Received: Added at production | Revised: Added at production | Accepted: Added at production
DOI: xxx/xxxx

LETTER

Re: Spline-based accelerated failure time model

Mark Clements*[1] | Benjamin Christoffersen[1] | Patrick Royston[2] | Michael Crowther[1]

[1]Department of Medical Epidemiology and Biostatistics, Karolinska Institutet, Stockholm, Sweden
[2]MRC Clinical Trials Unit at UCL, Institute of Clinical Trials and Methodology, University College London, London, UK

Correspondence
*Mark Clements, Department of Medical Epidemiology and Biostatistics, Karolinska Institutet, PO Box 281, SE-171 77, Stockholm, Sweden. Email: mark.clements@ki.se

Dear Editor,

Pang and colleagues provide a thoughtful development of an accelerated failure time (AFT) model with a smooth baseline hazard function[1]. We welcome this contribution to the literature.

The authors state that "it is impossible to derive the score function for the joint log-likelihood, due to the difficulty in obtaining the first derivative with respect to β in [Equation] (10), when it is considered as an unknown parameter in the spline-basis function $S_k(\cdot)$ that needs to be calculated using recursive formulas". Note that the score equations *can* be readily be calculated, which simplifies the statistical development. Following the notation in Pang et al[1], the log likelihood is calculated as

$$\text{logL} = \sum_{i=1}^{n} \left[\delta_i \left(\boldsymbol{\beta}^T \boldsymbol{X}_i + \sum_{k=1}^{K} \gamma_k S_k \left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) t_i \right) \right) - \int_0^{t_i} \exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) \exp\left(\sum_{k=1}^{K} \gamma_k S_k \left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) u \right) \right) du \right]$$

where i is an index over individuals, δ_i is an event indicator, t_i is the observed time, $\boldsymbol{X}_i = (X_{ij})$ is a vector of covariates, $\boldsymbol{\beta} = (\beta_j)$ is a vector of coefficients for the log acceleration factors, and γ_k are the coefficients for the k-th spline basis $S_k(\cdot)$. This clearly shows that the evaluation times for $S_k(\cdot)$ depend on $\boldsymbol{\beta}$. This log-likelihood could be maximised using derivative-free optimisation methods or use gradient-based optimisation. The gradient of the log likelihood with respect to β_j is

$$\begin{aligned} \frac{\partial \text{logL}}{\partial \beta_j} = \sum_{i=1}^{n} \Bigg[& \delta_i X_{ij} \left(1 + \sum_{k=1}^{K} \gamma_k S_k' \left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) t_i \right) \exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) t_i \right) - \\ & \int_0^{t_i} \exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) \exp\left(\sum_{k=1}^{K} \gamma_k S_k \left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) u \right) \right) X_{ij} \times \\ & \left(1 + \sum_{K=1}^{K} \gamma_k S_k' \left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) u \right) \exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i \right) u \right) du \Bigg] \end{aligned}$$

This calculation requires the evaluation of both the B-spline basis functions $S_k(\cdot)$ and their derivatives $S_k'(\cdot)$. Note that derivatives for B-splines can be readily calculated. The gradient of the log-likelihood with respect to γ_j is given in the Supporting Information. The Hessian can be derived analytically or calculated using finite differences of the gradients, which will allow for estimation of the covariance matrix. This would neatly avoid the need for both alternating conditional estimation and the bootstrap for variance estimation.

The authors stated that they were not aware of a model development with time-dependent acceleration factors. We recently implemented AFT models in R (`rstpm2::aft`) and Stata (`staft`) which allow for time-dependent acceleration factors. For details of our approach, see Crowther et al [2].

The spline knots for the baseline survival function may implicitly depend on the size of the acceleration factors. As a potential issue, the implementation by Pang and colleagues chooses the upper boundary knot as max $\left(\exp\left(\boldsymbol{\beta}^T \boldsymbol{X}_i\right) t_i\right)$, which is explicitly a function of the acceleration factors. This makes the evaluation of the analytical gradients difficult. Author BC has implemented a C++ implementation of the authors' model (https://github.com/boennecd/Spline-AFT-Model) with fixed boundary knots, Gauss–Legendre quadrature and analytical gradients. For the colon application provided by Pang et al [1] and using a fast laptop, the authors' R code took approximately 24000 seconds for a single fit without bootstrapping. In contrast, `smoothSurv`, our re-implementation, `rstpm2::aft` and `staft` took approximately 1500, 2, 0.4 and 0.15 seconds, respectively, including variance calculations. We do not recommend using the authors' current implementation.

References

1. Pang M, Platt RW, Schuster T, Abrahamowicz M. Spline-based accelerated failure time model. *Statistics in Medicine* 2021; 40(2): 481–497. doi: 10.1002/sim.8786

2. Crowther MJ, Royston P, Clements M. A flexible parametric accelerated failure time model. https://arxiv.org/abs/2006.06807; 2020.

□