Fast calibrated additive quantile regression

Matteo Fasiolo[1,†], Yannig Goude[2], Raphael Nedellec[2], and Simon N. Wood[1]

[1]School of Mathematics, University of Bristol, United Kingdom.
[2]EDF R&D, Palaiseau, France.
[†]Correspondence: matteo.fasiolo@bristol.ac.uk

January 4, 2017

Abstract

Quantile regression represents a flexible approach for modelling the impact of several covariates on the conditional distribution of the dependent variable, without making any parametric distributional assumption. In this work we develop Bayesian methodology for fitting additive quantile regression models, based on penalized splines. Gaussian random effects and parametric terms may also be included. The key objectives of our proposal are to achieve: a) fast estimation of the regression coefficients; b) fast automatic smoothing parameters selection; c) well-calibrated posterior credible intervals. Traditional quantile regression methods are based on the pinball loss or, equivalently, on the Asymmetric Laplace (AL) distribution, whose non-differentiability makes the first two goals difficult to achieve. Hence, the approach proposed here is based on a novel smooth loss function, which corresponds to a generalization of the AL density. This more tractable loss is related to kernel quantile estimators, which have favourable statistical properties relative to empirical quantile estimators. We show that, if this loss is adopted, the regression coefficients can be estimated using fast, well-known, methods. Further we show that, by integrating these methods with a novel calibration-based approach, it is also possible to select the smoothing parameters in an automatic and robust fashion, and to obtain reliable uncertainty estimates, even in the tails or under heteroscedasticity. Our work was motivated by a probabilistic electricity load forecasting application, which we use here to demonstrate the proposed approach. The methods described in this paper are implemented by the `qgam` R package, which can be found at https://github.com/mfasiolo/qgam.

Keywords: Quantile Regression; Generalized Additive Models; Smoothing; Penalized Regression Splines; Kernel Quantile Regression; Calibrated Bayes.

1 Introduction

The purpose of this work is developing fast, well-calibrated and automated Bayesian methodology for fitting additive quantile regression models. Computational efficiency and automation are achieved by adopting fast optimization methods for parameter estimation and smoothing parameter selection. In particular, for fixed smoothing parameters, the regression coefficients are estimated using either Penalized Iteratively Re-weighted Least Squares (PIRLS) or Newton's algorithm. The smoothing parameters themselves are selected by minimizing a marginal loss criterion. This is is done efficiently by an outer Newton's iteration, which uses implicit differentiation to obtain the derivatives of the estimated regression coefficients with respect to the smoothing parameters. These methods

are integrated with a simulation-based procedure, which calibrates the posterior credible intervals of the fitted quantiles. As the examples will show, this results in robust quantile estimates whose credible intervals approximately achieve nominal coverage levels. The whole procedure requires essentially no manual tuning, and it allows to fit models whose components can be parametric, random effects or spline-based non-parametric smooths of an arbitrary number of covariates.

This is an advance relative to existing methods, because stable and computationally efficient methods implementing non-parametric additive quantile regression are otherwise lacking. For instance, the `quantreg` R package, which is based on the methods of Koenker (2013), only permits additive models whose smooth terms are at most bi-dimensional, and it requires users to select the smoothing parameters manually. On the other hand, the gradient boosting quantile regression method implemented by the `mboost` R package (Hothorn et al., 2010) does not limit the dimensionality of the smooth terms, but it requires users to manually choose the degrees of freedom used by each base model. In addition, `mboost` uses bootstrapping to estimate parameter uncertainty, while the approach proposed here quantifies uncertainty using analytic approximations. Yue and Rue (2011) and Waldmann et al. (2013) describe how semi-parametric additive quantile regression can be fitted, if a Bayesian framework is adopted. Unfortunately, the second proposal does not seem to be, at the time of writing, readily available in software. The method of Yue and Rue (2011) is instead available within the `INLA` software (Martins et al., 2013), but its use is currently discouraged. The `vgam` R package (Yee, 2008) provides method for fitting additive quantile regression models, but also in this case the complexity of the smooth terms is determined manually. The work of Lin et al. (2013) is not an alternative to what we propose here, because their focus is variable selection, rather than smoothing.

In order to set quantile regression in a Bayesian context, we adopt the general belief-updating framework of Bissiri et al. (2013). This is because quantile regression is traditionally based on the pinball loss (Koenker, 2005), and not on a probabilistic model for the observations density, $p(y|\mathbf{x})$, which impedes direct application of Bayes's rule. Bissiri et al. (2013) provide a coherent framework for updating prior beliefs using a loss function, rather than the likelihood, to obtain a posterior density. Under the pinball loss, this approach leads to the adoption of an Asymmetric Laplace (AL) model for $p(y|\mathbf{x})$, as originally suggested by Yu and Moyeed (2001). Using this model, Bayesian inference can be performed using Markov chain Monte Carlo (MCMC) methods, as in Waldmann et al. (2013). However, basing quantile regression on the pinball loss or on the corresponding AL density, limits computational efficiency. In fact, this loss is piecewise linear, which impedes the use of computationally efficient fitting methods, designed to work with differentiable, strongly convex functions. Yue and Rue (2011) and Oh et al. (2012) address this issue by proposing smooth approximations to, respectively, the AL density and the pinball loss. Here we derive a new loss function by embedding AL in a family which, while being differentiable and log-concave, generalizes the AL distribution. Interestingly, the new loss is related to kernel quantile estimation methods and results from that literature suggest that the corresponding quantile estimator, beside being computationally tractable, might also be statistical superior to that obtained by minimizing the pinball loss.

The framework of Bissiri et al. (2013) introduces an extra parameter, called learning rate, which determines the relative weight of the prior and the loss within the posterior. In the context of quantile regression, this parameter is completely confounded with the scale of the AL density. The scale parameter of the new loss plays a similar role and it is important to select it carefully, because it controls both the wiggliness of the fitted quantile and the width of posterior credible intervals. Here the learning rate is selected automatically by matching the credible intervals of the conditional quantiles, derived using an asymptotic approximation of the posterior, to their empirical distribution. This approach leads to robustness in smoothing parameter selection, and to credible intervals that roughly achieve nominal coverage levels. In contrast, Waldmann et al. (2013) show

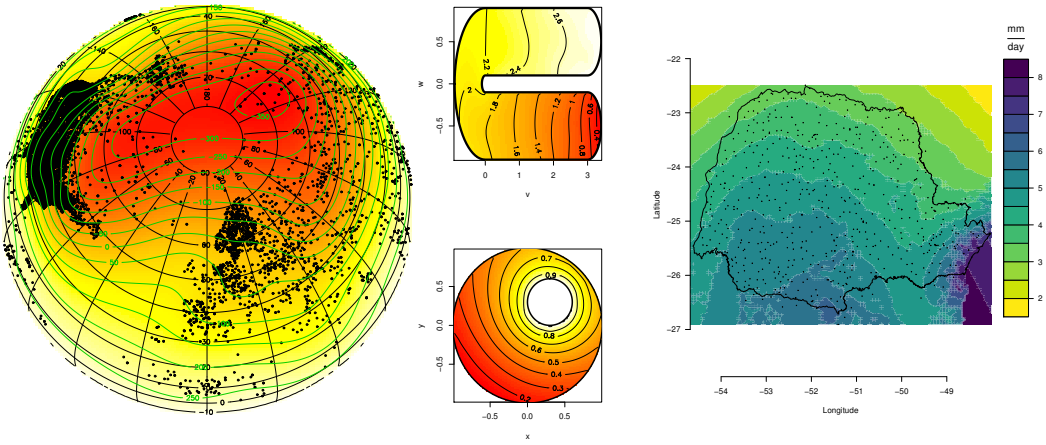

Figure 1: Some examples of the smooth components that may be included in the additive quantile regression models which can be fitted using the approach proposed here. Left: effect of spatial location, defined using splines on the sphere, on quantile $\tau = 0.1$ of minimum daily temperatures, estimated using the Global Historical Climatology Network (GHCN) dataset (Menne et al., 2012). The Gulf Stream is visible. Centre: finite area spatial components, based on soap film smoothers, of two GAM fits for $\tau = 0.5$. The data is simulated. Right: sum of the effects of spatial location, defined using an isotropic thin-plate spline basis, distance from the ocean and elevation, on quantile $\tau = 0.9$ of average weekly rainfall in Paraná state, Brazil. The dataset is available within the `R-INLA` R package (Lindgren and Rue, 2015).

that, if the AL density is adopted naively, it is difficult to achieve adequate coverage, even when MCMC methods are used.

The rest of the paper is organized as follows. In Section 2 we briefly review additive quantile regression and how it can be set in a Bayesian context using the framework of Bissiri et al. (2013). Section 3 describes the new loss function and its relation to kernel quantile estimators. In Section 4 we explain how additive quantile regression models can be fitted efficiently, if the new loss function is adopted. We propose a simulation-based approach for calibrating posterior credible intervals in Section 5, and we test it on simulated examples in Section 6. In Section 7 we demonstrate the performance of the proposed approach in the context of probabilistic electricity load forecasting.

2 Background

2.1 Quantile regression basics

Quantile regression aims at modelling the τ-th quantile (where $\tau \in (0, 1)$) of the response, y, conditionally on a d-dimensional vector of covariates, $\mathbf{x}$. More precisely, if $F(y|\mathbf{x})$ is the conditional c.d.f. of y, then the τ-th conditional quantile is

$$\mu = F^{-1}(\tau|\mathbf{x}) = \inf\{y : F(y|\mathbf{x}) \geq \tau\}.$$

The τ-th conditional quantile can also be defined as the minimizer of the expected loss

$$L(\mu|\mathbf{x}) = \mathbb{E}\{\ \rho_\tau(y - \mu)|\mathbf{x}\} = \int \rho_\tau(y - \mu) dF(y|\mathbf{x}), \tag{1}$$

w.r.t. $\mu = \mu(\mathbf{x})$, where

$$\rho_\tau(z) = (\tau - 1)z\mathbb{1}(z < 0) + \tau z\mathbb{1}(z \geq 0), \tag{2}$$

is the so-called pinball loss. Hence, given a sample of size n, one approximates $dF(y)$ with its empirical version, $dF_n(y)$, which leads to the quantile estimator

$$\hat{\mu} = \underset{\mu}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau\{y_i - \mu(\mathbf{x}_i)\},$$

where $\mathbf{x}_i$ is the i-the vector of covariates.

In this work we assume that $\mu(\mathbf{x})$ has an additive structure, such as (XXX Simon: I am not sure my explanation is good here)

$$\mu(\mathbf{x}) = \sum_{j=1}^{m} f_j(\mathbf{x}),$$

where all the m additive effects are smooth functions, defined in terms of spline bases. For instance, a marginal effect could be

$$f_j(\mathbf{x}) = \sum_{i=1}^{r} \beta_{ji} b_{ji}(x_j),$$

where β_{ji} are unknown coefficients and $b_{ji}(x_j)$ are known spline basis functions. Analogous expressions can be used to define the joint smooths. Notice that, in this framework, $\mu(\mathbf{x})$ is really $\mu(\mathbf{x}, \boldsymbol{\beta})$. See Wood (2006) for an introduction to additive models.

