$, (f) f_2, $\sigma = 0.20$. The mixed constraint fits are represented as dashed lines, the unconstrained fits as solid lines, and the true functions as dot dashed lines.

Table 3.2: The effective degrees of freedom and minimal GCV scores for the one-dimensional Gaussian model with monotone decrease and convexity constraint, $f_2(x_2)$.

	$\sigma = 0.05$		$\sigma = 0.1$		$\sigma = 0.2$	
	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$
Monotone model	2.98	$3.607 \cdot 10^{-3}$	2.55	0.01425	2.03	0.03351
Unconstrained model	5.31	$3.712 \cdot 10^{-3}$	4.34	0.01457	4.12	0.03348

monotone increase plus concavity constraint

$$\text{logit}(\mu_i) = f(x_i),$$

where $\text{logit}(\mu_i) = \log\{\mu_i/(1-\mu_i)\}$, $\mu_i = \text{E}(Y_i)$, $Y_i \sim \text{Bin}(n_b, \mu_i)$, n_b is a binomial denominator, $f(x_i) = \log(x_i)$. The $n = 200$ values of the covariate were simulated from the uniform distribution on $[1, 100]$, and before proceeding further the function f was scaled such that, the binomial probabilities belong to the interval $[0.02, 0.98]$. Three levels of noise were selected by setting $n_b = 1, 2$, and 4.

The data set was modeled using a cubic P-spline with a monotone increase and concavity restriction and $q = 20$. The model was fitted by the penalized likelihood maximization with λ selected by UBRE. Also unconstrained fits were obtained for all noise levels using unconstrained P-splines, with the same number of basis functions. Figure 3-9 illustrates this example. It shows the advantage of the proposed mixed constraint smoother, which reproduces the monotone and concave curve while the unconstrained fits tend to be too wiggly. Table 3.3 shows the edf of the fits together with the minimal UBRE scores.

Table 3.3: The effective degrees of freedom and minimal UBRE scores for the one-dimensional binomial model with monotone increase and concavity constraint.

	$n_b = 1$		$n_b = 2$		$n_b = 4$	
	τ	$\mathcal{V}_u$	τ	$\mathcal{V}_u$	τ	$\mathcal{V}_u$
Monotone model	2.04	-0.44742	2.71	-0.62343	2.27	-0.75867
Unconstrained model	7.77	-0.45045	2.98	-0.62945	2.97	-0.75105

The single bivariate term models subject to double or single monotonicity are considered in the next two examples.

Example 3. In this example the performance of the bivariate P-spline subject to a

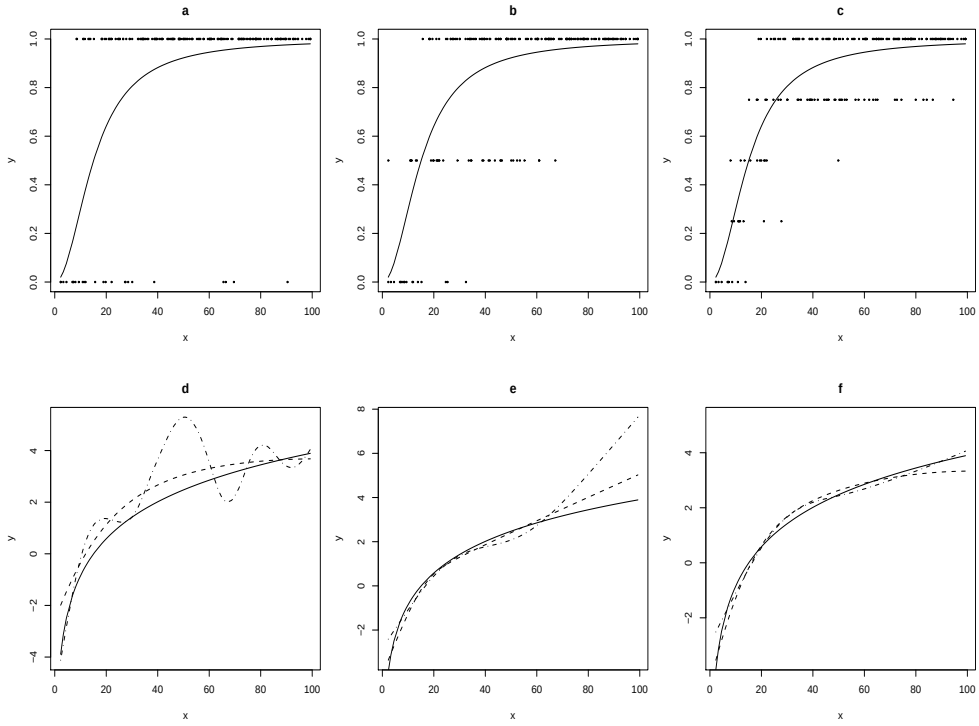


Figure 3-9: Illustration of single term binomial models subject to monotone increase plus concavity constraint. (a)-(c) Illustration of the simulated points and the true function for three noise levels, (a) $n_b = 1$, (b) $n_b = 2$, (c) $n_b = 4$. (d)-(f) Representation of the linear predictors: the true linear predictor (solid line), the mixed constrained fits on a linear predictor scale (dashed line), and the unconstrained fits on a linear predictor scale (dot dashed line), (d) $n_b = 1$, (e) $n_b = 2$, (f) $n_b = 4$.

double monotone increase restriction is shown using the following model

$$y_i = f(x_{1i}, x_{2i}) + \epsilon_i, \;\; \epsilon_i \sim N(\mu = 0, \sigma),$$

$$f(x_{1i}, x_{2i}) = \exp(4x_{1i}) / \{1 + \exp(4x_{1i})\} + 2\exp(2x_{2i} - 2),$$

where monotonicity is assumed along both directions, x_1 and x_2. The covariate values were generated from the uniform distributions on $[-1, 3]$ and $[0, 1]$ respectively. The function was scaled to have values on $[0, 1]$, and the level of noise was $\sigma = 0.10$. For both bivariate P-splines (unconstrained and double monotone increasing) $q_1 = q_2 = 10$ marginal basis functions were used. The true function and optimal fitted smooths obtained by the GCV minimization are shown on Figure 3-10. From the bottom right panel it may be seen that the unconstrained smooth is not monotone. The τ of the double monotone fit was 7.2 with the value of the GCV score 0.01028, for the unconstrained fit: $\tau = 25.42$ and $\mathcal{V}_g = 0.010443$.

Example 4. The last example illustrates bivariate term models subject to monotonicity along one direction only. Consider the same model as in the previous example, but now

$$f_1(x_{1i}, x_{2i}) = -\exp(4x_{1i}) / \{1 + \exp(4x_{1i})\} + 2\sin(\pi x_{2i}),$$

for the case of a monotone decrease constraint along the first covariate x_1. Secondly, consider

$$f_2(x_{3i}, x_{4i}) = 2\sin(\pi x_{3i}) + \exp(4x_{4i}) / \{1 + \exp(4x_{4i})\},$$

which is monotone increasing along the second covariate x_4. All covariate values were generated from uniform distributions, x_2 and x_3 on $[0, 1]$, and x_1 and x_4 on $[-1, 3]$. Both functions were scaled to $[0, 1]$, and the standard deviation of the Gaussian distribution was taken as $\sigma = 0.10$. Ten basis functions were used for the marginal constrained and unconstrained P-splines for both cases. The results of this simulated example are shown in Figure 3-11 and Figure 3-12. Table 3.4 shows the optimal GCV scores and the effective degrees of freedom of the fits.

Table 3.4: The effective degrees of freedom and minimal GCV scores for the Gaussian models with single monotonicity.

	f_1		f_2	
	τ	$\mathcal{V}_g$	τ	$\mathcal{V}_g$
Monotone model	7.91	0.010272	8.11	0.010284
Unconstrained model	27.37	0.010426	29.68	0.010421

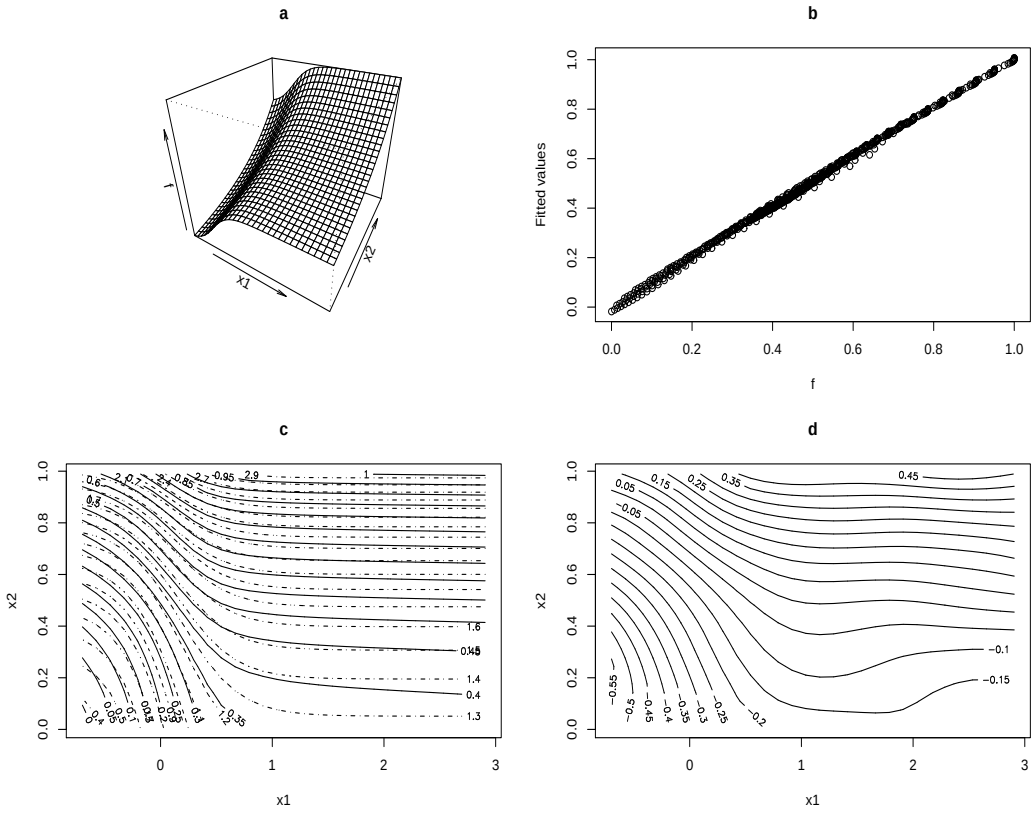


Figure 3-10: Illustration of the single bivariate term Gaussian model subject to double monotonicity. (a) Perspective plot of the true function. (b) Plot of the values of the true function against fitted values of the double monotonic fitted smooth. (c) Contour plots of the true function (dot dashed lines) and double monotonic fit (solid lines). (d) Contour plot of the unconstrained fit.

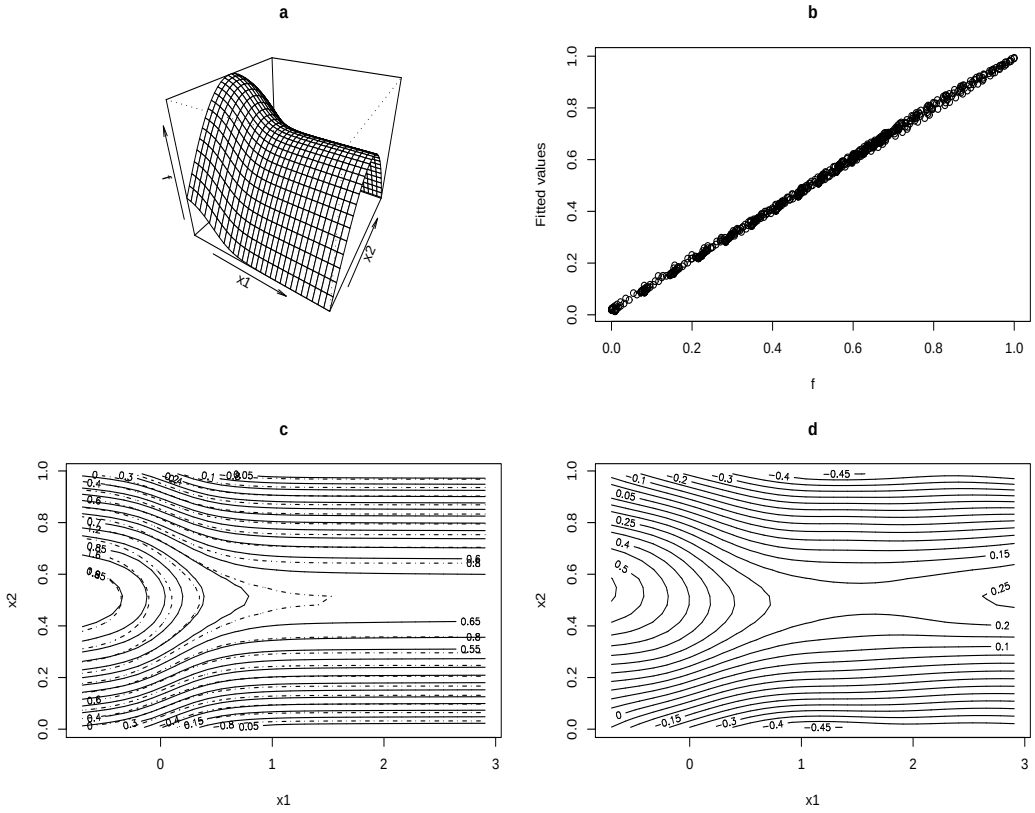


Figure 3-11: Illustration of the single bivariate term Gaussian model subject to single monotone decreasing constraint along the first covariate x_1. (a) Perspective plot of the true function, $f_1(x_{1i}, x_{2i})$. (b) Plot of the values of the true function against fitted values of the single monotone decreasing smooth. (c) Contour plots of the true function (dot dashed lines) and single monotonic fitted smooth (solid lines). (d) Contour plot of the unconstrained fit.

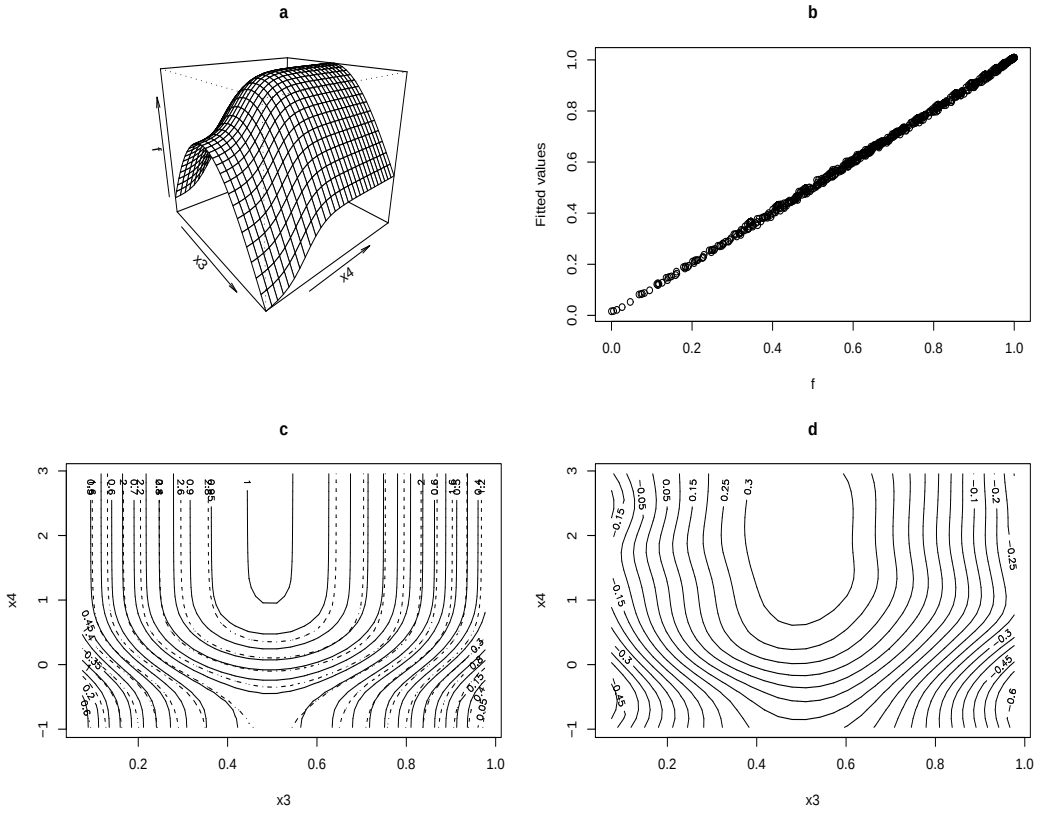


Figure 3-12: Illustration of the single bivariate term Gaussian model subject to single monotone increasing constraint along the second covariate x_4. (a) Perspective plot of the true function, $f_2(x_{3i}, x_{4i})$. (b) Plot of the values of the true function against fitted values of the single monotone increasing smooth. (c) Contour plots of the true function (dot dashed lines) and single monotonic smooth (solid lines). (d) Contour plot of the unconstrained fit.

The figures clearly show the advantage of the P-splines with monotonicity constraints. It should be mentioned that a general theory to monotone smoothing of functions with any number of covariates can be developed using the approach of Section 3.3. This can be a topic of further research.

Chapter 4

Generalized additive models with shape constraints on some terms

The previous two chapters were dealing with single smooth generalized regression models under shape constraint restrictions. This chapter generalizes the proposed approach to generalized additive models with shape constraints on some terms (mono-GAM). For simplicity of presentation, the discussion starts with an additive model with monotonicity constraint imposed only on one smooth term, and only B-spline bases used for representation of unconstrained smooth terms. Then it extends to a more general structure of mono-GAM which incorporates any available penalized regression splines for each unconstrained term, including multivariate terms, and bivariate terms with monotonicity constraints. The fitting procedure of a mono-GAM is based on an outer quasi-Newton iteration to update the log of the multiple smoothing parameters, $\rho_k = \log(\lambda_k)$, and each step of this procedure requires an inner Newton based P-IRLS to obtain working model parameters, $\boldsymbol{\beta}$, given $\boldsymbol{\lambda}$. The chapter introduces an efficient way of calculating derivatives of the working parameters with respect to ρ_k by extending the approach proposed in Wood (2011).

4.1 Penalized regression spline representation

4.1.1 Mono-GAM with monotonic and unconstrained univariate P-splines

Consider a generalized additive model of the following structure:

$$g(\mu_i) = \mathbf{X}_i^* \boldsymbol{\delta} + f_1(x_{1i}) + f_2(x_{2i}) + \ldots + f_p(x_{pi}), \;\; i = 1, \ldots, n, \tag{4.1}$$

where $\mu_i = \mathrm{E}(Y_i)$, g is a known link function, not necessarily canonical, and $Y_i \sim$ some exponential family distribution with the probability density function as in (2.2). $\mathbf{X}_i^*$ is the i^{th} row of a model matrix for strictly parametric model components (usually including the intercept), with corresponding vector of parameters $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_{q_0})^T$, f_j are smooth functions of the covariates x_j, $j = 1, \ldots, p$. For simplicity, suppose that an additional monotonicity constraint is imposed only on the first function, $f_1(x_1)$. After setting up the model with a monotonicity constraint on only one smooth function, extensions to a GAM with monotonicity or mixed constraints on several functions are not difficult.

To estimate the model (4.1) one can specify B-spline bases for each smooth function. Given a sequence of evenly spaced knots, $k_{j1} < k_{j2} < \ldots < k_{j,q_j+m_j+2}$, where q_j is the number of basis functions for the j^{th} smooth, an $(m_j + 1)^{th}$ order B-spline can be represented as:

$$f_j(x_j) = \sum_{l=1}^{q_j} B_{jl}^{m_j}(x_j)\beta_{jl}, \quad j = 2, \ldots, p,$$

where

$$B_{jl}^{m_j}(x_j) = \frac{x_j - k_{jl}}{k_{j,l+m_j+1} - k_{jl}} B_{jl}^{m_j-1}(x_j) + \frac{k_{j,l+m_j+2} - x_j}{k_{j,l+m_j+2} - k_{j,l+1}} B_{j,l+1}^{m_j-1}(x_j), \quad l = 1, \ldots, q_j,$$

$$B_{jl}^{-1}(x_j) = \begin{cases} 1, & k_{jl} \le x_j \le k_{j,l+1} \\ 0, & \text{otherwise.} \end{cases}$$

The first monotonic term of the GAM has the representation introduced in Section 2.1:

$$f_1(x_1) = \sum_{l=1}^{q_1} B_{1l}^{m_1}(x_1)\gamma_{1l},$$

where

$$\gamma_{11} = \beta_{11}, \quad \gamma_{1l} = \beta_{11} \pm \sum_{k=2}^{l} \exp(\beta_{1k}), \quad l = 2, \ldots, q_1.$$

The signs '+' or '−' in the coefficients γ_{1l} of the monotone smooth stand for increasing or decreasing constraints respectively.

Given the bases, the i^{th} row of the model matrix for each smooth will be

$$\mathbf{X}'_{j,i} = \left\{ B_{j1}^{m_j}(x_{ji}), B_{j2}^{m_j}(x_{ji}), ..., B_{jq_j}^{m_j}(x_{ji}) \right\}, \quad j = 1, \ldots, p.$$

Then each unconstrained smooth may be written as:

$$\mathbf{f}_j = \mathbf{X}'_j \boldsymbol{\beta}'_j, \;\; j = 2, \ldots, p,$$

where $\mathbf{f}_j$ is the vector such that $\mathbf{f}_{ji} = f_j(x_{ji})$, and $\boldsymbol{\beta}'_j = \left(\beta_{j1}, \beta_{j2}, \ldots, \beta_{jq_j}\right)^T$.

The constrained smooth can be represented as

$$\mathbf{f}_1 = \mathbf{X}'_1 \boldsymbol{\Sigma}'_1 \tilde{\boldsymbol{\beta}}'_1,$$

where

$$\boldsymbol{\Sigma}'_1 = \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ 1 & \pm 1 & 0 & ... & 0 \\ 1 & \pm 1 & \pm 1 & ... & 0 \\ . & . & . & . & . \\ 1 & \pm 1 & \pm 1 & ... & \pm 1 \end{pmatrix} \tag{4.2}$$

is a $q_1 \times q_1$ matrix with the elements $+1$ for increasing case and -1 for decreasing case, and $\tilde{\boldsymbol{\beta}}'_1 = (\beta_{11}, \exp(\beta_{12}), \ldots, \exp(\beta_{1q_1}))^T$.

In order to deal with the identifiability problem of this model one may set a 'centering constraint' on each unconstrained smooth (Wood, 2006a), that is the sum of the values of the j^{th} smooth is set to be zero $\sum_{i=1}^{n} f_{ji}(x_{ji}) = 0$ or

$$\mathbf{1}^T \mathbf{X}'_j \boldsymbol{\beta}'_j = 0, \;\; j = 2, \ldots, p.$$

To satisfy this restriction, first, find a matrix $\mathbf{Z}_j$, with $q_j - 1$ orthogonal columns, which satisfies

$$\mathbf{1}^T \mathbf{X}'_j \mathbf{Z}_j = \mathbf{0}.$$

By setting $\boldsymbol{\beta}'_j = \mathbf{Z}_j \boldsymbol{\beta}_j$, where $\boldsymbol{\beta}_j$ is a vector of $q_j - 1$ new parameters, the j^{th} smooth can be written as $\mathbf{f}_j = \mathbf{X}_j \boldsymbol{\beta}_j$ with $\mathbf{X}_j = \mathbf{X}'_j \mathbf{Z}_j$. The centering constraint will be satisfied by this re-parametrization.

To handle the identifiability problem of the monotone term one may constrain $\gamma_{11} = \beta_{11} = 0$, since this parameter is the 'intercept' term for the monotone smooth. Then the i^{th} row of the model matrix of the first smooth is

$$\mathbf{X}_{1,i} = \left\{ B_{12}^{m_1}(x_{1i}), B_{13}^{m_1}(x_{1i}), ..., B_{1q_1}^{m_1}(x_{1i}) \right\},$$

and the parameter vector $\tilde{\boldsymbol{\beta}}_1 = (\exp(\beta_{12}), \exp(\beta_{13}), \ldots, \exp(\beta_{1q_1}))^T$.

Having done this, the generalized additive model with monotonicity constraint

(mono-GAM) (4.1) may be written as:

$$g(\mu_i) = \mathbf{X}_i\tilde{\boldsymbol{\beta}}, \tag{4.3}$$

where $\mathbf{X} = [\mathbf{X}^* : \mathbf{X}_1\boldsymbol{\Sigma}_1 : \mathbf{X}_2 : \ldots : \mathbf{X}_p]$, $\tilde{\boldsymbol{\beta}}^T = \left[\boldsymbol{\delta}^T, \tilde{\boldsymbol{\beta}}_1^T, \boldsymbol{\beta}_2^T, \ldots, \boldsymbol{\beta}_p^T\right]$, and $\boldsymbol{\Sigma}_1$ is the first $q_1 - 1$ rows and columns of $\boldsymbol{\Sigma}_1'$ in (4.2).

To control the degree of smoothing, the smoothness penalties based on k^{th}-order differences of the working model coefficients is applied to each smooth of the mono-GAM. For the monotone smooth, as in the model with one monotone term (Section 2.1), the first-order difference penalty is used

$$P_1 = \sum_{l=2}^{q_1-1} (\beta_{1,l+1} - \beta_{1l})^2,$$

while for the unconstrained terms the degree of smoothness can be controlled by the second-order difference penalties

$$P_j = \sum_{l=1}^{q_j-2} (\beta_{j,l+2} - 2\beta_{j,l+1} + \beta_{jl})^2, \;\; j = 2, \ldots, p.$$

These penalties can be written in terms of the full working model coefficients vector $\boldsymbol{\beta}^T = \left[\boldsymbol{\theta}^T, \boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_p^T\right]$,

$$P_j = \boldsymbol{\beta}^T\mathbf{S}_j\boldsymbol{\beta},$$

where $\mathbf{S}_j$ is a $q \times q$ matrix with zeros everywhere except for elements which correspond to the coefficients of the j^{th} smooth, $q = q_0 + q_1 + \ldots + q_p - p$ is the total number of the coefficients. The $(q_j - 1) \times (q_j - 1)$ submatrix of these nonzero elements for an unconstrained term is

$$\begin{pmatrix} 1 & -2 & 1 & 0 & 0 & 0 & . & . \\ -2 & 5 & -4 & 1 & 0 & 0 & . & . \\ 1 & -4 & 6 & -4 & 1 & 0 & . & . \\ 0 & 1 & -4 & 6 & -4 & 1 & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \end{pmatrix},$$

while for the first term it is as in (2.10). It should be mentioned that higher-order difference penalties are possible here. For convenience the total penalty is represented

as a single matrix

$$\mathbf{S} = \sum_{k=1}^{p} \lambda_k \mathbf{S}_k,$$

where λ_k is a smoothing parameter for the k^{th} smooth, controlling its amount of smoothness.

After setting the penalties for each function the penalized log likelihood for the mono-GAM can be defined as

$$l_p(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{1}{2}\boldsymbol{\beta}^T \mathbf{S} \boldsymbol{\beta}. \tag{4.4}$$

Given the values of λ_k, to estimate $\boldsymbol{\beta}$ the penalized log likelihood (4.4) should be maximized.

4.1.2 Mono-GAM of a general structure

To illustrate the set up of mono-GAMs of a general structure this section considers a slightly more complicated example. A more general structure of the mono-GAM will now incorporate not only the P-splines but any available penalized regression splines for each unconstrained smooth term, including multivariate terms, and bivariate terms with monotonicity constraints. An example of the general mono-GAM may be written as

$$g(\mu_i) = \mathbf{X}_i^* \boldsymbol{\delta} + f_1(x_{1i}) + f_2(x_{2i}, x_{3i}) + f_3(x_{4i}) + f_4(x_{5i}, x_{6i}) \cdot x_{7i} + \ldots, \;\; i = 1, \ldots, n, \tag{4.5}$$

where some of the smooth terms are subject to monotonicity or monotonicity and convexity constraints. An additional feature is multiplication of the smooth term by a covariate: such models are referred to as 'variable coefficient models' (Hastie and Tibshirani, 1993; Wood, 2006a), and variables such as x_7 are sometimes known as 'by' variables.

In order to see how to estimate a mono-GAM of a general structure, consider the model (4.5) but with only four smooth terms

$$g(\mu_i) = \mathbf{X}_i^* \boldsymbol{\delta} + f_1(x_{1i}) + f_2(x_{2i}, x_{3i}) + f_3(x_{4i}) + f_4(x_{5i}, x_{6i}) \cdot x_{7i}, \tag{4.6}$$

where the first two functions $f_1(x_{1i})$ and $f_2(x_{2i}, x_{3i})$ are considered to be unconstrained, $f_3(x_{4i})$ is assumed to be monotone increasing, $f_4(x_{5i}, x_{6i})$ is subject to double monotone increase, and the smooth term f_4 is multiplied by the 'by' variable x_7.

To estimate such a model each smooth function of the model can be specified by means of penalized regression splines and for representation of each monotone smooth

function monotone P-splines can be used. Having chosen an appropriate set of basis functions, the first unconstrained smooth functions may be represented as

$$f_1(x_{1i}) = \sum_{j=1}^{q_1} B_{1j}(x_{1i})\beta'_{1j}.$$

The tensor product basis can be used for representing the second bivariate function

$$f_2(x_{2i}, x_{3i}) = \sum_{j=1}^{q_2}\sum_{k=1}^{q_3} B_{2j}(x_{2i}) \cdot B_{3j}(x_{3i})\beta'_{2,jk}.$$

For the two monotone smooths the approach of the previous sections will be used

$$f_3(x_{4i}) = \sum_{j=1}^{q_4} B_{4j}^{m_4}(x_{4i})\gamma_{3j},$$

where

$$\gamma_{31} = \beta_{31}, \;\; \gamma_{3l} = \beta_{31} + \sum_{k=2}^{l} \exp(\beta_{3k}), \;\; l = 2, \ldots, q_4,$$

and

$$f_4(x_{5i}, x_{6i}) = \sum_{j=1}^{q_5}\sum_{k=1}^{q_6} B_{5j}(x_{5i}) \cdot B_{6j}(x_{6i})\gamma'_{4,jk},$$

where

$$\gamma_{4,11} = \beta_{4,11}, \;\; \gamma_{4,1k} = \beta_{4,11} + \sum_{s=2}^{k} \exp(\beta_{4,1s}), \;\; k = 2, \ldots, q_6,$$

$$\gamma_{4,jk} = \beta_{4,11} + \sum_{s=2}^{k} \exp(\beta_{4,1s}) + \sum_{l=2}^{j}\sum_{s=1}^{k} \exp(\beta_{4,ls}), \;\; j = 2, \ldots, q_5, \;\; k = 1, \ldots, q_6,$$

in the notations of Section 3.3.1.

In the vector-matrix notations each smooth terms will be written as

$$\mathbf{f}_1 = \mathbf{X}'_1\boldsymbol{\beta}'_1, \;\; \mathbf{f}_2 = \mathbf{X}'_2\boldsymbol{\beta}'_2,$$

where $\mathbf{X}'_{2i} = \mathbf{X}_{2i} \otimes \mathbf{X}_{3i}$,

$$\mathbf{f}_3 = \mathbf{X}'_3\boldsymbol{\Sigma}'_3\tilde{\boldsymbol{\beta}}'_3,$$

and

$$\mathbf{f}_4 = \mathbf{X}'_4\boldsymbol{\Sigma}'_4\tilde{\boldsymbol{\beta}}'_4,$$

where $\mathbf{X}'_{4i} = \mathbf{X}_{5i} \otimes \mathbf{X}_{6i}$, $\mathbf{\Sigma}'_4 = \mathbf{\Sigma}_5 \otimes \mathbf{\Sigma}_6$. Σ'_3, $\mathbf{\Sigma}_5$ and $\mathbf{\Sigma}_6$ are as (2.7) but of the corresponding dimensions.

The model with smooth terms defined as above is not identifiable. To deal with the identifiability for the univariate smooths the same approach as in the previous section may be used. The 'centering constraint' can also be imposed on unconstrained smooth with more than one variable. How to handle the identifiability problem with the monotonic bivariate smooths will be covered in the next section. For now denote the re-parametrization matrix for $f_4(x_{5i}, x_{6i})$ as $\mathbf{Z}$. After imposing the corresponding identifiability constraints, the model (4.6) can be represented as in the previous section

$$g(\mu_i) = \mathbf{X}_i \tilde{\boldsymbol{\beta}},$$

where now

$$\mathbf{X} = \left[\mathbf{X}^* : \mathbf{X}_1 : \mathbf{X}_2 : \mathbf{X}_3\mathbf{\Sigma}_3 : \mathbf{J}_{x_7}\mathbf{X}'_4\mathbf{\Sigma}'_4\mathbf{Z}\right],$$

$\tilde{\boldsymbol{\beta}}^T = \left[\boldsymbol{\delta}^T, \boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T, \tilde{\boldsymbol{\beta}}_3^T, \tilde{\boldsymbol{\beta}}_4^T\right]$, $\mathbf{J}_{x_7}$ is a diagonal matrix with the values of x_{7i} on the main diagonal.

For measuring the wiggliness of the functions the penalties for each smooth described in the previous sections will be subtracted from the log-likelihood function for the model.

4.1.3 Identifiability constraint for tensor product with monotonic P-splines

Consider the bivariate smooth $f_p(x_i, z_i) = \mathbf{X}'_{pi}\mathbf{\Sigma}'_p\tilde{\boldsymbol{\beta}}'_p$, where the matrix $\mathbf{\Sigma}_p$ and vector $\tilde{\boldsymbol{\beta}}_p$ have the representations described in Section 3.3 and the unconstrained working vector of parameters, $\boldsymbol{\beta}'_p$, has the following arrangement

$$\boldsymbol{\beta}'_p = \left(\beta'_{p,11}, \beta'_{p,12} \ldots, \beta'_{p,1q_2}, \ldots, \beta'_{p,q_11}, \beta'_{p,q_12} \ldots, \beta'_{p,q_1q_2}\right)^T.$$

For the bivariate function with double monotonicity, by analogy with the univariate case we may set the first parameter $\beta'_{p,11} = 0$ as an identifiability constraint, since this parameter works as an intercept for marginal univariate smooths in the tensor product. But for the single monotonicity a different constraint will be used, since $\beta'_{p,11}$ is not an intercept in this case.

For the single monotonicity along x the identifiability constraint can be of the form:

$$\sum_{j=1}^{q_2} \beta'_{p,1j} = 0,$$

that is the sum of non-exponentiated working parameters is set to be zero. This is reasonable since every non-exponentiated parameter is an intercept of the corresponding marginal univariate smooth along the x. The same argument gives the identifiability constraint for the single monotonicity along z, but in this case the summation is performed along the first index in our notations: $\sum_{k=1}^{q_1} \beta'_{p,k1} = 0$.

These constraints can be imposed into the model by introducing a re-parametrization matrix $\mathbf{Z}$, such that $\tilde{\boldsymbol{\beta}}'_p = \mathbf{Z}\tilde{\boldsymbol{\beta}}_p$. For the single monotonicity along x,

$$\mathbf{Z} = \left(\begin{array}{c|c} \mathbf{Z}' & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right),$$

where

$$\mathbf{Z}' = \begin{pmatrix} -1 & 0 & 0 & 0 & \dots & 0 \\ 1 & -1 & 0 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ 0 & 0 & 1 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{pmatrix}$$

is a $q_2 \times (q_2 - 1)$ matrix, and the dimension of parameter vector $\tilde{\boldsymbol{\beta}}_p$ is now one less than the dimension of $\tilde{\boldsymbol{\beta}}'_p$. Hence, the bivariate smooth may be written as $f_p(x_i, z_i) = \mathbf{X}'_{pi}\boldsymbol{\Sigma}'_p\mathbf{Z}\tilde{\boldsymbol{\beta}}_p$ or $f_p(x_i, z_i) = \mathbf{X}_{pi}\tilde{\boldsymbol{\beta}}$, where $\mathbf{X}_{pi} = \mathbf{X}'_{pi}\boldsymbol{\Sigma}'_p\mathbf{Z}$. By analogy it is not difficult to find $\mathbf{Z}$ when the identifiability constraint is imposed on the function with monotonicity along z.

4.2 Fitting mono-GAM

The fitting procedure for the mono-GAM is analogous to the Newton based method presented in Sections 2.2 and 2.4. The only differences lie (i) in the notation of the penalty term of the penalized log likelihood, where $\lambda\mathbf{S}$ in the single smooth term becomes $\mathbf{S} = \sum_{k=1}^{p} \lambda_k \mathbf{S}_k$, (ii) in the fact that the model matrix $\mathbf{X}$ of the additive model now includes the matrices of parameter summation/subtraction $\boldsymbol{\Sigma}_j$, and (iii) in the definitions of the matrices $\mathbf{C}$ and $\mathbf{E}$, which are now $q \times q$ diagonal matrices with the following diagonal elements

$$C_j = \begin{cases} \exp(\beta_j), & \text{if } \beta_j \text{ is exponentiated in } \tilde{\boldsymbol{\beta}} \\ 1, & \text{otherwise} \end{cases}$$

and

$$E_j = \begin{cases} \sum_{i=1}^{n} \frac{\omega_i [\mathbf{XC}]_{ij}}{V(\mu_i) g'(\mu_i)} (y_i - \mu_i), & \text{if } \beta_j \text{ is exponentiated in } \tilde{\boldsymbol{\beta}} \\ 0, & \text{otherwise.} \end{cases}$$

Similarly to a single monotone model, for the additive case $\sqrt{|\mathbf{W}|}\mathbf{XC}$ is augmented by a matrix $\mathbf{B}$, where $\mathbf{B}^T\mathbf{B} = \mathbf{S}$, when forming the QR decomposition

$$\begin{bmatrix} \sqrt{|\mathbf{W}|}\mathbf{XC} \\ \mathbf{B} \end{bmatrix} = \mathbf{QR}. \tag{4.7}$$

Therefore, in the Newton algorithm, if $\boldsymbol{\beta}^{[k]}$ is the current estimate of $\boldsymbol{\beta}$, the next estimate is

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \mathbf{P}\mathbf{K}^T\sqrt{|\mathbf{W}|}\tilde{\mathbf{z}} - \mathbf{P}\mathbf{P}^T\mathbf{S}\boldsymbol{\beta}^{[k]}$$

(where $\mathbf{P}$, $\mathbf{K}$, and $\tilde{\mathbf{z}}$ are defined in Section 2.4).

The matrix for the effective degrees of freedom is

$$\mathbf{F} = \left\{(\mathbf{XC})^T\mathbf{WXC} - \mathbf{E} + \mathbf{S}\right\}^{-1}(\mathbf{XC})^T\mathbf{W}_1\mathbf{XC}. \tag{4.8}$$

Also, the same optimization issues as in Section 2.5 regarding the initialization of the model parameters, column rank deficiency of the model matrix, and the basis dimension, are relevant to the case of the mono-GAM. What differs in the mono-GAM case, is that grid search is not a practical strategy for finding multiple smoothing parameters. The following section therefore presents a computationally efficient method for estimating the multiple smoothing parameters.

4.3 Multiple smoothing parameter selection based on GCV/UBRE

Penalized likelihood maximization is used to estimate working model parameters, $\boldsymbol{\beta}$, given smoothing parameters $\boldsymbol{\lambda}$. This section discusses how to estimate multiple smoothing parameters. Section 2.6 introduced two criteria which can be minimized to estimate $\boldsymbol{\lambda}$: Mallows's C_p/UBRE (2.26), used when the scale parameter, ϕ, is known, and GCV (2.28) for an unknown scale parameter. When dealing with a single smooth generalized regression model the grid search may be used for smoothing parameter selection. However, for multiple smoothing parameters a computationally efficient (and stable) method must be developed.

One way to select the smoothing parameters of the GAM is to minimize the GCV or

UBRE scores for each working penalized linear model of the P-IRLS step. This method was proposed by Gu (1992) (see Wood, 2004 and 2006a, for further development) and is known as performance oriented iteration. The main problem with this approach is divergence or cycling of the P-IRLS scheme. Another alternative is to iterate the P-IRLS to convergence for each trial value of the smoothing parameter vector which is called 'nested' or 'outer' iteration. In this project an outer quasi-Newton iteration is used for minimizing UBRE/GCV to update $\hat{\rho}_k = \log(\hat{\lambda}_k)$ ($\log(\hat{\lambda}_k)$ is taken to avoid negative values of the smoothing parameter), and each step of this procedure will require an inner Newton based P-IRLS to obtain $\hat{\boldsymbol{\beta}}$, given $\hat{\boldsymbol{\lambda}}$ (Section 2.4).

