C-mix: a high dimensional mixture model for censored durations, with applications to genetic data

Simon Bussy  
Theoretical and Applied Statistics Laboratory  
Pierre and Marie Curie University, Paris, France  
email: simon.bussy@gmail.com

Agathe Guilloux  
LaMME, UEVE and UMR 8071  
Paris Saclay University, Evry, France  
email: agathe.guilloux@math.cnrs.fr

Stéphane Gaïffas  
Center for Applied Mathematics  
Ecole polytechnique, Palaiseau, France  
email: stephane.gaiffas@polytechnique.edu

Anne-Sophie Jannot  
Biomedical Informatics and Public Health Department  
European Georges Pompidou Hospital, Assistance Publique-Hôpitaux de Paris and INSERM UMRS 1138, Centre de Recherche des Cordeliers, Paris, France  
email: annesophie.jannot@aphp.fr

Abstract

We introduce a supervised learning mixture model for censored durations (C-mix) to simultaneously detect subgroups of patients with different prognosis and order them based on their risk. Our method is applicable in a high-dimensional setting where datasets contain a large number of biomedical covariates. To address this difficulty, we penalize the negative log-likelihood by the Elastic-Net, which leads to a sparse parameterization of the model and automatically pinpoints the relevant covariates for the survival prediction. Inference is achieved using an efficient Quasi-Newton Expectation Maximization (QNEM) algorithm, for which we provide convergence properties. The statistical performance of the method is examined on an extensive Monte Carlo simulation study, and finally illustrated on three publicly available genetic cancer datasets with high-dimensional covariates. We show that our approach outperforms the state-of-the-art survival models in this context, namely both the CURE and Cox proportional hazards models, in terms of C-index, AUC(t) and survival prediction. This has the potential to be a powerful tool for personalized medicine in cancerology.

Keywords. Cox proportional hazards model; CURE model; Elastic-net regularization; High-dimensional estimation; Mixture duration model; Survival analysis
1 Introduction

Predicting subgroups of patients with different prognosis is a key challenge for personalized medicine, see for instance Alizadeh et al. [2000], Rosenwald et al. [2002] where subgroups of patients with different survival rates are identified based on gene expression data. A substantial number of techniques can be found in the literature to predict the subgroup of a given patient in a classification setting, namely when subgroups are known in advance [Golub et al., 1999, Hastie et al., 2001, Tibshirani et al., 2002]. We consider the much more difficult case where subgroups are unknown.

To tackle this problem, a first widespread approach consists in using unsupervised learning techniques applied on the covariates, namely on the gene expression data for instance [Bhattacharjee et al., 2001, Beer et al., 2002, Sørlie et al., 2001]. The problem of such techniques is that there is no guarantee that the identified subgroups will be associated with the survival times: there is no reason that this method will lead to subgroups with different survival rates.

Another approach to identifying subgroups is conversely based exclusively on the survival times: patients are then assigned to a low-risk or a high-risk group based on whether they were still alive [Shipp et al., 2002, Van’t Veer et al., 2002]. The problem here is that the resulting subgroups may not be biologically meaningful since the method do not use the covariates, and prognosis prediction based on covariates is not possible.

The method we propose uses both the survival information of the patients and its covariates in a supervised learning way. Moreover, it relies on the idea that exploiting the subgroups structure of the data, say the fact that a portion of the population have a higher risk of early death, could improve the survival prediction of future patients (unseen during the learning phase).

To be more specific, based on the time $T \geq 0$ elapsed before the death, and the covariates $X \in \mathbb{R}^d$ corresponding to biomedical data (e.g. gene expression data, patients characteristics, therapeutic strategy or omics covariates), we seek to construct a score that measures, for each patient, the risk of an early death. Let us precise that other adverse event of interest such as rehospitalization, relapse or disease progression can be considered in the same statistical framework, not only the death.

In such setups, the resulting score can help decision-making for physicians (e.g., can a patient leave the hospital with low risk of relapse). As a byproduct, it can also shed light on the effect of the covariates (which combination of biomedical markers are relevant to a given event of interest).

Our methodology differs from the standard survival analysis approaches. The Cox proportional hazards (PH) model is by far the most widely used in such a setting, see Cox [1972] and Simon et al. [2011] for the penalized version. It is a regression model that describes the relation between intensity of events and covariates, given by

$$\lambda(t) = \lambda_0(t)\exp(x^T \beta_{\text{cox}}),$$  \hspace{1cm} (1)
impact on the hazard ratio of each covariate. Our proposed alternative to the Cox PH model directly fulfills the task of attributing a score to each patient, namely the probability of being part of the group at low risk of early death, thus delivering a simple tool for clinical practice. Moreover, the C-mix model can be trained very efficiently in high dimension, and outperforms the standard Cox PH model by far in the analysed datasets.

On the other hand, the CURE model (see Farewell [1982] and Kuk and Chen [1992]) considers one fraction of the population as cured or not subject to any risk of death. In this paper, we consider that there is always an event risk, no matter how small. Then, unlike in the CURE model, both sub-populations are at risk. Indeed, for a large number of applications (e.g. rehospitalization for patients with chronic diseases or relapse for patients with metastatic cancer), one can consider that all patients are at risk.

Mixture models have been considered in survival analysis, see Kuo and Peng [2000] for a general study about mixture model for survival data or De Angelis et al. [1999] in a cancer survival analysis setting, to name but a few. But, again, none of these papers consider the high dimensional setting.

Going back to the construction of a score, two difficulties arise. First, the biomedical data is high-dimensional. Second, the time $T$ is censored. Indeed in practice, information loss occurs of right censoring type, that is for patients still alive when the study ends. This is taken into acount by introducing the following: a time $C \geq 0$ when the individual “leaves” the target cohort, a right-censored duration $Y$ and a censoring indicator $\Delta$, defined by

$$Y = \min(T, C) \quad \text{and} \quad \Delta = 1_{\{T \leq C\}},$$

where $\min(a, b)$ denotes the minimum between two numbers $a$ and $b$, and $1$ denotes the indicator function.

We then introduce a binary latent variable $Z \in \{0, 1\}$ where $Z = 1$ means high-risk of early death and $Z = 0$ means low risk. Our focus is the conditional distribution of $Z$ given $X$, namely the conditional probability $\pi(x) = P[Z = 0|X = x]$ that the patient belongs to the group with low risk of death, given its covariates $x$. Now, suppose that, conditionally on the latent variable $Z$, the distribution of duration time $T$ is different, leading to a mixture in the distribution of the duration before the death. The advantage of such a latent class model is that it can handle heterogeneity within the patient population.

A precise description of the model is given in Section 2. Section 3 focuses on the regularized version of the model with an Elastic-Net penalization to exploit dimension reduction and prevent overfitting. Inference is presented under this framework, as well as the convergence properties of the developed algorithm. Section 4 highlights the simulation procedure used to evaluate the performances and compares it with state-of-the-art models. In Section 5, we apply our method to genetic datasets. Finally, we discuss the obtained results in Section 6.
2 A censored mixture model

Let us present the survival analysis framework. We assume that, the conditional distribution of the duration $T$ given $X = x$ is a mixture

$$f(t|X = x) = \pi(x) f_0(t; \alpha_0) + (1 - \pi(x)) f_1(t; \alpha_1)$$

of two densities $f_1$ and $f_0$, for $t \geq 0$ and $(\alpha_0, \alpha_1) \in \mathbb{R}^{d_0} \times \mathbb{R}^{d_1}$ some parameters to estimate. The weights combining these distributions depend on the patient biomedical covariates $x$. This is equivalent to saying that conditionally on a latent variable $Z = z \in \{0, 1\}$, the density of $T$ at time $t \geq 0$ is $f_z(t; \alpha_z)$, and we have

$$\mathbb{P}[Z = 0|X = x] = \pi(x) \quad \text{and} \quad \mathbb{P}[Z = 1|X = x] = 1 - \pi(x).$$

We consider a logistic link function for these weights, given by

$$\pi(x) = \pi_\beta(x) = \frac{1}{1 + e^{-x^\top \beta}},$$

where $\beta \in \mathbb{R}^d$ is a vector of coefficients that quantifies the impact of each biomedical covariates on the probability that a patient belongs to the low-risk population. The hidden status $Z$ has therefore a Bernoulli distribution $\mathcal{B}(1 - \pi_\beta(x))$. The intercept term is here omitted without loss of generality.

In order to write a likelihood and draw inference, we make the following hypothesis, which is classical in survival analysis [Klein and Moeschberger, 2005].

**Hypothesis 1** $T$ and $C$ are conditionally independent given $Z$ and $X$, and $C$ is independent of $Z$ and $X$.

Under this hypothesis, denoting $g$ the density of the censoring $C$, $F$ the cumulative distribution function corresponding to a given density $f$, $\overline{F} = 1 - F$ and $F(y^-) = \lim_{u \rightarrow y \wedge u} F(u)$, we have

$$\mathbb{P}[Y \leq y, \Delta = 1] = \mathbb{P}[T \leq y, T \leq C] = \int_0^y f(u) \overline{G}(u) du \quad \text{and}$$

$$\mathbb{P}[Y \leq y, \Delta = 0] = \mathbb{P}[C \leq y, C \leq T] = \int_0^y g(u) \overline{F}(u) du.$$

Then, denoting $\theta = (\alpha_0, \alpha_1, \beta)^\top$ the parameters to infer and considering an independent and identically distributed (i.i.d.) cohort of $n$ patients $(x_1, y_1, \delta_1), \ldots, (x_n, y_n, \delta_n) \in \mathbb{R}^d \times \mathbb{R}_+ \times \{0, 1\}$, the log-likelihood of the C-mix model can be written

$$\ell_n(\theta) = \ell_n(\theta ; y, \delta) = n^{-1} \sum_{i=1}^n \delta_i \log \left[ \{ \pi_\beta(x_i) f_0(y_i; \alpha_0) + (1 - \pi_\beta(x_i)) f_1(y_i; \alpha_1) \} \overline{G}(y_i^-) \right]$$

$$+ (1 - \delta_i) \log \left[ \{ \pi_\beta(x_i) \overline{F}_0(y_i^-; \alpha_0) + (1 - \pi_\beta(x_i)) \overline{F}_1(y_i^-; \alpha_1) \} g(y_i) \right],$$

where we use the notations $y = (y_1, \ldots, y_n)^\top$ and $\delta = (\delta_1, \ldots, \delta_n)^\top$. Note that from now on, all computations are done conditionally on the covariates $(x_i)_{i=1,\ldots,n}$. An important fact is that we do not need to know or parametrize $\overline{G}$ nor $g$, namely the distribution of the censoring, for inference in this model (since all $\overline{G}$ and $g$ terms vanish in Equation (4)).
3 Inference of C-mix

In this section, we describe the procedure for estimating the parameters of the C-mix model. We begin by presenting the Quasi-Newton Expectation Maximization (QNEM) algorithm we use for inference. We then focus our study on the convergence properties of the algorithm.

