Dimension reduction for integrative survival analysis

Aaron J. Molstad[†,*] and Rohit K. Patra[*]

Department of Statistics[*] and Genetics Institute[†]

University of Florida

Abstract

We propose a constrained maximum partial likelihood estimator for dimension reduction in integrative (e.g., pan-cancer) survival analysis with high-dimensional predictors. We assume that for each population in the study, the hazard function follows a distinct Cox proportional hazards model. To borrow information across populations, we assume that each of the hazard functions depend only on a small number of linear combinations of the predictors (i.e., "factors"). We estimate these linear combinations using an algorithm based on "distance-to-set" penalties. This allows us to impose both low-rankness and sparsity on the regression coefficient matrix estimator. We derive asymptotic results which reveal that our estimator is more efficient than fitting a separate proportional hazards model for each population. Numerical experiments suggest that our method outperforms competitors under various data generating models. We use our method to perform a pan-cancer survival analysis relating protein expression to survival across 18 distinct cancer types. Our approach identifies six linear combinations, depending on only 20 proteins, which explain survival across the cancer types. Finally, to validate our fitted model, we show that our estimated factors can lead to better prediction than competitors on four external datasets.

Keywords: Cox proportional hazards model, dimension reduction, integrative survival analysis, majorize-minimize, penalty method, reduced-rank regression, variable selection.

1 Introduction

Modeling a patient's survival time based on omic profiles (e.g., transcriptomic or proteomic) is a difficult, yet fundamentally important problem in cancer research. With the advent of personalized medicine, survival models help clinicians plan a course of treatment, and allow patients to make more informed decisions about their care. From a statistical perspective, the challenges are two-fold: first, omic data are often high-dimensional in the sense that the number of predictors (e.g., genes or proteins) can be much larger than the number of patients

included in the study (e.g., thousands of genes' expression measured on tens or hundreds of patients). Second, although numerous consortia have begun collecting multi-omic and clinical data on cancer patients (e.g., The Cancer Genome Atlas (TCGA) program), datasets often consist of relatively few patients and have high rates of censoring. Consequently, sample sizes are often too small to identify prognostic predictors or estimate potentially small effects with reasonable accuracy. To address the issues caused by high-dimensionality, small sample sizes, and high censoring rates, one approach is to analyze multiple independent datasets jointly, i.e., to perform an "integrative" survival analysis (Liu et al., 2014; Zhang et al., 2016; Maity et al., 2020). Integrative analyses can improve efficiency and can identify low-dimensional features which are shared across the multiple populations in the study.

In this article, we propose a new variable selection and dimension reduction method for such integrative survival analyses. Our methodological developments are motivated by a pan-cancer survival analysis in which we model survival as a function of patients' proteomic profiles jointly across 18 distinct cancer types. The proteome contributes to the complex pathophysiology of cancer in ways that cannot be explained by genomic or transcriptomic factors alone (Baladandayuthapani et al., 2014). Recent studies have demonstrated that protein expression levels can be discordant with DNA copy numbers and RNA expression levels (Shankavaram et al., 2007; Akbani et al., 2014), so using measurements of protein expression directly may provide novel insights regarding disease progression. More generally, pan-cancer survival models have been of recent interest because they may help identify therapeutic targets shared across cancer types (Wang et al., 2017; Maity et al., 2020).

To make matters concrete, suppose we are interested in modeling survival in J distinct cancer types. For each $j \in \{1, \ldots, J\}$, let $h_{*(j)}(t \mid \boldsymbol{x})$ be the hazard function for the jth cancer type evaluated at time t for a subject with observed protein expression $\boldsymbol{x} \in \mathbb{R}^p$. For any positive integer m, we will denote the set $\{1, 2, \ldots, m\}$ by $[m]$. We will model $h_{*(j)}(t \mid \boldsymbol{x})$ using the Cox proportional hazards model, i.e., we assume

$$h_{*(j)}(t \mid \boldsymbol{x}) = h_{*(j)0}(t) \exp(\boldsymbol{x}^\top \boldsymbol{b}_{*(j)}), \quad j \in [J], \tag{1}$$

where $h_{*(j)0}(t)$ and $\boldsymbol{b}_{*(j)} \in \mathbb{R}^p$ are the baseline hazard and regression coefficient vector for the jth cancer type, respectively. Let $\boldsymbol{B}_* = (\boldsymbol{b}_{*(1)}, \ldots, \boldsymbol{b}_{*(J)}) \in \mathbb{R}^{p \times J}$ be the matrix of unknown regression coefficients. Our proposed method assumes $\boldsymbol{B}_*$ is both low-rank and has many rows entirely zero, i.e., we assume the parametric restrictions

$$\mathrm{rank}(\boldsymbol{B}_*) \leq r_* \text{ for } r_* < \min\{p, J\}, \quad \text{and} \quad \|\boldsymbol{B}_*\|_{0,2} \leq s_* \text{ for } s_* \ll p, \tag{2}$$

where for a matrix $\boldsymbol{A}$ with lth row $\boldsymbol{A}_{l,\cdot}$, $\|\boldsymbol{A}\|_{0,2} = \sum_l \mathbf{1}(\|\boldsymbol{A}_{l,\cdot}\|_2 \neq 0)$ with $\mathbf{1}(\cdot)$ being the indicator function and $\|\boldsymbol{a}\|_2$ being the Euclidean norm of the vector $\boldsymbol{a}$. That is, for any matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\|_{0,2}$ is the number of nonzero rows of $\boldsymbol{A}$. Under the rank constraint in (2), we can decompose $\boldsymbol{B}_* = \boldsymbol{U}\boldsymbol{V}^\top$ with $\boldsymbol{U} \in \mathbb{R}^{p \times r_*}$ and $\boldsymbol{V} \in \mathbb{R}^{J \times r_*}$, so that we can write the linear predictor for a subject with the jth cancer type and protein expression $\boldsymbol{x} \in \mathbb{R}^p$ as

$$\boldsymbol{x}^\top \boldsymbol{b}_{*(j)} = (\boldsymbol{x}^\top \boldsymbol{u}_1)v_{(j)1} + (\boldsymbol{x}^\top \boldsymbol{u}_2)v_{(j)2} + \cdots + (\boldsymbol{x}^\top \boldsymbol{u}_{r_*})v_{(j)r_*}, \quad j \in [J],$$