For implementing a quasi-Newton iteration first order derivatives of the GCV or UBRE score with respect to ρ_k should be calculated. Both scores depend on the model deviance, $D(\hat{\boldsymbol{\beta}})$, and effective degrees of freedom, τ. Therefore, to calculate the derivatives of $D(\hat{\boldsymbol{\beta}})$ and τ with respect to ρ_k, first of all the derivatives of the working parameter vector, $\partial\hat{\boldsymbol{\beta}}/\partial\rho_k$, have to be obtained, which is the main challenge in this approach.

Wood (2006a) suggested expressions for calculating the derivative vector $\partial\hat{\boldsymbol{\beta}}/\partial\rho_k$ for each iteration step of the P-IRLS scheme, and used finite differencing of the first derivatives to get the Hessian. A computationally efficient and reliable method was developed by Wood (2008) which introduced a separate iterative procedure for calculation of $\hat{\boldsymbol{\beta}}$ derivatives. The first subsection presents an efficient way of obtaining derivatives of the model parameter estimates by extending the implicit function theorem approach taken in Wood (2011). The rest of the section covers calculation of all the other derivatives required in order to obtain the derivatives of the GCV or UBRE scores required for a quasi-Newton step.

4.3.1 Calculating the first derivatives of $\hat{\boldsymbol{\beta}}$ with respect to ρ_k

Let the penalized deviance be denoted by D_p :

$$D_p = D(\boldsymbol{\beta}) + \sum_k e^{\rho_k}\boldsymbol{\beta}^T\mathbf{S}_k\boldsymbol{\beta}. \tag{4.9}$$

For convenience (4.9) can be re-written as

$$D_p = D(\boldsymbol{\beta}) + P,$$

where $P = \sum_k e^{\rho_k}\boldsymbol{\beta}^T\mathbf{S}_k\boldsymbol{\beta}$.

Since maximizing the penalized log likelihood is the same as minimizing the penal-

ized deviance (4.9), the penalized maximum likelihood equations are equivalent to

$$\frac{\partial D_p}{\partial \beta_j} = 0, \;\; j = 1, \ldots, q, \tag{4.10}$$

q is the total number of parameters, and $\hat{\boldsymbol{\beta}}$ is the solution of these equations. From the Newton based P-IRLS we know that

$$\left[\frac{\partial^2 D_p}{\partial \beta_j \partial \beta_l}\right] = 2\left((\mathbf{XC})^T \mathbf{WXC} + \mathbf{S} - \mathbf{E}\right)$$

is invertible at convergence. Hence, implicit differentiation may be applied in order to compute the derivatives $\partial\hat{\boldsymbol{\beta}}/\partial\rho_k$. By differentiating both sides of the equations (4.10) with respect to ρ_k we get

$$\sum_{l=1}^{q} \frac{\partial^2 D_p}{\partial \beta_j \partial \beta_l} \frac{\partial \beta_l}{\partial \rho_k} + \frac{\partial D_p}{\partial \beta_j \partial \rho_k} = 0.$$

Therefore,

$$\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \rho_k} = -\left[\frac{\partial^2 D_p}{\partial \beta_j \partial \beta_l}\right]^{-1} \frac{\partial \nabla_\beta D_p}{\partial \rho_k}, \tag{4.11}$$

where

$$\frac{\partial \nabla_\beta D_p}{\partial \rho_k} = \frac{\partial \nabla_\beta P}{\partial \rho} = 2e^{\rho_k} \mathbf{S}_k \boldsymbol{\beta}.$$

Using the notation of Section 2.2 we have

$$\left[\frac{\partial^2 D_p}{\partial \beta_j \partial \beta_l}\right]^{-1} = \frac{1}{2} \mathbf{R}^{-1} \mathbf{U} (\mathbf{I} - \boldsymbol{\Lambda})^{-1} \mathbf{U}^T \mathbf{R}^{-T},$$

and, finally,

$$\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \rho_k} = -e^{\rho_k} \mathbf{P}\mathbf{P}^T \mathbf{S}_k \hat{\boldsymbol{\beta}}. \tag{4.12}$$

The calculation of the other required derivatives will be presented in the next sections.

4.3.2 Calculating the derivative of $D(\hat{\boldsymbol{\beta}})$

The first order partial derivatives of the deviance are

$$\frac{\partial D}{\partial \rho_k} = \sum_{j=1}^{q} \frac{\partial D}{\partial \hat{\beta}_j} \frac{\partial \hat{\beta}_j}{\partial \rho_k},$$

which requires the derivative with respect to $\hat{\boldsymbol{\beta}}$. From (2.27) and (2.12) it follows that

$$\frac{\partial D}{\partial \hat{\beta}_j} = -2\sum_{i=1}^{n} \omega_i \frac{y_i - \hat{\mu}_i}{V(\hat{\mu}_i)g'(\hat{\mu}_i)} \left[\mathbf{X}\right]_{ij}, \text{ if } \beta_j \text{ is not exponentiated in the parameter vector}$$

$\tilde{\boldsymbol{\beta}}$, and

$$\frac{\partial D}{\partial \hat{\beta}_j} = -2\sum_{i=1}^{n} \omega_i \frac{y_i - \hat{\mu}_i}{V(\hat{\mu}_i)g'(\hat{\mu}_i)} \left[\mathbf{X}\right]_{ij} \exp(\hat{\beta}_j), \text{ otherwise.}$$

Let $\mathbf{c}$ be a vector with

$$c_i = -2\omega_i(y_i - \hat{\mu}_i)/\{V(\hat{\mu}_i)g'(\hat{\mu}_i)\}, \quad i = 1, \ldots, n,$$

then the vector of the first order derivatives of D is $\partial D/\partial\hat{\boldsymbol{\beta}} = \mathbf{C}\mathbf{X}^T\mathbf{c}$, and

$$\frac{\partial D}{\partial \rho} = \left(\frac{\partial D}{\partial \hat{\boldsymbol{\beta}}}\right)^T \frac{\partial \hat{\boldsymbol{\beta}}}{\partial \rho_k}.$$

4.3.3 Calculating the derivatives of $\hat{\eta}_i$, $\hat{w}_i$, and E

In order to find the derivatives of the effective degrees of freedom it is necessary to compute the derivatives of the linear predictor $\hat{\eta}_i = \mathbf{X}_i\hat{\tilde{\boldsymbol{\beta}}}$, weight matrix $\mathbf{W}$ (2.16), and the derivatives of the matrix $\mathbf{E}$ (2.17). The values of $\hat{\boldsymbol{\beta}}$, $\hat{\mathbf{w}}$, $\hat{\boldsymbol{\mu}}$, and $\hat{\boldsymbol{\eta}}$ are taken as fixed at their converged values from the full Newton based iterative scheme. Define the following constants:

$$a_{1i} = \frac{(y_i - \hat{\mu}_i)g''(\hat{\mu}_i)}{g'(\hat{\mu}_i)} \text{ and } a_{2i} = \frac{\hat{w}_i^2}{\omega_i}\left\{V'(\hat{\mu}_i)g'(\hat{\mu}_i) + 2V(\hat{\mu}_i)g''(\hat{\mu}_i)\right\}.$$

Then

$$\frac{\partial \hat{\eta}_i}{\partial \rho_k} = \mathbf{X}_i\mathbf{C}\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \rho_k},$$

$$\frac{\partial \hat{w}_i}{\partial \rho_k} = -a_{2i}\hat{\alpha}_i\frac{\partial \hat{\eta}_i}{\partial \rho_k} + \frac{\omega_i}{V(\hat{\mu}_i)g'^2(\hat{\mu}_i)}\frac{\partial \hat{\alpha}_i}{\partial \rho_k},$$

where

$$\frac{\partial \hat{\alpha}_i}{\partial \rho_k} = -\frac{1}{g'(\hat{\mu}_i)}\frac{\partial \hat{\eta}_i}{\partial \rho_k}\left[\frac{V'(\hat{\mu}_i)}{V(\hat{\mu}_i)} + \frac{g''(\hat{\mu}_i)}{g'(\hat{\mu}_i)} + (y_i - \hat{\mu}_i)\left\{\left(\frac{V'(\hat{\mu}_i)}{V(\hat{\mu}_i)}\right)^2 + \left(\frac{g''(\hat{\mu}_i)}{g'(\hat{\mu}_i)}\right)^2 - \frac{V''(\hat{\mu}_i)}{V(\hat{\mu}_i)} - \frac{g'''(\hat{\mu}_i)}{g'(\hat{\mu}_i)}\right\}\right],$$

and

$$\frac{\partial \hat{w}_{1i}}{\partial \rho_k} = -a_{2i}\frac{\partial \hat{\eta}_i}{\partial \rho_k}.$$

Now define a $q \times q$ diagonal matrix $\mathbf{N}_k$ with $N_{kj} = \partial\hat{\beta}_j/\partial\rho_k$, if β_j is exponentiated in $\tilde{\boldsymbol{\beta}}$, and 0, otherwise. Also define the diagonal matrices

$$\mathbf{T}_k = \text{diag}\left(\ldots, \frac{\partial \hat{w}_i}{\partial \rho_k}\frac{1}{|\hat{w}|_i}, \ldots\right), \text{ and } \mathbf{T}_{1k} = \text{diag}\left(\ldots, \frac{\partial \hat{w}_{1i}}{\partial \rho_k}\frac{1}{\hat{w}_{1i}}, \ldots\right) \tag{4.13}$$

the derivatives of $\hat{w}_i$ and $\hat{w}_{1i}$ will be given later.

Finally, let $\mathbf{A}_{1,} = \text{diag}(a_{1i})$, $i = 1, \ldots, n$, and let $\mathbf{C}_1$ be a $q \times q$ diagonal matrix with the elements

$$C_{1j} = \begin{cases} \exp(\hat{\beta}_j), & \text{if } \beta_j \text{ is exponentiated in } \tilde{\boldsymbol{\beta}} \\ 0, & \text{otherwise,} \end{cases}$$

then the derivative of the diagonal elements of the matrix $\mathbf{E}$ is

$$\begin{aligned} \frac{\partial \text{diag}(\mathbf{E})}{\partial \rho_k} =& \mathbf{N}_k(\mathbf{X}\mathbf{C}_1)^T\mathbf{W}_1\mathbf{G}(\mathbf{y} - \hat{\boldsymbol{\mu}}) + (\mathbf{X}\mathbf{C}_1)^T\mathbf{W}_1\mathbf{T}_{1k}\mathbf{G}(\mathbf{y} - \hat{\boldsymbol{\mu}}) + \\ & (\mathbf{X}\mathbf{C}_1)^T\mathbf{W}_1\mathbf{A}_1\frac{\partial \hat{\boldsymbol{\eta}}}{\partial \rho_k} - (\mathbf{X}\mathbf{C}_1)^T\mathbf{W}_1\frac{\partial \hat{\boldsymbol{\eta}}}{\partial \rho_k}. \end{aligned} \tag{4.14}$$

4.3.4 Calculating the first order derivative of τ

The final step is to find the derivative of the effective degrees of freedom, τ. From Section 2.3 $\tau = \text{tr}(\mathbf{F})$, where

$$\mathbf{F} = \mathbb{G}^{-1}(\mathbf{X}\mathbf{C})^T\mathbf{W}_1\mathbf{X}\mathbf{C},$$

$\mathbb{G} = (\mathbf{X}\mathbf{C})^T\mathbf{W}\mathbf{X}\mathbf{C} - \mathbf{E} + \mathbf{S}$ (see (4.8)).

Using (4.13), the following derivatives can be found

$$\frac{\partial \mathbf{W}}{\partial \rho_k} = \mathbf{W}\mathbf{T}_k\mathbf{I}^+ \text{ and } \frac{\partial \mathbf{W}_1}{\partial \rho_k} = \mathbf{W}\mathbf{L}\mathbf{T}_{1k},$$

where the matrices $\mathbf{L}$ and $\mathbf{I}^+$ have been defined in Section 2.4.

Let $\mathbf{E}'$ denote the first derivative of the diagonal matrix $\mathbf{E}$ with respect to ρ_k. Noting that $\partial\mathbf{C}/\partial\rho_k = \mathbf{C}\mathbf{N}_k$ it follows that

$$\begin{aligned} \frac{\partial \mathbb{G}^{-1}}{\partial \rho} = -\mathbb{G}^{-1}\big\{& \mathbf{N}_k(\mathbf{X}\mathbf{C})^T\mathbf{W}\mathbf{X}\mathbf{C} + (\mathbf{X}\mathbf{C})^T\mathbf{W}\mathbf{T}_k\mathbf{I}^+\mathbf{X}\mathbf{C} \\ & + (\mathbf{X}\mathbf{C})^T\mathbf{W}\mathbf{X}\mathbf{C}\mathbf{N}_k + e^{\rho_k}\mathbf{S}_k - \mathbf{E}'\big\}\,\mathbb{G}^{-1}. \end{aligned}$$

Therefore,

$$\frac{\partial \mathbf{F}}{\partial \rho_k} = \frac{\partial \mathbb{G}^{-1}}{\partial \rho_k}(\mathbf{XC})^T\mathbf{WLXC} + \mathbb{G}^{-1}\mathbf{N}_k\mathbf{XC}^T\mathbf{WLXC} + \mathbb{G}^{-1}(\mathbf{XC})^T\mathbf{WLT}_k\mathbf{I}^+\mathbf{XC} + \mathbb{G}^{-1}(\mathbf{XC})^T\mathbf{WLXCN}_k.$$

It should be pointed out that

$$\mathbb{G}^{-1} = \mathbf{PP}^T, \;\; \mathbb{G}^{-1}(\mathbf{XC})^T\sqrt{|\mathbf{W}|} = \mathbf{PK}^T, \;\; \text{and}$$

$$\sqrt{|\mathbf{W}|}(\mathbf{XC})\mathbb{G}^{-1}(\mathbf{XC})^T\sqrt{|\mathbf{W}|} = \mathbf{KK}^T.$$

Then it follows that

$$\begin{aligned}\frac{\partial \mathrm{tr}(\mathbf{F})}{\partial \rho_k} &= -\mathrm{tr}(\mathbf{KP}^T\mathbf{N}_k\mathbf{R}^T\mathbf{Q}_1^T\mathbf{I}^+\mathbf{KK}^T\mathbf{LI}^+) - \mathrm{tr}(\mathbf{KK}^T\mathbf{T}_k\mathbf{KK}^T\mathbf{I}^+\mathbf{L}) \\ &\quad - \mathrm{tr}(\mathbf{KK}^T\mathbf{I}^+\mathbf{Q}_1\mathbf{RN}_k\mathbf{PK}^T\mathbf{I}^+\mathbf{L}) - e^{\rho_k}\mathrm{tr}(\mathbf{KP}^T\mathbf{S}_k\mathbf{PK}^T\mathbf{I}^+\mathbf{L}) + \mathrm{tr}(\mathbf{KP}^T\mathbf{E}'\mathbf{PK}^T\mathbf{I}^+\mathbf{L}) \\ &\quad + \mathrm{tr}(\mathbf{KP}^T\mathbf{N}_k\mathbf{R}^T\mathbf{Q}_1^T\mathbf{I}^+\mathbf{L}) + \mathrm{tr}(\mathbf{KK}^T\mathbf{I}^+\mathbf{LT}_1) + \mathrm{tr}(\mathbf{N}_k\mathbf{PK}^T\mathbf{LI}^+\mathbf{Q}_1\mathbf{R}).\end{aligned} \tag{4.15}$$

Given those derivatives the GCV/UBRE criterions can be minimized by a quasi-Newton algorithm such as the BFGS method (Dennis and Schnabel, 1996; Nocedal and Wright, 2006).

4.4 Simulations

In this section examples are considered, where only some terms are constrained to monotonicity or to convexity and monotonicity together.

Example 3.1: A gamma model with log link and a linear predictor consisting of the sum of three smooth functions, where the second function is assumed to be monotone increasing, was fitted

$$\log(\mu_i) = \eta_i = f_1(x_{1i}) + f_2(x_{2i}) + f_3(x_{3i}), \;\; i = 1, \ldots, n, \tag{4.16}$$

where $\mu_i = \mathrm{E}(Y_i)$, $Y_i \sim \text{Gamma}\left\{\nu = 1, \;\; \theta = \exp(\eta_i)\right\}$.

One hundred values for each of three covariates, X_1, X_2, and X_3, were simulated independently from Unif$(-3, 3)$ for the first and the third covariates and from Unif$(-1, 3)$ for X_2. The algebraic expressions of the functions in the linear predictor were taken from Leitenstorfer and Tutz (2007):

$$f_1(x) = 1.5\sin(1.5x),$$

$$f_2(x) = 1.5/\left[1 + \exp\{-10(x + 0.75)\}\right] + 1.5/\left[1 + \exp\{-5(x - 0.75)\}\right],$$

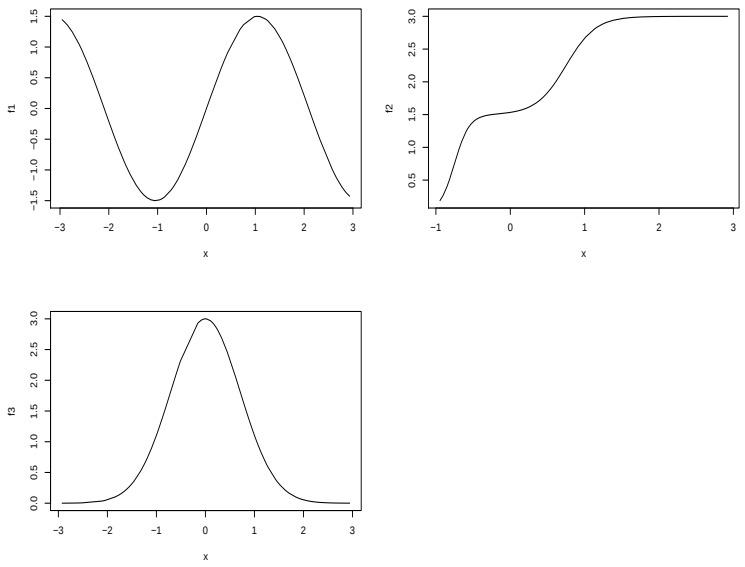


Figure 4-1: The test functions, $f_k(\cdot)$, $k = 1, 2, 3$, used in the simulation study of Example 3.1.

and

$$f_3(x) = 3\exp(-x^2).$$

Figure 4-1 shows the graphs of these functions.

Cubic P-splines of basis dimension $q_1 = q_3 = 15$ were used to fit the first and the third unconstrained terms, with penalties based on the second-order differences of the model coefficients. $f_2(x)$ was represented using a rank $q_2 = 30$ monotone cubic P-spline. The model was fitted by the proposed penalized likelihood maximization with the value of the multiple smoothing parameter $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3)^T$ found by the GCV minimization method given in Section 4.3. The estimated values of λ_k, $k = 1, 2, 3$, were 11.227, 0.010, and 3.359 respectively, with a minimal GCV score of 1.3425. The estimated effective degrees of freedom for each term were $\tau_1 = 4.70$, $\tau_2 = 3.03$, and $\tau_3 = 5.88$. The simulation results are illustrated in Figure 4-2. The panel (a) shows the actual versus fitted values of y. The rest of the panels, (b), (c), and (d), illustrate the estimates of the three smooth functions, on the 'centered' linear predictor scale (dashed curves) together with the true functions (solid curves) and the corresponding partial residuals shown as points. The partial residuals are obtained by adding Pearson residuals to the smooth terms (Wood, 2006a)

$$\hat{\epsilon}_{ki}^{partial} = f_k(x_{ki}) + \hat{\epsilon}_i^p, \;\; i = 1, \ldots, n,$$

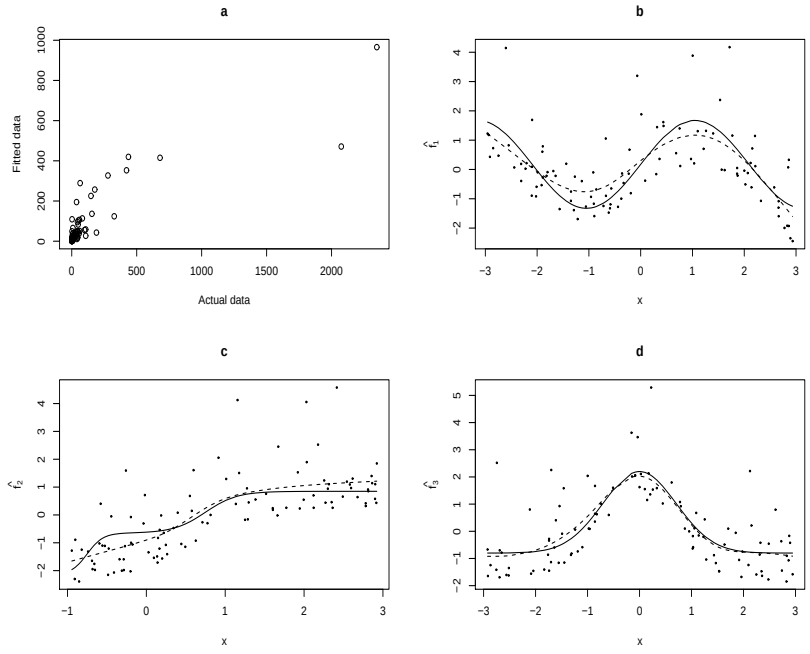


Figure 4-2: The simulation results for three term mono-GAM, Example 3.1.

where Pearson residuals are

$$\hat{\epsilon}_i^p = \frac{y_i - \hat{\mu}_i}{\sqrt{V(\hat{\mu}_i)}}.$$

Example 3.2: In this example the structure of the mono-GAM is the same as in the previous example (4.16). But now, $Y_i \sim \text{Pois}\left\{\exp(\eta_i)\right\}$, where the linear predictor η_i is the sum of the unconstrained, monotone, and monotone-convex smooth terms:

$$f_1(x) = 3\exp(-x^2),$$

$$f_2(x) = \exp(4x)/\left\{1 + \exp(4x)\right\},$$

and

$$f_3(x) = \exp(-3x).$$

Let $X_{1i} \sim \text{Unif}(-3, 3)$, $X_{2i} \sim \text{Unif}(-1, 3)$, and $X_{3i} \sim \text{Unif}(-1, 2)$, $i = 1, \ldots, n$, $n = 200$. The first smooth was fitted using $q_1 = 20$ cubic B-spline basis functions with second-order difference penalties. For the monotone term, $f_2(x)$, the monotone P-spline of rank $q_2 = 30$ was used, and the third smooth function was fitted using a mixed-constraint P-spline of rank $q_3 = 30$ described in Section 3.2.

The three optimal fitted curves on the linear predictor scale are represented on

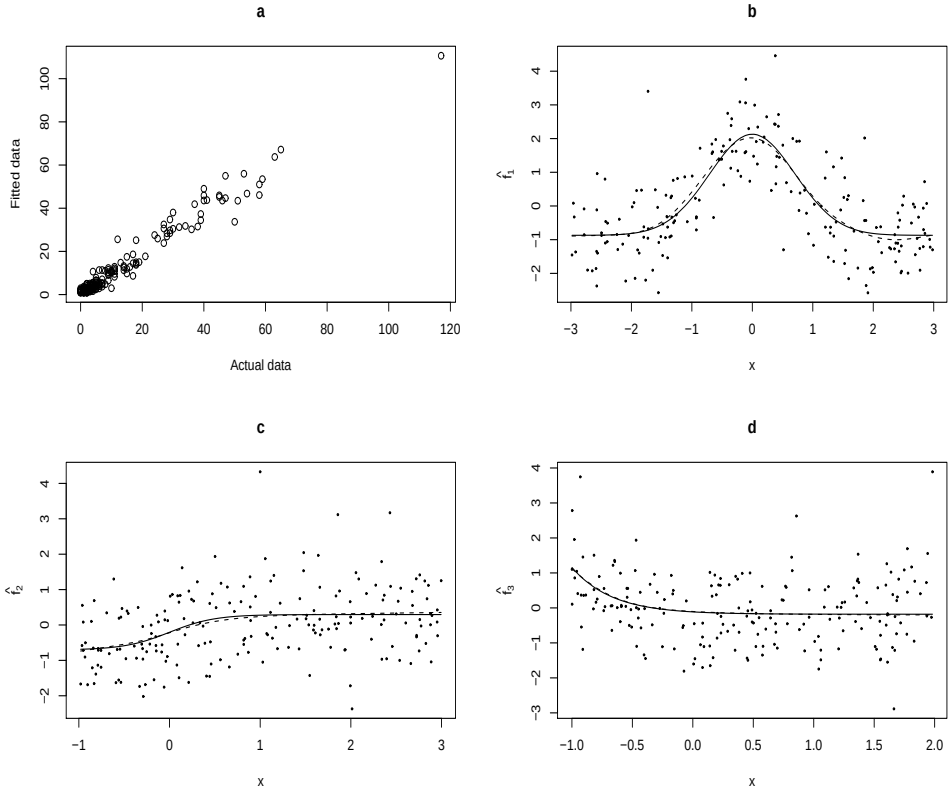


Figure 4-3: The simulation results for three term mono-GAM, Example 3.2.

panels (b), (c), and (d) of Figure 4-3 as dashed lines together with the graphs of the true functions illustrated as solid lines. The dots on those panels are the corresponding partial residuals. The first panel, (a), shows the actual against fitted values of the response variable.

The estimated degrees of freedom for each term were 7.64, 3.20, and 2.41. The values of the smoothing parameters selected by direct UBRE minimization were $\lambda_1 = 26.33$, $\lambda_2 = 7.21 \cdot 10^{-3}$, and $\lambda_3 = 2.20 \cdot 10^{-5}$ with the optimal $\mathcal{V}_u = 0.06995$.

For both examples the model estimation procedure includes two iterative procedures: a quasi-Newton method to update log smoothing parameters and a Newton based P-IRLS to obtain $\boldsymbol{\beta}$, given $\boldsymbol{\lambda}$, which is called for each step of the former optimization. The above simulated examples are given to illustrate the mono-GAM performance. A more extensive simulation study will be presented in Chapter 7.

Chapter 5

Confidence intervals for mono-GAM

The previous sections have dealt with the maximum penalized likelihood estimates of the model parameters $\boldsymbol{\beta}$ obtained by the full Newton method. A question about confidence intervals of those estimates arises, and also it is of interest to construct confidence intervals of the mono-GAM terms. The distribution of the working unconstrained parameters will be derived using the same approach as for the parameters of the unconstrained GAM (Wood, 2006a) with the smoothing parameters treated as fixed. Since there is a need for a simple, transparent calculation of the confidence intervals without simulations, the limiting distribution of the exponentiated model coefficients will be approximated by means of the delta method. The possibility of high negative correlation between the intercept and other parameters of the constrained smooth term will be dealt with by post-fit modification of the model centering constraints, in order to obtain the narrowest possible intervals for mono-GAM components. The performance of the proposed confidence intervals will be examined by estimating realized coverage probabilities of the intervals from simulation studies.

5.1 The delta method for deriving $\tilde{\boldsymbol{\beta}}$ distribution

Consider a general structure of the mono-GAM:

$$g(\mu_i) = \mathbf{X}_i^* \boldsymbol{\delta} + f_1(x_{1i}) + f_2(x_{2i}) + \ldots + f_p(x_{pi}), \;\; i = 1, \ldots, n, \tag{5.1}$$

where $Y_i \sim$ exponential family distribution, and some of the model smooth function, f_k, are shape constrained. As we saw in Chapter 4 the mono-GAM can be written as

a penalized GLM of the form

$$g(\mu_i) = \mathbf{X}_i\tilde{\boldsymbol{\beta}}, \tag{5.2}$$

where the components of the vector $\tilde{\boldsymbol{\beta}}$ are either $\exp(\beta_j)$ for most coefficients of the constrained smooth terms or β_j, otherwise. Given the values of the smoothing parameters, λ_k, the model parameters, $\boldsymbol{\beta}$, are estimated by maximizing the penalized log likelihood using the full Newton method.

Since the full Newton method and Fisher scoring result in the same estimate at their convergence, in order to get the distributional results on $\boldsymbol{\beta}$ we will deal with the expected values of the log likelihood Hessian as in case of the GAM (Wood, 2006a). That is the parameter estimates are considered to be of the form

$$\boldsymbol{\beta}^{[k+1]} = \boldsymbol{\beta}^{[k]} + \left\{(\mathbf{XC}^{[k]})^T\mathbf{W}^{[k]}\mathbf{XC}^{[k]} + \mathbf{S}\right\}^{-1}\left\{(\mathbf{XC}^{[k]})^T\mathbf{W}^{[k]}\mathbf{G}^{[k]}(\mathbf{y} - \boldsymbol{\mu}^{[k]}) - \mathbf{S}\boldsymbol{\beta}^{[k]}\right\}, \tag{5.3}$$

where $\mathbf{S} = \sum_{k=1}^{p}\lambda_k\mathbf{S}_k$, and the diagonal values of $\mathbf{W}$ are $w_i = 1/\left\{V(\mu_i)g'(\mu_i)^2\right\}$, $i = 1, \ldots, n$. Then it can be easily shown that the expression for $\boldsymbol{\beta}^{[k+1]}$ can be written as

$$\boldsymbol{\beta}^{[k+1]} = \left\{(\mathbf{XC}^{[k]})^T\mathbf{W}^{[k]}\mathbf{XC}^{[k]} + \mathbf{S}\right\}^{-1}(\mathbf{XC}^{[k]})^T\mathbf{W}^{[k]}\left\{\mathbf{G}^{[k]}(\mathbf{y} - \boldsymbol{\mu}^{[k]}) + \mathbf{XC}^{[k]}\boldsymbol{\beta}^{[k]}\right\}.$$

Define $\mathbf{z}^{[k]} = \mathbf{G}^{[k]}(\mathbf{y} - \boldsymbol{\mu}^{[k]}) + \mathbf{XC}^{[k]}\boldsymbol{\beta}^{[k]}$, which is referred to as a vector of pseudodata in the P-IRLS scheme of the GAM. Hence, the parameter estimators at convergence become

$$\hat{\boldsymbol{\beta}} = \left\{(\mathbf{XC})^T\mathbf{WXC} + \mathbf{S}\right\}^{-1}(\mathbf{XC})^T\mathbf{Wz}.$$

Since the variance of $z_i|\boldsymbol{\beta}$ is

$$\text{var}(z_i|\boldsymbol{\beta}) = g'(\mu_i)^2V(\mu_i)\phi = \frac{1}{w_i}\phi,$$

the corresponding covariance matrix of $\hat{\boldsymbol{\beta}}$ is

$$\mathbf{V}_{\hat{\boldsymbol{\beta}}} = \left\{(\mathbf{XC})^T\mathbf{WXC} + \mathbf{S}\right\}^{-1}(\mathbf{XC})^T\mathbf{WXC}\left\{(\mathbf{XC})^T\mathbf{WXC} + \mathbf{S}\right\}^{-1}\phi.$$

The confidence intervals constructed using $\mathbf{V}_{\hat{\boldsymbol{\beta}}}$ produce unsatisfactory realized coverage probabilities (Wood, 2006a). So a Bayesian approach will be used for obtaining the distribution (posterior) of the mono-GAM working coefficients. Bayesian interval estimates for smoothing spline models were proposed by Wahba (1983) and Silverman (1985). The extensions of their results to generalized additive models based on low rank penalized regression splines have been suggested by, for example, Lin and Zhang

(1999), Wood (2000), Wood and Augustin (2002), and Wood (2006b). In this project the Bayesian approach of Wood (2006b) has been applied. Section 5.1.1 will describe how it can be adopted for the mono-GAM. At the moment consider only the result of this procedure: the posterior parameter vector distribution can be written as

$$\boldsymbol{\beta}|\mathbf{v} \sim \mathrm{N}\left(\hat{\boldsymbol{\beta}}, \mathbf{V}_{\boldsymbol{\beta}}\right), \tag{5.4}$$

where a Bayesian posterior covariance matrix for the parameters is

$$\mathbf{V}_{\boldsymbol{\beta}} = \left\{(\mathbf{XC})^T\mathbf{WXC} + \mathbf{S}\right\}^{-1}\phi, \tag{5.5}$$

and $\mathbf{v} = (\mathbf{XC})^T\mathbf{Wz}$.

A simple and direct approach to approximating the distribution of $\tilde{\boldsymbol{\beta}}$ used the delta method. The delta method is a general method for establishing the asymptotic distribution of functions of a multinormally distributed random vector. In this project the delta method is used to construct a linear approximation of the exponential functions of parameters, $\tilde{\boldsymbol{\beta}}$, and derive the approximate distribution and covariance matrix for that simpler linear function.

Consider the Taylor series expansion of $\tilde{\boldsymbol{\beta}}$ as a vector of functions of $\boldsymbol{\beta}$

$$\begin{aligned}\tilde{\boldsymbol{\beta}} &= \hat{\tilde{\boldsymbol{\beta}}} + \mathrm{diag}\left(\nabla\hat{\tilde{\boldsymbol{\beta}}}\right)(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}) + \mathrm{diag}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})\mathrm{diag}\left(\nabla^2\hat{\tilde{\boldsymbol{\beta}}}^*\right)(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})/2 \\ &\approx \hat{\tilde{\boldsymbol{\beta}}} + \mathrm{diag}\left(\nabla\hat{\tilde{\boldsymbol{\beta}}}\right)(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}),\end{aligned} \tag{5.6}$$

where $\hat{\tilde{\boldsymbol{\beta}}}$ are $\tilde{\boldsymbol{\beta}}$ estimators, $\mathrm{diag}\left(\nabla\hat{\tilde{\boldsymbol{\beta}}}\right)$ is a diagonal matrix of size q, with the vector of the first order derivatives of $\tilde{\boldsymbol{\beta}}$ with respect to $\boldsymbol{\beta}$ evaluated at $\hat{\boldsymbol{\beta}}$ on the main diagonal. $\mathrm{diag}(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})$ is a diagonal matrix with $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$ on the main diagonal. $\nabla^2\hat{\tilde{\boldsymbol{\beta}}}^*$ is a vector of second order derivatives of $\tilde{\boldsymbol{\beta}}$ evaluated at $\hat{\boldsymbol{\beta}}^*$, some value between $\hat{\boldsymbol{\beta}}$ and $\boldsymbol{\beta}$. In the notation of Section 2.2, $\mathrm{diag}\left(\nabla\hat{\tilde{\boldsymbol{\beta}}}\right) = \mathbf{C}$. From (5.6) we have

$$\tilde{\boldsymbol{\beta}} - \hat{\tilde{\boldsymbol{\beta}}} \approx \mathrm{diag}\left(\nabla\hat{\tilde{\boldsymbol{\beta}}}\right)(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}),$$

which means that $\tilde{\boldsymbol{\beta}} - \hat{\tilde{\boldsymbol{\beta}}}$ is approximately a linear function of $\boldsymbol{\beta}$.

Finally, recalling that

$$\boldsymbol{\beta}|\mathbf{v} \sim \mathrm{N}\left(\hat{\boldsymbol{\beta}}, \mathbf{V}_{\boldsymbol{\beta}}\right),$$

and that linear functions of the normally distributed random variables follow normal

distributions, the approximate distribution of $\tilde{\boldsymbol{\beta}}$ (Rao, 1973; Davison, 2008) is

$$\tilde{\boldsymbol{\beta}} \sim \mathrm{N}\left(\hat{\tilde{\boldsymbol{\beta}}}, \mathbf{C}\mathbf{V}_{\boldsymbol{\beta}}\mathbf{C}\right),$$

where

$$\mathbf{V}_{\tilde{\boldsymbol{\beta}}} = \mathbf{C}\left\{(\mathbf{XC})^T\mathbf{W}\mathbf{XC} + \mathbf{S}\right\}^{-1}\mathbf{C}\phi.$$

5.1.1 Posterior distribution for the working parameters of a mono-GAM

In this section the posterior distribution for the mono-GAM working coefficients, $\boldsymbol{\beta}$, will be derived by using the approach of Wood (2006b). Define a random vector $\mathbf{v} = (\mathbf{XC})^T\mathbf{W}\mathbf{z}$. The covariance matrix of $\mathbf{v}|\boldsymbol{\beta}$ is $\mathrm{cov}(\mathbf{v}|\boldsymbol{\beta}) = (\mathbf{XC})^T\mathbf{W}(\mathbf{XC})\phi$. Applying the same arguments for examining the Lindeberg's conditions for the validity of the Central Limit Theorem (Lindeberg, 1922) as in the case of the unconstrained GAM, it can be shown that with sample size $n \to \infty$ the distribution of $\mathbf{v}|\boldsymbol{\beta}$ will tend to the multivariate normal with the following parameters

$$\mathbf{v}|\boldsymbol{\beta} \;\sim\; N\left((\mathbf{XC})^T\mathbf{W}\mathbf{XC}\boldsymbol{\beta},\; (\mathbf{XC})^T\mathbf{W}\mathbf{XC}\phi\right).$$

Consider the following prior for $\boldsymbol{\beta}$,

$$f_{\boldsymbol{\beta}}(\boldsymbol{\beta}) \;\propto\; e^{-\frac{1}{2}\boldsymbol{\beta}^T\sum\frac{\mathbf{S}_i}{\tau_i}\boldsymbol{\beta}},$$

where τ_i is a parameter controlling the dispersion of the prior. Noting from the above that

$$f(\mathbf{v}|\boldsymbol{\beta}) \;\propto\; e^{-\frac{1}{2}\left(\mathbf{v}-(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\boldsymbol{\beta}\right)^T\left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\right]^{-1}\left(\mathbf{v}-(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\boldsymbol{\beta}\right)/\phi},$$

Bayes rule gives

$$\begin{aligned} f(\boldsymbol{\beta}|\mathbf{v}) \;&\propto\; e^{-\frac{1}{2}\left\{\mathbf{v}^T\left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\right]^{-1}\mathbf{v}/\phi - 2\mathbf{v}^T\left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\right]^{-1}(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\boldsymbol{\beta}/\phi\right\}}\times \\ &\quad\times e^{-\frac{1}{2}\left\{\boldsymbol{\beta}^T(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\right]^{-1}(\mathbf{XC})^T\mathbf{W}\mathbf{XC}\boldsymbol{\beta}/\phi + \boldsymbol{\beta}^T\sum\frac{\mathbf{S}_i}{\tau_i}\boldsymbol{\beta}\right\}} \\ &\propto\; e^{-\frac{1}{2}\left\{-2\mathbf{v}^T\boldsymbol{\beta}/\phi + \boldsymbol{\beta}^T\left(\frac{(\mathbf{XC})^T\mathbf{W}\mathbf{XC}}{\phi} + \sum\frac{\mathbf{S}_i}{\tau_i}\right)\boldsymbol{\beta}\right\}}. \end{aligned} \tag{5.7}$$

One may note that for

$$\boldsymbol{\alpha} \;\sim\; N\left(\left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC} + \sum\lambda_i\mathbf{S}_i\right]^{-1}\mathbf{v},\; \left[(\mathbf{XC})^T\mathbf{W}\mathbf{XC} + \sum\lambda_i\mathbf{S}_i\right]^{-1}\phi\right),$$

$$
\begin{aligned}
f_{\boldsymbol{\alpha}}(\boldsymbol{\alpha}) \;&\propto \\
&\propto e^{-\frac{1}{2}\left\{\left(\boldsymbol{\alpha}-\left[(\mathbf{XC})^T\mathbf{WXC}+\sum\lambda_i\mathbf{S}_i\right]^{-1}\mathbf{v}\right)^T\left[(\mathbf{XC})^T\mathbf{WXC}+\sum\lambda_i\mathbf{S}_i\right]\left(\boldsymbol{\alpha}-\left[(\mathbf{XC})^T\mathbf{WXC}+\sum\lambda_i\mathbf{S}_i\right]^{-1}\mathbf{v}\right)/\phi\right\}} \\
&\propto e^{-\frac{1}{2}\left\{\boldsymbol{\alpha}^T\left[(\mathbf{XC})^T\mathbf{WXC}+\sum\lambda_i\mathbf{S}_i\right]\boldsymbol{\alpha}/\phi-2\mathbf{v}^T\boldsymbol{\alpha}/\phi\right\}} = \; e^{-\frac{1}{2}\left\{-2\mathbf{v}^T\boldsymbol{\alpha}/\phi+\boldsymbol{\alpha}^T\left[\frac{(\mathbf{XC})^T\mathbf{WXC}}{\phi}+\frac{\sum\lambda_i\mathbf{S}_i}{\phi}\right]\boldsymbol{\alpha}\right\}}.
\end{aligned}
\tag{5.8}
$$

By setting $\tau_i = \frac{\phi}{\lambda_i}$ and examining (5.7) and (5.8) we recognize the posterior parameter distribution

$$\boldsymbol{\beta}|\mathbf{v} \sim \mathrm{N}\left(\hat{\boldsymbol{\beta}}, \mathbf{V}_{\boldsymbol{\beta}}\right),$$

with

$$\mathbf{V}_{\boldsymbol{\beta}} = \left\{(\mathbf{XC})^T\mathbf{WXC}+\mathbf{S}\right\}^{-1}\phi.$$

Due to the nature of the identifiability constraints imposed on the monotone increasing (decreasing) terms in the additive models (see Section 4.1.1) an issue about the possible high negative (positive) correlations between the intercept and exponentiated parameters of the monotonic smooth terms arises. To deal with this issue it is proposed to apply the centering identifiability constraint in place of setting the first parameter of the monotone term to zero ('zeroed intercept' constraint) but to do so after fitting. Such re-parametrization reduces the correlation with the intercept, but it can not be used for the monotone smooths before fitting since it would destroy the monotonicity construction in that case.