3.1 QNEM algorithm

In order to avoid overfitting and to improve the prediction power of our model, we use Elastic-Net regularization [Zou and Hastie, 2005] by minimizing the penalized objective

$$\ell_{\text{pen}}(\theta) = -\ell_n(\theta) + \gamma((1 - \eta)\|\beta\|_1 + \frac{\eta}{2}\|\beta\|_2^2),$$

where we add a linear combination of the lasso ($\ell_1$) and ridge (squared $\ell_2$) penalties for a fixed $\eta \in [0, 1]$, tuning parameter $\gamma$, and where we denote $\|\beta\|_p = \left(\sum_{i=1}^d |\beta_i|^p\right)^{1/p}$ the $\ell_p$-norm of $\beta$. One advantage of this regularization method is its ability to perform model selection (the lasso part) and pinpoint the most important covariates relative to the prediction objective. On the other hand, the ridge part allows to handle potential correlation between covariates [Zou and Hastie, 2005]. Note that in practice, the intercept is not regularized.

In order to derive an algorithm for this objective, we introduce a so-called Quasi-Newton Expectation Maximization (QNEM), being a combination between an EM algorithm [Dempster et al., 1977] and a L-BFGS-B algorithm [Zhu et al., 1997]. For the EM part, we need to compute the negative completed log-likelihood (here scaled by $n^{-1}$), namely the negative joint distribution of $y, \delta$ and $z = (z_1, \ldots, z_n)^\top$. It can be written

$$\ell_{\text{comp}}(\theta) = \ell_{\text{comp}}(\theta; y, \delta, z)$$

$$= -n^{-1}\sum_{i=1}^n \delta_i \left\{ z_i \left[ \log \left(1 - \pi\beta(x_i)\right) + \log \left(f_1(y_i; \alpha_1)\right) \right] 
+ (1 - z_i) \left[ \log \left(\pi\beta(x_i)\right) + \log \left(f_0(y_i; \alpha_0)\right) \right] + \log \left(\bar{G}(y^-)\right) \right\} 
+ (1 - \delta_i) \left\{ z_i \left[ \log \left(1 - \pi\beta(x_i)\right) + \log \left(\bar{F}_1(y^-; \alpha_1)\right) \right] 
+ (1 - z_i) \left[ \log \left(\pi\beta(x_i)\right) + \log \left(\bar{F}_0(y^-; \alpha_0)\right) \right] + \log \left(g(y_i)\right) \right\}. $$

Suppose that we are at step $l + 1$ of the algorithm, with current iterate $\theta^{(l)} = (\alpha_0^{(l)}, \alpha_1^{(l)}, \beta^{(l)})^\top$. For the E-step, we need to compute the expected log-likelihood given by

$$Q_n(\theta, \theta^{(l)}) = \mathbb{E}_{\theta^{(l)}}[\ell_{\text{comp}}(\theta)|y, \delta].$$

We note that

$$q_i^{(l)} = \mathbb{E}_{\theta^{(l)}}[z_i|y_i, \delta_i] = \mathbb{P}_{\theta^{(l)}}[z_i = 1|y_i, \delta_i] = \frac{\Lambda_{1,i}^{(l)}}{\Lambda_{0,i}^{(l)} + \Lambda_{1,i}^{(l)}}$$

(4)
with
\[
\Lambda^{(l)}_{1,i} = (f_1(y_i; \alpha^{(l)}_1)\widetilde{G}(y_i^-))^{\delta_i}(g(y_i)\widetilde{F}_1(y_i^-; \alpha^{(l)}_1))^{1-\delta_i}(1 - \pi_{\beta^{(l)}}(x_i)) \quad \text{and} \quad (5)
\]
\[
\Lambda^{(l)}_{0,i} = (f_0(y_i; \alpha^{(l)}_0)\widetilde{G}(y_i^-))^{\delta_i}(g(y_i)\widetilde{F}_0(y_i^-; \alpha^{(l)}_0))^{1-\delta_i}\pi_{\beta^{(l)}}(x_i), \quad (6)
\]
so that
\[
Q_n(\theta, \theta^{(l)}) = -n^{-1} \sum_{i=1}^{n} \{ (1 - q^{(l)}_i) \log \pi_{\beta}(x_i) + q^{(l)}_i \log (1 - \pi_{\beta}(x_i)) \}
\]
\[
+ (1 - q^{(l)}_i)\{ \log (\pi_{\beta}(x_i)) + \log (f_0(y_i; \alpha_0)) \} + \log(\widetilde{G}(y_i^-)) \}
\]
\[
+ (1 - \delta_i)\{ q^{(l)}_i \log (1 - \pi_{\beta}(x_i)) + \log (\widetilde{F}_1(y_i^-; \alpha_1)) \}
\]
\[
+ (1 - q^{(l)}_i)\{ \log (\pi_{\beta}(x_i)) + \log (\widetilde{F}_0(y_i^-; \alpha_0)) \} + \log (g(y_i)) \}.
\]
Depending on the chosen distributions \( f_0 \) and \( f_1 \), the M-step can either be explicit for the updates of \( \alpha_0 \) and \( \alpha_1 \) (see Section 3.3 below for the geometric distributions case), or obtained using a minimization algorithm otherwise.

Let us focus now on the update of \( \beta \) in the M-step of the algorithm. We use the following basic manipulations on the quantities involved in \( Q_n \) that depend on \( \beta \)
\[
- n^{-1} \sum_{i=1}^{n} \{ (1 - q^{(l)}_i) \log \pi_{\beta}(x_i) + q^{(l)}_i \log (1 - \pi_{\beta}(x_i)) \}
\]
\[
= -n^{-1} \sum_{i=1}^{n} \{ q^{(l)}_i \log \left( \frac{1 - \pi_{\beta}(x_i)}{\pi_{\beta}(x_i)} \right) \} + \log \pi_{\beta}(x_i) \}
\]
\[
= n^{-1} \sum_{i=1}^{n} \{ q^{(l)}_i x_i^T \beta + \log(1 + e^{-x_i^T \beta}) \},
\]
which uses the fact that, by construction of the logistic link function, the log odd-ratio is linear. The update for \( \beta \) therefore requires to minimize
\[
R^{(l)}_n(\beta) + \gamma((1 - \eta)\|\beta\|_1 + \frac{\eta}{2}\|\beta\|_2^2), \quad (7)
\]
where we introduced
\[
R^{(l)}_n(\beta) = n^{-1} \sum_{i=1}^{n} \{ q^{(l)}_i x_i^T \beta + \log(1 + e^{-x_i^T \beta}) \}.
\]
The minimization Problem (7) is a convex problem. It looks like the logistic regression objective, where labels are not fixed but softly encoded by the expectation step (computation of \( q^{(l)}_i \) above, see Equation (4)).

We minimize (7) using the well-known L-BFGS-B algorithm [Zhu et al., 1997]. This algorithm belongs to the class of quasi-Newton optimization routines, which solve the given minimization problem by computing approximations of the inverse Hessian matrix of the objective function. It can deal with differentiable convex objectives with box constraints. In order to use it with \( \ell_1 \) penalization, which is not differentiable, we use the trick borrowed from Andrew and Gao [2007]: for
Let \( a \in \mathbb{R} \), write \( |a| = a_+ + a_- \), where \( a_+ \) and \( a_- \) are respectively the positive and negative part of \( a \), and add the constraints \( a_+ \geq 0 \) and \( a_- \geq 0 \). Namely, we rewrite the minimization problem (7) as the following differentiable problem with box constraints

\[
\minimize R_n^{(l)}(\beta^+ - \beta^-) + \gamma(1 - \eta) \sum_{j=1}^{d} (\beta_j^+ + \beta_j^-) + \gamma \frac{\eta}{2} \|\beta^+ - \beta^-\|_2^2
\]

subject to \( \beta_j^+ \geq 0 \) and \( \beta_j^- \geq 0 \) for all \( j \in \{1, \ldots, d\} \).

where \( \beta^\pm = (\beta_1^\pm, \ldots, \beta_d^\pm)^T \). The L-BFGS-B solver requires the exact value of the gradient, which is easily given by

\[
\frac{\partial R_n^{(l)}(\beta)}{\partial \beta} = n^{-1} \sum_{i=1}^{n} \left( q_i^{(l)} - \frac{1}{1 + e^{x_i^\top \beta}} \right) x_i.
\]

In Algorithm 1, we describe the main steps of the QNEM algorithm to minimize the function given in Equation (3).

### Algorithm 1: QNEM Algorithm for inference of the C-mix model

**Require:** Training data \((x_i, y_i, \delta_i)_{i \in \{1, \ldots, n\}}\); starting parameters \( \alpha_0^{(0)}, \alpha_1^{(0)}, \beta^{(0)} \); tuning parameter \( \gamma \geq 0 \).

1. **for** \( l = 0, \ldots \), until convergence **do**
   2. Compute \( q_i^{(l)} \) using Equation (4).
   3. Compute \( \alpha_0^{(l+1)} \) and \( \alpha_1^{(l+1)} \).
   4. Compute \( \beta^{(l+1)} \) by solving (8) with the L-BFGS-B algorithm.
5. **end for**
6. **return** Last parameters \( \alpha_0^{(l)}, \alpha_1^{(l)}, \beta^{(l)} \).

The penalization parameter \( \gamma \) is chosen using cross-validation, see Appendix A for precise statements about this procedure and about other numerical details.

### 3.2 Convergence to a stationary point

We are addressing here convergence properties of the QNEM algorithm described in Section 3.1 for the minimization of the objective function defined in Equation (3). Let us denote

\[
Q_n^{pen}(\theta, \theta^{(l)}) = Q_n(\theta, \theta^{(l)}) + \gamma \left( (1 - \eta) \|\beta\|_1 + \frac{\eta}{2} \|\beta\|_2^2 \right).
\]

Convergence properties of the EM algorithm in a general setting are well known, see Wu [1983]. In the QNEM algorithm, since we only improve \( Q_n^{pen}(\theta, \theta^{(l)}) \) instead of a minimization of \( Q_n(\theta, \theta^{(l)}) \), we are not in the EM algorithm setting but in a so-called generalized EM (GEM) algorithm setting [Dempster et al., 1977]. For such an algorithm, we do have the descent property, in the sense that the criterion function given in Equation (3) is reduced at each iteration, namely

\[
\ell_n^{pen}(\theta^{(l+1)}) \leq \ell_n^{pen}(\theta^{(l)}).
\]

Let us make two hypothesis.
Hypothesis 2 The two duration densities $f_0$ and $f_1$ are such that $\ell^{pen}_n$ is bounded for all $\theta$.

Hypothesis 3 $Q^{pen}_n(\theta, \theta(i))$ is continuous in $\theta$ and $\theta(i)$, and for any fixed $\theta(i)$, $Q^{pen}_n(\theta, \theta(i))$ is a convex function in $\theta$ and is strictly convex in each coordinate of $\theta$.