2.2 Bayesian quantile regression via coherent belief-updating

In order to set quantile regression in a Bayesian framework, we need a mechanism for updating a prior, $p(\boldsymbol{\beta})$, on the regression coefficients to the corresponding posterior, $p(\boldsymbol{\beta}|\mathbf{y})$. Typically this is achieved by specifying a likelihood function and by applying Bayes' rule. The difficulty here is that quantile regression is based on the pinball loss, not on a probabilistic model for the observation density, $p(y|\boldsymbol{\beta})$. Hence, the likelihood function is missing, which impedes the application of Bayes' rule. Fortunately, this obstacle can be overcome by exploiting the general belief-updating framework of Bissiri et al. (2013). In particular, within this framework a prior belief distribution can be updated to produce a posterior while using a loss function, rather than a full likelihood, to connect model parameters to the data. Before showing how this applies to quantile regression, we briefly outline the framework in its general form.

Assume that we are interested in finding the vector of model parameters $\boldsymbol{\beta}$ minimizing

$$\mathbb{E}\{L(\boldsymbol{\beta})\} = \int L(y, \boldsymbol{\beta}) f(y) dy, \tag{3}$$

where $L(\cdot, \cdot)$ is a general loss function and $f(y)$ is the p.d.f. of y. Suppose that we have a prior believe about $\boldsymbol{\beta}$, which is quantified by the prior density $p(\boldsymbol{\beta})$. Then Bissiri et al. (2013) argue that, given some data y, a coherent approach to updating $p(\boldsymbol{\beta})$ is represented by the posterior

$$p(\boldsymbol{\beta}|y) = \frac{e^{-L(y,\boldsymbol{\beta})} p(\boldsymbol{\beta})}{\int e^{-L(y,\boldsymbol{\beta})} p(\boldsymbol{\beta})\, d\boldsymbol{\beta}}.$$

When multiple samples, $\mathbf{y} = \{y_1, \ldots, y_n\}$, are available this becomes

$$p(\boldsymbol{\beta}|\mathbf{y}) = \frac{e^{-\sum_{i=1}^{n} L(y_i,\boldsymbol{\beta})} p(\boldsymbol{\beta})}{\int e^{-\sum_{i=1}^{n} L(y_i,\boldsymbol{\beta})} p(\boldsymbol{\beta})\, d\boldsymbol{\beta}}, \tag{4}$$

where $-\sum_{i=1}^{n} L(y_i, \boldsymbol{\beta})$ is an estimate of (3). In addition, $p(\boldsymbol{\beta}|\mathbf{y})$ often includes a so-called "learning rate" $\nu > 0$, which determines the relative weight of the loss and of the prior. One way of setting up a scaled posterior is

$$p(\boldsymbol{\beta}|\mathbf{y}) \propto \nu e^{-\nu \sum_{i=1}^{n} L(y_i, \boldsymbol{\beta})} p(\boldsymbol{\beta}). \tag{5}$$

Similarly to Syring and Martin (2015) we call (5) the scaled "Gibbs posterior", while we refer to the negative of its normalizing constant as the "marginal loss".

Quantile regression, which is based on the pinball loss, fits squarely into this framework. In fact, if we let $\nu = 1/\sigma$, with $\sigma > 0$, we have

$$p(\boldsymbol{\beta}|\mathbf{y}) \propto \prod_{i=1}^{n} p_{AL}\{y_i|\mu(\mathbf{x}_i), \sigma, \tau\}\, p(\boldsymbol{\beta}), \tag{6}$$

where $\mu(\mathbf{x})$ implicitly depends of $\boldsymbol{\beta}$ and

$$p_{AL}(y|\mu, \sigma, \tau) = \frac{\tau(1-\tau)}{\sigma} \exp\Big\{ -\rho_\tau\Big(\frac{y-\mu}{\sigma}\Big)\Big\}, \tag{7}$$

is the Asymmetric Laplace (AL) density with location μ, scale σ and asymmetry parameter τ. Notice that the negative AL log-density is proportional to the pinball loss, which is the reason why Yu and Moyeed (2001) originally proposed to base Bayesian quantile regression on this density.

Having defined a belief-updating rule based on the pinball loss, it is necessary to specify the prior on the regression coefficients. In a penalized spline regression context the prior on $\boldsymbol{\beta}$ is often a Gaussian density, centred at the origin, and with a positive semi-definite covariance matrix. Given such a prior, the regression coefficients could, in theory, be estimated by maximizing the corresponding Gibbs posterior. The difficulty with this approach is that p_{AL} is non-differentiable at its mode, while $\log p_{AL}$ is piecewise linear. Hence standard optimizers, such as Newton's algorithm, cannot be used, because they require differentiable, strongly convex objective functions. In Section 3 we address this issue by proposing a smooth generalization of the AL density.

As the examples will show, selecting σ correctly is of critical importance for the purpose of obtaining smooths quantile fits and reliable uncertainty estimates. Relative to the proposal Yu and Moyeed (2001), the framework proposed here allows for further flexibility when dealing with this parameter. In fact, we model σ as follows

$$\sigma(\mathbf{x}) = \exp\{\sigma_0\} \exp\Big\{ \sum_{j=1}^{m} f_j(\mathbf{x})\Big\}, \tag{8}$$

where σ_0 is scalar and the additive effects $f_j(\cdot)$ are smooth functions, described in terms of splines bases. Notice that $\sigma(\mathbf{x})$ plays a double role in our context: it can be seen as the reciprocal of the learning rate or as the scale parameter of the AL density. We select $\sigma(\mathbf{x})$ using an hybrid approach, which exploits both interpretation. In particular, σ_0 is selected using an outer iteration, which approximately calibrates the posterior credible intervals of the conditional quantile, $\mu(\mathbf{x})$. For fixed σ_0, the effects $f_j(\cdot)$ are determined using the efficient methods described in Section 4.

3 Generalizing the Asymmetric Laplace density

The next section briefly introduces a smooth generalization of the AL density. Section 3.2 then provides more details regarding the interpretation of the new density and its use within the context of quantile regression.

3.1 The Extended Log-F (ELF) density

We consider the family of densities with exponential tails described by Jones (2008)

$$p_G(y|\alpha,\beta) = K_G^{-1}(\alpha,\beta)\exp\big\{\alpha y - (\alpha+\beta)G^{[2]}(y)\big\},$$

where $\alpha,\beta>0$, $K_G(\alpha,\beta)$ is a normalizing constant,

$$G^{[2]}(y) = \int_{-\infty}^{y}\int_{-\infty}^{t} g(z)dzdt = \int_{-\infty}^{y} G(t)dt,$$

while $g(z)$ and $G(z)$ are, respectively, the p.d.f and c.d.f. of a (fictitious) r.v. z. Importantly, this family nests the AL distribution, which is recovered by choosing $g(z)$ to be the Dirac delta and by imposing $\alpha=1-\tau$, $\beta=\tau$, with $0<\tau<1$. Adding location and scale parameters is trivial.

Obviously, we do not aim at recovering p_{AL} (which is non-differentiable) exactly, but we propose to substitute the Dirac delta with a smoother p.d.f.. We achieve this by choosing $G(z) = G(z|\lambda) = \exp(z/\lambda)/\{1+\exp(z/\lambda)\}$, which is the c.d.f. of a logistic random variable centered at zero and with scale λ. Notice that, as $\lambda\to 0$, we have that $G(z|\lambda)\to\mathbb{1}(z>0)$ which is the c.d.f. corresponding to the Dirac delta density. With this choice we have $G^{[2]}(y|\lambda) = \lambda\log\{1+\exp(y/\lambda)\}$, which leads to

$$p_F(y|\tau,\lambda) = \frac{e^{(1-\tau)y}(1+e^{\frac{y}{\lambda}})^{-\lambda}}{\lambda\text{Beta}\big[\lambda(1-\tau),\lambda\tau\big]}. \tag{9}$$

where $\text{Beta}(\cdot,\cdot)$ is the beta function. The location-scale extension of (9) is simply

$$\tilde{p}_F(y|\mu,\sigma,\tau,\lambda) = \frac{1}{\sigma}p_F\big\{(y-\mu)/\sigma|\tau,\lambda\big\} = \frac{e^{(1-\tau)\frac{y-\mu}{\sigma}}(1+e^{\frac{y-\mu}{\lambda\sigma}})^{-\lambda}}{\lambda\sigma\text{Beta}\big[\lambda(1-\tau),\lambda\tau\big]}, \tag{10}$$

We refer to (10) as the Extended Log-F (ELF) density, because imposing $\lambda=1$ leads to the log-F density described by Jones (2008). Appendix D contains additional details regarding the new density. Most of these are necessary to fit semi-parametric additive models, using the methods described in Section 4.

3.2 Motivating and interpreting the ELF density

Similarly to the log-F density of Jones (2008), the ELF density is related to kernel quantile estimation methods. Indeed, equating to zero the first derivative of the ELF log-likelihood w.r.t. μ leads to

$$\frac{1}{n}\sum_{i=1}^{n} G(y_i|\mu,\lambda\sigma) = 1-\tau, \tag{11}$$

whose solution, $\hat{\mu}$, is a standard inversion kernel quantile estimator at $1-\tau$ (Jones and Yu, 2007). In fact, the l.h.s. of (11) is a logistic kernel estimator of the c.d.f., with bandwidth $\lambda\sigma$. As $\lambda\sigma\to 0$, the ELF density converges to the AL density, which leads to the empirical c.d.f. estimator (Cheng et al., 2006). Read (1972) proves this estimator to be inadmissible w.r.t. the integrated squared loss, while Falk (1984) finds the corresponding empirical quantile estimator to be asymptotically inferior to kernel estimators, in terms of relative deficiency. In addition, Cheng et al. (2006) provide considerable empirical evidence in favour of kernel estimators. Hence, trying to approximate the pinball loss as closely as possible, by choosing $\lambda\sigma\approx 0$, leads to a quantile estimator that, besides being difficult to compute algorithmically, is also statistically sub-optimal. Given that the ELF distribution, with appropriately chosen bandwidth $\lambda\sigma$, is statistically superior to the AL

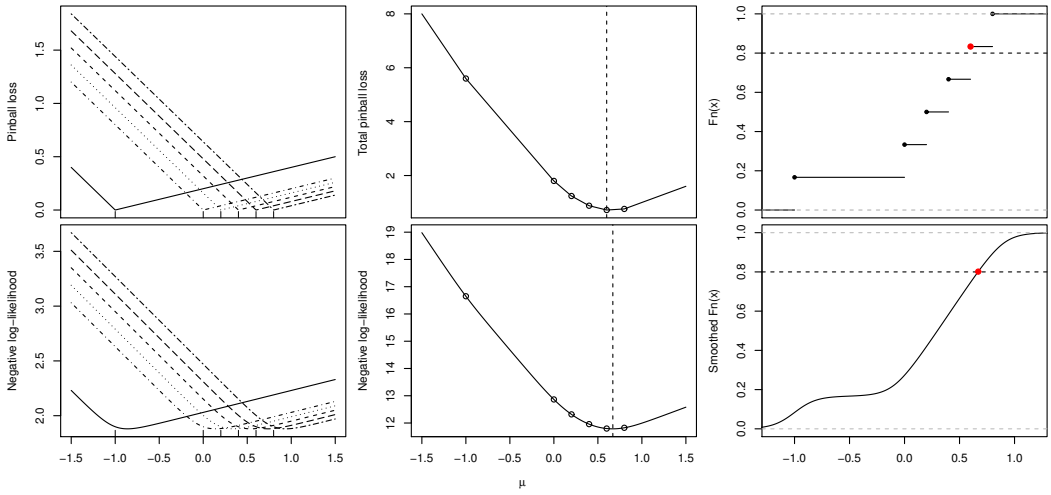

Figure 2: Left: individual pinball losses (top) and negative ELF log-densities (bottom) of six sample points, with $\tau = 0.8$. Centre: corresponding total pinball loss, which is piecewise linear, and negative ELF log-likelihood, which is continuously differentiable, with minima indicated by the dashed lines. Right: minimizing the pinball loss or the ELF negative log-likelihood is equivalent to using, respectively, the empirical c.d.f. or its kernel smoothed version to estimate the quantile location.

distribution for the purpose of quantile estimation, we consider the former distribution to be a generalization of, rather than an approximation to, the latter.