5.2 Imposing centering constraint

This section explains how the centering constraint may be imposed on each shape constrained smooth term in order to overcome the problem with the high correlation between the parameters of these smooth terms and the intercept.

Univariate smooth term with shape constraints

Consider the univariate shape constrained function $f(x_i)$, that can be represented before an identifiability constraint was imposed as

$$f(x_i) = \mathbf{X}_i'\boldsymbol{\Sigma}'\tilde{\boldsymbol{\beta}}',$$

where $\tilde{\boldsymbol{\beta}}' = (\beta_1, \exp(\beta_2), \ldots, \exp(\beta_q))^T$. For simplicity of notation, the index denoting the order of the smooth term in the mono-GAM settings (5.1) and the covariate index

have been omitted.

As an identifiability constraint for the shape constrained smooth it was proposed to set $\beta_1 = 0$ (see Section 4.1.1), so by denoting $\tilde{\boldsymbol{\beta}} = (0, \exp(\beta_2), \ldots, \exp(\beta_q))^T$ we have

$$f(x_i) = \mathbf{X}_i' \boldsymbol{\Sigma}' \tilde{\boldsymbol{\beta}}. \tag{5.9}$$

After the fitting procedure a centering constraint $\sum_{i=1}^{n} f_a(x_i) = 0$ may be imposed by re-parametrization of $\tilde{\boldsymbol{\beta}}$, with $f_a(x_i)$ differing from $f(x_i)$ only by a constant. Such a centering constraint may be written as $\mathbf{A}\tilde{\boldsymbol{\beta}}' = 0$, $\mathbf{A} = \mathbf{1}^T \mathbf{X}' \boldsymbol{\Sigma}'$, where $\mathbf{1}$ is an n vector of ones. Now we should re-parameterize the smooth in terms of new parameters $\tilde{\boldsymbol{\beta}}_a$ such that

$$\tilde{\boldsymbol{\beta}}' = \mathbf{Z}_a \tilde{\boldsymbol{\beta}}_a.$$

A general way of doing that is to find the matrix $\mathbf{Z}_a$ which satisfies

$$\mathbf{1}^T \mathbf{X}' \boldsymbol{\Sigma}' \mathbf{Z}_a = \mathbf{0}, \tag{5.10}$$

and has $q-1$ orthogonal columns (Wood, 2006a). For this purpose, the QR decomposition of $\mathbf{A}^T$ can be used. Suppose

$$\mathbf{A}^T = \mathbf{Q}_a \begin{bmatrix} \mathbf{P} \\ \mathbf{0} \end{bmatrix},$$

then $\mathbf{Q}_a$ may be partitioned as $\mathbf{Q}_a \equiv (\mathbf{D} : \mathbf{Z}_a)$, where $\mathbf{D}$ consists only of the first column and $\mathbf{Z}_a$ is of the dimension $q \times (q-1)$.

Now, since two fits obtained by using those two identifiability constraints ('zeroed intercept' and the centering constraint) will differ only by the constant, say c, we may write

$$\mathbf{X}' \boldsymbol{\Sigma}' \mathbf{Z}_a \tilde{\boldsymbol{\beta}}_a = \mathbf{X}' \boldsymbol{\Sigma}' \tilde{\boldsymbol{\beta}} + \mathbf{c}, \tag{5.11}$$

where $\mathbf{c}$ is an n vector of the constant c. In order to find c one may sum up all n equations of (5.11), which results in

$$\sum_{i=1}^{n} f_a(x_i) = \sum_{i=1}^{n} f(x_i) + nc,$$

from which it follows that

$$c = -\frac{1}{n} \sum_{i=1}^{n} f(x_i),$$

and since the centering constraint is imposed after the fitting procedure the value of c

can be easily found.

The next step is to find $\tilde{\boldsymbol{\beta}}_a$. Applying the QR decomposition, $\mathbf{X}'\boldsymbol{\Sigma}' = \mathbf{QR}$, and multiplying both sides of the expression (5.11) by $\mathbf{Q}^T$ from the left, we get

$$\mathbf{RZ}_a\tilde{\boldsymbol{\beta}}_a = \mathbf{R}\tilde{\boldsymbol{\beta}} + \mathbf{Q}^T\mathbf{c}. \tag{5.12}$$

From the property of the B-splines (see Section 2.1) $\sum_{j=1}^{q} B_j^m(x_i) = 1$, and since the first column of matrix $\boldsymbol{\Sigma}'$ is a column of ones for all shape constraints (see Sections 2.1, 3.1, 3.2), the first column of the model matrix $\mathbf{X}'\boldsymbol{\Sigma}'$ consists only of ones. Due to this property and the fact that the matrix $\mathbf{R}$ in the QR decomposition is upper triangular with the first element, say, R_{11}, the elements of the first column of the matrix $\mathbf{Q}$ are equal to $Q_{i1} = \frac{1}{R_{11}}$, $i = 1, \ldots, n$. Hence, the first column of $\mathbf{Q}$ is constant, but $\mathbf{Q}$ is an orthogonal matrix, therefore,

$$\mathbf{1}^T\mathbf{Q}_{\cdot j} = 0, \tag{5.13}$$

for any $j > 1$, where $\mathbf{Q}_{\cdot j}$ denotes the j^{th} column of $\mathbf{Q}$. Moreover, by constructing the first Householder matrix $\mathbf{H}_1$, when forming $\mathbf{Q}$, it it easy to see that the elements of the first column of $\mathbf{Q}$ are

$$Q_{i1} = 1/\sqrt{n},$$

where n is the number of observations.

From (5.13) it follows that

1. $\mathbf{Q}^T\mathbf{c} = (c\sqrt{n}, 0, 0, \ldots, 0)^T$;

2. the first row of $\mathbf{RZ}_a$ is a row of $0'$s. This is because from (5.10)

$$\mathbf{1}^T\mathbf{QRZ}_a = \mathbf{0},$$

while $\mathbf{1}^T\mathbf{Q} = (\sqrt{n}, 0, 0, \ldots, 0)$ and to make the right-hand side of the above expression $\mathbf{0}$, the first row of $\mathbf{RZ}_a$ must consist only of $0'$s.

Taking these features into account from (5.12) we get

$$\tilde{\boldsymbol{\beta}}_a = (\mathbf{R}^*\mathbf{Z}_a)^{-1}\mathbf{R}^*\tilde{\boldsymbol{\beta}}, \tag{5.14}$$

where $\mathbf{R}^*$ is the matrix $\mathbf{R}$ without its first row.

It should be noted that using R functions for QR decomposition one may reduce the computational cost by not forming $\mathbf{Z}_a$ explicitly, since in fact $\mathbf{Z}_a$ is only pre-multiplied by $\mathbf{R}$ and post-multiplied by $\tilde{\boldsymbol{\beta}}_a$.

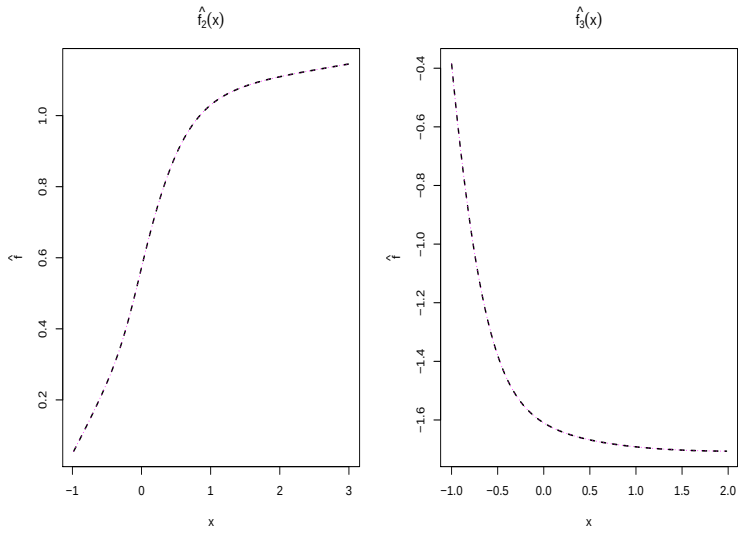


Figure 5-1: Illustration of the equivalency of the centering identifiability constraint with the monotonic identifiability constraint, Example 3.2. There are two coincided curves on each plot: curves with the 'zeroed intercept' (black dashed lines) and with the centering constraint (pink dotted lines). The curves with the centering constraint have a constant subtracted.

Finally, the function $f(x_i)$ may be written as

$$f(x_i) = \mathbf{X}_i' \boldsymbol{\Sigma}' \mathbf{Z}_a (\mathbf{R}^* \mathbf{Z}_a)^{-1} \mathbf{R}^* \tilde{\boldsymbol{\beta}} - c, \tag{5.15}$$

and correspondingly, the covariance matrix of this smooth is

$$\mathbf{V}_f = \left\{ \mathbf{X}' \boldsymbol{\Sigma}' \mathbf{Z}_a (\mathbf{R}^* \mathbf{Z}_a)^{-1} \mathbf{R}^* \right\} \mathbf{V}_{\tilde{\boldsymbol{\beta}}} \left\{ \mathbf{X}' \boldsymbol{\Sigma}' \mathbf{Z}_a (\mathbf{R}^* \mathbf{Z}_a)^{-1} \mathbf{R}^* \right\}^T,$$

where $\mathbf{V}_{\tilde{\boldsymbol{\beta}}}$ was derived in the previous section by using the delta method, and to handle $\beta_1 = 0$ it should be augmented with an initial row and column of zeros.

Figure 5-1 shows the equivalency of the two types of identifiability constraints, the centering constraint and the 'zeroed intercept' constraint. Two fits for the second and third smooth terms of Example 3.2 were obtained by using the 'zeroed intercept' (5.9) and centering constraint (5.15). The coincidence of the dashed lines of the fits with the initial identifiability constraint and the dotted lines of the fits with the centering constraint supports the approach of this section.

The 95% component-wise Bayesian confidence intervals for the mono-GAM of Example 3.2 are illustrated in Figure 5-2, while Figure 5-3 shows an uncorrected version

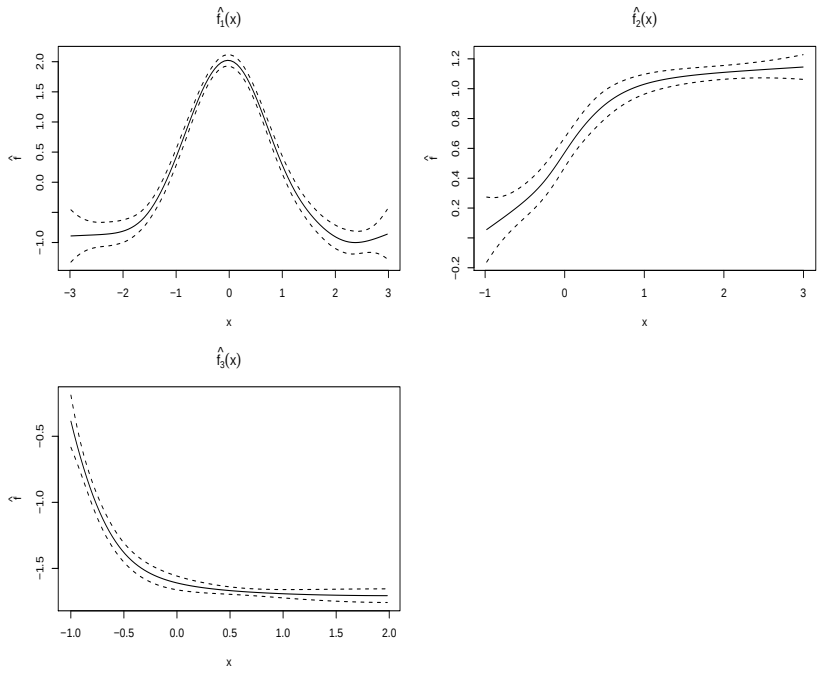


Figure 5-2: Illustration of the corrected Bayesian confidence intervals, Example 3.2.

of the confidence intervals without imposing centering constraints on the monotone smooth terms. The function estimates are given by the solid curves, the dashed curves are boundaries of the 95% confidence regions for each function. For the first unconstrained smooth the confidence interval is equivalent to the confidence interval obtained by the Bayesian approach of Wood (2006b), since in this case the part of the diagonal matrix, $\mathbf{C}$, which corresponds to the first smooth becomes an identity matrix, and hence the covariance matrix (5.5) is the same as for the unconstrained GAM. The slight narrowing in the confidence intervals on the plateau regions of $\hat{f}_2(x)$ and $\hat{f}_3(x)$ is due to the insignificant increment in the model parameters when the corresponding β_j tend to minus infinity and their variances are near zero.

Bivariate smooth term with monotonicity restriction

For the bivariate function with double monotonicity $f(x_{1i}, x_{2i})$ the same approach as above may be used, since in this case the identifiability constraint is analogous, $\beta_{11} = 0$ (Section 4.1.3). But a different constraint is applied to the bivariate function with single monotonicity, which is $\sum_{j=1}^{q_2} \beta_{1j} = 0$ for single monotonicity along the first covariate and $\sum_{i=1}^{q_1} \beta_{i1} = 0$ for monotonicity only along the second covariate (q_1 and q_2 are the numbers

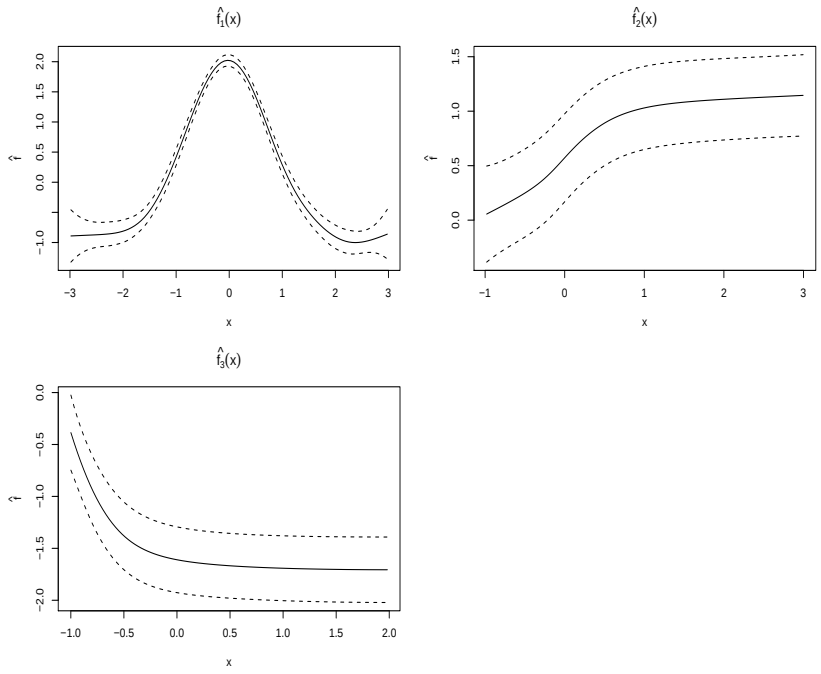


Figure 5-3: Illustration of the uncorrected Bayesian confidence intervals before imposing centering constraints, Example 3.2.

of basis functions of the marginal smooths).

Using the tensor product construction described in Section 3.3.2 the bivariate function with single monotonicity may be written as $f(x_{1i}, x_{2i}) = \mathbf{X}_i'\boldsymbol{\Sigma}'\boldsymbol{\beta}'$. Let these identifiability constraints be denoted as $\boldsymbol{\beta}' = \mathbf{Z}_c\tilde{\boldsymbol{\beta}}_c$, where $\tilde{\boldsymbol{\beta}}_c$ is a $(q_1 \cdot q_2 - 1)$ vector of unconstrained parameters, $\mathbf{Z}_c$ was introduced in Section 4.1.3. Following the previous subsection we have

$$\mathbf{R}\mathbf{Z}_a\tilde{\boldsymbol{\beta}}_a = \mathbf{R}\mathbf{Z}_c\tilde{\boldsymbol{\beta}}_c + \mathbf{Q}^T\mathbf{c}.$$

Unfortunately, in this case $\mathbf{1}^T\mathbf{Q}_{\cdot j} \neq 0$, for $j > 1$, therefore

$$\tilde{\boldsymbol{\beta}}_a = \left\{(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_a\right\}^{-1}(\mathbf{R}\mathbf{Z}_a)^T\left(\mathbf{R}\mathbf{Z}_c\tilde{\boldsymbol{\beta}}_c + c\mathbf{Q}^T\mathbf{1}\right). \tag{5.16}$$

From (5.16) it follows

$$f(x_{1i}, x_{2i}) = \mathbf{X}_i'\boldsymbol{\Sigma}'\mathbf{Z}_a\left\{(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_a\right\}^{-1}(\mathbf{R}\mathbf{Z}_a)^T\left(\mathbf{R}\mathbf{Z}_c\tilde{\boldsymbol{\beta}}_c + c\mathbf{Q}^T\mathbf{1}\right) - c.$$

Fortunately, for constructing the confidence intervals we only need the covariance ma-

trix of $f(x_{1i}, x_{2i})$, which has almost the same representation as in the univariate case

$$\mathbf{V}_f = \left[\mathbf{X}'\boldsymbol{\Sigma}'\mathbf{Z}_a \left\{(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_a\right\}^{-1}(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_c\right] \mathbf{V}_{\tilde{\boldsymbol{\beta}}} \left[\mathbf{X}'\boldsymbol{\Sigma}'\mathbf{Z}_a \left\{(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_a\right\}^{-1}(\mathbf{R}\mathbf{Z}_a)^T\mathbf{R}\mathbf{Z}_c\right]^T.$$

5.3 Simulation from the posterior distribution

It should be mentioned that there is another alternative for constructing the Bayesian confidence intervals for the monotonic smooth terms of the mono-GAM, which is based on simulation from the posterior distribution (5.4). After obtaining the simulated values of the model coefficients the values of the smooth terms can be evaluated and the quantiles of the approximate posterior cumulative distribution functions of the smooths will be used for the confidence intervals construction. While this sounds reasonable, in reality the non-linear dependence on the parameters results in large values of the posterior covariance matrix of $\tilde{\boldsymbol{\beta}}$ and implausible confidence intervals for the monotone smooth terms. Figure 5-4 illustrates the problem with the Bayesian confidence intervals constructed by this approach.

The Bayesian simulation confidence intervals for Example 3.2 were constructed here. Solid curves are the estimates of the functions, the dashed curves are boundaries of the 95% confidence regions for each function. Huge problems are visible in the case of the shaped constrained smooths $f_2(x)$ and $f_3(x)$. The large asymmetry and widening from the left to the right of these confidence intervals may be explained by the log normality of the exponentiated parameters and their summation (for monotone increasing smooth) or subtraction (for monotone decreasing and convex smooth) in the shape constrained P-spline settings. The variance of, for example, the first parameter of the second smooth, β_{20}, in this settings is 1.0278, which is increased further for the exponentiated value. Moreover, there is a strong negative correlation between the first parameter, β_1, and all except one parameters of $f_2(x)$, which ranges from -0.1243 till $-4.72e-4$, as well as possible high positive correlation between β_1 and parameters of the third smooth.

So the delta method approach benefits from not only avoiding simulations but also in preventing the occurrence of the issues described above. The proposed Bayesian approach for confidence intervals construction makes a couple of assumptions. It uses the linear approximation of the exponentiated parameters, and in the case of non-Gaussian models adopts large sample inference. Also the smoothing parameters are treated as fixed. The simulation study presented in the next section will examine

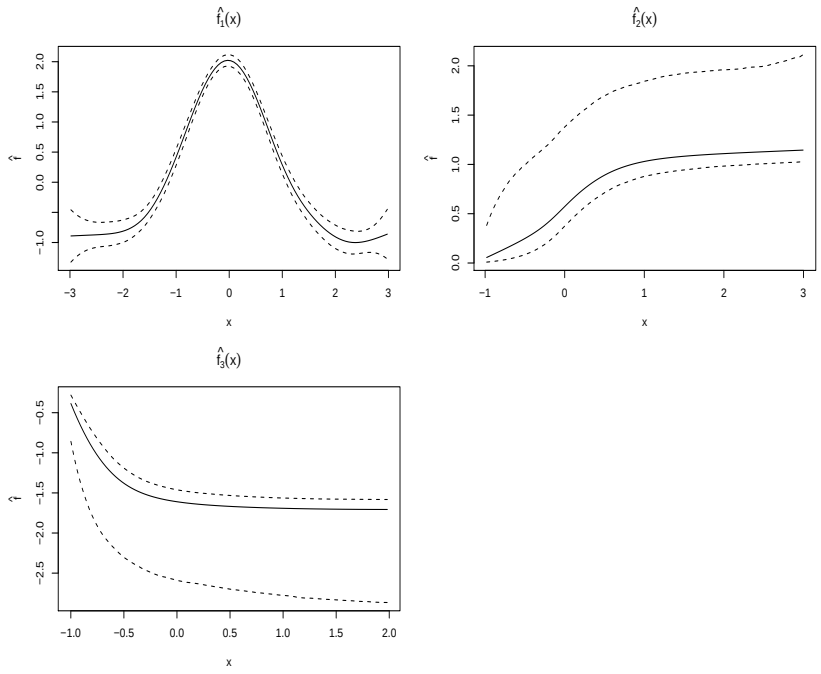


Figure 5-4: Illustration of the problems with the Bayesian confidence intervals for three term mono-GAM, Example 3.2.

how these restrictions affect the performance of the confidence intervals. The realized coverage probabilities will be taken as a measure of their performance.

5.4 Coverage probabilities

5.4.1 Single smooth term models

The simulation study of confidence interval performance is conducted in an analogous manner to Wood (2006b).

The simulation study starts with the single term models. In the next section the effectiveness of the confidence intervals for mono-GAMs is investigated. Two univariate (monotone increasing and monotone decreasing) functions and one bivariate function, with a single monotone decreasing restriction along the first covariate, are examined:

$$\begin{aligned} f_1(x) =& d\left[\exp(4x)/\left\{1+\exp(4x)\right\}+2\right], \\ f_2(x) =& d(x-3)^6, \quad -1 \leq x \leq 2, \\ f_3(x_1,x_2) =& -d\left[\exp(4x_1)/\left\{1+\exp(4x_1)\right\}+2\sin(\pi x_2)\right], \end{aligned}$$

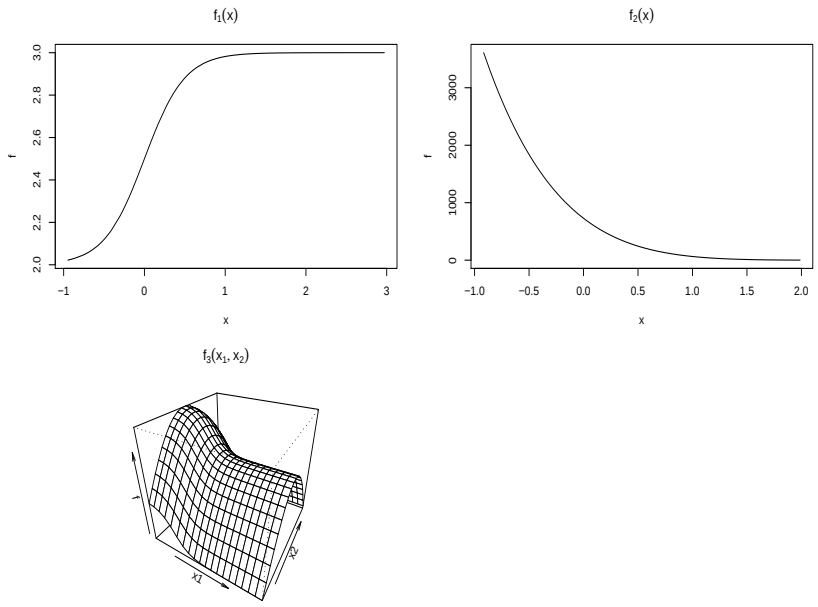


Figure 5-5: Functions used for the simulation study of Section 5.4.1.

where d is a constant which controls the noise level in, for example, Poisson error models ($d = 1$ for the Gaussian case). Figure 5-5 shows the shapes of these functions.

The covariates were simulated from a uniform distribution: Unif$(-1, 3)$ for the covariate of the first univariate function and Unif$(-1, 2)$ for the second one; for the bivariate function $f_3(x_1, x_2)$ the covariate x_1 was drawn from Unif$(-1, 3)$ and x_2 from Unif$(0, 1)-$distribution. Gaussian and Poisson models with three noise levels each were considered at two sample sizes, $n = 200$ and $n = 500$ in the univariate cases, and $n = 400$ for the bivariate function. The canonical link functions were applied for each of the models. For the Gaussian case the functions were rescaled to $[0, 1]$, and three different values of the standard deviation were used ($\sigma = 0.05, 0.1$, and 0.2). In order to know the signal to noise ratio, R^2 between the simulated data and the truth may be calculated, where

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - f_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{5.17}$$

y_i are simulated data and $\bar{y}$ is their mean value. The R^2 values for the first function are 0.98, 0.93, and 0.76 for three different values of $\sigma = 0.05, 0.1$, and 0.2, respectively. For $f_2(x)$ the values are 0.96, 0.87, and 0.63. The signal to noise ratios of the third bivariate model are 0.96, 0.86, and 0.60.

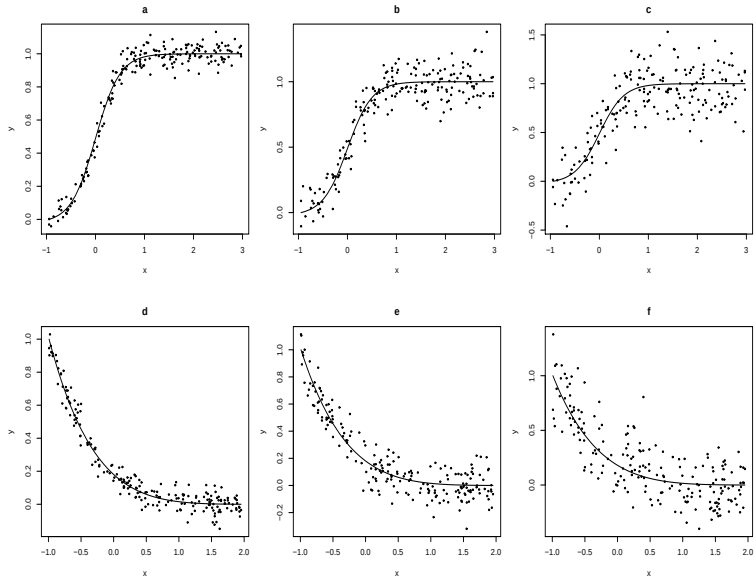


Figure 5-6: Illustration of the simulated data for f_1 and f_2 with each of the three noise levels of the Gaussian distribution. (a) f_1, $\sigma = 0.05$. (b) f_1, $\sigma = 0.10$. (c) f_1, $\sigma = 0.20$. (d) f_2, $\sigma = 0.05$. (e) f_2, $\sigma = 0.10$. (f) f_2, $\sigma = 0.20$. Solid lines show the true functions.

Three different values of d were taken for each of the three Poisson models, these values were $d = 0.5, 1, 1.5$ for f_1 with the signal to noise ratio 0.09, 0.60, and 0.91 respectively; $d = 0.03, 0.05, 0.08$ for f_2 with R^2 of 0.36, 0.80, and 0.98; and $d = 1, 2, 2.5$ for the third function where R^2 were 0.49, 0.92, 0.99. Illustrations of the single replicate data sets for the first two function are shown in Figure 5-6 in the Gaussian case and in Figure 5-7 in the Poisson.

Fifteen basis functions of the cubic monotone P-splines were used to model the univariate data sets. For the last function the bivariate P-spline with single monotonicity was employed with the marginal basis dimensions $q_1 = q_2 = 10$. The models were fitted by penalized likelihood maximization with smoothing parameter selected by GCV in the Gaussian case and by UBRE for the Poisson models. The illustration of the 95% confidence intervals for a typical replicate for the first two functions at each of three noise levels is shown in Figure 5-8 for the Gaussian case and in Figure 5-9 for the Poisson model. The contour plots of the confidence intervals for the the bivariate models are given in Figures 5-10 to 5-12 for the Gaussian models.

$N = 500$ replicate data sets were simulated for each model, and three confidence levels were considered, 90%, 95%, and 99%. The realized coverage proportions were calculated for the values of the functions at each of the covariate values ('across-the-

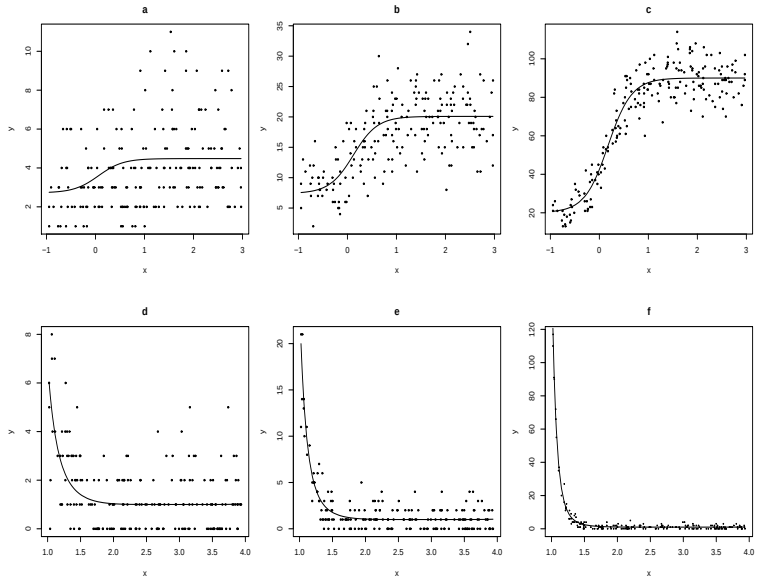


Figure 5-7: Illustration of the simulated data for f_1 and f_2 with each of the three noise levels of the Poisson distribution. (a) f_1, $d = 0.5$. (b) f_1, $d = 1$. (c) f_1, $d = 1.5$. (d) f_2, $d = 0.03$. (e) f_2, $d = 0.05$. (f) f_2, $d = 0.08$. Solid lines show the true functions.

function' coverage proportion) for each of the three confidence levels, and the mean coverage probability and its standard error were obtained over 500 simulated data sets. The realized coverage probabilities for the univariate functions are presented in Figure 5-13 for the normal case and in Figure 5-14 for the Poisson models.

For the normal case the realized coverage probabilities are near the corresponding nominal values, and the larger sample size improves the standard errors as expected. The results for the Poisson models are quite good with the exception of the first Poisson model with the low signal strength, which may be explained by the fact that the optimal fit inclines toward a straight line model (Marra and Wood). The opposite situation occurs with the second decreasing model, the high signal to noise ratio produces the poor coverage. The reason for this lies in the shape of the true function which is much steeper and not smooth for the high signal strength (see Figure 5-7), so the current smoothing method for a non-smooth truth can produce poor fits, and the coverage proportion is much less than the nominal value.

Figure 5-15 illustrates the realized coverage probabilities for the bivariate function $f_3(x_1, x_2)$ with the single monotonicity along the first covariate. The results for the Gaussian and Poisson error models shown in this figure are quite good, the realized coverage probabilities are comparative with the actual values.

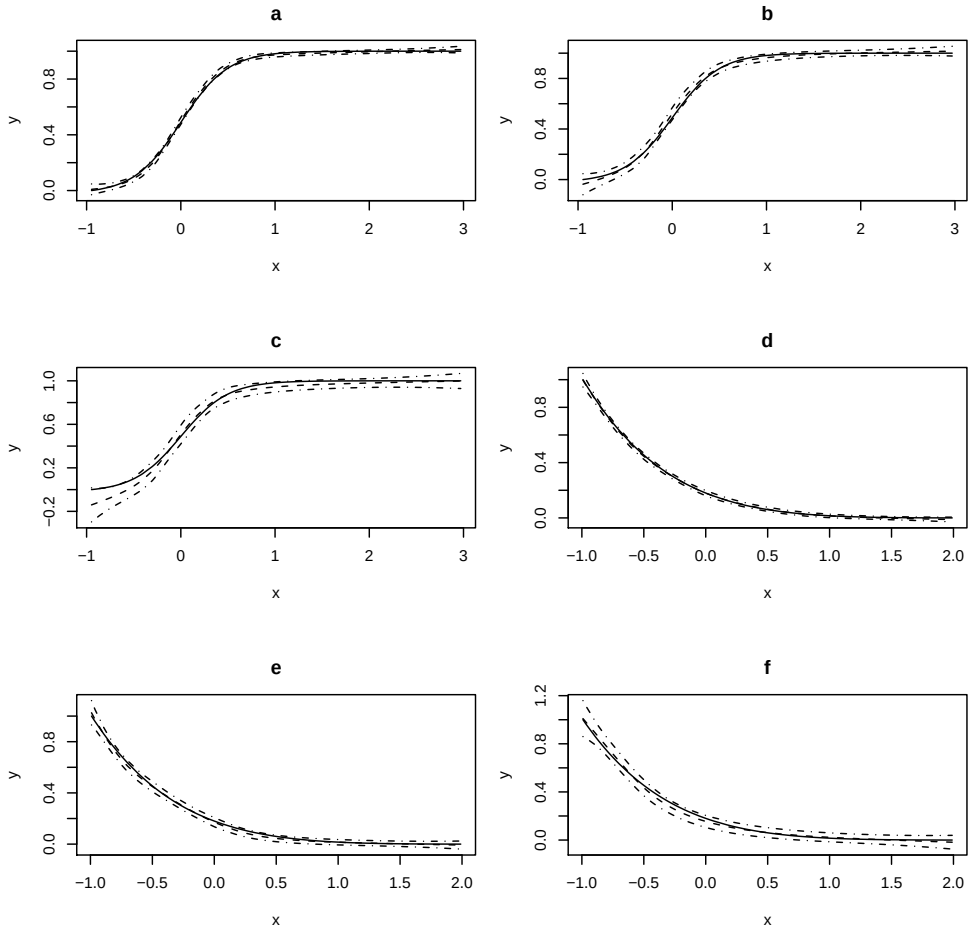


Figure 5-8: Illustration of the 95% confidence intervals for a typical replicate of f_1 and f_2 with each of the three noise levels of the Gaussian distribution. a) f_1, $\sigma = 0.05$. (b) f_1, $\sigma = 0.10$. (c) f_1, $\sigma = 0.20$. (d) f_2, $\sigma = 0.05$. (e) f_2, $\sigma = 0.10$. (f) f_2, $\sigma = 0.20$. The sample size was 200. Solid lines show the true functions, dashed lines represent the fitted curves, and dot-dashed lines show the bounds of the 95% confidence regions.

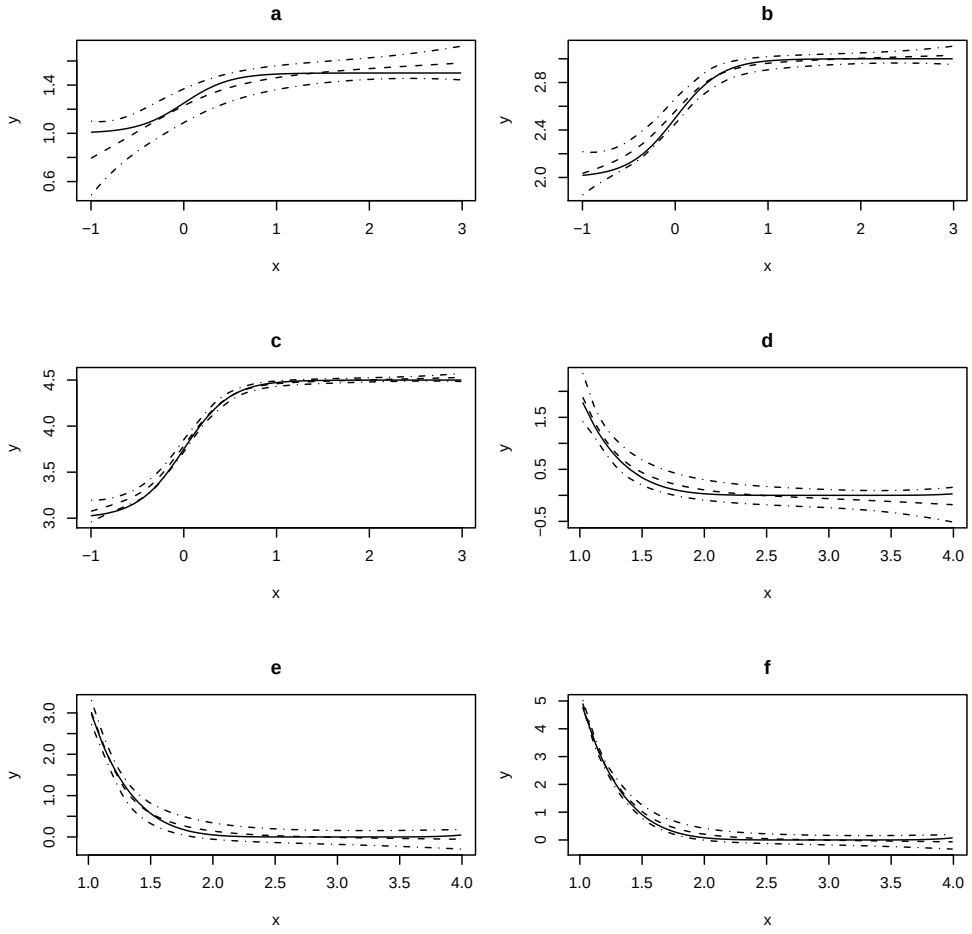


Figure 5-9: Illustration of the 95% confidence intervals for a typical replicate of f_1 and f_2 with each of the three noise levels of the Poisson distribution. (a) f_1, $d = 0.5$. (b) f_1, $d = 1$. (c) f_1, $d = 1.5$. (d) f_2, $d = 0.03$. (e) f_2, $d = 0.05$. (f) f_2, $d = 0.08$. The sample size was 200. Solid lines show the true functions, dashed lines represent the fitted curves, and dot-dashed lines show the bounds of the 95% confidence regions.

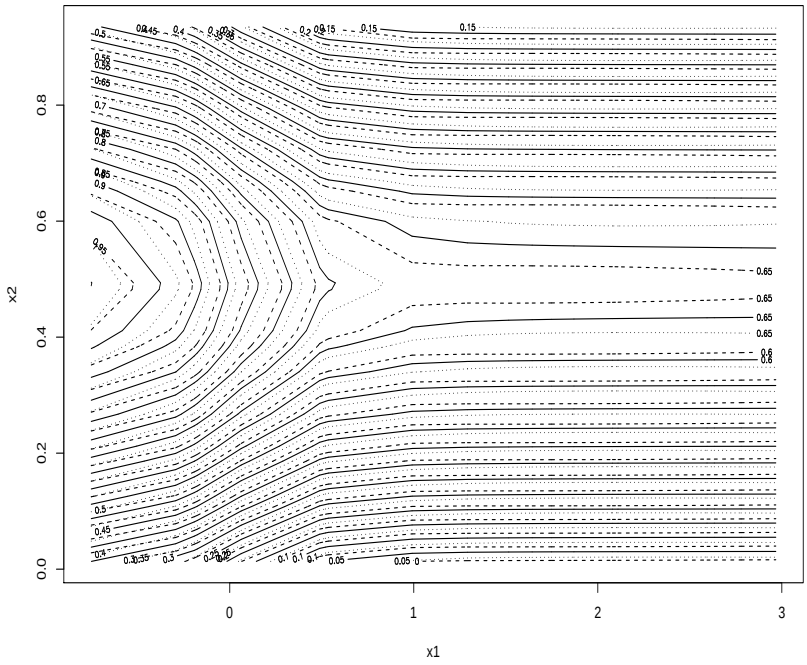


Figure 5-10: Illustration of the 95% confidence intervals for a typical replicate of $f_3(x_1, x_2)$, Gaussian distribution with $\sigma = 0.05$. The sample size was 400. Solid lines show the contour plot of the fitted curve. Dashed lines show the contour plot of the lower bounds of 95% confidence region, the dotted lines - upper bounds of the 95% confidence region.

