Journal of Econometrics 161 (2011) 166–181

Contents lists available at ScienceDirect

Journal of Econometrics

journal homepage: www.elsevier.com/locate/jeconom

Nonparametric function estimation subject to monotonicity, convexity and other shape constraints

Thomas S. Shively [a,*], Stephen G. Walker [b], Paul Damien [a]

[a] University of Texas at Austin, United States
[b] University of Kent, United Kingdom

ARTICLE INFO

Article history:
Received 2 March 2009
Received in revised form
3 June 2010
Accepted 6 December 2010
Available online 21 December 2010

JEL classification:
C11—Bayesian analysis
C14—Semiparametric and nonparametric methods

Keywords:
Fixed-knot splines
Free-knot splines
Log-concave likelihood functions
MCMC sampling algorithm
Small sample properties

ABSTRACT

This paper uses free-knot and fixed-knot regression splines in a Bayesian context to develop methods for the nonparametric estimation of functions subject to shape constraints in models with log-concave likelihood functions. The shape constraints we consider include monotonicity, convexity and functions with a single minimum. A computationally efficient MCMC sampling algorithm is developed that converges faster than previous methods for non-Gaussian models. Simulation results indicate the monotonically constrained function estimates have good small sample properties relative to (i) unconstrained function estimates, and (ii) function estimates obtained from other constrained estimation methods when such methods exist. Also, asymptotic results show the methodology provides consistent estimates for a large class of smooth functions. Two detailed illustrations exemplify the ideas.

1. Introduction

This paper uses regression splines in a Bayesian context to develop methods for the nonparametric estimation of functions subject to shape constraints in models with log-concave likelihood functions. The shape constraints we consider include monotonicity, convexity and functions with a single minimum. The class of models with log-concave likelihood functions contains many of those used regularly in the economics, finance and marketing literature including proportional hazard function models, generalized mixed models, and non-homogeneous Poisson processes, among others.

We consider shape-constrained function estimation using both fixed-knot and free-knot spline models. These models were originally proposed by Smith and Kohn (1996) and Denison et al. (1998), respectively, for unconstrained nonparametric function estimation. The advantage of using free-knot models is that the data are allowed to specify the number and location of the knots. However, the disadvantage is the increased complexity of the MCMC algorithm required to implement the method because the knots and spline coefficients in a free-knot model must be generated jointly.

The natural way to impose shape constraints in both fixed-knot and free-knot regression spline models is to impose restrictions on the spline coefficients. In a Bayesian context the constraints are imposed through the prior distributions on the coefficients. Shively et al. (2009) use this idea to develop a Bayesian method for monotone function estimation using fixed-knot splines in Gaussian regression models. The current paper departs from Shively et al. (2009) in the following key respects: (1) shape constraints via prior distributions on the spline coefficients are developed for free-knot spline models; (2) the methodology development applies to the family of models that have log-concave likelihood functions, not just Gaussian regressions; and (3) additional shape constraints that include convexity and functions restricted to a single minimum are developed for both free-knot and fixed-knot models.

We also develop a new MCMC slice sampler, requiring only one auxiliary variable, to do a full Bayesian analysis in the context of models with log-concave likelihood functions. The sampler is computationally efficient and numerically stable, and works well for large data sets. Given the general nature of the sampler, it is

* Corresponding address: Department of Information, Risk, and Operations Management, Mail Code B6500, University of Texas, Austin, TX 78712, United States. Tel.: +1 512 471 1753; fax: +1 512 471 0587.
E-mail address: Tom.Shively@mccombs.utexas.edu (T.S. Shively).

0304-4076/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.jeconom.2010.12.001

noted that this new MCMC method holds promise for families of models other than the ones in this paper. A simulation analysis in Section 3 shows that the algorithm converges faster than other MCMC algorithms.

Nonparametric function estimation subject to shape constraints has been studied extensively. Early work in the estimation of monotone functions includes Wright and Wegman (1980) and Friedman and Tibshirani (1984). More recently, Neelon and Dunson (2004) and Shively et al. (2009) developed monotone estimation methods in the context of Gaussian models, Manski and Tamer (2002) and Banerjee et al. (2009) discussed the problem in binary models, and Dunson (2005) and Schipper et al. (2007) considered the problem in Poisson and generalized mixed models, respectively.

Research in the nonparametric estimation of convex and concave functions includes Mammen (1991), Groeneboom et al. (2001) and Yatchew and Härdle (2006), among others. In terms of applications, convex function estimation is used extensively in derivative asset pricing models (see, for example, Yatchew and Härdle (2006) and the references therein, Broadie et al. (2000a,b) and Aït-Sahalia and Duarte (2003)). Functions constrained to have a single minimum are used in the energy economics literature. For example, Pardo et al. (2002) and Ihara et al. (2008) use these types of constrained functions to model the relationship between temperature and electricity demand, although the functions in these papers are parametric in nature.

To describe the general model used in this paper, let y_i, $i = 1, \ldots, n$, be a set of observable data, x_i an $r \times 1$ vector of regressor variables, $f_1(x_{1i}), \ldots, f_r(x_{ri})$ a set of unknown functions, some of which are shape-constrained, and ϕ_i a set of unknown parameters. The density function for y_i is $\pi(y_i|f_1(x_{1i}), \ldots, f_r(x_{ri}), \phi_i)$. Using this notation, we develop computationally efficient nonparametric shape-constrained function estimation techniques for models in which π is log-concave in $f_1(x_{1i}), \ldots, f_r(x_{ri})$ and ϕ_i. The method can be easily generalized to models for which π is a unimodal likelihood function with only a slight increase in the computational requirements.

Incorporating an appropriate shape constraint assumption into a model often results in considerably better function estimates than can be obtained using unconstrained function estimation techniques. This is illustrated in the simulation results in Section 5 for the estimation of a monotone function in the context of Cox's (1972) nonparametric hazard model. We also show through simulation that our estimator performs well in finite sample sizes for Poisson models relative to Dunson's (2005) and Schipper et al.'s (2007) nonparametric monotone function estimation methods. In addition, the asymptotic results in Section 6 and Appendix D show that our estimator provides consistent estimates.

The paper is organized as follows. Section 2 develops the nonparametric function estimation methods subject to different shape constraints. Section 3 outlines the MCMC sampling algorithms for the fixed-knot and free-knot models in the context of monotonicity constraints and log-concave likelihood functions; corresponding appendices provide details for the algorithms. Section 4 discusses the implementation of the function estimation methodology for specific models while Section 5 gives simulation results to show the small sample properties of our estimator relative to previously proposed estimators. Section 6 discusses the estimator's asymptotic properties and shows that it is a consistent estimator of a monotone function in a Poisson model. Extensions of these properties to other classes of models are treated in an Appendix. Section 7 contains two examples. The first illustrates monotone function estimation in the context of a discrete-time nonparametric proportional hazard model. The second applies the methodology to the estimation of a function constrained to have a single minimum in a Gaussian model with autocorrelated errors.

2. General model

The complete Bayesian model specifications for fixed-knot and free-knot models under monotonicity, convexity and single-minimum restrictions are detailed in this section. Sections 2.1 and 2.2 outline the general model, including the likelihood function and the fixed-knot and free-knot spline models. The prior distributions on the spline coefficients that constrain functions to be monotonic, convex or have a single minimum are discussed in Sections 2.3–2.5. The methodology outlined in Sections 2.1–2.5 and implemented using the MCMC sampling algorithms in Section 3 is discussed in the context of univariate function estimation only. However, the methodology and sampling schemes also apply to additive models as illustrated in the two examples in Section 7.

2.1. Likelihood function

Let $f_0(x)$ represent the unknown function of interest, ϕ_i represent a vector of parameters, and $\pi(y_i|f_0(x_i), \phi_i)$ represent the density function for y_i conditional on $f_0(x_i)$ and ϕ_i. The assumption we make regarding the density function π is that it is log-concave in $f_0(x_i)$ and ϕ_i. For example, for a Poisson-gamma model with counts y_i and frailty parameter ϕ_i:

$$\pi(y_i|f_0(x_i), \phi_i) \propto \exp\{-\phi_i f_0(x_i)\}(\phi_i f_0(x_i))^{y_i} \tag{1}$$

where the ϕ_i are independent and identically distributed gamma random variables. Without loss of generality, we will always assume $0 \leq x_1 \leq \cdots \leq x_n \leq 1$. Similarly, for a Weibull proportional hazard function model with hazard function $\lambda_i(t) = f_0(x_i)\lambda_0(t)$ where $\lambda_0(t) = \eta t^{\eta-1}$, observational data $y_i = (t_i, \delta_i)$ where $\delta_i = 1$ indicates that t_i is an uncensored observation and $\delta_i = 0$ indicates t_i is censored at c (so $t_i = c$ if $\delta_i = 0$), and $\phi_i = \eta$ for all i, the density function is:

$$\pi(t_i, \delta_i|f_0(x_i), \eta) = \left(\prod_{i:\delta_i=1} (\eta t_i^{\eta-1})\right) \times \exp\left\{\sum_{i=1}^{n} \delta_i f_0(x_i) - t_i^{\eta} \exp\{f_0(x_i)\}\right\}. \tag{2}$$

In this case, π is log-concave in $f_0(x_i)$ and η.

The interpretation of $f_0(x_i)$ and ϕ_i is model dependent. In the Poisson-gamma model, $E(Y_i) = \phi_i f_0(x_i)$ while in the proportional hazard model $f_0(x_i)$ is the proportionality constant for the ith hazard function. In many models, $f_0(x_i)$ is required to be positive. If this is the case, the restriction will be imposed through the prior on $f_0(x_i)$.

The proof in Section 6 and Appendix D that shows our function estimator is consistent holds for the class of models where π is log-concave in $f_0(x_i)$ and ϕ_i. However, in terms of the MCMC sampling algorithm we can weaken the assumption to one requiring only that the likelihood function be unimodal. The weaker assumption results in a slight increase in the computational requirements but the MCMC sampling algorithm still converges quickly.

Letting $y = (y_1, \ldots, y_n)'$ and assuming the y_i's are independent, the density function for y, and therefore the likelihood function for $f_0(x_1), \ldots, f_0(x_n), \phi_1, \ldots, \phi_n$, can be expressed as

$$\pi(y|f_0(x_1), \ldots, f_0(x_n), \phi_1, \ldots, \phi_n) = \prod_{i=1}^{n} \pi(y_i|f_0(x_i), \phi_i). \tag{3}$$

In practice, we use finitely parametrized fixed-knot and free-knot regression spline models to approximate $f_0(x)$. If $\tilde{f}(x)$ represents the fixed-knot or free-knot spline function then the resulting approximating model is

$$\pi(y|\tilde{f}(x_1), \ldots, \tilde{f}(x_n), \phi) = \prod_{i=1}^{n} \pi(y_i|\tilde{f}(x_i), \phi_i)$$

where $\phi = (\phi_1, \ldots, \phi_n)'$. We use $E(\tilde{f}(x)|y)$ as the estimate of $f_0(x)$.

2.2. Fixed-knot and free-knot regression spline models

This section provides a brief description of the fixed-knot and free-knot spline models along with the associated notation that will be used in the remainder of the paper. Prior distributions for all parameter values other than the regression spline coefficients are also given. The functions are constrained to take on specific shapes by putting constraints on the spline coefficients through their prior distributions. These priors are discussed in Sections 2.3–2.5.

2.2.1. Fixed-knot regression spline model

The fixed-knot quadratic regression spline approximation to the function $f_0(x)$ is

$$f^{(m)}(x) = \alpha + \beta_1 x + \beta_2 x^2 + \beta_3 (x-\eta_1)_+^2 + \cdots + \beta_{m+2}(x-\eta_m)_+^2 \quad (4)$$

where $\eta_1, \ldots, \eta_m$ are m fixed "knots" placed along the domain of the independent variable x such that $0 < \eta_1 < \cdots < \eta_m < 1$ and $(z)_+ = \max(0, z)$. Variable selection is used to determine which knots remain in the model. Smith and Kohn (1996) used this model for unconstrained nonparametric function estimation. Quadratic regression splines are used instead of the more typical cubic regression splines because they impose a degree of smoothness on the function but the constraints required to ensure various shape constraints are more tractable than for cubic splines.

For notational purposes, define J_j such that $J_j = 0$ if $\beta_j = 0$ and $J_j = 1$ if $\beta_j \neq 0$. The J_j values are assumed to be a priori independent with $\pi(J_j = 0) = p$ for $j = 1, \ldots, m+2$. The values used for p are discussed in Section 5. Alternative priors for J_j can also be used. For example, a prior that assigns equal probability to each number of knots could be used (see Cripps et al. (2005)). However, simulation results (available on request) indicate the prior $\pi(J_j = 0) = p$ works best in practice. The prior for α is $N(0, 10^{10})$.

2.2.2. Free-knot regression spline model

The free-knot linear regression spline approximation to the function $f_0(x)$ is

$$f^{(K)}(x) = \alpha + \beta_0 x + \beta_1 (x-\xi_1)_+ + \cdots + \beta_K (x-\xi_K)_+ \quad (5)$$

where $\xi_1, \ldots, \xi_K$ are K knots on the domain of the independent variable x such that $0 < \xi_1 < \cdots < \xi_K < 1$. We use ξ_j to represent the free knots to distinguish them from the fixed knots represented by η_j in the previous model. The general framework for the free-knot spline model is similar to the one originally proposed by Denison et al. (1998) for unconstrained function estimation. We use linear regression splines in the free-knot model because they impose a degree of smoothing similar to that imposed by quadratic regression splines with fixed knots. The free-knot model discussed below can be modified in a straightforward way to allow for piecewise constant or quadratic regression splines.