Notice that, while the product of λ and σ determines the bandwidth of the kernel estimator, these parameters are not interchangeable in our penalized regression framework. Hence, while in Section 5 we propose a method for selecting σ, in the rest of this section we assume σ to be fixed, and we focus on determining λ. To do this, we firstly quantify the asymptotic bias of the proposed quantile estimator, measured on the cumulative probability scale.

Theorem 3.1. *For fixed τ, let μ_0 be the corresponding true quantile and let μ^* be the minimizer of*

$$\tilde{L}(\mu) = \mathbb{E}\big[-\log \tilde{p}_F(y|\mu, \sigma, \tau, \lambda)\big].$$

Indicate with $f(y)$ and $F(y)$ the p.d.f. and c.d.f. of y. Then

$$|F(\mu^*) - F(\mu_0)| \leq 2\log(2)\lambda\sigma \sup_y f(y).$$

Proof. See Appendix A. □

Theorem 3.1 makes it clear that obtaining consistent quantile estimates generally requires reducing λ, as $n \to \infty$. But, to maintain computational stability, λ cannot be decreased too rapidly. In fact, as will be explained in Section 4.1, the curvature

$$\frac{\partial^2 \log \tilde{p}_F(y_i|\mu, \sigma, \tau, \lambda)}{\partial \mu^2} = -\frac{1}{\sigma} g(y|\mu, \lambda\sigma), \quad \text{for } i = 1, \ldots, n,$$

where $g(y|\mu, \lambda\sigma)$ is the p.d.f. of a logistic r.v. with location μ and scale $\lambda\sigma$, determines the i-th weight in the PIRLS iteration. Given that $g(y|\mu, \lambda\sigma)$ becomes more peaked as $\lambda \to 0$, the distribution of the weights get more skewed. This eventually leads to numerical instability, as PIRLS relies on fewer and fewer observations with relatively large weights.

Hence, it is of interest determining how fast λ can be decreased, as $n \to \infty$, without compromising stability. The following theorem is useful in this regard.

Theorem 3.2. *Indicate $g(y_i|\mu, \lambda\sigma)$ with u_i, for $i = 1, \ldots, n$, and let $\tilde{u}^n = \max_{i \in \{1,\ldots,n\}} u_i$. Also, define $q = (1 - \sqrt{1-c})/2$, where $c \in (0, 1)$. Then*

$$\mathbb{E}\left\{ \sum_{i=1}^{n} \mathbb{1}\left(\frac{u_i}{\tilde{u}^n} \geq c \right) \right\} \geq 2n\lambda\sigma f(y^*) \log\left(\frac{1-q}{q} \right),$$

for some $y^* \in (\mu + \sigma\lambda \log q/(1-q), \mu + \sigma\lambda \log(1-q)/q)$.

Proof. See Appendix B. □

Theorem 3.2 offers a lower bound on the expected number of observations whose curvature is greater, in absolute value, than a fraction c of the maximum curvature in the sample. It guarantees that, as long as λ decreases slower that n^{-1}, the expected number of observations with non-negligible relative curvature is bound to increase, thus assuring numerical stability. This rate is much faster than typical optimal rates for kernel quantile estimation. For instance, all the selection methods described by Cheng et al. (2006) lead to bandwidths that are $O(n^{-1/3})$. This assures that, at least for large n, it should be possible to select λ using statistically motivated bandwidth selection methods, with little risk of hitting the lower bound on λ imposed by computational stability considerations.

The analysis offered so far overlooked the fact that, in our context, the distribution of y depends on the covariates $\mathbf{x}$. However, if the number of unique values of $\mathbf{x}$ is kept fixed as $n \to \infty$, then our analysis clearly applies at each covariate value. Our considerations should still be applicable when the each observed vector of covariates is unique, but further work is required to clarify this. It is not clear to us whether the kernel bandwidth selection methods described by Cheng et al. (2006) could be adapted to this general setting. In fact, their methods require estimating $f(y|\mathbf{x})$ and $f'(y|\mathbf{x})$ using kernel densities, but this is not trivial to do when each y_i is associated with a unique $\mathbf{x}_i$. To limit the scope of this work, we follow a simpler approach to bandwidth selection, based on Theorem 3.1. In particular, assume that the user has chosen the maximal value, $\epsilon \in (0, 1)$, of the asymptotic bias, $|F(\mu^*|\mathbf{x}) - F(\mu_0|\mathbf{x})|$. Then, for fixed σ, it is sufficient to approximate $\sup_y f(y|\mathbf{x})$ to determine λ. For instance, if a Gaussian model $y \sim \mathrm{N}\{\alpha(\mathbf{x}), \kappa^2\}$ is used, then $\sup_y f(y|\mathbf{x}) \approx 1/\sqrt{2\pi\kappa^2}$, which leads to

$$\lambda^* = \epsilon \frac{\sqrt{2\pi\kappa^2}}{2\log(2)\sigma}. \tag{12}$$

Obviously, in an heteroscedastic setting, it might be desirable to let κ^2 and σ, and therefore λ^*, depend on $\mathbf{x}$. This can be achieved by adopting model (8) for σ, and by using a Gaussian additive model where both α and κ^2 are allowed to vary with $\mathbf{x}$. The advantage of manipulating ϵ is that is it does not depend on the scale of y, and that it is arguably more interpretable than λ. In this work we select ϵ to be as small as possible, subject to numerical stability. Given Theorem 3.2 and the results of (Cheng et al., 2006), it is clear that this approach should result in values of λ that lay slightly above the $O(n^{-1})$ computational stability boundary, but below the $O(n^{-1/3})$ area of optimal bandwidths.

4 Model fitting for fixed σ_0

Having defined a smooth generalization of the AL density, we describe an efficient framework for fitting spline-based additive quantile models. In particular, here we assume σ_0 to be fixed, and explain how to estimate the regression coefficients and the smoothing parameters.

4.1 Estimating the regression coefficients

Let $\mathbf{X}^\mu$ and $\mathbf{X}^\sigma$ be the design matrices corresponding to, respectively, $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$. Also $\mu(\mathbf{x}_i) = \mathbf{X}^\mu_{i:}\boldsymbol{\beta}^\mu$ and $\sigma(\mathbf{x}_i) = \exp(\sigma_0 + \mathbf{X}^\sigma_{i:}\boldsymbol{\beta}^\sigma)$, where $\mathbf{X}^\mu_{i:}$ is the i-th row of $\mathbf{X}^\mu$, while $\boldsymbol{\beta}^\mu$ and $\boldsymbol{\beta}^\sigma$ are vectors of regression coefficients. Here we use the prior $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbf{S}^-)$, where $\boldsymbol{\beta} = \{\boldsymbol{\beta}^\mu, \boldsymbol{\beta}^\sigma\}$ and $\mathbf{S}^-$ is an appropriate generalized matrix inverse of $\mathbf{S}^\gamma$. The latter is defined as $\mathbf{S}^\gamma = \sum_{i=1}^m \gamma_i \mathbf{S}_j$, where m indicates the total number of additive effects, $\boldsymbol{\gamma} = \{\gamma_1, \ldots, \gamma_m\}$ is a vector of positive smoothing parameters, while the $\mathbf{S}_j$s are positive semi-definite matrices, used to penalize the wiggliness of $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$. To simplify the notation, in this section we indicate $-\log \tilde{p}_F(y_i|\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i), \tau, \lambda)$ with $lo_i\{\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i)\}$, which is consistent with the fact that τ and λ are assumed to be fixed here. Depending on context, we refer to $lo_i\{\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i)\}$ as the i-th element of the ELF loss or negative log-likelihood function. Then, the negative Gibbs posterior log-density of $\boldsymbol{\beta}$ is proportional to the penalized loss

$$V(\boldsymbol{\beta}, \boldsymbol{\gamma}, \sigma_0) = \sum_{i=1}^n lo_i\{\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i)\} + \frac{1}{2}\sum_{j=1}^m \gamma_j \boldsymbol{\beta}^\mathsf{T}\mathbf{S}_j\boldsymbol{\beta}. \tag{13}$$

For fixed $\boldsymbol{\gamma}$ and σ_0, Maximum A Posteriori (MAP) estimates of the regression coefficients, $\boldsymbol{\beta}$, can be obtained by minimizing (13). Given that the objective function is smooth and convex, this can be done efficiently using Newton's algorithm. An important special case arises when $\sigma(\mathbf{x}) = \exp(\sigma_0)$, because this simpler model can be fitted using PIRLS. In particular, notice that the minimizer, $\hat{\boldsymbol{\beta}}$, of (13) corresponds to that of

$$\tilde{V}(\boldsymbol{\beta}, \boldsymbol{\gamma}, \sigma_0) = \sum_{i=1}^n Dev_i(\boldsymbol{\beta}, \sigma_0) + \sum_{j=1}^m \gamma_j \boldsymbol{\beta}^\mathsf{T}\mathbf{S}_j\boldsymbol{\beta}, \tag{14}$$

where $Dev_i(\boldsymbol{\beta}, \sigma_0) = 2[\tilde{ll}_i(\sigma_0) + lo_i\{\mu(\mathbf{x}_i), \sigma_0\}]$ and $\tilde{ll}_i(\sigma_0)$ are, respectively, the i-th component of the model deviance and of the saturated loss, $\tilde{ll}(\sigma_0)$. Now, in this simplified setting, we indicate $\boldsymbol{\beta}^\mu$ and $\mathbf{X}^\mu$ with $\boldsymbol{\beta}$ and $\mathbf{X}$, and the regression coefficients can be estimated by iteratively minimizing

$$\sum_{i=1}^n w_i(z_i - \mathbf{X}_{i:}\boldsymbol{\beta})^2 + \sum_{j=1}^m \gamma_j \boldsymbol{\beta}^\mathsf{T}\mathbf{S}_j\boldsymbol{\beta}, \tag{15}$$

where

$$z_i = \mu_i - \frac{1}{2w_i}\frac{\partial Dev_i}{\partial \mu_i}, \quad w_i = \frac{1}{2}\frac{\partial^2 Dev_i}{\partial \mu_i^2},$$

while $\mu_i = \mu(\mathbf{x}_i)$ and $Dev_i = Dev_i(\boldsymbol{\beta}, \sigma_0)$.