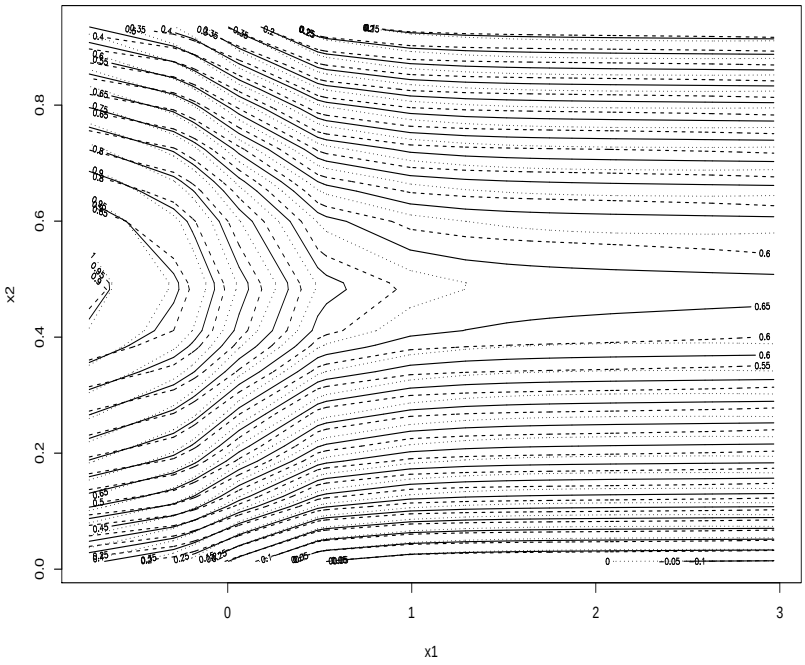


Figure 5-11: Illustration of the 95% confidence intervals for a typical replicate of $f_3(x_1, x_2)$, Gaussian distribution with $\sigma = 0.10$. The sample size was 400. Solid lines show the contour plot of the fitted curve. Dashed lines show the contour plot of the lower bounds of 95% confidence region, the dotted lines - upper bounds of the 95% confidence region.

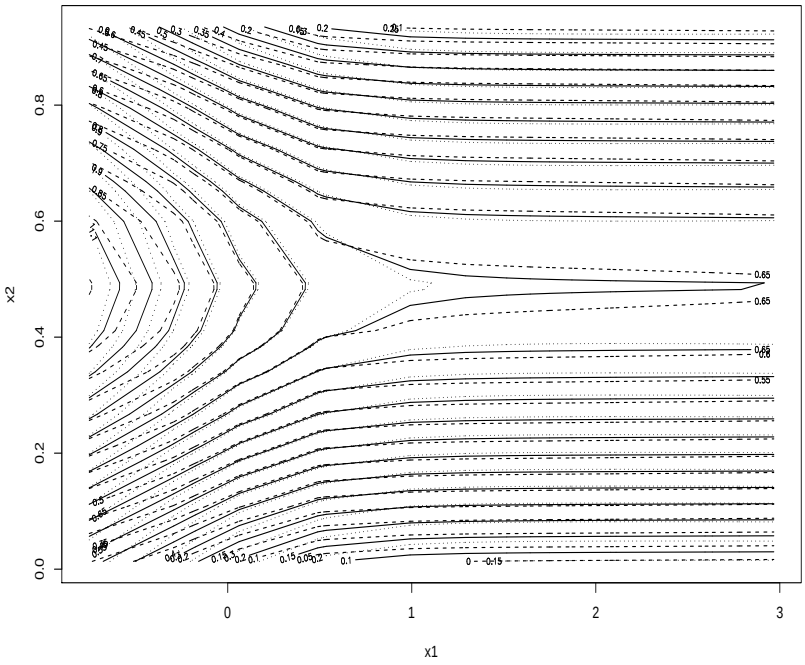


Figure 5-12: Illustration of the 95% confidence intervals for a typical replicate of $f_3(x_1, x_2)$, Gaussian distribution with $\sigma = 0.20$. The sample size was 400. Solid lines show the contour plot of the fitted curve. Dashed lines show the contour plot of the lower bounds of 95% confidence region, the dotted lines - upper bounds of the 95% confidence region.

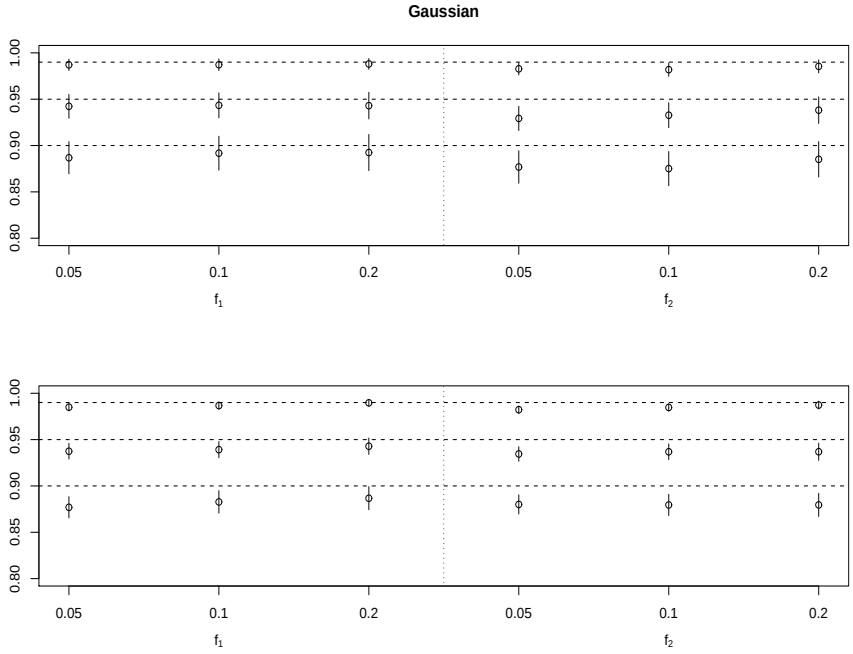


Figure 5-13: Realized coverage probabilities for confidence intervals from two single smooth term model simulation studies, for normal data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each function. The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

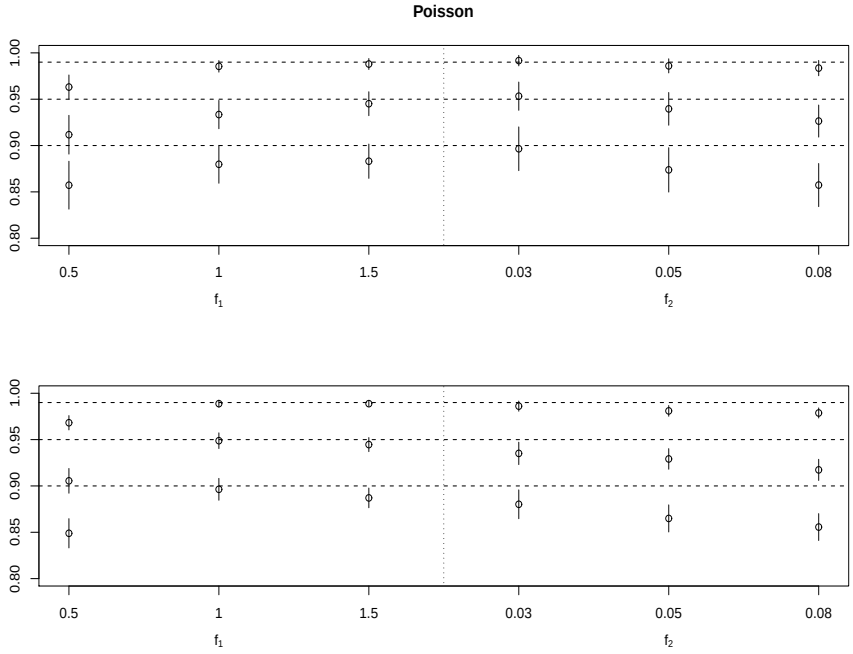


Figure 5-14: Realized coverage probabilities for confidence intervals from two single smooth term model simulation studies, for Poisson data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each function. The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

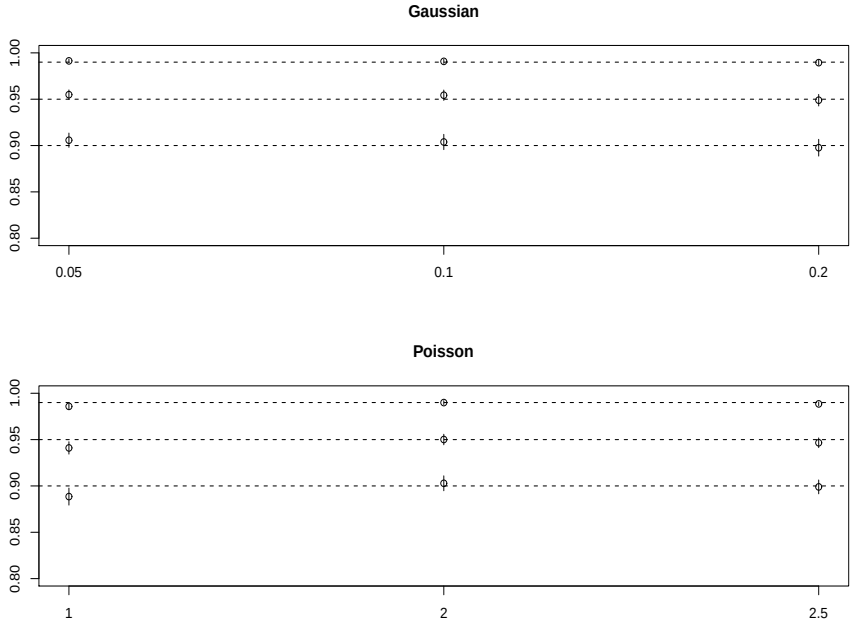


Figure 5-15: Realized coverage probabilities for confidence intervals from the bivariate smooth term model simulation studies, for Gaussian data (top panel) and Poisson data (bottom panel) for $n = 400$. Three noise levels are used for each function. The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

5.4.2 Mono-GAMs

In this section the performance of the confidence intervals is investigated in two mono-GAM settings in which the monotone and unconstrained smooth terms are added up to build the linear predictor. In the first simulation study unconstrained, monotone increasing, and monotone decreasing smooth terms make up the linear predictor:

$$g(\mu_i) = f_1(x_{1i}) + f_2(x_{2i}) + f_3(x_{3i}), \;\; i = 1, \ldots, n,$$

where

$$f_1(x_1) = 3\exp(-x_1^2),$$

$$f_2(x_2) = \exp(4x_2)/\left\{1 + \exp(4x_2)\right\} + 2,$$

$$f_3(x_3) = \exp(-x_3/4),$$

for simulation. The graphs of these functions are illustrated in Figure 5-16. n values for each of three covariates were independently simulated from the uniform distributions on $[-3, 3]$ for x_1, $[-1, 3]$ for x_2, and on $[-5, 5]$ for the third covariate x_3. Each function was rescaled to $[0, 1]$ before simulation.

In the second set of the simulation study a monotone increasing and two unconstrained smooth functions with the following algebraic expressions were used for simulation:

$$f_1(x_1) = 1.5/\left[1 + \exp\left\{-10(x_1 + 0.75)\right\}\right] + 1.5/\left[1 + \exp\left\{-5(x_1 - 0.75)\right\}\right],$$

$$f_2(x_2) = 1.5\sin(1.5x_2),$$

$$f_3(x_3) = 2\sin(\pi x_3).$$

Figure 5-17 shows the shapes of the functions.

The n values of the first two covariates were drawn independently from Unif$(-3, 3)$, while the third one from Unif$(0, 1)$. As in the previous section the Gaussian and Poisson models with the identity and log link functions, respectively, were considered, at each of three noise levels and two sample sizes (n = 200, 500). Five hundred replicates for each generalized additive model with monotonicity constraints were fitted using the method described in Chapter 4. The unconstrained smooth terms were represented by penalized cubic regression splines while for the monotone smooths the monotone P-splines introduced in Sections 2.1 and 3.1 were used. To fit the models penalized likelihood maximization with the smoothing parameter selection by GCV (Gaussian models) or UBRE (Poisson models) was used. The confidence intervals were obtained

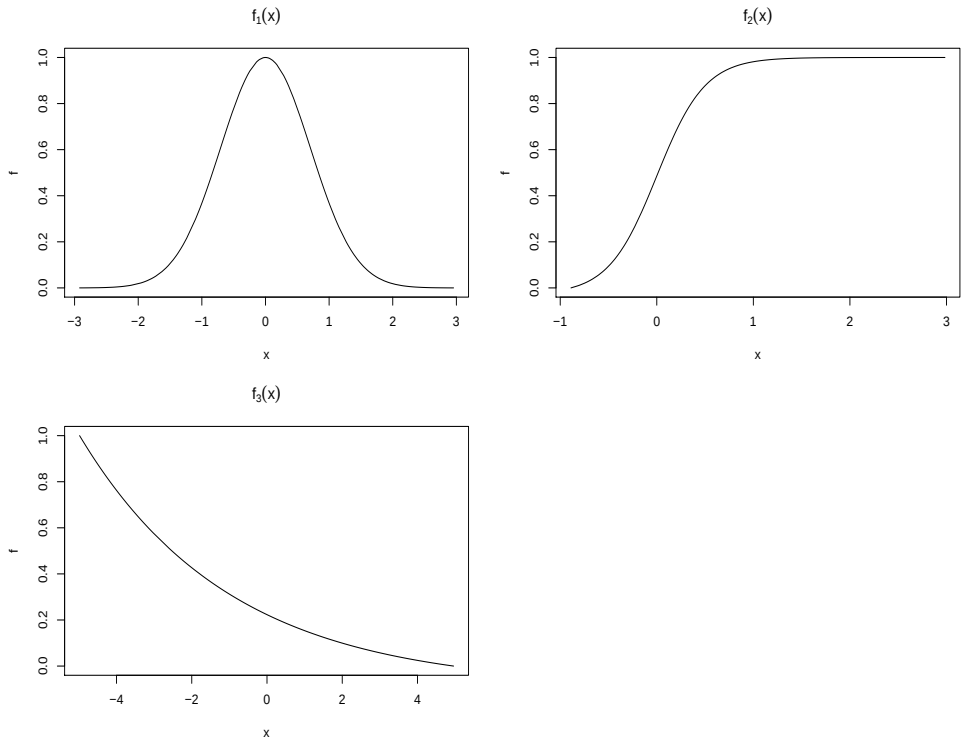


Figure 5-16: Shapes of the smooth terms used in the first simulation study of the mono-GAM.

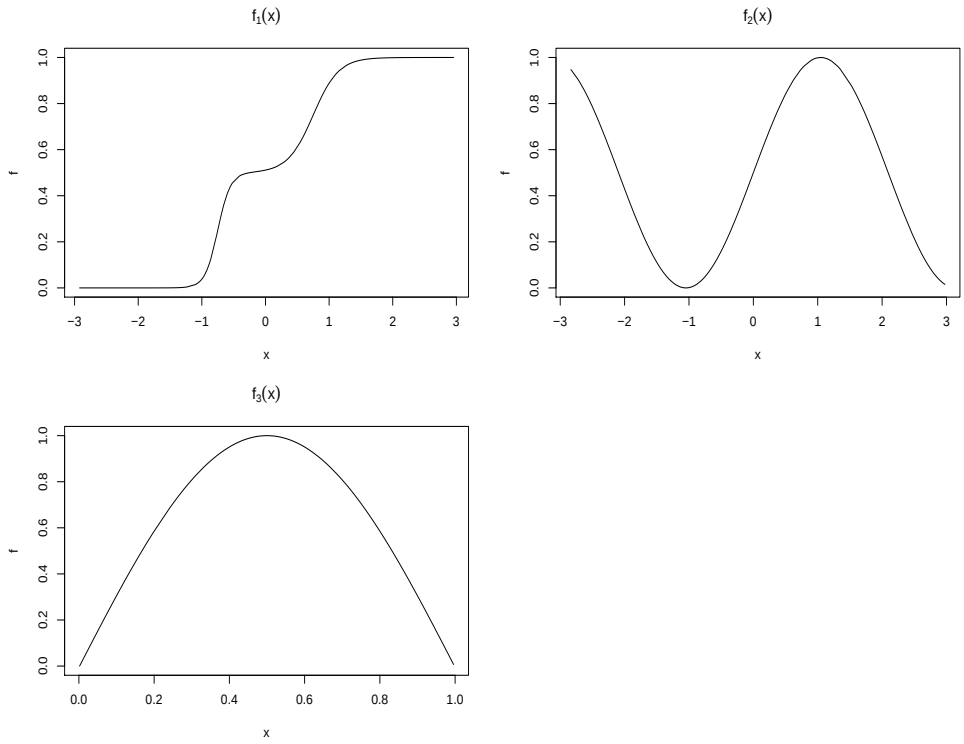


Figure 5-17: Shapes of the smooth terms used in the second simulation study of the mono-GAM.

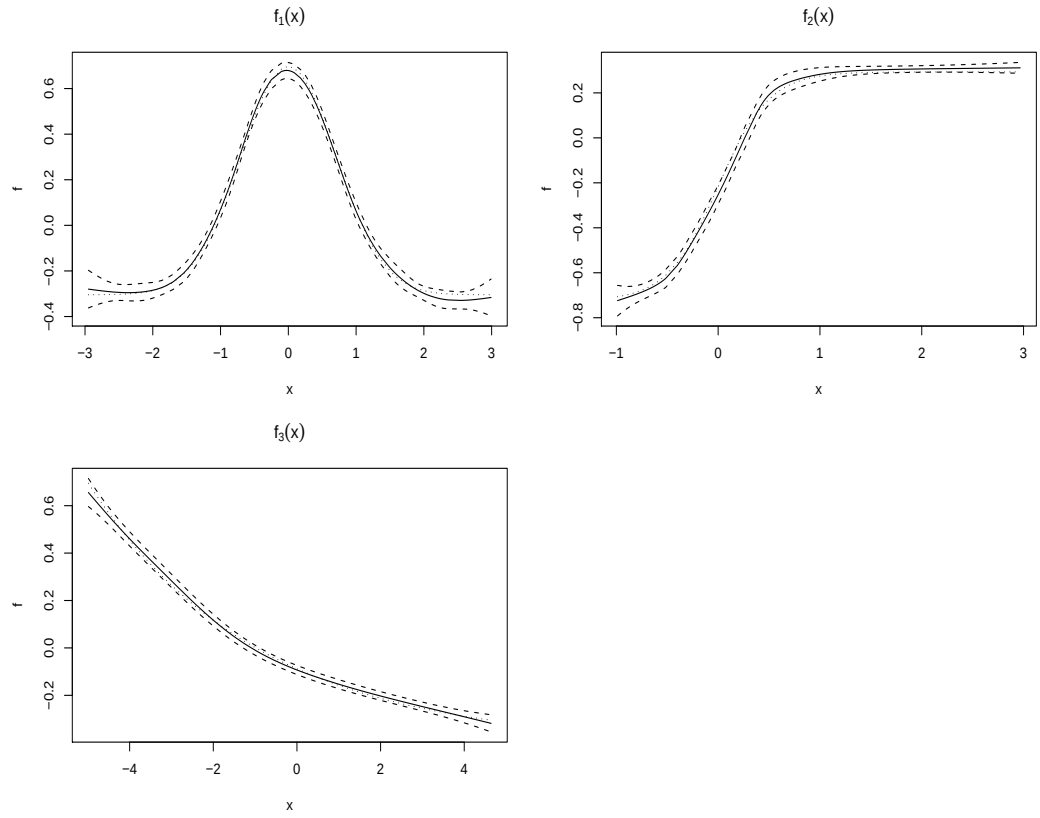


Figure 5-18: Illustration of the 95% confidence intervals for each of three component functions of a typical replicate of the Gaussian model of the first mono-GAM example. $\sigma = 0.1$, $n = 200$. Solid lines show the fitted curves, dotted curves represent the true functions, dashed lines show the bounds of the 95% confidence regions.

for the overall linear predictor and for each component smooth term. The confidence intervals for each component function separately for a typical replicate of the Gaussian model are shown in Figure 5-18 for the first simulation study and in Figure 5-19 for the second one. The labels of the vertical axes show the covariate of the smooth term along with the estimated degrees of freedom of that term.

For each replicate of each simulation study the 90%, 95%, and 99% confidence intervals were obtained, the 'across-the-function' coverage proportion was calculated for each case, and then averaged across all replicates. The results from the first mono-GAM are shown in Figure 5-20 for the Gaussian case and in Figure 5-21 for the Poisson model. From these figures one may note that the coverage probabilities are close to the nominal, with an exception for the third component, $f_3(x_3)$, of the Poisson case. The reason for this poor coverage is probably the same as the reason given for the poor

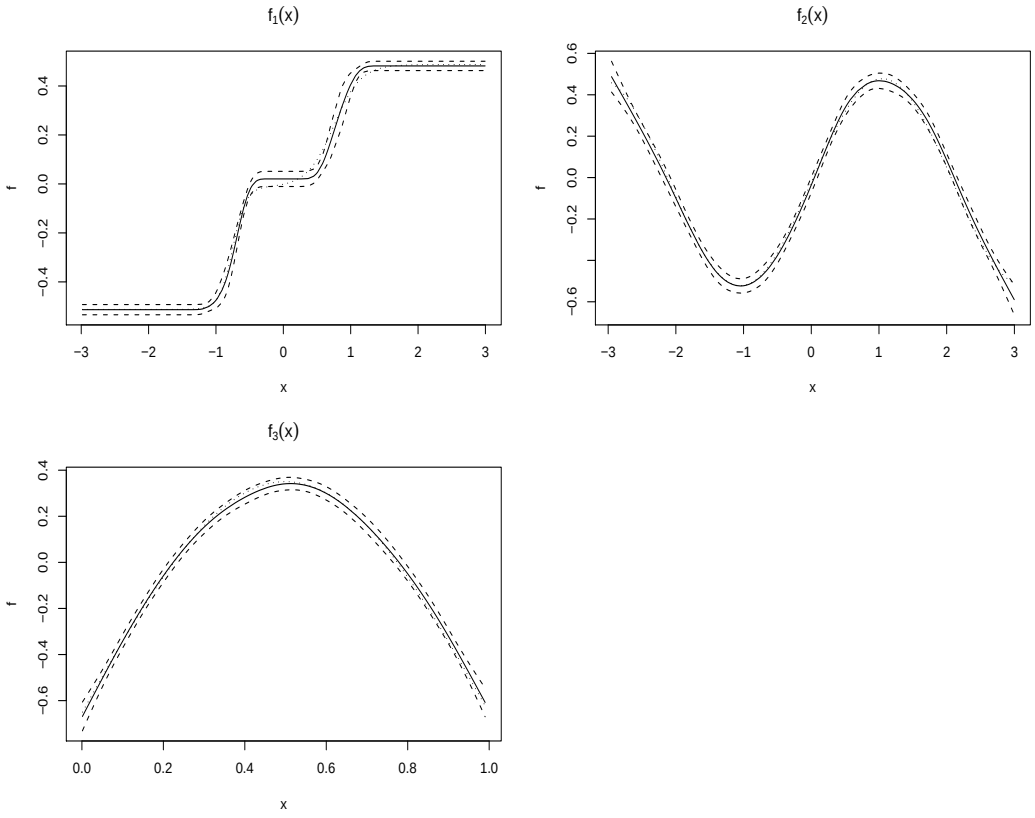


Figure 5-19: Illustration of the 95% component-wise confidence intervals for a typical replicate of the Gaussian model of the second mono-GAM example. $\sigma = 0.1$, $n = 200$. Solid lines show the fitted curves, dotted curves represent the true functions, dashed lines show the bounds of the 95% confidence regions.

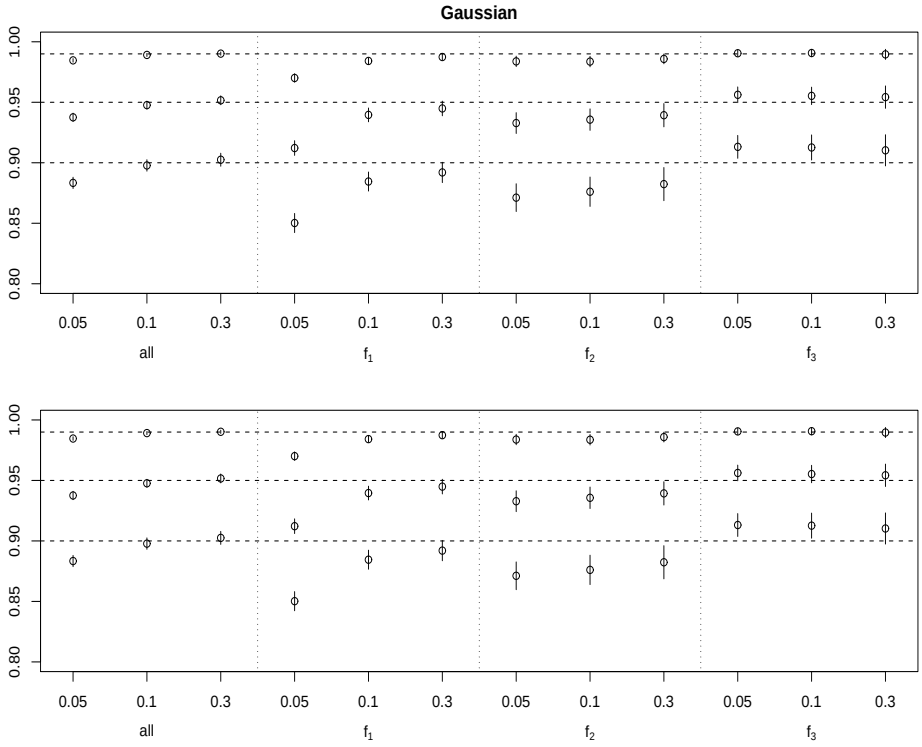


Figure 5-20: Realized coverage probabilities for confidence intervals from the mono-GAM simulation study of the first example, for normal data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each smooth term and for the overall model ("all"). The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

coverage in the single monotone decreasing case of the previous section. It seems that at the low signal-to-noise ratio a straight line model is tending to be chosen by the UBRE criterion. The same features can be observed in Figures 5-22 and 5-23 which illustrate the realized coverage probabilities of the second example. The only departure from the above mentioned quality is noticeable on the coverage proportion of the monotone increasing smooth $f_1(x_1)$ of the Gaussian model when $n = 200$. The worse coverage for the greater noise level of this situation may be accounted for by the 'two-step' shape of the function which is difficult to capture accurately from relatively few, noisy, data. The coverage probabilities are better for the larger sample size, as expected.

The simulation studies show that the confidence intervals behave much better than might be expected, although there may be some extreme cases that produce over-smoothed models and correspondingly poor coverage probabilities.

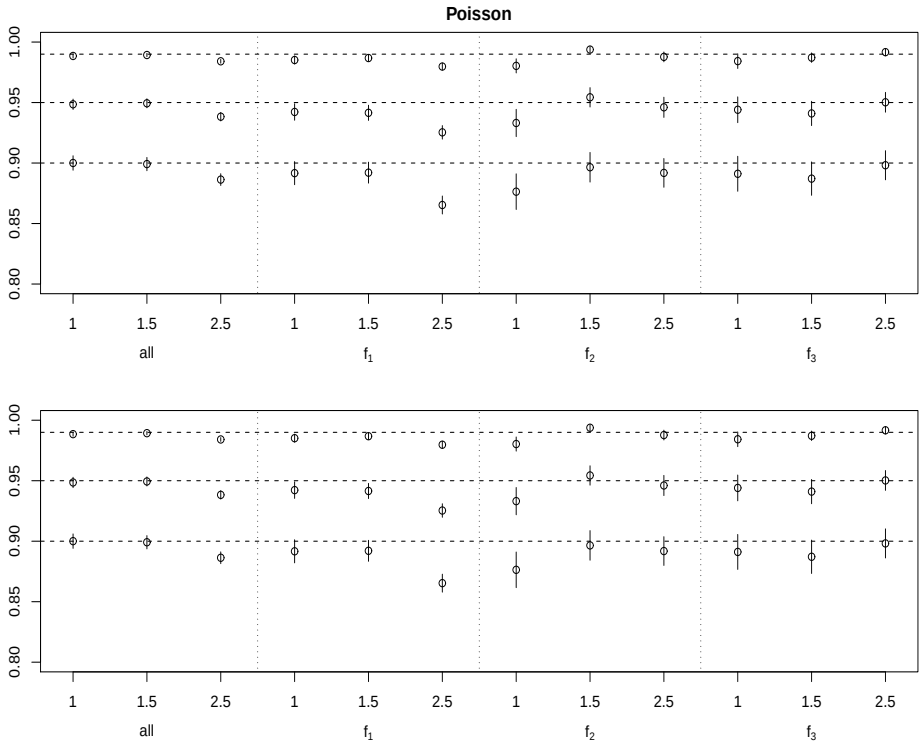


Figure 5-21: Realized coverage probabilities for confidence intervals from the mono-GAM simulation study of the first example, for Poisson data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each smooth term and for the overall model ("all"). The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

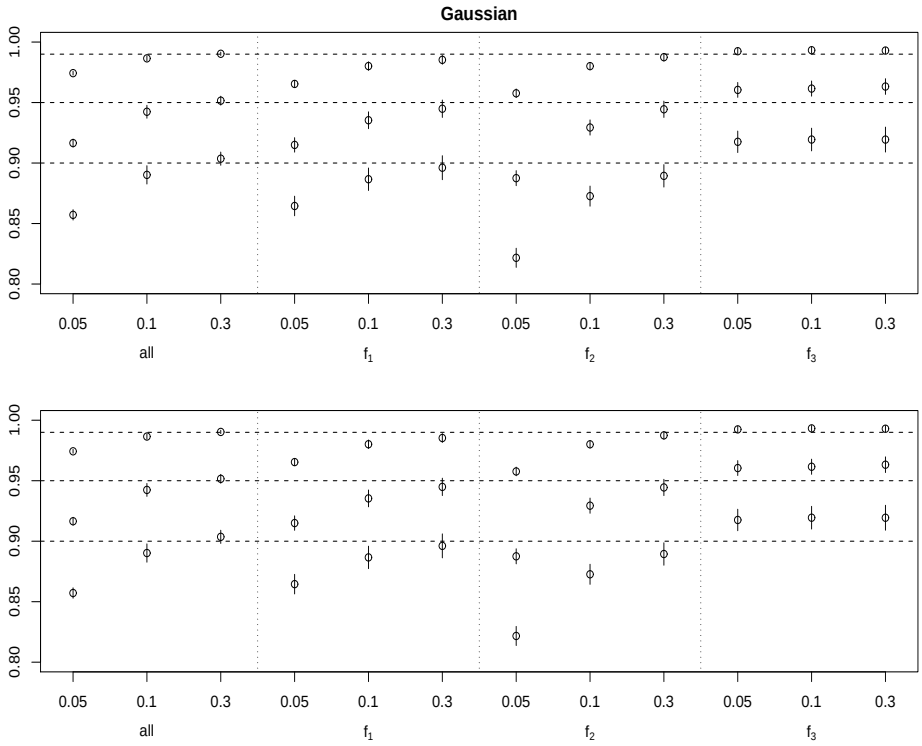


Figure 5-22: Realized coverage probabilities for confidence intervals from the mono-GAM simulation study of the second example, for normal data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each smooth term and for the overall model ("all"). The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

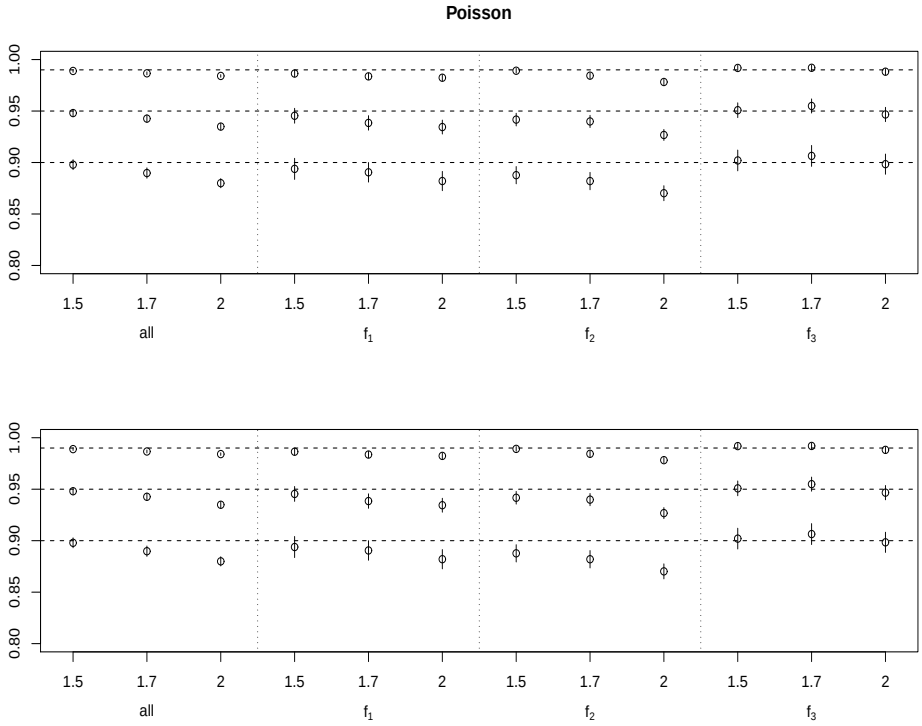


Figure 5-23: Realized coverage probabilities for confidence intervals from the mono-GAM simulation study of the second example, for Poisson data for $n = 200$ (top panel) and $n = 500$ (bottom panel). Three noise levels are used for each smooth term and for the overall model ("all"). The nominal coverage probabilities of 0.90, 0.95, and 0.99, are shown as horizontal dashed lines. $'\circ'$ indicates the average realized coverage probabilities over 500 replicate data sets. Vertical lines show twice standard error intervals of the mean coverage probabilities.

Chapter 6

R package monogam

This chapter describes the design and usage of the R package `monogam` which implements generalized additive modelling with monotonicity restrictions on some smooth terms using the framework presented in the previous chapters. The model setup is the same as in `mgcv(gam)` with the added shape constrained smooths implemented by use of the `mgcv` constructor method function for smooth terms: `smooth.construct`. By using this approach the unconstrained smooths of one or more variables available in `mgcv` can be included in a model, as well as other user defined smooths.

In order to be consistent with the unconstrained GAM the package provides many similar functions to `mgcv`. The key functions are listed in the following table with the last column pointing to the section where each function is described.

Function	Description	Section
`monogam`	the main function to fit a mono-GAM to data	6.1, 6.3
`print.monogam`	printing the basic fitting information	6.1
`plot.monogam`	plotting component smooth functions of the mono-GAM on the linear predictor scale	6.2
`summary.monogam`	extracting the model fitting results	6.4
`monogam.check`	for plotting standard diagnostic information and printing information relating to a quasi-Newton optimization	6.5
`predict.monogam`	producing predictions based on a new or the original set of values of the model covariates	6.6

These functions are almost clones of the corresponding `mgcv` library codes with some modifications to adopt shape preserving smooth terms introduced in Chapters 1 and 2, and with differences in the construction of the Bayesian confidence intervals, which were described in Chapter 5.

The first section provides information about the built-in smoothing bases with shape constraints. This is followed by the `plot` method for plotting the univariate and bivariate smooth terms with or without confidence intervals. There is also a description of the inclusion of a 'by' variable and parametric model terms in the mono-GAM setting. Separate sections are addressed to each of the rest key functions.

6.1 Built-in shape constrained smoothers

This section describes the shape constrained smoothing bases available in the `monogam` package. It starts with information about the univariate smoothers subject to monotonicity and monotonicity plus convexity, which were proposed in Sections 2.1, 3.1, and 3.2. Then the usage of bivariate smooths with monotonicity restrictions, introduced in

Section 3.3, is described. The section also illustrates the `print` method. The code for generating data sets of this section is given in Appendix A.

Univariate shape constrained smoothers

The second example of Chapter 4 may be implemented as follows:

```
> b1 <- monogam(y~s(x1,k=20,bs="ps",m=2)+s(x2,k=30,bs="mpi",m=2)+
     s(x3,k=30,bs="mdcx",m=2), family=poisson(link="log"),data=dat1,
      optimizer="bfgs")
```

where the formula is similar to that used with function `gam()` from `mgcv`: `y` represents the response variable and the smooth terms of the relevant covariate are coded using `s` functions. `s` is an `mgcv` function which defines the smooth term within the generalized additive model, `k` is the basis dimension, `bs` denotes the type of penalized smoothing basis to be used for the smooth, and `m` is the order of the smoothing basis. For this example the unconstrained penalized P-spline basis (`bs="ps"`) is used for the first smooth term, a monotone increasing P-spline (`bs="mpi"`) for the second smooth, and a monotone decreasing plus convex P-spline (`bs="mdcx"`) for the last one. The `data` argument of the `monogam` function is a list or data frame containing all the variables required by the formula. The default method for smoothing parameter selection is the BFGS algorithm for GCV/UBRE minimization implemented in the `monogam` package, which uses the GCV/UBRE derivatives described in Section 4.3. The package also allows use of `optim()` or `nlm()` numerical optimization methods for smoothing parameter estimation by specifying the argument `optimizer` (`optimizer="optim"`, `optimizer="nlm"`, or `optimizer="nlm.fd"`). The last option uses the finite-difference approximation of the criterion derivatives. Each of the alternative methods built in to the `optim()` routine can be used for mono-GAM estimation by indicating it in the argument `optim.method`, consisting of two elements: the method itself as the first element and the second element indicating whether the finite difference approximation should be used ("fd") or analytical gradients ("grad"). The default is `optim.method=c("Nelder-Mead","fd")`.

By typing `b1` or `print(b1)` the short-form model summary is printed.

```
> b1
Family: poisson
Link function: log

Formula:
```

```
y ~ s(x1, k = 20, bs = "ps", m = 2) + s(x2, k = 30, bs = "mpi",
    m = 2) + s(x3, k = 30, bs = "mdcx", m = 2)

Estimated degrees of freedom:
7.6355 3.2014 2.4047  total = 14.24153

UBRE score: 0.06995077
```

The print method displays the model distribution family together with the link function, formula, and the effective degrees of freedom for each term, the total edf includes also one degree of freedom of the model intercept. For this example the edf of the unconstrained smooth was 7.64, the monotone increasing term had 3.2 effective degrees of freedom, and the edf was 2.4 for the last smooth component. The UBRE score of the fitted model is reported at the end.

Any unconstrained smoothing basis built in to the `mgcv` package such as, for example, cubic regression splines, cyclic cubic regression splines, or cyclic P-splines, may be added to the linear predictor of the mono-GAM. Besides the two shape constrained smooths used in the above example, there are four other univariate shape preserving smoothing bases built in to `monogam`. The full list is shown in Table 6.1.

Table 6.1: Univariate shape constrained smoothing bases available in the package

Bases name	Description
`"mpi"`	Monotone increasing P-splines
`"mpd"`	Monotone decreasing P-splines
`"micx"`	Monotone increasing and convex P-splines
`"micv"`	Monotone increasing and concave P-splines
`"mdcx"`	Monotone decreasing and convex P-splines
`"mdcv"`	Monotone decreasing and concave P-splines

Bivariate smooths with monotonicity restriction

The tensor product bivariate smooths with double or single monotonicity, introduced in Section 3.3, can be added to the model via a model formula `s()` term. The built-in bivariate smoothing bases are displayed in Table 6.2.

Table 6.2: Tensor product bivariate shape constrained smoothing bases available in the package

Bases name	Description
`"tedmi"`	Tensor product smoothing constructor for a bivariate function subject to double monotone increasing constraint
`"tedmd"`	Tensor product smoothing constructor for a bivariate function subject to double monotone decreasing constraint
`"tesmi1"`	Tensor product smoothing constructor for a bivariate function monotone increasing in the first covariate
`"tesmi2"`	Tensor product smoothing constructor for a bivariate function monotone increasing in the second covariate
`"tesmd1"`	Tensor product smoothing constructor for a bivariate function monotone decreasing in the first covariate
`tesmd2"`	Tensor product smoothing constructor for a bivariate function monotone decreasing in the second covariate

As an example, the Gaussian model with double monotonicity of Example 3 of Section 3.4

$$y_i = f(x_{1i}, x_{2i}) + \epsilon_i, \;\; \epsilon_i \sim N(0, \sigma),$$

can be fitted by calling

```
> b2 <- monogam(y~s(x1,x2,k=c(10,10),bs="tedmi",m=2),
+                family=gaussian(link="identity"),data=dat2)
> b2

Family: gaussian
Link function: identity
```

```
Formula:
y ~ s(x1, x2, k = c(10, 10), bs = "tedmi", m = 2)

Estimated degrees of freedom:
7.1962  total = 8.196193

GCV score: 0.01027492
```

It should be noted that the `k` argument of `s()` formula term is supplied as a vector denoting the marginal dimensions for each marginal basis. This is different to the unconstrained `mgcv(gam)` where `k` is the dimension of the basis used to represent the smooth term. If the dimensions are the same for both marginal bases, as in this example, then `k` can also be supplied as a constant, e.g., `k=10`.