where $\boldsymbol{u}_l \in \mathbb{R}^p$ is the lth column of $\boldsymbol{U}$ and $(v_{(j)1}, v_{(j)2}, \ldots, v_{(j)r_*})^\top \in \mathbb{R}^{r_*}$ is the jth row of $\boldsymbol{V}$. We may interpret the $\mathsf{f}_k(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{u}_k \in \mathbb{R}$ for $k \in [r_*]$ as an unobservable low-dimensional set of "factors" which contain all the useful information from $\boldsymbol{x}$ about survival in all J cancer types. In turn, we may then interpret the rows of $\boldsymbol{V}$ as the distinct regression coefficients for each of the cancer types in the space of the f_k. The $(0, 2)$-norm assumption on $\boldsymbol{B}_*$ (i.e., $\|\boldsymbol{B}_*\|_{0,2} \leq s_*$) implies that the same s_* elements of the vectors $\boldsymbol{u}_k$ can be nonzero. Letting $\mathcal{S}_* = \{l : \boldsymbol{B}_{*l,\cdot} \neq 0, l \in [p]\}$, this implies that the factors f_k depend only on the predictors indexed by $\mathcal{S}_*$, or stated in terms of the hazards, $h_{*(j)}(t \mid \boldsymbol{x}) = h_{*(j)}(t \mid \boldsymbol{x}_{\mathcal{S}_*})$ for all $j \in [J]$, where for any vector $\boldsymbol{a}$ and set $\mathcal{S} \subseteq [p]$, $\boldsymbol{a}_{\mathcal{S}}$ denotes the subvector of $\boldsymbol{a}$ containing only the elements indexed by $\mathcal{S}$. Together, these assumptions improve parsimony and interpretability. For example, the kth factor's biological relevance can be interpreted through the coefficients $\boldsymbol{u}_k$ for $k \in [r_*]$.

Our analysis of the motivating data in Section 6 provides strong evidence that (2) is justifiable in the context of pan-cancer proteomics and survival. In particular, cross-validation estimates (r_*, s_*) to be $(6, 20)$ (when $J = 18$ and $p = 210$). Many of these 20 selected proteins have been identified in the recent literature. Furthermore, examining the estimated factors (e.g. see Figure 2) reveals that they contain proteomic information which can distinguish cancer types (i.e., the factors do not contradict well-established molecular heterogeneity).

In the next section, we introduce a constrained maximum partial likelihood estimator for $\boldsymbol{B}_*$ under the parametric restriction (2) in high-dimensional settings. In subsequent sections, we propose a new algorithm to compute our estimator and establish asymptotic theory thereof. Our work provides contributions on multiple fronts. Methodologically, we propose a new framework for jointly modeling survival in distinct populations with high-dimensional predictors. In contrast to related methods, our framework allows practitioners to identify interpretable low-dimensional features shared across populations. Computationally, we provide a new procedure for fitting sparse reduced-rank regression models under nonconvex L_0-type constraints. Compared to existing algorithms for related problems, our algorithm

has closed-form updates and is conceptually simple as it is based on the majorize-minimize principle (Lange, 2016, Chapter 1). Theoretically, we find the asymptotic distribution of our estimator and quantify the gain in information from exploiting the rank constraint in (2). In contrast to classical results in reduced-rank regression (Anderson, 1999), we do not have a closed-form for the constrained maximum likelihood estimator of $\boldsymbol{B}_*$, so our proof technique is more general.

For the remainder, let $n_{(j)}$ denote the observed sample size for the jth population (e.g., jth cancer type), and let $t_{(j)1}, \ldots, t_{(j)n_{(j)}}$ denote the possibly unobserved survival times for the jth population. For each $j \in [J]$, we observe $(y_{(j)1}, \delta_{(j)1}, \boldsymbol{x}_{(j)1}), \ldots, (y_{(j)n_{(j)}}, \delta_{(j)n_{(j)}}, \boldsymbol{x}_{(j)n_{(j)}})$ where $y_{(j)i} = \min(t_{(j)i}, c_{(j)i})$, $c_{(j)i}$ is the censoring time for the ith subject in the jth population, $\delta_{(j)i} = \mathbf{1}(y_{(j)i} = t_{(j)i})$, and $\boldsymbol{x}_{(j)i} \in \mathbb{R}^p$ is the vector of predictors (e.g., protein expression) measured on the ith subject from the jth population. Let $\mathcal{R}_{(j)i} = \{k : y_{(j)k} \geq y_{(j)i}\}$ be jth population's risk set at time $y_{(j)i}$. Finally, let $\|\boldsymbol{A}\|_F^2 = \mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{A})$ be the squared Frobenius norm of a matrix $\boldsymbol{A}$. Throughout, we use upper-case bold letters to denote matrices and lower-case bold symbols to denote vectors.

2 Methodology

2.1 Estimation criterion

To fit the proportional hazards models (1) under the assumption (2), we maximize a constrained partial likelihood. Specifically, to allow for tied events, we propose to maximize a penalized and constrained version of the Breslow approximation of the partial likelihood. That is, the partial log-likelihood (approximation) we use is

$$\boldsymbol{\mathcal{L}}(\boldsymbol{B}) = \sum_{j=1}^{J} \sum_{i=1}^{n_{(j)}} \delta_{(j)i} \left[\boldsymbol{x}_{(j)i}^\top \boldsymbol{b}_{(j)} - \log \left\{ \sum_{k \in \mathcal{R}_{(j)i}} \exp\left(\boldsymbol{x}_{(j)k}^\top \boldsymbol{b}_{(j)} \right) \right\} \right].$$

We thus propose to estimate $\boldsymbol{B}_*$ using

$$\underset{\boldsymbol{B} \in \mathcal{C}_r \cap \mathcal{A}_s}{\arg\min} \left\{ -\boldsymbol{\mathcal{L}}(\boldsymbol{B}) + \mu \|\boldsymbol{B}\|_F^2 \right\} \tag{3}$$

where $\mathcal{C}_r = \{\boldsymbol{B} \in \mathbb{R}^{p \times J} : \mathrm{rank}(\boldsymbol{B}) \leq r\}$, $\mathcal{A}_s = \{\boldsymbol{B} \in \mathbb{R}^{p \times J} : \|\boldsymbol{B}\|_{0,2} \leq s\}$, and $\mu > 0$ is a small positive tuning parameter. The ridge penalty, controlled by μ, is used to impose a small degree of shrinkage so that we may consider an effective number of parameters larger than

the sample size. In both our simulations and real data analysis, we simply fix μ to be some small constant (rather than selecting it by cross-validation). The tuning parameters r and s, however, should be selected by cross-validation or an information criterion. In Web Appendix B, we propose using a version of the cross-validation criterion from Dai and Breheny (2019) to select tuning parameters.