4.2 Selecting the smoothing parameters

A natural approach to selecting $\boldsymbol{\gamma}$ for fixed σ_0, is minimizing the marginal loss

$$G(\boldsymbol{\gamma}, \sigma_0) = -\int \exp\left[-\sum_{i=1}^n lo_i\{\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i)\}\right] p(\boldsymbol{\beta}|\boldsymbol{\gamma}) d\boldsymbol{\beta}. \tag{16}$$

which, as we noted in Section 2.2, is the normalizing constant of the Gibbs posterior. This is very important from a computational point of view, because $G(\boldsymbol{\gamma}, \sigma_0)$ can be computed and minimized using efficient methods, originally developed to handle marginal likelihoods. In particular $G(\boldsymbol{\gamma}, \sigma_0)$, which generally involves an intractable integral, can be

approximated using a Laplace approximation. This results in the Laplace Approximate Marginal Loss (LAML) criterion

$$\hat{G}(\boldsymbol{\gamma}, \sigma_0) = V(\hat{\boldsymbol{\beta}}, \boldsymbol{\gamma}, \sigma_0) + \frac{1}{2}\Big(\log|\mathbf{H}| - \log|\mathbf{S}^{\boldsymbol{\gamma}}|_+ \Big) - \frac{M_p}{2}\log(2\pi), \tag{17}$$

where $\hat{\boldsymbol{\beta}}$ is the minimizer of (13), $\mathbf{H}$ is the Hessian of (13) evaluated at $\hat{\boldsymbol{\beta}}$, M_p is the dimension of the null space of $\mathbf{S}^{\boldsymbol{\gamma}}$ and $|\mathbf{S}^{\boldsymbol{\gamma}}|_+$ is the product of its non-zero eigenvalues. If $\sigma(\mathbf{x}) = \exp(\sigma_0)$, then the LAML becomes

$$\tilde{G}(\boldsymbol{\gamma}, \sigma_0) = \frac{1}{2}\tilde{V}(\hat{\boldsymbol{\beta}}, \boldsymbol{\gamma}, \sigma_0) - \tilde{ll}(\sigma_0) + \frac{1}{2}\Big[\log|\mathbf{X}^{\mathsf{T}}\mathbf{W}\mathbf{X} + \mathbf{S}^{\boldsymbol{\gamma}}| - \log|\mathbf{S}^{\boldsymbol{\gamma}}|_+ \Big] - \frac{M_p}{2}\log(2\pi), \tag{18}$$

where $\mathbf{X} = \mathbf{X}^{\mu}$, $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}^{\mu}$, $\mathbf{W}$ is a diagonal matrix such that $\mathbf{W}_{ii} = w_i$ and $\tilde{ll}(\sigma_0)$ is the saturated loss.

For fixed σ_0, LAML can be efficiently minimized w.r.t. $\boldsymbol{\gamma}$, using an outer Newton's algorithm. Numerically stable formulas for computing LAML and its derivatives are provided by Wood et al. (2016). Importantly, the derivatives of $\hat{\boldsymbol{\beta}}$ w.r.t. $\boldsymbol{\gamma}$ are obtained by implicit differentiation which, in the general case, requires computing mixed derivatives up to fourth order of the ELF density w.r.t. μ and σ. Instead, if $\sigma(\boldsymbol{x}) = \exp(\sigma_0)$, the derivatives w.r.t. σ are not needed. Notice that the w_is in (15) and (18) can be very close to zero when fitting quantile regression models based on the ELF density, hence obtaining reliable and numerically stable estimates required modifying the PIRLS iteration and the computation of (18) and its derivatives. This more stable implementation is described in Appendix C.

Potentially, it possible to minimize LAML w.r.t. both $\boldsymbol{\gamma}$ and σ_0. While selecting σ_0 in this way is computationally efficient, the examples presented in Section 6 illustrate that this leads to highly variable fits and unreliable posterior credible intervals. Section 5 presents a calibration-based approach to the selection of σ_0 which, as the examples will show, greatly alleviates both issues.

5 Selecting σ_0 by posterior calibration

The starting point of our proposal is the work of Syring and Martin (2015), who select the learning rate, ν, so that the resulting Gibbs posterior is approximately well calibrated. In particular, consider a model parametrized by $\boldsymbol{\beta}$ and let $C_\alpha(\nu, \mathbf{y})$ be the $100\,\alpha\,\%$ credible interval for $\boldsymbol{\beta}$, at level $\alpha \in (0, 1)$. Then, Syring and Martin (2015) select ν so that

$$\mathbb{P}\{\boldsymbol{\beta}^0 \in C_\alpha(\nu, \mathbf{y})\} \approx \alpha,$$

where $\mathbb{P}$ is the objective probability measure, based on the data-generating process, and $\boldsymbol{\beta}^0$ is the true parameter.

Going back to our context, recall that we are interested in selecting σ_0, which is inversely proportional to ν, and indicate the regression coefficients with $\boldsymbol{\beta} = \{\boldsymbol{\beta}^{\mu}, \boldsymbol{\beta}^{\sigma}\}$. Our proposal aims at obtaining calibrated credible intervals for $\mu(\mathbf{x})$, rather than for $\boldsymbol{\beta}$ or $\sigma(\mathbf{x})$, which seems more practically relevant in the context of non-parametric quantile regression. Also, differently from Syring and Martin (2015), we do not calibrate $C_\alpha(\sigma_0, \mathbf{y})$ at a single level α, but consider all levels jointly. In particular, let us define $\hat{\mathbf{V}}_{\boldsymbol{\beta}} = (\hat{\boldsymbol{I}} + \mathbf{S}^{\boldsymbol{\gamma}})^{-1}$, where $\hat{\boldsymbol{I}}$ is the observed Fisher information matrix, and notice that the Gibbs posterior can be seen as a posterior based on misspecified parametric likelihood (the ELF density in our context). Given that the posterior of misspecified models is asymptotically Gaussian (Müller, 2013), it is justifiable to approximate to the posterior of $\boldsymbol{\beta}^{\mu}$ as follows

$$\boldsymbol{\mu}|\mathbf{y} \sim \mathrm{N}(\hat{\boldsymbol{\mu}}, \hat{\mathbf{V}}_{\boldsymbol{\mu}}),$$

Algorithm 1 Estimating $A(\sigma_0)$ for fixed σ_0

Assume that τ is fixed and that λ is a function of σ_0 based, for instance, on (12). Then $A(\sigma_0)$ can estimated as follows:

1: using the full design matrix, $\mathbf{X}$, and response, $\mathbf{y}$, estimate $\boldsymbol{\gamma}$ by minimizing (17) or (18). Given $\hat{\boldsymbol{\gamma}}$, estimate $\boldsymbol{\beta}$ by minimizing LAML and obtain the reference estimate $\hat{\boldsymbol{\mu}}^0 = \mathbf{X}^\mu \hat{\boldsymbol{\beta}}^\mu$.

2: For $i = 1, \dots, k$

1. Given $\hat{\boldsymbol{\gamma}}$, estimate $\boldsymbol{\beta}$ by minimizing the penalized loss (13) or (14), based on the i-th design matrices pair, $\mathbf{X}_i$, and response, $\mathbf{y}^i$. Indicate the resulting estimate with $\hat{\boldsymbol{\beta}}_i$.
2. Obtain the bootstrapped quantile prediction vector $\hat{\boldsymbol{\mu}}^i = \bar{\mathbf{X}}_i^\mu \hat{\boldsymbol{\beta}}_i^\mu$.
3. Calculate the standardized deviations of the bootstrapped fit from the full data fit

$$z_{(i-1)n+j} = (\hat{\mu}_j^0 - \hat{\mu}_j^i)\{(\hat{\mathbf{V}}_{\boldsymbol{\mu}}^i)_{jj}\}^{-\frac{1}{2}}, \quad \text{for } j = 1, \dots, n,$$

where $\hat{\mathbf{V}}_{\boldsymbol{\mu}}^i$ is the posterior covariance matrix based on the i-th bootstrapped sample. Notice that $\boldsymbol{z}$ is a vector of length nk.

3: Calculate the Anderson-Darling statistic

$$\hat{A}(\sigma_0)^2 = -nk - \sum_{l=1}^{nk} \frac{2l-1}{nk}[\log \Phi(z_l) + \log\{1 - \Phi(z_{nk+1-l})\}].$$

where $\hat{\boldsymbol{\mu}} = \mathbf{X}^\mu \hat{\boldsymbol{\beta}}^\mu$, $\hat{\mathbf{V}}_{\boldsymbol{\mu}} = \mathbf{X}^\mu \hat{\mathbf{V}}_{\boldsymbol{\beta}^\mu} (\mathbf{X}^\mu)^\mathsf{T}$ and $\hat{\mathbf{V}}_{\boldsymbol{\beta}^\mu}$ is the sub-matrix of $\hat{\mathbf{V}}_{\boldsymbol{\beta}}$ representing the posterior covariance matrix of $\boldsymbol{\beta}^\mu$. Hence, if $\boldsymbol{\mu}^0$ is the true quantile vector, the random variables $z_i = (\mu_i^0 - \hat{\mu}_i)(\hat{\mathbf{V}}_{\boldsymbol{\mu}})_{ii}^{-\frac{1}{2}}$, for $i = 1, \dots, n$, should approximately follow a standard normal distribution, under a well calibrated posterior. Ideally, we could calibrate the posterior to the data-generating process by minimizing a criterion such as

$$A(\sigma_0) = n \int \{F(z) - \Phi(z)\}^2 v(z)\, d\Phi(z),$$

w.r.t. σ_0. Here $F(z)$ and $\Phi(z)$ are, respectively, the objective c.d.f. of z and a standard normal c.d.f., with the former being implicitly dependent on σ_0, while $v(z) > 0$ is a weighting function. We choose $v(z) = [\Phi(z)\{1 - \Phi(z)\}]^{-1}$, which results in $A(\sigma_0)$ being the distance used in the Anderson-Darling test for normality (Anderson and Darling, 1954). Notice that, if calibration at a single level α is the only object of interest, this can be achieved by choosing $v(z) = \delta\{\Phi^{-1}(\alpha)\}$, where $\delta(\cdot)$ is the Dirac delta function.