The next code demonstrates fitting of a bivariate smooth monotone increasing in the second covariate, x_2 (see Example 4, Section 3.4).

```
> b3 <- monogam(y~s(x1,x2,k=10,bs="tesmi2",m=2),
+             family=gaussian(link="identity"), data=dat3)
> b3

Family: gaussian
Link function: identity

Formula:
y ~ s(x1, x2, k = 10, bs = "tesmi2", m = 2)

Estimated degrees of freedom:
16.186  total = 17.18597

GCV score: 0.01022341
```

In principal, unconstrained smooths of any number of covariates can be added to `monogam` model formula via smooths built in to `gam`, such as thin plate regression splines or tensor products of any unconstrained bases available.

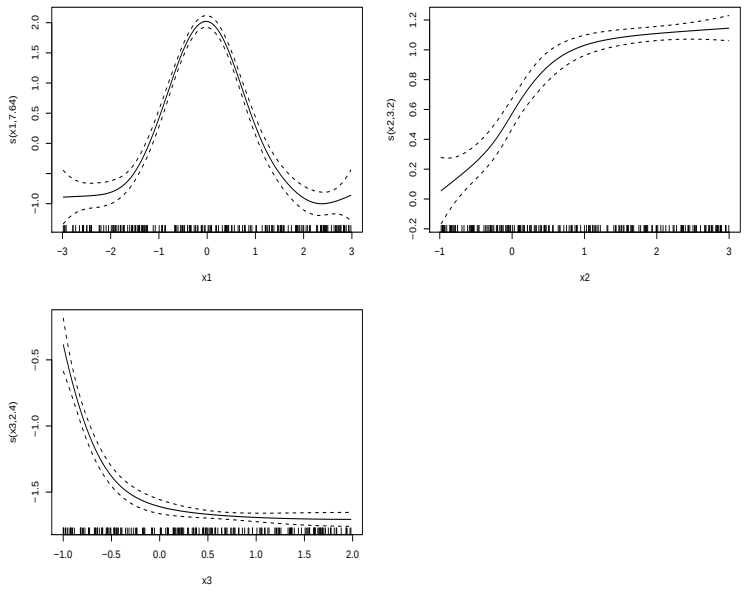
6.2 Plot method

This section illustrates the `plot` method for a fitted mono-GAM. The `plot.monogam` function is similar to `plot.gam` with differences in the construction of the Bayesian confidence intervals for the smooth shape-constrained model terms. It produces plots showing one and two-dimensional smooth components of the fitted model. Optionally, the partial residuals can be added to the one-dimensional plots and standard error lines to one and two-dimensional smooths. For convenience, and to aid understanding, all figures of this section are shown straight after the relevant code, without captions.

Plots of one-dimensional smooths

The results of the first univariate additive model can be checked by plotting the fitted model components. When the `monogam` object is passed to the `plot()` function, the plot method produces three curves for each fitted smooth component, on the linear predictor scale.

```
> par(mfrow=c(2,2),mar=c(5,5,1,1))
> plot(b1,scale=0)
```



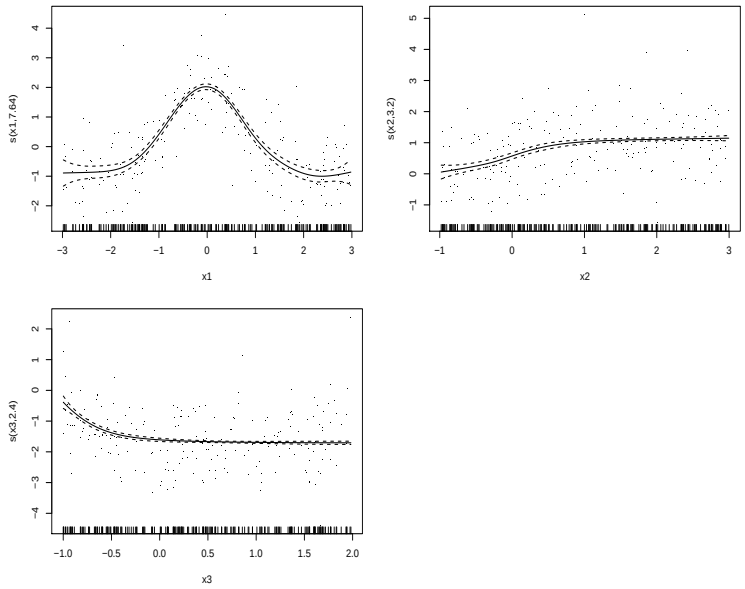
The default plotting results in separate plots for the fitted curves of each smooth model component shown as solid lines, and the dashed lines indicate two standard errors about the fits. The option for confidence interval plotting can be switched off

by setting argument `se=FALSE`. It is also possible to supply a positive number for `se` in which case the standard errors are multiplied by this number when obtaining the confidence intervals of the model components.

The default value of the `scale` argument is -1 which produces plots with the same y-axis scale for each component, 0 produces a different y-axis scale for each plot. The label of the vertical axis reports the illustrated smooth model component, the corresponding covariate, and the estimated degrees of freedom, which are given in the round brackets.

In addition, it is possible to show the partial residuals for each smooth by using the `residuals` arguments. The following code illustrates this.

```
> par(mfrow=c(2,2),mar=c(5,5,1,1))
> plot(b1,scale=0,residuals=TRUE)
```



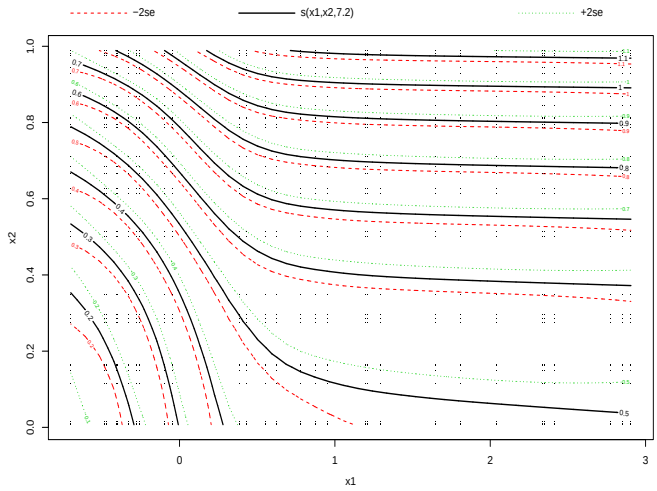
The plot method allows modification of the range of the axis scales on a plot by specifying `xlim` or `ylim` arguments of the `plot()` function. There are also arguments for specifying plot labels: `xlab` and `ylab` for axes labels, and `main` for a title.

Plots of two-dimensional smooths

The `plot` method provides two options for visualizing three-dimensional data. These are contour plots, for producing contours representing the value of the linear predictor, and perspective plots for producing 3D surfaces. The contour plot for two-dimensional

smooths can be plotted by passing the `monogam` object to the `plot()` function. This is illustrated on the second example from the previous section.

```
> plot(b2)
```

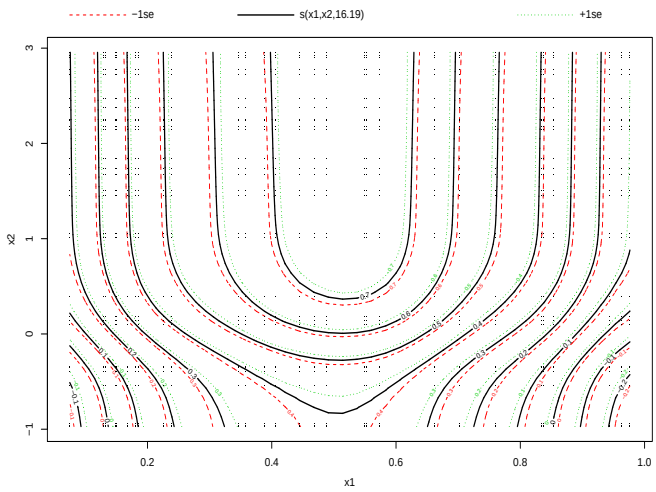


There are 3 contour plots:

1) of the estimate (black),
2) of the estimate plus 2 standard errors (green),
3) of the estimate minus 2 standard errors (red).

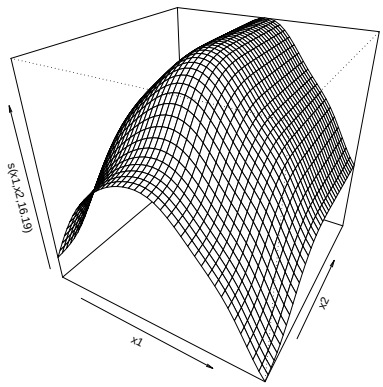
The next bit of code shows the contour plot of the bivariate model with single monotonicity along the x_2 direction (see Section 6.1), where the `se` argument was supplied with the numerical value.

```
> plot(b3,se=1)
```



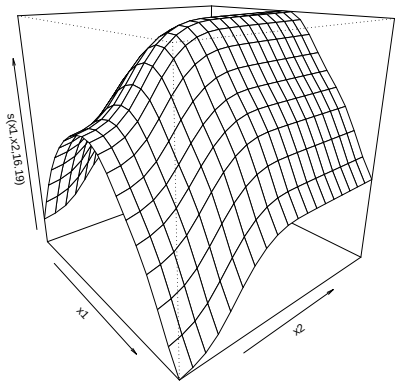
The 3D surface of the same fit can be plotted by changing the default logical value of the `pers` argument to `TRUE`.

```
> plot(b3, pers=TRUE)
```



As for the `persp()` function, it is possible to change the viewing direction of the surface using the `theta` argument for azimuthal direction and `phi` the colatitude. The square root of the number of points used for construction of the 2D function estimates is controlled by the `n2` argument, the default value is 40. The next line demonstrates the use of these arguments.

```
> plot(b3,theta = 50, phi = 20,pers=TRUE,n2=20)
```



It should be mentioned that, as for `plot.gam()`, the `plot` method for the `monogam` object can not produce plots of the smooths of more than two covariates.

6.3 Example with a 'by' variable and parametric model terms

As for `gam`, `monogam` allows inclusion of a 'by' variable as well as parametric model terms. Therefore, it is not difficult to fit a mono-GAM of the general structure (4.5) introduced in Section 4.1.2. The current section illustrates two basic examples of mono-GAM with a parametric term, and a variable coefficient model.

Mono-GAM with a parametric model term

For simplicity, consider the Poisson model mentioned at the beginning of this chapter, but now only the first two smooth terms are added in to the linear predictor and it includes a parametric term x_3:

$$\log(\mu_i) = \eta_i = x_{3i} + f_1(x_{1i}) + f_2(x_{2i}),$$

where $\mu_i = \mathrm{E}(Y_i)$, $Y_i \sim \mathrm{Pois}\left\{\exp(\eta_i)\right\}$. The following code shows the use of `monogam` to fit this model.

```
> b4 <- monogam(y~x3+s(x1,k=20,bs="ps")+s(x2,k=30,bs="mpi"),
+        family=poisson(link="log"),data=dat)
> b4

Family: poisson
Link function: log

Formula:
y ~ x3 + s(x1, k = 20, bs = "ps") + s(x2, k = 30, bs = "mpi")

Estimated degrees of freedom:
8.1184 3.6589  total = 13.77727

UBRE score: 0.09896245

> par(mfrow=c(1,2),mar=c(10,5,7,2))
> plot(b4,scale=0)
```

The plot of the smooth terms are illustrated in Figure 6-1. As mentioned previously the argument `scale=0` sets the different y-axis scale for each plot. The degrees of smoothness were selected by the UBRE criterion. Note that the total degrees of freedom equal the sum of the edf for two smooth terms plus two, since one degree of freedom is for the model intercept and one degree for the parametric term x_3.

Variable coefficient model

The last example concerns a 'by' variable term, which is multiplied by the second monotonic smooth component of the linear predictor:

$$\log(\mu_i) = f_1(x_{1i}) + f_2(x_{2i})x_{3i}, \tag{6.1}$$

where $x_{3i} \sim \text{Pois}\{\exp(\eta_i)\}$. The following code fits the model (6.1). The print and plot methods are then called.

```
> b5 <- monogam(y~s(x1,k=20,bs="ps")+s(x2,k=30,bs="mpi",by=x3),
+        family=poisson(link="log"),data=dat)
> b5
```

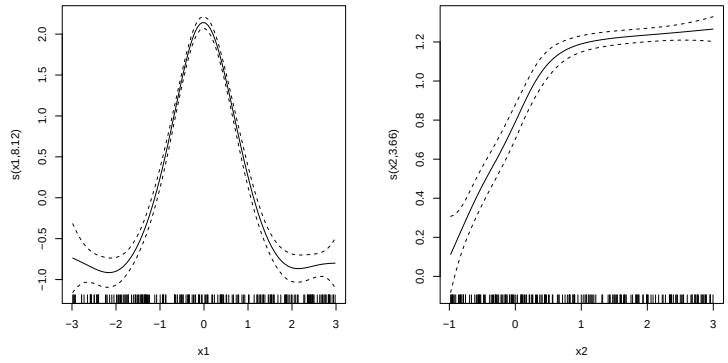


Figure 6-1: Plot of smooth model components of the Poisson model with a parametric term.

```
Family: poisson
Link function: log

Formula:
y ~ s(x1, k = 20, bs = "ps") + s(x2, k = 30, bs = "mpi", by = x3)

Estimated degrees of freedom:
10.077  3.339  total = 14.41621

UBRE score: 0.1842804

> par(mfrow=c(1,2),mar=c(10,5,7,2))
> plot(b5,scale=0)
```

This yields the plot displayed in Figure 6-2. The 'by' variable, x_3, is included in the formula through the term `s()`, as an argument `by`. Section 4.1.2 explains how to estimate the 'by' variable model.

6.4 Summary method

More detailed fitting results can be obtained by using the `summary` method. The code `summary.gam` of the `mgcv` package is used for `summary.monogam`, with slight modifica-

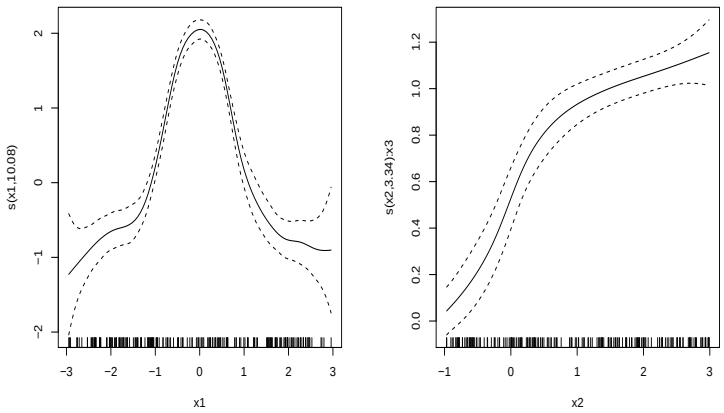


Figure 6-2: Plot of smooth model components of the Poisson model with a 'by' variable.

tions to accept the exponentiated parameters of the monotone smooth terms and the corresponding covariance matrix described in Chapter 5.

The `summary` method for the last model (6.1) of the previous section produces:

```
> summary(b5)

Family: poisson
Link function: log

Formula:
y ~ s(x1, k = 20, bs = "ps") + s(x2, k = 30, bs = "mpi", by = x3)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.80791    0.06136   13.17   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Approximate significance of smooth terms:
            edf Ref.df Chi.sq p-value
s(x1)    10.077 10.077 1206.6  <2e-16 ***
```

```
s(x2):x3  3.339  3.339  275.5  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

R-sq.(adj) =  0.932   Deviance explained = 90.8%
UBRE score = 0.18428  Scale est. = 1         n = 200
```

The only parametric coefficient for this example is a model intercept. The estimated value of the intercept and its significance are given. Approximate significance is reported for each smooth component. The same approach as for `gam` is used to obtain all significance measures (Wood, 2006a). It should be mentioned that the p-values act properly for un-penalized models, however, since smoothing parameters should be estimated in most of the cases, but they are actually treated as fixed in the distributions used for testing, the p-values are not strictly correct.

The adjusted r^2

$$r^2_{adj} = 1 - \frac{\sum \hat{\epsilon}_i^2/(n-\tau)}{\sum (y_i - \bar{y})^2/(n-1)},$$

where $(n-\tau)$ is the residual degrees of freedom, indicates how the model works in explaining the variability in the response variable. Also the percentage deviance explained is presented which is calculated using the following:

$$\text{Deviance explained} = \left\{ D_{\text{null}} - D(\hat{\boldsymbol{\beta}}) \right\} / D_{\text{null}},$$

where the null deviance, D_{null}, is the deviance for a model consisting of a single constant term, and $D(\hat{\boldsymbol{\beta}})$ is the deviance of the fitted model. When the scale parameter of the model is unknown, its estimate is reported, otherwise, the known value is printed out.

6.5 Model checking

After fitting the model, it is natural to check the model assumptions graphically. The `monogam.check()` function provides some standard residual plots and prints information about convergence results of the numerical optimization method used to select smoothing parameters. This function is similar to the `gam.check()` routine of the `mgcv` library. For the model `b5` the `monogam.check()` function produces the following:

```
> monogam.check(b5)

Method: UBRE   Optimizer: bfgs
```

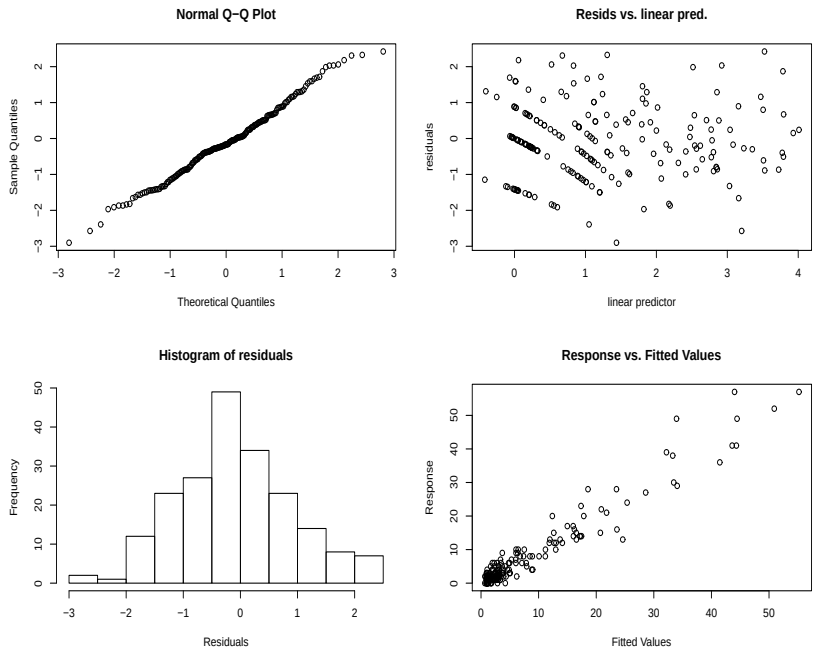


Figure 6-3: Model checking plots for the Poisson model with a 'by' variable.

```
Number of iterations of smoothing parameter selection performed was 2 .
Full convergence.
Gradient range: [-2.764789e-10,5.248437e-07]
(score 0.1842804 & scale 1)
The optimal smoothing parameter(s): 3.58252 0.00725 .
```

The resulting plots are shown in Figure 6-3. The upper left panel is a normal quantile-quantile plot. If the assumption about model distribution is correct then this plot should look like a straight line relationship. The upper right panel is the residuals versus fitted values on a linear predictor scale which allows checking of the independence of the response variables: there should be no trend in the mean of the residuals; and also checking of the constant variance assumption: a trend in the variability of the residuals would violate this assumption. The histogram of residuals shown in the lower left panel provides another way of checking the model distribution assumption, it should be approximately consistent with the normal distribution, if the distribution is reasonable. The lower right panel is the response against fitted values. It shows a positive linear relationship and a sensible dispersion.

6.6 Prediction method

Finally, it is possible to predict the expected values of the response variable at new sets of values for the model covariates. `predict.monogam` is the `predict` method function. The design, functionality, and layout of this function follow closely that of `predict.gam` in the `mgcv` library (Wood, 2006a), with the exception of some modifications to allow shape preserving smooth terms. The function produces predictions at new values of the model covariates or the original values, used for the model fit. Also it is possible to obtain standard errors of those predictions based on the posterior distribution of the model coefficients.

To produce predictions on a new data set, the new values of covariates should be supplied as a data frame in an argument `newdata`. For the previous example, to make predictions in two new points, the following code can be called:

```
> nd <- data.frame(x1=c(-0.5,1.5),x2=c(-0.85,0.95),x3=c(0.15,0.9))
> predict.monogam(b5,newdata=nd)
       1        2
2.419149 1.150741
```

The resulted predictions by default are the values of the linear predictor. The function argument `type` controls the type of prediction (`type="link"` by default). For predictions on the scale of the response variable, this argument should be set to `"response"`. Also the approximate standard errors may be returned by setting `se=TRUE`.

```
> predict(b5,newdata=nd,type="response",se=TRUE)
$fit
        1         2
11.236294  3.160534

$se.fit
        1         2
0.9563697 0.4752105
```

For an additive model it is useful to get predictions for each component of the linear predictor, excluding the intercept, with or without corresponding standard errors. The following code illustrates this.

```
> predict(b5,newdata=nd,type="terms",se=TRUE)
$fit
```

```
       s(x1)   s(x2):x3
1  1.5992109 0.01202865
2 -0.4887201 0.83155148

$se.fit
       s(x1)   s(x2):x3
1 0.07881036 0.02863583
2 0.13488008 0.04790978

attr(,"constant")
(Intercept)
  0.8079096
```

If the data frame `newdata` is not supplied then the predictions are returned for all the original data used for fitting procedure. The following prints only the first three fitted values on the linear predictor scale, without standard errors:

```
> predict(b5)[1:3]
         1          2          3
1.20365266 0.08277808 1.91563910
```

and with standard errors on the response scale:

```
> pr <- predict(b5,type="response",se=TRUE)
> pr$fit[1:3]
       1        2        3
3.332266 1.086301 6.791278
> pr$se[1:3]
        1         2         3
0.4698944 0.1556089 0.6206724
```

For simplicity only simulated examples have been considered in this chapter. Use of the `monogam` package for generalized additive modelling with monotonicity constraints of real data sets will be discussed in Chapter 8.

Chapter 7

Simulations: comparison with alternative methods

To illustrate the performance of the mono-GAM with parameter estimation by the Newton-Raphson based method and smoothness selection by direct minimization of the GCV/UBRE score based on the implicit function theorem, some simulation studies on various models were conducted. Comparison with unconstrained GAM, the quadratic programming approach to shape preserving smoothing (Wood, 1994), and constrained P-splines regression (Bollaerts et al., 2006b) was undertaken. Simulated examples on univariate single smooth term models, bivariate single smooth models, and additive models with a mixture of unconstrained and monotone smooth terms were considered for evaluation of the performance of the four different approaches, and for timing comparison.

7.1 Single univariate monotone smooth term models

In this section the performance of the proposed method is compared on a single smooth monotone generalized regression model

$$g(\mu_i) = f(x_i), \;\; i = 1, \ldots, n,$$

where $\mathrm{E}(Y_i) = \mu_i$ and Y_i follows a Gaussian or Poisson distribution. Both a gaussian model with an identity link function and a Poisson model with a logarithmic link were considered.

The following simulation scheme was performed:

Gaussian model: Sample of sizes $n = 100$ and $n = 200$ were simulated from

$y_i = f_t(x_i) + \epsilon_i$, where $\epsilon_i \sim$ i.i.d. $N(0, \sigma^2)$ and the test function, $f_t(x)$, was

$$f_t(x) = \exp(4x)/\{1 + \exp(4x)\} + 0.5.$$

The covariate values, x_i, were generated from the uniform distribution on $[-1, 3]$ and the function $f_t(x_i)$ was used to get the true mean μ_i. Three values of the standard deviation were taken to control the noise level, $\sigma = 0.05$, 0.1, and 0.2. The noise levels in these situations were such that the signal to noise ratios, R^2 (5.17), were about 0.98, 0.93, and 0.76 respectively. 300 replicate data sets were generated for this model each of three noise levels and for two sample sizes.

Poisson model: Sample of sizes $n = 100$ and $n = 200$ were generated from $\log(\mu_i) = f_t(x_i)$, where $\mathrm{E}(Y_i) = \mu_i$ and $Y_i \sim \mathrm{Poi}\left[\exp\left\{f_t(x_i)\right\}\right]$,

$$f_t(x) = d \times \left[\exp(4x)/\{1 + \exp(4x)\} + 0.5\right].$$

d was used to control the level of noise in this case: $d = 2$, 3, and 4, which correspond to signal to noise ratios of about 0.76, 0.95, and 0.99. As above, x_i were generated from the uniform distribution on $[-1, 3]$. 400 replicates were produced for the Poisson distribution for three levels of noise and for two sample sizes.

The mono-GAM approach was compared to unconstrained penalized regression splines as implemented in R package `mgcv`, the quadratic programming approach (QP) (see Section 2.7.1), and P-splines regression with additional discrete penalties (DPP) (see Section 2.7.2). For the implementation of the quadratic programming approach with linear inequality constraints to preserve monotonicity, R function `pcls()` of the `mgcv` library was used. This function solves a penalized least squares problem subject to linear equality and inequality constraints using quadratic programming by the algorithm given in Gill et al. (1981). Since no code was available for performing the approach of Bollaerts et al. (2006b), R routines were written for its implementation.

For three alternative approaches, the smooth function $f(x)$ was represented by a P-spline with $q = 15$, while monotonic P-splines were used for the mono-GAM method with the same basis dimension. Smoothing parameters were selected by GCV (2.28) for the Gaussian case and UBRE (2.26) for the Poisson model for each replicate. For mono-GAM a quasi-Newton algorithm, BFGS, with the derivatives calculation as proposed in Section 4.3 was used to optimize the smoothing parameter estimation criterion. For better comparability a BFGS numerical optimization method was selected while running the `gam(mgcv)` function. Smoothing parameters for the quadratic programming problems were chosen from the unconstrained GAM. Since there was no efficient

method for smoothing parameter estimation with the DPP models, direct grid search for the GCV/UBRE optimal smoothing parameter was applied in this case.

Gaussian data

The performance of the smoothing methods was evaluated by the mean sum of squared differences between the fitted, $\hat{f}(x)$, and true values of $f_t(x)$ in the case of a Gaussian distribution,

$$\text{MSE} = n^{-1} \sum_{i=1}^{n} \left\{ \hat{f}(x_i) - f_t(x_i) \right\}^2 .$$

The results are summarized in Figure 7-1. The simulation results show that the performance of the mono-GAM is better than that of the unconstrained GAM and the DPP approach for all considered levels of noise. There is also a slight advantage of the mono-GAM over the quadratic programming method. When comparing mono-GAM and QP we expect some differences because i) although the constraints are the same for both methods, the penalties are different and ii) QP uses ad hoc smoothing parameter selection and mono-GAM does not. Point ii) possibly explains the mono-GAM superior performance. The results also suggest that the monotone P-splines work better than other approaches for greater levels of noise.

The new method has higher computational cost (see Figure 7-2) than the quadratic programming approach and the unconstrained GAM. This is mainly due to the expensive singular value decompositions of the working model matrix. However, taking into account the optimality of the smoothness selection of the new method and the much less advantageous selection of λ from the unconstrained model in the quadratic programming approach, the computational speed seems to be reasonable. Since separate boxplots for each of three noise levels did not give any additional information, the time distribution was illustrated for all three noise levels jointly. All computations were run on Intel(R) Pentium(R) Dual CPU, E2160 @ 1.80 GHz, 1.79 GHz, 1.98 GB of RAM, and performed with R 2.10.1 (R core development team, 2009).

Poisson data

Following Wood (2008) for the Poisson regression a predictive deviance loss (PDL) may be taken as a measure of the fitting method performance. In order to calculate the PDL, 10000 sets of new data were generated from the truth after fitting the model, and the mean value of the response variable, $\hat{\boldsymbol{\mu}}$, was predicted using the fitted model. Then,

$$\text{PDL} = \text{D}^*(\mathbf{y}, \hat{\boldsymbol{\mu}}) - \text{D}^*(\mathbf{y}, \boldsymbol{\mu}),$$

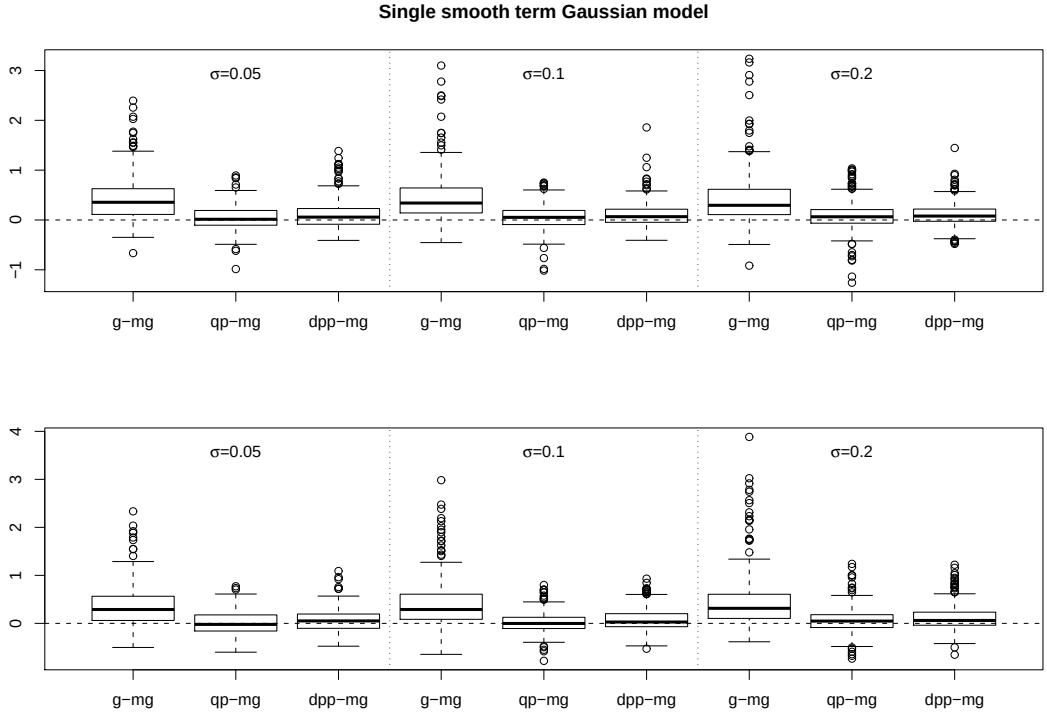


Figure 7-1: MSE comparisons between mono-GAM (mg), GAM (g), quadratic programming (qp), and P-splines with additional discrete penalty (dpp) approaches for the Gaussian distribution for each of three noise levels. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of differences in relative MSE between each alternative method and mono-GAM. 300 replicates were used. Relative MSE was calculated by dividing the MSE value by the average MSE of mono-GAM for the given case. (Section 7.1, Gaussian data)

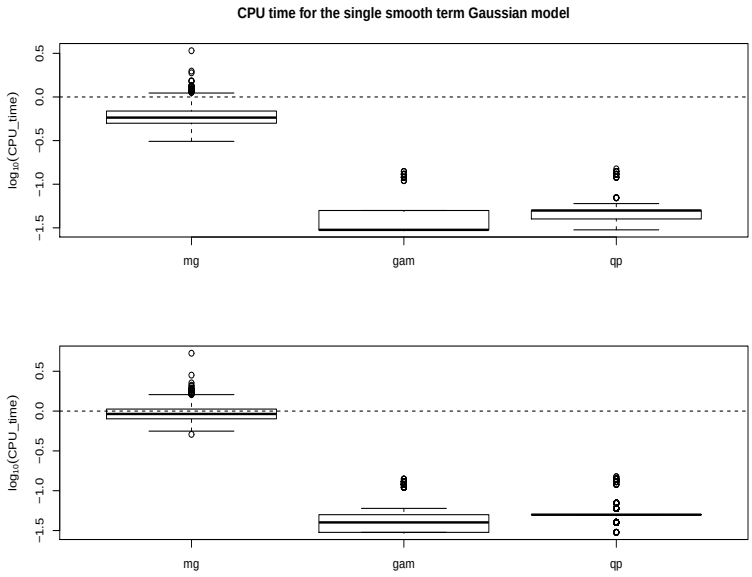


Figure 7-2: CPU time comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Gaussian data. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of $\log_{10}$ CPU time in seconds for thee noise levels combined. (Section 7.1, Gaussian data)

where $\mathrm{D}^*(\mathbf{y}, \hat{\boldsymbol{\mu}})$ is the mean predictive deviance of the fitted model, $\mathrm{D}^*(\mathbf{y}, \boldsymbol{\mu})$ is the mean predictive deviance using the known true smooth. For the Poisson model

$$\mathrm{D}^*(\mathbf{y}, \boldsymbol{\mu}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ 2y_i \log\left(\frac{y_i}{\mu_i}\right) - 2(y_i - \mu_i) \right\}.$$

The results for 400 replications of the simulated data are given in Figure 7-3. Mono-GAM outperforms the alternative methods for the cases with greater noise levels ($d = 2$, 3), but when the signal to noise ration $R^2 = 0.99$ the difference in the predictive deviance loss of the mono-GAM from the QP and DPP approaches are negligible. The distribution of the CPU time for the three methods combined is shown in Figure 7-4. However, note that in this case the mono-GAM was faster than quadratic programming for the case of the greatest noise level.

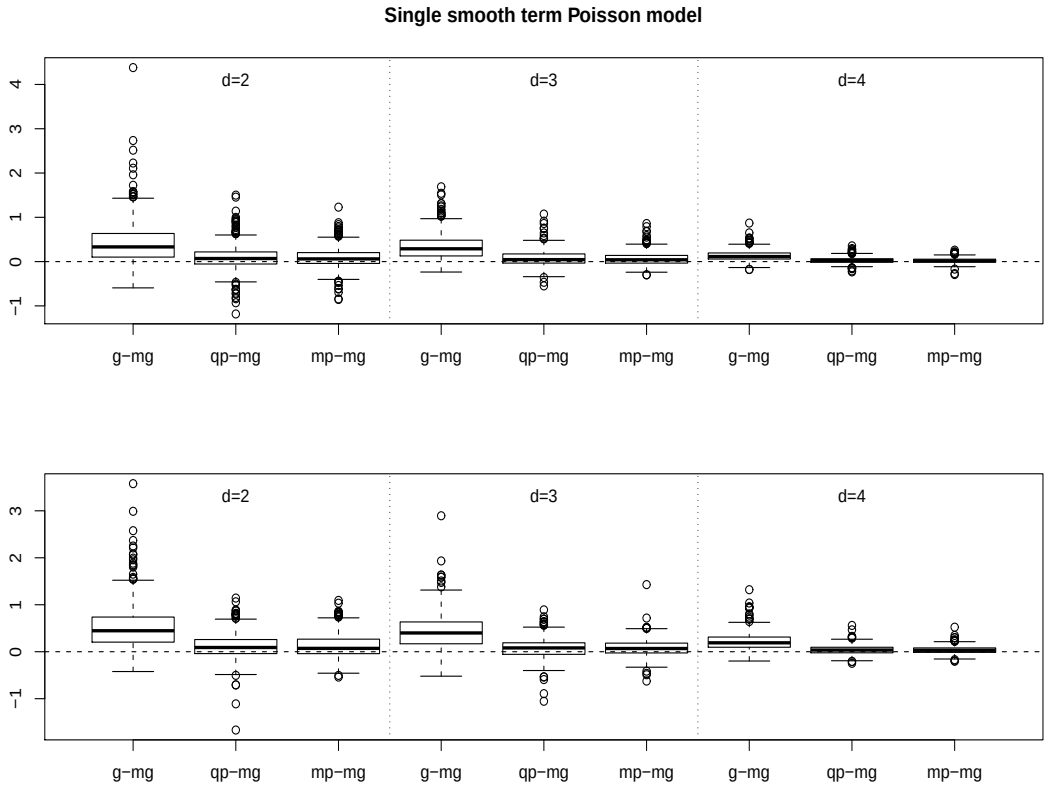


Figure 7-3: PDL comparisons between mono-GAM (mg), GAM (g), quadratic programming (qp), and P-splines with additional discrete penalties (dpp) approaches for the Poisson distribution for each of three noise levels. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of differences in relative PDL between each alternative method and mono-GAM. 400 replicates were used. Relative PDL was calculated by dividing the PDL value by the average PDL of mono-GAM for the given case. (Section 7.1, Poisson data)

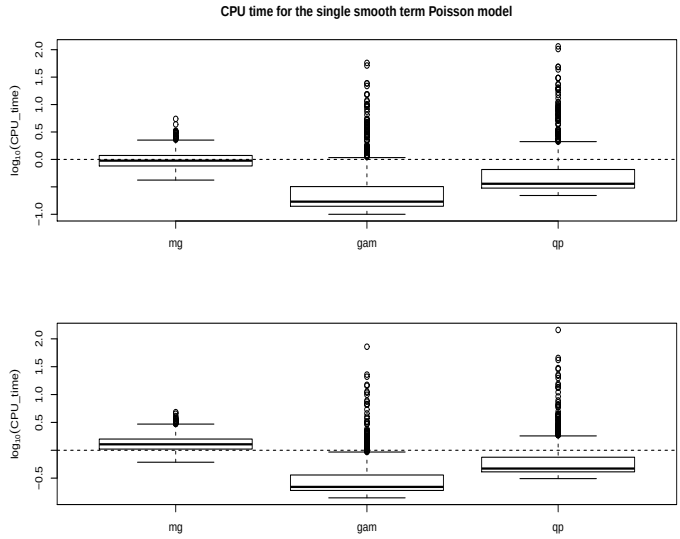


Figure 7-4: CPU time comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Poisson data. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of $\log_{10}$ CPU time per replicate in seconds for three noise levels combined. (Section 7.1, Poisson data)

7.2 Single bivariate monotone smooth term models

We consider a bivariate smooth term model under monotonicity constraint,

$$g(\mu_i) = f(x_{1i}, x_{2i}), \;\; i = 1, \ldots, n,$$

where $\mathrm{E}(Y_i) = \mu_i$ and Y_i follow Gaussian or Poisson distribution as in the previous study. Suppose that the bivariate function $f(x_1, x_2)$ is subject to a single monotone increasing constraint along the second covariate x_2 only.

The simulation scheme was as follows:

Gaussian model: Data were simulated from $y_i = f_t(x_{1i}, x_{2i}) + \epsilon_i$, where ϵ_i are independent normal random variables with parameters $(0, \sigma^2)$. The true function was

$$f_t(x_1, x_2) = 2\sin(\pi x_1) + \exp(4x_2)/\left\{1 + \exp(4x_2)\right\}.$$

The shape of this function is shown in Figure 7-5. 30 values of each of the two covariates, x_{1i} and x_{2i}, were generated from the uniform distribution on $[0, 1]$ and $[-1, 3]$ correspondingly. For each combination of (x_{1i}, x_{2i}) the value of the function $f_t(x_{1i}, x_{2i})$ was calculated to get the true mean μ_i. The function was scaled to have values on

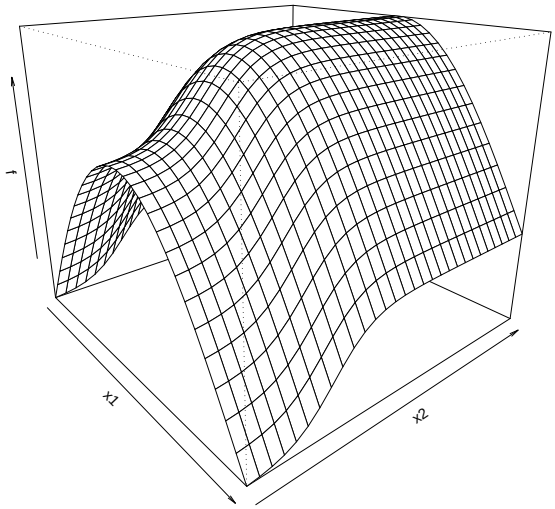


Figure 7-5: Shape of the bivariate function used for the second simulation study. (Section 7.2)

$[0.5, 1.5]$, and three values of the standard deviation were applied as previously, i.e. $\sigma = 0.05$, 0.1, and 0.2. These values of the standard deviation gave noise levels such that the signal to noise ratio, R^2, were about 0.96, 0.85, and 0.59 respectively. 300 replicate data sets were produced for this model at each of three noise levels.