For notational purposes, let $\tilde{\xi}_K = (\xi_1, \ldots, \xi_K)'$ and $\tilde{\beta}_K = (\beta_0, \ldots, \beta_K)'$. We use uninformative priors on K and α. In particular, $pr(K = k) = 1/k_{\max}$, $k = 1, \ldots, k_{\max}$ where $k_{\max}$ is specified by the user, while the prior distribution for α is $N(0, 10^{10})$. For estimating monotone and convex functions the knots are assumed to be uniformly distributed on (0, 1] so the prior for $\tilde{\xi}_K|K$ is the density function of the order statistics for K independent random variables drawn from a $U(0, 1)$ distribution. It is also possible to use more informative prior distributions for K and $\tilde{\xi}_K|K$ if information is available about the number and location of the knots. For estimating a function with a single minimum, we put a prior distribution on the knot associated with the minimum of the function. The remaining knots are assumed to be uniformly distributed conditional on the location of this knot. This is discussed further in Section 2.5.

2.3. Prior distributions on the regression spline coefficients to impose monotonicity

The prior distributions on the β_j-coefficients used in the fixed-knot spline model to impose monotonicity are discussed in Shively et al. (2009). They showed how to implement the resulting constrained estimation methodology in the context of a Gaussian regression model. We show in Section 3.1 how to implement the methodology in any model with a log-concave likelihood function.

In the free-knot linear regression spline model with a given K, the constraints on the β_j-coefficients to impose monotonicity are $\beta_0 \geq 0, \beta_0 + \beta_1 \geq 0, \ldots, \sum_{j=0}^{K} \beta_j \geq 0$. Note that the constraints change as knots are added and deleted (i.e. as K changes). This is handled in the MCMC sampling algorithm discussed in Section 3.2 and Appendix C that we use to implement the methodology. In the free-knot model, unlike the fixed-knot model, the knots and regression coefficients must be generated jointly for each j with the monotonicity constraints updated appropriately.

In general, the linear restrictions on the elements of $\tilde{\beta}_K$ required to ensure the function is non-decreasing can be written as $\tilde{\gamma}_K = L_K \tilde{\beta}_K$, where $\tilde{\gamma}_K = (\gamma_0, \ldots, \gamma_K)'$ and L_K is a $(K+1) \times (K+1)$ matrix with the ijth element $= 1$ if $i \geq j$ and $= 0$ otherwise, and each element of $\tilde{\gamma}_K$ must be greater than or equal to zero. The portion of the $\tilde{\gamma}_K$ parameter space that guarantees a non-decreasing function is the multi-dimensional generalization of the first quadrant including the hyperplanes that border this space. The prior distribution we use for $\tilde{\gamma}_K|K$ is similar to the one used in Shively et al. (2009) for a fixed-knot model. More specifically, the prior is a mixture distribution of a normal distribution $N(0, \tau^2 I)$ constrained to the multi-dimensional generalization of the first quadrant, where I is the identity matrix with appropriate dimension and τ^2 is typically set to the number of observations, and probability distributions on the boundaries of this space. Putting distributions on the boundaries allows us to obtain good estimates when the function being estimated has significant flat portions. Neelon and Dunson (2004) use a mixture prior in their monotone function estimation methodology for the same reason. Setting τ^2 to the number of observations makes the prior similar to one used by Smith et al. (1998) in an unconstrained function estimation problem.

2.4. Prior distributions on the regression spline coefficients to impose convexity

This section provides a description of the fixed-knot regression spline prior on $f_0(x)$ used to impose convexity. A similar prior can be used in the context of a free-knot model. Consider the fixed-knot model for $f^{(m)}(x)$ in (4). $f^{(m)}(x)$ is a convex function if the second derivative is non-negative for all x where

$$\frac{d^2 f^{(m)}(x)}{dx^2} = 2\beta_2 + 2\beta_3 I(x > \eta_1) + \cdots + 2\beta_{m+2} I(x > \eta_m) \quad (6)$$

and $I(x > \eta) = 1$ if $x > \eta$ and $= 0$ otherwise. Constraints are imposed on the β_j-coefficients to ensure that the resulting function is convex. The constraints depend on $J_2, \ldots, J_{m+1}$ and change as variables enter and leave the model. For example, if $J_j = 1$ for all j then the constraints are $\beta_2 \geq 0, \beta_2 + \beta_3 \geq 0, \ldots, \sum_{j=2}^{m+2} \beta_j \geq 0$ (β_1 is unconstrained). Let $J = (J_2, \ldots, J_{m+2})$, β_J consist of the elements $\beta_2, \ldots, \beta_{m+2}$ corresponding to those elements of J that are equal to one, and L_J be an $\sum_{j=2}^{m+2} J_j \times \sum_{j=2}^{m+2} J_j$ matrix where the ijth element is 1 if $i \geq j$ and 0 otherwise. The linear restrictions on the elements of β_J required to ensure the function is convex can be written as $\gamma_J = L_J \beta_J$, where each element of γ_J must be greater than or equal to zero. The prior on γ_J is similar to the one discussed in Shively et al. (2009) used to impose monotonicity.

2.5. Prior distribution to ensure the function has a single minimum

This section discusses the free-knot regression spline prior used to impose the constraint that realizations from the prior and posterior function spaces have a single minimum. A prior distribution is also placed on the location of the minimum. Other than the constraint of a single minimum the function is estimated nonparametrically. The method in this section generalizes previous methods used in the energy economics literature where the location of the minimum is specified a priori or is estimated using a least squares-type approach (see Pardo et al. (2002) and Ihara et al. (2008)). Previous methods also typically assumed the parametric form of the function on either side of the minimum is known.

The shape constraint in this section is different from the monotonicity and convexity constraints discussed in Sections 2.3 and 2.4 because monotonicity and convexity are imposed by restricting the first and second derivatives, respectively, to be positive. The function constraint discussed below allows these derivatives to change signs but ensures that the first derivative changes sign exactly once.

A similar prior can be used with fixed-knot splines to impose the constraint of a single minimum. The disadvantage of using the fixed-knot model is that it is difficult to put a prior on the location of the minimum. In many examples, there will be important information available about this value that should be incorporated into the model.

The free-knot spline function we use is

$$\begin{aligned} f^{(K)}(x_i) = {} & \alpha + \beta_0 x_i + \beta_1^{(1)}(x_i - \xi_1^{(1)})_+ + \cdots + \beta_{K_1}^{(1)}(x_i - \xi_{K_1}^{(1)})_+ \\ & + \beta_{\min}(x_i - \xi_{\min})_+ + \beta_1^{(2)}(x_i - \xi_1^{(2)})_+ + \cdots \\ & + \beta_{K_2}^{(2)}(x_i - \xi_{K_2}^{(2)})_+ \end{aligned}$$

where $\xi_1^{(1)} < \cdots < \xi_{K_1}^{(1)} < \xi_{\min} < \xi_1^{(2)} < \cdots < \xi_{K_2}^{(2)}$, and $\xi_{\min}$ is the location of the function minimum. The goal is to develop priors on the coefficients so the spline function is monotone decreasing for $x < \xi_{\min}$ and monotone increasing for $x > \xi_{\min}$.

The number of knots less than $\xi_{\min}$ is K_1 with the associated knots and spline coefficients denoted $\tilde{\xi}_{K_1}^{(1)} = (\xi_1^{(1)}, \ldots, \xi_{K_1}^{(1)})'$ and $\tilde{\beta}_{K_1}^{(1)} = (\beta_1^{(1)}, \ldots, \beta_{K_1}^{(1)})'$, respectively. Similarly, the number of knots greater than $\xi_{\min}$ is K_2 with the associated knot and coefficient vectors given by $\tilde{\xi}_{K_2}^{(2)}$ and $\tilde{\beta}_{K_2}^{(2)}$.

A prior distribution is placed on $\xi_{\min}$. Given $\xi_{\min}$ and K_1, the prior for $\tilde{\xi}_{K_1}^{(1)}|\xi_{\min}, K_1$ is the density function of the order statistics for K_1 independent random variables drawn from a $U(0, 1)$ distribution. A similar prior is used for $\tilde{\xi}_{K_2}^{(2)}|\xi_{\min}, K_2$. Uninformative prior distributions are placed on K_1 and K_2. In particular, $pr(K_j = k) = 1/k_{j,\max}$, $j = 1, 2$ and $k = 1, \ldots, k_{j,\max}$ where $k_{1,\max}$ and $k_{2,\max}$ are specified by the user.

The spline coefficients β_0 and $\tilde{\beta}_{K_1}^{(1)}$ are constrained through their prior distribution so $f^{(K)}(x_i)$ is monotone decreasing on the interval $(0, \xi_{\min})$. Similarly, the coefficients $\beta_{\min}$ and $\tilde{\beta}_{K_2}^{(2)}$ are constrained so the function is monotone increasing on the interval $(\xi_{\min}, 1)$. As with the strictly monotone free-knot spline prior discussed in Section 2.3, we reparametrize to $\tilde{\gamma}_{K_1}^{(1)}$ and $\tilde{\gamma}_{K_2}^{(2)}$ and use mixture priors for $\tilde{\gamma}_{K_1}^{(1)}|K_1$ and $\tilde{\gamma}_{K_2}^{(2)}|K_2$ with the γ-coefficients constrained to the appropriate regions to ensure decreasing and increasing functions on $(0, \xi_{\min})$ and $(\xi_{\min}, 1)$. Finally, the knot $\xi_{\min}$ must remain in the model since it represents the required "turning point" of the function.

The MCMC algorithm given in Section 3.2 for strictly monotone function estimation in the context of free-knot splines can be modified to generate the values K_1, $\tilde{\xi}_{K_1}^{(1)}$, $\tilde{\gamma}_{K_1}^{(1)}$ and K_2, $\tilde{\xi}_{K_2}^{(2)}$, $\tilde{\gamma}_{K_2}^{(2)}$. The primary differences are in the conditional distribution of the knots and the constraints on $\tilde{\gamma}_{K_1}^{(1)}$ and $\tilde{\gamma}_{K_2}^{(2)}$, or equivalently, $\tilde{\beta}_{K_1}^{(1)}$ and $\tilde{\beta}_{K_2}^{(2)}$, to ensure the function has the appropriate shape.

The prior we use guarantees that each function realization from the prior and posterior function spaces has a single minimum. In principle, it is possible the posterior mean of the function $E(f^{(K)}(x)|y)$ will have more than one minimum because the space of functions with a single minimum is not a convex set. However, simulation results (available from the authors on request) indicate this seldom occurs in practice when the data generating function has a single minimum.

3. MCMC sampling algorithm

This section outlines a new MCMC sampling algorithm with good convergence properties for estimating shape-constrained functions in models with log-concave likelihood functions. In particular, Sections 3.1 and 3.2 show how to construct MCMC algorithms for fixed-knot and free-knot models, respectively, with monotonicity imposed. Given the prior distributions in Sections 2.4 and 2.5, the algorithms can be modified to handle functions constrained to be convex or have a single minimum.

The key idea in both the fixed-knot and free-knot algorithms is to incorporate a single latent variable to make them computationally tractable for models with log-concave likelihood functions. We show in Section 3.3 that the algorithm for the fixed-knot model has good convergence properties relative to existing methods when existing methods can be used. The algorithms will also work with a slight modification for likelihood functions with a single mode but that are not log-concave.

The main difference between the algorithms in Sections 3.1 and 3.2 is that the knots, spline coefficients and variable selection indicators must be generated jointly in the free-knot algorithm while in the fixed-knot algorithm only the spline coefficients and variable selection indicators need to be generated. Both algorithms handle the changing size of the spline coefficient vector as well as the changing constraints as knots enter and leave the model. The prior on the spline coefficients for the fixed-knot model is the same as the one given in Shively et al. (2009) in the context of Gaussian models. However, the MCMC algorithm in Section 3.1 to implement the methodology in non-Gaussian models is considerably different than the algorithm given in their paper.

3.1. MCMC sampling algorithm for the fixed-knot monotone spline prior

For the fixed-knot model in (4), define $J = (J_1, \ldots, J_{m+2})$, β_J to consist of the elements of $\beta = (\beta_1, \ldots, \beta_{m+2})$ corresponding to those elements of J that are nonzero, L_J to be a lower triangular matrix as defined in Shively et al. (2009), and $\gamma_J = L_J\beta_J$. Then the prior distribution on $\gamma_J|J$ given in Shively et al. to impose monotonicity (see their paper for details) is a mixture distribution of a normal distribution $N(0, \tau^2 I)$ constrained to the multi-dimensional generalization of the first quadrant and probability distributions on the boundaries of this space.

The model for $f^{(m)}$ in (4) can be written in matrix notation as $f^{(m)} = \iota\alpha + X_J\beta_J$ where X_J consists of the columns of $X = (x, x^2, \ldots, (x - \eta_m)^2)$ corresponding to the nonzero elements of J. To make the model analytically tractable for use in an MCMC algorithm, we reparametrize to give $f^{(m)} = \iota\alpha + W_J\gamma_J$ where $W_J = X_J L_J^{-1}$. The corresponding likelihood function is given by

$$\pi(y|\alpha, J, \gamma_J, \phi) = \prod_{i=1}^{n} \pi(y_i|\alpha, J, \gamma_J, \phi). \tag{7}$$

We show how to generate $(J_j, \gamma_j)|y,\ j = 1, \ldots, m+2$, and therefore the $f^{(m)}(x_i)|y$ values used to estimate $E(f^{(m)}(x_i)|y)$.

The likelihood function in (7) can be rewritten as

$$\pi(y|\alpha, J, \gamma_J, \phi) = \exp\{-s(y, \alpha, J, \gamma_J, \phi)\} \tag{8}$$

where $s(y, \alpha, J, \gamma_J, \phi) = -\sum_{i=1}^{n} \log[\pi(y_i|\alpha, J, \gamma_J, \phi_i)]$. The key to the MCMC algorithm is to introduce the scalar latent variable v such that

$$\pi(\alpha, J, \gamma_J, \phi, \theta, v|y) \propto e^{-v} I(v > s(y, \alpha, J, \gamma_J, \phi))\pi(\alpha, J, \gamma_J, \phi|\theta)\pi(\theta).$$

For notational purposes, let $J_{(-j)} = J$ without the jth element and $\gamma_{(-j)} = \gamma$ without the jth element. Using this notation, the MCMC sampling algorithm described below is used to carry out function estimation. For a discussion of Bayesian inference using MCMC methods see Gelfand and Smith (1990) and Tierney (1994).