Given that $F(z)$ is unknown, it must be substituted with its empirical version $F_n(z)$. Simulations from $F_n(z)$ can be obtained by bootstrapping (that is, sampling with replacement) the full dataset, fitting the model and then calculating the standardized deviations from the true quantile curve as above. Obviously $\boldsymbol{\mu}^0$ is unknown as well, but it can be substituted with $\hat{\boldsymbol{\mu}}^0$, which is the maximizer of the Gibbs posterior based on all the observed data, not on a bootstrap dataset. More precisely, let $\mathcal{X}$ and be the original $n \times d$ matrix of covariates and let $\mathbf{X} = \{\mathbf{X}^\mu, \mathbf{X}^\sigma\}$ be the corresponding pair of design matrices. Indicate the k bootstrap samples of $\mathcal{X}$ and $\mathbf{y}$ with $\mathcal{X}_1, \dots, \mathcal{X}_k$ and $\mathbf{y}^1, \dots, \mathbf{y}^k$, respectively. Also, let $\mathbf{X}_i = \{\mathbf{X}_i^\mu, \mathbf{X}_i^\sigma\}$ be the pair of design matrices derived from $\mathcal{X}_i$, where $i \in \{1, \dots, k\}$. Importantly, $\mathbf{X}_i$ differs from $\bar{\mathbf{X}}_i = \{\bar{\mathbf{X}}_i^\mu, \bar{\mathbf{X}}_i^\sigma\}$, which indicates the pair of design matrices obtained by constructing the spline bases using $\mathcal{X}_i$, and evaluating them at each row of

τ	0.01	0.05	0.5	0.95	0.99
CAL	28.7(0.7)	125.1(1.1)	670.4(3)	242.2(2.8)	77.4(3)
LAML	34(2.1)	127(1.6)	670.8(3.2)	256.6(3.5)	125.1(11)
BOOST	31.8(1.5)	126.7(1.8)	670.9(2.9)	243.2(3.4)	77.9(4)

Table 1: Mean(standard deviation) of the pinball loss for each quantile and method.

$\mathcal{X}$. Given these inputs, Algorithm 1 details the steps needed to estimate $A(\sigma_0)$ for fixed σ_0. An important feature of this procedure is that we estimate the smoothing parameters only once, using the full dataset. This is critical, because re-estimating them using each bootstrap sample would be very computationally expensive.

Given Algorithm 1, minimizing $\hat{A}(\sigma_0)$ w.r.t σ_0 is quite simple. In fact, if the bootstrap samples are simulated only once, the objective is a deterministic function of σ_0, and it can be minimized using standard root-finding algorithms, such as bisection. In our experience, the objective is generally smooth and it has a unique minimum. Indeed, decreasing σ_0 leads to more wiggly fits and hence to over-dispersion of z, relative to a standard normal. Increasing σ_0 has the opposite effect.

6 Simulated examples

Before applying the proposed quantile regression framework to load forecasting, we test it using two simulated examples. In particular, in Section 6.1 we fit an additive quantile model where $\sigma(\mathbf{x}) = \exp(\sigma_0)$ to homoscedastic data, while in Section 6.2 we consider an heteroscedastic example, where adequate interval coverage can be achieved only by letting the scale vary smoothly with $\mathbf{x}$.

6.1 An additive example

Consider the following additive model

$$y_i = x_i + x_i^2 - z_i + 2\sin(z_i) + 0.1v_i^3 + 3\cos(v_i) + e_i, \tag{19}$$

where $e_i \sim \text{Gamma}(3,1)$, $x_i \sim \text{Unif}(-4,4)$, $z_i \sim \text{Unif}(-8,8)$ and $v_i \sim \text{Unif}(-4,4)$. We aim at estimating the conditional quantile vectors corresponding to $\tau = 0.01, 0.05, 0.5, 0.95$ and 0.99. For this purpose, we fit an additive quantile regression model for each τ, using the ELF density and $\sigma(\mathbf{x}) = \exp(\sigma_0)$. We select σ_0 using either the calibration approach of Section 5 or by minimizing LAML w.r.t. both σ_0 and γ. We also consider quantile regression by gradient boosting, as implemented in the `mboost` R package (Hothorn et al., 2010).

The fitted model includes a smooth effect for each covariate, based on spline bases of rank 30. Beside selecting the rank of the bases, the boosting approach requires also selecting the degrees of freedom of each effect, which we set to 6. The number of boosting iterations was selected by minimizing the out-of-bag empirical risk, based on the pinball loss and on 100 bootstrap datasets. The boosting step size was equal to 0.1. To select σ_0 with the calibration approach we used 100 bootstrap datasets, and we minimized $\hat{A}(\sigma_0)$, for each quantile, using Brent's method (Brent, 2013). We selected λ using formula (12), with $\epsilon = 0.05$ and κ^2 set equal to the variance estimated using an initial Gaussian additive model fit. Fitting this model has a negligible impact on the computational cost, as it has to be done only once, before starting the calibration.

We firstly evaluate the accuracy of estimated quantile vectors using the pinball loss. To do this we simulated $n = 10^3$ data points from (19), and we fitted an additive model for each τ using each approach. We repeated the process 20 times, and Table 1 reports the

average pinball loss and its standard deviation. Quantile regression using the ELF density and calibration achieves the lowest loss for all quantiles. Also, the variabily of the loss is lower than with the remaining methods, suggesting that this method leads to smoother quantile estimates. LAML estimation of σ_0 performs very badly for the most extreme quantiles. We comment on this issue later. Gradient boosting does worse than calibrated quantile regression, but at least its relative performance seems to fairly constant with τ. On an Intel 2.50GHz CPU, fitting an additive quantile regression model with the ELF density takes around 0.33s for $\tau = 0.5$ and 0.35s for $\tau = 0.01$, if σ_0 is held fixed. Having fixed the number of steps, gradient boosting takes around 1.2s and 17s to estimate the same quantiles. Gradient boosting is particularly slow for $\tau = 0.01$ because the empirical risk criterion used by `mboost` is minimized at a large number of steps ($\approx 3 \times 10^4$). The time spent calibrating σ_0 or selecting the number of boosting steps is roughly proportional to that needed to fit a single model.

τ	0.01	0.05	0.5	0.95	0.99
CAL95	0.950	0.947	0.946	0.927	0.774
LAML95	0.440	0.752	0.903	0.330	0.069
CAL75	0.754	0.748	0.746	0.708	0.525
LAML75	0.270	0.505	0.673	0.197	0.040
CAL50	0.504	0.501	0.497	0.468	0.326
LAML50	0.160	0.312	0.435	0.116	0.024

Table 2: Empirical coverage achieved by selecting σ_0 using calibration or LAML, for each τ and confidence level.

Table 2 reports the empirical coverage, at 95%, 75% and 50% level, achieved by the credible intervals for $\boldsymbol{\mu}$, using calibration or LAML to select σ_0. The coverage was calculated using 100 simulations from model (19). We do not check the coverage achieved by gradient boosting, because analytic formulas are, to our best knowledge, unavailable and confidence intervals must be obtained by bootstrapping. Notice that the coverage achieved using LAML for selecting σ_0 is well below nominal levels for all quantiles. In particular, for $\tau = 0.99$, the credible intervals are so narrow that the coverage is negligible. Using calibration leads to empirical coverages matching nominal levels almost exactly, for $\tau = 0.01$, 0.05 and 0.5. However, some under-coverage might be occurring for $\tau = 0.95$, and it certainly is for $\tau = 0.99$. Arguably, here we are dealing with a worst-case scenario for quantile regression. Indeed, the observation density is highly skewed to the right but, when $\tau \approx 1$, the pinball loss is very steep on that side. The same holds true under the ELF-based loss, hence few observation located above the quantile have a strong influence on the fit. When σ_0 is selected by LAML, these points are almost interpolated, which results in severe overfitting. The calibration procedure selects much higher values of σ_0, which in turn lead to higher estimates of $\boldsymbol{\gamma}$ in the nested iteration minimizing (18). Indeed, when calibration is used, the average total number of effective degrees of freedom is between 30 and 40 for $\tau = 0.01, 0.05$ and 0.5, but only around 13 for $\tau = 0.99$.

6.2 An heteroscedastic example

Here we consider the following heteroscedastic data generating process

$$y_i = x_i + x_i^2 + e_i, \quad e_i \sim \mathrm{N}\{0, \kappa(x_i)^2\}, \quad \kappa(x_i) = 1.2 + \sin(2x_i), \tag{20}$$

where $x_i \sim \mathrm{Unif}(-4, 4)$. We simulate $n = 10^3$ data points from (20) and we fit quantile models for the median and the 95th percentile. In particular, we consider a simplified model where $\sigma(x) = \exp(\sigma_0)$ and a full model where the scale is allowed to vary with

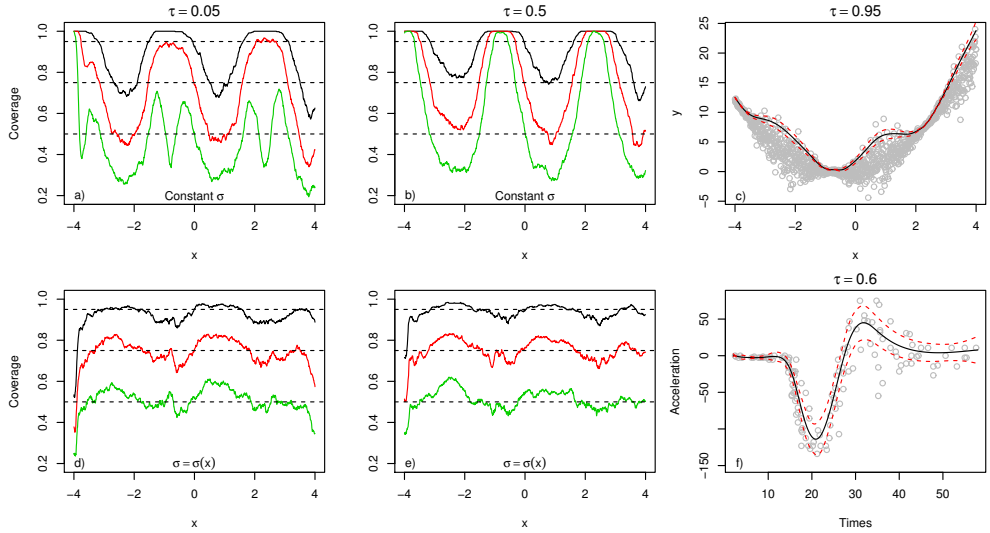

Figure 3: *a*, *b*, *d* and *e*: nominal (dashed) vs empirical (solid) coverage at 50, 75 and 95% level, using a simplified (*a* and *b*) or full quantile model (*d* and *e*). *c*: full fit using $\tau = 0.95$ with data from model (20). *d*: full fit for quantile $\tau = 0.6$ using the motorcycle dataset.

x. We use cubic regression spline bases of rank 30 for both location and scale. We set $\epsilon = 0.05$ and determine λ using (12). Notice that, when the full model is used, we use a preliminary Gaussian model where both mean and variance depend on x, hence $\lambda = \lambda(x)$. We fit this Gaussian model using the methods described by Wood et al. (2016).

The first two columns in Figure 3 compare nominal and empirical coverages of credible intervals for $\mu(x)$, obtained by fitting 500 dataset simulated from (20). It is evident that the simplified model provides unreliable intervals even at the median, and at any nominal level. The intervals provided by the full model are much closer to nominal levels, even though they seems to be slightly too conservative where $\kappa^2(x)$ is high and slightly too narrow where $\kappa^2(x)$ in low. Figure 3 also shows a model fit to the motorcycle dataset, obtained using $\tau = 0.6$, an adaptive P-spline basis of rank 30 for $\mu(x)$ and a thin-plate spline basis of rank 5 for $\sigma(x)$.

