Binacox: automatic cut-point detection in high-dimensional Cox model with applications in genetics

Simon Bussy\textsuperscript{1,2}, Mokhtar Z. Alaya\textsuperscript{3}, Anne-Sophie Jannot\textsuperscript{4}, Agathe Guilloux\textsuperscript{5}

Abstract

We introduce \textit{binacox}, a prognostic method to deal with the problem of detecting multiple cut-points per feature in a multivariate setting where a large number of continuous features are available. The method is based on the Cox model and combines one-hot encoding with the \textit{binarsity} penalty, which uses total-variation regularization together with an extra linear constraint, and enables feature selection. Original nonasymptotic oracle inequalities for prediction (in terms of Kullback–Leibler divergence) and estimation with a fast rate of convergence are established. The statistical performance of the method is examined in an extensive Monte Carlo simulation study, and then illustrated on three publicly available genetic cancer datasets. On these high-dimensional data sets, our proposed method outperforms state-of-the-art survival models regarding risk prediction in terms of the C-index, with a computing time orders of magnitude faster. In addition, it provides powerful interpretability from a clinical perspective by automatically pinpointing significant cut-points in relevant variables.

KEYWORDS
feature binarization, genetic cancer data, nonasymptotic oracle inequality, proximal methods, survival analysis, total variation

1 INTRODUCTION

Determining significant prognostic biomarkers is of increasing importance in many areas of medicine. Scores used in clinical practice often categorize continuous features into binary ones using expert-driven cut-points. For instance, the Wells score, which categorizes patients into low-, moderate-, and high-risk groups for pulmonary embolism (Wells \textit{et al.}, 2000), is one of the most extensively validated predictive scores. One of the categorized features used in this score is “having a heart rate of over 100 beats per minute, or not.” When used in routine care, this type of threshold makes a score more interpretable from a clinical point of view: above this threshold, patients have higher risk of unfavorable outcome. Another well-known clinical example where finding optimal thresholds is paramount is the platelet transfusion decision, based on a platelet-count threshold (Curley \textit{et al.}, 2019); or the piece-wise constant effect assumption made on the body mass index where prognoses are generally worst for obese and underweight individuals (Bédat \textit{et al.}, 2015) (hence with two cut-points here).

With the increasing availability of high-dimensional data sets, data-driven predictive scores are becoming increasingly important. A convenient tool for finding multiple cut-points in a multivariate and high-dimensional setting—and then automatically building interpretable predictive scores—is therefore of high interest. For instance, in genetic oncology studies, similar questions occur because the effect of certain genes’ expression on survival times is often nonlinear. Therefore, to develop such scores, one has to deal with a two-sided problem:
first to select relevant features, and second to find relevant thresholds—also called cut-off values or cut-points—for these selected continuous features, without prior or expert knowledge.

Solving this problem means applying nonlinearities to feature effects that most models cannot detect. This also offers the ability to classify patients into several groups in terms of their continuous feature values relative to the cut-points. More importantly, this can also lead to a better understanding of the features’ effects on the outcome of interest; this strategy might uncover biological thresholds as well as potential criteria for new prospective studies, help diagnose diseases, and make treatment recommendations.

Indeed, good cut-point detection is a common issue in medical studies, and numerous methods have been proposed for determining a single cut-point for a given feature. This ranges from choosing the mean or median, to methods based on distribution of values, or association with clinical outcomes, for example, the minimal p-value from multiple log-rank tests, see, for instance, Rota et al. (2015). However, the choice of the actual cut-points is not a straightforward problem, even for a single cut-point (Lausen and Schumacher, 1992; Contal and O’Quigley, 1999; Klein and Wu, 2003). Recently, Icuma et al. (2018) proposed a Bayesian approach with accelerated failure time modeling, but still only allowing one cut-point per feature.

Indeed, although many studies have been devoted to finding one optimal cut-point, there is often need in medical settings to determine not only one but multiple cut-points (e.g., the body mass index example discussed in the introduction). Methods exist to deal with multiple cut-point detection for one-dimensional signals (see, for instance, Bleakley and Vert (2011)) and Harchaoui and Lévy-Leduc (2010) that use a group fused Lasso or total-variation penalty, respectively, and for multivariate time series (Cho and Fryzlewicz, 2015). Although cut-point detection is also a paramount issue in survival analysis (Faraggi and Simon, 1996), methods that have been developed in this setting only look at a single feature at a time (e.g., Motzer et al. (1999) and Leblanc and Crowley (1993) that use survival trees, or more recently Chang et al. (2019)). To our knowledge, a multivariate survival analysis method well-suited to detect multiple cut-points per feature in a high-dimensional setting has not been previously proposed.

Let us consider the usual survival analysis framework. Following Andersen et al. (2012), let nonnegative random variables T and C stand for the time of the event of interest and censoring time, respectively, and X denote the p-dimensional vector of features (e.g., patient characteristics, therapeutic strategy, and atomic features). The event of interest could be, for instance, death, re-hospitalization, relapse, or disease progression. Conditionally on X, T and C are assumed to be independent, which is classical in survival analysis (Klein and Moeschberger, 2005). We then denote Z the right-censored time and Δ the censoring indicator, defined as

\[ Z = T \wedge C \quad \text{and} \quad \Delta = 1(T \leq C), \]

respectively, where a \wedge b denotes the minimum between two numbers a and b, and 1(\cdot) the indicator function taking the value 1 if the condition in (\cdot) is satisfied and 0 otherwise.

The Cox proportional hazards model (Cox, 1972) is by far the most widely used in survival analysis. It describes the relation between the hazard function and the features by

\[ \lambda(t|X = x) = \lambda_0(t)e^{x^T \beta_{\text{cox}}}, \]

where \( \lambda_0 \) is a baseline hazard function describing how the event risk changes over time at baseline levels of features, and \( \beta_{\text{cox}} \in \mathbb{R}^p \) a vector quantifying the multiplicative impact on the hazard ratio of each feature.

High-dimensional settings are becoming increasingly frequent, in particular for genetic data applications where cut-point estimation is a common problem (see, for instance, Cheang et al. (2009)), but also in other contexts where the number of available features to consider as potential risk factors is tremendous, particularly with the development of electronic health records. A penalized version of the Cox model well suited for such settings is proposed in Simon et al. (2011), but it cannot model nonlinearity. Theory for using lasso-type methods in the Cox model was developed in Huang et al. (2013). Other methods have been put forward to deal with this problem in similar settings, like boosting Cox models (Li and Luan, 2005) and random survival forests (Ishwaran et al., 2008). However, none of these identify cut-point values, which is of major interest for both interpretation and clinical benefit.