(0) Start with some initial values $v^{[0]}$, $\alpha^{[0]}$, $J^{[0]}$, $\gamma^{[0]}$, $\phi^{[0]}$ and $\theta^{[0]}$.
(1) Generate v conditional on $\alpha, J, \gamma_J, \phi, \theta, y$;
(2) Generate (J_j, γ_j) conditional on $v, J_{(-j)}, \gamma_{(-j)}, \alpha, \phi, \theta, y;\ j = 1, \ldots, m+2$; (J_j, γ_j) will be generated as a block;
(3) Generate α conditional on $v, J, \gamma_J, \phi, \theta, y$;
(4) Generate ϕ conditional on $v, \alpha, J, \gamma_J, \theta, y$;
(5) Generate θ conditional on $v, \alpha, J, \gamma_J, \phi, y$.

Let $\alpha^{[l]}$, $\gamma^{[l]}$ and $J^{[l]}$ be the iterates of α, γ and J in the sampling period. If w_{Ji} represents the ith row of W_J, then an estimate of the posterior mean of the ith element of $f^{(m)}$, and therefore an estimate of $f_0(x_i)$ is $\frac{1}{L}\sum_{l=1}^{L}[\alpha^{[l]} + w_{J^{[l]},i}\gamma_{J^{[l]}}^{[l]}]$.

We will focus on generating v and (J_j, γ_j) in steps 1 and 2. Generating α and ϕ in steps 3 and 4 is done similarly to generating γ_j when $J_j = 1$ in step 2, and generating θ in step 5 is model specific.

1. *Generate* v:
   Generate v^* from an Exp(1) distribution and compute $v = v^* + s(y, \alpha, J, \gamma_J, \phi)$.
2. *Generate* $(J_j, \gamma_j); j = 1, \ldots, m+2$.

The actual method for generating (J_j, γ_j) uses rejection sampling. To motivate the importance of using rejection sampling as well as to outline the structure of the general method we first consider an exact method that does not require rejection sampling. The disadvantage of the exact method is that it is too computationally intensive to implement in practice and is numerically unstable. We show in Appendix A how to modify the exact method to use rejection sampling so that the resulting algorithm is efficient and stable. We note that the rejection sampling step typically has a high acceptance rate in the models we have worked with. The reason for this is discussed in the Appendix.

In the exact method (as well as the rejection sampling method), (J_j, γ_j) is generated as a block by generating J_j first, and then γ_j. To generate J_j and γ_j, we have

$$\pi(J_j, \gamma_j|\cdots) \propto I[v > s(y, \alpha, J, \gamma_J, \phi)]\pi(\gamma_j|J_j)\pi(J_j) \tag{9}$$

where "$\cdots$" represents $(y, \alpha, J_{(-j)}, \gamma_{(-j)}, \phi, \theta, v)$. For $J_j = 0$, this yields

$$\pi(J_j = 0|\cdots) \propto I(v > s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi))\pi(J_j = 0). \tag{10}$$

Note that $\pi(J_j = 0|\cdots) = 0$ if $v < s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi)$. To find $\pi(J_j = 1|\cdots)$, we integrate γ_j out of the density function in (9) with J_j set to one. To accomplish this, let

$$\tilde{s}(\gamma_j) = s(\gamma_j; y, \alpha, J_j = 1, J_{(-j)}, \gamma_{(-j)}, \phi) - v$$

so $\tilde{s}(\gamma_j)$ is a convex function (because the likelihood function is assumed to be log-concave). Then

$$\pi(J_j = 1, \gamma_j|\cdots) \propto I[\tilde{s}(\gamma_j) < 0]\pi(\gamma_j|J_j = 1)\pi(J_j = 1)$$

where $\pi(\gamma_j|J_j = 1)$ is a mixture distribution consisting of a point mass at zero and a $N(0, \tau^2)$ distribution constrained to $(0, \infty)$. The mixture distribution for $\gamma_j|J_j = 1$ is a result of the mixture prior on $\gamma_J|J$. If $\tilde{s}(\gamma_j)$ is greater than zero for all $\gamma_j \geq 0$, then $\pi(J_j = 1|\cdots) = 0$. Otherwise, let $a^*_{\min}$ and $a_{\max}$ represent the roots of this function. Noting that the monotonicity restriction is $\gamma_j \geq 0$, let $a_{\min} = \max\{0, a^*_{\min}\}$. Then

$$\pi(J_j = 1, \gamma_j|\cdots) \propto I(a_{\min} < \gamma_j < a_{\max})\pi(\gamma_j|J_j = 1)\pi(J_j = 1). \tag{11}$$

Given the values $a_{\min}$ and $a_{\max}$, $\pi(J_j = 1|\cdots)$ can be computed.

Unfortunately, $a^*_{\min}$ and $a_{\max}$ can only be computed numerically and obtaining them to a sufficient degree of accuracy is often computationally intensive with numerical problems arising if the roots are not sufficiently accurate. For this reason, we find bounds on the values $a_{\min}$ and $a_{\max}$, denoted $b_{\min}$ and $b_{\max}$, such that $b_{\min} \leq a_{\min}$ and $a_{\max} < b_{\max}$, and then do the appropriate sampling using a rejection sampling algorithm. The rejection sampling algorithm is discussed in detail in Appendix A. However, the basic idea is that the approximating density function in rejection sampling is obtained using $b_{\min}$ and $b_{\max}$ in place of $a_{\min}$ and $a_{\max}$ in the true density function given in (11). The true and approximating densities are then the same up to a constant except on the intervals $(b_{\min}, a_{\min})$ and $(a_{\max}, b_{\max})$. For reasons discussed in Appendix A, if the candidate draw for J_j is 0, it is always accepted. Also, if the candidate draw for J_j is 1 and γ_j is in the interval $(a_{\min}, a_{\max})$ it is always accepted (because the true and approximating densities are the same up to a constant). If the candidate draw for γ_j is in $(b_{\min}, a_{\min})$ or $(a_{\max}, b_{\max})$, it is rejected. However, using the method of finding bounds outlined in Appendix B typically gives bounds very close to $a_{\min}$ and $a_{\max}$. This means the rejection region will be small and the overall acceptance rate will be high. In fact, given the monotonicity constraint, the lower bound is often exact because it is often the case that $b_{\min} = a_{\min} = 0$. In this situation, only a bound on $a_{\max}$ is required.

3.2. MCMC sampling algorithm for the free-knot spline prior

To develop the MCMC algorithm for the free-knot model we use a reparametrized version of the model in (5) that is probabilistically equivalent. The reparametrized model is

$$f^{(m)}(x) = \alpha + \beta_0 x + \beta_1 J_1(x - \xi_1)_+ + \cdots + \beta_m J_m(x - \xi_m)_+ \tag{12}$$

where $m = k_{\max}$ and $J = (J_1, \ldots, J_m)$ with $J_j = 0$ or 1. Also, let ξ_J and β_J consist of the elements of ξ_j and β_j, respectively, corresponding to those elements of J that are equal to one, let X_J consist of the vector x and the regressor variables in (12) corresponding to those elements of J that are equal to one, and let L_J be a $(K+1)\times(K+1)$ matrix with ijth element $= 1$ if $i \geq j$ and $= 0$ otherwise, where $K = \sum_{j=1}^{m} J_j$. Note that the vector ξ_J has length K, where K is the same value as defined in Section 2.2.2. The model in (12) can be written in matrix notation as $f^{(m)} = \iota\alpha + X_J\beta_J$, or equivalently as $f^{(m)} = \iota\alpha + W_J\gamma_J$ where $\gamma_J = L_J\beta_J$ and $W_J = X_J L_J^{-1}$.

To make the reparametrized model probabilistically equivalent to the model in (5), the priors on J, $\xi_J|J$ and $\gamma_J|J$ must be specified appropriately. The prior for J is

$$pr[J = (j_1, \ldots, j_m)] = \frac{1}{(m+1)\binom{m}{k}}$$

where $j_1, \ldots, j_m$ are 0 or 1 and $k = \sum_{i=1}^{m} j_i$. Using this prior for J assigns equal probability to each number of knots. The priors for $\xi_J|J$ and $\gamma_J|J$ are the same as the priors for $\xi_K|K$ and $\gamma_K|K$ given in Section 2.2.2.

Table 1
Summary of the autocorrelation function values for different models, sampling methods and function values.

Func. value	Method	ACF lag				
		10	20	50	100	200
		Panel A: Poisson model				
$f(0.25)$	Single latent variable	0.209	0.141	0.084	0.060	0.040
	DWW	0.974	0.949	0.879	0.777	0.621
$f(0.50)$	Single latent variable	0.201	0.153	0.109	0.084	0.059
	DWW	0.987	0.974	0.936	0.880	0.787
$f(0.75)$	Single latent variable	0.113	0.089	0.067	0.053	0.037
	DWW	0.987	0.974	0.937	0.881	0.782
		Panel B: Probit/logit model				
$f(0.25)$	Single latent variable-Probit	0.070	0.033	0.011	0.004	0.001
	Single latent variable-Logit	0.029	0.013	0.005	0.002	0.000
	Albert and Chib	0.200	0.077	0.025	0.014	0.005
$f(0.50)$	Single latent variable-Probit	0.070	0.034	0.013	0.005	0.001
	Single latent variable-Logit	0.029	0.019	0.005	0.001	0.000
	Albert and Chib	0.263	0.108	0.039	0.023	0.015
$f(0.75)$	Single latent variable-Probit	0.020	0.014	0.006	0.001	0.000
	Single latent variable-Logit	0.008	0.006	0.002	0.001	0.001
	Albert and Chib-Probit	0.171	0.020	0.016	0.011	0.005
		Panel C: Gaussian model				
$f(0.25)$	Single latent variable	0.174	0.096	0.042	0.028	0.016
	SSW	0.152	0.075	0.029	0.019	0.013
$f(0.50)$	Single latent variable	0.215	0.131	0.082	0.054	0.033
	SSW	0.181	0.101	0.065	0.047	0.029
$f(0.75)$	Single latent variable	0.096	0.066	0.044	0.029	0.021
	SSW	0.065	0.051	0.034	0.026	0.013

The reported autocorrelation coefficients are averages across 50 runs of a simulation.

The likelihood function can now be written similarly to the likelihood function in (7) and (8) in Section 3.1 with the knots ξ_J included in the list of parameter values. As in Section 3.1, we introduce the latent variable v such that

$$\pi(\alpha, J, \xi_J, \gamma_J, \phi, \theta, v|y) \propto e^{-v} I(v > s(y, \alpha, J, \xi_J, \gamma_J, \phi)) \times \pi(\alpha, J, \xi_J, \gamma_J, \phi|\theta)\pi(\theta).$$

An MCMC algorithm can now be constructed similar to the one in Section 3.1 except that step (2) is different. Step (2) becomes:

(2) Generate (J_j, ξ_j, γ_j) conditional on $v, J_{(-j)}, \xi_{(-j)}, \gamma_{(-j)}, \alpha, \phi, \theta, y; j = 1, \ldots, m$; (J_j, ξ_j, γ_j) will be generated as a block.

To generate (J_j, ξ_j, γ_j) as a block requires a Metropolis–Hastings step. The acceptance rates are typically over 70% and often over 90% which implies the approximating distribution is a good one. $(J_j, \xi_j, \gamma_j)|\cdots$, where "$\cdots$" represents $v, J_{(-j)}, \xi_{(-j)}, \gamma_{(-j)}, \alpha, \phi, \theta, y$, is generated from an approximating distribution by generating J_j, then $\xi_j|J_j$ and finally $\gamma_j|\xi_j, J_j$ from conditional distributions that are good approximations to the true conditional distributions. The approximating conditional distributions are discussed in Appendix C.

3.3. Convergence rates and CPU times

This section reports simulation results to compare the convergence rates of the MCMC algorithm discussed in Section 3.1 for the fixed-knot spline model with the convergence rates of existing MCMC methods designed for three specific models: Poisson, probit/logit (i.e. binary data) and Gaussian. For each data set and each method, we compute the autocorrelation function, efficiency factors and CPU time per 1000 effective observations to measure convergence rates and compare these measures across methods.

For the Poisson model, the method of Section 3.1 is compared to Damien et al.'s (1999) method (DWW) that uses $2n$ latent variables with the simulation results showing the single latent variable method converges much faster. For the probit/logit model the method is compared to Albert and Chib's (1993) probit model method that uses n latent variables. The results show that the efficiency factor for the single latent variable method used in conjunction with a logit model is approximately four times the efficiency factor for Albert and Chib's method and requires less than half the CPU time per 1000 effective observations.

For the Gaussian model the method is compared to Shively et al.'s (2009) method (SSW) that does not require any latent variables. The comparison with the SSW method provides a useful measure of the impact of incorporating a single latent variable into the MCMC algorithm. As the simulation results show (see below), the algorithm converges only slightly slower than the SSW method. This suggests there is only a small impact on convergence rates and CPU times of incorporating the latent variable into an MCMC algorithm in the way we propose. However, the method provides an algorithm for a wide variety of models that can be difficult to handle using existing methods. Also, once the shell program is written, it can be easily modified to handle different models by changing only the $s(\gamma)$ and $s'(\gamma)$ functions.

For each model, 50 runs of a simulation are done. A sample size of $n = 400$ observations and x-values equally spaced on the interval (0, 1] are used for each run. The warm-up and sampling periods are both 50,000 for all the MCMC methods. The model used to generate the Poisson data is given in (1) with $\phi_i = 1$ for all i; the probit model $pr[Y_i = 1|f_0(x_i)] = \Phi[f_0(x_i)]$ where Φ is the standard normal cumulative distribution function is used to generate the binary data; and the Gaussian model is $y_i = f_0(x_i) + \varepsilon_i$, ε_i i.i.d. $N(0, 1)$. For each model, $f_0(x) = 0.1+2x^2$ (note that for the Poisson model $f_0(x)$ is the mean function and must be positive for all x).

The autocorrelation coefficients are computed for the function values $f(0.25), f(0.50)$ and $f(0.75)$ using iterates from the sampling period. The averages of the autocorrelation coefficients for lags 10, 20, 50, 100 and 200 across the 50 runs of the simulation are reported in Table 1 for the different methods and models.