Though the feasible set $\mathcal{C}_r \cap \mathcal{A}_s$ in (3) is nonconvex, we will later demonstrate that replacing it with a convex approximation leads to an estimator which is often very biased, difficult to tune, and arguably no easier to compute than (3). For more details, see Web Appendix A.1 and our comparison to such an estimator in Section 5.

2.2 Related methods and analyses

Dimension reduction methods for survival analysis have primarily focused on single population analyses, e.g., using sufficient dimension reduction (Li et al., 1999; Li and Li, 2004) or reduced-rank regression (Perperoglou et al., 2006; Fiocco et al., 2005). Perperoglou et al. (2006) used reduced-rank regression to estimate regression coefficients under the assumption of nonproportional hazards in a single population survival analysis. Fiocco et al. (2005) assumed a low-rank decomposition of the regression coefficient matrix under a Cox proportional hazards model for competing risks. Fiocco et al. (2005) require that multiple event times are measured on each subject from a single population, whereas we are focused on integrative survival analyses.

Numerous methods exist for the integration of multiple cancer datasets. Many of these methods focus on modeling continuous (uncensored) outcomes and the identification of shared nonzero regression coefficients (Zhao et al., 2015; Huang et al., 2017). Similar approaches for variable selection have been proposed under an accelerated failure time (AFT) model (Liu et al., 2014; Zhang et al., 2016). For example, Maity et al. (2020) developed a hierarchical Bayesian AFT model which performs variable selection and borrows information across populations through the correlation structure of the prior distributions. Extending these approaches to the Cox model is nontrivial owing to the computational and theoretical challenges of working with the constrained partial likelihood. Along these lines, Tang et al. (2019) proposed a method for fitting a Cox proportional hazards model when there exist predefined subgroups of subjects in a study. Their approach assumes that all subgroups have the same baseline hazard function and assumes that some regression coefficients are equivalent across pairs of subgroups. These assumptions are somewhat restrictive, but lead to an optimization problem that can be solved using existing algorithms and software.

The work most closely related to our own is the method proposed by Wang et al. (2017), who make a low-rank assumption on $\boldsymbol{B}_*$ and propose a nuclear norm penalized maximum partial likelihood estimator. This estimator can impose low-rankness and is the solution to a convex optimization problem, but does not perform variable selection, and thus can perform poorly in high-dimensional settings. Moreover, the nuclear norm penalty imposes global shrinkage towards the origin, so although one can obtain a low-rank estimate of $\boldsymbol{B}_*$, this often comes at the cost of excess shrinkage. We discuss the method of Wang et al. (2017) and propose a sparse variant thereof in Web Appendix A.1. As we will show in our simulation studies, this approach imposes substantial bias, has poor variable selection performance, and is too difficult to tune to be useful in practice.

Identifying shared prognostic factors across cancer types has also been of recent interest. For example, Hieronymus et al. (2018) found that tumor copy number alteration (CNA) burden was a significant prognostic factor in five distinct cancer types. For each $j \in [J]$, Hieronymus et al. (2018) effectively assumed $h_{*(j)}(t \mid c_{(j)i}) = h_{*(j)0}(t)\exp(c_{(j)i}v_{(j)})$ where $c_{(j)i} \in \mathbb{R}$ is the CNA burden for the ith subject with the jth cancer type and $v_{(j)} \in \mathbb{R}$ is an unknown regression coefficient for the jth cancer type. However, CNA burden is simply the proportion of the genome affected by CNAs, so we can express $c_{(j)i} = \boldsymbol{x}_{(j)i}^\top 1_p/p$ where $\boldsymbol{x}_{(j)i} \in \{0,1\}^p$ is a vector whose kth component equals one if there is a CNA at the kth genomic location and zero otherwise for each $k \in [p]$, and $1_p \in \mathbb{R}^p$ is the p-dimensional vector of ones. If instead, we replaced $1_p/p$ with $\boldsymbol{u} \in \mathbb{R}^p$, their hazard function for the jth cancer type would be $h_{*(j)0}(t)\exp\{(\boldsymbol{x}_{(j)i}^\top \boldsymbol{u})v_{(j)}\}$, which corresponds exactly to the rank one version of our model, i.e., $\boldsymbol{B}_* = \boldsymbol{u}\boldsymbol{v}^\top$ where $\boldsymbol{v} = (v_{(1)}, \ldots, v_{(J)})^\top \in \mathbb{R}^J$. Thus with (3), a practitioner could instead estimate both $\boldsymbol{u}$ and $\boldsymbol{v}$ simultaneously, which would allow for a more flexible factor than CNA burden to be discovered. For example, our method could identify genomic locations at which the CNAs are relevant to survival in multiple cancers by taking a (sparse) weighted sum of CNAs rather than simply taking the proportion of the genome affected by CNAs. In a separate study, Hong et al. (2020) found that genes among the CASP family serve as useful prognostic factors for breast cancer, hepatocellular carcinoma, and pancreatic cancer. Under (2), this would correspond to taking the $\boldsymbol{x}_{(j)i}$ to be gene expression profiles and assuming $r_* = 1$ with $\boldsymbol{B}_* = \boldsymbol{u}\boldsymbol{v}^\top$, but constraining $\boldsymbol{u}$ to be a vector which has zeros in all positions except those corresponding to genes in the CASP family. Hence, (3) could also discover this type of prognostic factor, but without requiring prior specification of a particular gene family.

Finally, we note that the assumptions in (2) are similar to those made in sparse reduced-

rank multivariate response regression (Chen and Huang, 2012; She, 2017). The methods of Chen and Huang (2012) and She (2017) are designed for settings where multiple (uncensored) responses are measured on each subject from a single population. As such, these methods cannot be applied in the context of integrative survival analysis. We further elaborate on the distinction between our method and existing methods for sparse reduced-rank regression in Web Appendix H.