Under Hypothesis 2, $l \mapsto \ell^{pen}_n(\theta(i))$ decreases monotonically to some finite limit. By adding Hypothesis 3, convergence of the QNEM algorithm to a stationary point can be shown. In particular, the stationary point is here a local minimum.

Theorem 1 Under Hypothesis 2 and 3, and considering the QNEM algorithm for the criterion function defined in Equation (3), every cluster point $\bar{\theta}$ of the sequence $\{\theta(i); l = 0, 1, 2, \ldots\}$ generated by the QNEM algorithm is a stationary point of the criterion function defined in Equation (3).

A proof is given in Appendix B.

3.3 Parameterization

In practice, we deal with discrete times in days. It turns out that the times of the data used for applications in Section 5 is well fitted by Weibull distributions. This choice of distribution is very popular in survival analysis, see for instance Klein and Moeschberger [2005]. We then first derive the QNEM algorithm with

$$f_z(t; \alpha_z) = (1 - k_z)t^{\mu_z} - (1 - k_z)(t+1)^{\mu_z}$$

with here $\alpha_z = (k_z, \mu_z) \in (0, 1) \times \mathbb{R}_+$, $k_z$ being the scale parameter and $\mu_z$ the shape parameter of the distribution.

As explained in the following Section 4, we select the best model using a cross-validation procedure based on the C-index metric, and the performances are evaluated according to both C-index and AUC($t$) metrics (see Sections 4.3 for details). Those two metrics have the following property: if we apply any mapping on the marker vector (predicted on a test set) such that the order between all vector coefficient values is conserved, then both C-index and AUC($t$) estimates remain unchanged. In other words, by denoting $(M_i)_{i \in \{1, \ldots, n_{test}\}}$ the vector of markers predicted on a test set of $n_{test}$ individuals, if $\Phi$ is a function such that for all $(i, j) \in \{1, \ldots, n_{test}\}^2$, $(M_i < M_j \Rightarrow \Phi(M_i) < \Phi(M_j))$, then both C-index and AUC($t$) estimates induced by $(M_i)_{i \in \{1, \ldots, n_{test}\}}$ or by $(\Phi(M_i))_{i \in \{1, \ldots, n_{test}\}}$ are the same.

The order in the marker coefficients is actually paramount when the performances are evaluated according to the mentioned metrics. Furthermore, it turns out that empirically, if we add a constraint on the mixture of Weibull that enforces an order like relation between the two distributions $f_0$ and $f_1$, the performances are improved. To be more precise, the constraint to impose is that the two density curves do not intersect. We then choose to impose the following: the two scale parameters are equal, i.e. $k_0 = k_1 = k$. Indeed under this hypothesis, we do have that for all $k \in (0, 1)$, $(\mu_0 < \mu_1 \Rightarrow \forall t \in \mathbb{R}_+, f_0(t; \alpha_0) > f_1(t; \alpha_1))$.

With this Weibull parameterization, updates for $\alpha_z$ are not explicit in the QNEM algorithm, and consequently require some iterations of a minimization algorithm.
Seeking to have explicit updates for $\alpha_z$, we then derive the algorithm with geometric distributions instead of Weibull (geometric being a particular case of Weibull with $\mu_z = 1$), namely $f_z(t; \alpha_z) = \alpha_z (1 - \alpha_z)^{t-1}$ with $\alpha_z \in (0, 1)$.

With this parameterization, Equations (5) and (6) write respectively

$$
\Lambda^{(l)}_{1,i} = (\alpha_1^{(l)}(1 - \alpha_1^{(l)})^{y_i-1})^{\delta_i} ((1 - \alpha_1^{(l)})^{y_i})^{1 - \delta_i} (1 - \pi_{\beta^{(l)}}(x_i)) \quad \text{and} \quad
\Lambda^{(l)}_{0,i} = (\alpha_0^{(l)}(1 - \alpha_0^{(l)})^{y_i-1})^{\delta_i} ((1 - \alpha_0^{(l)})^{y_i})^{1 - \delta_i} \pi_{\beta^{(l)}}(x_i),
$$

which leads to the following explicit M-step

$$
\alpha_0^{(l+1)} = \frac{\sum_{i=1}^{n} \delta_i (1 - q_i^{(l)})}{\sum_{i=1}^{n} (1 - q_i^{(l)}) y_i} \quad \text{and} \quad \alpha_1^{(l+1)} \frac{\sum_{i=1}^{n} \delta_i q_i^{(l)}}{\sum_{i=1}^{n} q_i^{(l)} y_i}.
$$

In this setting, implementation is hence straightforward. Note that Hypothesis 2 and 3 are immediately satisfied with this geometric parameterization.

In Section 5, we note that performances are similar for the C-mix model with Weibull or geometric distributions on all considered biomedical datasets. The geometric parameterization leading to more straightforward computations, it is the one used to parameterize the C-mix model in what follows, if not otherwise stated. Let us focus now on the performance evaluation of the C-mix model and its comparison with the Cox PH and CURE models.

## 4 Performance evaluation

In this section, we first briefly introduce the models we consider for performance comparisons. Then, we provide details regarding the simulation study and data generation. The chosen metrics for evaluating performances are then presented, followed by the results.

### 4.1 Competing models

The first model we consider is the well known Cox PH Model. In this model introduced in Cox [1972], the parameter $\beta^{\text{cox}}$ is estimated by minimizing the partial log-likelihood given by

$$
\ell_n^{\text{cox}}(\beta) = n^{-1} \sum_{i=1}^{n} \delta_i (x_i^\top \beta - \log \sum_{i': y_{i'} \geq y_i} \exp(x_{i'}^\top \beta)).
$$

We use respectively the R packages `survival` and `glmnet` [Simon et al., 2011] for the partial log-likelihood and the minimization of the following quantity

$$
-\ell_n^{\text{cox}}(\beta) + \gamma ((1 - \eta) \| \beta \|_1 + \frac{\eta}{2} \| \beta \|^2_2),
$$

where $\gamma$ is chosen by the same cross-validation procedure than the C-mix model, for a given $\eta$ (see Appendix A. Ties are handled via the Breslow approximation of the partial likelihood [Breslow, 1972].
We remark that the model considered in this paper cannot be reduced to a Cox model. Indeed, the C-mix model intensity can be written (in the geometric case)

\[
\lambda(t) = \frac{\alpha_1(1 - \alpha_1)t^{-1} + \alpha_0(1 - \alpha_0)t^{-1} \exp(x^T\beta)}{(1 - \alpha_1)t + (1 - \alpha_0)t \exp(x^T\beta)},
\]

while it is given by Equation (1) in the Cox model.

The second model we consider is the CURE model, introduced in Farewell [1982], with a logistic function for the incidence part and a parametric survival model for \(S(t|Z = 1)\), where \(Z = 0\) means that patient is cured, \(Z = 1\) means that patient is not cured, and \(S(t) = \exp(-\int_0^t \lambda(s)ds)\) denotes the survival function. In this model, we then have \(S(t|Z = 0)\) constant and equal to 1. We add an Elastic-Net regularization term, and since we were not able to find any open source package where CURE models were implemented with a regularized objective, we used the QNEM algorithm in the particular case of CURE model. We just add the constraint that the geometric distribution \(G(\alpha_0)\) corresponding to the cured group of patients \((Z = 0)\) has a parameter \(\alpha_0 = 0\), which does not change over the algorithm iterations.

The QNEM algorithm can be used in this particular case, where some terms have disappeared from the completed log-likelihood, since in the CURE model case we have \(\{i \in \{1, \ldots, n\}: z_i = 0, \delta_i = 1\} = \emptyset\).

### 4.2 Simulation design

In order to assess the proposed method, we perform an extensive Monte Carlo simulation study. Since we want to compare the performances of the 3 models mentioned above, we consider 3 simulation cases for the time distribution: one for each competing model. We first choose a coefficient vector \(\beta = (K, \ldots, K, 0, \ldots, 0) \in \mathbb{R}^d\), with \(K \in \mathbb{R}\) being the value of the active coefficients and \(s \in \{1, \ldots, d\}\) a sparsity parameter. For a desired low-risk patients proportion \(\pi_0 \in [0, 1]\), the high-risk patients index set is given by

\[
\mathcal{H} = \{[(1 - \pi_0) \times n] \text{ random samples without replacement}\} \subset \{1, \ldots, n\},
\]

where \([a]\) denotes the largest integer less than or equal to \(a \in \mathbb{R}\). For the generation of the covariates matrix, we first take \([x_{ij}] \in \mathbb{R}^{n \times d} \sim \mathcal{N}(0, \Sigma(\rho))\), with \(\Sigma(\rho)\) a \((d \times d)\) Toeplitz covariance matrix [Mukherjee and Maiti, 1988] with correlation \(\rho \in (0, 1)\). We then add a gap \(\in \mathbb{R}^+\) value for patients \(i \in \mathcal{H}\) and subtract it for patients \(i \notin \mathcal{H}\), only on active covariates plus a proportion \(r_{cf} \in [0, 1]\) of the non-active covariates considered as confusion factors, that is

\[
x_{ij} \leftarrow x_{ij} \pm \text{gap for } j \in \{1, \ldots, s, \ldots, [(d - s)r_{cf}]\}.
\]

Note that this is equivalent to generate the covariates according to a gaussian mixture.

Then we generate \(Z_i \sim \mathcal{B}(1 - \pi_\beta(x_i))\) in the C-mix or CURE simulation case, where \(\pi_\beta(x_i)\) is computed given Equation (2), with geometric distributions for the durations (see Section 3.3). We obtain \(T_i \sim \mathcal{G}(\alpha_{Z_i})\) in the C-mix case, and \(T_i \sim\)
\( \infty \mathbb{1}_{\{Z_i=0\}} + \mathcal{G}(\alpha_1) \mathbb{1}_{\{Z_i=1\}} \) in the CURE case. For the Cox PH model, we take \( T_i \sim -\log(U_i) \exp(-x_i^\top \beta) \), with \( U_i \sim \mathcal{U}([0, 1]) \) and where \( \mathcal{U}([a, b]) \) stands for the uniform distribution on a segment \([a, b]\).

The distribution of the censoring variable \( C_i \) is geometric \( \mathcal{G}(\alpha_c) \), with \( \alpha_c \in (0, 1) \). The parameter \( \alpha_c \) is tuned to maintain a desired censoring rate \( r_c \in [0, 1] \), using a formula given in Appendix C. The values of the chosen hyper parameters are summarized in Table 1.

Table 1: Hyper-parameters choice for simulation

| \( \eta \) | \( n \) | \( d \) | \( s \) | \( r_{cf} \) | \( K \) | \( \rho \) | \( \pi_0 \) | gap | \( r_c \) | \( \alpha_0 \) | \( \alpha_1 \) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.1 | 100, 200, 500 | 30, 100 | 10 | 0.3 | 1 | 0.5 | 0.75 | 0.1, 0.3, 1 | 0.2, 0.5 | 0.01 | 0.5 |

Note that when simulating under the CURE model, the proportion of censored time events is at least equal to \( \pi_0 \) : we then choose \( \pi_0 = 0.2 \) for the CURE simulations only.