7 Application to probabilistic load forecasting

We verify the performance of the proposed methodology in the context of probabilistic electricity load forecasting. We firstly consider the dataset used in the load forecasting track of the Global Energy Competition 2014 (GEFCom2014). This covers the period between January 2005 and December 2011, and it includes hourly load consumption and temperatures. The latter were measured at 25 weather stations, but here we average the temperature records of only four stations to obtain a single variable. See Gaillard et al. (2016) for details on how this subset of stations was selected. We also consider a dataset containing hourly electricity demand from the UK grid, covering the period between January 2011 and June 2016, which we obtained from www.nationalgrid.com. We integrate it with hourly temperature data covering the same period, and measured at ten major cities in the UK, from the National Centers for Environmental Information (NCEI). To obtain a single temperature index, we calculate the weighted average of these records, using weights proportional to the population of each city. We aim at predicting 20 conditional quantiles, equally spaced between $\tau = 0.05$ and $\tau = 0.95$. Given that

load consumption is strongly dependent on the time of the day, it is common practice (e.g. Gaillard et al. (2016)) to fit a different model for each half-hour. To limit the computational burden, here we consider only the period between between 11:30 and 12am. We use the period 2005-09 of the GEFCom2014 dataset for training, leaving the last two years for testing. Similarly, we use the last 12 months of the UK dataset for testing.

Gaillard et al. (2016) proposed a quantile regression method which ranked 1st on both the load and the price forecasting track of GEFCom2014. This is a two-step procedure, called quantGAM, which was partially motivated by the lack of reliable software for fitting additive quantile models. Very briefly, their method firstly fits a Gaussian additive model to model mean load and, optionally, a second one to model the variance of the residuals from the first fit. Then, for each quantile, they fit a linear quantile regression to model the load, using the effects estimated by the Gaussian fits as covariates. We compare their method to our proposal and to gradient boosting, using the set of covariates proposed by Gaillard et al. (2016): hourly temperatures (T_t); smoothed temperature (T_t^s), obtained by exponentially smoothing T_t; a cyclic variable indicating the position within the year (S_t); a factor variable indicating the day of the week (D_t); a sequential index representing time (t); the observed load at the same time of previous day (L_{t-1}). The effects of T_t, T_t^s, S_t and t are modelled using splines, while D_t and L_{t-1} are modelled, respectively, using dummy variables and a linear effect. For the GEFCom2014 dataset we use bases of rank 30 for T_t, T_t^s and S_t , while we limit the rank to 4 for the effect of t, which captures the long term trend. A periodic smooth is used for S_t. For gradient boosting we use 6 degrees of freedom for T_t and T_t^s, 15 for S_t and 4 for t. Notice that these parameters need to be tuned manually, which can be time consuming. For the UK dataset we use the similar set-up, but we use bases of rank 20 for T_t and T_t^s. We also add a binary variable, H_t, indicating bank holidays and, given that UK electricity consumption drops sharply during the Christmas period, we use an adaptive periodic basis for S_t.

We use $\sigma(\mathbf{x}) = \exp(\sigma_0)$ and we calibrate σ_0 using 100 bootstrapped datasets. We use 100 datasets also to select the number of boosting steps, while the step-size used for gradient boosting is equal to 0.02 for GEFCom2014 and to 0.1 for the UK dataset. Under GEFCom2014 we use $\epsilon = 0.05$ in (12) to determine λ, with κ^2 being estimated using a Gaussian additive model. For the UK dataset we use $\epsilon = 0.1$. Decreasing ϵ further leads to numerical instabilities for the most extreme quantiles. These problems are easily detected in the course of the calibration, because they result in discontinuities in $\hat{A}(\sigma_0)$. Having tuned σ_0 and the number of boosting steps on the training sets, we forecast electricity load one week ahead on the test sets. To forecast load one week ahead, we use the observed temperature over that week. Obviously future temperatures would not be available in an operational setting, and a forecast would be used instead. But using a forecast would add further uncertainty to the results of the comparison performed here, hence we prefer using observed temperatures. Week by week we predict the load for the next seven days, and then we re-fit all models using the newly observed values of load and temperatures.

Figure 4 shows, for each τ and both datasets, the pinball loss incurred by each method on the testing set, divided by the pinball loss of a Gaussian additive fit. It is satisfactory to notice that the calibrated quantile regression approach does better than a Gaussian fit for almost all quantiles. On the GEFCom2014 dataset, the proposed approach does better than quantGAM for $\tau < 0.4$, but it incurs slightly larger losses for higher quantiles. Gradient boosting is better than a Gaussian model, but it does worse than quantGAM or our approach above the median. However, on UK grid data boosting achieves the lowest loss on most quantiles. On this dataset, our approach does generally better than quantGAM for $\tau \leq 0.6$ and slightly worse for higher quantiles. Notice that the scales on the two plots containing the relative losses are different: departures from normality are much stronger in the UK dataset.

With our method, fitting an additive quantile model on GEFCom2014 data takes around 1s for the median and 2s for $\tau = 0.95$, if σ_0 is held fixed to the value selected

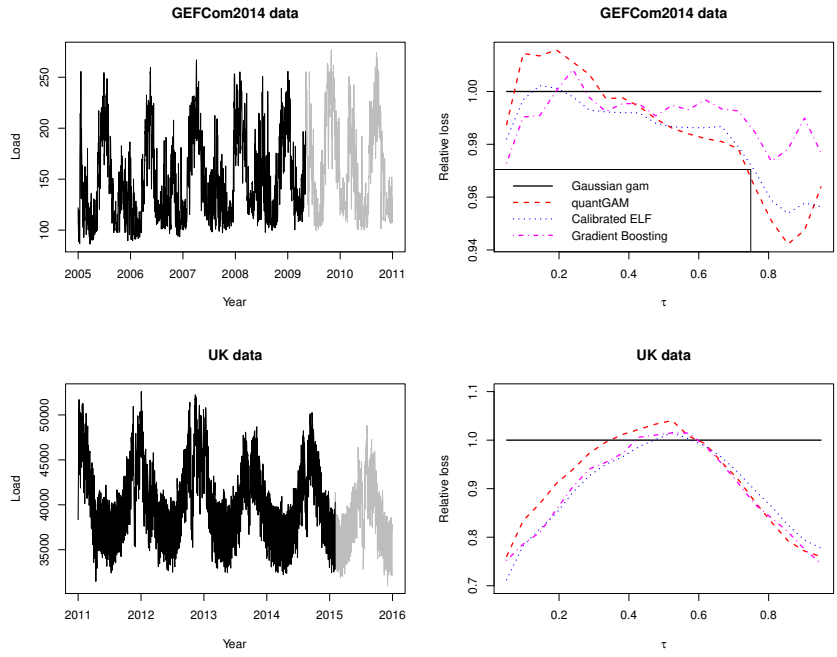

Figure 4: Left: observed loads divided between training (black) and testing (grey) sets. Right: relative pinball losses for each τ and method.

by the calibration. The time taken by gradient boosting depends on the selected number of boosting steps, and it varied between around 4s ($\approx$ 3000 steps) for $\tau = 0.5$ and 24s ($\approx 2 \times 10^4$ steps) for $\tau = 0.95$.

8 Conclusion

The main contribution of this work has been to provide a stable and computationally efficient framework for fitting semi-parametric additive quantile regression models. We have addressed the lack of curvature of the pinball loss by embedding the corresponding AL density into an extended log-F distribution. While this requires choosing a tuning parameter, λ, we have showed how this can be expressed in terms of a more interpretable, and scale invariant, approximate bound on the asymptotic bias, ϵ. Using the new loss, the regression coefficients can be estimated using fast MAP methods, while the smoothing parameters can be selected automatically by LAML minimization. Secondly, we have described a calibration approach for selecting the learning rate, $\nu = 1/\sigma_0$, which leads to smooth quantile fits and reliable credible intervals. In the electricity load forecasting examples we have demonstrated the practical utility of the proposed approach, whose performance is competitive with that of gradient boosting and of the quantile regression method of Gaillard et al. (2016).

In this work we have selected λ or, equivalently, ϵ to be as small as possible, subject to numerical stability. Oh et al. (2012) and Yue and Rue (2011) have followed a similar approach, although using different approximations to the pinball loss or AL density. Further work is needed to clarify whether smoothing the AL density can provide statistical, in addition to computational, advantages in the context of non-parametric quantile regression. If this turns out to be the case, it would be interesting developing more statistically motivated selection methods for λ, possibly based on the kernel interpretation of the ELF

density.

References

Anderson, T. W. and D. A. Darling (1954). A test of goodness of fit. *Journal of the American statistical association 49*(268), 765–769.

Bissiri, P., C. Holmes, and S. Walker (2013). A general framework for updating belief distributions. *arXiv preprint arXiv:1306.6430*.

Brent, R. P. (2013). *Algorithms for minimization without derivatives*. Courier Corporation.

Cheng, M.-Y., S. Sun, et al. (2006). Bandwidth selection for kernel quantile estimation. *Journal of the Chinese Statistical Association 44*(3), 271–295.

Falk, M. (1984). Relative deficiency of kernel type estimators of quantiles. *The Annals of Statistics*, 261–268.

Gaillard, P., Y. Goude, and R. Nedellec (2016). Additive models and robust aggregation for gefcom2014 probabilistic electric load and electricity price forecasting. *International Journal of Forecasting*.

Hothorn, T., P. Bühlmann, T. Kneib, M. Schmid, and B. Hofner (2010). Model-based boosting 2.0. *The Journal of Machine Learning Research 11*, 2109–2113.

Jones, M. (2008). On a class of distributions with simple exponential tails. *Statistica Sinica 18*(3), 1101–1110.

Jones, M. and K. Yu (2007). Improved double kernel local linear quantile regression. *Statistical Modelling 7*(4), 377–389.

Koenker, R. (2005). *Quantile regression*. Number 38. Cambridge university press.

Koenker, R. (2013). Quantreg: quantile regression. *R package version 5*.

Lin, C.-Y., H. Bondell, H. H. Zhang, and H. Zou (2013). Variable selection for non-parametric quantile regression via smoothing spline analysis of variance. *Stat 2*(1), 255–268.

Lindgren, F. and H. Rue (2015). Bayesian spatial modelling with r-inla. *Journal of Statistical Software 63*(19).

Mächler, M. (2012). Accurately computing $\log(1-\exp(-|a|))$. *URL http://cran. r-project. org/web/packages/Rmpfr/vignettes/log1mexp-note. pdf*.

Martins, T. G., D. Simpson, F. Lindgren, and H. Rue (2013). Bayesian computing with inla: new features. *Computational Statistics & Data Analysis 67*, 68–83.

Menne, M. J., I. Durre, R. S. Vose, B. E. Gleason, and T. G. Houston (2012). An overview of the global historical climatology network-daily database. *Journal of Atmospheric and Oceanic Technology 29*(7), 897–910.

Müller, U. K. (2013). Risk of bayesian inference in misspecified models, and the sandwich covariance matrix. *Econometrica 81*(5), 1805–1849.

Oh, H.-S., T. C. Lee, and D. W. Nychka (2012). Fast nonparametric quantile regression with arbitrary smoothing methods. *Journal of Computational and Graphical Statistics*.

Read, R. (1972). The asymptotic inadmissibility of the sample distribution function. *The Annals of Mathematical Statistics*, 89–95.

Syring, N. and R. Martin (2015). Scaling the gibbs posterior credible regions. *arXiv preprint arXiv:1509.00922*.