For each function value, the efficiency factor is defined as

$$\text{Efficiency factor} = \frac{1}{1 + 2\sum_{h=1}^{\infty} \rho_h} \tag{13}$$

Table 2
Summary of the efficiency factors for different models, sampling methods and function values.

Method	Efficiency factors		
	$f(0.25)$	$f(0.50)$	$f(0.75)$
Panel A: Poisson model			
Single latent variable	0.040	0.039	0.090
DWW	–	–	–
Panel B: Probit/Logit model			
Single latent variable-probit	0.150	0.134	0.361
Single latent variable-logit	0.242	0.230	0.578
Albert and Chib	0.065	0.049	0.088
Panel C: Gaussian model			
Single latent variable	0.059	0.047	0.081
SSW	0.073	0.060	0.105

The efficiency factors are averages across 50 runs of a simulation.

Table 3
Summary of the CPU times for different models, sampling methods and function values.

Method	CPU time	
	Per 1000 actual observations	Per 1000 effective observations
Panel A: Poisson model		
Single latent variable	1.283	32.917
DWW	1.676	–
Panel B: Probit/Logit model		
Single latent variable-Probit	10.152	75.658
Single latent variable-Logit	2.851	12.393
Albert and Chib	1.337	27.373
Panel C: Gaussian model		
Single latent variable	0.963	19.924
SSW	1.015	16.932

The CPU times are averages across 50 runs of a simulation.

(see Gamerman and Lopes, 2006, page 126) where ρ_h is the autocorrelation coefficient for the iterates from the sampling period at lag h for the function value. The sum of the autocorrelation coefficients in (13) is truncated at lag 200. The averages of the efficiency factors across the 50 runs of the simulation for the function values $f(0.25)$, $f(0.50)$ and $f(0.75)$ are reported in Table 2.

Following Gamerman and Lopes (2006), the effective sample size is defined as (*Efficiency factor*) $\times n_{sampling}$ where $n_{sampling}$ is the actual number of iterates in the MCMC sampling scheme. The effective sample size can be interpreted as the size of a sample of independent iterates that will give the same MCMC sampling variance as the $n_{sampling}$ correlated iterates give. Since the efficiency factors vary slightly across function values, we compute the effective sample sizes for the function value $f(0.50)$. Table 3 reports the CPU times per 1000 effective observations for the function value $f(0.50)$ for the different methods and models. The CPU times per 1000 actual observations are also reported in Table 3. The CPU times are averages across the 50 runs of the simulation. All runs were done on a Dell Precision 490 workstation.

Panel A of Table 1 reports the averages of the autocorrelation coefficients when the DWW and single latent variable methods are applied to a Poisson model and shows the considerably faster convergence rates for the latent variable method. Panel A of Table 2 reports the efficiency factors for the latent variable method applied to the Poisson model. The efficiency factors are not reported for the DWW method because the ACF converges so slowly. Panel A of Table 3 reports the CPU times per 1000 actual and effective observations.

Panel B of Tables 1–3 report the averages of the autocorrelations, efficiency factors and CPU times for the Albert and Chib (1993) probit method and the latent variable method applied to probit and logit models (the 50 simulated data sets used in Panel B are generated from a probit model and the functions are then estimated using the three methods). The results show that the single latent variable method for a logit model converges considerably faster and requires less CPU time per 1000 effective observations.[1]

Panel C of Tables 1–3 report the averages of the autocorrelations, efficiency factors and CPU times for the SSW and single latent variable methods applied to a Gaussian model. As discussed above, the single latent variable method converges only slightly slower than the SSW method and the average CPU time per 1000 effective observations is only slightly greater.

4. Application to specific models

This section discusses the application of the shape-constrained function estimation methodology developed in Sections 2 and 3 to generalized additive and hazard function models, and shows specifically what the $s(\gamma)$ function is in each case. The method applies to a wide class of models such as generalized mixed and extreme value models, and non-homogeneous Poisson processes. The $s(\gamma)$ function is determined similarly in each of these cases.

4.1. Generalized additive models

Dunson (2005) develops a methodology for monotone function estimation in the context of a Poisson-gamma model while Schipper et al. (2007) generalize his method to the class of generalized additive models. Generalized additive models have been applied in a wide variety of fields, including economics (Kim and Marschke, 2005) and transportation (Kweon and Kockleman, 2005), among others.

The density function for the dependent variable in a generalized additive model can be written $\pi(y_i|\theta_i,\phi) = \exp(\{[y_i\theta_i - b(\theta_i)]/a_i(\phi)\} + c(y_i,\phi))$. Let $\mu_i = E(Y_i) = g^{-1}(f_0(x_i))$ where g is a link function and f_0 is an unknown function. We model f_0 with the fixed-knot regression spline given in (4) (a similar representation applies for the free-knot spline in (5)). Using the γ-parametrization, the likelihood function is

$$\begin{aligned}&\pi(y_1,\ldots,y_n|\alpha,J,\gamma,\phi)\\&\quad=\exp\left\{\sum_{i=1}^{n}\left[\frac{y_i(\alpha+w_{Ji}\gamma_J)-b(\alpha+w_{Ji}\gamma_J)}{a_i(\phi)}+c(y_i,\phi)\right]\right\}\\&\quad=\exp\{-s(y,\alpha,J,\gamma_J,\phi)\}.\end{aligned}$$

For the common types of generalized linear models (e.g. Gaussian, binomial, Poisson, negative binomial, etc.), $s(y,\alpha,J,\gamma_J,\phi)$ is a convex function in α, the elements of γ_J, and ϕ. Note that if there is only a single x-variable in the mean function then the form of the link function does not matter. If the mean must be positive (as in the Poisson model) this restriction can be imposed through the

[1] We note that the form of the link function in a nonparametric regression with a single x-variable does not impact the estimated probabilities because the flexibility of the function f_0 compensates for the different link functions. If there are multiple x-variables then the form of the link function will have an impact. If it is known that binary data are generated from a probit model and there are multiple x-variables, then Albert and Chib's (1993) method is the appropriate one to use. Even though it converges more slowly, each iteration is considerably faster than the single latent variable method as applied to a probit model. The MCMC algorithm using a single latent variable is computationally intensive for the probit model because it requires computing the standard normal cdf in the $s(\gamma)$ function. The MCMC algorithm for the logit model does not require such a calculation and is consequently much faster.

prior on the regression spline coefficients. This is discussed in more detail in Section 5.

The methodology can be easily generalized to allow for non-canonical link functions (which we consider in Section 5 in the context of a Poisson model). It can also be generalized to handle multivariate generalized additive models (e.g. a multinomial model) and generalized additive mixed models such as the Poisson-gamma model discussed in Section 2.

4.2. Hazard function models

Hazard function models are used extensively in economics (see, for example, Abbring and van den Berg (2003)), marketing (Mitra and Golder, 2002), and especially in finance (Duffie et al. (2007) and Bharath and Shumway (2008)). As Bharath and Shumway state, "Hazard models have recently been applied by a number of authors and probably represent the state of the art in default forecasting with reduced-form models". Many of the relationships in these types of economic models can be assumed to be monotonic or convex based on subject matter theory. For example, in Duffie et al. (2007), the estimated default intensities are expected to be monotonically decreasing in the distance to default.

In this section we consider Cox's nonparametric proportional hazard function model. Let t_i, $i = 1, \ldots, n$, represent the time of "death" for the ith subject. Since the ordering of the subjects is arbitrary, we will assume they are ordered in the order of their deaths, i.e. subject 1 dies first at time t_1, subject 2 dies second at time t_2, etc. Note that "subjects" and "deaths" have a variety of interpretations. In Duffie et al. (2007), the subjects are corporations and deaths are defaults.

The hazard function for subject i is

$$h_i(t) = \exp\{f_0[x_i(t)]\}h_0(t) \tag{14}$$

where $h_0(t)$ is an unknown hazard function, $x_i(t)$ is the value of the covariate for subject i at time t, and f_0 is a monotone or convex function with unknown functional form. Note that the covariate $x_i(t)$ is allowed to vary across both i and t. We model f_0 with the fixed-knot spline in (4). Then, using the γ-parametrization for the model, the partial likelihood given $t_1, \ldots, t_n$ corresponding to the likelihood function in (8) can be written

$$\begin{aligned}
&\pi(t_1, \ldots, t_n|\alpha, J, \gamma_J) \\
&\quad = \exp\left\{\sum_{i=1}^{n}\left[(\alpha + w_{Ji}\gamma_J) - \log\left(\sum_{k\in \text{ Risk set at time } t_i} \exp\{\alpha + w_{Jk}\gamma_J\}\right)\right]\right\} \\
&\quad = \exp\{-s(t, \alpha, J, \gamma_J)\}
\end{aligned}$$

where $t = (t_1, \ldots, t_n)$ and the *Risk set at time* t_i includes the subjects still alive the instant before time t_i. $s(t, \alpha, J, \gamma_J)$ is a convex function in α and the elements of γ_J.

5. Simulation results

The estimation methodology developed in Sections 2 and 3 is very general and applies to a wide class of models. However, for conciseness, the simulations in this section focus on monotone function estimation in two specific models: (1) Poisson models because nonparametric monotone function estimators have been studied extensively by Dunson (2005) and Schipper et al. (2007) for variations on this model; and (2) Cox's nonparametric hazard model because this model is used frequently in the economics, finance and marketing literature.

5.1. Poisson model

Here we compare the small sample properties of the free-knot and fixed-knot monotone regression spline estimators and the estimator proposed by Schipper et al. (2007). We note that all three methods can be applied to the class of generalized additive mixed models, including Poisson-gamma and Poisson dose models (see Schipper et al.). However, to focus on the quality of the function estimates we compare the methods in the context of a Poisson model. Dunson's (2005) and Schipper et al. 's methods are similar, with Schipper et al. 's method applying to a wider class of models. Also, Schipper et al. 's method has better small properties so we only report the simulation results for their method.

Using the Poisson model,

$$\pi(y|x) = \exp\{-\mu_0(x)\}\frac{[\mu_0(x)]^y}{y!},$$

the simulation experiment sets $n = 400$ and considers the following four mean functions:

(a) $\mu_0(x) = 2$ (flat function);
(b) $\mu_0(x) = 0.1 + 3x$ (linear function);
(c) $\mu_0(x) = \exp\{1.386x^3\}$ (exponential function);
(d) $\mu_0(x) = 1 + 3F(x)$, where $F(\cdot)$ is the distribution function for a $N(0.5, (0.1)^2)$ random variable.

These functions are chosen to represent a range of possible functions that might occur in practice. We note that the functions include ones with significant flat portions as well as ones that "change direction sharply". All the functions except the flat function have a range of three. The $n = 400$ x-values are equally spaced on $(0, 1]$.

For the fixed-knot spline estimator, $m = 9$ equally spaced knots are used and $p_j = \pi(J_j = 0) = 0.8$ while for the free-knot spline model $k_{\max} = 9$ is used. Also, for both models we set $f_0(x) = \mu_0(x)$ in (1) (with $\phi_i = 1$ for all i) and constrain $f_0(x)$ to be positive using a prior on α that is constrained to $(0, \infty)$, i.e. we estimate $\mu_0(x)$ directly rather than $\log[\mu_0(x)]$ as is often done. The MCMC sampling scheme was run for a warm-up period of 50,000 iterations and a sampling period of 200,000 iterations. Convergence occurred well before this many iterations.

If $\hat{\mu}_0(x_i)$ is the estimate of $\mu_0(x_i)$, we use the root-mean-square-error

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mu_0(x_i) - \hat{\mu}_0(x_i))^2}$$

to quantify the accuracy of $\hat{\mu}_0(x_i)$. The simulation results in Table 4 indicate that the free-knot and fixed-knot spline estimators both do better than the Schipper et al. (2007) estimator for each function considered. One of the advantages of fixed-knot and free-knot spline models that is reflected in the results is that they are adept at estimating functions that have a high degree of local variability yet still do well for globally smooth functions. The fixed-knot spline does better than the free-knot spline for two of the four functions. For the fixed-knot spline, using $m = 19$ knots gave similar results. All results are based on 50 simulation runs.

5.2. Cox nonparametric hazard function models

This section compares the small sample properties of the fixed-knot regression spline estimator with and without the monotonicity assumption imposed in a Cox hazard function model. The results show the substantial increase in the quality of the function estimates in situations where it is appropriate to impose monotonicity. We are unaware of other nonparametric monotone function estimation methods for the Cox hazard function model so there are no other estimation methodologies available for comparison purposes as there are for Poisson models.

The simulation experiment sets $n = 200$ and generates the order of the 200 "deaths" using the proportional hazard model in (14). The x-values vary across subjects and are constant through time. They are generated from a uniform distribution on the interval $(0, 1)$. The following four functions for $f_0(x)$ are considered (with corresponding proportionality function $\rho_0(x) = \exp\{f_0(x)\}$):

Table 4
Summary of root-mean square errors for the Poisson model.

Mean function	Fixed-knot spline	Free-knot spline		Schipper et al. method	
	Average RMSE	Average RMSE	Percentage increase (%)	Average RMSE	Percentage increase (%)
Flat	0.064	0.066	3.1	0.086	34.4
Linear	0.106	0.089	−16.0	0.124	17.0
Exponential	0.136	0.153	12.5	0.174	27.9
Normal dist. function	0.167	0.161	−3.6	0.209	25.1

All results are based on 50 simulation runs. The functions are defined above. Percentage increases/decreases are from the fixed-knot spline method.

Table 5
Summary of root-mean square errors for the Cox proportional hazard model.

Mean function	Monotone regression spline	Unconstrained regression spline	
	Average RMSE	Average RMSE	Percentage increase (%)
Flat	0.039	0.080	105.1
Linear	0.110	0.142	29.1
Exponential	0.102	0.115	12.7
Normal dist. function	0.137	0.161	17.5

All results are based on 50 simulation runs. The functions are defined above. Percentage increases are from the monotone regression spline method.