3   Computation

3.1   Penalty method based on "distance-to-set" penalties

To compute (3), we use the penalty method (Nocedal and Wright, 2006, Chapter 17) in concert with the majorize-minimize principle (Lange, 2016, Chapter 1). The penalty method is especially appealing for (3) since we can employ "distance-to-set" penalties, which can be majorized by smooth functions (Xu et al., 2017; Keys et al., 2019). In particular, to compute (3), we solve a sequence of problems of the form

$$\widetilde{\boldsymbol{B}}_\rho = \arg\min_{\boldsymbol{B}\in\mathbb{R}^{p\times J}} \left\{ -\mathcal{L}(\boldsymbol{B}) + \mu\|\boldsymbol{B}\|_F^2 + \frac{\rho}{2}\mathrm{dist}(\boldsymbol{B},\mathcal{C}_r)^2 + \frac{\rho}{2}\mathrm{dist}(\boldsymbol{B},\mathcal{A}_s)^2 \right\}, \tag{4}$$

where $\mathrm{dist}(\boldsymbol{B},\mathcal{C}_r)$ is the Euclidean distance from $\boldsymbol{B}$ and its nearest point in the set $\mathcal{C}_r$, i.e.,

$$\mathrm{dist}(\boldsymbol{B},\mathcal{C}_r) = \inf_{\boldsymbol{A}\in\mathcal{C}_r} \|\boldsymbol{B}-\boldsymbol{A}\|_F = \|\boldsymbol{B}-\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B})\|_F,$$

with $\mathbf{P}_{\mathcal{C}_r}$ denoting the projection onto $\mathcal{C}_r$. By taking the penalty parameter $\rho \to \infty$, it will occur that $\widetilde{\boldsymbol{B}}_\rho \in \mathcal{C}_r \cap \mathcal{A}_s$ as any iterate outside the set $\mathcal{C}_r \cap \mathcal{A}_s$ will lead to a large objective function value. To use this approach in practice, we iteratively compute $\widetilde{\boldsymbol{B}}_\rho$ for an increasing sequence of ρ values – initializing the algorithm for $\widetilde{\boldsymbol{B}}_\rho$ at the solution for the previous (smaller) value of ρ. In the next subsection, Section 3.2, we propose a majorize-minimize algorithm for (4) with ρ fixed. In the following subsection, Section 3.3, we detail our implementation of the penalty method for computing (3) more broadly.

3.2   Majorize-minimize algorithm for computing $\widetilde{\boldsymbol{B}}_\rho$

To compute (4), we appeal to the majorize-minimize principle. Following Xu et al. (2017), given current (kth) iterate $\boldsymbol{B}^k$, we first majorize both distance penalties using a variation of

$$\mathcal{M}_{\mathcal{C}_r}(\boldsymbol{B} \mid \boldsymbol{B}^k) = \|\boldsymbol{B} - \mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)\|_F^2.$$

The function $\mathcal{M}_{\mathcal{C}_r}(\cdot \mid \boldsymbol{B}^k)$ majorizes $\mathrm{dist}(\cdot, \mathcal{C}_r)^2$ at $\boldsymbol{B}^k$ because

$$\mathrm{dist}(\boldsymbol{B}^k, \mathcal{C}_r)^2 = \mathcal{M}_{\mathcal{C}_r}(\boldsymbol{B}^k \mid \boldsymbol{B}^k) \quad \text{and} \quad \mathcal{M}_{\mathcal{C}_r}(\boldsymbol{B} \mid \boldsymbol{B}^k) \geq \mathrm{dist}(\boldsymbol{B}, \mathcal{C}_r)^2 \tag{5}$$

for all $\boldsymbol{B} \in \mathbb{R}^{p \times J}$. Moreover, $\nabla \mathcal{M}_{\mathcal{C}_r}(\boldsymbol{B} \mid \boldsymbol{B}^k) = 2\{\boldsymbol{B} - \mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)\}$ when $\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)$ is single valued. An analogous majorizer can be constructed for $\mathrm{dist}(\cdot, \mathcal{A}_s)^2$. Because both $\mathcal{C}_r$ and $\mathcal{A}_s$ are closed, projections onto each are single valued except on a set of matrices with Lebesgue measure zero (Keys et al., 2019, Proposition 6). In addition, both projections can be computed efficiently. Letting $\boldsymbol{L}_r^k \in \mathbb{R}^{p \times r}$ and $\boldsymbol{R}_r^k \in \mathbb{R}^{J \times r}$ denote the leading r left and right singular vectors of $\boldsymbol{B}^k$, respectively, and letting $\boldsymbol{D}_r^k \in \mathbb{R}^{r \times r}$ be a diagonal matrix with $\boldsymbol{B}^k$'s largest r singular values along its diagonal, $\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k) = \boldsymbol{L}_r^k \boldsymbol{D}_r^k \boldsymbol{R}_r^{k\top}$. Similarly, the projection of $\boldsymbol{B}^k$ onto $\mathcal{A}_s$ simply requires setting the rows with $(s+1)$th through pth largest Euclidean norms to zero. That is, $\mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k) = \mathbb{A}_s(\boldsymbol{B}^k)\boldsymbol{B}^k$ where $\mathbb{A}_s(\boldsymbol{B}^k) \in \mathbb{R}^{p \times p}$ is a diagonal matrix with (l,l)th element equal to one if $\|\boldsymbol{B}_{l,\cdot}^k\|_2$ is one of the s largest for all $l \in [p]$ and zero otherwise.

Letting $\mathcal{F}_\rho$ denote the objective function from (4), we can majorize $\mathcal{F}_\rho$ at $\boldsymbol{B}^k$ using that

$$\mathcal{F}_\rho(\boldsymbol{B}) \leq -\boldsymbol{\mathcal{L}}(\boldsymbol{B}) + \mu\|\boldsymbol{B}\|_F^2 + \frac{\rho}{2}\|\boldsymbol{B} - \mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)\|_F^2 + \frac{\rho}{2}\|\boldsymbol{B} - \mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k)\|_F^2 \tag{6}$$

for all $\boldsymbol{B} \in \mathbb{R}^{p \times J}$. Thus if $\boldsymbol{B}^{k+1}$ minimizes the right hand side of (6), i.e., is defined as

$$\underset{\boldsymbol{B} \in \mathbb{R}^{p \times J}}{\arg\min} \left\{ -\boldsymbol{\mathcal{L}}(\boldsymbol{B}) + \mu\|\boldsymbol{B}\|_F^2 + \frac{\rho}{2}\|\boldsymbol{B} - \mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)\|_F^2 + \frac{\rho}{2}\|\boldsymbol{B} - \mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k)\|_F^2 \right\}, \tag{7}$$