Poisson model: Data were simulated from $\log(\mu_i) = f_t(x_{1i}, x_{2i})$, where $\mathrm{E}(Y_i) = \mu_i$ and $Y_i \sim \mathrm{Poi}\left[\exp\left\{f_t(x_{1i}, x_{2i})\right\}\right]$. The true function was

$$f_t(x_1, x_2) = d \times \left[2\sin(\pi x_1) + \exp(4x_2)/\left\{1 + \exp(4x_2)\right\}\right],$$

where the values of d were 0.7, 1.2, and 1.8, which correspond to R^2 of about 0.47, 0.88, and 0.98.

30 values of each of the two covariates, x_{1i} and x_{2i}, were generated from the uniform distribution on $[0, 1]$ and $[-1, 3]$ correspondingly. For each combination of (x_{1i}, x_{2i}) the value of the function $f_t(x_{1i}, x_{2i})$ was calculated to get the true linear predictor. 300 replicate data sets were produced for this model at each of three noise levels.

Since there was no advantage of the DPP approach over the mono-GAM for the univariate models and moreover, the direct grid search for multiple optimal smoothing

parameters is computationally expensive, (and it is time expensive to write R routines for the implementation of this method) the comparison for this example and for an example of the next section were performed only with the unconstrained GAM and quadratic programming approach.

The tensor product bivariate smooth with single monotonicity introduced in Section 3.3.2 with the marginal basis dimension $q_1 = q_2 = 9$ was used for the mono-GAM construction. The following code shows fitting by the mono-GAM approach with the data supplied as a data frame `dat`:

```
b <- monogam(y~s(x1,x2,k=c(q1,q2),bs="tesmi2", family=gaussian,
               data=dat)
```

For implementing the unconstrained GAM a tensor product of P-splines for both marginal bases was used, fitted by an 'outer' optimization method using BFGS for minimizing the smoothing parameter estimation criterion, GCV/UBRE. The following code illustrates this:

```
b1 <- gam(y~te(x1,x2,bs=c("ps","ps"),k=c(q1,q2)), family=gaussian,
           data=dat, optimizer=c("outer","bfgs"))
```

The BFGS method is not the default option for `gam()`, but it was used for better comparability with mono-GAM which uses BFGS for GCV/UBRE optimization.

The same tensor product construction was used when performing the quadratic programming approach. In this case, firstly the unconstrained `gam()` function was called in order to estimate the smoothing parameters, then, the `pcls()` routine was applied to solve the quadratic programming problem with linear inequality constraints, given $\boldsymbol{\lambda}$ from the unconstrained fit. The single monotone increasing condition along x_2 (see Section 3.3.2) is

$$\triangle_2^1 \beta_{jk} > 0,$$

where $\triangle_2^1 \beta_{jk} = \beta_{jk} - \beta_{j,(k-1)}$ for the vector of unconstrained working model parameters $\boldsymbol{\beta}$ expressed in the following order:

$$\boldsymbol{\beta} = (\beta_{11}, \beta_{12}, \ldots, \beta_{1q_2}, \beta_{21}, \ldots, \beta_{2q_2}, \ldots, \beta_{q_1 q_2})^T .$$

The above condition of increasing subsequence of $\boldsymbol{\beta}$ can be written as linear inequality constraints:

$$\mathbf{A}\boldsymbol{\beta} > \mathbf{0},$$

where $\mathbf{0}$ is a vector of zeros of dimension $q_1(q_2-1)$, $\mathbf{A} = \mathbf{I}_1 \otimes \mathbf{P}_2$ is a Kronecker product of an identity matrix $\mathbf{I}_1$ of size q_1 and a $(q_2-1) \times q_2$ matrix

$$\mathbf{P}_2 = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & . \\ 0 & -1 & 1 & 0 & 0 & . \\ 0 & 0 & -1 & 1 & 0 & . \\ . & . & . & . & . & . \end{pmatrix}.$$

The QP approach can be implemented in R as follows:

```
f.ug <- gam(y~te(x1,x2,bs=c("ps","ps"),k=c(q1,q2),np=FALSE),data=dat,
                optimizer=c("outer","bfgs"))
# create a model and penalty matrices to be supplied into pcls()...
s <- smooth.construct(te(x1,x2,bs=c("ps","ps"),k=c(q1,q2),np=FALSE),
                      dat, knots=NULL)
# create matrix of coefficients for linear inequality constraints...
P <- diff(diag(q2),difference=1); I1 <- diag(q1); A <- I1%x%P
# create a single list argument to pcls()...
M <- list(X=s$X,p=rep(seq(0.1,1,length.out=q2),q1),C=matrix(0,0,0),
          sp=f.ug$sp,y=y,w=y*0+1)
M$Ain <- A; M$bin <-  rep(1e-12,q1*(q2-1))
M$off<- c(0,0); M$S <- s$S
p <- pcls(M)    # fit spline and get the estimated parameter vector
fv <- Predict.matrix(s,data.frame(x1=x11,x2=x22))%*%p # constrained fit
```

When fitting Poisson models by the QP approach the `pcls()` function should be incorporated in to a P-IRLS loop (see Section 2.7.1).

Figure 7-6 illustrates the simulation results for Gaussian and Poisson data. For the Gaussian model the differences in MSE for both alternative methods are larger than those for the univariate model of the previous section. This is possibly because of the shape of the bivariate function which has a large plateau region, is not always captured by the unconstrained GAM. As before, for the Poisson distribution at the lowest noise level, mono-GAM, GAM, and QP approaches are almost indistinguishable. CPU time performance is shown in Figure 7-7. Mono-GAM was slower than unconstrained GAM and QP for the Gaussian data, but its time performance was better for the Poisson data.

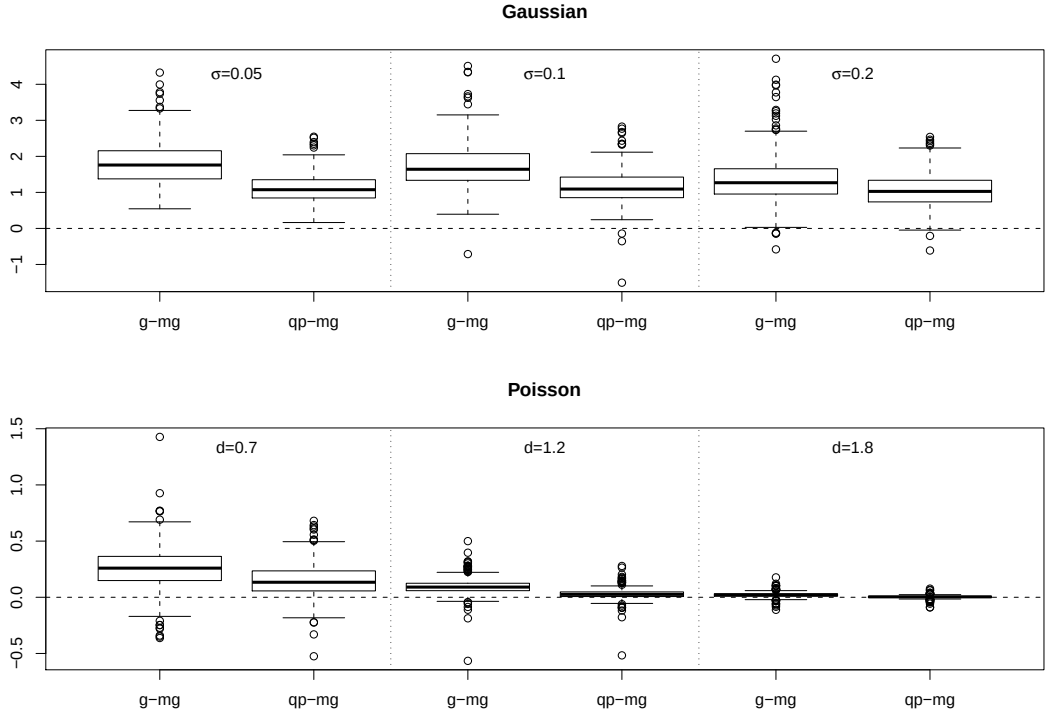


Figure 7-6: Upper panel: MSE comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Gaussian distribution for each of three noise levels. Lower panel: PDL comparisons for the Poisson data. The panels illustrates the results for 300 replicates of the sample size $n = 900$. Boxplots show the distributions of differences in relative MSE/PDL between each alternative method and mono-GAM. Relative MSE/PDL was calculated by dividing the MSE/PDL value by the average MSE/PDL of mono-GAM for the given case. (Section 7.2)

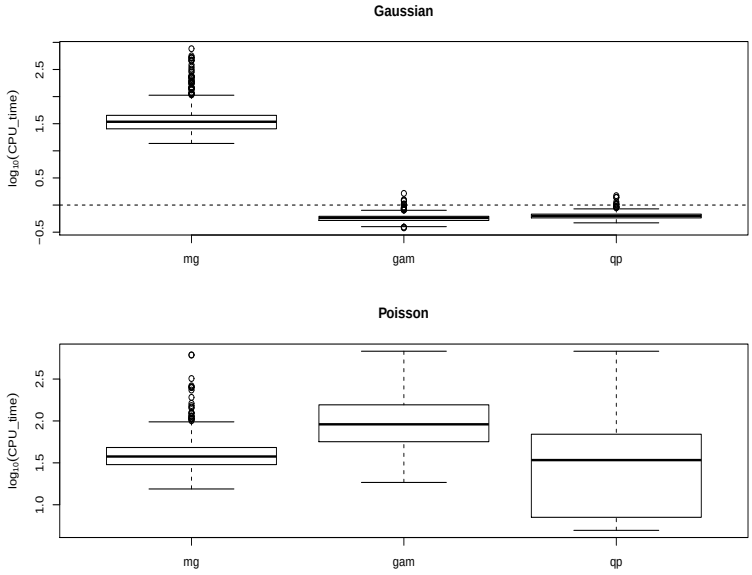


Figure 7-7: CPU time comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Gaussian and Poisson bivariate data. Boxplots show the distributions of $\log_{10}$ CPU time per replicate in seconds. (Section 7.2)

7.3 Additive models

The last simulation study concerns an additive model with two smooth terms, one of which is subject to monotonicity constraint and the other is unconstrained:

$$g(\mu_i) = f_1(x_{1i}) + f_2(x_{2i}), \;\; i = 1, \ldots, n, \tag{7.1}$$

where $\mathrm{E}(Y_i) = \mu_i$, $Y_i \sim$ Gaussian or Poisson distribution. Suppose that the first smooth term is subject to monotonicity but the second one is unconstrained.

Samples of sizes $n = 100$ and $n = 200$ were generated from the next two expressions (Gaussian and Poisson models):

$$y_i = f_{t1}(x_{1i}) + f_{t2}(x_{2i}) + \epsilon_i, \;\; \epsilon_i \sim N(0, \sigma^2),$$

$$\log(\mu_i) = d \times \{f_{t1}(x_{1i}) + f_{t2}(x_{2i})\}, \;\; \mu_i = \mathrm{E}(Y_i), \;\; Y_i \sim \mathrm{Poi}(\mu_i),$$

where d is used to control noise level for the Poisson distribution. The following true functions were used for this study:

$$f_{t1}(x_1) = \exp(4x_1)/\{1 + \exp(4x_1)\},$$

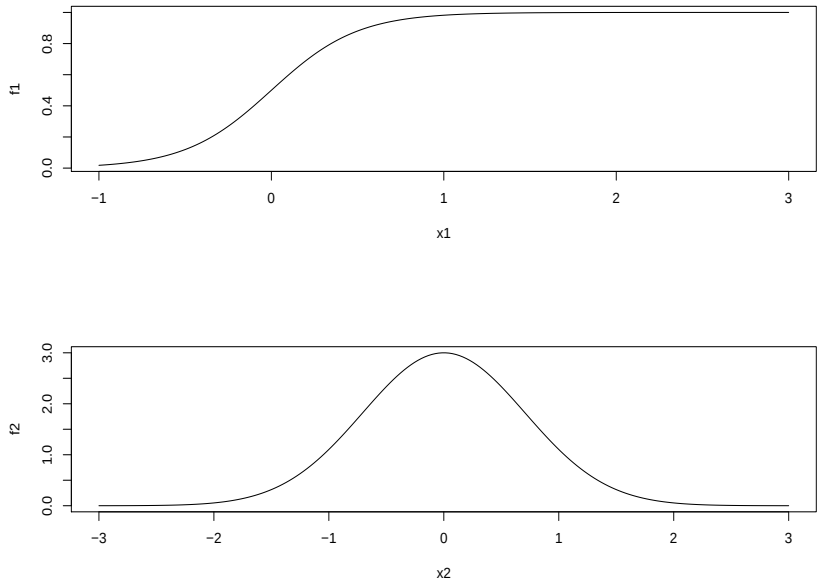


Figure 7-8: Shape of the bivariate function used for the second simulation study. (Section 7.3)

$$f_{t2}(x_2) = 3\exp(-x_2^2).$$

Figure 7-8 shows the shapes of these functions.

The covariate values, x_{1i} and x_{2i}, were simulated from the uniform distribution on $[-1, 3]$ and $[-3, 3]$ respectively. The same values of the standard deviation, σ, as in the previous examples were used for the Gaussian data. The values of d for the Poisson model were 0.5, 0.7, and 1.2, which gave the signal to noise ratio about 0.58, 0.84, and 0.99. 300 replicates were produced for both distributions at each of three levels of noise and for two sample sizes.

For the mono-GAM implementation a monotone second order P-spline of the dimension $q_1 = 30$ was used to represent the first monotonic smooth term and P-spline of the second order with $q_2 = 15$ for the second unconstrained term. The same basis dimensions were applied for two other methods. For an unconstrained GAM, P-splines were used for both model components. The following code fits the mono-GAM and unconstrained GAM using `mgcv` package:

```
b <- monogam(y~ s(x1,k=q1,bs="mpi")+s(x2,k=q2,bs="ps"),
                family=gaussian, data=dat)
b1 <- gam(y~ s(x1,k=q1,bs="ps")+s(x2,k=q2,bs="ps"),family=gaussian,
```

```
data=dat, optimizer=c("outer","bfgs"))
```

Because of the identifiability constraints used for GAM, the simple increasing parameters constraint used in the one-dimensional case cannot be used here. Therefore, for implementing the quadratic programming approach to monotonicity preserving constraint, linear inequality constraints for the QP problem were generated using a finite difference approximation to the first derivative of the smooth represented by cubic regression spline bases. To obtain the monotonicity condition on the first derivative $f_1'(x_1) > 0$, both smooth terms were first represented by cubic regression splines (Wood, 2006a) so that the model (7.1) can be written as $g(\mu_i) = \mathbf{X}_i\boldsymbol{\beta}$, where $\mathbf{X} = [\mathbf{1} : \mathbf{X}_1 : \mathbf{X}_2]$, $\mathbf{X}_i$ is a model matrix for the i^{th} smooth term, and $\boldsymbol{\beta}^T = (\boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T)$. Then, a sequence $\mathbf{x}_1'$ of 100 evenly distributed values of the first covariate, x_1, was generated over the range $[-1, 3]$. These were the points where the derivative $f_1'(x_1)$ should be evaluated. To get the linear inequality constraints two prediction matrices were created, $\mathbf{X}_0' = [\mathbf{1} : \mathbf{X}_{01}' : \mathbf{X}_{02}']$ and $\mathbf{X}_1' = [\mathbf{1} : \mathbf{X}_{11}' : \mathbf{X}_{12}']$. The first model matrix for the first smooth term $\mathbf{X}_{01}'$ was constructed for the sequence $\mathbf{x}_1'$, while $\mathbf{X}_{02}'$ was build for a constant vector, say, $\mathbf{x}_2' = (0.5, \ldots, 0.5)^T$. The same value of the second covariate, x_{2i}, can be taken since only on the first smooth term is subject to monotonicity, so the finite differences should be applied only in the model matrix of the first smooth. For the prediction matrix $\mathbf{X}_{11}'$ a small increment ε, representing the finite difference interval, was added to each covariate value of the first smooth, i.e. $x_{1i}' + \varepsilon$, where $\varepsilon = 10^{-7}$ was taken for this study. The same constant vector $\mathbf{x}_2'$ was used for predicting $\mathbf{X}_{11}'$. Finally, the monotonicity constraint based on finite difference approximation of $f_1'(x_1)$ can be written as linear inequality constraints for the QP problem:

$$\frac{1}{\varepsilon}\left(\mathbf{X}_1' - \mathbf{X}_0'\right)\boldsymbol{\beta} > \mathbf{0},$$

where $\mathbf{0}$ is a vector of zeros. R function `pcls()` was applied to solve the quadratic programming problem subject to these constraints to fit the additive model, given $\boldsymbol{\lambda}$ from the unconstrained fit. A description of the QP approach was given in Section 2.7.1. It should be mentioned that cubic regression splines tend to have slightly better MSE performance than P-splines (Wood, 2006a) and moreover, the conditions built on finite differences are not only sufficient but also necessary for monotonicity. So this is a challenging test for mono-GAM.

The simulation results on comparison of three alternative approaches to the additive model are illustrated in Figures 7-9 and 7-10. The results show that mono-GAM works better than the other two alternative methods. Note that for the Gaussian data the

performance of GAM was better than the performance of the QP approach, but the difference in MSE between mono-GAM and GAM is much less than that in the previous simulation studies. Also it can be noticed that in this case GAM reconstructed the truth better than the QP method. The explanation for that may lay in the fact that there was only one monotonic term, and both GAM and mono-GAM gave similar fits for the unconstrained term, f_2. At lower noise level GAM might also reconstruct the monotone shape of f_1 for some replicates. The worse performance of the QP than of GAM was due to the smoothing parameter estimation from the unconstrained fit which sometimes resulted in more wiggly tails of the smooth term than those of the unconstrained GAM.

From Figure 7-10 one may note that for the Poisson data of the samples size $n = 100$ all three methods worked similarly, but with an increase in sample size mono-GAM outperformed the other two approaches. As in the Gaussian case unconstrained GAM worked better than QP.

As mentioned before, due to the singular value decomposition used for the working model matrix in the mono-GAM fitting procedure, mono-GAM fits slower than GAM and QP (see Figures 7-11 and 7-12). For the Poisson models the time for GAM was higher for QP which was unexpected, since the QP procedure fits an unconstrained GAM, first, for $\boldsymbol{\lambda}$ estimation. This happened because the default faster Newton method was applied when implementing QP, while a slower BFGS approach was used for GAM.

To summarize, the simulation studies show that the new method to monotone smoothing may have practical advantage over the alternative methods considered. It is slower than unconstrained GAM and quadratic programming approaches. However, unconstrained GAM may not reflect monotonicity, while smoothing parameter selection for mono-GAM is well founded, in contrast to the ad hoc method of choosing λ from an unconstrained fit, and then refitting subject to constraint used with QP. Finally, the practical MSE performance of mono-GAM seems to be better than the alternatives.

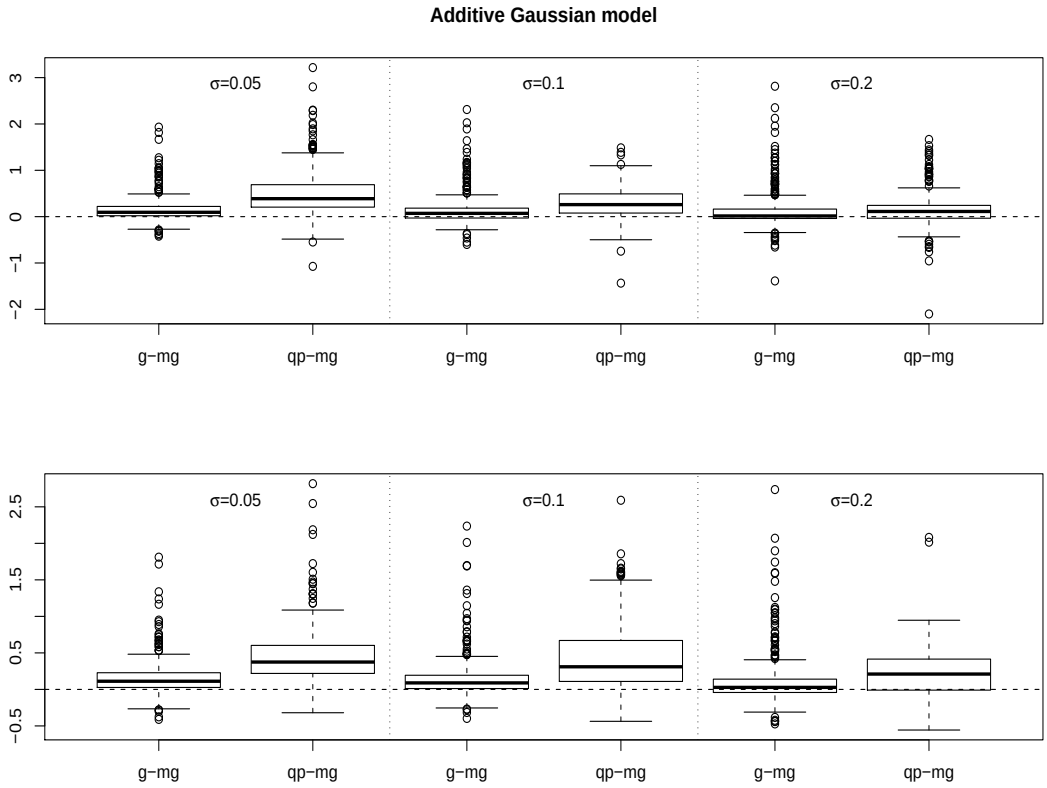


Figure 7-9: MSE comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Gaussian distribution for each of three noise levels. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of differences in relative MSE between each alternative method and mono-GAM. 300 replicates were used. Relative MSE was calculated by dividing the MSE value by the average MSE of mono-GAM for the given case. (Section 7.3)

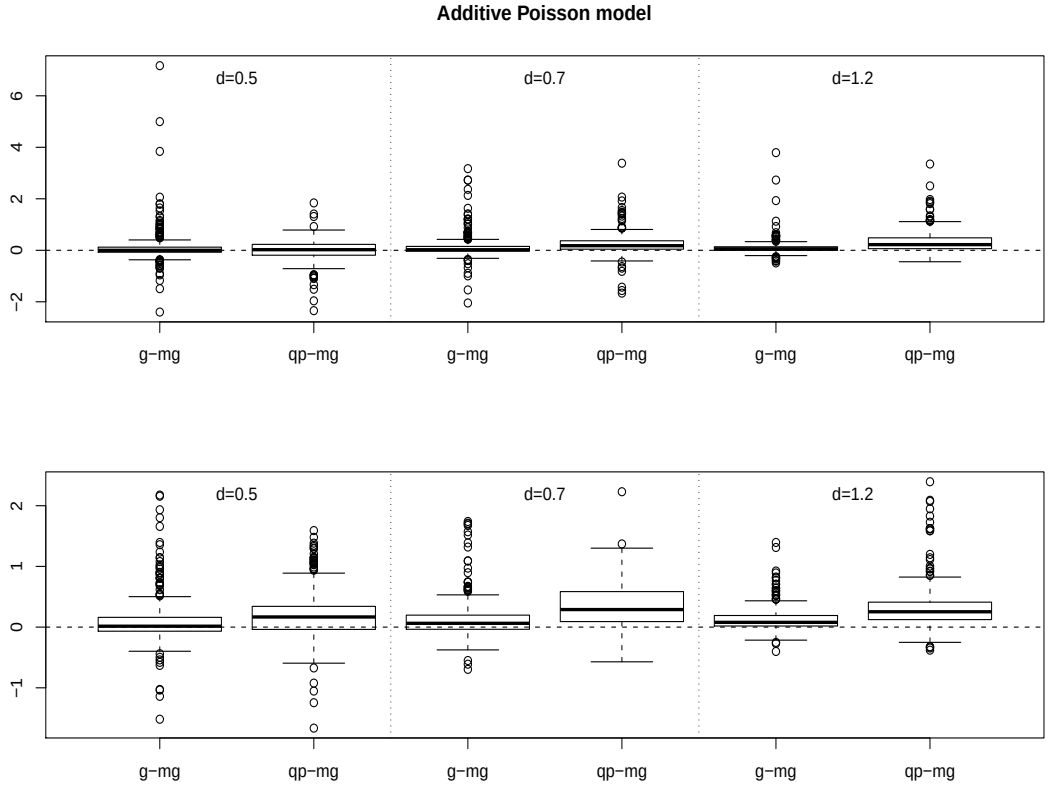


Figure 7-10: PDL comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Poisson distribution for each of three noise levels. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of differences in relative PDL between each alternative method and mono-GAM. 300 replicates were used. Relative PDL was calculated by dividing the PDL value by the average PDL of mono-GAM for the given case. (Section 7.3)

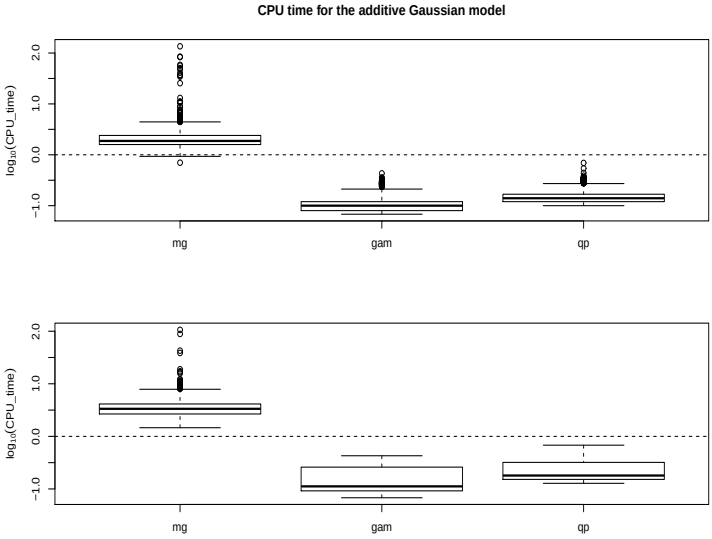


Figure 7-11: CPU time comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Gaussian additive data. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of $\log_{10}$ CPU time in seconds for thee noise levels combined. (Section 7.3)

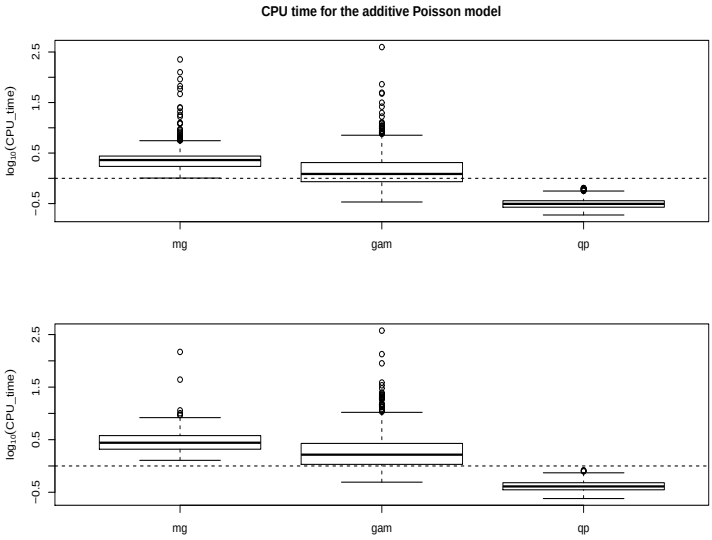


Figure 7-12: CPU time comparisons between mono-GAM (mg), GAM (g), and quadratic programming (qp) approaches for the Poisson additive data. The upper panel illustrates the results for $n = 100$, the lower for $n = 200$. Boxplots show the distributions of $\log_{10}$ CPU time per replicate in seconds for three noise levels combined. (Section 7.3)

Chapter 8

Application to real data

This chapter presents three applications of mono-GAM to different data sets. The purpose of the first application is to investigate whether proximity to municipal incinerators in Great Britain increases the risk of stomach cancer (Shaddick et al., 2007). It is supposed that the risk of cancer is a decreasing function of distance from an incinerator. The second application uses data from the National Morbidity, Mortality, and Air Pollution Study (Peng and Welty, 2004). The relationship between daily death rate in Chicago and air pollution levels is investigated. Modelling these data assumes that death rate increases with increase in levels of ozone, sulphur dioxide, and levels of particular matter. The third example studies a prediction of tree height as a function of tree diameter and additional tree-stand-level parameters. The large sample of analyzed cross-sectional data are from the Northwest German Forest Research Institute, Department of Forest Growth, Göttingen, Germany. The proposed height-diameter model includes strictly parametric model components and both monotonic and unconstrained smooth terms.

8.1 Incinerator data

In this section the mono-GAM is illustrated with an application to modelling cancer risk around municipal solid waste incinerators in Great Britain. The first large-scale study to investigate whether proximity to incinerators is associated with an increased risk of cancer was presented in Elliott et al. (1996). It analyzed data from 72 municipal solid waste incinerators in Great Britain. Decline in risk with distance from polluting source for a number of cancers was investigated. There was significant evidence for such a decline for stomach cancer, among several others. Diggle et al. (1997) reanalyzed the data on cancer of the stomach for three of those 72 incinerators which were selected

from the previous study to give a range of results. A parametric modelling approach based on a point process which assumed independence of the response variables was described in that paper. The assumption of independence of response variables might not always be credible especially for spatial data from close areas. A Poisson regression model that uses a latent process to incorporate correlation between response variables was proposed in Shaddick et al. (2007). In this approach a gamma distribution is used for the latent variable, instead of the more common log-normal distribution. Data from a single incinerator from those 72 sources, located in the northeast of England, were considered in this paper. This incinerator had a significant result for a test on monotone decreasing risk with distance from the polluting source (Elliott et al., 1996). The same data are analyzed using the mono-GAM approach in this section.

The data are from 44 enumeration districts (census-defined administrative areas) whose centroids lay within 7.5 km of the incinerator. The response variables, Y_i, $i = 1, \ldots, 44$, are observed numbers of cases of stomach cancer for each enumeration district. Estimates of the expected number of cases, E_i, are also available for risk determination, $\texttt{risk}_i = Y_i/E_i$. The expected numbers were obtained using national rates for the whole of Great Britain, standardized for age and sex. The two covariates are a distance (km) from the incinerator and a deprivation score, the Carstairs score. Deprivation scores were calculated based on three socio-economic variables: unemployment, overcrowding, and social class of head of household. These three variables were first standardized to have zero mean and unit variance across Great Britain. Then a z-score for each of the three variables was determined for each ED. The ED deprivation score was a sum of the three z-scores.

Under the model, it is assumed that, Y_i are independent Poisson variables, $Y_i \sim \text{Poi}(\mu_i)$, where $\mu_i = \lambda_i E_i$, μ_i is the rate of the Poisson distribution with E_i the expected number of cases (in area i) and λ_i the relative risk.

Shaddick et al. (2007) proposed a model under which the effect of a covariate, e.g., $\texttt{distance}_i$, on cancer risk was linear through an exponential function, i.e.

$$\lambda_i = \exp(\beta_0 + \beta_1 \texttt{distance}_i).$$

Since the risk of cancer is supposed to decrease with the distance from the incinerator, in this report a smooth monotone decreasing function, $f(\texttt{distance}_i)$, is suggested for modelling its relationship with the distance

$$\lambda_i = \exp\left\{f(\texttt{distance}_i)\right\}.$$

Hence, the model can be represented as the following:

$$\log(\lambda_i) = f(\texttt{distance}_i) \;\Rightarrow\; \log\left(\frac{\mu_i}{E_i}\right) = f(\texttt{distance}_i) \;\Rightarrow$$

$$\log(\mu_i) = \log(E_i) + f(\texttt{distance}_i),$$

which is a single smooth generalized Poisson regression model under monotonicity constraint, where $\log(E_i)$ is treated as an offset (a variable with a model parameter equal to 1). Therefore, the mono-GAM approach can be applied to fit such a model. Carstairs score is known to be a good predictor of cancer rates (Elliott et al., 1996; Shaddick et al., 2007), so its effect can also be included into the modelling of cancer risk. The following four models have been considered for this application:

Model 1: $\log\{\mathrm{E}(Y_i)\} = \log(E_i) + f_1(\texttt{distance}_i)$,

where $f_1(\texttt{distance}_i)$ is subject to monotone decreasing constraint.

Model 2: $\log\{\mathrm{E}(Y_i)\} = \log(E_i) + f_2(\texttt{Carstairs}_i)$,

where $f_2(\texttt{Carstairs}_i)$ is expected to be monotone increasing.

Model 3: $\log\{\mathrm{E}(Y_i)\} = \log(E_i) + f_1(\texttt{distance}_i) + f_2(\texttt{Carstairs}_i)$.

Model 4: $\log\{\mathrm{E}(Y_i)\} = \log(E_i) + f_3(-\texttt{distance}_i, \texttt{Carstairs}_i)$,

where a bivariate function $f_3(-\texttt{distance}_i, \texttt{Carstairs}_i)$ is subject to double monotone increasing constraint.

The following code shows fitting of model 1 by `monogam`:

```
> d1 <- monogam(y~ offset(log(E))+ s(dist,k=15,bs="mpd",m=2),
+         family=poisson(link="log"),data=data)
> d1

Family: poisson
Link function: log

Formula:
y ~ offset(log(E)) + s(dist, k = 15, bs = "mpd", m = 2)

Estimated degrees of freedom:
2.2756  total = 3.275602
```

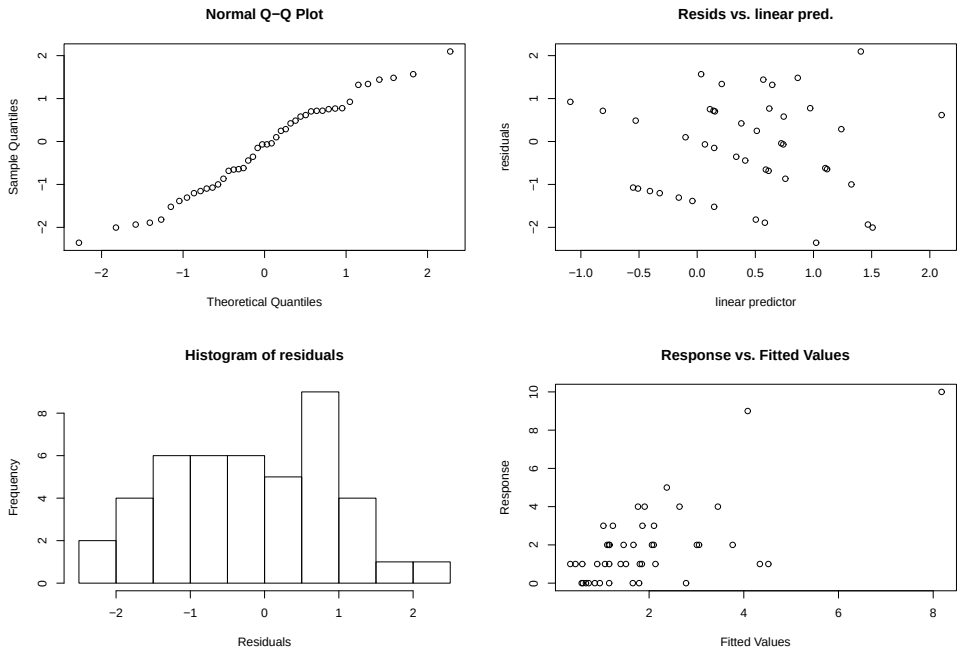


Figure 8-1: Diagnostic plots for model d1 (incinerator data).

```
UBRE score: 0.4026083
```

The data were supplied as a list `data`. As the smooth term was assumed to be monotone decreasing it had been represented using a monotone decreasing P-spline (`"mpd"`). The checking plots are given in Figure 8-1. Taking into account that for this application there were only 44 data available, the diagnostic plots suggest that the model assumptions were not obviously wrong.

The first model for comparison has been also fitted without constraint.

```
> d1_gam <- gam(y~ offset(log(E))+ s(dist,k=15,bs="cr",m=2),
+         family=poisson(link="log"),data=data,method="REML")
> d1_gam

Family: poisson
Link function: log

Formula:
y ~ offset(log(E)) + s(dist, k = 15, bs = "cr", m = 2)
```

```
Estimated degrees of freedom:
4.4344  total = 5.434434

REML score: 77.85201
```

The smoothing parameter of the unconstrained model was estimated by restricted maximum likelihood (REML) (Wood, 2011) since initial estimation by the UBRE used for mono-GAM resulted in an overfitted smooth. The estimated smooths and risk functions for both methods are illustrated in Figure 8-2. The estimate of the cancer risk function was obtained by

$$\hat{\texttt{risk}}_i = \frac{\hat{\mu}_i}{E_i} = \exp\left\{\hat{f}_1(\texttt{distance}_i)\right\}.$$

Note, that REML also resulted in a non-monotone smooth, which supports the mono-GAM approach. The values of the adjusted r^2 and percentage deviance explained (see Section 6.4) for the GAM were less than for mono-GAM.

Model 2 describes the number of cases of stomach cancer through a smooth function of deprivation score. This function is assumed to be monotone increasing since it was shown (Elliott et al., 1996) that poor people (low Carstairs score) live closer to incinerators. Model 2 can be fitted as the following:

```
d2 <- monogam(y~ offset(log(E)) + s(Carstairs,k=15,bs="mpi",m=2),
          family=poisson,data=data)
```

The estimated results are shown in Figure 8-3. The UBRE value for this model was 0.5166, which is higher than for the previous model. The other two measures of the model performance, the adjusted r^2 and the deviance explained, also gave slightly worse results for model 2:

$$r^2_{\text{d1}} = 0.411, \;\; r^2_{\text{d2}} = 0.347,$$

$$\text{Deviance explained}_{d1} = 37.5\%, \text{Deviance explained}_{d2} = 31.8\%.$$

The third model incorporates both covariates, `dist` and `Carstairs`, assuming their additive effect. The next fits and prints some results of model 3.

```
d3 <- monogam(y~ offset(log(E)) + s(dist,k=15,bs="mpd",m=2)+
        s(Carstairs,k=15,bs="mpi",m=2),family=poisson,data=data)
```

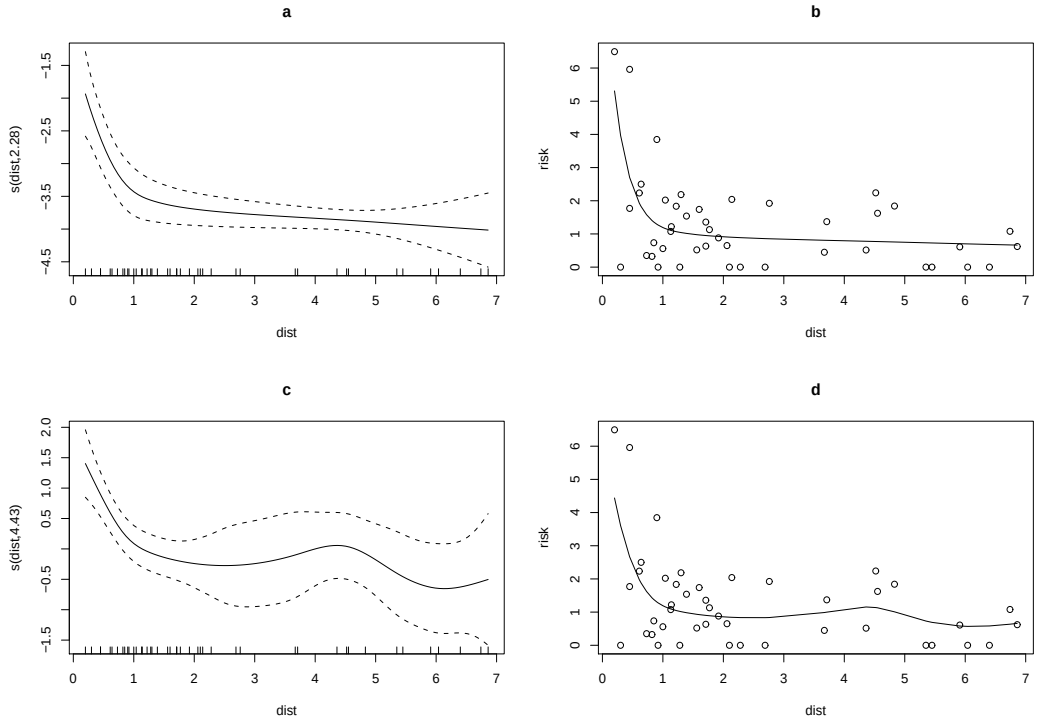


Figure 8-2: The estimated smooth and cancer risk function for monotone and unconstrained versions of model d1 (incinerator data). (a) - the estimated smooth of mono-GAM + 95% confidence interval; (b) - the mono-GAM estimated risk as the function of distance; (c) - the GAM estimated smooth + 95% confidence interval; (d) - the GAM estimated risk as the function of distance. Points show the observed data.