Finally, we want to assess the stability of the C-mix model in terms of variable selection and compare it to the CURE and Cox PH models. To this end, we follow the same simulation procedure explained in the previous lines. For each simulation case, we make vary the two hyper-parameters that impact the most the stability of the variable selection, that is the gap varying in \([0, 2]\) and the confusion rate \( r_{cf} \) varying in \([0, 1]\). All other hyper-parameters are the same than in Table 1, except \( s = 150 \) and with the choice \((n, d) = (200, 300)\). For a given hyper-parameters configuration \((\text{gap}, r_{cf})\), we use the following approach to evaluate the variable selection power of the models. Denoting \( \tilde{\beta}_i = |\hat{\beta}_i|/\max\{ |\hat{\beta}_i|, i \in \{1, \ldots, d\} \} \), if we consider that \( \tilde{\beta}_i \) is the predicted probability that the true \( \beta_i \) equals \( K \), then we are in a binary prediction setting and we use the resulting AUC of this problem.

### 4.3 Metrics

We detail in this section the metrics considered to evaluate risk prediction performances. Let us denote by \( M \) the marker under study. Note that \( M = \pi \hat{\beta}(X) \) in the C-mix and the CURE model cases, and \( M = \exp(X^\top \hat{\beta}_{\text{cox}}) \) in the Cox PH model case. We denote by \( h \) the probability density function of marker \( M \), and assume that the marker is measured once at \( t = 0 \).

For any threshold \( \xi \), cumulative true positive rates and dynamic false positive rates are two functions of time respectively defined as \( \text{TPR}^C(\xi, t) = \mathbb{P}[M > \xi | T \leq t] \) and \( \text{FPR}^B(\xi, t) = \mathbb{P}[M > \xi | T > t] \). Then, as introduced in Heagerty et al. [2000], the cumulative dynamic time-dependent AUC is defined as follows

\[
\text{AUC}^{C, B}(t) = \int_{-\infty}^{\infty} \text{TPR}^C(\xi, t) \left| \frac{\partial \text{FPR}^B(\xi, t)}{\partial \xi} \right| d\xi,
\]

that we simply denote \( \text{AUC}(t) \) in the following. We use the Inverse Probability of Censoring Weighting (IPCW) estimate of this quantity with a Kaplan-Meier estimator of the conditional survival function \( \mathbb{P}[T > t | M = m] \), as proposed in Blanche et al. [2013] and already implemented in the R package timeROC.
A common concordance measure that does not depend on time is the C-index [Harrell et al., 1996] defined by

\[ C = P[M_i > M_j | T_i < T_j], \]

with \( i \neq j \) two independent patients (which does not depend on \( i, j \) under the i.i.d. sample hypothesis). In our case, \( T \) is subject to right censoring, so one would typically consider the modified \( C_\tau \) defined by

\[ C_\tau = P[M_i > M_j | Y_i < Y_j, Y_i < \tau], \]

with \( \tau \) corresponding to the fixed and prespecified follow-up period duration [Heagerty and Zheng, 2005]. A Kaplan-Meier estimator for the censoring distribution leads to a nonparametric and consistent estimator of \( C_\tau \) [Uno et al., 2011], already implemented in the R package survival.

Hence in the following, we consider both AUC(\( t \)) and C-index metrics to assess performances.

### 4.4 Results of simulation

We present now the simulation results concerning the C-index metric in the case \((d, r_c) = (30, 0.5)\) in Table 2. See Appendix D for results on other configurations for \((d, r_c)\). Each value is obtained by computing the C-index average and standard deviation (in parenthesis) over 100 simulations. The AUC(\( t \)) average (bold line) and standard deviation (bands) over the same 100 simulations are then given in Figure 1, where \( n = 100 \). Note that the value of the gap can be viewed as a difficulty level of the problem, since the higher the value of the gap, the clearer the separation between the two populations (low risk and high risk patients).

The results measured both by AUC(\( t \)) and C-index lead to the same conclusion: the C-mix model almost always leads to the best results, even under model misspecification, i.e. when data is generated according to the CURE or Cox PH model. Namely, under CURE simulations, C-mix and CURE give very close results, with a strong improvement over Cox PH. Under Cox PH and C-mix simulations, C-mix outperforms both Cox PH and CURE. Surprisingly enough, this exhibits a strong generalization property of the C-mix model, over both Cox PH and CURE.

Note that this phenomenon is particularly strong for small gap values, while with an increasing gap (or an increasing sample size \( n \)), all procedures barely exhibit the same performance. It can be first explained by the non parametric baseline function in the Cox PH model, and second by the fact that unlike the Cox PH model, the C-mix and CURE models exploit directly the mixture aspect.

Finally, Figure 2 gives the results concerning the stability of the variable selection aspect of the competing models. The C-mix model appears to be the best method as well considering the variable selection aspect, even under model misspecification.

We notice a general behaviour of our method that we describe in the following, which is also shared by the CURE model only when the data is simulated according to itself, and which justifies the log scale for the gap to clearly distinguish the three following phases. For very small gap values (less than 0.2), the confusion rate \( r_{cf} \) does not impact the variable selection performances, since adding very small gap
values to the covariates is almost imperceptible. This means that the resulting AUC is the same when there is no confusion factors and when \( r_{cf} = 1 \) (that is when there are half active covariates and half confusion ones). For medium gap values (saying between 0.2 and 1), the confusion factors are more difficult to identify by the model as there number goes up (that is when \( r_{cf} \) increases), which is precisely the confusion factors effect we expect to observe. Then, for large gap values (more than 1), the model succeeds in vanishing properly all confusion factors since the two subpopulations are more clearly separated regarding the covariates, and the problem becomes naturally easier as the gap increases.

Figure 1: Average (bold lines) and standard deviation (bands) for AUC\((t)\) on 100 simulated data with \( n = 100, d = 30 \) and \( r_c = 0.5 \). Rows correspond to the model simulated (cf. Section 4.2) while columns correspond to different gap values (the problem becomes more difficult as the gap value decreases). Surprisingly, our method gives almost always the best results, even under model misspecification (see Cox PH and CURE simulation cases on the second and third rows).
Table 2: Average C-index on 100 simulated data and standard deviation in parenthesis, with $d = 30$ and $r_c = 0.5$. For each configuration, the best result appears in bold.

| Simulation | gap | Estimation          | n = 100       | n = 200       | n = 500       |
|------------|-----|---------------------|---------------|---------------|---------------|
|            |     | C-mix | CURE | Cox PH | C-mix | CURE | Cox PH | C-mix | CURE | Cox PH |
| C-mix      | 0.1 | 0.786 (0.057) | 0.745 (0.076) | 0.701 (0.075) | 0.792 (0.040) | 0.770 (0.048) | 0.739 (0.055) | 0.806 (0.021) | 0.798 (0.023) | 0.790 (0.024) |
|            | 0.3 | 0.796 (0.055) | 0.739 (0.094) | 0.714 (0.088) | 0.794 (0.030) | 0.760 (0.058) | 0.744 (0.055) | 0.801 (0.021) | 0.784 (0.027) | 0.783 (0.026) |
|            | 1   | 0.768 (0.062) | 0.734 (0.084) | 0.756 (0.066) | 0.766 (0.045) | 0.736 (0.054) | 0.764 (0.042) | 0.772 (0.026) | 0.761 (0.027) | 0.772 (0.025) |
| CURE       | 0.1 | 0.779 (0.064) | 0.772 (0.062) | 0.722 (0.073) | 0.790 (0.038) | 0.790 (0.038) | 0.758 (0.049) | 0.798 (0.025) | 0.799 (0.024) | 0.787 (0.025) |
|            | 0.3 | 0.733 (0.073) | 0.732 (0.072) | 0.686 (0.072) | 0.740 (0.053) | 0.741 (0.053) | 0.714 (0.060) | 0.751 (0.029) | 0.751 (0.029) | 0.738 (0.030) |
|            | 1   | 0.659 (0.078) | 0.658 (0.078) | 0.635 (0.070) | 0.658 (0.053) | 0.658 (0.053) | 0.647 (0.047) | 0.657 (0.031) | 0.657 (0.031) | 0.656 (0.032) |
| Cox PH     | 0.1 | 0.940 (0.041) | 0.937 (0.044) | 0.850 (0.097) | 0.959 (0.021) | 0.958 (0.020) | 0.915 (0.042) | 0.964 (0.012) | 0.964 (0.012) | 0.950 (0.016) |
|            | 0.3 | 0.956 (0.030) | 0.955 (0.029) | 0.864 (0.090) | 0.966 (0.020) | 0.965 (0.020) | 0.926 (0.043) | 0.968 (0.013) | 0.969 (0.012) | 0.959 (0.016) |
|            | 1   | 0.983 (0.016) | 0.985 (0.015) | 0.981 (0.019) | 0.984 (0.012) | 0.985 (0.011) | 0.988 (0.010) | 0.984 (0.007) | 0.985 (0.006) | 0.990 (0.005) |
Figure 2: Average AUC calculated according to Section 4.2 and obtained after 100 simulated data for each \((\text{gap}, r_{cf})\) configuration. Note that the gap values are log-scaled. Rows correspond to the model simulated while columns correspond to the model under consideration for the variable selection evaluation procedure. Our method gives the best results in terms of variable selection, even under model misspecification.

5 Application to genetic data

In this section, we apply our method on three genetic datasets and compare its performance to the Cox PH and CURE models. We extracted normalized expression data and survival times \(Y\) in days from breast invasive carcinoma (BRCA, \(n = 1211\)), glioblastoma multiforme (GBM, \(n = 168\)) and kidney renal clear cell carcinoma (KIRC, \(n = 605\)).

These datasets are available on The Cancer Genome Atlas (TCGA) platform, which aims at accelerating the understanding of the molecular basis of cancer through
the application of genomic technologies, including large-scale genome sequencing. For each patient, 20531 covariates corresponding to the normalized gene expressions are available. We randomly split all datasets into a training set and a test set (30% for testing, cross-validation is done on the training).

We compare the three models both in terms of C-index and AUC(t) on the test sets. Inference of the Cox PH model fails in very high dimension on the considered data with the `glmnet` package. We therefore make a first variable selection (screening) among the 20531 covariates. To do so, we compute the C-index obtained by univariate Cox PH models (not to confer advantage to our method), namely Cox PH models fitted on each covariate separately. We then ordered the obtained 20531 C-indexes by decreasing order and extracted the top $d = 100$, $d = 300$ and $d = 1000$ covariates. We then apply the three methods on the obtained covariates.

The results in terms of AUC(t) curves are given in Figure 3 for $d = 300$, where we distinguish the C-mix model with geometric or Weibull distributions.

![AUC(t) comparison](image)

Figure 3: AUC(t) comparison on the three TCGA data sets considered, for $d = 300$. We observe that C-mix model leads to the best results (higher is better) and outperforms both Cox PH and CURE in all cases. Results are similar in terms of performances for the C-mix model with geometric or Weibull distributions.