then we would be ensured that $\mathcal{F}_\rho(\boldsymbol{B}^{k+1}) \leq \mathcal{F}_\rho(\boldsymbol{B}^k)$ by (5). However, computing (7) is itself a challenging optimization problem. Although (7) can be solved column-by-column of $\boldsymbol{B}$, each column-wise update would require its own iterative algorithm to compute. To avoid solving this problem at each iteration, we can instead employ a further (approximate)

majorization of $-\mathcal{L}(\boldsymbol{B})$, which can be minimized efficiently. Let

$$\ell_{(j)}(\boldsymbol{b}_{(j)}) = \sum_{i=1}^{n_{(j)}} \delta_{(j)i} \left[\boldsymbol{x}_{(j)i}^\top \boldsymbol{b}_{(j)} - \log \left\{ \sum_{k \in \mathcal{R}_{(j)i}} \exp \left(\boldsymbol{x}_{(j)k}^\top \boldsymbol{b}_{(j)} \right) \right\} \right]$$

denote the jth population's contribution to the partial log-likelihood for $j \in [J]$. Following Simon et al. (2011), we use a quadratic approximation of $-\ell_{(j)}(\boldsymbol{b}_{(j)})$. Let $\boldsymbol{\eta}_{(j)}^k = \boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)}^k$ where $\boldsymbol{X}_{(j)} = (\boldsymbol{x}_{(j)1}, \dots, \boldsymbol{x}_{(j)n_{(j)}})^\top \in \mathbb{R}^{n_{(j)} \times p}$. For square matrix $\boldsymbol{A}$, let $\mathrm{Diag}(\boldsymbol{A})$ be a matrix with (j, k)th entry equal to $\boldsymbol{A}_{j,k}$ if $j = k$ and zero otherwise. Then, letting $f_{(j)}$ denote the function $\boldsymbol{\eta}_{(j)} \mapsto -\ell_{(j)}(\boldsymbol{b}_{(j)})$, and letting $\nabla f_{(j)}$ and $\nabla^2 f_{(j)}$ denote the gradient and Hessian of $f_{(j)}$, we have that

$$\begin{aligned} -\ell_{(j)}(\boldsymbol{b}_{(j)}) \approx\ & -\ell_{(j)}(\boldsymbol{b}_{(j)}^k) + \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right)^\top \nabla f_{(j)}(\boldsymbol{\eta}_{(j)}^k) \\ & + \frac{1}{2} \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right)^\top \mathrm{Diag} \left[\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k) \right] \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right). \end{aligned}$$

Hence, letting

$$\begin{aligned} g_{(j)}(\boldsymbol{b}_{(j)} \mid \boldsymbol{b}_{(j)}^k) =\ & -\ell_{(j)}(\boldsymbol{b}_{(j)}^k) + \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right)^\top \nabla f_{(j)}(\boldsymbol{\eta}_{(j)}^k) \\ & + \frac{1}{2} \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right)^\top \mathrm{Diag} \left[\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k) \right] \left(\boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)} - \boldsymbol{\eta}_{(j)}^k \right) + \frac{\mu}{2} \boldsymbol{b}_{(j)}^\top \boldsymbol{b}_{(j)}, \end{aligned}$$

and letting $[\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)]_{\cdot,j}$ denote the jth column of the matrix $\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)$, we define

$$\boldsymbol{b}_{(j)}^{k+1} = \operatorname*{arg\,min}_{\boldsymbol{b}_{(j)} \in \mathbb{R}^p} \left\{ g_{(j)}(\boldsymbol{b}_{(j)} \mid \boldsymbol{b}_{(j)}^k) + \frac{\rho}{2} \| \boldsymbol{b}_{(j)} - [\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k)]_{\cdot,j} \|_2^2 + \frac{\rho}{2} \| \boldsymbol{b}_{(j)} - [\mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k)]_{\cdot,j} \|_2^2 \right\} \quad (8)$$

for $j \in [J]$, and define $\boldsymbol{B}^{k+1} = (\boldsymbol{b}_{(1)}^{k+1}, \dots, \boldsymbol{b}_{(J)}^{k+1})$. Minimizing this particular approximation of $\mathcal{F}_\rho$ is efficient because each $\boldsymbol{b}_{(j)}^{k+1}$ can be computed in parallel and in closed-form. Letting

$$\boldsymbol{Z}(\boldsymbol{\eta}_{(j)}^k) = \boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)}^k - \mathrm{Diag}\{\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k)\}^{-1} \nabla f_{(j)}(\boldsymbol{\eta}_{(j)}^k), \quad \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) = \mathrm{Diag}\{\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k)\},$$

and $\lambda = 2\rho + \mu$, we have that

$$\boldsymbol{b}_{(j)}^{k+1} = \left\{ \boldsymbol{X}_{(j)}^\top \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) \boldsymbol{X}_{(j)} + \lambda I_p \right\}^{-1} \left\{ \boldsymbol{X}_{(j)}^\top \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) \boldsymbol{Z}(\boldsymbol{\eta}_{(j)}^k) + \rho [\mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k) + \mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k)]_{\cdot,j} \right\}. \tag{9}$$

Computing (9) may be time consuming when $p > n_{(j)}$. In this case, we may use the Woodbury identity, i.e., $\{ \boldsymbol{X}_{(j)}^\top \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) \boldsymbol{X}_{(j)} + \lambda I_p \}^{-1} = \lambda^{-2} [\lambda I_p - \boldsymbol{X}_{(j)}^\top \{ \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k)^{-1} + \lambda^{-1} \boldsymbol{X}_{(j)} \boldsymbol{X}_{(j)}^\top \}^{-1} \boldsymbol{X}_{(j)}]$,

---

Algorithm 1 Majorize-minimize algorithm for (4)

Inputs: $(\mu, r, s, k^{\max}, \boldsymbol{B}^{\text{init}}, \rho) \in (0, \infty) \times [\min(p, J)] \times [p] \times \mathbb{N} \times \mathbb{R}^{p \times J} \times (0, \infty)$

1. Set $\boldsymbol{B}^0 = \boldsymbol{B}^{\text{init}}$ and $k = 0$

2. Compute $\breve{\boldsymbol{B}} = \mathbf{P}_{\mathcal{C}_r}(\boldsymbol{B}^k) + \mathbf{P}_{\mathcal{A}_s}(\boldsymbol{B}^k)$