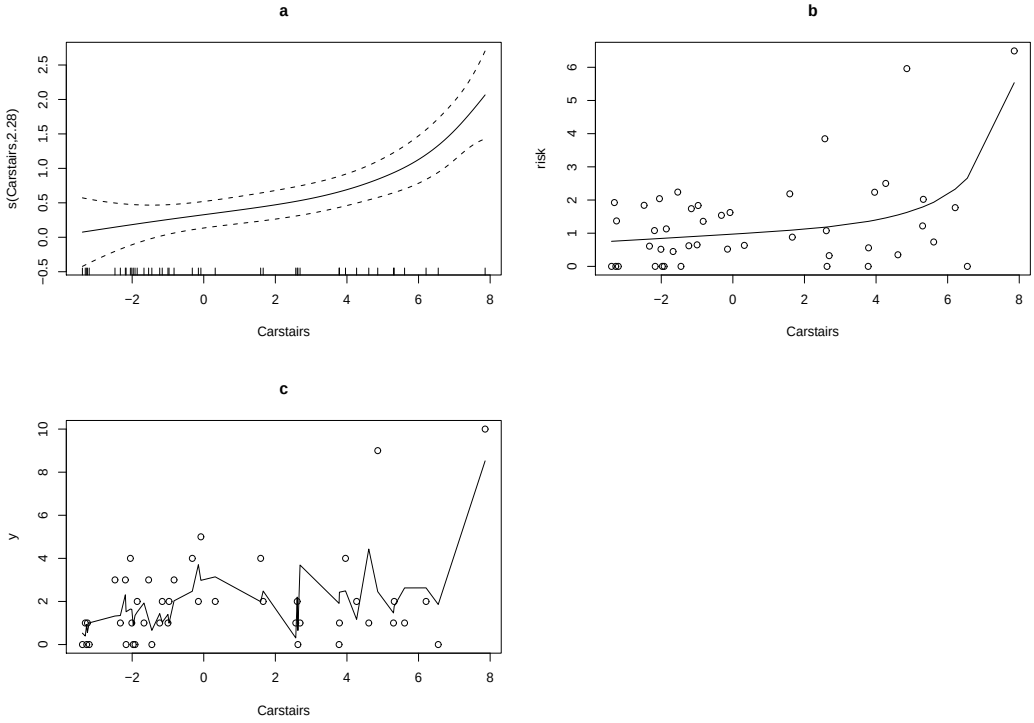


Figure 8-3: The estimated smooth, risk, and fitted curve of model d2 (incinerator data). (a) - the estimated smooth + 95% confidence interval; (b) - the estimated risk as the increasing function of Carstairs; (c) - the overall fitted curve, $\hat{\mu}$. Points show the observed data.

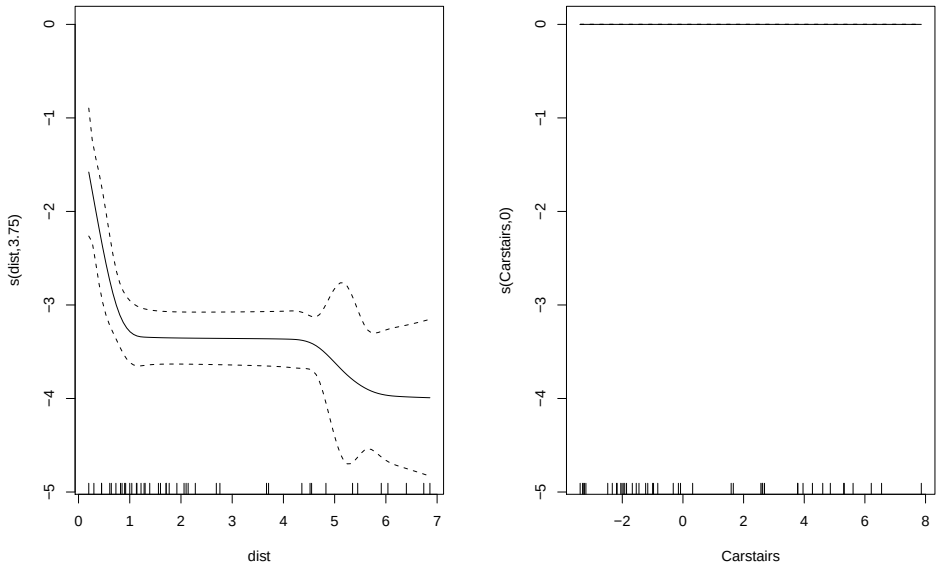


Figure 8-4: The estimated smooths of model d3 (incinerator data).

The estimated effective degrees of freedom of $f_2(\texttt{Carstairs})$ was about zero. This smoothing term was insignificant in this model, with all its coefficients near zero (see Figure 8-4). This can be explained by a high correlation between two covariates, $(\text{corr}(\texttt{distance}, \texttt{Carstairs}) = -0.723)$.

Considering a linear effect of Carstairs in place of the smooth function, f_2, as it was proposed in Shaddick et al. (2007),

$$\log\{\mathrm{E}(Y_i)\} = \log(E_i) + f_1(\texttt{distance}_i) + \beta \cdot \texttt{Carstairs}_i,$$

also resulted in an insignificant value for β.

The last model considers a bivariate function, $f_3(-\texttt{distance}_i, \texttt{Carstairs}_i)$. In order to impose double monotonicity with decrease for `distance` and increase for `Carstairs`, the first covariate of distance was taken with the negative sign. After such a transformation, a double monotone increasing constraint can be used via the smoothing basis `"tedmi"` of the `monogam` library (see Section 6.1).

```
> y<- data$y; dist<- -data$dist; Carstairs<- data$Carstairs; E<- data$E
> d4 <- monogam(y~offset(log(E))+s(dist,Carstairs,k=c(6,6),bs="tedmi"),
+          family=poisson)
```

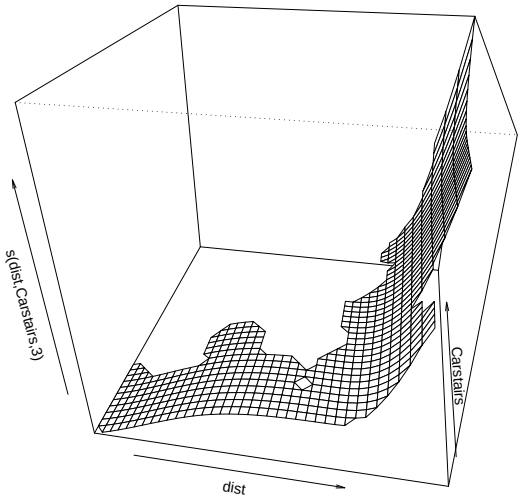


Figure 8-5: Perspective plot of the estimated bivariate smooths of model d4 (incinerator data).

The perspective plot of the estimated smooth is illustrated in Figure 8-5. This plot also supports the previous result. The Carstairs score does not provide any additional information for modelling cancer risk when the explanatory variable is the distance. The graph of the estimated smooth has almost no increasing trend with respect to the second covariate. The measures of the model performance, such as UBRE, adjusted r^2, and the deviance explained are slightly better for the first model with only the effect of distance included than for the bivariate model.

Comparing all four suggested models we may conclude that it is sufficient to consider only distance (model d1) as a predictor for the risk of stomach cancer. Moreover, there is evidence for decrease in risk of cancer of stomach with distance from the incinerator.

8.2 Air pollution data

The second application is concerned with an air pollution study which investigates the relationship between non-accidental daily mortality and air pollution. In Wood (2006a) such an analysis is discussed which uses Poisson additive models. The analyzed

air pollution data were from the National Morbidity, Mortality, and Air Pollution Study (Peng and Welty, 2004) which contains 5114 daily measurements on different variables for 108 United States cities. As an example a single city (Chicago) study was examined in Wood (2006a). The response variable was the daily death rate in Chicago (`death`) for the years 1987 – 1994. Four explanatory variables were considered: average daily temperature (`tempd`), levels of ozone (`o3median`), levels of particulate matter (`pm10median`), and `time`. Since it might be expected that mortality increases with increase in air pollution levels, modelling with mono-GAM may be useful. This section shows application of mono-GAM to the same Chicago air pollution data.

The preliminary modelling and examination of the data showed (Wood, 2006a) that the mortality rate at a given day could be better predicted if the aggregated air pollution levels and aggregated mean temperature were incorporated into the model, rather than levels of pollution and temperature on the day in question. For aggregating it was proposed to use the sum of each covariate except time, over the current day and three preceding days. Hence, three aggregated predictors are as follows

$$\texttt{tmp}_i = \sum_{j=i-3}^{i} \texttt{tempd}_j, \;\; \texttt{o3}_i = \sum_{j=i-3}^{i} \texttt{o3median}_j, \;\; \texttt{pm10}_i = \sum_{j=i-3}^{i} \texttt{pm10median}_j.$$

Assuming that the observed numbers of daily death are independent Poisson random variables, the following additive model structure can be considered

$$\log\{\mathrm{E}(\texttt{death}_i)\} = f_1(\texttt{time}_i) + f_2(\texttt{pm10}_i) + f_3(\texttt{o3}_i) + f_4(\texttt{tmp}_i), \tag{8.1}$$

where $f_1 - f_4$ are smooth functions and additional monotone increasing constraints are assumed on f_2 and f_3, since increase in air pollution levels is expected to increase mortality rate. The following code fits this model and prints fitting results.

```
> p1 <- monogam(death ~ s(time,bs="cr",k=200)+s(pm10,bs="mpi",k=20)+
+            s(o3,bs="mpi",k=20)+s(tmp,bs="cr",k=20),family=poisson)
> p1

Family: poisson
Link function: log

Formula:
death ~ s(time, bs = "cr", k = 200) + s(pm10, bs = "mpi", k = 20) +
    s(o3, bs = "mpi", k = 20) + s(tmp, bs = "cr", k = 20)
```

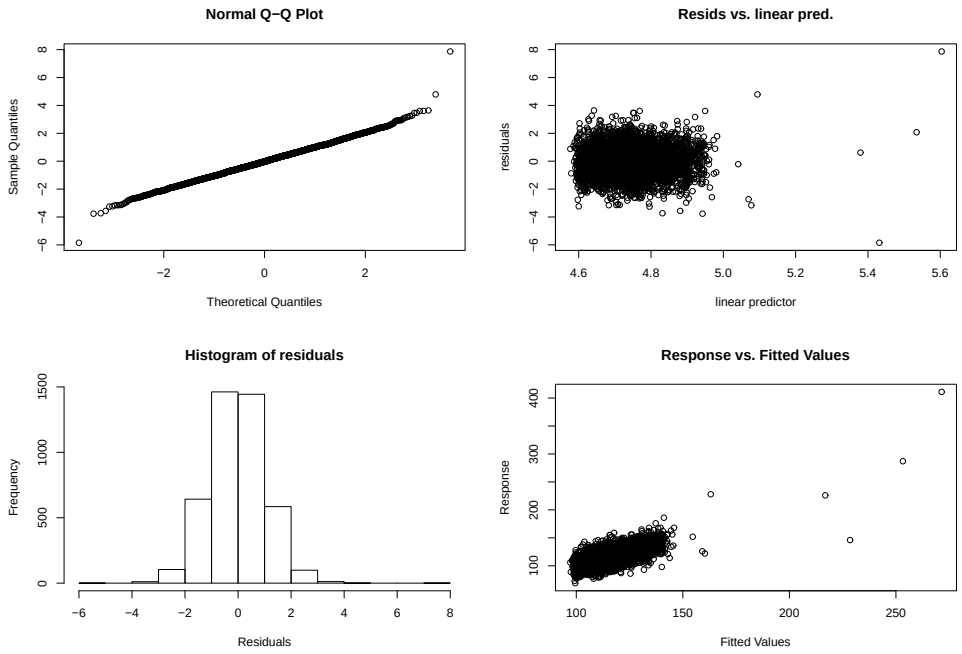


Figure 8-6: Diagnostic plots for model p1 (air pollution data).

```
Estimated degrees of freedom:
145.4226   6.7548   1.0409  18.0328  total = 172.2510

UBRE score: 0.1629705
```

Cubic regression splines have been used for unconstrained smooth terms. By default, the smoothing parameters have been selected using UBRE score. The checking plots are illustrated in Figure 8-6. There is no clear evidence to conclude that the model specification is wrong. Figure 8-7 shows the plots of the smooth estimates.

Though the effect of the ozone level is only with one degree of freedom, it is positive and increasing (the bottom left plot of Figure 8-7). The rapid increase in the smooth of aggregated mean temperature (the bottom right plot) can be explained by four highest daily death rates occurred on four consecutive days of very high temperature and also high level of ozone. 200 data of that period plotted by calling

```
> plot(tmp[3000:3200],death[3000:3200])
```

are shown in Figure 8-8. Note, there are four outliers on the right side of this plot corresponding to the highest death rates.

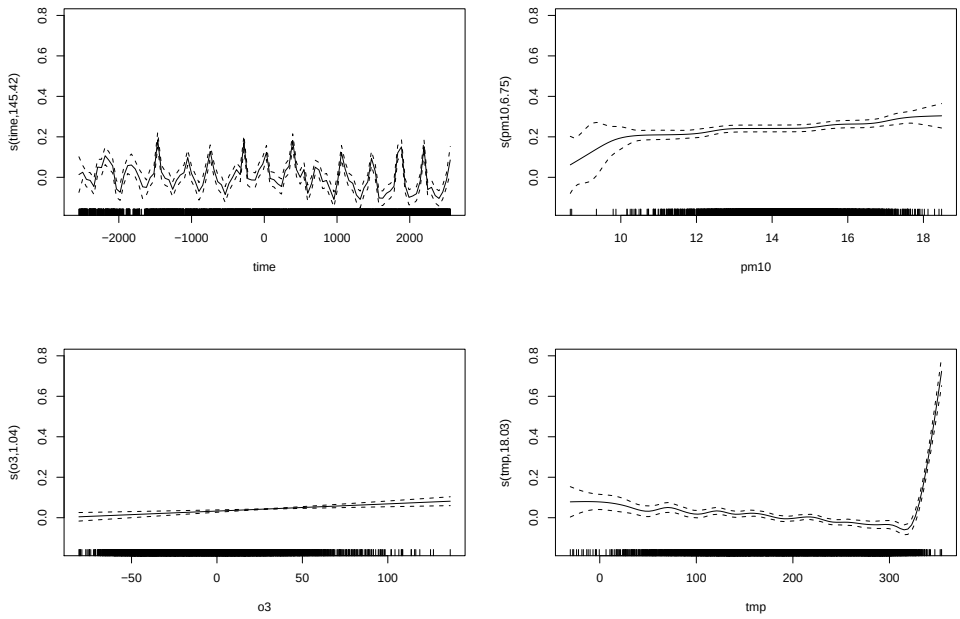


Figure 8-7: The estimates of the smooth terms of model p1 (air pollution data).

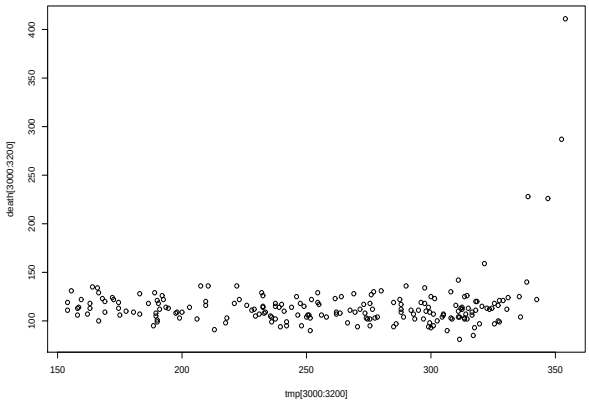


Figure 8-8: Plot of the observed combinations of daily death rate and aggregated temperature for the period of the highest death rate.

Since high temperature together with high level of ozone might result in very high daily death rates, it is suggested to consider a bivariate smooth of these predictors. The following model is considered as the second alternative

$$\log\{\mathrm{E}(\texttt{death}_i)\} = f_1(\texttt{time}_i) + f_2(\texttt{pm10}_i) + f_3(\texttt{o3}_i, \texttt{tmp}_i), \tag{8.2}$$

where $f_2(\texttt{pm10}_i)$ is a monotone increasing function and $f_3(\texttt{o3}_i, \texttt{tmp}_i)$ is subject to single monotonicity along the first covariate. A tensor product spline with single monotonicity, `"tesmi1"`, (Section 3.3.2) can be used for f_3 representation. The following fits the model and checks it.

```
p2 <- monogam(death ~ s(time,bs="cr",k=200)+s(pm10,bs="mpi",k=10)+
        s(o3,tmp,k=c(10,10),bs="tesmi1",family=poisson)
monogam.check(p2_1)
```

Figure 8-9 shows the diagnostic plots of this model. It may be noted that there is a slight improvement in comparison to Figure 8-6. The estimates of the univariate smooths of model `p2` are illustrated in Figure 8-10. The perspective plot of the estimated bivariate smooth is shown in Figure 8-11. The second model also has a lower UBRE score (UBRE (`p1`) = 0.16297, UBRE (`p2`) = 0.14134) which implies that `p2` is a preferable model.

The current approach has been applied to air pollution data for Chicago only. It would be of interest to apply the same model to other cities, to see whether the relationship between non-accidental mortality and air pollution can be described by the proposed mono-GAM in other locations.

8.3 Forest data

8.3.1 Motivation

Two of the main questions of forest management planning are what is the current status of forests and how forests will develop in future. To estimate forest stock and assortment for, for example, forest districts or federal states, a tree volume has to be predicted and then summed up to get timber volume estimates for a considered forest area. A tree volume evaluation is based on two tree parameters: tree height and tree diameter. Since measuring tree diameter at breast height (1.3 m), is relatively cheap, but measuring tree heights is cost intensive, it is desirable to model tree height as a function of tree diameter and some other tree-specific parameters. An important feature of the height-diameter relationship is that it develops over time and varies from

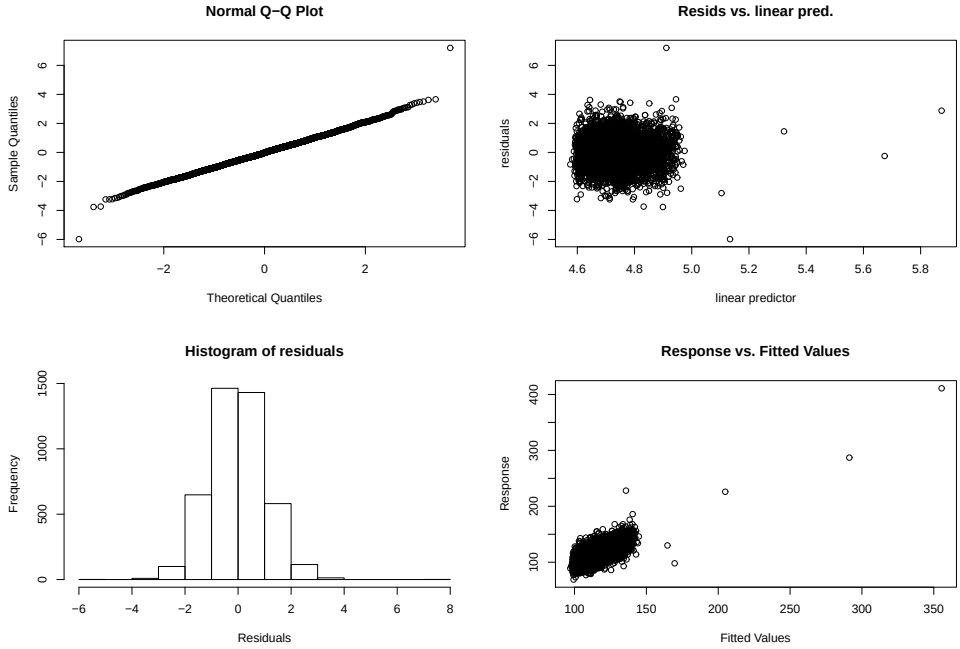


Figure 8-9: Diagnostic plots for model p2 (air pollution data).

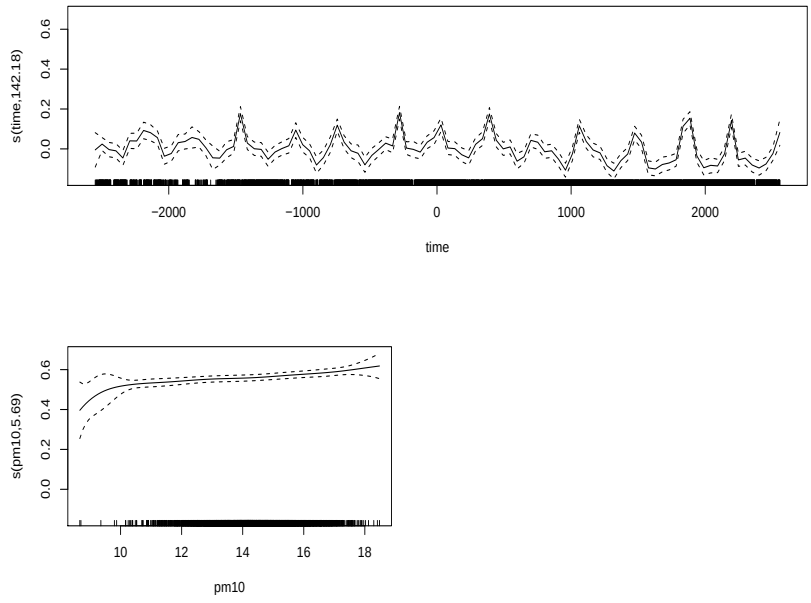


Figure 8-10: The estimates of the univariate smooth terms of model p2 (air pollution data).

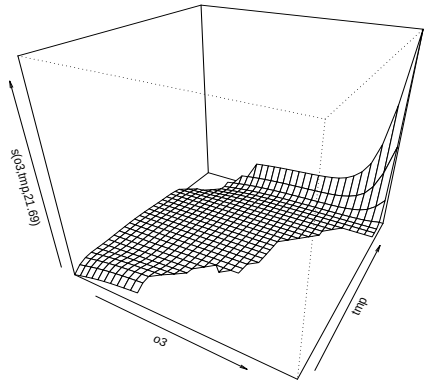


Figure 8-11: Perspective plot of the estimated bivariate smooth terms of model p2 (air pollution data).

stand to stand (Lappi, 1997; Mehtätalo, 2004) (stand is a group of growing trees in a given area). In Mehtätalo (2005) it is noted that trees reach maturity at different ages which depend on site conditions. It takes longer for trees to grow from samplings to a mature stand on poor sites than on rich sites.

The study discussed in this section has been conducted by Dr. Matthias Schmidt at the Northwest German Forest Research Institute, Department of Forest Growth, Göttingen, Germany. There are several goals of the study. The first one is to develop site-sensitive prediction of tree height for the current status of forests in Germany. Also the model has to be valid for all regions of Germany taking into account the spatial variability of the height-diameter relationship. Since climate change has already affected forests in Central Europe and a heavier impact is anticipated in the future, the model should be applicable for prediction of future tree height development and be able to quantify the impact of climate change on height development and tree growth.

Many studies of forest research have been devoted to modelling height-diameter relationship (see, e.g., Jayaraman and Lappi, 2001; Eerikäinen, 2003; Mehtätalo, 2004). Several approaches are available now for height predictions. In this study a random parameter height-diameter model developed by Lappi (1997) is used as a starting model. It is then extended to include some tree-specific and site-specific variables in the fixed parts of the model. As some of the fixed effects are supposed to be monotone, a mono-GAM approach can be applied in the middle stage of the complete model development.

8.3.2 Data

The data analyzed here are observations of 29 324 trees and some site-specific variables from the forest enterprise (district) inventories conducted in Lower-Saxony. Lower Saxony is the second largest federal state of Germany and is located in the north-western part. Several reforms have been conducted so that the number of forest districts has decreased over time. Every year two or three state owned forest districts are inventoried. The data come from inventories in the time interval 1996 – 2008. There are almost no consecutive inventories (no longitudinal data), but all forest districts are inventoried, with the exception of a small area of the "Nationalpark Harz". The data are a subset of a larger database for the whole of Germany. This larger data base consists of 199 915 single tree records, assessed by the Northwest German Forest Research Station.

The surveyed trees are Norway spruce, the most common species in Europe. The response variable is a tree height measured in meters. Two types of covariates are considered: tree-specific and site-specific. The tree specific variables are tree diameter at breast height (`dbh`) measured in centimeters, tree age, year of germination, altitude, topex indices, northing, and easting. The year of germination variable serves as a proxy for the effect of observed growth trends in Europe, that are supposed to be mainly a result of increased nitrogen emission into the soil since 1950s. Also the mean quadratic diameter ($\mathtt{d}_g$) of a stand is used as a covariate describing the hierarchic position of a tree of a certain diameter within the stand. It is defined as the following

$$\mathtt{d}_g = 2\sqrt{\frac{1}{\pi n}\sum_{i=1}^{n}\left(\frac{\pi}{4}\cdot \mathtt{dbh}_i^2\right)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\mathtt{dbh}_i^2},$$

where n is the number of inventoried trees in the stand.

Topex index describes topographic exposure in the South-West direction and terrain morphology. A digital terrain model (DTM) with a resolution of 90 by 90 m was used for topex calculation. If a tree is located on a summit it is highly exposed and its topex index is negative, positive topex means that the location is in a valley, flat areas result in topex values near zero (see, e.g., Scott and Mitchell, 2005). Since exposure to the south west might result in drought stress, the topex index is used as a proxy for drought stress.

The additional site-specific (climate) explanatory variables are temperature summed over days of the vegetation period (growing season), and De Martonne's aridity index. An aridity index is a fraction of annual precipitation in millimeters over mean annual temperature in degrees Centigrade minus ten (P/T-10) (Thornthwaite, 1931). The

advantage of the site-specific covariates, compared to proxies is that they give an opportunity to assess the climate change impact on tree growth. Table 8.1 summarizes the data under study.

Table 8.1: Forest data characteristics

	Min	Median	Max
Tree height (m)	3.7	21.8	47.3
dbh (cm)	7	30.45	104
Tree age (years)	20	54	199
Year of germination (Y)	1803	1948	1988
Altitude (m)	0	307	947
Topographic exposure on South-West direction (DTM 90 x 90)	-84560	1489	89208
Temperature sum over the vegetation period	833.55	1996.59	2456.80
Aridity index	24.802	44.740	87.463

8.3.3 Modelling approach

This section describes an approach to modelling the longitudinal height-diameter (h-d) relationship. A difficulty with the h-d relationship is that it is not stable but varies from stand to stand and develops over time (Lappi, 1997; Mehtätalo, 2004). The development of the h-d model consists of several steps. The following gives a brief description of the model development.

1. Initial specification of the height-diameter model as an exponential model with random parameters. Decomposition of the random parameters of the model into fixed effects, a random stand effect, and a random time effect. 'A priori' determination of non-linear model parameters.

2. Estimation of the age effect only as a part of the fixed effect (using a subsample that originated from one measurement point in time). Estimation of other site-specific and/or climate-specific effects on the trend. All one-dimensional effects are assumed to be monotonic smooth functions.

3. Inclusion of the fixed effects into the complete model and its re-parametrization as a linear mixed model.

The preliminary model selection step does not require use of mono-GAM. It is based on modelling approaches to height prediction commonly used in forest management.

The first subsection briefly describes the initial steps of the model development carried out by Dr. Matthias Schmidt at the Department of Forest Growth and Yield, the Northwest German Forest Research Station, Germany. The data base for the whole of Germany was applied for this 'a priori' estimation of specific model parameters. In the second stage, several additional predictors are incorporated into the model. As some of the predictors are supposed to have monotone effects on the height-diameter relationship, a mono-GAM approach can be applied at this step. The complete model is a linear mixed model which is shown in the last subsection. At the point of writing this thesis, the full data analysis has not been completed, and the collaboration with Dr. M. Schmidt is ongoing. But since the purpose of this chapter is only to demonstrate the use of mono-GAM, this section focuses on discussion of the middle step of the model development.

Initial model development

As a starting point, the following longitudinal height-diameter model known as the Korf function is used for the description of the relationship between tree height and diameter (Lappi, 1997):

$$\log(H_{kti}) = A_{kt} - B_{kt}\left(\texttt{dbh}_{kti} + \lambda\right)^{-C_{kt}} + \epsilon_{kti}, \tag{8.3}$$

where H_{kti} is a height of tree i on sample plot k at time t, $\texttt{dbh}_{kti}$ is a diameter at breast height of tree i on sample plot k at time t; ϵ_{kti} are gaussian errors; A_{kt}, B_{kt}, λ, and C_{kt} are parameters of the model. Height-diameter curves differ for different plots and for different points of time, therefore, the model parameters vary over plots and with time. Since parameters A_{kt} and B_{kt} are highly correlated, it is suggested to reparameterize `dbh` as follows (Jayaraman and Lappi, 2001):

$$x_{kti} = \frac{\left(\texttt{dbh}_{kti} + \lambda\right)^{-C_{kt}} - (30+\lambda)^{-C_{kt}}}{(10+\lambda)^{-C_{kt}} - (30+\lambda)^{-C_{kt}}}.$$

The model (8.3) can now be written as

$$\log(H_{kti}) = A_{kt} - B_{kt}x_{kti} + \epsilon_{kti}, \tag{8.4}$$

where A_{kt} and B_{kt} are not highly correlated and have biological meanings. A_{kt} is the expected value of the log height of trees with diameter 30 cm for sample plot k at time t; and B_{kt} is the expected value of the difference in the $\ln(H_{kti})$ between trees of diameters 30 cm and 10 cm for sample plot k at time t. These interpretations are very important since the parameters will be described as functions of additional tree and

stand-level covariates in the second step of the model development.

The model (8.4) is linear with respect to A_{kt} and B_{kt}. Taking into consideration the random stand effect and random time effect within stands these parameters can be represented at the first stage as

$$A_{kt} = A + \alpha_k + \alpha_{kt}, \;\; B_{kt} = B + \beta_k + \beta_{kt},$$

where A and B represent fixed effects which have to be estimated; α_k and β_k are random stand level effects with zero means and constant variance; and α_{kt} and β_{kt} are random time effects within stands with zero means but possibly not constant variance.

It may be noted that (8.4) is overparameterized. Moreover, a model of that specification cannot be linearized with respect to the parameters λ and C_{kt}. Therefore, it is suggested firstly to estimate λ and C_{kt}. These parameters were selected by testing a variety of combinations of λ and C when fitting a linear mixed model

$$\log(H_{kti}) = A + Bx_{kti} + \alpha_k + \alpha_{kt} - (\beta_k + \beta_{kt})x_{kti} + \epsilon_{kti}, \;\; \mathbf{b} \sim N(\mathbf{0}, \mathbf{D}),$$

where $\mathbf{b}$ is a vector of random stand and time effects. R library `nlme` was used for this procedure. The combination of the parameters with the lowest error variance was $\lambda = 7$ and $C = 1.225$. There were no clear trends found in λ and C over different mean stand age and the models were not very sensitive to the value C.

The next step is to estimate the fixed effects of the h-d model, A and B. Since the model should be used not only for height estimation for a given point of time but also for predicting future height development, first, A and B are described as functions of tree age. Tree age is highly correlated with year of germination and their effects on tree height cannot be separated feasibly. That is why for an 'a priori' estimation of the age effect on the intercept A, only tree age was integrated into the model as a covariate. The age effect on the slope B was assumed to be linear (Lappi, 1997; Mehtätalo, 2004). For estimating the age effect, a subsample of 98 274 tree records of the same year of inventory, 1987, was analyzed. Because of the large data set it was performed only on a computer with 32 Gigabytes memory and no random time effect could be incorporated. The following additive mixed model was fitted using `gamm(mgcv)`:

$$\log(H_{kti}) = f_{1a}(\texttt{Age}_{kti}) - p_{ob}x_{kti} - p_{1b} \cdot \texttt{Age}_{kti} \cdot x_{ki} + \alpha_k - \beta_k x_{kti} + \epsilon_{kti}.$$

The resulted age effect was not monotone increasing as it should be, so it was approximated by a parametric function using parameters of the Chapman-Richards

function (Richards, 1959)

$$f_{1a}(\texttt{Age}_{kti}) = p_{1a} + p_{2a}\left(1 - e^{-p_{3a}\texttt{Age}_{kt}}\right)^{p_{4a}}, \tag{8.5}$$

with only the parameter p_{2a} to be refitted later, when including several other predictors (effects) and using the whole data base.

One of the model requirements is to predict tree heights of a forest stand. Since every stand has different characteristics, additional measured stand variables should be incorporated into the h-d model. These additional covariates are assumed to have fixed additive effects on the model together with the estimated age effect. The next subsection describes different ways of estimating several other tree-specific, and also climate-specific, effects on the trend of the h-d curve using a mono-GAM approach.

Estimating non-linear model effects using mono-GAM

After estimating the age effect on the h-d curve, additional fixed effects can be estimated such as altitude, year of germination, topographic exposure, and others. The following model is considered:

$$\begin{aligned}\log(H_{kti}) = \hat{p}_{1a} + p_{2a}\left(1 - e^{-\hat{p}_{3a}\texttt{Alt}}\right)^{\hat{p}_{4a}} &+ f_{2a}(\texttt{dg.neu}) + f_{3a}(\texttt{Topex567}) + f_{4a}(\texttt{HNN}) \\ &+ f_{5a}(\texttt{keimjahr}) + f_{6a}(\texttt{Rechtswert}, \texttt{Hochwert}) - p_{0b} \cdot x_{kti} + p_{1b} \cdot \texttt{Alt} \cdot x_{kti} \\ &+ p_{2b} \cdot \texttt{HNN} \cdot x_{kti} + \epsilon_{kti},\end{aligned}$$

where the the Chapman-Richards parameters (8.5) were used as constants except of p_{2a} which has to be re-estimated. `dg.neu` denotes the mean quadratic diameter within a stand, `Alt` is a tree age, `Topex567` is a topographic exposure to South-West, `HNN` is an altitude, `keimjahr` is a year of germination, $f_{6a}(\texttt{Rechtswert}, \texttt{Hochwert})$ is a bivariate function of easting and northing. The age trend from the inventory 1987 was fixed and transferred into this model. The notation of the covariates are the same as in the data base kindly provided by the the Northwest German Forest Research Institute.

By introducing a variable $z_{kt} = \left(1 - e^{-\hat{p}_{3a}\texttt{Alt}}\right)^{\hat{p}_{4a}}$ and denoting $A_1 = \hat{p}_{1a}$ the above model may be re-written as

$$\begin{aligned}\log(H_{kti}) = A_1 + p_{2a}z_{kt} &+ f_{2a}(\texttt{dg.neu}) + f_{3a}(\texttt{Topex567}) + f_{4a}(\texttt{HNN}) + f_{5a}(\texttt{keimjahr}) \\ &+ f_{6a}(\texttt{Rechtswert}, \texttt{Hochwert}) - p_{0b} \cdot x_{kti} + p_{1b} \cdot \texttt{Alt} \cdot x_{kti} \\ &+ p_{2b} \cdot \texttt{HNN} \cdot x_{kti} + \epsilon_{kti},\end{aligned} \tag{8.6}$$

The model assumes a linear combination of the covariates effects. Since, from expert knowledge, the effects of some of the covariates are known to be monotone, the mono-GAM approach can be applied at this stage of the model development.

For estimating the non-linear model effects, three different additive models with monotonicity constraints are considered here. The first mono-GAM is the model (8.6) with monotonicity restrictions on one-dimensional smooth components. A more complicated model with variable coefficients is proposed as the second alternative. Since the purpose of the height-diameter model is not only to predict tree height for the current status of forests but also to make site (climate) sensitive prediction for the future status, several site-specific covariates are incorporated in the last mono-GAM. All three models were fitted to the data recorded in Lower-Saxony (29 324 tree records).

MODEL 1:

Model 1 is the model (8.6) with monotonicity restrictions on four one-dimensional smooth terms. The effect of `dg.neu` is expected to be monotone increasing since the larger the mean quadratic diameter compared to the diameter of a tree, the more the tree is suppressed and has to invest more into height growth than into diameter growth to struggle for the light. The effect of `Topex567` should be monotone increasing since an exposure to the south west might result in drought stress. From expert knowledge, the effect of `HNN` should be monotone decreasing since the growth conditions become worse with increasing altitude. The function of `keimjahr` should be increasing with the tree age already integrated into the model, because of increased nitrogen emission into the soil since 1950s.

Monotone P-splines with 20 basis functions were used for representation of the monotone smooths of (8.6). Since the bivariate function $f_{6a}(\texttt{Rechtswert}, \texttt{Hochwert})$ is a function of geographic co-ordinates, it was represented by a thin plate regression spline (Wood, 2006a) which is in some sense an optimal smoother when the isotropy of the wiggliness penalty (treating wiggliness in all directions equally) is a desirable feature. The following fits and checks the model:

```
> m1 <- monogam(log(h)~-1+offset(A1) + s(dg.neu,k=20,bs="mpi",m=2) +
        s(Topex567,k=20,bs="mpi",m=2) + s(HNN,k=20,bs="mpd",m=2) +
        s(keimjahr,k=20,bs="mpi",m=2)+s(Rechtswert,Hochwert)+zkt+xkti+
        Alt_xkti+HNN_xkti,data=dat.full, family=gaussian)
> monogam.check(m1)

Method: GCV   Optimizer: bfgs
Number of iterations of smoothing parameter selection performed was 5 .
```

```
Full convergence.
Gradient range: [-3.125233e-09,2.628042e-07]
(score 0.01915454 & scale 0.01911566)
```

The constant $A1 = \hat{p}_{1a}$ represents an intercept of this model to guarantee a plausible effect of the year of germination. The checking plots are shown in Figure 8-12. These diagnostic plots confirm the overall height-diameter model specification as a mixed model. When estimating the trend functions the random effects have not been taken into account, so the residual plots reveal this problem. Lappi (1997) found out that the variance var(ϵ_{kti}) had a decreasing trend, especially for trees with large diameters. He proposed an expression for the variance as a decreasing function of the diameter. On the residuals versus linear predictor plot of Figure 8-12 (an upper right plot) a clear decreasing trend in the variance can be seen.

Figure 8-13 illustrates the estimated smooth terms for model `m1` obtained by calling the `plot` function

```
plot(m1, pages=1, scale=0)
```

In the previous approach for estimating the same trend functions, first, `gam(mgcv)` was used and then monotonicity of the estimated smooths were achieved by parametric approximations. This step has now been skipped by applying the `monogam`. All the estimated smooths are in agreement with the previous results, taking into account that for the current results, only the data from Lower-Saxony were analyzed. The upper right plot of Figure 8-13 shows that the effect of `Topex567` is not very strong: that is probably because the digital terrain model used for it has a low resolution of $90 \times 90m$. The effect of the year of germination, `keimjahr`, is quite unstable for 1980 and later due to high correlation with tree age. Also the soil reached a steady state by 1980, being saturated with nitrogen.

More detailed fitting results were obtained by using the `summary` method.

```
> summary(m1)

Family: gaussian
Link function: identity

Formula:
log(h) ~ -1 + offset(A1) + s(dg_neu, k = 20, bs = "mpi", m = 2) +
```

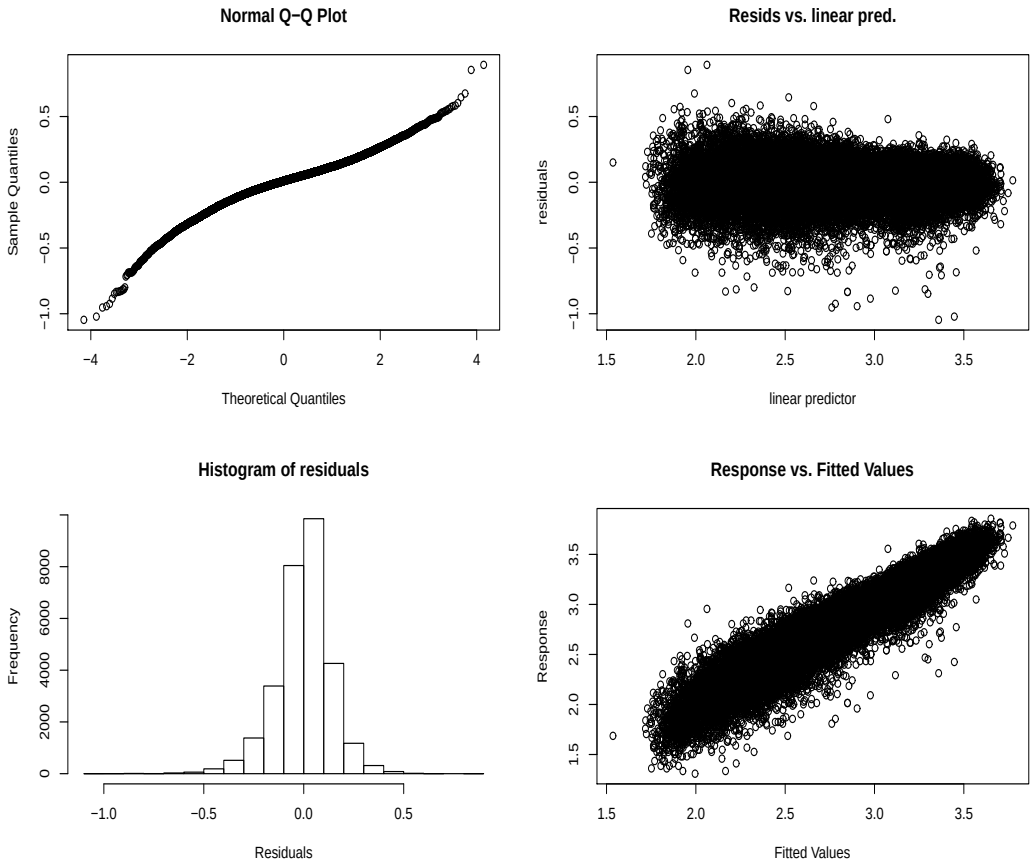


Figure 8-12: Model checking plots for model 1, note the heavier-than-Gaussian tails.