Then it appears that the performances are very close in terms of AUC(t) between the C-mix model with geometric or Weibull distributions, which is also validated if we compare the corresponding C-index for these two parameterizations in Table 3.

Table 3: C-index comparison between geometric or Weibull parameterizations for the C-mix model on the three TCGA data sets considered (with $d = 300$). In all cases, results are very similar for the two distribution choices.

| Parameterization | Geometric | Weibull |
|------------------|-----------|---------|
| BRCA Cancer      | 0.782     | 0.780   |
| GBM Cancer       | 0.755     | 0.754   |
| KIRC Cancer      | 0.849     | 0.835   |

Similar conclusions in terms of C-index, AUC(t) and computing time can be made on all considered datasets and for any choice of $d$. Hence, as already mentioned in Section 3.3, we only concentrate on the geometric parameterization for the C-mix model. The results in terms of C-index are then given in Table 4.

A more direct approach to compare performances between models, rather than only focus on the marker order aspect, is to predict the survival of patients in
Table 4: C-index comparison on the three TCGA data sets considered. In all cases, C-mix gives the best results (in bold).

| Cancer | BRCA | GBM | KIRC |
|--------|------|-----|------|
| Model  | C-mix | CURE Cox PH | C-mix | CURE Cox PH | C-mix | CURE Cox PH |
| 100    | 0.792 | 0.764 0.705 | 0.826 | 0.695 0.571 | 0.768 | 0.732 0.716 |
| d 300  | 0.782 | 0.753 0.723 | 0.849 | 0.697 0.571 | 0.755 | 0.691 0.698 |
| 1000   | 0.817 | 0.613 0.577 | 0.775 | 0.699 0.592 | 0.743 | 0.690 0.685 |

the test set within a specified short time. For the Cox PH model, the survival $P[T_i > t|X_i = x_i]$ for patient $i$ in the test set is estimated by

$$\hat{S}_i(t|X_i = x_i) = [\hat{S}_0^{\text{cox}}(t)]^{\exp(x_i^T\hat{\beta}^{\text{cox}})},$$

where $\hat{S}_0^{\text{cox}}$ is the estimated survival function of baseline population ($x = 0$) obtained using the Breslow estimate of $\lambda_0$ [Breslow, 1972]. For the CURE or the C-mix models, it is naturally estimated by

$$\hat{S}_i(t|X_i = x_i) = \pi\hat{\beta}(x_i)\hat{S}_0(t) + (1 - \pi\hat{\beta}(x_i))\hat{S}_1(t),$$

where $\hat{S}_0$ and $\hat{S}_1$ are the Kaplan-Meier estimators [Kaplan and Meier, 1958] of the low and high risk subgroups respectively, learned by the C-mix or CURE models (patients with $\pi\hat{\beta}(x_i) > 0.5$ are clustered in the low risk subgroup, others in the high risk one). The corresponding estimated survival curves are given in Figure 4.

We observe that the subgroups obtained by the C-mix are more clearly separated...

![Figure 4: Estimated survival curves per subgroups (blue for low risk and red for high risk) with the corresponding 95% confidence bands for the C-mix and CURE models: BRCA in column (a), GBM in column (b) and KIRC in column (c).](image)
in terms of survival than those obtained by the CURE model.

For a given time $\epsilon$, one can now use $\hat{S}_i(\epsilon|X_i = x_i)$ for each model to predict whether or not $T_i > \epsilon$ on the test set, resulting on a binary classification problem that we assess using the classical AUC score. Note that patients on test set such that $\delta_i = 0$ and $Y_i \leq \epsilon$ have to be excluded from the procedure. By moving $\epsilon$ within the first years of follow-up, since it is the more interesting for physicians in practice, one obtains the curves given in Figure 5.

![Figure 5: Comparison of the survival prediction performances between models on the three TCGA data sets considered (still with $d = 300$). Performances are, onces again, much better for the C-mix over the two other standard methods.](image)

Let us now focus on the runtime comparison between the models in Table 5. We choose the BRCA dataset to illustrate this point, since it is the larger one ($n = 1211$) and consequently provides more clearer time-consuming differences.

Table 5: Computing time comparison in second on the BRCA dataset ($n = 1211$), with corresponding C-index in parenthesis and best result in bold in each case. This times concern the learning task for each model with the best hyper parameter selected after the cross validation procedure. It turns out that our method is by far the fastest in addition to providing the best performances. In particular, the QNEM algorithm is faster than the R implementation glmnet.

| Model | C-mix | CURE | Cox PH |
|-------|-------|------|--------|
| $d$   |       |      |        |
| 100   | 0.025 (0.792) | 1.992 (0.764) | 0.446 (0.705) |
| 300   | 0.027 (0.782) | 2.343 (0.753) | 0.810 (0.723) |
| 1000  | 0.139 (0.817) | 12.067 (0.613) | 2.145 (0.577) |

We also notice that despite using the same QNEM algorithm steps, our CURE model implementation is slower since convergence takes more time to be reached, as shows Figure 6.

In Appendix E, the top 20 selected genes for each cancer type and for all models are presented (for $d = 300$). Literature on those genes is mined to estimate two simple scores that provide information about how related they are to cancer in general first, and second to cancer plus the survival aspect, according to scientific publications. It turns out that some genes have been widely studied in the literature (e.g. FLT3 for the GBM cancer), while for others, very few publications were retrieved (e.g. TRMT2B still for the GBM cancer).
Figure 6: Convergence comparison between C-mix and CURE models through the QNEM algorithm. The relative objective is here defined at iteration $l$ as $(\ell_{n}^{\text{pen}}(\hat{\theta}(l)) - \ell_{n}^{\text{pen}}(\hat{\theta})) / \ell_{n}^{\text{pen}}(\hat{\theta})$, where $\hat{\theta}$ is naturally the parameter vector returned at the end of the QNEM algorithm, that is once convergence is reached. Note that both iteration and relative objective axis are log-scaled for clarity. We observe that convergence for the C-mix model is dramatically faster than the CURE one.

6 Concluding remarks

In this paper, a mixture model for censored durations (C-mix) has been introduced, and a new efficient estimation algorithm (QNEM) has been derived, that considers a penalization of the likelihood in order to perform covariate selection and to prevent overfitting. A strong improvement is provided over the CURE and Cox PH approaches, which are, by far, the most widely used for biomedical data analysis. But more importantly, our method detects relevant subgroups of patients regarding their risk of early death in a supervised learning procedure, and takes advantage of this identification to improve survival prediction over more standard methods. An extensive Monte Carlo simulation study has been carried out to evaluate the performance of the developed estimation procedure. It showed that our approach is robust to model misspecification. The proposed methodology has then been applied on three high dimensional datasets. On these datasets, C-mix outperforms both Cox PH and CURE, in terms of AUC($t$), C-index or survival prediction. Moreover, many gene expressions pinpointed by the feature selection aspect of our regularized method are relevant for medical interpretations (e.g. NFKBIA, LEF1, SUSD3 or FAIM3 for the BRCA cancer, see Zhou et al. [2007] or Oskarsson et al. [2011]), whilst others must involve further investigations in the genetic research community. Finally, our analysis provides, as a by-product, a new robust implementation of CURE models in high dimension. One interesting future research direction is the generalization of the C-mix model to K classes (and not only two), which is in progress and should
lead to the release of a R library.

Software

All the methodology discussed in this paper is implemented in Python. The code is available from https://github.com/SimonBussy/C-mix in the form of annotated programs, together with a notebook tutorial.

Acknowledgements

The results shown in this paper are based upon data generated by the TCGA Research Network and freely available from http://cancergenome.nih.gov/. Conflict of Interest: None declared.

Appendices

A Numerical details

Let us first give some details about the starting point of Algorithm 1. We simply use $\beta^{(0)}$ as the zero vector, and for $\alpha_0^{(0)}$ and $\alpha_1^{(0)}$, we fit a censored parametric mixture model on $(y_i)_{i=1,...,n}$ with an EM algorithm.

Concerning the V-fold cross validation procedure for tuning $\gamma$, we use $V = 5$ and the cross-validation metric is the C-index. Let us precise that we choose $\gamma$ as the largest value such that error is within one standard error of the minimum, and that a grid-search is made during the cross-validation on an interval $[\gamma_{\text{max}} \times 10^{-4}, \gamma_{\text{max}}]$, with $\gamma_{\text{max}}$ the interval upper bound computed in the following.

Let us consider the following convex minimization problem resulting from Equation (7), at a given step $l$:

$$\hat{\beta} \in \argmin_{\beta \in \mathbb{R}^d} R_n^{(l)}(\beta) + \gamma((1 - \eta)\|\beta\|_1 + \frac{\eta}{2}\|\beta\|_2^2).$$

Regarding the grid of candidate values for $\gamma$, we consider $\gamma_1 \leq \gamma_2 \leq \cdots \leq \gamma_{\text{max}}$. At $\gamma_{\text{max}}$, all coefficients $\hat{\beta}_j$ for $j \in \{1, \ldots, d\}$ are exactly zero. The KKT conditions [Boyd and Vandenberghe, 2004] claim that

$$\begin{cases}
\frac{\partial R_n^{(l)}(\hat{\beta})}{\partial \hat{\beta}_j} = \gamma(1 - \eta) \sgn(\hat{\beta}_j) + \eta \hat{\beta}_j & \forall j \in \hat{A} \\
\left| \frac{\partial R_n^{(l)}(\hat{\beta})}{\partial \hat{\beta}_j} \right| < \gamma(1 - \eta) & \forall j \notin \hat{A}
\end{cases}.$$
where \( \hat{A} = \{ j \in \{1, \ldots, d\} : \hat{\beta}_j \neq 0 \} \) is the active set of the \( \hat{\beta} \) estimator, and for all \( x \in \mathbb{R} \setminus \{0\} \), \( \text{sgn}(x) = 1_{\{x>0\}} - 1_{\{x<0\}} \). Then, using (9), one obtains

\[
\forall j \in \{1, \ldots, d\}, \; \hat{\beta}_j = 0 \Rightarrow \forall j \in \{1, \ldots, d\}, \; \left| n^{-1} \sum_{i=1}^{n} (q_i^{(l)} - \frac{1}{2}) x_{ij} \right| < \gamma(1 - \eta)
\]

\[
\Rightarrow \gamma(1 - \eta) > \max_{j \in \{1, \ldots, d\}} \left| n^{-1} \sum_{i=1}^{n} (q_i^{(l)} - \frac{1}{2}) x_{ij} \right|.
\]

Hence, since \( |q_i^{(l)} - \frac{1}{2}| \leq \frac{1}{2} \), we choose the following upper bound for the grid search interval during the cross-validation procedure

\[
\gamma_{\text{max}} = \frac{1}{2n(1 - \eta)} \max_{j \in \{1, \ldots, d\}} \sum_{i=1}^{n} |x_{ij}|.
\]