In this paper, we propose a method called binacox that estimates multiple cut-points in a Cox model with high-dimensional features. First, the binacox method one-hot encodes the continuous input features (Wu and Coggshall, 2012) through a mapping to a new binarized space of much higher dimension, and then trains the Cox model in this space, regularized with the binarsity penalty (Alaya et al., 2019) that combines total-variation regularization with an extra sum-to-zero constraint, and enables feature selection. Cut-points of the initial continuous input features are then detected by the jumps in the regression coefficient vectors, which the binarsity penalty forces to be piecewise-constant. The main contribution of this paper is twofold. First we introduce the idea of using a total-variation penalty with an extra linear constraint on the
weights of a Cox model trained on a binarization of the raw continuous features. This leads to a procedure that automatically detects relevant features and allows multiple cut-points per feature. Second, the oracle inequality in prediction of Section 3 (see Theorem 1) is stated in terms of Kullback–Leibler divergence, as opposed to the results in Huang et al. (2013) (for the lasso penalty) expressed in Bregman divergence. The arguments used to obtain our results are then different, and also differ from the ones used in Kong and Nan (2014). A precise description of the model is given in Section 2. Section 3 highlights the good theoretical properties of the binacox method by establishing fast oracle inequalities for prediction and for estimation. Section 4 presents the simulation procedure used to evaluate the performance of our method and compares it with existing ones. In Section 5, we apply our method to high-dimensional genetic data sets. Finally, we discuss the obtained results in Section 6.

2 | MODEL AND METHOD

Consider an independent and identically distributed sample

\[(X_1, Z_1, \Delta_1), \ldots, (X_n, Z_n, \Delta_n) \in [0,1]^p \times \mathbb{R}_+ \times \{0,1\},\]

where the condition \(X_i \in [0,1]^p\) for all \(i = 1, \ldots, n\) is always true after an appropriate rescaling preprocessing step, without loss of generality. Let \(\mathbf{X} = [X_{i,j}]_{1 \leq i \leq n, 1 \leq j \leq p}\) be the \(n \times p\) fixed design matrix vertically stacking the \(n\) samples of \(p\) raw features so that \(X_{i,.} = X_i\). In order to simplify the presentation of our results, we assume in the paper that the raw features \(X_{i,j}\) are continuous for all \(j = 1, \ldots, p\), but this is not a limitation in practice. Assume that the hazard function for patient \(i\) is given by

\[\lambda^*(t|X_i) = \lambda_0^*(t)e^{f^*(X_i)},\]

(1)

where \(\lambda_0^*(t)\) is the baseline hazard function. Our goal is to estimate \(f^*\).

2.1 | Binarization

Let \(\mathbf{X}_l^B\) be the sparse binarized matrix with an extended number \(p + d\) of columns, typically with \(d \gg p\), where continuous input features have been one-hot encoded (Wu and Coggeshall, 2012). The \(j\)th column \(X_{i,.j}\) is then replaced by \(d_j + 1 \geq 2\) columns \(X_{i,1,j}, \ldots, X_{i,d_j+1,j}\) containing only zeros and ones, where the \(i\)th row \(X_i^B \in \mathbb{R}^{p+d}\) with

\[d = \sum_{j=1}^p d_j\]

is written

\[X_i^B = (X_{i,1,1}, \ldots, X_{i,1,d_j+1}, \ldots, X_{i,p,1}, \ldots, X_{i,p,d_j+1})^T.\]

We consider the intervals \(I_{j,1}, \ldots, I_{j,d_j+1}\) defining a partition of \([0,1]\), that is

\[\bigcup_{k=1}^{d_j+1} I_{j,k} = [0,1]\]

and \(I_{j,k} \cap I_{j,k'} = \emptyset\) for all \(k \neq k'\) with \(k, k' = 1, \ldots, d_j + 1\). Now for \(i = 1, \ldots, n\) and \(l = 1, \ldots, d_j + 1\), we define

\[X_{i,j,l}^B = \begin{cases} 1 & \text{if } X_{i,j} \in I_{j,l}, \\ 0 & \text{otherwise.} \end{cases}\]

We then denote \(I_{j,l} = (\mu_{j,l-1}, \mu_{j,l})\) for \(l = 1, \ldots, d_j + 1\), with the convention \(\mu_{j,0} = 0\) and \(\mu_{j,d_j+1} = 1\). A natural choice for the \(j\)th feature is given by the quantiles, namely \(\mu_{j,l} = q_j(l/(d_j + 1))\), where \(q_j(\alpha)\) denotes a quantile of order \(\alpha \in [0,1]\) for \(X_{i,j}\). If training data also contain unordered qualitative features, one-hot encoding with \(\ell_1\)-penalization can be used, for instance.

To each binarized feature \(X_{i,j,l}^B\) corresponds a parameter \(\beta_{j,l}\), and the vectors associated with the binarization of the \(j\)th feature are naturally denoted \(\beta_{j,*} = (\beta_{j,1}, \ldots, \beta_{j,d_j+1})^T\) and \(\mu_{j,*} = (\mu_{j,1}, \ldots, \mu_{j,d_j+1})^T\). Hence, we define a candidate for the estimation of \(f^*\) defined in (1) as

\[f_{\beta}(X_i) = \beta^T X_i^B = \sum_{j=1}^p \sum_{l=1}^{d_j+1} \beta_{j,l} \mathbb{1}(X_{i,j} \in I_{j,l}).\]

(2)

The full parameter vectors of size \(p + d\) and \(d\), respectively, are finally obtained by concatenation of the vectors \(\beta_{j,*}\) and \(\mu_{j,*}\), that is,

\[\beta = (\beta_{1,*}^T, \ldots, \beta_{p,*}^T)^T = (\beta_{1,1}, \ldots, \beta_{1,d_j+1}, \ldots, \beta_{p,1}, \ldots, \beta_{p,d_j+1})^T,\]

and

\[\mu = (\mu_{1,*}^T, \ldots, \mu_{p,*}^T)^T = (\mu_{1,1}, \ldots, \mu_{1,d_j}, \ldots, \mu_{p,1}, \ldots, \mu_{p,d_j})^T.\]

2.2 | Estimation procedure

In the following, for a fixed vector \(\mu\) of quantization, we define the binarized partial negative log-likelihood
(rescaled by $1/n$) as follows:

$$
\epsilon_n(f_\beta) = -\frac{1}{n} \sum_{i=1}^n \Delta_i \left\{ f_\beta(X_i) - \log \sum_{j'=2}^Z \exp f_\beta(X_{i,j'}) \right\}.
$$

(3)

Our approach consists in minimizing the function $\epsilon_n$ plus the binarity penalization term introduced in Alaya et al. (2019). The resulting optimization problem is written

$$
\hat{\beta} \in \arg\min_{\beta \in \mathcal{B}_{p+d}(R)} \left\{ \epsilon_n(f_\beta) + \text{bina}(\beta) \right\},
$$

(4)

where $\mathcal{B}_{p+d}(R) = \{ \beta \in \mathbb{R}^{p+d} : \sum_{j=1}^p \| \beta_j \|_\infty \leq R \}$ and

$$
\text{bina}(\beta) = \sum_{j=1}^p \left( \sum_{l=2}^{d_j+1} \omega_j |\beta_{j,l} - \beta_{j,l-1}| + \delta_j(\beta_{j,*}) \right),
$$