Waldmann, E., T. Kneib, Y. R. Yue, S. Lang, and C. Flexeder (2013). Bayesian semiparametric additive quantile regression. *Statistical Modelling 13*(3), 223–252.

Wood, S. (2006). *Generalized additive models: an introduction with R*. CRC press.

Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73*(1), 3–36.

Wood, S. N., N. Pya, and B. Safken (2016). On smooth modelling with regular likelihoods. *In press*.

Yee, T. W. (2008). The vgam package. *R News 8*(2), 28–39.

Yu, K. and R. A. Moyeed (2001). Bayesian quantile regression. *Statistics & Probability Letters 54*(4), 437–447.

Yue, Y. R. and H. Rue (2011). Bayesian inference for additive mixed quantile regression models. *Computational Statistics & Data Analysis 55*(1), 84–96.

Appendices

A Proof of Theorem 3.1

To simplify the notation, indicate $\tilde{p}_F(y|\mu,\sigma,\tau,\lambda)$ with $\tilde{p}_F(y)$. We start from

$$F(\mu^*) - F(\mu_0) = \int \Big\{ -\frac{\partial \log \tilde{p}_F(y)}{\partial \mu} - \frac{\partial \rho_\tau(y)}{\partial \mu} \Big\} f(y)\, dy.$$

which is implied by the proof of Proposition 1 in Oh et al. (2012). We proceed to bound the r.h.s. from above. Simple manipulations lead to

$$\int \Big\{ -\frac{\partial \log \tilde{p}_F(y)}{\partial \mu} - \frac{\partial \rho_\tau(y)}{\partial \mu} \Big\} f(y)\, dy = \int \Big\{ \mathbb{1}(y > \mu) - \Phi(y|\mu, \lambda\sigma) \Big\} f(y)\, dy, \qquad (21)$$

where $\mathbb{1}(\cdot)$ is an indicator function and $\Phi(y|\mu,\lambda\sigma)$ is the c.d.f. of a logistic random variable, with location μ and scale $\lambda\sigma$. Then we have

$$\begin{aligned} |F(\mu^*) - F(\mu_0)| &\leq \int \Big| \mathbb{1}(y > \mu) - \Phi(y|\mu,\lambda\sigma) \Big| \sup_y f(y)\, dy \\ &= 2 \sup_y f(y) \int_{-\infty}^{\mu} \Phi(y|\mu,\lambda\sigma)\, dy, \end{aligned}$$

where the second equality holds due to the symmetry of the integrand around μ. Using the substitution $z = (y - \mu)/\lambda\sigma$ leads to

$$\begin{aligned} |F(\mu^*) - F(\mu_0)| &\leq 2\lambda\sigma \sup_y f(y) \int_{-\infty}^{0} \frac{1}{1+e^{-z}}\, dz \\ &= 2\log(2)\lambda\sigma \sup_y f(y). \end{aligned}$$

Finally, notice that the r.h.s. of (21) makes it clear that, if $f(y)$ is symmetric around μ, then $|F(\mu^*) - F(\mu_0)| = 0$.

B Proof of Theorem 3.2

Define $\bar{u} = \max_y g(y|\mu, \lambda\sigma) = (4\lambda\sigma)^{-1}$ and notice that

$$E\Big\{\sum_{i=1}^{n} \mathbb{1}\Big(\frac{u_i}{\tilde{u}^n} \geq c\Big)\Big\} \geq E\Big\{\sum_{i=1}^{n} \mathbb{1}\Big(\frac{u_i}{\bar{u}} \geq c\Big)\Big\} = n\,\mathrm{Prob}\Big\{\frac{u}{\bar{u}} \geq c\Big\}.$$

Now, the symmetry of g around μ leads to

$$\mathrm{Prob}\Big\{\frac{u}{\bar{u}} \geq c\Big\} = \mathrm{Prob}\Big\{Q(q) \leq y_i \leq Q(1-q)\Big\},$$

where $Q(q) = \mu + \lambda\sigma\log\{q/(1-q)\}$ is the quantile function of a logistic distribution and $q = (1 - \sqrt{1-c})/2$, which is obtained by solving

$$\frac{u}{\bar{u}} = \frac{g\{Q(q)|\mu, \lambda\sigma\}}{(4\lambda\sigma)^{-1}} = c,$$

for q. Finally, we have

$$n\,\mathrm{Prob}\Big\{\frac{u}{\bar{u}} \geq c\Big\} = \int_{Q(q)}^{Q(1-q)} f(y)\, dy = nf(y^*)\{Q(1-q) - Q(q)\},$$

for some $Q(q) < y^* < Q(1-q)$, due to the Mean Value Theorem. Elementary manipulations lead to the final result.

C Stabilizing computation the ELF density

C.1 Dealing with zero weights in PIRLS

Quantile regression with the ELF density requires that we work with many weights that can be very close to zero, while the corresponding log-likelihood or deviance derivative is far from zero. This can lead to a situation in which the vector containing $w_i z_i$ is well scaled, while the vector containing $\sqrt{|w_i|}z_i$ is very poorly scaled. This scaling problem can reverse the usual stability improvement of QR-based least squares estimation over direct normal equation solution.

We adopt the notation of Wood (2011). Let $\bar{\mathbf{W}}$ be a diagonal matrix with $\bar{W}_{ii} = |w_i|$ and let $\mathbf{E}$ be a matrix such that $\mathbf{S}^\gamma = \mathbf{E}^\mathsf{T}\mathbf{E}$. Then let $\boldsymbol{\mathcal{Q}\mathcal{R}}$ be the QR decomposition of $\sqrt{\bar{\mathbf{W}}}\mathbf{X}$ and define the further QR decomposition

$$\begin{pmatrix} \boldsymbol{\mathcal{R}} \\ \mathbf{E} \end{pmatrix} = \mathbf{QR}.$$

Define the matrix $\mathbf{Q}_1 = \boldsymbol{\mathcal{Q}}\mathbf{Q}[1{:}d, :]$, where d is the number of columns of $\mathbf{X}$ and $\mathbf{Q}[1{:}d, :]$ indicates the first d rows of $\mathbf{Q}$. We also need to define the diagonal matrix $\mathbf{I}^-$, such that I^-_{ii} is equal to 0 if $w_i > 0$ and 1 otherwise, and the singular value decomposition $\mathbf{I}^-\mathbf{Q}_1 = \mathbf{U}\mathbf{D}\mathbf{V}^\mathsf{T}$. See Wood (2011) for details on how to deal with non-identifiable parameters.

Using this notation, Wood (2011) shows that

$$\hat{\boldsymbol{\beta}} = \mathbf{R}^{-1}\mathbf{V}(\mathbf{I} - 2\mathbf{D}^2)^{-1}\mathbf{V}^\mathsf{T}\mathbf{Q}_1^\mathsf{T}\sqrt{\bar{\mathbf{W}}}\bar{\mathbf{z}} = \mathbf{R}^{-1}\mathbf{f},$$

where $\bar{\mathbf{z}}$ is a vector such that $\bar{z}_i = z_i$ if $w_i \geq 0$ and $\bar{z}_i = -z_i$ otherwise, while the definition of $\mathbf{f}$ should be obvious. Now we can test for stability of the computation to the scaling of $\sqrt{\bar{\mathbf{W}}}\bar{\mathbf{z}}$ by testing whether

$$\mathbf{R}\mathbf{Q}_1^\mathsf{T}\sqrt{\bar{\mathbf{W}}}\bar{\mathbf{z}} = \mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{z},$$

to sufficient accuracy. If it does not, then we recompute $\mathbf{f}$ using

$$\mathbf{f} = \mathbf{V}(\mathbf{I} - 2\mathbf{D}^2)\mathbf{V}^\mathsf{T}\mathbf{R}^{-1}\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{z}.$$

If we define the matrices

$$\mathbf{P} = \mathbf{R}^{-1}\mathbf{V}(\mathbf{I} - 2\mathbf{D}^2)^{-\frac{1}{2}}, \qquad \mathbf{K} = \mathbf{Q}_1\mathbf{V}(\mathbf{I} - 2\mathbf{D}^2)^{-\frac{1}{2}},$$

then another possibility, that may be more convenient when using $\hat{\boldsymbol{\beta}} = \mathbf{P}\mathbf{K}^\mathsf{T}\sqrt{\bar{\mathbf{W}}}\bar{\mathbf{z}}$, is to test whether $\mathbf{K}^\mathsf{T}\sqrt{\bar{\mathbf{W}}}\bar{\mathbf{z}} = \mathbf{P}^\mathsf{T}\mathbf{W}\mathbf{z}$ to sufficient accuracy, and to use $\hat{\boldsymbol{\beta}} = \mathbf{P}\mathbf{P}^\mathsf{T}\mathbf{W}\mathbf{z}$ if not.

C.2 Dealing with zero weights in LAML

Here we show how the gradient and Hessian of $\log|\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}}|$, which are needed to maximize the LAML using Newton's algorithm, can be computed in a stable manner. In order to be consistent with the notation of Wood (2011), in this section we indicate the smoothing parameter vector with $\boldsymbol{\lambda}$, rather than with $\boldsymbol{\gamma}$, and the penalty matrix with $\mathbf{S}_{\boldsymbol{\lambda}}$, rather than $\mathbf{S}^{\boldsymbol{\gamma}}$. Notice that $(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1} = \mathbf{P}\mathbf{P}^\mathsf{T}$, hence

$$\begin{aligned}
\frac{\partial \log|\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}}|}{\partial \rho_k} &= \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X}\right\} + \lambda_k \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{S}_k\right\} \\
&= \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X}\mathbf{P}\right\} + \lambda_k \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{S}_k\mathbf{P}\right\}.
\end{aligned}$$

Then the Hessian is

$$\begin{aligned}
\frac{\partial^2 \log|\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}}|}{\partial \rho_k \partial \rho_j} &= \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{X}^\mathsf{T}\frac{\partial^2 \mathbf{W}}{\partial \rho_k \partial \rho_k}\mathbf{X}\right\} + \delta_k^j \lambda_j \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{S}_j\right\} \\
&- \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\left(\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X} + \lambda_j\mathbf{S}_j\right)(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_j}\mathbf{X}\right\} \\
&- \lambda_k \mathrm{tr}\left\{(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\left(\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X} + \lambda_j\mathbf{S}_j\right)(\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}})^{-1}\mathbf{S}_k\right\},
\end{aligned}$$

so that

$$\begin{aligned}
\frac{\partial^2 \log|\mathbf{X}^\mathsf{T}\mathbf{W}\mathbf{X} + \mathbf{S}_{\boldsymbol{\lambda}}|}{\partial \rho_k \partial \rho_j} &= \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial^2 \mathbf{W}}{\partial \rho_k \partial \rho_j}\mathbf{X}\mathbf{P}\right\} + \lambda_k \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{S}_k\mathbf{P}\right\} \\
&- \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_j}\mathbf{X}\mathbf{P}\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X}\mathbf{P}\right\} \\
&- \lambda_j \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{S}_j\mathbf{P}\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_k}\mathbf{X}\mathbf{P}\right\} - \lambda_k \mathrm{tr}\left\{\mathbf{P}^\mathsf{T}\mathbf{X}^\mathsf{T}\frac{\partial \mathbf{W}}{\partial \rho_j}\mathbf{X}\mathbf{P}\mathbf{P}^\mathsf{T}\mathbf{S}_k\mathbf{P}\right\} \\
&- \lambda_j\lambda_k \mathrm{tr}(\mathbf{P}^\mathsf{T}\mathbf{S}_j\mathbf{P}\mathbf{P}^\mathsf{T}\mathbf{S}_k\mathbf{P}).
\end{aligned}$$

If we define the diagonal matrices $\mathbf{T}_j = \text{diag}(\partial w_i/\partial\rho_j)$ and $\mathbf{T}_{jk} = \text{diag}(\partial^2 w_i/\partial\rho_j\partial\rho_k)$, then this last expression corresponds to the equivalent formula in Wood (2011) and can be computed in the same way. The point of all this is that, if we followed the original formulation of Wood (2011), we would be dividing by the (almost zero) weights in the definition of $\mathbf{T}_j$ and $\mathbf{T}_{jk}$. This is avoided here.