(a) $f_0(x) = 0$ so $\rho_0(x) = 1$ (flat function);
(b) $f_0(x) = \log(1+x)$ so $\rho_0(x) = 1 + x$ (linear function);
(c) $f_0(x) = 0.693x^3$ so $\rho_0(x) = \exp\{0.693x^3\}$ (exponential function);
(d) $f_0(x) = \log(1 + F(x))$ so $\rho_0(x) = 1 + F(x)$, where $F(\cdot)$ is the distribution function for a $N(0.5, (0.1)^2)$ random variable.

These functions are chosen to represent a variety of functional forms that are likely to occur in practice. All the proportionality functions except the flat function have a range of one to make comparisons across functions easier.

For both the monotone and unconstrained regression spline estimator, $m = 19$ equally spaced knots are used and $p_j = \pi(J_j = 0)$ is set to 0.8 for each j. The MCMC sampling schemes are run using warm-up periods of 5000 iterations and sampling periods of 40,000 iterations.

If $\hat{f}_0(x_i)$ is the estimate of $f_0(x_i)$, we use the root-mean-square-error

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_0(x_i) - \hat{f}_0(x_i))^2}$$

to quantify the accuracy of $\hat{f}_0(x)$. Table 5 gives the *RMSE* for the monotone and unconstrained estimators. The results indicate the value of imposing monotonicity to obtain better function estimates.

6. Consistency property

This section considers Bayesian consistency for monotone function estimation in the context of the Poisson model used in Section 5.1. We then show how to generalize the argument in Appendix D to prove consistency for the entire class of generalized additive models discussed in Section 4.1 and the proportional hazard function model discussed in Section 4.2. For each model, we develop conditions on the class of prior distributions for f, non-decreasing on [0, 1], so that f converges to some true underlying f_0 as sample sizes tend to infinity, and show that our prior is a member of this class. Similar proofs can also be used to show the convex function estimators obtained using the prior in Section 2.4 provides consistent estimates of the true underlying function f_0 in generalized additive and hazard function models.

Consider the Poisson model whereby y_i given x_i is Poisson with mean $f(x_i)$ and f is non-decreasing on [0, 1], the space of which is denoted by Ω. The aim here is to find conditions on the prior $\Pi(\mathrm{d}f)$ so that

$$\Pi(A_\varepsilon|(x_1, y_1), \ldots, (x_n, y_n)) \to 0 \quad \text{a.s.}$$

for all $\varepsilon > 0$, where $A_\varepsilon = \{f : d(f, f_0) > \varepsilon\}$,

$$d(f, f_0) = \int d_H(p(\cdot|f(x)), p(\cdot|f_0(x)))G_0(\mathrm{d}x),$$

d_H is a version of the Hellinger distance, $d_H(p_1, p_2) = 1 - \int \sqrt{p_1 p_2}$, so that

$$d_H(p(\cdot|f(x)), p(\cdot|f_0(x))) = 1 - \exp\lfloor -0.5\{f^{1/2}(x) - f_0^{1/2}(x)\}^2\rfloor,$$

where f_0 is the true function, $p(\cdot|f(x))$ denotes the Poisson distribution with mean $f(x)$ and G_0 is the distribution of the (x_i).

The posterior mass assigned to a set A will be written as $\Pi_n(A) = J_{nA}/I_n$ where

$$J_{nA} = \int_A \prod_{i=1}^{n} \frac{p(y_i|f(x_i))}{p(y_i|f_0(x_i))}\Pi(\mathrm{d}f)$$

and

$$I_n = \int_\Omega \prod_{i=1}^{n} \frac{p(y_i|f(x_i))}{p(y_i|f_0(x_i))}\Pi(\mathrm{d}f).$$

The aim is to show that $J_{nA_\varepsilon} < \mathrm{e}^{-nd}$ a.s. for all large n, for some $d > 0$, and that $I_n > \mathrm{e}^{-nc}$ a.s. for all large n for any $c > 0$. Consistency then follows because we can take $c < d$.

From the expression for J_{nA} we see that

$$\frac{J_{n+1\,A}}{J_{nA}} = \frac{m_{nA}(y_{n+1}|x_{n+1})}{p(y_{n+1}|f_0(x_{n+1}))},$$

where

$$m_{nA}(y_{n+1}|x_{n+1}) = \int_A p(y_{n+1}|f(x_{n+1}))\Pi_A(\mathrm{d}f|(x_1, y_1), \ldots, (x_n, y_n))$$

and Π_A is Π restricted and normalized to the set A. Then

$$E\left(\sqrt{\frac{J_{n+1\,A}}{J_{nA}}}\,\middle|\,\mathcal{F}_n\right) = 1 - \int d_H(m_{nA}(\cdot|x), p(\cdot|f_0(x)))G_0(\mathrm{d}x)$$

where $\mathcal{F}_n = \sigma((x_1, y_1), \ldots, (x_n, y_n))$.

Now let A be a subset of Ω such that for all f_1 and f_2 in A it is that $d(f_1, f_2) < \delta$. We can fill up the space of $\Omega - A_\varepsilon^c$ with such sets, say $\{A_j\}_{j=1}^{\infty}$. This follows since

$$d(f_1, f_2) \le 0.5 \sup_x |f_1(x) - f_2(x)|$$

and the space of continuous real valued functions on [0, 1] is separable with respect to the uniform metric. Then, for all j, and for some $f_j \in A_j$, using the triangular inequality,

$$\int d_H(m_{nA_j}(\cdot|x), p(\cdot|f_0(x)))G_0(\mathrm{d}x)$$
$$\geq \int d_H(p(\cdot|f_j(x)), p(\cdot|f_0(x)))G_0(\mathrm{d}x)$$
$$- \int d_H(m_{nA_j}(\cdot|x), p_j(\cdot|f_0(x)))G_0(\mathrm{d}x)$$

and so

$$\int d_H(m_{nA_j}(\cdot|x), p(\cdot|f_0(x)))G_0(\mathrm{d}x) \geq \varepsilon - \delta = \varepsilon/2$$

once we have taken $\delta = \varepsilon/2$. Hence,

$$E\sqrt{J_{nA_j}} \leq (1-\varepsilon/2)^n \sqrt{\Pi(A_j)}.$$

Therefore, in order to achieve

$$\sum_j \sqrt{J_{nA_j}} \leq \mathrm{e}^{-nd} \quad \text{a.s. for all large } n$$

we need

$$\sum_j \sqrt{\Pi(A_j)} < +\infty.$$

Now

$$\Pi_n(A_\varepsilon) = \sum_j \Pi_n(A_j) \leq \sum_j \sqrt{\Pi_n(A_j)} = I_n^{-0.5} \sum_j \sqrt{J_{nA_j}}$$

and therefore the required consistency result now follows, since for all suitable f_0 with a Kullback–Leibler support condition of Π, it is that $I_n > \mathrm{e}^{-nc}$ a.s. for all large n, for any $c > 0$. See, for example, Walker (2004).

Hence, we need to find conditions on Π so that $\sum_j \sqrt{\Pi(A_j)} < +\infty$. Equivalently, to establish the restriction on the prior distribution of the parameters $\{\gamma_k\}_{k=1}^{\infty}$, where each γ_k is a 3-vector of positive r.v., in order to ensure $\sum_j \sqrt{\Pi(A_j)} < +\infty$. The prior for γ_{kl}, $l = 1, 2, 3$, will be denoted by π_{kl} and all are independent. We can obtain a set A, i.e. f_1 and f_2 in A, by having the associated parameters γ_1 and γ_2 so that

$$|\gamma_{1kl} - \gamma_{2kl}| < \delta_k$$

for all k and l where $\sum_k \delta_k < \delta^*$ for some δ^* related to δ. We will use $\phi_1 = \gamma_{11}, \phi_2 = \gamma_{12}, \ldots, \phi_4 = \gamma_{21}$ and so on, and π_k denotes the prior for ϕ_k.

Now define, for $\tilde{\delta}_k = \delta_{[k/3]}$,

$$B_{mk} = (m\tilde{\delta}_k, (m+1)\tilde{\delta}_k),$$

for $m = 0, 1, \ldots$, so we are looking for

$$\lim_{N\to\infty} \sum_{r_1=0}^{\infty} \cdots \sum_{r_N=0}^{\infty} \prod_{k=1}^{N} \sqrt{\pi_k(B_{r_k k})} < +\infty,$$

that is

$$\prod_{k=1}^{\infty} \left\{ 1 + \sum_{r=1}^{\infty} \sqrt{\pi_k(B_{rk})} \right\} < +\infty.$$

Using

$$\pi_k(B_{rk}) < pr(\phi_k^{2+} > r^{2+}\tilde{\delta}_k^{2+}) < E(\phi_k^{2+})/(r\tilde{\delta}_k)^{2+},$$

where 2+ etc. means $2 + a$ for any $a > 0$, (1) holds if

$$\prod_{k=1}^{\infty} \{1 + \xi \tilde{\delta}_k^{-1-} \sqrt{E(\phi_k^{2+})}\} < +\infty$$

where $\xi < \infty$ does not depend on k. This holds, taking $\tilde{\delta}_k \propto k^{-1-}$, when

$$\sum_{k=1}^{\infty} k^{1+} \sqrt{E(\phi_k^{2+})} < +\infty.$$

Hence, it is sufficient to take

$E(\gamma_{[k/3]l}^{2+}) \propto k^{-4-}$;

that is, for some $a > 0$ and $b > 0$,

$E(\gamma_{[k/3]l}^{2+a}) \propto k^{-4-b}$.

7. Examples of shape-constrained function estimation

The first example applies the monotone function estimation methodology to a discrete-time nonparametric proportional hazard model in the context of unemployment data. The second applies the methodology to the estimation of a function constrained to have a single minimum in a Gaussian model for electricity consumption with autocorrelated errors.

7.1. Monotone function estimation in a discrete-time proportional hazard model

This section uses a discrete-time hazard function model to analyze Spanish male unemployment data. The data consist of 1279 unemployed workers who started receiving unemployment insurance (UI) benefits in February 1987. These data were first analyzed by Jenkins and Garcia-Serrano (2004). An excellent in-depth discussion of the data and the importance of the analysis are given in their paper.

One of the goals of their analysis and the one we consider here is to model the monthly re-employment hazard rate, $h_i(t)$, i.e. to model the probability that UI recipient i will get a job in month t given that he is unemployed at the end of month $t-1$. The explanatory variables included in the hazard model are:

(1) x_{1it}: Time-to-exhaustion of UI benefits for recipient i in month t. The hazard rate is expected to be a non-increasing function of the time-to-exhaustion of benefits because the incentive to get a job increases as the UI benefits run out. This is a time-varying covariate because time-to-exhaustion decreases as t increases.
(2) x_{2it}: Income replacement rate for recipient i in month t. The amount of UI benefits paid to a specific recipient is a percentage of his most recent salary. The replacement rate varies across recipients according to a well-defined rule. The replacement rates may also vary across the unemployment period for a specific recipient with x_{2it} taking on one value for the first six months, a possibly lesser value for the second six months, and a lesser value still for the final 12 months. The hazard rate is expected to be a non-increasing function of the replacement rate because the incentive to get a job increases as the income replacement rate decreases. The income replacement rate is a time-varying covariate.
(3) x_{3i}: Age of recipient i in the month he begins receiving UI benefits. It is unclear from a subject matter perspective what the relationship between the recipient's age and hazard rate will be. For this reason, the function associated with age is unconstrained (i.e. it is estimated nonparametrically but without any monotonicity constraints imposed).
(4) D_{1i}: Dummy variable representing recipient i's family status (=1 if he has a family).
(5) $D_{2i}, D_{3i}, D_{4i}, D_{5i}$: Dummy variables representing the region of Spain recipient i resides in (Center, North-East, South, and Islands, respectively). North is the baseline region left out of the model.
(6) D_{6i}: Dummy variable representing whether recipient i's last job before starting UI benefits was temporary or permanent (=1 if he had a temporary job).

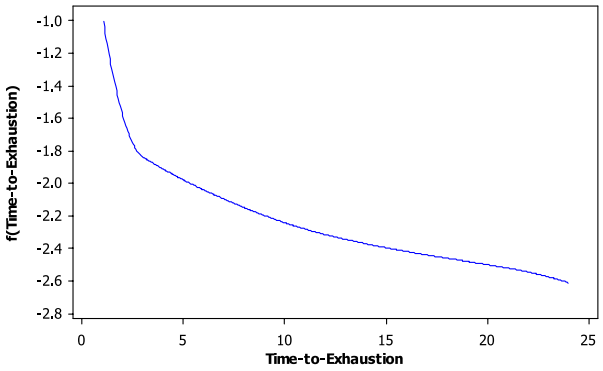

Fig. 1a. Function estimate for time-to-exhaustion.

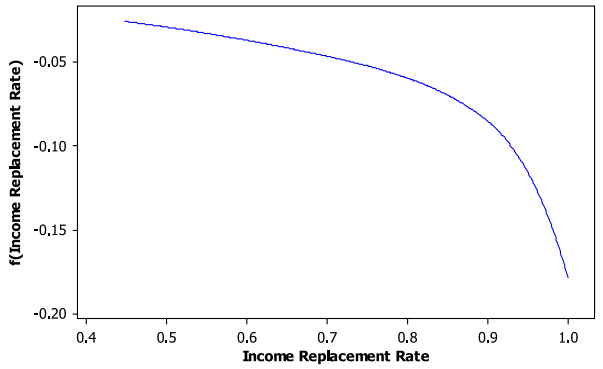

Fig. 1b. Function estimate for income replacement rate.

The maximum length of time a recipient is eligible for UI benefits is 24 months, although many workers have a shorter eligibility period. The observation for a specific recipient is censored if his eligibility is exhausted or if he stops receiving UI benefits for any reason other than getting a job (for example, death, permanent disability, emigration, etc.).