(5)

with

$$
\delta_j(u) = \begin{cases} 
0 & \text{if } n_j^T u = 0, \\
\infty & \text{otherwise},
\end{cases}
$$

and where $n_j = (n_{j,1}, ..., n_{j,d_j+1})^T \in \mathbb{N}^{d_j+1}$ with $n_{j,l} = |\{ i = 1, ..., n : X_{i,j,l} \in I_{j,l} \}|$ for all $j = 1, ..., p$ and $l = 1, ..., d_j + 1$. The constraint over $\mathcal{B}_{p+d}(R)$ is standard in the literature for obtaining proofs of oracle inequalities for sparse generalized linear models (Van de Geer et al., 2008), and is discussed in detail in Section 3 right after Theorem 1. For a given numerical constant $c > 0$, the weights $\omega_{j,l}$ have an explicit form given by

$$
\omega_{j,l} = 11.32 \sqrt{\frac{c + \log(p + d) + L_{n,c}}{n}} V_{j,l}
+ 18.62 \frac{c + 1 + \log(p + d) + L_{n,c}}{n}
= \Theta \left( \sqrt{\frac{\log(p + d)}{n}} V_{j,l} \right),
$$

where

$$
L_{n,c} = 2 \log \log \frac{2nV_{j,l} + 18.66(c + \log(p + d))}{8}
$$

and with

$$
V_{j,l} = \left\{ i = 1, ..., n : X_{i,j,l} \in \bigcup_{u=l}^{d_j+1} I_{j,u} \right\} / n.
$$

It turns out that the binarity penalty is well suited to our problem. First, it tackles the problem that $X^0$ is not full rank by construction, since $\sum_{l=1}^{d_j+1} X_{i,j,l} = 1$ for all $j = 1, ..., p$, which means that the columns in each block sum to $1$. This problem is solved since the penalty imposes the linear constraint $\sum_{l=1}^{d_j+1} n_{j,l} \beta_{j,l} = 0$ in each block with the $\delta_j(\cdot)$ term. Note that if the $I_{j,l}$ are taken as the interquantile intervals, we have that $n_{j,l}$ are all equal for $l = 1, ..., d_j + 1$, and we get the standard sum-to-zero constraint $\sum_{l=1}^{d_j+1} \hat{\beta}_{j,l} = 0$. Then, the other term in the penalty consists of a within-block weighted total variation penalty:

$$
\| \hat{\beta}_{j,*} \|_{\text{TV,} \omega_{j,*}} = \sum_{l=2}^{d_j+1} \omega_{j,l} |\hat{\beta}_{j,l} - \hat{\beta}_{j,l-1}|,
$$

(6)

which takes advantage of the fact that within each block, binarized features are ordered. The effect is then to keep the number of different values taken by $\hat{\beta}_{j,*}$ to a minimum, which makes significant cut-points appear, as detailed hereafter.

For all $\beta \in \mathbb{R}^{p+d}$, let $A(\beta) = [A_1(\beta), ..., A_p(\beta)]$ be the concatenation of the support sets relative to the total-variation penalization, namely

$$
A_j(\beta) = \{ l : \beta_{j,l} \neq \beta_{j,l-1}, \text{ for } l = 2, ..., d_j + 1 \}
$$

for all $j = 1, ..., p$. Similarly, we denote $A^c(\beta) = [A^c_1(\beta), ..., A^c_p(\beta)]$ the complementary set of $A(\beta)$. We then write

$$
A_j(\hat{\beta}) = \{ \hat{l}_{j,1}, ..., \hat{l}_{j,s_j} \},
$$

(7)

where $\hat{l}_{j,1} < ... < \hat{l}_{j,s_j}$ and $s_j = |A_j(\hat{\beta})|$. Some details on the algorithm used to solve the regularization problem (4) are given in Appendices A.5 (with, among others, some explanations about an ad hoc de-noising step) and A.6 in the Supplementary Material.

## 3 | THEORETICAL GUARANTEES

### 3.1 | A fast oracle inequality for prediction

This section is devoted to a first theoretical result. In order to evaluate the prediction error, we first define the (empirical) Kullback–Leibler divergence (Senoussi, 1990) $KL_n$ between the true function $f^*$ and any candidate $f$ as

$$
KL_n(f^*, f) = \frac{1}{n} \sum_{i=1}^n \int_0^\tau \log \left( \frac{e^{f(X_i)} \sum_{l=1}^n Y_i(t) e^{f(X_i)}}{e^{f(X_i)} \sum_{l=1}^n Y_i(t) e^{f^*(X_i)}} \right) \times Y_i(t) \lambda_0^*(t) e^{f^*(X_i)} dt,
$$

(8)
The inequality $\Lambda_0^+(\tau) = \int_0^\tau \lambda_0^+(s) \, ds$.

**Assumption 2.** Let $\varepsilon \in (0, 1)$ and define $t_{n,p,d,\varepsilon}$ as the solution of

$$2.221 (p + d)^2 \exp\{-n t_{n,p,d,\varepsilon}^2/(2 + 2 t_{n,p,d,\varepsilon}^2/3)\} = \varepsilon.$$ 

For any concatenation set $L = [L_1, \ldots, L_p]$ such that $\sum_{j=1}^p |L_j| \leq K^*$, assume that $\kappa_t^2(L) \geq \Xi_t(L)$, where

$$\Xi_t(L) = 4 |L| \left( \frac{8 \max_j (d_j + 1) \max_{j,i} \omega_{ji}}{\min_{j,i} \omega_{ji}} \right)^2 \left\{ \left( 1 + e^{2/\kappa_{t}^2(\tau)} \right)^2 \lambda_t^+(\tau) \right\}.$$ 

Note that $\kappa_t^2(L)$ is the smallest eigenvalue of a population integrated covariance matrix defined in (9), so it is reasonable to treat it as a constant. Moreover, $t_{n,p,d,\varepsilon}^2$ is of order $n^{-1} \log ((p + d)^2/\varepsilon)$, so if $|L| \log (p + d)/n$ is sufficiently small, Assumption 2 is verified. With these preparations made, let us now state the oracle inequality for prediction satisfied by our estimator of $f^*$, which is, by construction, given by $\hat{f} = \hat{f}_{\hat{\beta}}$ (see (2)).

**Theorem 1.** The inequality

$$K L_n(f^*, f_{\hat{\beta}}) \leq \inf_{\hat{\beta}} \left\{ 3 K L_n(f^*, f_{\hat{\beta}}) + \frac{1024 (f^\infty_{\hat{\beta}} + R + 2) |A(\hat{\beta})| \max_{1 \leq j \leq p} \|\omega_{j*}\|_{A(\hat{\beta})}^2}{\kappa_t^2(\hat{\beta}) - \Xi_t(\hat{\beta})} \right\}$$

holds with a probability greater than $1 - 57.1 e^{-c} - e^{-n s^{(0)}(\tau)^2/8c^2 t_{n,p,d,\varepsilon}^2} - 3 \varepsilon$ for some $c > 0$, where the infimum is over the set of vectors $\hat{\beta} \in \mathbb{R}_{p+d}(R)$ such that $n_{t,j}^{(0)} \hat{\beta}_{j*} = 0$ for all $j = 1, \ldots, p$, and such that $|A(\hat{\beta})| \leq K^*$.