**B Proof of Theorem 1**

Without loss of generality, suppose \( \alpha_0 \in \mathbb{R}^{d_0} \) and \( \alpha_1 \in \mathbb{R}^{d_1} \), so that \( \theta = (\alpha_0, \alpha_1, \beta) \in \Theta \subset \mathbb{R}^{d_0+d_1+d} \). We denote \( \bar{\theta} \) a cluster point of the sequence \( S = \{\theta(l); l = 0, 1, 2, \ldots\} \) generated by the QNEM algorithm, i.e. \( \forall \epsilon > 0, V_{\epsilon}(\bar{\theta}) \cap S \setminus \{\bar{\theta}\} \neq \emptyset \), with \( V_{\epsilon}(\bar{\theta}) \) the epsilon-neighbourhood of \( \bar{\theta} \). We want to prove that \( \bar{\theta} \) is a stationary point of the non-differentiable function \( \theta \mapsto \ell_{\text{pen}}^{\theta}(\theta) \), which means [Tseng, 2001]:

\[
\forall r \in \mathbb{R}^{d_0+d_1+d}, \nu_n^{\theta} \left( \bar{\theta}; r \right) = \lim_{\zeta \to 0} \frac{\ell_{\text{pen}}^{\theta}(\bar{\theta} + r\zeta) - \ell_{\text{pen}}^{\theta}(\bar{\theta})}{\zeta} \geq 0. \tag{10}
\]

The proof is inspired by Bertsekas [1995]. The conditional density of the complete data given the observed data can be written as

\[
k(\theta) = \frac{\exp \left( \ell_{\text{comp}}^{\theta}(\theta) \right)}{\exp \left( \ell_{n}(\theta) \right)}.
\]

Then, one has

\[
\ell_{\text{pen}}^{\theta}(\theta) = \mathbb{E}_{\theta(l)}^{\theta} \left( \mathcal{Q}_{\text{pen}}^{\theta}(- \theta, \theta(l)) - H(\theta, \theta(l)) \right), \tag{11}
\]

where we introduced \( H(\theta, \theta(l)) = \mathbb{E}_{\theta(l)} \left[ \log (k(\theta)) \right] \). The key argument relies on the following facts that hold under Hypothesis (2) and (3):

- \( \mathbb{Q}_{n}^{\theta}(\theta, \theta(l)) \) is continuous in \( \theta \) and \( \theta(l) \),

- for any fixed \( \theta(l) \) (at the \( (l+1) \)-th M step of the algorithm), \( \mathbb{Q}_{n,\theta(l)}^{\theta}(\theta) \) is convex in \( \theta \) and strictly convex in each coordinate of \( \theta \).

Let \( r \in \mathbb{R}^{d_0+d_1+d} \) be an arbitrary direction, then equations (10) and (11) yield

\[
\ell_{\text{pen}}^{\theta} \left( \bar{\theta}; r \right) = \mathbb{Q}_{n,\theta}^{\theta} \left( \bar{\theta}; r \right) - \langle \nabla H_{\bar{\theta}}(\bar{\theta}), r \rangle.
\]

Hence, by Jensen’s inequality we get

\[
\forall \theta \in \Theta, H(\theta(l), \theta(l)) \leq H(\theta, \theta(l)), \tag{12}
\]
and so \( \theta \mapsto H_{\theta}(\theta) \) is minimized for \( \theta = \theta(1) \), then we have \( \nabla H_{\theta}(\bar{\theta}) = 0 \). It remains to prove that \( Q_{n,\theta}^{\text{pen}}(\bar{\theta}, r) \geq 0 \). Let \( D = d_0 + d_1 + d \) be the number of coordinates of \( \theta = (\theta_1, \ldots, \theta_D) \) and focus on the proof of the following expression

\[
\forall x_1, Q_{n,\theta}^{\text{pen}}(\bar{\theta}) \leq Q_{n,\theta}^{\text{pen}}(x_1, \bar{\theta}_2, \ldots, \bar{\theta}_D). \tag{13}
\]

Denoting \( w_i^{(l)} = (\theta_1^{(l+1)}, \ldots, \theta_i^{(l+1)}), \theta_1^{(l)}, \ldots, \theta_D^{(l)} \) and from the definition of the QNEM algorithm, we first have

\[
Q_{n,\theta(1)}^{\text{pen}}(\theta^{(1)}) \geq Q_{n,\theta(1)}^{\text{pen}}(w_1^{(l)}) \geq \cdots \geq Q_{n,\theta(1)}^{\text{pen}}(w_{D-1}^{(l)}) \geq Q_{n,\theta(1)}^{\text{pen}}(\theta^{(l+1)}), \tag{14}
\]

and second for all \( x_1, Q_{n,\theta(1)}^{\text{pen}}(w_1^{(l)}) \leq Q_{n,\theta(1)}^{\text{pen}}(x_1, \theta_2^{(l)}, \ldots, \theta_D^{(l)}). \) Consequently, if \( (w_1^{(l)})_{l \in \mathbb{N}} \) converges to \( \bar{\theta} \), one obtains (13) by continuity taking the limit \( l \to \infty \). Let us now suppose that \((w_1^{(l)})_{l \in \mathbb{N}}\) does not converge to \( \bar{\theta} \). So that \((\bar{w}_1^{(l)} - \theta^{(l)})_{l \in \mathbb{N}}\) does not converge to 0, or equivalently: there exists a subsequence \((\bar{w}_1^{(l)} - \theta^{(l)})_{l \in \mathbb{N}}\) not converging to 0.

Then, denoting \( \psi^{(l)} = \|w_1^{(l)} - \theta^{(l)}\|_2 \), we may assume that there exists \( \bar{\psi} > 0 \) such that \( \forall j \in \mathbb{N}, \psi^{(l)}_j > \bar{\psi} \) by removing from the subsequence \((w_1^{(l)} - \theta^{(l)})_{j \in \mathbb{N}}\) any terms for which \( \psi^{(l)}_j = 0 \). Let \( s^{(l)}_1 = \frac{w^{(l)}_1 - \theta^{(l)}}{\psi^{(l)}_1} \), so that \( (s^{(l)}_1)_{j \in \mathbb{N}} \) belongs to a compact set \( (\|s^{(l)}_1\| = 1) \) and then converges to \( \bar{s}_1 \neq 0 \). Let us fix some \( \epsilon \in [0, 1] \), then \( 0 \leq \epsilon \bar{\psi} \leq \psi^{(l)}_1 \). Moreover, \( \psi^{(l)}_j + \psi^{(l)}_1 \bar{s}_1 \) lies on the segment joining \( \theta^{(l)}_j \) and \( w_1^{(l)} \), and consequently belongs to \( \Theta \) since \( \Theta \) is convex. As \( Q_{n,\theta^{(l)}_j}^{\text{pen}}(.) \) is convex and \( w_1^{(l)} \) minimizes this function over all values that differ from \( \theta^{(l)}_j \) along the first coordinate, one has

\[
Q_{n,\theta^{(l)}_j}^{\text{pen}}(w_1^{(l)}) = Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) + \psi^{(l)}_1 \bar{s}_1^{(l)}_1 \]

\[
\leq Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) + \epsilon \bar{\psi} s^{(l)}_1 \]

\[
\leq Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j). \tag{15}
\]

We finally obtain

\[
0 \leq Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) - Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) + \epsilon \bar{\psi} s^{(l)}_1 \]

\[
\leq (15) \quad Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) - Q_{n,\theta^{(l)}_j}^{\text{pen}}(w_1^{(l)})
\]

\[
\leq (14) \quad Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j) - Q_{n,\theta^{(l)}_j}^{\text{pen}}(\theta^{(l)}_j + 1)
\]

\[
\leq (11) \quad \ell_n^{\text{pen}}(\theta^{(l)}_j) - \ell_n^{\text{pen}}(\theta^{(l)}_j + 1) + H_{\theta^{(l)}_j}(\theta^{(l)}_j) - H_{\theta^{(l)}_j}(\theta^{(l)}_j + 1)
\]

\[
\leq (12) \quad 0
\]

By continuity of the function \( Q_{n,\theta}^{\text{pen}}(x,y) \) in both \( x \) and \( y \) and taking the limit \( j \to \infty \), we conclude that \( \forall \epsilon \in [0, 1], Q_{n,\theta}^{\text{pen}}(\bar{\theta} + \epsilon \bar{\psi} \bar{s}_1) = Q_{n,\theta}^{\text{pen}}(\bar{\theta}) \). Since \( \bar{\psi} \bar{s}_1 \neq 0 \), this
contradicts the strict convexity of \( x_1 \mapsto Q_{n,\theta(t)}^{pen}(x_1, \theta_2^{(l)}, \ldots, \theta_D^{(l)}) \) and establishes that \((w_1^{(l)})_{l \in \mathbb{N}}\) converges to \(\bar{\theta}\).

Hence (13) is proved. Repeating the argument to each coordinate, we deduce that \(\bar{\theta}\) is a coordinate-wise minimum, and finally conclude that \(\ell_n^{pen} (\theta; r) \geq 0\) [Tseng, 2001]. Thus, \(\bar{\theta}\) is a stationary point of the criterion function defined in Equation (3).

\[\Box\]

C Tuning of the censoring level

Suppose that we want to generate data following the procedure detailed in Section 4.2, in the C-mix with geometric distributions or CURE case. The question here is to choose \(\alpha_c\) for a desired censoring rate \(r_c\), and for some fixed parameters \(\alpha_0\), \(\alpha_1\) and \(\pi_0\). We write

\[
1 - r_c = E[\delta] = \sum_{k=0}^{+\infty} \sum_{j=1}^{+\infty} \left[ \alpha_0(1 - \alpha_0)^{j-1}\pi_0 + \alpha_1(1 - \alpha_1)^{j-1}(1 - \pi_0) \right] \alpha_c(1 - \alpha_c)^{j+k-1}
\]

\[
= \frac{\alpha_0\pi_0[1 - (1 - \alpha_1)(1 - \alpha_c)] + \alpha_1(1 - \pi_0)[1 - (1 - \alpha_0)(1 - \alpha_c)]}{[1 - (1 - \alpha_0)(1 - \alpha_c)][1 - (1 - \alpha_1)(1 - \alpha_c)]}.
\]

Then, if we denote \(\tilde{r}_c = 1 - r_c\), \(\tilde{\alpha}_c = 1 - \alpha_c\), \(\tilde{\alpha}_0 = 1 - \alpha_0\), \(\tilde{\alpha}_1 = 1 - \alpha_1\) and \(\tilde{\pi}_0 = 1 - \pi_0\), we can choose \(\alpha_c\) for a fixed \(r_c\) by solving the following quadratic equation

\[
(\tilde{r}_c \tilde{\alpha}_0 \tilde{\alpha}_1)\tilde{\alpha}_c^2 + (\alpha_0\pi_0\tilde{\alpha}_1 + \alpha_1\tilde{\pi}_0 \tilde{\alpha}_0 - \tilde{r}_c(\tilde{\alpha}_1 + \tilde{\alpha}_0))\tilde{\alpha}_c + (r_c - \alpha_0\pi_0 - \alpha_1\tilde{\pi}_0) = 0,
\]

for which one can prove that there is always a unique root in \((0, 1)\).