D Details regarding the new density

D.1 Derivatives of the log-likelihood

The logarithm of the proposed density (10) is

$$ll(y) = \log\tilde{p}_F(y|\mu,\sigma,\tau,\lambda) = (1-\tau)\frac{y-\mu}{\sigma} - \lambda\log\left[1+e^{\frac{y-\mu}{\lambda\sigma}}\right] - \log\left\{\lambda\sigma\text{Beta}\big[\lambda(1-\tau),\lambda\tau\big]\right\},$$

When evaluating this numerically, it is important to approximate $\log(1+e^z)$ with $z+e^{-z}$ when $z = (y-\mu)/\lambda\sigma > 18$, as suggested by Mächler (2012). The gradient is

$$\frac{\partial ll(y)}{\partial\mu} = \frac{1}{\sigma}\left[\Phi(y|\mu,\lambda\sigma) - 1 + \tau\right], \quad \frac{\partial ll(y)}{\partial\sigma} = \frac{y-\mu}{\sigma^2}\left[\Phi(y|\mu,\lambda\sigma) - 1 + \tau\right] - \frac{1}{\sigma},$$

where $\Phi(y|\mu,\lambda\sigma)$ is the c.d.f. of a logistic density with location μ and scale $\lambda\sigma$.

The Hessian is

$$\frac{\partial^2 ll(y)}{\partial\mu^2} = -\frac{1}{\sigma}\phi(y|\mu,\lambda\sigma),$$

$$\frac{\partial^2 ll(y)}{\partial\sigma^2} = 2\frac{y-\mu}{\sigma^3}\left[1-\tau-\Phi(y|\mu,\lambda\sigma) - \frac{1}{2}(y-\mu)\phi(y|\mu,\lambda\sigma)\right] + \frac{1}{\sigma^2},$$

$$\frac{\partial^2 ll(y)}{\partial\mu\partial\sigma} = -\frac{1}{\sigma^2}\left[(y-\mu)\phi(y|\mu,\lambda\sigma) + \Phi(y|\mu,\lambda\sigma) - 1 + \tau\right],$$

where $\phi(y|\mu,\lambda\sigma)$ is the p.d.f of a logistic density with location μ and scale $\lambda\sigma$.

Now, define $z = (y-\mu)/(\lambda\sigma)$ so that

$$\Phi(y|\mu,\lambda\sigma) = \Phi(z|0,1) = \Phi(z) = \frac{1}{1+e^{-z}},$$

is the sigmoid function. Also, define $\Phi^{(k)}(y) = \partial\Phi^{(k)}(y)/\partial y^k$. Then, derivatives of higher order are

$$\frac{\partial^3 ll(y)}{\partial\mu^3} = \frac{\Phi^{(2)}(z)}{\lambda^2\sigma^3}, \qquad \frac{\partial^4 ll(y)}{\partial\mu^4} = -\frac{\Phi^{(3)}(z)}{\lambda^3\sigma^4},$$

$$\frac{\partial^3 ll(y)}{\partial\sigma^3} = -\frac{3}{\sigma}\frac{\partial^2 ll(y)}{\partial\sigma^2} + \frac{\lambda z^2}{\sigma^3}\left[3\Phi^{(1)}(z) + z\Phi^{(2)}(z) + \frac{1}{\lambda z^2}\right],$$

$$\frac{\partial^4 ll(y)}{\partial\sigma^4} = -\frac{4}{\sigma}\left[2\frac{\partial^3 ll(y)}{\partial\sigma^3} + \frac{3}{\sigma}\frac{\partial^2 ll(y)}{\partial\sigma^2}\right] - \frac{\lambda z^3}{\sigma^4}\left[4\Phi^{(2)}(z) + z\Phi^{(3)}(z) - \frac{2}{\lambda z^3}\right],$$

$$\frac{\partial^3 ll(y)}{\partial\mu^2\partial\sigma} = \frac{1}{\lambda\sigma^3}\big[z\Phi^{(2)}(z) + 2\Phi^{(1)}(z)\big],$$

$$\frac{\partial^4 ll(y)}{\partial\mu^3\partial\sigma} = -\frac{1}{\lambda^2\sigma^4}\big[z\Phi^{(3)}(z) + 3\Phi^{(2)}(z)\big],$$

$$\frac{\partial^3 ll(y)}{\partial\mu\partial\sigma^2} = \frac{1}{\sigma^3}\left\{2\big[\Phi(z) - 1 + \tau\big] + 4z\Phi^{(1)}(z) + z^2\Phi^{(2)}(z)\right\},$$

$$\frac{\partial^4 ll(y)}{\partial\mu\partial\sigma^3} = -\frac{3}{\sigma}\frac{\partial^3 ll(y)}{\partial\mu\partial\sigma^2} - \frac{z}{\sigma^4}\big[6\Phi^{(1)}(z) + 6z\Phi^{(2)}(z) + z^2\Phi^{(3)}(z)\big],$$

$$\frac{\partial^4 ll(y)}{\partial\mu^2\partial\sigma^2} = -\frac{1}{\lambda\sigma^4}\bigg\{z^2\Phi^{(3)}(z) + 6z\Phi^{(2)}(z) + 6\Phi^{(1)}(z)\bigg\},$$

where

$$\Phi^{(1)}(z) = \frac{\partial\Phi(z)}{\partial z} = \Phi(z)\big[1-\Phi(z)\big],$$
$$\Phi^{(2)}(z) = \Phi^{(1)}(z) - 2\Phi^{(1)}(z)\Phi(z),$$
$$\Phi^{(3)}(z) = \Phi^{(2)}(z) - 2\Phi^{(2)}(z)\Phi(z) - 2\Phi^{(1)}(z)^2.$$

D.2 Saturated log-likelihood, deviance and expected Fisher information

To find the saturated log-likelihood ll_s we need to maximize $\tilde{p}_F(y|\mu,\sigma,\tau,\lambda)$ w.r.t. μ. Hence we need to impose

$$\frac{\partial ll(y)}{\partial\mu} = \frac{1}{\sigma}\bigg[\frac{1}{1+e^{-\frac{y-\mu}{\lambda\sigma}}} - 1 + \tau\bigg] = 0,$$

which leads to

$$\hat{\mu} = \lambda\sigma\log\Big(\frac{\tau}{1-\tau}\Big) + y.$$

Hence, the saturated log-likelihood is

$$ll_s(y) = (1-\tau)\lambda\log\big(1-\tau\big) + \lambda\tau\log(\tau) - \log\bigg\{\lambda\sigma\mathrm{Beta}\big[\lambda(1-\tau),\lambda\tau\big]\bigg\},$$

and has derivatives

$$\frac{\partial ll_s(y)}{\partial\sigma} = -\frac{1}{\sigma}, \quad \frac{\partial^2 ll_s(y)}{\partial\sigma^2} = \frac{1}{\sigma^2}.$$

This implies that the deviance is

$$dev(y) = 2\big[ll_s(y) - ll(y)\big] = 2\bigg\{(1-\tau)\lambda\log\big(1-\tau\big) + \lambda\tau\log(\tau) - (1-\tau)\frac{y-\mu}{\sigma} + \lambda\log\bigg[1+e^{\frac{y-\mu}{\lambda\sigma}}\bigg]\bigg\}.$$

The entry of the expected Fisher information matrix corresponding to μ is

$$E\bigg[\frac{\partial^2 ll(y)}{\partial\mu^2}\bigg] = -\frac{1}{\lambda\sigma^2}\int_{-\infty}^{\infty}\frac{e^{-\frac{y-\mu}{\lambda\sigma}}}{\big(1+e^{-\frac{y-\mu}{\lambda\sigma}}\big)^2}\frac{e^{(1-\tau)\frac{x-\mu}{\sigma}}(1+e^{\frac{x-\mu}{\lambda\sigma}})^{-\lambda}}{\lambda\sigma\mathrm{Beta}\big[\lambda(1-\tau),\lambda\tau\big]}dy,$$

and some algebra leads to

$$E\bigg[\frac{\partial^2 ll(y)}{\partial\mu^2}\bigg] = -\frac{1}{\sigma^2}\frac{\tau(1-\tau)}{\lambda+1}.$$

D.3 Random variables generation and moments

Notice that, if a r.v. y has density $\tilde{p}_F(y|\mu,\sigma,\tau,\lambda)$, then

$$z \sim B\big[\lambda(1-\tau),\lambda\tau\big],$$

where $B(\alpha, \beta)$ is a Beta distribution and $z = \Phi\{(y-\mu)/\sigma\lambda\}$. Hence, to simulate a random variables y from $\tilde{p}_F(y|\mu, \sigma, \tau, \lambda)$ we could use

$$z \sim B\big[\lambda(1-\tau), \lambda\tau\big], \qquad y = \sigma\lambda\Phi^{-1}(z) + \mu = \sigma\lambda \log\left(\frac{z}{1-z}\right) + \mu.$$

Unfortunately, `rbeta` in R produces many 1s when the shape parameters are fairly small, leading to the overflow of $z/(1-z)$. A better scheme is

$$u \sim \Gamma\{\lambda(1-\tau), 1\}, \quad v \sim \Gamma(\lambda\tau),$$

$$y = \sigma\lambda(\log u - \log v) + \mu.$$

The former parametrization is useful for deriving the first two moments of $y \sim \tilde{p}_F(y|\mu, \sigma, \tau, \lambda)$, in fact

$$E(y) = E\left[\sigma\lambda \log\left(\frac{z}{1-z}\right) + \mu\right] = \sigma\lambda \log E\left[\left(\frac{z}{1-z}\right)\right] + \mu = \sigma\lambda\big\{\Gamma[\lambda(1-\tau)] - \Gamma(\lambda\tau)\big\} + \mu,$$

where $\psi(x)$ is the digamma function

$$\psi(x) = \frac{dlog\Gamma(x)}{dx} = \frac{\Gamma'(x)}{\Gamma(x)}.$$

The variance is

$$var(x) = var\left[\sigma\lambda \log\left(\frac{z}{1-z}\right) + \mu\right] = \sigma^2\lambda^2\big\{\psi'[\lambda(1-\tau)] + \psi'(\lambda\tau)\big\},$$

where $\psi'(x) = d\psi(x)/dx$ is the trigamma function.