The proof of Theorem 1 is postponed to Appendix B in the Supplementary Material. The second term in the right-hand side of (11) can be viewed as a “variance” (or “complexity”) term, and its dominant term satisfies

$$\frac{|A(\hat{\beta})| \max_{1 \leq j \leq p} \|\omega_{j*}\|_{A(\hat{\beta})}^2}{\kappa_t^2(\hat{\beta}) - \Xi_t(\hat{\beta})} \leq \frac{|A(\hat{\beta})|}{\kappa_t^2(\hat{\beta}) - \Xi_t(\hat{\beta})} \log (p + d) / n,$$

where the symbol $\leq$ means that the inequality holds up to a multiplicative constant. Then, one obtains the expected fast convergence rate for the variance $O(\log (p + d)/n)$ for the estimator $\hat{f}$. In Section 3.2, we adapt Theorem 1 to the case...
where the true $f^*$ lies in a Cox model with cut-points, while the case of a Cox generalized additive model with Lipschitz components is considered in Appendix A.4, with a rate of convergence of order $O(|\alpha|^2n^{-\frac{1}{4}})$.

The value $|A(\beta)|$ characterizes the sparsity of the vector $\beta$, since it counts the number of nonequal consecutive values of $\beta$. If $\beta$ is block-sparse, namely whenever $|A(\beta)| \ll p$ where $A(\beta) = \{j = 1, ..., p : \beta_j, \neq 0\}$ (meaning that few raw features are useful for prediction), then $|A(\beta)| \leq |A(\beta)| \max_{j \in A(\beta)} |A_j(\beta)|$, which means that $|A(\beta)|$ is controlled by the block sparsity $|A(\beta)|$. Also, the oracle inequality still holds for vectors such that $n_{j,k}^\star \beta_{j,k} = 0$, which is natural since the binarsity penalization imposes these extra linear constraints.

The assumption $\beta \in \mathcal{B}_{p+d}(R)$ is a technical one, allowing a connection, via the notion of self-concordance (Bach, 2010), between the empirical squared $\ell_2$-norm and the empirical Kullback–Leibler (see Lemma 3). Also, note that

$$
\max\frac{}{ 1 \leq n \leq n} |\beta^\top X_n^\beta| \leq \sum_{j=1}^p \|\beta_j\|_{\infty} \leq |A(\beta)| \times \|\beta\|_{\infty}, \quad (12)
$$

where $\|\beta\|_{\infty} = \max_{1 \leq j \leq p} \|\beta_j\|_{\infty}$. The first inequality in (12) comes from the fact that the entries of $X^\beta$ are in $[0, 1]$, and entails that $\max_{1 \leq n \leq n} |\beta^\top X_n^\beta| \leq R$ whenever $\beta \in \mathcal{B}_{p+d}(R)$.

The second inequality in (12) shows that $R$ can be upper bounded by $|A(\beta)| \times \|\beta\|_{\infty}$, and therefore the constraint $\beta \in \mathcal{B}_{p+d}(R)$ becomes merely a box constraint on $\beta$, which depends on the dimensionality of the features through $|A(\beta)|$ only. The fact that the procedure depends on $R$, and that the oracle inequality stated in Theorem 1 depends linearly on $R$, is commonly found in the literature on sparse generalized linear models, see Van de Geer et al. (2008), Bach (2010), and Ivanoff et al. (2016). However, the constraint $\mathcal{B}_{p+d}(R)$ is a technicality that is not used in the numerical experiments in Sections 4 and 5.

Note in addition that our proof is different from that of Huang et al. (2013) and could be applied in their setting (Lasso in the Cox model with time-dependent covariates). Alternative oracle inequalities, in terms of the Kullback–Leibler divergence instead of the symmetric Bregman divergence, could hence be proven.

Our proof and result also differ from the ones of Kong and Nan (2014), which follows the lines of Van de Geer et al. (2008) adapting it to the Cox model. In particular, their bound is given for the excess risk from an expected partial likelihood (integrated also for the covariates distribution), whereas in our paper we bound an empirical Kullback–Leibler divergence introduced in Senoussi (1990) and which has the properties of a divergence.

### 3.2 Piecewise constant case

We now assume that the true $f^*$ lies in the cut-points model.

**Assumption 3.** Assume that $f^*$ has the following form

$$
f^*(X_i) = \sum_{j=1}^p f_j^*(x_{ij}) = \sum_{j=1}^p \sum_{k=1}^{K_j^*+1} \beta_{j,k}^* 1(X_{ij} \in I_{j,k}^*), \quad (13)
$$

with $I_{j,k}^* = (\mu_{j,k-1}^*, \mu_{j,k}^*)$ for $k = 1, ..., K_j^* + 1$ and where $\beta_{j,k}^* \neq \beta_{j,k+1}^*$ for $k = 1, ..., K_j^*$. We impose that

$$
\sum_{i=1}^n f_j^*(x_{ij}) = 0 \quad \text{for all } j = 1, ..., p \quad (14)
$$

to ensure identifiability.

The identifiability condition is common, see, for example, Meier et al. (2009) for a similar constraint in generalized additive models, which can also be written as a sum-to-zero constraint in each $\beta_j^*$’s block, that is

$$
\sum_{k=1}^{K_j^*+1} \beta_{j,k}^* n_{j,k}^* = 0 \quad \text{for all } j = 1, ..., p, \quad (15)
$$

where $n_{j,k}^* = \{|i = 1, ..., n : x_{ij} \in I_{j,k}^*\}$. This constraint could be replaced by $E[f_j^*(X_{ij})] = 0$ for all $j = 1, ..., p$. In Section 4, we use (14) to generate the data for the simulation study. For each feature $j = 1, ..., p$, the $\mu_{j,k}$’s $(k = 1, ..., K_j^*)$ are the so-called cut-points, and are such that $\mu_{j,1}^* < \mu_{j,2}^* < \cdots < \mu_{j,K_j^*}^*$, with the conventions $\mu_{j,0}^* = 0$ and $\mu_{j,K_j^*+1}^* = 1$. Denoting $K^* = \sum_{j=1}^p K_j^*$, the vector of regression coefficients $\beta^* \in \mathbb{R}^{K^*+p}$ is given by

$$
\beta^* = (\beta_{1,1}^*, ..., \beta_{p,1}^*, \beta_{1,2}^*, ..., \beta_{p,2}^*, ..., \beta_{1,K_1^*+1}^*, ..., \beta_{p,K_p^*}^*)^\top,
$$

and the cut-points vector $\mu^* \in \mathbb{R}^{K^*}$ by

$$
\mu^* = (\mu_{1,1}^*, ..., \mu_{p,1}^*, ..., \mu_{1,K_1^*}^*, ..., \mu_{p,K_p^*}).
$$

Under this assumption, our goal is now to simultaneously estimate $\mu^*$ and $\beta^*$, which also requires estimation of the unknown $K_j^*$ for all $j = 1, ..., p$. We obtain the following
\( \mu_j \)'s estimator

\[
\hat{\mu}_j, \star = (\hat{\mu}_{j,1}, \ldots, \hat{\mu}_{j,j+1})^T
\]

(16)

for all \( j = 1, \ldots, p \). By construction, \( K_j^* \) is estimated by \( \hat{K}_j = s_j \), see the lines following Equation (7) for a definition of the \( \hat{l}_{j,k} \) and \( s_j \). Finally, an estimator of \( \hat{\beta}^\star \) is obtained from (4).