D Extended simulation results

Table 6 below presents the results of simulation for the configurations \((d, r_c) = (30, 0.2), (100,0.2)\) and \((100,0.5)\).

E Selected genes per model on the TCGA datasets

In Tables 7, 8 and 9 hereafter, we detail the 20 most significant covariates for each model and for the three considered datasets. For each selected gene, we precise the corresponding effect in percentage, where we define the effect of covariate \(j\) as \(100 \times |\beta_j| / \|\beta\|_1\%\). Then, to explore physiopathological and epidemiological background that could explain the role of the selected genes in cancer prognosis, we search in MEDLINE (search performed on the 15th september 2016 at http://www.nlm.nih.gov/bsd/pmresources.html) the number of publications for different requests: (1) selected gene name (e.g. UBTF), (2) selected gene name and cancer (e.g. UBTF AND cancer[Mesh]), (3) selected gene name and cancer survival (e.g. UBTF AND cancer[Mesh] AND survival). We then estimate \(f_1\) defined here as the frequency of publication dealing with cancer among all publications for this
Table 6: Average C-index on 100 simulated data for different configurations \((d, r_c)\), with geometric distributions for the C-mix model. For each configuration, the best result appears in bold.

\[(d, r_c) = (30, 0.2)\]

| Simulation gap | Estimation | n = 100 | n = 200 | n = 500 |
|----------------|------------|---------|---------|---------|
|                | C-mix      | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  |
| 0.1            | 0.753 (0.055) | 0.637 (0.069) | 0.658 (0.081) | 0.762 (0.034) | 0.664 (0.070) | 0.704 (0.051) | 0.767 (0.023) | 0.686 (0.062) | 0.749 (0.025) |
| 0.3            | 0.756 (0.050) | 0.599 (0.073) | 0.657 (0.075) | 0.761 (0.033) | 0.600 (0.064) | 0.713 (0.050) | 0.757 (0.020) | 0.565 (0.049) | 0.740 (0.021) |
| 1              | 0.723 (0.059) | 0.710 (0.063) | 0.714 (0.062) | 0.723 (0.042) | 0.718 (0.044) | 0.721 (0.040) | 0.727 (0.026) | 0.723 (0.028) | 0.726 (0.025) |

\[(d, r_c) = (100, 0.2)\]

| Simulation gap | Estimation | n = 100 | n = 200 | n = 500 |
|----------------|------------|---------|---------|---------|
|                | C-mix      | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  |
| 0.1            | 0.736 (0.048) | 0.601 (0.081) | 0.656 (0.066) | 0.757 (0.037) | 0.629 (0.079) | 0.697 (0.057) | 0.767 (0.020) | 0.659 (0.073) | 0.744 (0.024) |
| 0.3            | 0.733 (0.056) | 0.582 (0.063) | 0.648 (0.073) | 0.757 (0.035) | 0.572 (0.047) | 0.699 (0.057) | 0.758 (0.023) | 0.558 (0.040) | 0.736 (0.031) |
| 1              | 0.723 (0.067) | 0.717 (0.073) | 0.705 (0.063) | 0.721 (0.041) | 0.716 (0.041) | 0.719 (0.046) | 0.724 (0.023) | 0.720 (0.025) | 0.726 (0.023) |

\[(d, r_c) = (100, 0.5)\]

| Simulation gap | Estimation | n = 100 | n = 200 | n = 500 |
|----------------|------------|---------|---------|---------|
|                | C-mix      | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  |
| 0.1            | 0.773 (0.064) | 0.710 (0.087) | 0.678 (0.078) | 0.798 (0.038) | 0.767 (0.057) | 0.741 (0.055) | 0.804 (0.022) | 0.795 (0.024) | 0.788 (0.025) |
| 0.3            | 0.781 (0.057) | 0.696 (0.103) | 0.697 (0.087) | 0.798 (0.034) | 0.741 (0.064) | 0.741 (0.055) | 0.800 (0.021) | 0.778 (0.036) | 0.785 (0.023) |
| 1              | 0.772 (0.064) | 0.742 (0.081) | 0.760 (0.071) | 0.772 (0.044) | 0.732 (0.074) | 0.771 (0.041) | 0.770 (0.028) | 0.740 (0.059) | 0.771 (0.029) |

| Simulation gap | Estimation | n = 100 | n = 200 | n = 500 |
|----------------|------------|---------|---------|---------|
|                | C-mix      | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  |
| 0.1            | 0.755 (0.070) | 0.759 (0.068) | 0.692 (0.082) | 0.780 (0.044) | 0.782 (0.043) | 0.752 (0.052) | 0.795 (0.025) | 0.795 (0.025) | 0.785 (0.026) |
| 0.3            | 0.730 (0.077) | 0.737 (0.076) | 0.674 (0.086) | 0.740 (0.042) | 0.740 (0.041) | 0.708 (0.055) | 0.753 (0.028) | 0.753 (0.027) | 0.740 (0.031) |
| 1              | 0.663 (0.075) | 0.660 (0.076) | 0.659 (0.064) | 0.661 (0.053) | 0.661 (0.052) | 0.658 (0.050) | 0.657 (0.032) | 0.657 (0.033) | 0.657 (0.034) |

| Simulation gap | Estimation | n = 100 | n = 200 | n = 500 |
|----------------|------------|---------|---------|---------|
|                | C-mix      | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  | C-mix   | CURE    | Cox PH  |
| 0.1            | 0.916 (0.069) | 0.924 (0.056) | 0.837 (0.097) | 0.950 (0.028) | 0.949 (0.029) | 0.911 (0.052) | 0.964 (0.012) | 0.964 (0.012) | 0.951 (0.016) |
| 0.3            | 0.937 (0.047) | 0.934 (0.050) | 0.863 (0.071) | 0.955 (0.026) | 0.956 (0.022) | 0.925 (0.037) | 0.968 (0.012) | 0.968 (0.012) | 0.958 (0.015) |
| 1              | 0.963 (0.029) | 0.967 (0.027) | 0.973 (0.024) | 0.966 (0.019) | 0.970 (0.017) | 0.984 (0.012) | 0.962 (0.012) | 0.966 (0.011) | 0.988 (0.006) |
gene, i.e. $(2)/(1)$, and $f_2$ defined as the frequency of publication dealing with survival among publications dealing with cancer, i.e. $(3)/(2)$. A $f_1$ (respectively $f_2$) close to 1 just informs that the corresponding gene is well known to be highly related to cancer (respectively to cancer survival) by the genetic research community. Note that the CURE and Cox PH models tend to have a smaller support than the C-mix one, since they tend to select less than 20 genes.

Table 7: Top 20 selected genes per model for the BRCA cancer, with the corresponding effects. Dots (·) mean zeros.

| Genes     | Model effects (%) | MEDLINE data |        |        |
|-----------|-------------------|--------------|--------|--------|
|           | C-mix  | CURE  | Cox PH | (1)    | $f_1$  | $f_2$   |
| PHKB|5257 | 9.8   | 7.2   | 4.3    | 1079   | 0.20  | 0.37    |
| UBTF|7343 | 7.8   | 5.8   | 21.7   | 14     | 0.21  |        |
| LOC100132707 | 5.7     | 3.9   | 18.8  |        |        |        |
| CHTF8|54921| 4.4   | -     | 7.2    | 1      | 1     |        |
| NFKBIA|4792| 4.3   | 1.9   | 3.4    | 247    | 0.27  | 0.22    |
| EPB41L4B|54566| 3.6   | 2.6   | -      | 19     | 0.47  | 0.22    |
| UGP2|7360 | 3.6   | 2.2   | -      | 19     | 0.15  | 1       |
| DPY19L2P1|554236| 3.3   | -     | 3.3    | 1      |        |        |
| TRMT2B|79979| 3.3   | 2.2   | -      |        |        |        |
| HSD3B7|80270| 3.2   | 1.9   | 7.6    | 19     | 0.05  |        |
| DLAT|1737 | 3.2   | 2.9   | -      | 75     | 0.16  | 0.16    |
| NIPAL2|79815| 2.8   | 1.9   | -      |        |        |        |
| FGD3|89846| 2.7   | -     | 5.9    | 10     | 0.2   | 0.5     |
| JRKL|8690 | 2.7   | 2.6   | -      | 2      |        |        |
| ZBED1|9189 | 2.5   | 2.4   | -      | 6      |        |        |
| KCNJ1|3767 | 2.3   | -     | -      | 647    | 0.02  |        |
| WAC|51322| 2.0   | 3.2   | -      | 260    | 0.05  | 0.25    |
| FLT3|2322 | 2.0   | -     | -      | 4435   | 0.55  | 0.42    |
| STK3|6788 | 1.9   | 2.3   | -      | 107    | 0.32  | 0.15    |
| PAOX|196743| 1.9   | 1.9   | -      | 18     | 0.11  |        |
| C14orf68|283600| -     | 3.3   | -      |        |        |        |
| LIN7C|55327| -     | 3.1   | -      | 36     | 0.06  |        |
| PNRC2|55629| -     | 2.1   | -      | 15     |        |        |
| SLC39A7|7922| -     | 1.8   | -      | 22     | 0.18  |        |
| MAGT1|84061| -     | 1.7   | -      | 50     | 0.12  | 0.17    |
| IRF2|3660 | -     | -     | 10.9   | 310    | 0.21  | 0.14    |
| PEL0|53918| -     | -     | 7.0    | 265    | 0.08  | 0.04    |
| SUSD3|203328| -     | -     | 5.3    | 5      | 0.6   | 0.67    |
| LEF1|51176| -     | -     | 3.2    | 940    | 0.29  | 0.23    |
| CPA4|51200| -     | -     | 1.4    | 18     | 0.22  |        |
Table 8: Top 20 selected genes per model for the GBM cancer, with the corresponding effects. Dots (·) mean zeros.

