## Statistical Appendix: Cox survival analysis using Gaussian process priors

In this supplementary file, we describe in a detail how to apply the Gaussian processes (GP) in Cox survival analyses using the proportional hazards model. This statistical methodology is applied in the paper "Stratification of the risk for gastrointestinal stromal tumour recurrence after surgery: a combined analysis of ten population-based cohorts".

For the individual $i$, where $i = 1, \ldots, n$, we have observed survival time $y_i$ (possibly right censored) with a censoring indicator $\delta_i$, where $\delta_i = 0$ if the $i$th observation is uncensored and $\delta_i = 1$ if the observation is right censored. The traditional approach to analyze continuous time-to-event data is to assume the Cox proportional hazard function[1]

$$h_i(t) = h_0(t) \exp(\boldsymbol{x}_i^T \boldsymbol{\beta}), \tag{1}$$

where $h_0$ is the unspecified baseline hazard rate, $\boldsymbol{x}_i$ is the $d \times 1$ vector of covariates for the $i$th patient and $\boldsymbol{\beta}$ is the vector of regression coefficients. The matrix $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^T$ of size $n \times d$ includes all covariate observations.

The Cox model with a linear predictor can be extended to more general form to enable, for example, additive and non-linear effects of covariates.[2,3] We extend the proportional hazards model by

$$h_i(t) = \exp(\log(h_0(t)) + \eta_i(\boldsymbol{x}_i)), \tag{2}$$

where the linear predictor is replaced with the latent predictor $\eta_i$ depending on the covariates $\boldsymbol{x}_i$. By assuming a Gaussian process prior[4] over $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n)^T$, smooth nonlinear effects of continuous covariates are possible, and if there are dependencies between covariates, the GP can model these interactions implicitly. A zero-mean GP prior is set for $\boldsymbol{\eta}$, which results in the zero-mean multivariate Gaussian distribution

$$p(\boldsymbol{\eta}|X) = \mathcal{N}(\mathbf{0}, C(X, X)), \tag{3}$$

where $C(X, X)$ is the $n \times n$ covariance matrix whose elements are given by the covariance function of the GP. The covariance function defines the smoothness and scale properties of the latent function, and we choose a sum of constant and non-stationary neural network covariance function[5]

$$c(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sigma_c + \frac{2}{\pi} \sin^{-1}\left( \frac{2\tilde{\boldsymbol{x}}_i^{\mathrm{T}} \Sigma \tilde{\boldsymbol{x}}_j}{(1 + 2\tilde{\boldsymbol{x}}_i^{\mathrm{T}} \Sigma \tilde{\boldsymbol{x}}_i)(1 + 2\tilde{\boldsymbol{x}}_j^{\mathrm{T}} \Sigma \tilde{\boldsymbol{x}}_j)} \right), \tag{4}$$

where $\sigma_c$ is the constant covariance part, $\tilde{\boldsymbol{x}} = (1, x_1, \ldots, x_d)^{\mathrm{T}}$ is an input vector augmented with 1, and $\Sigma = \mathrm{diag}(\sigma_0^2, \sigma_1^2, \ldots, \sigma_d^2)$ is a diagonal weight prior, where $\sigma_0^2$ is a variance for the bias parameter controlling the functions offset from the origin and $\sigma_1^2, \ldots, \sigma_d^2$ are the variances for the weight parameters. The constant covariance part models the mean hazard level, and the neural network covariance part models the nonlinear function. A neural network covariance function was chosen, since it is suitable for modeling saturating effects, and based on cross-validation, it resulted in a better predictive performance as compared to a squared exponential or Matérn covariance function. The constant covariance was fixed to $\sigma_c = 1$, and uniform prior on $\sigma_0, \sigma_1, \ldots, \sigma_d$ was used for hierarchical standard deviations as recommended by Gelman[6].