3. For $j \in [J]$ in parallel

   (a) Update $\boldsymbol{\eta}_{(j)}^k = \boldsymbol{X}_{(j)} \boldsymbol{b}_{(j)}^k$

   (b) Update $\boldsymbol{b}_{(j)}^{k+1} = \left\{ \boldsymbol{X}_{(j)}^\top \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) \boldsymbol{X}_{(j)} + (2\rho + \mu) I_p \right\}^{-1} \left\{ \boldsymbol{X}_{(j)}^\top \boldsymbol{W}(\boldsymbol{\eta}_{(j)}^k) \boldsymbol{Z}(\boldsymbol{\eta}_{(j)}^k) + \rho \breve{\boldsymbol{b}}_{(j)} \right\}$

4. If the objective function value has not converged and $k \leq k^{\max}$, set $k = k + 1$ and return to (2); otherwise, terminate

---

so that computing $\boldsymbol{b}_{(j)}^{k+1}$ requires inverting only an $n_{(j)} \times n_{(j)}$ matrix.

We summarize the algorithm for computing $\widetilde{\boldsymbol{B}}_\rho$ (with ρ fixed) in Algorithm 1 and embed this within the penalty method for computing (3) in Algorithm 2. Note that our approach is not strictly adhering to the majorize-minimize principle. The objective function constructed from summing the terms in (8) over all j is not, strictly speaking, a majorizing function of (4). To ensure that $\boldsymbol{B}^{k+1} = (\boldsymbol{b}_{(1)}^{k+1}, \ldots, \boldsymbol{b}_{(J)}^{k+1})$ yields a decrement of the original objective function $\mathcal{F}_\rho$, one could replace each $\text{Diag}\{\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k)\}$ with $\phi I_{n_{(j)}}$ for some sufficiently large positive constant ϕ. In practice, we found that replacing $\text{Diag}\{\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k)\}$ is not necessary. Updates based on (8) led to a decrease in the objective function in every scenario we considered, and led to substantially faster convergence than any other approach we tried. However, this modified version of our algorithm is amenable to convergence analysis.

Proposition 1 *Let* $\boldsymbol{B}^{k+1} = (\boldsymbol{b}_{(1)}^{k+1}, \ldots, \boldsymbol{b}_{(J)}^{k+1})$ *be defined as in* (8) *with* $\phi I_{n_{(j)}}$ *replacing* $\text{Diag}\{\nabla^2 f_{(j)}(\boldsymbol{\eta}_{(j)}^k)\}$ *for some* $\phi > 0$ *sufficiently large and fixed. Then,* $\mathcal{F}_\rho(\boldsymbol{B}^{k+1}) \leq \mathcal{F}_\rho(\boldsymbol{B}^k)$ *for* $k = 1, 2, 3, \ldots$, $\lim_{k \to \infty} \|\boldsymbol{B}^{k+1} - \boldsymbol{B}^k\|_F = 0$, *and if each limit point of* $\{\boldsymbol{B}^k\}_{k=1}^\infty$ *is isolated, then the iterates converge to a critical point of* $\mathcal{F}_\rho$.

Proposition 1 – which is a direct application of Proposition 9 of Keys et al. (2019) – follows, in part, from the majorize-minimize principle and the strong convexity of $g_{(j)}(\cdot \mid \boldsymbol{b}_{(j)}^k)$.

3.3 Implementation details

To use the penalty method, we must apply Algorithm 1 to repeatedly solve (4) for an increasing sequence of penalty parameters ρ. Starting with a relatively small initial value of

Algorithm 2 Penalty method for computing (3)

Inputs: $(\mu, r, s, k^{\max}, \delta_{\mathrm{incr}}, \rho_0, \epsilon) \in (0, \infty) \times [\min(p, J)] \times [p] \times \mathbb{N} \times (1, \infty) \times (0, \infty) \times (0, \infty)$

1. Initialize $\widetilde{\boldsymbol{B}} \in \mathbb{R}^{p \times J}$ and set $\rho_{\mathrm{old}} = \rho_0$

2. Compute $\widetilde{\boldsymbol{B}}_{\rho_{\mathrm{new}}}$, the final iterate of Algorithm 1 with inputs $(\mu, r, s, k^{\max}, \widetilde{\boldsymbol{B}}, \rho_{\mathrm{old}})$

3. Set $\widetilde{\boldsymbol{B}}_{\rho_{\mathrm{old}}} = \widetilde{\boldsymbol{B}}_{\rho_{\mathrm{new}}}$ and set $\rho_{\mathrm{new}} = \delta_{\mathrm{incr}}\rho_{\mathrm{old}}$

4. Compute $\widetilde{\boldsymbol{B}}_{\rho_{\mathrm{new}}}$, the final iterate of Algorithm 1 with inputs $(\mu, r, s, k^{\max}, \widetilde{\boldsymbol{B}}_{\rho_{\mathrm{old}}}, \rho_{\mathrm{new}})$

5. If $\max\{\mathrm{dist}(\widetilde{\boldsymbol{B}}_{\rho_{\mathrm{new}}}, \mathcal{C}_r)^2, \mathrm{dist}(\widetilde{\boldsymbol{B}}_{\rho_{\mathrm{new}}}, \mathcal{A}_s)^2\} < \epsilon$, terminate; otherwise, set $\rho_{\mathrm{old}} = \rho_{\mathrm{new}}$ then return to (3)

$\rho > 0$, we compute $\widetilde{\boldsymbol{B}}_\rho$ using Algorithm 1. We then increase ρ by a fixed multiplicative factor, say 1.2, and compute $\widetilde{\boldsymbol{B}}_{1.2\rho}$ after initializing the algorithm at $\widetilde{\boldsymbol{B}}_\rho$. We repeat this procedure until for some sufficiently large ρ and corresponding $\widetilde{\boldsymbol{B}}_\rho$, we have both $\mathrm{dist}(\widetilde{\boldsymbol{B}}_\rho, \mathcal{C}_r)^2 < \epsilon$ and $\mathrm{dist}(\widetilde{\boldsymbol{B}}_\rho, \mathcal{A}_s)^2 < \epsilon$ for some convergence tolerance $\epsilon > 0$. We call the limit of this sequence $\widehat{\boldsymbol{B}}$, which can be thought of as (3) for a fixed r, s, and μ.