The hazard model used in this section is similar to the one used by Jenkins and Garcia-Serrano (2004):

$$\frac{h_i(t)}{1-h_i(t)} = \frac{h_0(t)}{1-h_0(t)} \times \exp\left\{f_1(x_{1it}) + f_2(x_{2it}) + f_3(x_{3i}) + \sum_{k=1}^{6} \lambda_k D_{ki}\right\}$$

where $h_0(t)$ is the baseline hazard function. f_1, f_2 and f_3 are unknown functions estimated nonparametrically with f_1 and f_2 restricted to be non-increasing functions. Estimates of the functions f_1 and f_2 are plotted in Fig. 1. The estimated values of $\lambda_1, \ldots, \lambda_6$ with their standard errors in parentheses are 0.09(0.09), −0.08(0.12), −0.23(0.15), −0.36(0.13), −0.13 (0.23), 0.51(0.13), respectively.

The estimate of the function f_1 increases as time-to-exhaustion decreases, and it increases at an increasing rate. This confirms the intuition that the probability a UI recipient gets a job will increase significantly as the benefits run out. The estimate of the function f_2 is decreasing as the income replacement rate increases. This also supports the intuition discussed above that the probability a UI recipient gets a job will be lower the higher the income replacement rate is. The function f_3 is not shown to conserve space but it shows very little change over the range of age values in the data. The dummy variable with the largest coefficient is D_{6i}. The coefficient estimate $\hat{\lambda}_6 = 0.51$ indicates a UI recipient whose last position was a temporary position is considerably more likely to get a job in any given month than a recipient whose last position was a permanent position.

7.2. Function estimation in a model for electricity consumption with the constraint that the function has a single minimum

The deregulation of the electricity market in many countries has generated substantial interest in determining the relationship between electricity demand and weather variables, especially temperature (see for example, Pardo et al. (2002) and Psiloglou et al. (2009)). It is well-known that the relationship between demand and temperature has a single minimum although the location of the minimum is different in different regions. Pardo et al. specified that the minimum occurs at 18 °C in their analysis of Spanish electricity data while Psiloglou et al. determined that the minimum occurs at 20 °C in Athens and 16 °C in London.

This section contains an analysis of electricity consumption and weather data from the New South Wales, Australia electricity market for the period January 1, 2004 to December 31, 2004. These data were originally analyzed in Panagiotelis and Smith (2008) in a different context. The data are available every half hour. We analyze the 7 am, 3 pm and 7 pm observations as representative of the demand over the course of the day. 7 am represents morning demand before most people leave for work and temperatures tend to be low, 3 pm represents mid-afternoon when temperatures tend to be highest, and 7 pm represents early evening when most people are home from work.

The following model was fit to electricity demand (separate models are fit to the three sets of observations):

$$Demand_t = \alpha_0 + \alpha_1 Time_t + f(Temp_t) + \sum_{j=1}^{11} \omega_j M_{jt} + \sum_{j=1}^{6} \lambda_j D_{jt} + \sum_{j=1}^{5} \delta_j PH_{jt} + \varepsilon_t \quad (15)$$

where $Demand_t$ and $Temp_t$ are the electricity demand and temperature on day t. The function $f(Temp)$ is estimated nonparametrically and constrained to have a single minimum. $Time_t$ is a trend variable that takes on the value t on day t. There are strong seasonal and day of the week effects so eleven monthly dummy variables, M_{jt}, $j = 1, \ldots, 11$, and six dummy variables for days of the week, D_{jt}, $j = 1, \ldots, 6$, were included. Dummy variables were also included for the public holidays of New Year's Day, Good Friday, Easter Sunday, Christmas and Boxing Day.

The model in (15) is similar to the one estimated by Pardo et al. except they modeled $f(Temp_t)$ as

$$f(Temp_t) = \beta_1 HDD_t + \beta_2 CDD_t$$

where $HDD_t = \max(Temp_{ref} - Temp_t, 0)$ and $CDD_t = \max(Temp_t - Temp_{ref}, 0)$ and $Temp_{ref}$ is a reference temperature that is selected to adequately separate the cold and heat branches of the demand/temperature relationship.

When the model in (15) is estimated there is a significant problem with autocorrelation in the residuals. For example, for the 7 pm data the first four autocorrelation coefficients are 0.50, 0.33, 0.20 and 0.11 and are all more than three standard errors from zero. Pardo et al. find similar autocorrelation in their model for electricity consumption. To account for this we assume a first-order autocorrelation process for the ε_t. The autocorrelation is incorporated into the constrained function estimation methodology using a technique similar to Smith et al. (1998). A uniform $U(0, 0.95)$ prior is placed on the autocorrelation coefficient. When the model is re-estimated the first six autocorrelation coefficients are all within two standard errors of zero as are the weekly autocorrelation coefficients at lags 7, 14 and 21.

The estimates of the functions $f(Temp_t)$ obtained for the model modified to account for autocorrelation are given in Fig. 2a (7 am

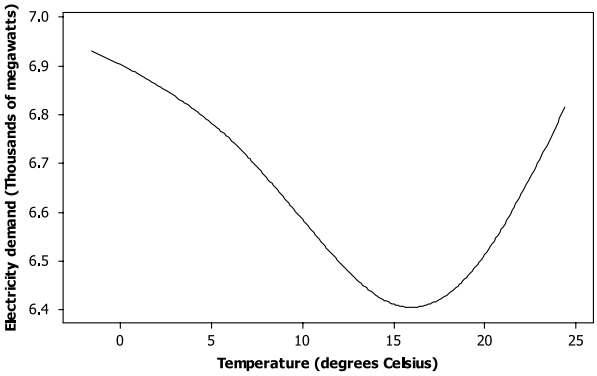

Fig. 2a. Electricity demand at 7 am.

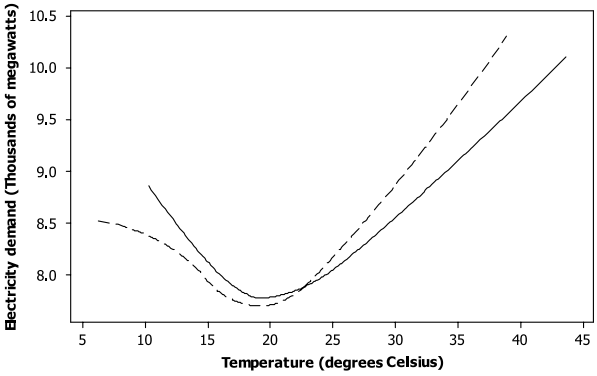

Fig. 2b. Electricity demand at 3 pm and 7 pm.

function) and Fig. 2b (the solid line is the 3 pm function and dashed line is the 7 pm function). The 7 am function is plotted separately because the y-axis scale is different due to morning electricity demand being considerably lower. The minimum of the 7 am function occurs at 16 °C while the minimum of the 3 and 7 pm functions both occur between 19 and 20 °C. Also, for lower temperatures at both 7 am and 7 pm the demand function begins to flatten out. Such a flattening out could not be modeled by a convex function.

8. Conclusion

This paper developed methodology for estimating shape constrained functions nonparametrically in models with log-concave likelihood functions. Both free-knot and fixed-knot regression spline models were used and the shape constraints included monotonicity, convexity and functions with a single minimum. The free-knot and fixed-knot estimation methods for monotonically constrained functions in a Poisson model were shown to outperform existing methods by up to 34%. In addition, the function estimates obtained using the fixed-knot model for monotonically constrained functions were shown to be consistent in the context of a Poisson model as well as in generalized additive and hazard function models.

We also developed a new MCMC slice sampler requiring only a single auxiliary variable that allows a full Bayesian analysis to be done in models with log-concave likelihood functions. The sampler was shown to converge faster than existing methods. It was developed in the context of nonparametric function estimation but given its general nature it holds promise for the implementation of other estimation techniques in different types of models.

A limitation of the current paper is that it is not possible to apply the methodology outlined in Section 2 directly to the estimation of shape constrained multivariate functions. This is an important problem in economics because imposing shape constraints such as monotonicity and concavity on the indirect utility function are fundamental to the estimation of demand and cost functions. This is discussed extensively in Varian's (1984) Microeconomics book as well as many papers in the economics literature; see Barnett and Serletis (2008) for an excellent survey of constrained function estimation methods in consumer demand modeling. A system of demand equations can be derived from the indirect utility function if regularity conditions are imposed that include monotonicity and quasi-convexity. Flexible methods for modeling such functions include second order local approximation methods (see, for example, the translog flexible functional form in Christensen et al. (1975) and the generalized Leontief functional form in Diewert (1974)), the globally regular flexible functional forms introduced by Barnett (1983) and Cooper and McLaren (1996), and the semi-parametric approaches of Gallant (1981), Gallant and Golub (1984), and Barnett and Jonas (1983).

Future research includes the possibility of generalizing the spline estimation methodology in Sections 2 and 3 to handle multivariate functions with monotonicity and quasi-convexity imposed to allow for its use in estimating cost and demand functions. A significant issue for generalizing the methodology to constrained multivariate functions is that there will be more linear constraints on the multivariate spline basis function coefficients than there are coefficients. This means the constraints cannot be imposed using the $\gamma = L\beta$ formulation in Section 2 without extensive modification. Also, a new MCMC algorithm is necessary for the Gaussian case to traverse the multivariate function space efficiently and avoid the "curse of dimensionality". Given an MCMC algorithm to implement constrained multivariate function estimation in a Gaussian model, we can generalize it to any model with a log-concave likelihood function by using the technique introduced in Section 3 of incorporating a single auxiliary variable into the algorithm.

Acknowledgements

We would like to thank the Editor, Associate Editor and two reviewers for their helpful comments and suggestions. They improved the paper considerably. Tom Shively and Paul Damien's work was partially supported by a Faculty Research grant from the McCombs School of Business at the University of Texas at Austin.

Appendix A. Rejection sampling algorithm to generate (J_j, γ_j)

This Appendix describes the rejection sampling algorithm used to generate (J_j, γ_j). First, following Chib and Greenberg's (1995) description but using our notation, the rejection sampling algorithm can be summarized as follows: Let $\pi(J_j, \gamma_j) \propto g(J_j, \gamma_j)$. Also, let $h(J_j, \gamma_j)$ be a density that can be easily sampled and suppose there is a constant c such that $g(J_j, \gamma_j) \leq ch(J_j, \gamma_j)$ for all (J_j, γ_j). Then to obtain a random variate from π,

(1) Generate a candidate (J_j, γ_j) from h and a value $u \sim Unif(0, 1)$;

(2) If $u \leq \frac{g(J_j,\gamma_j)}{ch(J_j,\gamma_j)}$, then return (J_j, γ_j); otherwise, go to (1) and repeat.

This rejection sampling algorithm holds for the combined discrete/continuous distribution $\pi(J_j, \gamma_j|\cdots)$ in our model as well as for purely continuous densities.

The function g in the rejection sampling algorithm is set to

$$g(J_j = 0) = I(v > s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi))\pi(J_j = 0)$$

and

$$g(J_j = 1, \gamma_j) = I(a_{\min} < \gamma_j < a_{\max})\pi(\gamma_j|J_j = 1)\pi(J_j = 1).$$

The approximating density function h at $J_j = 0$ is defined as

$$h(J_j = 0) = c_{app} I(v > s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi))\pi(J_j = 0)$$

where c_{app} is a constant that does not depend on J_j or γ_j. Note that $h(J_j = 0)$ is the same as $g(J_j = 0)$ except for the constant c_{app}.

To construct the approximating density function h at $(J_j = 1, \gamma_j)$, suppose we have: (1) a computationally efficient method to determine if the minimum value of $\tilde{s}(\gamma_j)$ is greater than zero for all $\gamma_j > 0$ (where $\tilde{s}(\gamma_j)$ is defined in Section 3.1); and (2) if $\tilde{s}(\gamma_j) < 0$ for some $\gamma_j > 0$, a computationally efficient method for computing bounds $b_{\min}$ and $b_{\max}$ on the values $a_{\min}$ and $a_{\max}$. Methods to accomplish (1) and (2) are discussed in Appendix B.

[]figA.1 []figA.2

Given these bounds, let

$$h(J_j = 1, \gamma_j) = c_{app} I(b_{\min} < \gamma_j < b_{\max})\pi(\gamma_j | J_j = 1)\pi(J_j = 1).$$

Note that $h(J_j = 1, \gamma_j)$ is the same as $g(J_j = 1, \gamma_j)$ up to a constant for $a_{\min} < \gamma_j < a_{\max}$. However, $g(J_j = 1, \gamma_j) = 0$ for $\gamma_j < a_{\min}$ and $\gamma_j > a_{\max}$ whereas $h(J_j = 1, \gamma_j)$ has positive values for $b_{\min} < \gamma_j < a_{\min}$ and $a_{\max} < \gamma_j < b_{\max}$.

Therefore, if $b_{\min}$ and $b_{\max}$ are close to $a_{\min}$ and $a_{\max}$, the exact and approximating functions g and h are very similar and the acceptance rate in a rejection sampling algorithm will be high. The only time the generated value will be rejected is if $J_j = 1$ and $b_{\min} \le \gamma_j < a_{\min}$ or $a_{\max} < \gamma_j \le b_{\max}$. The values $b_{\min}$ and $a_{\min}$ are often both zero so there is often no "lower end" rejection region. Also, $b_{\max}$ is typically close to $a_{\max}$ so the upper rejection region tends to be small.

To generate (J_j, γ_j) from the approximating density h we will generate J_j first, and then γ_j. J_j is generated by analytically integrating γ_j out of the expression for $h(J_j = 1, \gamma_j)$ to give $h(J_j = 1)$ and then using $h(J_j = 0)$ and $h(J_j = 1)$. If $J_j = 0$, then γ_j does not need to be generated. If $J_j = 1$ and $b_{\min} = 0$, then $\gamma_j | J_j = 1$ is generated from the mixture distribution of a point mass at zero and the normal distribution $N(0, \tau^2)$ constrained to $(0, b_{\max})$. If $b_{\min} > 0$, then $\gamma_j | J_j = 1$ is drawn from the normal distribution $N(0, \tau^2)$ constrained to $(b_{\min}, b_{\max})$.