**Theorem 2.** Fix \( \varepsilon > 0 \) and consider that intervals \( I_{j,l} \) (\( j = 1, \ldots, p \), \( l = 1, \ldots, d_j + 1 \)) are now chosen as interquantiles (as in Subsection 2.1), that is \( I_{j,l} = (\mu_{j,l-1}, \mu_{j,l}] \) with \( \mu_{j,l} = q_{j,l}(I/(D + 1)) \) where \( D \) is the integer part of

\[
6 \Delta_{\rho,\max}^2 K^* C^* f_{\infty}^* (\max_{1 \leq l \leq n} \int_0^1 \lambda^*(t | X_i) dt + \varepsilon) \cdot n^2,
\]

for some constant \( C_{cut} \) and with \( \Delta_{\rho,\max} = \max_{1 \leq l \leq p} \max_{K^* \leq K \leq K^* + 1} |\hat{\beta}_{j,k}^\star - \hat{\beta}_{j,k}| \), \( C_{cut}^* f_{\infty}^* = (1 + K^*) f_{\infty}^* \). Under Assumption 3, the following holds with high probability

\[
KL_n(f^\star, f_{\hat{\beta}}) \leq \left( 6 \Delta_{\rho,\max}^2 K^* C^* f_{\infty}^* (\max_{1 \leq l \leq n} \int_0^1 \lambda^*(t | X_i) dt + \varepsilon) + C_{cut} K^* |\hat{\alpha}|^\star \right) n^{-1}.
\]

The proof of Theorem 2 is postponed to Appendix C in the Supplementary Material. A fast rate oracle inequality for estimation in this context is established in Appendix A.3 in the Supplementary Material, and the special case where we consider a Cox generalized additive model with Lipschitz components for \( f^\star \) is considered in Appendix A.4.

### 4 | PERFORMANCE EVALUATION

#### 4.1 | Simulation

In order to assess the methods, we run an extensive Monte Carlo simulation study. Let us first present the design used in the following. We first take \( [X_{i,j}] \in \mathbb{R}^{n \times p} \sim \mathcal{N}(0, \Sigma(\rho)) \), with \( \Sigma(\rho) \) a \( (p \times p) \) Toeplitz covariance matrix (Mukherjee and Maht, 1988) with correlation \( \rho \in (0,1) \), such that \( \Sigma(\rho)_{i,j} = \rho^{\mid i - j \mid} \). For each feature \( j = 1, \ldots, p \), we sample the cut-points \( \mu_{j,k}^\star \) uniformly without replacement from the estimated quantiles \( q_j(u/10) \) for \( u = 1, \ldots, 9 \) and \( k = 1, \ldots, K_j^\star \). In this way, we avoid having undetectable cut-points (with very few examples above the cut-point value) or pairs of overly close together indissociable cut-points. We choose the same \( K_j^\star \) values for all \( j = 1, \ldots, p \). Now that the true cut-points vector \( \mu^\star \) has been generated, one can compute the corresponding binarized version of the features, which we denote \( x^\star_{j,l} \) for the \( l \)th example. Then, we generate \( c_{j,k} \sim (-1)^k \mathcal{N}(1,0.5) \) for all \( k = 1, \ldots, K_j^\star + 1 \) and \( j = 1, \ldots, p \) to make sure we create “real” cut-points, and take

\[
\hat{\beta}_{j,k}^\star = c_{j,k} - (K_j^\star + 1)^{-1} \sum_{k=1}^{K_j^\star + 1} c_{j,k}
\]

in order to impose the sum-to-zero constraint of the true coefficients in each block. We also induce a sparsity aspect by uniformly selecting a proportion \( r_s \) of features \( j \in S \) with no cut-point effect, that is, features for which we enforce \( \hat{\beta}_{j,k}^\star = 0 \) for all \( k = 1, \ldots, K_j^\star + 1 \). Finally, we generate \( U_j \sim \mathcal{U}([0,1]) \), \( U_j \sim \mathcal{U}([0,1]) \) stands for the uniform distribution on a segment \([a,b]\). The distribution of the censoring variable \( \zeta_j \) is the geometric distribution \( \mathcal{G}(\alpha_c) \), where \( \alpha_c \in (0,1) \) is empirically tuned to maintain a desired censoring rate \( r_c \in [0,1] \). The choice of all hyper-parameters is driven by the applications on real data presented in Section 5, and summarized in Table 1. Figure 1 gives an example of data generated according to the design we have just described.

We evaluate the methods being analyzed using two metrics. The first assesses the estimation of the cut-points values by

\[
m_1 = |S^\star|^{-1} \sum_{j \in S^\star} H(M_j^\star, \widehat{M}_j),
\]

where \( M_j^\star = \{\mu_{j,1}^\star, \ldots, \mu_{j,K_j^\star}^\star\} \) (respectively, \( \widehat{M}_j = \{\hat{\mu}_{j,1}, \ldots, \hat{\mu}_{j,K_j^\star}\} \) is the set of true (respectively, estimated) cut-points for feature \( j \), \( S^\star = \{j, j \in S \cap \{l, \widehat{M}_l = \emptyset\}\} \) the indexes corresponding to features with at least one true cut-point and one detected cut-point, and \( H(A,B) \) the Hausdorff distance between the sets \( A \) and \( B \), defined as \( H(A,B) = \max(\mathcal{E}(A|B), \mathcal{E}(B|A)) \), where \( \mathcal{E}(A|B) = \sup_{a \in A} \inf_{b \in B} |a - b| \). This is inspired by Harchaoui and Lévy-Leduc (2010), except that in our case,
Figure 1: Top: illustration of data simulated with $p = 2$, $K_1^* = K_2^* = 2$, and $n = 1000$. Dots represent failure times ($z_i = t_i$) while crosses represent censoring times ($z_i = c_i$), and the color gradient represents the $z_i$ values (red for low and blue for high). Bottom: $\beta^*$ is plotted, with a dotted line to demarcate the two blocks (since $p = 2$). To each interval, $I_{j,k}^*$ corresponds an effect $\beta_{j,k}^*$. Note that both $\beta_{1,2}^*$ and $\beta_{2,2}^*$ are high, so all subjects having $x_i \in I_{1,2}^*$ and $x_i \in I_{2,2}^*$ at the same time have a high risk for the event to occur quickly. This induces the dark red dots in the small square in the center of the figure. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

both $M_j^*$ and $\hat{M}_j$ can be empty, which explains the use of $S'$. The second metric we use is precisely focused on the sparsity aspect; it assesses the ability for each method to detect features with no cut-points, and is defined by

$$m_2 = |S'|^{-1} \sum_{j \in S} \hat{K}_j.$$

The state-of-the-art competing methods for automatic cut-points detection in a survival analysis context are based on multiple log-rank tests (“MT” for “Multiple testing”), and we consider the Bonferroni correction (see Bland and Altman (1995)), denoted “MT-B,” as well as a correction proposed in Lausen and Schumacher (1992), denoted “MT-LS.” A precise description of these methods is given in Appendix A.2 of the Supplementary material.