| Genes         | Model effects (%) | MEDLINE data |
|---------------|-------------------|--------------|
|               | C-mix  | CURE  | Cox PH | (1) f₁ | f₂ |
| ARM CX6[54470] | 4.9    | -     | 23.6   | 1      | ·  |
| FAM35A[54537]  | 4.4    | -     | 21.8   | ·      | ·  |
| CLEC4GP1[440508] | 3.9   | 5.1   | 2.8    | ·      | ·  |
| INSL3[3640]    | 3.6    | 2.7   | 1.7    | 404    | 0.06 0.12 |
| REM1[28954]    | 3.2    | ·     | ·      | 54     | 0.05 0.66 |
| FAM35B2[439965] | 3.0   | ·     | ·      | ·      | ·  |
| TSPAN4[7106]   | 2.7    | ·     | ·      | 16     | 0.31 0.4 |
| AP3M1[26985]   | 2.7    | ·     | ·      | 2      | 0.5  |
| PXN[5829]      | 2.6    | ·     | 15.4   | 891    | 0.25 0.18 |
| PDE4C[5143]    | 2.5    | ·     | ·      | 67     | 0.06 0.25 |
| PGBD5[79605]   | 2.5    | ·     | ·      | 5      | 0.25  |
| NRG1[3084]     | 2.4    | ·     | 18.5   | 1207   | 0.12 0.29 |
| LOC653786      | 2.2    | ·     | ·      | ·      | ·  |
| FERMT1[55612]  | 2.1    | ·     | ·      | 115    | 0.19 0.18 |
| PLD3[23646]    | 2.0    | ·     | ·      | 38     | 0.10 0.25 |
| MIER1[57708]   | 1.9    | ·     | 2.1    | 16     | 0.31  |
| UTP14C[9724]   | 1.8    | ·     | ·      | 5      | 0.4  |
| AZU1[566]      | 1.8    | ·     | ·      | 15     | 0.2  0.33 |
| KCNC4[3749]    | 1.7    | ·     | ·      | 30     | 0.1  0.33 |
| FAM35B[414241] | 1.6    | ·     | ·      | ·      | ·  |
| CRELD1[78987]  | ·      | 32.2  | ·      | 32     | 0.03  |
| HMGN5[79366]   | ·      | 21.2  | ·      | 41     | 0.54 0.32 |
| PNLDC1[154197] | ·      | 12.2  | ·      | 3      | ·  |
| LOC493754      | ·      | 9.8   | ·      | ·      | ·  |
| KIAA0146[23514] | ·     | 8.7   | ·      | 3      | 0.67  |
| TMCO655374     | ·      | 3.6   | ·      | 4      | 0.25  |
| ABLIM1[3983]   | ·      | 2.1   | ·      | 20     | 0.2  |
| OSBP1L11[14885] | ·    | 1.0   | ·      | ·      | ·  |
| TRAPPC1[58485] | ·      | 0.9   | ·      | 4      | 0.75  |
| TBCE[219899]   | ·      | 0.5   | ·      | 7      | 0.28  |
| RPL39L[116832] | ·      | ·     | 8.8   | 10     | 0.7  0.14 |
| GALE[2582]     | ·      | ·     | 3.5   | 540    | 0.02  |
| BBC3[27113]    | ·      | ·     | 0.7   | 561    | 0.54 0.38 |
| DUSP6[1848]    | ·      | ·     | 0.6   | 307    | 0.30 0.22 |
Table 9: Top 20 selected genes per model for the KIRC cancer, with the corresponding effects. Dots (·) mean zeros.

| Genes        | Model effects (%) | MEDLINE data |
|--------------|-------------------|--------------|
|              | C-mix  | CURE  | Cox PH | (1) | f_1 | f_2 |
| BCL2L12|83596   | 8.6    | 2.7    | -   | 64  | 0.72 | 0.39 |
| MARS|4141        | 7.5    | 6.9    | 7.2 | 577 | 0.02 | 0.1 |
| NUMBLL|9253        | 7.2    | 28.6   | 3.3 | 56  | 0.14 | 0.25 |
| CKAP4|10970       | 6.1    | 10.6   | 22.3| 825 | 0.63 | 0.11 |
| HN1|51155        | 5.8    | 3.8    | -   | 13  | 0.38 | 0.2 |
| GIPC2|54810        | 5.7    | -      | -   | 15  | 0.6  | 0.11 |
| NPR3|4883         | 5.2    | -      | -   | 105 | 0.05 | 0.6 |
| GBA3|57733        | 5.0    | -      | -   | 19  | 0.10 |       |
| SLC47A1|55244      | 5.0    | -      | -   | 70  | 0.06 |       |
| ALDH3A2|224         | 4.7    | -      | 2.6 | 52  | 0.06 | 0.33 |
| CCNF|899         | 4.2    | 2.8    | -   | 50  | 0.24 | 0.08 |
| EHHADH|1962        | 3.9    | -      | -   | 90  | 0.1  |       |
| SGCB|6443         | 3.3    | -      | -   | 30  | -    |       |
| GFPT2|9945         | 2.7    | 1.3    | -   | 18  | 0.22 | 0.25 |
| PPAP2B|8613        | 2.3    | -      | -   | 29  | 0.17 | 0.2 |
| MBOAT7|79143       | 1.9    | 13.8   | 11.1| 15  | -    | -    |
| OSBPL1A|114876     | 1.5    | -      | -   | 7   | -    |       |
| C16orf57|79650       | 1.2    | -      | -   | 26  | -    |       |
| ATXNL3|56970       | 0.9    | 2.5    | -   | 9   | -    |       |
| C16orf59|80178       | 0.8    | -      | -   | 3   | 0.66 | -    |
| STRADA|92335       | -      | 20.7   | 53.5| 9   | -    |       |
| ABCC10|89845       | -      | 3.9    | -   | 80  | 0.32 | 0.23 |
| MDK|4192        | -      | 1.2    | -   | 789 | 0.38 | 0.23 |
| C16orf59|80178       | -      | 1.1    | -   | 3   | 0.6  | -    |
References

Ash A Alizadeh, Michael B Eisen, R Eric Davis, Chi Ma, Izidore S Lossos, Andreas Rosenwald, Jennifer C Boldrick, Hajeer Sabet, Truc Tran, Xin Yu, et al. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403(6769):503–511, 2000.

Galen Andrew and Jianfeng Gao. Scalable training of l1-regularized log-linear models. In *International Conference on Machine Learning*, pages 33–40. ACM, 2007.

David G Beer, Sharon LR Kardia, Chiang-Ching Huang, Thomas J Giordano, Albert M Levin, David E Misek, Lin Lin, Guoan Chen, Tarek G Gharib, Dafydd G Thomas, et al. Gene-expression profiles predict survival of patients with lung adenocarcinoma. *Nature medicine*, 8(8):816–824, 2002.

D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, Belmont, MA, 1995.

Arindam Bhattacharjee, William G Richards, Jane Staunton, Cheng Li, Stefano Monti, Priya Vasa, Christine Ladd, Javad Beheshti, Raphael Bueno, Michael Gillette, et al. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences*, 98(24):13790–13795, 2001.

Paul Blanche, Aurélien Latouche, and Vivian Viallon. Time-dependent auc with right-censored data: A survey. *Risk Assessment and Evaluation of Predictions*, 215:239–251, 2013.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Norman E Breslow. Contribution to the discussion of the paper by dr cox. *Journal of the Royal Statistical Society, Series B*, 34(2):216–217, 1972.

D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972.

R De Angelis, R Capocaccia, T Hakulinen, B Soderman, and A Verdecchia. Mixture models for cancer survival analysis: application to population-based data with covariates. *Statistics in medicine*, 18(4):441–454, 1999.

AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

VT Farewell. The use of mixture models for the analysis of sureval data with long-term survivors. *Biometrics*, 38(4):1041–1046, 1982.

Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537, 1999.
Frank E Harrell, Kerry L Lee, and Daniel B Mark. Tutorial in biostatistics multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in medicine*, 15:361–387, 1996.

Trevor Hastie, Robert Tibshirani, David Botstein, and Patrick Brown. Supervised harvesting of expression trees. *Genome Biology*, 2(1):research0003–1, 2001.

Patrick J Heagerty and Yingye Zheng. Survival model predictive accuracy and roc curves. *Biometrics*, 61(1):92–105, 2005.

Patrick J Heagerty, Thomas Lumley, and Margaret S Pepe. Time-dependent roc curves for censored survival data and a diagnostic marker. *Biometrics*, 56(2):337–344, 2000.

Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

John P Klein and Melvin L Moeschberger. *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media, 2005.

Anthony YC Kuk and Chen-Hsin Chen. A mixture model combining logistic regression with proportional hazards regression. *Biometrika*, 79(3):531–541, 1992.

Lynn Kuo and Fengchun Peng. A mixture-model approach to the analysis of survival data. *Biostatistics-Basel*, 5:255–272, 2000.

B. N. Mukherjee and S. S. Maiti. On some properties of positive definite toeplitz matrices and their possible applications. *Linear algebra and its applications*, 102:211–240, 1988.

Thordur Oskarsson, Swarnali Acharyya, Xiang HF Zhang, Sakari Vanharanta, Sohail F Tavazoie, Patrick G Morris, Robert J Downey, Katia Manova-Todorova, Edi Brogi, and Joan Massagué. Breast cancer cells produce tenascin c as a metastatic niche component to colonize the lungs. *Nature medicine*, 17(7):867–874, 2011.

Andreas Rosenwald, George Wright, Wing C Chan, Joseph M Connors, Elias Campo, Richard I Fisher, Randy D Gascoyne, H Konrad Muller-Hermelink, Erlend B Smeland, Jena M Giltnane, et al. The use of molecular profiling to predict survival after chemotherapy for diffuse large-b-cell lymphoma. *New England Journal of Medicine*, 346(25):1937–1947, 2002.

Margaret A Shipp, Ken N Ross, Pablo Tamayo, Andrew P Weng, Jeffery L Kutzok, Ricardo CT Aguiair, Michelle Gaasenbeek, Michael Angelo, Michael Reich, Geraldine S Pinkus, et al. Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature medicine*, 8(1):68–74, 2002.

Noah Simon, Jerome Friedman, Trevor Hastie, Rob Tibshirani, et al. Regularization paths for coxs proportional hazards model via coordinate descent. *Journal of statistical software*, 39(5):1–13, 2011.
Therese Sørlie, Charles M Perou, Robert Tibshirani, Turid Aas, Stephanie Geisler, Hilde Johnsen, Trevor Hastie, Michael B Eisen, Matt Van De Rijn, Stefanie S Jeffrey, et al. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, 98(19):10869–10874, 2001.

Robert Tibshirani, Trevor Hastie, Balasubramanian Narasimhan, and Gilbert Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10):6567–6572, 2002.

Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

Hajime Uno, Tianxi Cai, Michael J Pencina, Ralph B D’Agostino, and LJ Wei. On the c-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Statistics in medicine*, 30(10):1105–1117, 2011.

Laura J Van’t Veer, Hongyue Dai, Marc J Van De Vijver, Yudong D He, Augustinus AM Hart, Mao Mao, Hans L Peterse, Karin van der Kooy, Matthew J Marton, Anke T Witteveen, et al. Gene expression profiling predicts clinical outcome of breast cancer. *nature*, 415(6871):530–536, 2002.

CF Jeff Wu. On the convergence properties of the em algorithm. *The Annals of Statistics*, 11:95–103, 1983.

Yamei Zhou, Christina Yau, Joe W Gray, Karen Chew, Shanaz H Dairkee, Dan H Moore, Urs Eppenberger, Serenella Eppenberger-Castori, and Christopher C Benz. Enhanced nfκb and ap-1 transcriptional activity associated with anti-oestrogen resistant breast cancer. *BMC cancer*, 7(1):1, 2007.

Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.