A piecewise log-constant baseline hazard[7] is assumed by partitioning the time axis into $K$ intervals with equal lengths: $0 = s_0 < s_1 < s_2 < \ldots < s_K$, where $s_K > y_i$ for all $i = 1, \ldots, n$. In the interval $k$, where $k = 1, \ldots, K$, hazard is assumed to be constant:

$$h_0(t) = \lambda_k \quad \text{for} \quad t \in (s_{k-1}, s_k]. \tag{5}$$

For the $i$th individual the hazard rate in the $k$th time interval is then

$$h_i(t) = \exp(f_k + \eta_i(\boldsymbol{x}_i)), \quad t \in (s_{k-1}, s_k], \tag{6}$$

where $f_k = \log(\lambda_k)$. To assume smooth hazard rate functions, we place another Gaussian process prior for $\boldsymbol{f} = (f_1, \ldots, f_K)^T$. We define a vector containing the mean locations of $K$ time intervals as $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)^T$. The GP prior assumed for the logarithm of the hazard rate results in

$$p(\boldsymbol{f}|\boldsymbol{\tau}) = \mathcal{N}(\mathbf{0}, C_\tau(\boldsymbol{\tau}, \boldsymbol{\tau})), \tag{7}$$

where the covariance matrix $C_\tau$ is of size $K \times K$. In the study, we fixed K=200. The covariance function for modelling of the hazard rate was chosen to be the neural network covariance function.

The likelihood contribution for the possibly right censored $i$th observation $(y_i, \delta_i)$ is assumed to be

$$l_i = h_i(y_i)^{(1-\delta_i)} \exp\left(-\int_0^{y_i} h_i(t)dt\right). \tag{8}$$

Using the piecewise log-constant assumption for the hazard rate function, the contribution of the observation $i$ for the likelihood results in

$$l_i = [\lambda_k \exp(\eta_i)]^{(1-\delta_i)} \exp\left(-[(y_i - s_{k-1})\lambda_k + \sum_{g=1}^{k-1}(s_g - s_{g-1})\lambda_g] \exp(\eta_i)\right), \tag{9}$$

where $y_i \in (s_{k-1}, s_k]$.[3,7] By applying the Bayes theorem, the prior information and likelihood contributions are combined, and the posterior distribution of the latent variables can be computed. Due to the form of the likelihood function, the resulting posterior becomes non-Gaussian and analytically exact inference is intractable.

We use a Gaussian approximation to integrate over the latents variables $\boldsymbol{\eta}$ and $\boldsymbol{f}$. The posterior distribution is approximated by doing a second order Taylor expansion of the logarithm of the posterior around the posterior mode, as presented by Rasmussen and Williams.[4] We select the hyperparameters of the covariance function using type II maximum a posterior (MAP) estimation. In the computation of predictive densities, we use Monte Carlo approximation by drawing 10000 latent samples from the joint Gaussian posterior.

The following transformations were used for the covariates. Square root transformation was applied to reduce the skewness of tumour size and mitotic count. Square root of tumor size, square root of mitotic count and age were normalized to have zero mean and unit variance. Tumour rupture status was coded as missing=[0,0], no-rupture=[1,0] and rupture=[0,1]. The site of ruptured tumour was coded as a binary indicator vector.

The receiver operating characteristics (ROC) curves and the corresponding areas under the curve (AUC) for the pooled series were computed using ten-fold cross-validation to simulate predictive accuracy in an unseen population[8]. The Bayesian confidence intervals and pairwise comparison probabilities (Bayesian $p$-values) for the AUC values for the pooled and validation series were computed using Bayesian bootstrap as described in the reference[8].

In the prognostic contour maps (Fig. 5) the probability of tumour recurrence was stratified to make it easier to read the colour coding. This influenced the predictive accuracy of the model only marginally reducing the AUC-value by 0·003. When computing the maps, all patients were included in the analysis, and the tumour rupture status was coded either as missing, present or absent.

## References


1 Cox D R. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)* 1972; **34**: 187-220.

2 Kneib T. Mixed model-based inference in geoadditive hazard regression for interval-censored survival times. *Computational Statistics & Data Analysis* 2006; **51**: 777-92.

3 Martino S, Akerkar R, Rue H. Approximate Bayesian inference for survival models. *Scandinavian Journal of Statistics* 2011; **38**: 514–28.

4 Rasmussen C, Williams C. Gaussian Processes for Machine Learning, MIT Press, Cambridge, 2006.

5 Williams C. Computation with infinite neural networks. *Neural Computation* 1998; **10**: 1203–16.

6 Gelman A. Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis* 2006; **1**: 515–34.

7 Ibrahim, J G, Chen M-H, Sinha D. Bayesian Survival Analysis, Springer, New York, 2001.

8 Vehtari, A, Lampinen, J. Bayesian model assessment and comparison using cross-validation predictive densities. *Neural Computation* 2002, **14**: 2439–68.