4.2 Simulation results

Figure 2 illustrates how the methods considered behave on the data shown in Figure 1. With the help of this example, we can clearly see the good performance of the binacox method: the position, strength, and number of cut-points are well estimated. The MT-B and MT-LS methods can only detect one cut-point by construction. Both methods detect “the most significant” cut-point for each of the two features, namely those corresponding to the highest jumps in $\beta_j^*$, (see Figure 1): $\mu_{1,1}^*$ and $\mu_{2,2}^*$.

With regards to the shape of the “$p$-value curves,” one can see that for each of the two features, the two “main” local maxima correspond to the true cut-points. One could then imagine creating a method for detecting such maxima, but this is beyond the scope of this paper (plus it would still be based on MT methods, which have high computational costs, as detailed hereafter).

Now let us look at the computing time required for the methods considered. As the multiple testing-related methods are univariate, we can directly parallelize their computations across dimensions (which is what we did in the applications), so let us consider here a single feature $X$ ($p = 1$). Following the competing methods, we have to compute all log-rank test $p$-values computed on the populations \{$y_i : x_i > \mu$\} and \{$y_i : x_i \leq \mu$\} for $i = 1, \ldots, n$, for $\mu$ taking all $x_i$ values between the 10th and 90th empirical quantiles of $X$. We denote “MT all” this method in Figure 3(a), and compare its computing times with the binacox method for various values of $n$. We also show the “MT grid” method that only computes the $p$-values for candidates $\mu_{j,l}$ used in the binacox method.

Since the number of candidates does not change with $n$ for the “MT grid” method, the computing time ratio between “MT all” and “MT grid” naturally increases, going roughly from one to two orders of magnitude higher when $n$ goes from 300 to 4000. Hence to make computations much faster, we will use the “MT grid” for all multiple testing-related methods in the following. The resulting loss of precision in the MT-related methods is negligible for a high enough $d_j$ ($= 50$ in practice).

Next, we emphasize the fact that the binacox method is still roughly five times faster than the “MT grid” method, and it remains very fast when we increase the dimension, as shown in Figure 3(b). It turns out that the computational time grows roughly logarithmically with $p$.

Let us compare now the results of simulations in terms of the $m_1$ and $m_2$ metrics introduced in Section 4.1. Figure 4 gives a comparison of the methods considered for the cut-point estimation aspect, that is, in terms of the $m_1$ score. It appears that the binacox method outperforms the MT-related methods when $K_j^* > 1$, and is competitive
FIGURE 2  Top: Illustration of the main quantities involved in the binacox method, with estimations obtained for the data represented in Figure 1. Our algorithm detects the correct number of cut-points \( \hat{K}_j = 2 \), and estimates their positions accurately, as well as their amplitudes. Bottom: results obtained using the multiple testing-related methods introduced in Appendix A.2 of the Supplementary material. Here the Benjamini–Hochberg (BH) (Benjamini and Hochberg, 1995) threshold lines overlap that corresponding to \( \alpha = 5\% \). The BH procedure would consider as cut-points all \( \mu_{j,l} \) values for which the corresponding dark green (MT) line’s values are above this, thus detecting far too many cut-points. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

FIGURE 3  Computing time for the methods considered. This figure appears in color in the electronic version of this article, and any mention of color refers to that version (A) Average computing times in seconds (with the black lines representing \( \pm \) the standard deviation) obtained on 100 simulated datasets (according to Section 4.1 with \( p = 1 \) and \( K^* = 2 \)) for training the binacox method versus the multiple testing methods, where cut-point candidates are either all \( x_i \) values between the 10th and 90th empirical quantiles of \( X \) (“MT all”), or the same candidates as the grid considered by the binacox method (“MT grid”). (B) Average (bold) computing times in seconds and standard deviation (bands) obtained on 100 simulated datasets (according to Section 4.1 with \( K^*_j = 2 \)) for training the binacox method when increasing the dimension \( p \) up to 100. The method remains very fast in high-dimensional settings when \( K^*_j = 1 \) except for small values of \( n \). This is due to an overestimation in the number of cut-points by the binacox method (see Figure 5), especially when \( p \) is high and \( n \) is small, which gives higher \( m_1 \) values, even if the “true” cut-point is actually well estimated. Note that for such values of \( p \), the binacox method runs much faster than the MT-related methods.

Figure 5, on the other hand, assesses the ability of each method to detect features with no cut-points using the \( m_2 \) metric, that is, the ability to estimate \( \hat{K}^*_j = 0 \) for \( j \in S \). The binacox method appears to be quite effective at detecting features with no cut-point when \( n \) takes a high enough value compared to \( p \), which is not the case for the MT-related methods.

5  |  APPLICATION ON GENETIC DATA

In this section, we apply our method to three biomedical data sets. We extracted normalized expression data and survival times \( Z \) in days from breast invasive carcinoma (BRCA, \( n = 1211 \)), glioblastoma multiforme (GBM,
FIGURE 4 Average (bold) $m_1$ scores and standard deviation (bands) obtained on 100 datasets simulated according to Section 4.1 with $p = 50$ and $K^*_j$ equal to 1, 2, and 3 (for all $j = 1, ..., p$) for the left, center, and right sub-figures, respectively) for varying $n$. The lower the value of $m_1$, the better the result; the binacox method clearly outperforms the other methods when there is more than one cut-point, and is competitive with other methods when there is only one cut-point, but performs worse when $n$ is small because it overestimates $K^*_j$. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

FIGURE 5 Average (bold) $m_2$ scores and standard deviation (bands) obtained on 100 datasets simulated according to Section 4.1 with $p = 50$ for varying $n$. MT-B and MT-LS tend to detect a cut-point when there is none (no matter the value of $n$), while binacox overestimates the number of cut-points for small values of $n$ but detects $S$ well for $p = 50$ on the simulated data when $n > 1000$. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

$n = 168$), and kidney renal clear cell carcinoma (KIRC, $n = 605$). These data sets are available on The Cancer Genome Atlas (TCGA) platform, which aims to accelerate the understanding of the molecular basis of cancer with the help of genomic technology, including large-scale genome sequencing. For each patient, 20,531 features corresponding to normalized gene expression values are available.

As we saw in Section 4.2, the MT-related methods are intractable in such high-dimensional cases. We therefore include a screening step to select the portion of features most relevant to our problem from the 20,531 available. The screening step is used for all competing models. To do so, we fit the binacox method on each $j$th block separately and take the resulting $\|\hat{\beta}_j\|_{TV}$ as a score that roughly assesses the propensity for feature $j$ to have one (or more) relevant cut-point(s). We then select the features corresponding to the top $P$ values with $P = 50$, this choice being suggested by the distribution of the obtained scores given in Figure 6 of Appendix A.8.1 in the Supplementary Material.