To complete the rejection sampling algorithm, let $c = 1/c_{app}$ in the expression $u \le \frac{g(J_j, \gamma_j)}{ch(J_j, \gamma_j)}$ in step (2) of the algorithm. Then, if $J_j = 0$ is drawn

$$\begin{aligned}\frac{g(J_j = 0)}{ch(J_j = 0)} &= \frac{I\left(v > s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi)\right)\pi(J_j = 0)}{\frac{1}{c_{app}}\left[c_{app} I\left(v > s(y, \alpha, J_j = 0, J_{(-j)}, \gamma_{(-j)}, \phi)\right)\pi(J_j = 0)\right]} \\ &= 1\end{aligned}$$

and we always accept. If $J_j = 1$, we have

$$\begin{aligned}\frac{g(J_j = 1, \gamma_j)}{ch(J_j = 1, \gamma_j)} &= \frac{I\left(a_{\min} < \gamma_j < a_{\max}\right)\pi(\gamma_j | J_j = 1)\pi(J_j = 1)}{\frac{1}{c_{app}}\left[c_{app} I\left(b_{\min} < \gamma_j < b_{\max}\right)\pi(\gamma_j | J_j = 1)\pi(J_j = 1)\right]} \\ &= \begin{cases} 1 & \text{if } a_{\min} \le \gamma_j \le a_{\max} \\ 0 & \text{otherwise.} \end{cases}\end{aligned}$$

Therefore, if $a_{\min} \le \gamma_j \le a_{\max}$ we accept with probability one. Otherwise, we reject. The condition $a_{\min} \le \gamma_j \le a_{\max}$ is easily checked by computing $\tilde{s}(\gamma_j)$. If $\tilde{s}(\gamma_j) > 0$ and we reject, the generated γ_j value will be used to improve either the lower or upper bound so the computation is not "wasted"—see Appendix B for details. Given that $b_{\min}$ and $b_{\max}$ are typically very tight bounds on $a_{\min}$ and $a_{\max}$, we will almost always accept.

We note that once the code for the new algorithm is available, it is a simple matter to modify it for a wide variety of models by changing the $s(\gamma)$ function appropriately.

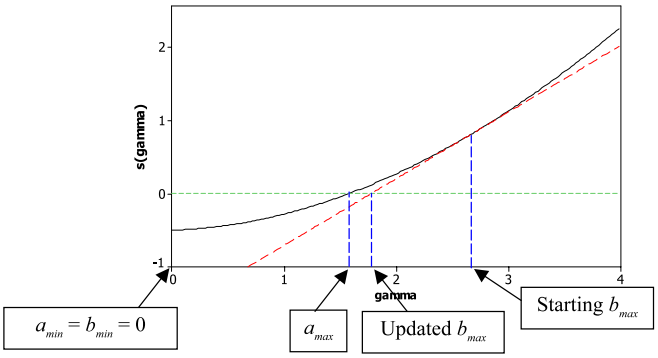

Fig. A.1. Plot of s(gamma) vs. gamma with tangent line.

Appendix B. Bounds on the roots $a_{\min}$ and $a_{\max}$

There are two cases to consider:

Case 1: The current value of J_j (denoted $J_j^{(current)}$) equals 1. In this case $\tilde{s}(\gamma_j^{(current)}) < 0$, i.e. $\gamma_j^{(current)}$ is a "feasible" value of γ_j so the function $\tilde{s}(\gamma_j)$ must have roots.

Case 2: The current value $J_j^{(current)} = 0$. In this case, the minimum value of $\tilde{s}(\gamma_j)$ may be greater than zero. If this is true, then $\pi(J_j = 1 | J_{(-j)}, \gamma_{(-j)}, \alpha, v, y) = 0$ and $J_j = 0$ with probability one. If the minimum value of $\tilde{s}(\gamma_j)$ is less than zero, then roots exist and $\pi(J_j = 1 | J_{(-j)}, \gamma_{(-j)}, \alpha, v, y)$ is positive.

Case 1: $J_j^{(current)} = 1$.

There are two possibilities:

Case 1a: The lower root of $\tilde{s}(\gamma_j)$ is $a^*_{\min} < 0$ in which case $a_{\min} = 0$;

Case 1b: The lower root of $\tilde{s}(\gamma_j)$ is $a^*_{\min} > 0$ in which case $a_{\min} = a^*_{\min} > 0$.

It is straightforward to check whether case 1a or 1b holds by computing $\tilde{s}(0)$. If $\tilde{s}(0) < 0$, then $a_{\min} = 0$ (this happens a large percentage of the time). If $\tilde{s}(0) > 0$, then zero provides an initial bound on $a_{\min}$. A bound on $a_{\max}$ needs to be computed in either situation.

Given a starting value $b_{\max}$ for the upper bound such that $\tilde{s}(b_{\max}) > 0$ and $\tilde{s}'(b_{\max}) > 0$ (such a $b_{\max}$ is easy to find—see Appendix B.1), we compute the intersection point of the tangent line at $b_{\max}$ with the γ-axis to give an updated $b_{\max}$ (which is $b_{\max} - \frac{\tilde{s}(b_{\max})}{\tilde{s}'(b_{\max})}$)—see Fig. A.1. The updated $b_{\max}$ will tend to be close to $a_{\max}$ given the nature of the function $\tilde{s}(\gamma_j)$, but it is always the case that $a_{\max}$ will be less than the updated $b_{\max}$ because $\tilde{s}(\gamma_j)$ is convex.

Now generate (J_j, γ_j) using $h(J_j, \gamma_j)$ as discussed in Appendix A. If $J_j = 0$, then it is accepted. If $J_j = 1$ so γ_j is also generated, then compute $\tilde{s}(\gamma_j)$. If $\tilde{s}(\gamma_j) < 0$, then accept $(J_j = 1, \gamma_j)$. If $\tilde{s}(\gamma_j) > 0$, then set $b_{\max} = \gamma_j$ and repeat the process.

Note that the first (J_j, γ_j) value generated is typically accepted and we almost always accept after two draws, especially if $a_{\min} = 0$. However, if the second draw is rejected, then the tangent method is used to continually update $b_{\min}$ and $b_{\max}$ until a draw is accepted. For example, if $\tilde{s}'(\gamma_j) < 0$, then the new lower bound $b_{\min} = \gamma_j - \frac{\tilde{s}(\gamma_j)}{\tilde{s}'(\gamma_j)}$ is computed.

Case 2: $J_j^{(current)} = 0$.

There are three possibilities:

Case 2a: The lower root of $\tilde{s}(\gamma_j)$ is $a^*_{\min} < 0$ in which case $a_{\min} = 0$;

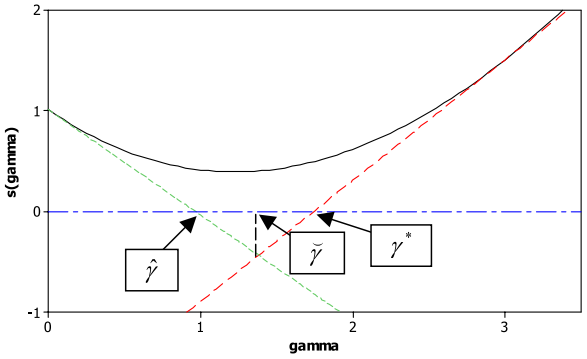

Fig. A.2. Plot of s(gamma) vs. gamma with tangent lines.

Case 2b: The lower root of $\tilde{s}(\gamma_j)$ is $a^*_{\min} > 0$ in which case $a_{\min} = a^*_{\min} > 0$;

Case 2c: The minimum value of $\tilde{s}(\gamma_j)$ is greater than zero and no roots exist.

Note that cases 2a and 2b are the same as cases 1a and 1b and can be handled the same. If case 2c occurs, then $J_j = 0$ with probability one.

The key for case 2 is using an efficient method to determine if case 2c occurs. We begin by computing $\tilde{s}(0)$. If $\tilde{s}(0) < 0$, then case 2a holds. If $\tilde{s}(0) > 0$, then compute $\tilde{s}'(0)$ and the intersection point of the tangent line at zero with the γ-axis, i.e. $\hat{\gamma} = -\frac{\tilde{s}(0)}{\tilde{s}'(0)}$ (see Fig. A.2). If $s'(\hat{\gamma}) > 0$, then the minimum value of $\tilde{s}(\gamma)$ is positive and case 2c holds.

If $\tilde{s}'(\hat{\gamma}) < 0$, then compute $\tilde{s}(b_{\max})$ and $\tilde{s}'(b_{\max})$ where $b_{\max}$ is the starting value of the upper bound as defined in Appendix B.1. If the intersection point of the tangent line at $b_{\max}$ with the γ-axis, $\gamma^* = b_{\max} - \frac{\tilde{s}(b_{\max})}{\tilde{s}'(b_{\max})}$ (see Fig. A.2), is less than zero or $\tilde{s}'(\gamma^*) < 0$, then the minimum value of $\tilde{s}(\gamma)$ is positive and case 2c holds.

If $\tilde{s}'(\hat{\gamma}) < 0$ and $\tilde{s}'(\gamma^*) > 0$, then compute the intersection point of the two tangent lines, denoted $\bar{\gamma}$ (see Fig. A.2). If $\tilde{s}(\bar{\gamma}) < 0$, then roots exist. In this case, $\hat{\gamma}$ and γ^* are used as the bounds $b_{\min}$ and $b_{\max}$ (they will typically be excellent bounds) and then proceed as in case 1b.

If $\tilde{s}(\bar{\gamma}) > 0$ (as in Fig. A.2), then compute $\tilde{s}'(\bar{\gamma})$. If $\tilde{s}'(\bar{\gamma}) < 0$, set $\hat{\gamma} = \bar{\gamma}$, otherwise set $\gamma^* = \bar{\gamma}$, and repeat the process until either $\tilde{s}(\bar{\gamma}) < 0$ (in which case roots exist), the newly computed tangent line indicates the minimum value of $\tilde{s}(\gamma)$ is greater than zero, or $\tilde{s}'(\bar{\gamma}) < 10^{-6}$ (with the latter two possibilities indicating no positive roots exist). It typically takes only a couple of iterations to determine if the minimum value of $\tilde{s}(\gamma)$ is greater than zero.

B.1. Starting value for $b_{\max}$

To obtain a starting value for $b_{\max}$ we use an idea similar in spirit to Gilks and Wild's (1995) method for finding starting values in their adaptive rejection sampling algorithm. After the roots have been found (or we find no roots exist) on a given iteration for a specified regression coefficient, we use a Gilks and Wild-type approximation to the likelihood function using the already computed values of $\tilde{s}(\gamma)$ and $\tilde{s}'(\gamma)$. The approximated likelihood is normalized so it can be treated as a density function, and then the 95th percentile value is computed. This value is used as the starting value $b_{\max}$ in the next iteration. It is important to check that $\tilde{s}'(b_{\max})$ is positive at the beginning of the next iteration. It typically is but in the few cases it is not, we continue to double $b_{\max}$ until the derivative is positive. The value $b_{\min} = 0$ is always used as the starting value for the lower bound.

Appendix C. Approximating conditional distributions in the MCMC algorithm for a free-knot spline model

This Appendix provides the approximating conditional distributions $\pi_{Approx}(J_j|\cdots)$, $\pi_{Approx}(\xi_j|J_j = 1, \ldots)$ and $\pi_{Approx}(\gamma_j|J_j = 1, \xi_j, \ldots)$ used in step (2) of the MCMC algorithm in Section 3.2. We first note that $\pi(J_j = 0|\cdots)$ is given in (10) because ξ_j and γ_j drop out of both the fixed-knot and free-knot models when $J_j = 0$. To provide an approximation to $\pi(J_j = 1|\cdots)$, let

$$\tilde{s}(\xi_j, \gamma_j) = s(\xi_j, \gamma_j; y, \alpha, J_j = 1, J_{(-j)}, \xi_{(-j)}, \gamma_{(-j)}, \phi) - v.$$

Then

$$\pi(J_j = 1, \xi_j, \gamma_j|\cdots) \propto I[\tilde{s}(\xi_j, \gamma_j) < 0]\pi(\gamma_j|J_j = 1) \times \pi(\xi_j|J_j = 1)\pi(J_j = 1)$$

and γ_j and ξ_j must be integrated out to give $\pi(J_j = 1|\cdots)$. To integrate out γ_j we follow Section 3.1 and let $a^*_{\min}(\xi_j)$ and $a_{\max}(\xi_j)$ represent the roots of $\tilde{s}(\xi_j, \gamma_j)$ for a given ξ_j and let $a_{\min}(\xi_j) = \max[0, a^*_{\min}(\xi_j)]$. Also, let $b_{\min}(\xi_j)$ and $b_{\max}(\xi_j)$ represent the bounds on $a_{\min}(\xi_j)$ and $a_{\max}(\xi_j)$ corresponding to $b_{\min}$ and $b_{\max}$ in Section 3.1. For a fixed ξ_j, these bounds can be computed similarly to $b_{\min}$ and $b_{\max}$. γ_j can now be integrated out by integrating the mixture prior $\pi(\gamma_j|J_j = 1)$ over the interval $[b_{\min}(\xi_j), b_{\max}(\xi_j)]$ to give the approximating density

$$\pi_{Approx}(J_j = 1, \xi_j|\cdots) \propto \left[\int_{b_{\min}(\xi_j)}^{b_{\max}(\xi_j)} \pi(\gamma_j|J_j = 1)\mathrm{d}\gamma_j\right] \times \pi(\xi_j|J_j = 1)\pi(J_j = 1).$$

Calculating this density requires at most two standard normal cdf calculations (depending on whether $b_{\min}(\xi_j) = 0$ or >0). For a given ξ_j, $\pi_{Approx}(J_j = 1, \xi_j|\cdots) = 0$ if $\tilde{s}(\xi_j, \gamma_j) > 0$ for all γ_j.