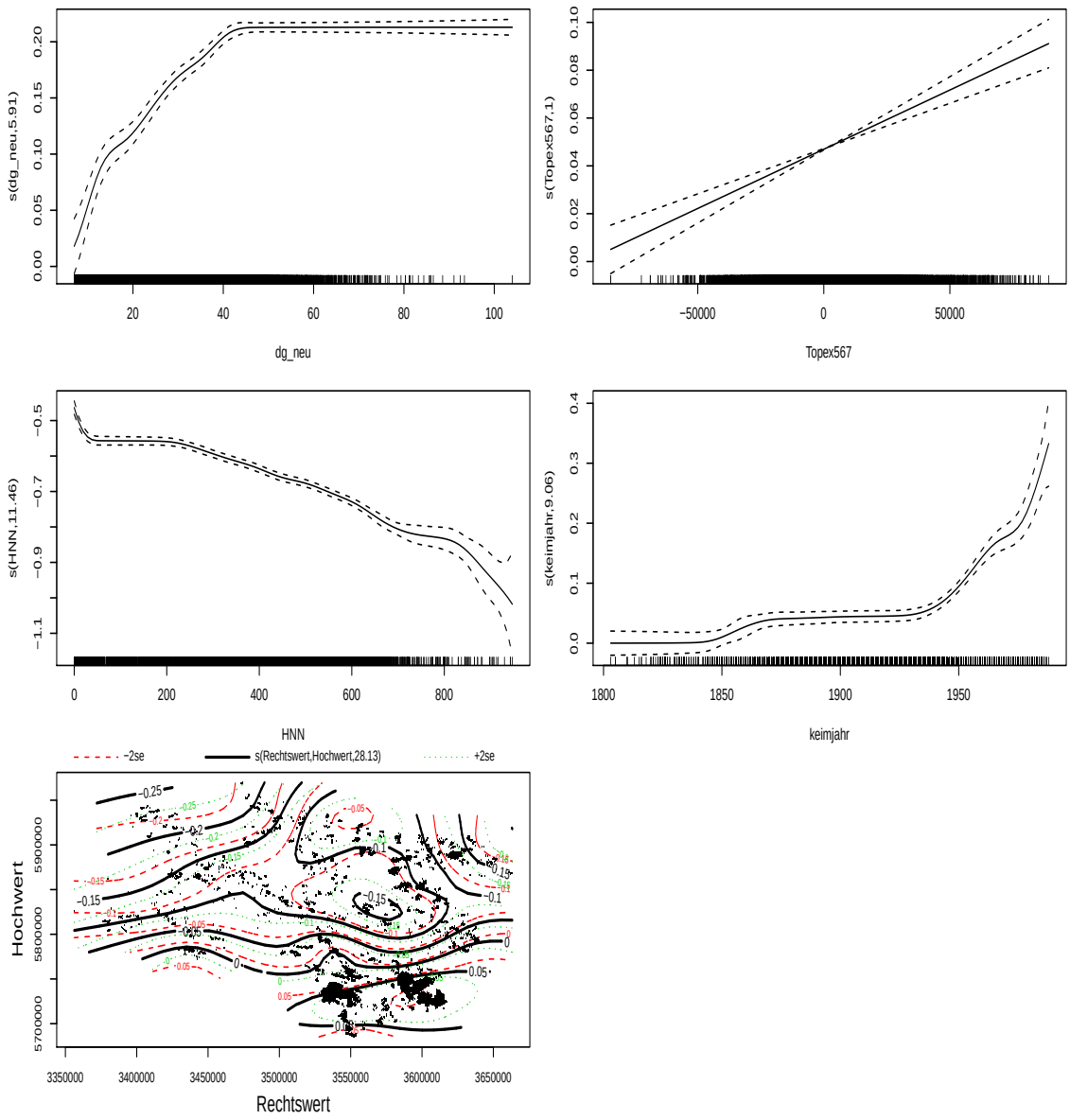


Figure 8-13: The estimated smooth terms of model 1.

```
    s(Topex567, k = 20, bs = "mpi", m = 2) + s(HNN, k = 20, bs = "mpd",
    m = 2) + s(keimjahr, k = 20, bs = "mpi", m = 2) + s(Rechtswert,
    Hochwert) + zkt + xkti + HNN_xkti + Alt_xkti

Parametric coefficients:
          Estimate Std. Error t value Pr(>|t|)
zkt      1.266e+00  4.248e-02  29.811  < 2e-16 ***
xkti     5.173e-01  1.184e-02  43.701  < 2e-16 ***
HNN_xkti 4.043e-05  1.024e-05   3.949 7.85e-05 ***
Alt_xkti 2.774e-03  1.799e-04  15.418  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Approximate significance of smooth terms:
                           edf Ref.df      F p-value
s(dg_neu)                5.909  5.909  43.59  <2e-16 ***
s(Topex567)              1.000  1.000  72.13  <2e-16 ***
s(HNN)                  11.422 11.422 122.57  <2e-16 ***
s(keimjahr)              9.061  9.061  27.22  <2e-16 ***
s(Rechtswert,Hochwert) 28.131 28.131  60.69  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

R-sq.(adj) =  0.905   Deviance explained =   97%
GCV score = 0.019155  Scale est. = 0.019116  n = 29324
```

The significance of every parameter of the parametric term is given together with the significance of the smooth terms. All the terms were significant in this case. As was noted before, the effective degrees of freedom of the `Topex567` smooth was only 1.

MODEL 2: with variable coefficients

A more complicated model with variable coefficients is considered as model 2. In the previous case the age effect on the slope B of the h-d curve was assumed to be linear as was the effect of altitude. Now, suppose that both predictors have non-linear

effects on B. Then the model becomes:

$$\begin{aligned}\log(H_{kti}) =& A_1 + p_{2a} \cdot z_{kt} + f_{2a}(\texttt{dg.neu}) + f_{3a}(\texttt{Topex567}) + f_{4a}(\texttt{HNN}) \\ &+ f_{5a}(\texttt{keimjahr}) + f_{6a}(\texttt{Rechtswert}, \texttt{Hochwert}) - p_{0b} \cdot x_{kti} \\ &+ p_{1b} \cdot f_{1b}(\texttt{Alt}) \cdot x_{kti} + p_{2b} \cdot f_{2b}(\texttt{HNN}) \cdot x_{kti} + \epsilon_{kti},\end{aligned}$$

where the non-linear effects of age and altitude are represented by the smooth functions $f_{1b}(\texttt{Alt})$ and $f_{2b}(\texttt{HNN})$, and both of them are assumed to be monotone increasing. Such a model can be fitted using 'by' variables (see Sections 4.1.2 and 6.3). The next lines fit the model and give `summary` results.

```
> m2 <- monogam(log(h)~-1+offset(A1)+s(dg_neu,k=20,bs="mpi",m=2)+
    s(Topex567,k=20,bs="mpi",m=2)+
    s(HNN,k=20,bs="mpd",m=2)+s(keimjahr,k=20,bs="mpi",m=2)+
    s(Rechtswert,Hochwert)+zkt+xkti+s(HNN,k=20,bs="mpi",m=2,by=xkti)+
    s(Alt,k=20,bs="mpi",by=xkti),data=dat.full,
    family=gaussian(link="identity"))
> summary(m2)

 Family: gaussian
 Link function: identity

 Formula:
 log(h) ~ -1 + offset(A1) + s(dg_neu, k = 20, bs = "mpi", m = 2) +
     s(Topex567, k = 20, bs = "mpi", m = 2) + s(HNN, k = 20, bs = "mpd",
     m = 2) + s(keimjahr, k = 20, bs = "mpi", m = 2) + s(Rechtswert,
     Hochwert) + zkt + xkti + s(HNN, k = 20, bs = "mpi", m = 2,
     by = xkti) + s(Alt, k = 20, bs = "mpi", by = xkti)

Parametric coefficients:
     Estimate Std. Error t value Pr(>|t|)
zkt   1.20896    0.04574  26.431  < 2e-16 ***
xkti  0.50162    0.07032   7.134 9.99e-13 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Approximate significance of smooth terms:
                          edf Ref.df      F p-value
```

```
s(dg_neu)                5.965  5.965  45.52  <2e-16 ***
s(Topex567)              1.001  1.001  71.85  <2e-16 ***
s(HNN)                  11.425 11.425 113.41  <2e-16 ***
s(keimjahr)              6.885  6.885  19.89  <2e-16 ***
s(Rechtswert,Hochwert) 28.024 28.024  58.47  <2e-16 ***
s(HNN):xkti              1.026  1.026  69.42  <2e-16 ***
s(Alt):xkti              9.318  9.318  28.52  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

R-sq.(adj) =  0.906   Deviance explained =   97%
GCV score = 0.019109  Scale est. = 0.019067  n = 29324
```

The summary results suggest that both terms with variable coefficients are significant. The GCV score for this model is slightly less than for model 1. Since the checking plots of this model and for the next considered model 3 are similar to those of model 1 they are not shown. The estimated smooth components are illustrated in Figure 8-14. The last two plots show the plausible effects of age and altitude.

MODEL 3: with site-sensitive covariates

The previous two models are not sensitive to site conditions: only proxy variables were incorporated there. Since one of the requirements for the model is to be able to predict future tree height development and assess the impact of climate change, two site-specific predictors are used in model 3. They are temperature summed over days of the vegetation period/growing season (`temp.veg`) and De Martonne's aridity index (`ari`). Both predictors are supposed to have monotone increasing effects on the h-d curve. The following model is considered:

$$\begin{aligned}\log\{\mathrm{E}(H_{kti})\} =& A_1 + p_{2a}\cdot z_{kt} + f_{2a}(\texttt{rel.d}) + f_{3a}(\texttt{keimjahr}) + f_{4a}(\texttt{temp.veg})\\ &+ f_{5a}(\texttt{ari}) + f_{6a}(\texttt{Rechtswert},\texttt{Hochwert}) - p_{0b}\cdot x_{kti}\\ &+ p_{1b}\cdot \texttt{Alt}\cdot x_{kti} + p_{2b}\cdot \texttt{HNN}\cdot x_{kti},\end{aligned}$$

where `rel.d` denotes a relative diameter at breast height, $\texttt{rel.d} = \texttt{dbh}_i/\texttt{d}_g$, the expression for the mean quadratic diameter of a stand $\texttt{d}_g$ was shown previously. The relative diameter is a measure of the ranking of a tree within the population of a stand. That means that two trees with the same $\texttt{dbh}_i$, but from the different stands, will have different rankings if other trees within the stand differ. Also to avoid the transformation

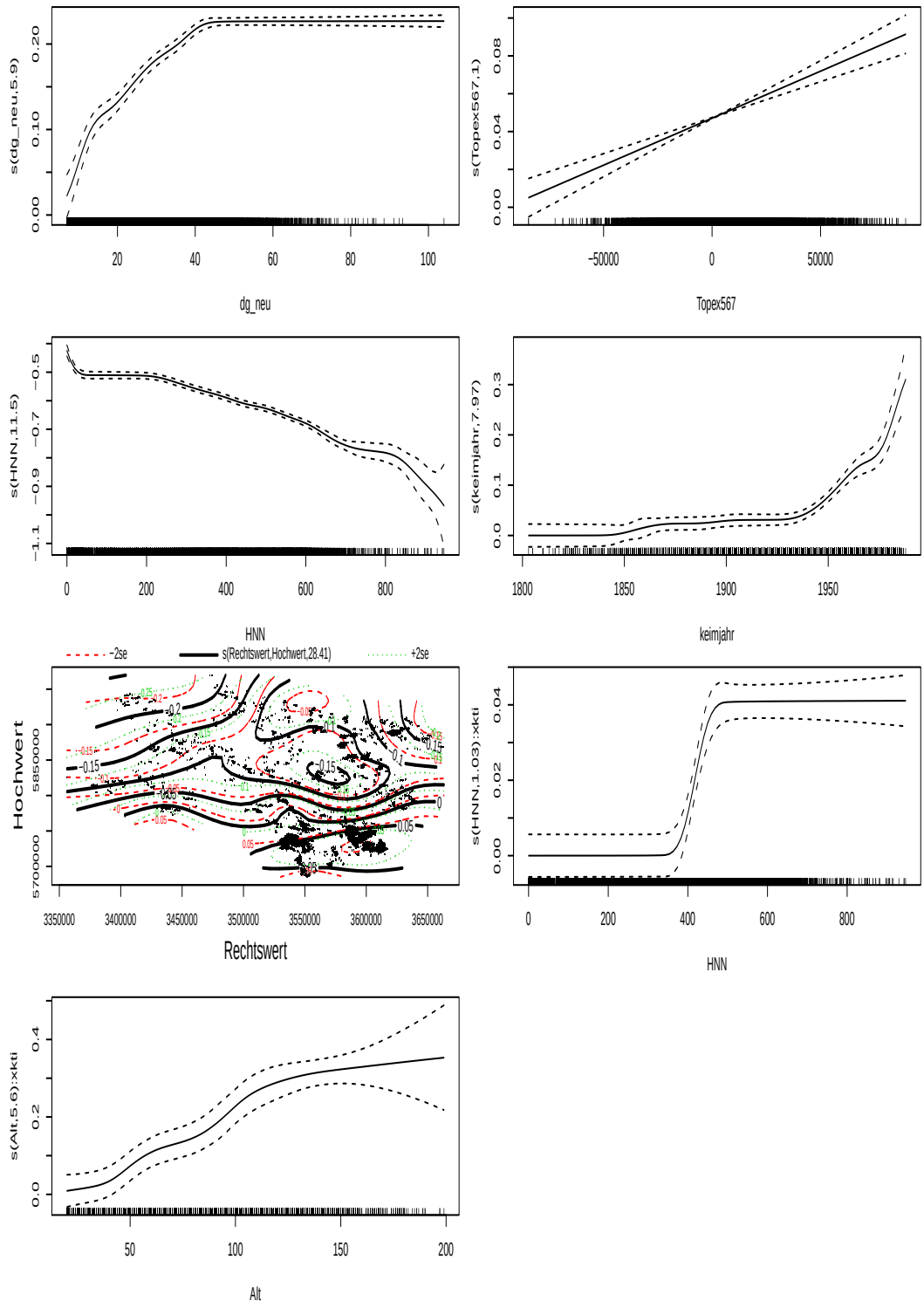


Figure 8-14: The estimated smooth terms of model 2 with variable coefficients.

bias gaussian distribution with the log link function is used here instead of the identity link for the log height. The next code shows the use of `monogam` to fit this model and get fitting results:

```
> m3 <- monogam(h~-1+offset(A1.fit)+s(rel_d,k=30,bs="mpd")+
     s(keimjahr,k=30,bs="mpi")+ s(temp_veg_mgcv,k=30,bs="mpi")+
     s(ari,k=30,bs="mpi")+ s(Rechtswert,Hochwert)+zkt+xkti+HNN_xkti+
     Alt_xkti, data=BI_climate[ind,], family=gaussian(link='log'),
     optimizer="optim", optim.method=c("BFGS","fd"))
> summary(m3)

Family: gaussian
Link function: log

Formula:
h ~ -1 + offset(A1.fit) + s(rel_d, k = 30, bs = "mpd") + s(keimjahr,
    k = 30, bs = "mpi") + s(temp_veg_mgcv, k = 30, bs = "mpi") +
    s(ari, k = 30, bs = "mpi") + s(Rechtswert, Hochwert) + zkt +
    xkti + HNN_xkti + Alt_xkti

Parametric coefficients:
          Estimate Std. Error t value Pr(>|t|)
zkt      1.571e+00  5.431e-02  28.932  < 2e-16 ***
xkti     5.296e-01  9.209e-03  57.507  < 2e-16 ***
HNN_xkti 8.195e-05  1.423e-05   5.758 8.59e-09 ***
Alt_xkti 5.001e-03  1.559e-04  32.076  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Approximate significance of smooth terms:
                           edf Ref.df     F p-value
s(rel_d)                 8.111   8.11 81.50  <2e-16 ***
s(keimjahr)             15.452  15.45 37.23  <2e-16 ***
s(temp_veg_mgcv)        22.892  22.89 13.98  <2e-16 ***
s(ari)                  17.909  17.91 16.81  <2e-16 ***
s(Rechtswert,Hochwert) 28.904  28.90 94.21  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
R-sq.(adj) =  0.909   Deviance explained = 96.3%
GCV score = 5.7992  Scale est. = 5.7799    n = 29326
```

For GCV minimization the `optim()` numerical optimization method with the finite-difference approximation of the derivatives was used as an alternative to the default BFGS method. It may be noted that the GCV score of this model is higher than for models 1 and 2. Figure 8-15 shows the estimates of the components of model 3. Both site predictors displayed credible monotone effects on the tree height.

Despite of the fact that the full data analysis has not been completed yet (at the point of writing this thesis), it can be seen that the second step of the h-d model development has been simplified and improved by using the mono-GAM approach.

Final re-parameterization as a linear mixed model

In the previous subsections the first two steps of the h-d model development were shown. After estimating the smooth (monotonic) model effects, they can be included into the complete model which is then re-parameterized as a linear mixed model:

$$\begin{aligned}\log(H_{kti}) =& A_1 + p_{2a} \cdot z_{kt} + m_1 \hat{f_{2a}}(\texttt{rel.d}) + m_2 \hat{f_{3a}}(\texttt{keimjahr}) + m_3 \hat{f_{4a}}(\texttt{temp.veg}) \\ &+ m_4 \hat{f_{5a}}(\texttt{ari}) + m_5 \hat{f_{6a}}(\texttt{Topex567}) + m_6 \hat{f_{7a}}(\texttt{HNN}) \\ &+ m_7 \hat{f_{8a}}(\texttt{Rechtswert}, \texttt{Hochwert}) - p_{0b} \cdot x_{kti} + p_{1b} \cdot \texttt{Alt} \cdot x_{kti} \\ &+ p_{2b} \cdot \texttt{HNN} \cdot x_{kti} + (\alpha_k + \alpha_{kt}) - (\beta_k - \beta_{kt}) x_{kti} + \epsilon_{kti},\end{aligned}$$

where the previously obtained estimates of the non-linear parameters are used as fixed constants and only linear parameters have to be re-estimated.

Summary

The data applications have demonstrated the efficacy and practicality of mono-GAM. Data sets with sample sizes ranging from 44 to 29 324 have been successfully analyzed. It has been demonstrated that mono-GAM may be useful in ecological and environmental studies in which monotone effects of some explanatory variables are expected from expert knowledge. Examples of such presumed monotone relationships can also be found in other research areas such as growth curves and dose-response curves in medicine, production functions (e.g., effect of labour input on quantity produced is assumed to be monotone increasing and concave) in economics, or the relationship between price and quantity demanded in business.

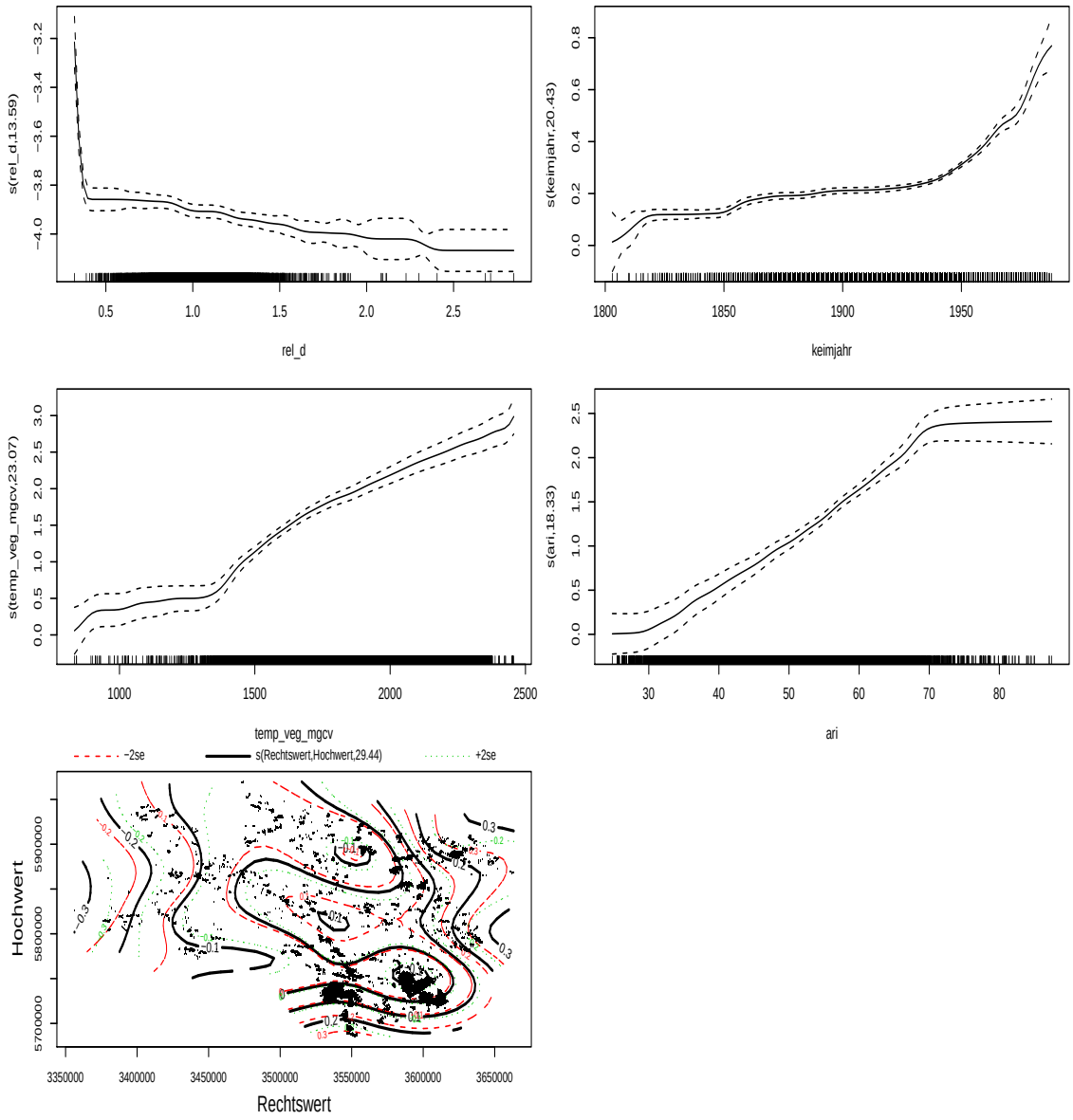


Figure 8-15: The estimated smooth terms of model 3 with the site-sensitive covariates.

Appendix A

R code used for data generating

This appendix provides the code for generating data used for the simulation examples of Section 6.1.

The following code generates data for the Poisson model with log link and a linear predictor which is the sum of unconstrained, monotone, and monotone plus convex smooth terms (model `b1` of Section 6.1).

```
set.seed(4)
n <- 200
x1 <- runif(n)*6-3
f1 <- 3*exp(-x1^2) # unconstrained term
x2 <- runif(n)*4-1;
f2 <- exp(4*x2)/(1+exp(4*x2)) # monotone increasing smooth
x3 <- runif(n)*3-1;
f3 <- exp(-3*x3)/15  # monotone decreasing and convex smooth
f <- f1+f2+f3
y <- rpois(n,exp(f))
dat1 <- data.frame(x1=x1,x2=x2,x3=x3,y=y)
```

The next code shows the data simulation for the Gaussian model with double monotonicity, `b2`.

```
set.seed(2)
n <- 30
x1 <- sort(runif(n)*4-1); x2 <- sort(runif(n))
f1 <- matrix(0,n,n)
for (i in 1:n) for (j in 1:n){
   f1[i,j] <- exp(4*x1[i])/(1+exp(4*x1[i])) + 2*exp(2*x2[j]-2)}
f0 <- as.vector(t(f1)); f <- (f0-min(f0))/(max(f0)-min(f0))
```

```
y <- f+rnorm(length(f))*0.1
x11 <- matrix(0,n,n); x11[,1:n] <- x1;
x11 <- as.vector(t(x11)); x22 <- rep(x2,n)
dat2 <- list(x1=x11,x2=x22,y=y)
```

Finally, the last bit of code generates data for `b3`, the single bivariate smooth regression model monotone increasing in the second covariate.

```
set.seed(2)
n <- 30
x1 <- runif(n)*1; x2 <- runif(n)*4-1
f1 <- matrix(0,n,n)
for (i in 1:n) for (j in 1:n){
      f1[i,j] <- 2*sin(pi*x1[i])+ exp(4*x2[j])/(1+exp(4*x2[j]))}
f0 <- as.vector(t(f1)); f <- (f0-min(f0))/(max(f0)-min(f0))
y <- f+rnorm(length(f))*0.1
x11 <- matrix(0,n,n)
x11[,1:n] <- x1; x11 <- as.vector(t(x11))
x22 <- rep(x2,n)
dat3 <- list(x1=x11,x2=x22,y=y)
```

Bibliography

R.S. Anderssen and P. Bloomfield. A time series approach to numerical differentiation. *Technometrics*, 16(1):69–75, 1974.

L.-E. Andersson and T. Elfving. An algorithm for constrained interpolation. *SIAM Journal on Scientific and Statistical Computing*, 8:1012–1025, 1987.

A. Antoniadis, J. Bigot, and I. Gijbels. Penalized wavelet monotone regression. *Statistics and Probability Letters*, 77:1608–1621, 2007.

M. Banerjee. Estimating monotone, unimodal and U-shaped failure rates using asymptotic pivots. *Statistica Sinica*, 18(2):467–492, 2008.

R. Barlow, D. Bartholemew, J. Bremner, and H. Brunk. *Statistical inference under order restrictions : the theory and application of isotonic regression*. New York: John Wiley, 1972.

R.K. Beatson. Monotone and convex approximation by splines: error estimates and a curve fitting algorithm. *SIAM Journal on Numerical Analysis*, 19(6):1278–1285, 1982.

K. Bollaerts, P.H. Eilers, and M. Aerts. Quantile regression with monotonicity restrictions using P-splines and the L_1- norm. *Statistical Modelling*, 6:189–207, 2006a.

K. Bollaerts, P.H. Eilers, and I. van Mechelen. Simple and multiple P-splines regression with shape constraints. *British Journal of Mathematical and Statistical Psychology*, 59:451–469, 2006b.

A. Brezger and W.J. Steiner. Monotonic regression based on Bayesian P-splines: an application to estimating price response functions from store-level scanner data. *Journal of Business and Economic Statistics*, 26(1):90–104, 2008.

A. Brezger, T. Kneib, and S. Lang. BayesX: Analysing Bayesian structured additive regression models. *Journal of Statistical Software*, 14(11), 2005.

P. Craven and G. Wahba. Smoothing noisy data with spline functions. *Numerische Mathematik*, 31:377–403, 1979.

A.C. Davison. *Statistical models*. New York: Springer, 2008.

C. De Boor. *A Practical Guide to Splines*. Cambridge University Press, 1978.

I.C. Demetriou. A theorem for piecewise convex-concave data approximation. *Journal of Computational and Applied Mathematics*, (164–165):245–254, 2004a.

I.C. Demetriou. Least squares convex-concave data smoothing. *Computational Optimization and Applications*, 29:197–217, 2004b.

I.C. Demetriou. The minimum sum of squares change to univariate data that gives convexity. *IMA Journal of Numerical Analysis*, (11):433–448, 1991.

Jr. Dennis, J.E. and R.B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics, 1996.

H. Dette and K.F. Pilz. A comparative study of monotone nonparametric kernel estimates. *Journal of Statistical Computation and Simulation*, 76(1):41–56, 2006.

H. Dette, N. Neumeyer, and K.F. Pilz. A simple nonparametric estimator of a strictly monotone regression function. *Bernoulli*, 12(3):469–490, 2006.

P. Diggle, S. Morris, P. Elliott, and G. Shaddick. Regression modelling of disease risk in relation to point sources. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 160(3):491–505, 1997.

D.B. Dunson. Bayesian semiparametric isotonic regression for count data. *Journal of the American Statistical Association*, 100(470):618–627, 2005.

D.B. Dunson and B. Neelon. Bayesian inference on order-constrained parameters in generalized linear models. *Biometrics*, 59:286–295, 2003.

K. Eerikäinen. Predicting the height-diameter pattern of planted Pinus kesiya stands in Zambia and Zimbabwe. *Forest Ecology and Management*, 175:355–366, 2003.

P.H. Eilers and B.D. Marx. Flexible smoothing with B-splines and penalties. *Statistical Science*, 11:89–121, 1996.

T. Elfving and L.-E. Andersson. An algorithm for computing constrained smoothing spline functions. *Numerische Mathematik*, 52:583–595, 1988.

P. Elliott, G. Shaddick, I. Kleinschmidt, D. Jolley, P. Walls, J. Beresford, and C. Grundy. Cancer incidence near municipal solid waste incinerators in Great Britain. *British Journal of Cancer*, 73:702–710, 1996.

L. Fahrmeir, T. Kneib, and S. Lang. Penalized structured additive regression for space-time data: a Bayesian perspective. *Statistica Sinica*, 14(3):731–761, 2004.

J.H. Friedman and B.W. Silverman. Flexible parsimonious smoothing and additive modelling (with discussions). *Technometrics*, 31:3–39, 1989.

J.H. Friedman and R. Tibshirani. The monotone smoothing of scatterplots. *Technometrics*, 26:243–250, 1984.

D. Ghosh. Incorporating monotonicity into the evaluation of a biomarker. *Biostatistics*, 8:402–413, 2007.

P.E. Gill, W. Murray, and Wright M.H. *Practical Optimization*. London: Academic Press, 1981.

G.H. Golub and C.F. van Loan. *Matrix computations*. Baltimore: Johns Hopkins University Press, 3rd edition, 1996.

C. Gu. *Smoothing Spline ANOVA Models*. New York: Springer, 2002.

C. Gu. Cross validating non-Gaussian data. *Journal of Computational and Graphical Statistics*, 1(2):196–179, 1992.

P. Hall and L.-S. Huang. Nonparametric kernel regression subject to monotonicity constraints. *The Annals of Statistics*, 29(3), 2001.

T. Hastie and R. Tibshirani. Bayesian backfitting (with discussions). *Statistical Science*, 155(3):196–223, 2000.

T. Hastie and R. Tibshirani. Generalized additive models. *Statistical Science*, 1:297–318, 1986.

T. Hastie and R. Tibshirani. *Generalized Additive Models*. Chapman & Hall, 1990.

T. Hastie and R. Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B*, 55(4):757–796, 1993.

X. He and P. Ng. COBS: qualitatively constrained smoothing via linear programming. *Computational Statistics*, 14:315–337, 1999.

X. He and P. Shi. Convergence rate of B-spline estimators of nonparametric conditional quantile functions. *Journal of Nonpararmetric Statistics*, 3:299–308, 1994.

X. He and P. Shi. Monotone B-spline smoothing. *Journal of the American Statistical Association*, 93(442):643–650, 1998.

C.C. Holmes and N.A. Heard. Generalized monotonic regression using random change points. *Statistics in Medicine*, 22:623–638, 2003.

U. Hornung. Interpolation by smooth functions under restrictions on the derivatives. *Journal of Approximation Theory*, 28:227–237, 1980.

L.D. Irvine, S.P. Martin, and P.W. Smith. Constrained interpolation and smoothing. *Constructive Approximation*, 2:129–151, 1986.

K. Jayaraman and J. Lappi. Estimation of height-diameter curves through multilevel models with special reference to even-aged teak stands. *Forest Ecology and Management*, 142:155–162, 2001.

C. Kelly and J. Rice. Monotone smoothing with application to dose-response curves and the assessment of synergism. *Biometrics*, 46:1071–1085, 1990.

R. Koenker, P. Ng, and S. Portnoy. Quantile smoothing splines. *Biometrika*, 81(4), 1994.

K. Kopotun, D. Leviatan, and A.V. Prymak. Constrained spline smoothing. *SIAM Journal on Numerical Analysis*, 46(4), 2008.

J.B. Kruskal. Analysis of factorial experiments by estimating monotone transformations of the data. *Journal of the Royal Statistical Society: Series B*, 27:251–263, 1965.

A. Lahtinen. Monotone interpolation with application to estimation of taper curves. *Annals of Numerical Mathematics*, 3:151–161, 1996.

S. Lang and A. Brezger. Bayesian P-splines. *Journal of Computational and Graphical Statistics*, 13(1):183–212, 2004.

J. Lappi. A longitudinal analysis of height/diameter curves. *Forest Science*, 43:555–570, 1997.

F. Leitenstorfer and G. Tutz. Generalized monotonic regression based on B-splines with an application to air pollution data. *Biostatistics*, 8(3):654–673, 2007.

X. Lin and D. Zhang. Inference in generalized additive mixed models by using smoothing splines. *Journal of the Royal Statistical Society: Series B*, 61:381–400, 1999.

J.W. Lindeberg. Eine neue herleitung des esponentialgesetzes in der wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, 15:211–225, 1922.

C.L. Mallows. Some comments on c_p. *Technometrics*, 15:661–675, 1973.

E. Mammen. Estimating a smooth monotone regression function. *The Annals of Statistics*, 19(2):724–740, 1991.

E. Mammen and C. Thomas-Agnan. Smoothing splines and shape restrictions. *Scandinavian Journal of Statistics*, 26:239–252, 1999.

E. Mammen, J.S. Marron, B.A. Turlach, and M.P. Wand. A general projection framework for constrained smoothing. *Statistical Science*, 16(3):232–248, 2001.

G. Marra and S.N. Wood. Coverage properties of confidence intervals for generalized additive model components. *Submitted to Scandinavian Journal of Statistics*.

R. Matzkin. Semiparametric estimation of monotone and concave utility functions for polychotomous choice models. *Econometrica*, 59(5):1315–1327, 1991.

D.F. McAllister and J.A. Roulier. An algorithm for computing a shape preserving osculatory quadratic spline. *ACM Transactions Mathematical Software*, 7:331–341, 1981.

L. Mehtätalo. A longitudinal height-diameter model for Norway spruce in Finland. *Canadian Journal of Forest Research*, 34(1):131–140, 2004.

L. Mehtätalo. Height-diameter models for Scots pine and birch in Finland. *Silva Fennica*, 39(1):55–66, 2005.

M. Meyer. Inference using shape-restricted regression splines. *The Annals of Applied Statistics*, 2(3):1013–1033, 2008.

M. Meyer and M. Woodroofe. On the degrees of freedom in shape-restricted regression. *The Annals of Statistics*, 28(4):1083–1104, 2000.

H. Mukerjee. Monotone nonparametric regression. *The Annals of Statistics*, 16(2): 741–750, 1988.

P. Ng and M. Maechler. A fast and efficient implementation of qualitatively constrained quantile smoothing splines. *Statistical Modelling*, 7(4):315–328, 2007.

J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer Science+Business Media, LLC, 2006.

F. O'Sillivan. A statistical perspective on ill-posed inverse problems. *Statistical Science*, 1:502–518, 1986.

J.K. Pal and M. Banerjee. Estimation of smooth regression functions in monotone response models. *Journal of Statistical Planning and Inference*, 138:3125–3143, 2008.

R.D. Peng and L.J. Welty. The NMMAPSdata package. *R news*, 4(2):10–14, 2004.

J.O. Ramsay. Monotone regression splines in action (with discussion). *Statistical Science*, 3(4):425–461, 1988.

J.O. Ramsay. Estimating smooth monotone functions. *Journal of the Royal Statistical Society: Series B*, 60(2):365–375, 1998.

C.R Rao. *Linear Stattistical Inference and Its Applications*. New York: Wiley, 1973.

F.J. Richards. A flexible growth function for empirical use. *Journal of Experimental Botany*, 10(29):290–300, 1959.

R.A. Rigby and D.M. Stasinopoulos. Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54 (3):507–554, 2005.

V. Rousson. Monotone fitting for developmental variables. *Journal of Applied Statistics*, 35(6):659–670, 2008.

M. Sarfraz. A rational cubic spline for the visualization of monotonic data. *Computers and Graphics*, 24:509–516, 2000.

M. Sarfraz. A rational cubic spline for the visualization of monotonic data: an alternative approach. *Computers and Graphics*, 27:107–121, 2003.

M. Schipper and J.M.G. Taylor. Generalized monotonic functional mixed models with application to modelling normal tissue complications. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 57(2):149–163, 2008.

J. Schmidt and I. Scholz. A dual algorithm for convex-concave data smoothing by cubic C^2-splines. *Numerische Mathematik*, 57(1):333–350, 1990.

R. Scott and S. Mitchell. Empirical modelling of windthrow risk in partially harvested stands using tree neighbourhood and stand attributes. *Forest Ecology and Management*, 218:193–209, 2005.

G. Shaddick, L.L. Choo, and S.G. Walker. Modelling correlated count data with covariates. *Journal of Statistical Computation and Simulation*, 77(11):945–954, 2007.

B.W. Silverman. Some aspects of the spline smoothing approach to nonparametric regression curve fitting. *Journal of the Royal Statistical Society: Series B*, 47:1–52, 1985.

I. Takeuchi, Q.V. Le, T.D. Sears, and A.J. Smola. Nonparametric quantile estimation. *Journal of Machine Learning Research*, 7:1231–1264, 2006.

C.W. Thornthwaite. The climates of North America: according to a new classification. *Geographical Review*, 21(4):633–655, 1931.

B. Turlach. Shape constrained smoothing using smoothing splines. *Computational Statistics*, 20:81–103, 2005.

E. Vassiliou and I.C. Demetriou. An adaptive algorithm for least squares piecewise monotonic data fitting. *Computational Statistics and Data Analysis*, 49:591–609, 2005.

G. Wahba. Bayesian confidence intervals for the cross validated smoothing spline. *Journal of the Royal Statistical Society: Series B*, 45:133–150, 1983.

G. Wahba. A comparison of GCV and GML for choosing the smoothing parameter in the generalized spline smoothing problem. *The Annals of Statistics*, 13(4), 1985.

G. Wahba. *Spline models for observational data*. Philadelphia: SIAM, 1990.

Z. Wang. An algorithm for generalized monotonic smoothing. *Journal of Applied Statistics*, 27(4), 2000.

S.N. Wood. Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society: Series B*, 62:413–428, 2000.

S.N. Wood. Partially specified ecological models. *Ecological Monographs*, 71(1):1–25, 2001.

S.N. Wood. Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association*, 99:673–686, 2004.

S.N. Wood. *Generalized Additive Models. An Introduction with R*. Chapman & Hall, 2006a.

S.N. Wood. On confidence intervals for generalized additive models based on penalized regression splines. *Australian and New Zealand Journal of Statistics*, 48(4):445–464, 2006b.

S.N. Wood. Fast stable direct fitting and smoothness selection for generalized additive models. *Journal of the Royal Statistical Society: Series B*, 70(3):495–518, 2008.

S.N. Wood. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B*, 73(1):1–34, 2011.

S.N. Wood. Monotonic smoothing splines fitted by cross validation. *SIAM (Society for Industrial and Applied Mathematics) Journal on Scientific Computing*, 15(5): 1126–1133, 1994.

S.N. Wood and N.H. Augustin. GAMs with integrated model selection using penalized regression splines and applications to environmental modelling. *Ecological Modelling*, 157:157–177, 2002.

J.T. Zhang. A simple and efficient monotone smoother using smoothing splines. *Journal of Nonparametric Statistics*, 16(5):779–796, 2004.