5.1 Estimation results

In Figure 6, we present the results obtained by the methods considered on the GBM cancer data set for the top 10 features ordered according to the binacox $\|\hat{\beta}_j\|_{TV}$ values. We observe that all cut-points detected by the univariate multiple testing methods MT-B or MT-LS are also detected by the multivariate binacox (which detects more cut-points); see Table 2. Furthermore, it turns out that these top 10
FIGURE 6  Illustration of the results obtained on the top 10 features ordered according to the binacox $\|\beta_j\|_{TV}$ values on the GBM dataset. The binacox method detects multiple cut-points and sheds light on nonlinear effects for various genes. The BH thresholds are shown, but are unusable in practice. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

TABLE 2  Estimated cut-point values for each method on the top 10 genes presented in Figure 6 for GBM. Dots (\cdot) mean “no cut-point detected”

| Genes     | Binacox       | MT-B   | MT-LS |
|-----------|---------------|--------|-------|
| SOD3 6649 | 200.87, 326.40, 606.48 | \cdot   | \cdot   |
| LOC 400752| 31.46, 62.50  | 19.65  | 19.65  |
| C11orf63 79864 | 40.30, 109.67 | 19.65  | 19.65  |
| KTI12 112970 | 219.60, 305.70 | 219.60 | 219.60 |
| HOXC8 3224 | 3.30, 15.75   | 3.30   | 3.30   |
| DDXS 1655 | 10,630.11, 13,094.89 | \cdot   | \cdot   |
| FKBP1L 360132 | 111.72     | \cdot   | \cdot   |
| HOXAI 3198 | 67.28        | 107.53  | 107.53  |
| MOSC2 54996 | 107.53       | 107.53  | 107.53  |
| ZNF680 340252 | 385.85, 638.06 | 385.85  | 385.85  |

5.1 Genes are relevant to GBM, the most aggressive cancer that begins in the brain. For instance, the first gene, SOD3, is relevant from a physiopathological point of view since its polymorphisms are already known as GBM risk factors (Rajaraman et al., 2008).

Relevant results were also obtained on the KIRC and BRCA data sets; these are postponed to Appendix A.8.2 in the Supplementary Material.

5.2 Risk prediction

Let us now investigate how performances are impacted in terms of risk prediction when detected cut-points are taken into account; namely, comparing predictions when training a Cox model on the original continuous feature space versus on the $\hat{\mu}$-binarized space constructed with the cut-point estimates. We randomly split the three data sets 100 times into training and validation sets (30% for testing) and compare the average C-index on the validation sets in Table 3 when the $\hat{\mu}$-binarized space is constructed based on the $\hat{\mu}$’s obtained either from the binacox method, MT-B, or MT-LS. We also compare performances obtained by two nonlinear multivariate methods known to perform well in high-dimensional settings: boosted Cox (CoxBoost) (Li and Luan, 2005) and random survival forests (RSF) (Ishwaran et al., 2008).

The binacox method clearly improves risk prediction compared to classical Cox, as well as with respect to the MT-B and MT-LS methods. Moreover, it also outperforms both CoxBoost and RSF. To the best of our knowledge, no better performances have been achieved on this data in the literature (Yousefi et al., 2017). See also Appendix A.8 of the Supplementary Material for additional details and results on computing times.

6 DISCUSSION

In this paper, we introduced the binacox method, designed for estimating multiple cut-points in a Cox model with high-dimensional features. We illustrated the good theoretical properties of the model by establishing nonasymptotic oracle inequalities for prediction and estimation. An extensive Monte Carlo simulation study was then carried out to evaluate the method’s performance. It showed that our approach outperforms existing methods, with computing times orders of magnitude faster. Moreover, in addition to the raw feature selection ability of the binacox
method, it succeeds in detecting multiple cut-points per feature. We also applied binacox to three publicly available high-dimensional genetics data sets. Furthermore, several genes pinpointed by the model turn out to be biologically relevant, while others require further investigation in the genetics research community. More importantly, our method provides powerful and innovative interpretation aspects that could be useful in both clinical research and daily practice. Indeed, the estimated cut-points could be directly considered in clinical practice. Thus, the method could be an interesting alternative to more classical methods found in the medical literature to deal with prognosis studies in high-dimensional frameworks, providing a new way to model nonlinear feature associations, and giving rise to new data-driven risk scores. Our study lays the groundwork for the development of powerful methods that could one day help provide improved personalized care.

ACKNOWLEDGMENTS
Mokhtar Z. Alaya is grateful for a grant from DIM Math Innov Région Ile-de-France http://www.dim-mathinnov.fr. Agathe Guilloux’s work has been supported by the INCA-DGOS grant PTR-K 2014. The authors thank professor J.P. Vert for his support and the interesting discussions about selection bias.

CONFLICT OF INTEREST
The authors have no relevant conflicts of interest to disclose.

OPEN RESEARCH BADGES
This article has earned an Open Materials badge for making publicly available the components of the research methodology needed to reproduce the reported procedure and analysis. All materials are available at https://github.com/jjfeng/aACP.

DATA AVAILABILITY STATEMENT
The results shown in this paper are based on data generated by the Cancer Genome Atlas (TCGA) Research Network and freely available at http://cancergenome.nih.gov.

ORCID
Simon Bussy https://orcid.org/0000-0002-6059-0685

REFERENCES
Alaya, M.Z., Bussy, S., Gaïffas, S., & Guilloux, A. (2019) Binarsity: a penalization for one-hot encoded features in linear supervised learning. Journal of Machine Learning Research, 20, 1–34.
Andersen, P.K., Borgan, Ø., Gill, R.D., & Keiding, N. (2012) Statistical Models Based on Counting Processes. Amsterdam: Springer Science & Business Media.
Bach, F. (2010) Self-concordant analysis for logistic regression. Electronic Journal of Statistics, 4, 384–414.
Bacry, E., Bompaire, M., Deegan, P., Gaïffas, S., & Poulsen, S.V. (2017) Tick: a python library for statistical learning, with an emphasis on Hawkes processes and time-dependent models. Journal of Machine Learning Research, 18, 7937–7941.
Bédat, B., Niclaus, N., Jannot, A.-S., Andres, A., Tosso, C., Morel, P., et al. (2015) Impact of recipient body mass index on short-term and long-term survival of pancreatic grafts. Transplantation, 99, 94–99.
Benjamini, Y. & Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 57, 289–300.
Bland, J.M. & Altman, D.G. (1995) Multiple significance tests: the Bonferroni method. BMJ, 310, 170.
Bleakley, K. & Vert, J.P. (2011) The group fused Lasso for multiple change-point detection. Technical Report HAL-00602121.
Chang, C., Hsieh, M.-K., Chiang, A.J., Tsai, Y.-H., Liu, C.-C., & Chen, J. (2019) Methods for estimating the optimal number and location of cut points in multivariate survival analysis: a statistical solution to the controversial effect of BMI. Computational Statistics, 34, 1649–1674.
Cheang, M.C.U., Chia, S.K., Voduc, D., Gao, D., Leung, S., Snider, J. et al. (2009) Ki67 index, HER2 status, and prognosis of patients with luminal B breast cancer. JNCI: Journal of the National Cancer Institute, 101, 736–750.
Cho, H. & Fryzlewicz, P. (2015) Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 77, 475–507.
Contal, C. & O’Quigley, J. (1999) An application of changepoint methods in studying the effect of age on survival in breast cancer. Computational Statistics & Data Analysis, 30, 253–270.
Cox, D.R. (1972) Regression models and life-tables. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 34, 187–220.
Curley, A., Stanworth, S.J., Willoughby, K., Fustolo-Gunnink, S.F., Venkatesh, V., Hudson, C., et al. (2019) Randomized trial of