In the integral over ξ_j required to obtain $\pi_{Approx}(J_j = 1|\cdots)$, let ξ_L and ξ_U represent the bounding knot values for ξ_j. More specifically, let $\xi_L = \xi_i$ where i is the largest index less than j with $J_i = 1$. If all $J_i = 0$ for $i < j$, then $\xi_L = 0$. Similarly, let $\xi_U = \xi_i$ where i is the smallest index greater than j with $J_i = 1$. If all $J_i = 0$ for $i > j$, then $\xi_U = 1$. For example, if $m = 5$, $j = 3$ and $J = (0, 1, -, 0, 1)$, then $\xi_L = \xi_2$ and $\xi_U = \xi_5$. Then

$$\pi_{Approx}(J_j = 1|\cdots) = \int_{\xi_L}^{\xi_U} \pi_{Approx}(J_j = 1, \xi_j|\cdots)\mathrm{d}\xi_j.$$

This integral cannot be done analytically. However, the function $\pi_{Approx}(J_j = 1, \xi_j|\cdots)$ is well-approximated by a step function computed at ξ_L and the x_i values with $\xi_L < x_i < \xi_U$. Using this step function approximation gives $\pi_{Approx}(J_j = 1|\cdots)$ up to a constant. A temporary value of J_j can now be generated using $\pi(J_j = 0|\cdots)$ and $\pi_{Approx}(J_j = 1|\cdots)$.

If $J_j = 0$, then ξ_j and γ_j do not need to be generated. If $J_j = 1$, then a temporary value of ξ_j is generated from the step function approximation

$$\pi_{Approx}(\xi_j|J_j = 1, \ldots) \propto \pi_{Approx}(J_j = 1, \xi_j|\cdots).$$

Finally, $\gamma_j|J_j = 1, \xi_j, \ldots$ is generated from a constrained normal distribution with bounds $b_{\min}(\xi_j)$ and $b_{\max}(\xi_j)$.

The true density function value at the value (J_j, ξ_j, γ_j) is

$$\pi(J_j, \xi_j, \gamma_j|\cdots) \propto \mathrm{e}^{-v} I(v > s(y, \alpha, J, \xi_J, \gamma_J, \phi)) \times \pi(\gamma_j|J_j)\pi(\xi_j|J_j)\pi(J_j).$$

This is straightforward to compute to obtain the Metropolis–Hastings acceptance probability and therefore the generated value of $(J_j, \xi_j, \gamma_j)|\cdots$.

Appendix D. Consistency proof for generalized additive and hazard function models

This Appendix generalizes the argument in Section 6 for the Poisson model to show consistency for the class of generalized additive and hazard function models. The more general argument uses the existence of the maximum likelihood estimator (MLE) when working with log-concave densities and relies on the properties of the MLE.

We consider the case when $p(y|x) = \exp\{s(y, f(x), \phi)\}$ and s is concave in $\xi = (\phi, f)$. Under this scenario it is that $\hat{\xi}$ exists and is unique (see Walther, 2002); where $\hat{\xi}$ maximizes

$$\sum_{i=1}^{n} s(y_i, f(x_i), \phi).$$

Now, as before, define

$$d(\xi, \xi_0) = \int d_H(p(\cdot|\xi, x), p(\cdot|\xi_0, x))G_0(\mathrm{d}x),$$

and so we can obtain

$$\begin{aligned}\Pi_n(A_\varepsilon) &= \Pi(A_\varepsilon|(x_1, y_1), \ldots, (x_n, y_n)) \\ &\leq \left\{\prod_{i=1}^{n} \frac{p(y_i|\hat{\xi}, x_i)}{p(y_i|\xi_0, x_i)}\right\}^{1/2} \int_{A_\varepsilon} \left\{\prod_{i=1}^{n} \frac{p(y_i|\xi, x_i)}{p(y_i|\xi_0, x_i)}\right\}^{1/2} \Pi(\mathrm{d}\xi) \\ &\quad \Big/ \int \prod_{i=1}^{n} \frac{p(y_i|\xi, x_i)}{p(y_i|\xi_0, x_i)} \Pi(\mathrm{d}\xi).\end{aligned}$$

We can now use a result in Walker and Hjort (2001), Section 3.3, to demonstrate that $\Pi_n(A_\varepsilon) \to 0$ a.s. for all $\varepsilon > 0$, with the classical consistency condition, that

$$n^{-1} \sum_{i=1}^{n} \{s(y_i, \hat{\xi}, x_i) - s(y_i, \xi_0, x_i)\} \to 0 \quad \text{a.s.}$$

This condition guarantees that

$$\left\{\prod_{i=1}^{n} \frac{p(y_i|\hat{\xi}, x_i)}{p(y_i|\xi_0, x_i)}\right\}^{1/2} \leq \mathrm{e}^{nd}$$

a.s. for all large n for any $d > 0$. A usual Kullback–Leibler support condition ensures that

$$\int \prod_{i=1}^{n} \frac{p(y_i|\xi, x_i)}{p(y_i|\xi_0, x_i)} \Pi(\mathrm{d}\xi) > \mathrm{e}^{-nc}$$

a.s. for all large n for any $c > 0$. Finally, taking expectations and using a Markov inequality combined with Borel–Cantelli it is easy to show that

$$\int_{A_\varepsilon} \left\{\prod_{i=1}^{n} \frac{p(y_i|\xi, x_i)}{p(y_i|\xi_0, x_i)}\right\}^{1/2} \Pi(\mathrm{d}\xi) < \mathrm{e}^{-n\delta_\varepsilon}$$

a.s. for all large n for some $\delta_\varepsilon > 0$. Putting all these together we can see that $\Pi_n(A_\varepsilon) \to 0$ a.s.

Now let P_n be the empirical distribution of the $(x_i, y_i)_{i=1}^n$ and P_0 the true distribution of (x, y). Then the above classical consistency condition holds when

$$\int |s(y, \hat{\xi}, x) - s(y, \xi_0, x)| dP_0(x, y) \to 0 \quad \text{a.s.}$$

and

$$\sup_{\xi} \left| \int g(y, x, \xi) d(P_n - P_0) \right| \to 0 \quad \text{a.s.}$$

where $g(x, y, \xi) = s(y, f(x), \phi) - s(y, f_0(x), \phi_0)$. The former condition is straightforward under suitable continuity conditions for s. The latter result is a uniform law of large number criterion on which there is an abundance of literature. For early work see Pollard (1984) and Giné and Zinn (1984).

References

Abbring, J.H., van den Berg, G.J., 2003. The nonparametric identification of treatment effects in duration models. Econometrica 71, 1491–1517.

Albert, J., Chib, S., 1993. Bayesian analysis of binary and polychotomous response data. Journal of the American Statistical Association 88, 669–679.

Aït-Sahalia, Y., Duarte, J., 2003. Nonparametric option pricing under shape restrictions. Journal of Econometrics 116, 9–47.

Banerjee, M., Mukherjee, D., Mishra, S., 2009. Semiparametric binary regression models under shape constraints with an application to Indian schooling data. Journal of Econometrics 149, 101–117.

Barnett, W.A., 1983. New indices of money supply and the flexible Laurent demand systems. Journal of Business and Economic Statistics 1, 7–23. Reprinted in Barnett, W.A. J. Binner (Eds.), Functional Structure and Approximation in Econometrics, Elsevier, Amsterdam, 2004.

Barnett, W.A., Jonas, A., 1983. The Müntz–Szatz demand system: an application of a globally well behaved series expansion. Economics Letters 11, 337–342. Reprinted in Barnett, W.A. J. Binner (Eds.). Functional Structure and Approximation in Econometrics, Elsevier, Amsterdam, 2004.

Barnett, W.A., Serletis, A., 2008. Consumer preferences and demand systems. Journal of Econometrics 147, 210–224.

Bharath, S.T., Shumway, T., 2008. Forecasting default with the Merton distance to default model. Review of Financial Studies 21, 1339–1369.

Broadie, M., Detemple, J., Ghysels, E., Torrés, O., 2000a. Nonparametric estimation of American options' exercise boundaries and call prices. Journal of Economic Dynamics and Control 24, 1829–1857.

Broadie, M., Detemple, J., Ghysels, E., Torrés, O., 2000b. American options with stochastic dividends and volatility: a nonparametric investigation. Journal of Econometrics 94, 53–92.

Chib, S., Greenberg, E., 1995. Understanding the Metropolis–Hastings algorithm. American Statistician 49, 327–335.

Christensen, L.R., Jorgenson, D.W., Lau, L.J., 1975. Transcendental logarithmic utility functions. American Economic Review 70, 422–432.

Cooper, R.J., McLaren, K.R., 1996. A system of demand equations satisfying effectively global regularity conditions. Review of Economics and Statistics 78, 359–364.

Cox, D.R., 1972. Regression models and life tables (with discussion). Journal of the Royal Statistical Society, Series B 34, 187–220.

Cripps, E., Carter, C., Kohn, R., 2005. Variable selection and covariance selection in multivariate regression models. In: Dey, D.K., Rao, C.R. (Eds.), Handbook of Statistics. In: Bayesian Thinking: Modeling and Computation, vol. 25. Elsevier, North-Holland, Amsterdam, pp. 519–552.

Damien, P., Wakefield, J., Walker, S.G., 1999. Gibbs sampling for Bayesian non-conjugate and hierarchical models by using auxiliary variables. Journal of the Royal Statistical Society, Series B 61, 331–344.

Denison, D.G.T., Malick, B.K., Smith, A.F.M., 1998. Automatic Bayesian curve fitting. Journal of the Royal Statistical Society, Series B 60, 333–350.

Diewert, W.E., 1974. Applications of duality theory. In: Intriligator, M., Kendrick, D. (Eds.), Frontiers in Quantitative Economics, vol. 2. North-Holland, Amsterdam.

Duffie, D., Saita, L., Wang, K., 2007. Multi-period corporate default prediction with stochastic covariates. Journal of Financial Economics 83, 635–665.

Dunson, D.B., 2005. Bayesian semiparametric isotonic regression for count data. Journal of the American Statistical Association 100, 618–627.

Friedman, J., Tibshirani, R., 1984. The monotone smoothing of scatterplots. Technometrics 26, 243–250.

Gallant, A.R., 1981. On the bias of flexible functional forms and an essentially unbiased form: the Fourier functional form. Journal of Econometrics 15, 211–245.

Gallant, A.R., Golub, G.H., 1984. Imposing curvature restrictions on flexible functional forms. Journal of Econometrics 26, 295–321.

Gamerman, D., Lopes, H.F., 2006. Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, 2nd ed. Springer, New York.

Gelfand, A.E., Smith, A.F.M., 1990. Sampling-based approaches to calculating marginal densities. Journal of the American Statistical Association 85, 398–409.

Gilks, W.R., Wild, P., 1995. Adaptive rejection sampling for Gibbs sampling. Applied Statistics 41, 337–348.

Giné, E., Zinn, J., 1984. On the central limit theorem for empirical processes. Annals of Probability 12, 929–989.

Groeneboom, P., Jongblood, G., Wellner, J.A., 2001. Estimation of a convex function: characterizations and asymptotic theory. Annals of Statistics 29, 1653–1698.

Ihara, T., Genchi, Y., Sato, T., Yamaguchi, K., Endo, Y., 2008. City-block-scale sensitivity of electricity consumption to air temperature and air humidity in business districts of Tokyo, Japan. Energy 33, 1634–1645.

Jenkins, S.P., Garcia-Serrano, C., 2004. The relationship between unemployment benefits and re-employment probabilities: evidence from Spain. Oxford Bulletin of Economics and Statistics 66, 239–260.

Kim, J., Marschke, G., 2005. Labor mobility of scientists, technological diffusion, and the firm's patenting decision. RAND Journal of Economics 36, 298–317.

Kweon, Y., Kockleman, K.M., 2005. Safety effects of speed limit changes. Transportation Research Record 1908, 148–158.

Mammen, E., 1991. Estimating a smooth monotone regression function. Annals of Statistics 19, 724–740.

Manski, C.F., Tamer, E., 2002. Inference in regression with interval data on a regressor or outcome. Econometrica 70, 519–546.

Mitra, D., Golder, P.N., 2002. Whose culture matters? Near-market knowledge and its impact on foreign market entry. Journal of Marketing Research 39, 350–365.

Neelon, B., Dunson, D.B., 2004. Bayesian isotonic regression and trend analysis. Biometrics 60, 398–406.

Panagiotelis, A., Smith, M., 2008. Bayesian identification, selection and estimation of semiparametric functions in high-dimensional additive models. Journal of Econometrics 143, 291–316.
Pardo, A., Meneu, V., Valor, E., 2002. Temperature and seasonality influences on Spanish electricity load. Energy Economics 24, 55–70.
Pollard, D., 1984. Convergence of Stochastic Processes. Springer, New York.
Psiloglou, B.E., Giannakopoulos, C., Majithia, S., Petrakis, M., 2009. Factors affecting electricity demand in Athens, Greece and London, UK: a comparative assessment. Energy 34, 1855–1863.
Schipper, M., Taylor, J.M.G., Lin, X., 2007. Bayesian generalized monotonic functional mixed models for the effects of radiation dose histograms on normal tissue complications. Statistics in Medicine 26, 4643–4656.
Shively, T.S., Sager, T.W., Walker, S.G., 2009. A Bayesian approach to nonparametric monotone function estimation. Journal of the Royal Statistical Society, Series B 71, 159–175.
Smith, M., Kohn, R., 1996. Nonparametric regression using Bayesian variable selection. Journal of Econometrics 75, 317–343.
Smith, M., Wong, C., Kohn, R., 1998. Additive nonparametric regression with autocorrelated errors. Journal of the Royal Statistical Society, Series B 60, 311–331.
Tierney, L., 1994. Markov chains for exploring posterior distributions. Annals of Statistics 22, 1701–1762.
Varian, H., 1984. Microeconomic Analysis, 2nd ed. Norton, Company, New York.
Walker, S.G., 2004. New approaches to Bayesian consistency. Annals of Statistics 32, 2028–2043.
Walker, S.G., Hjort, N.L., 2001. On Bayesian consistency. Journal of the Royal Statistical Society, Series B 63, 811–821.
Walther, G., 2002. Detecting the presence of mixing with multiscale maximum likelihood. Journal of the American Statistical Association 97, 508–514.
Wright, I., Wegman, E., 1980. Isotonic, convex, and related splines. Annals of Statistics 8, 1023–1035.
Yatchew, A., Härdle, W., 2006. Nonparametric state price density estimation using constrained least squares and the bootstrap. Journal of Econometrics 133, 579–599.