Following Keys et al. (2019), we find that it is not necessary to solve (4) exactly for each penalty parameter ρ. Instead, we can run Algorithm 1 for a fixed number of iterations (e.g., we set $k^{\max} = 10$ in our implementation) to approximate (4) before incrementing ρ. Often, 10 iterations was enough for convergence for even moderately sized ρ, e.g., $\rho > (1.2)^5\rho_0$, where ρ_0 is the initial penalty parameter value. The number of allowed iterations, ρ_0, and by what factor ρ is increased at each iteration of the penalty method can be diagnosed by the user, but we found our default implementation to work well across a range of scenarios. Software implementing this method is available for download from GitHub (Molstad and Patra, 2022).

In addition, we found that using "warm-starting" could also improve estimation accuracy and convergence speed. Suppose that we consider models defined by $(r, s) \in \{r_1, \ldots, r_L\} \times \{s_1, \ldots, s_M\}$ where $r_1 > r_2 > \cdots > r_L$. Treating $s = s_i$ as fixed, we start by computing $\widehat{\boldsymbol{B}}$ with $r = r_1$ after initializing at the matrix of zeros. Then, for computing $\widehat{\boldsymbol{B}}$ with $r = r_k$, we initialize the algorithm at the solution for $r = r_{k-1}$ for $k \in \{2, \ldots, L\}$. We repeat this separately for each candidate s.

4 Asymptotic properties

Next, we establish the asymptotic distribution of a version of our estimator. We focus on quantifying the efficiency gained when using the rank constraint by studying (3) with $s = p$ and $\mu = 0$. We treat p as fixed throughout this section. To simplify notation, let $\widehat{\boldsymbol{B}}_r$ denote the rank r constrained version of (3), i.e., define $\widehat{\boldsymbol{B}}_r = \arg\max_{\boldsymbol{B} \in \mathcal{C}_r \cap \mathcal{D}_M} \mathcal{L}(\boldsymbol{B})$ where $\mathcal{D}_M = \{\boldsymbol{B} \in \mathbb{R}^{p \times J} : \|\boldsymbol{B}\|_F \leq M\}$ for large constant M. The constraint that $\widehat{\boldsymbol{B}}_r \in \mathcal{D}_M$ serves to regularize the estimator in a manner similar to the ridge penalty. However, we omit notation indicating dependence on M since, for the remainder of this section, we assume M is taken to be a arbitrarily large constant such that $\|\boldsymbol{B}_*\|_F \ll M < \infty$.

In the following, we establish asymptotic normality, derive the asymptotic variance, and verify that the covariance of each column of the standard maximum partial likelihood estimator minus that of our estimator is positive semidefinite. Together, these results provide a theoretical justification of the efficiency gains observed in our simulation studies in Section 5. Throughout, let $\boldsymbol{A}^+$ be the Moore-Penrose pseudoinverse of a matrix $\boldsymbol{A}$, and let $\boldsymbol{A}_1 \otimes \boldsymbol{A}_2$ be the Kronecker product of matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$. Let $\text{BlockDiag}[\{\boldsymbol{A}_k\}_{k=1}^{K}]$ be the block diagonal matrix with kth diagonal block $\boldsymbol{A}_k$ for $k \in [K]$, and let $\text{vec}(\boldsymbol{A})$ be the vector formed by stacking the columns of the matrix $\boldsymbol{A}$. Finally, we use the notation $\boldsymbol{A}_1 \succeq \boldsymbol{A}_2$ to indicate that $\boldsymbol{A}_1 - \boldsymbol{A}_2$ is positive semidefinite.

Our asymptotic results will require a number of assumptions. First, we assume that $\text{rank}(\boldsymbol{B}_*) = r_*$ with $r_* \leq \min\{p, J\}$, and assume that $r = r_*$ (i.e., the rank of (3) is correctly specified). Similarly, letting $n = \sum_{j=1}^{J} n_{(j)}$, we assume that for all $j \in [J]$, the ratio $n_{(j)}/n \to \kappa_{(j)}$ for some constant $\kappa_{(j)} > 0$ as $n \to \infty$. The remainder of our assumptions apply to each population separately: these are the standard assumptions needed for asymptotic normality under the Cox proportional hazards model with right censoring, e.g., see Chapter 6 of Van der Vaart (2002) or Hjort and Pollard (2011). Notably, we do not require anything about the relationships between populations other than the low-rankness of $\boldsymbol{B}_*$. For the sake of space, we formally state and discuss these assumptions, (A1)–(A7), in Web Appendix D.

Before stating our main result, we first note that when $r = \min\{p, J\}$, $\widehat{\boldsymbol{B}}_r$ is equivalent to J separate maximum partial likelihood estimators: one for each column of $\boldsymbol{B}_*$. We denote this estimator $\bar{\boldsymbol{B}}$ and define $\bar{\boldsymbol{b}}_{(j)} \in \mathbb{R}^p$ as the jth column of $\bar{\boldsymbol{B}}$ for $j \in [J]$. Applying standard asymptotic results for the Cox proportional hazards model, we know that as $n \to \infty$ under our assumptions, $\sqrt{n_{(j)}}(\bar{\boldsymbol{b}}_{(j)} - \boldsymbol{b}_{*(j)}) \xrightarrow{d} \text{N}_p(0, \boldsymbol{D}_{*(j)}^{-1})$ for $j \in [J]$, where the exact form of $\boldsymbol{D}_{*(j)}$ is given in Lemma E.2 of Web Appendix E. As we will show, the asymptotic distribution of $\widehat{\boldsymbol{B}}_{r_*}$ will depend on the $\boldsymbol{D}_{*(j)}$. However, unlike the unconstrained maximum partial

likelihood estimator $\bar{\boldsymbol{B}}$, all $\boldsymbol{D}_{*(j)}$ affect the asymptotic covariance of each column of $\widehat{\boldsymbol{B}}_{r_*}$. The following result, proved in Web Appendix D, establishes the asymptotic distribution of the rank-constrained maximum partial likelihood estimator.