TABLE 3  Comparison of average C-indexes (and standard deviation in parentheses) on 100 random train/test splits for the Cox model trained on continuous features versus on its binarized version constructed using the considered methods’ cut-point estimates, and the CoxBoost and RSF methods. On the three data sets, the binacox method gives the best results (in bold)

| Cancer | Continuous | Binacox | MT-B | MT-LS | CoxBoost | RSF |
|--------|------------|---------|------|-------|----------|-----|
| GBM    | 0.567 (0.042) | **0.602 (0.050)** | 0.574 (0.052) | 0.573 (0.048) | 0.574 (0.043) | 0.568 (0.045) |
| KIRC   | 0.669 (0.032) | **0.702 (0.031)** | 0.672 (0.034) | 0.672 (0.034) | 0.679 (0.030) | 0.699 (0.034) |
| BRCA   | 0.586 (0.052) | **0.671 (0.044)** | 0.630 (0.058) | 0.626 (0.053) | 0.592 (0.054) | 0.654 (0.045) |
platelet-transfusion thresholds in neonates. New England Journal of Medicine, 380, 242–251.

Faraggi, D. & Simon, R. (1996) A simulation study of cross-validation for selecting an optimal cutpoint in univariate survival analysis. Statistics in Medicine, 15, 2203–2213.

Harchaoui, Z. & Lévy-Leduc, C. (2010) Multiple change-point estimation with a total variation penalty. Journal of the American Statistical Association, 105, 1480–1493.

Huang, J., Sun, T., Ying, Z., Yu, Y. & Zhang, C.H. (2013) Oracle inequalities for the lasso in the Cox model. The Annals of Statistics, 41, 1142–1165.

Icuma, T.R., Achcar, J.A., Martinez, E.Z. & Davarzani, N. (2018) Determination of optimum medical cut points for continuous covariates in lifetime regression models. Model Assisted Statistics and Applications, 13, 141–159.

Ishwaran, H., Kogalur, U.B., Blackstone, E.H. & Lauer, M.S. (2008) Random survival forests. The Annals of Applied Statistics, 2, 841–860.

Ivanoff, S., Picard, F. & Rivoirard, V. (2016) Adaptive lasso and grouped lasso for functional poisson regression. Journal of Machine Learning Research, 17, 1903–1948.

Klein, J.P. & Moeschberger, M.L. (2005) Survival analysis: techniques for censored and truncated data. Amsterdam: Springer Science & Business Media.

Klein, J.P. & Wu, J. (2003) Discretizing a continuous covariate in survival studies. Handbook of Statistics, 23, 27–42.

Kong, S. & Nan, B. (2014) Non-asymptotic oracle inequalities for the high-dimensional Cox regression via Lasso. Statistica Sinica, 24, 25.

Lausen, B. & Schumacher, M. (1992) Maximally selected rank statistics. Biometrics, 48, 73–85.

Leblanc, M. & Crowley, J. (1993) Survival trees by goodness of split. Journal of the American Statistical Association, 88, 457–467.

Li, H. & Luan, Y. (2005) Boosting proportional hazards models using smoothing splines, with applications to high-dimensional microarray data. Bioinformatics, 21, 2403–2409.

Meier, L., Van de Geer, S. & Bühlmann, P. (2009) High-dimensional additive modeling. The Annals of Statistics, 37, 3779–3821.

Motzer, R.J., Mazumdar, M., Bacc, J., Berg, W., Amsterdam, A. & Ferrara, J. (1999) Survival and prognostic stratification of 670 patients with advanced renal cell carcinoma. Journal of Clinical Oncology, 17, 2530–2540.

Mukherjee, B.N. & Maiti, S.S. (1988) On some properties of positive definite Toeplitz matrices and their possible applications. Linear Algebra and its Applications, 102, 211–240.

Rajaraman, P., Hutchinson, A., Rothman, N., Black, P.M., Fine, H.A., Loeffler, J.S. et al. (2008) Oxidative response gene polymorphisms and risk of adult brain tumors. Neuro-Oncology, 10, 709–715.

Rota, M., Antolini, L. & Valsecchi, M.G. (2015) Optimal cut-point definition in biomarkers: the case of censored failure time outcome. BMC Medical Research Methodology, 15, 24.

Senoussi, R. (1990) Problème d’identification dans le modèle de Cox. Annales de l’Institut Henri Poincaré Probabilités et Statistiques, 26, 45–64.

Simon, N., Friedman, J., Hastie, T. & Tibshirani, R. (2011) Regularization paths for Cox’s proportional hazards model via coordinate descent. Journal of Statistical Software, 39, 1–13.

Van de Geer, S. & Bühlmann, P. (2009) On the conditions used to prove oracle results for the lasso. Electronic Journal of Statistics, 3, 1360–1392.

Van de Geer, S.A. (2008) High-dimensional generalized linear models and the lasso. The Annals of Statistics, 36, 614–645.

Wells, P.S., Anderson, D.R., Rodger, M., Ginsberg, J.S., Kearon, C., Gent, M. et al. (2000) Derivation of a simple clinical model to categorize patients probability of pulmonary embolism: increasing the models utility with the SimpliRED D-dimer. Thrombosis and Haemostasis, 83, 416–420.

Wu, J. & Coggeshall, S. (2012) Foundations of predictive analytics, 1st edition. Boca Raton, FL: CRC Press.

Yousefi, S., Amrollahi, F., Amgad, M., Dong, C., Lewis, J.E., Song, C. et al. (2017) Predicting clinical outcomes from large scale cancer genomic profiles with deep survival models. Scientific Reports, 7, 11707.

SUPPORTING INFORMATION

Web Appendices referenced in Section 2–5 are available with this paper at the Biometrics website on Wiley Online Library. The includes details on the algorithm implementation, additional results on genetic data, and the proofs of the theoretical results. Moreover, all methodology discussed in the paper is implemented in Python/C++ and R and also available online at the Biometrics website, with the code that generates all figures (also open-sourced at https://github.com/SimonBussy/binacox) in the form of annotated programs, together with notebook tutorials. The binacox method is implemented in the tick library (Bacry et al., 2017).

How to cite this article: Bussy, S., Alaya, M.Z., Jannot, A.-S., Guilloux, A. Binacox: automatic cut-point detection in high-dimensional Cox model with applications in genetics. Biometrics. 2022;78:1414–1426.
https://doi.org/10.1111/biom.13547