Theorem 1 *Let $\boldsymbol{U} \in \mathbb{R}^{p \times r_*}$ and $\boldsymbol{V} \in \mathbb{R}^{J \times r_*}$ be any pair of rank r_* matrices such that $\boldsymbol{B}_* = \boldsymbol{U}\boldsymbol{V}^\top$. Define the matrices $\boldsymbol{T} = [\boldsymbol{V} \otimes I_p, I_J \otimes \boldsymbol{U}]$ and $\boldsymbol{D}_* = \mathrm{BlockDiag}\left[\{\kappa_{(j)}\boldsymbol{D}_{*(j)}\}_{j=1}^J\right].$ Then under assumptions (A1)–(A7),*

$$\sqrt{n}\{\mathrm{vec}(\widehat{\boldsymbol{B}}_{r_*} - \boldsymbol{B}_*)\} \xrightarrow{d} \mathrm{N}_{Jp}\left(0, \boldsymbol{T}(\boldsymbol{T}^\top \boldsymbol{D}_* \boldsymbol{T})^+ \boldsymbol{T}^\top\right).$$

Before discussing the result, we comment briefly on the proof. There are two fundamental differences between our proof and those of existing asymptotics for reduced-rank regression estimators. Firstly, in contrast to the classical reduced-rank regression asymptotics (Stoica and Viberg, 1996; Anderson, 1999), we do not have closed-form expressions for estimates of a particular $\boldsymbol{U}$ and $\boldsymbol{V}$. Secondly, we are not dealing with a quadratic objective function, so many other standard techniques could not be applied. Instead, we prove Theorem 1 by first defining a particular identifiable decomposition of $\boldsymbol{B}_*$, showing that we estimate the components of this particular decomposition consistently, using this to establish the asymptotic normality of a function of these components, and finally showing that these results hold for any decomposition of $\boldsymbol{B}_*$.

To better understand the implication of Theorem 1, we express the asymptotic covariance in terms of any pair $(\boldsymbol{U}, \boldsymbol{V})$ where $\boldsymbol{B}_* = \boldsymbol{U}\boldsymbol{V}^\top$. Defining $\widetilde{\boldsymbol{D}}_{*(j)} = \kappa_{(j)}\boldsymbol{D}_{*(j)}$ for all $j \in [J]$, $\boldsymbol{P}_{\boldsymbol{A},\boldsymbol{B}} = \boldsymbol{A}(\boldsymbol{A}^\top \boldsymbol{B} \boldsymbol{A})^{-1}\boldsymbol{A}^\top \boldsymbol{B}$, and $\boldsymbol{P}_{\boldsymbol{A},\boldsymbol{B}}^\perp = I - \boldsymbol{P}_{\boldsymbol{A},\boldsymbol{B}}$ for any projection matrix $\boldsymbol{P}_{\boldsymbol{A},\boldsymbol{B}}$, we have that asymptotic covariance from Theorem 1 can be expressed

$$\boldsymbol{T}(\boldsymbol{T}^\top \boldsymbol{D}_* \boldsymbol{T})^+ \boldsymbol{T}^\top = \mathrm{BlockDiag}\left[\{\boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(j)}}\widetilde{\boldsymbol{D}}_{*(j)}^{-1}\}_{j=1}^J\right]$$

$$+ \begin{pmatrix} \boldsymbol{V}_{1,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(1)}}^\perp \\ \boldsymbol{V}_{2,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(2)}}^\perp \\ \vdots \\ \boldsymbol{V}_{J,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(J)}}^\perp \end{pmatrix} \left(\sum_{j=1}^J \boldsymbol{V}_{j,\cdot}\boldsymbol{V}_{j,\cdot}^\top \otimes \widetilde{\boldsymbol{D}}_{*(j)}\boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(j)}}^\perp\right)^+ \begin{pmatrix} \boldsymbol{V}_{1,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(1)}}^\perp \\ \boldsymbol{V}_{2,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(2)}}^\perp \\ \vdots \\ \boldsymbol{V}_{J,\cdot}^\top \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(J)}}^\perp \end{pmatrix}^\top$$

where $\boldsymbol{V}_{j,\cdot} \in \mathbb{R}^r$ denotes the jth row of $\boldsymbol{V}$. That is, the asymptotic covariance consists of the sum of two matrices: a block diagonal matrix whose components depend only on the $\widetilde{\boldsymbol{D}}_{*(j)}$ and the column space of $\boldsymbol{U}$, and a matrix which depends on the $\widetilde{\boldsymbol{D}}_{*(j)}$ and the column spaces of $\boldsymbol{U}$ and $\boldsymbol{V}$. The covariance between columns of $\sqrt{n}(\widehat{\boldsymbol{B}}_{r_*} - \boldsymbol{B}_*)$ comes only from the latter

13

matrix. For example, the covariance between the lth and mth columns is

$$\left(\boldsymbol{V}_{l,\cdot}^{\top} \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(l)}}^{\perp}\right) \left(\sum_{j=1}^{J} \boldsymbol{V}_{j,\cdot} \boldsymbol{V}_{j,\cdot}^{\top} \otimes \widetilde{\boldsymbol{D}}_{*(j)} \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(j)}}^{\perp}\right)^{+} \left(\boldsymbol{V}_{m,\cdot}^{\top} \otimes \boldsymbol{P}_{\boldsymbol{U},\widetilde{\boldsymbol{D}}_{*(m)}}^{\perp}\right)^{\top}.$$

Note that the asymptotic covariance of the maximum partial likelihood estimator is rank deficient. It can be checked that $\mathrm{rank}[\mathrm{avar}\{\mathrm{vec}(\widehat{\boldsymbol{B}}_{r_*} - \boldsymbol{B}_*)\}] \leq (J+p-r_*)r_*$, where $(J+p-r_*)r_*$ is the number of identifiable parameters in $\boldsymbol{B}_*$ when $\mathrm{rank}(\boldsymbol{B}_*) = r_*$ and avar denotes the asymptotic covariance.

Finally, we can use the result of Theorem 1 to verify that the rank restriction leads to an efficiency gain over separate maximum partial likelihood estimators.

Theorem 2 *Under the conditions of Theorem 1,*

$$\mathrm{avar}\{\sqrt{n}(\bar{\boldsymbol{b}}_{(j)} - \boldsymbol{b}_{*(j)})\} \succeq \mathrm{avar}\{\sqrt{n}(\widehat{\boldsymbol{b}}_{(j)r_*} - \boldsymbol{b}_{*(j)})\} \quad \text{for all} \ \ j \in [J],$$

where $\widehat{\boldsymbol{b}}_{(j)r_*}$ is the jth column of $\widehat{\boldsymbol{B}}_{r_*}$.

Theorem 2 reveals that each column of the rank-constrained estimator has covariance no greater than the unconstrained maximum partial likelihood estimator in the sense that their difference is negative semidefinite. The result of Theorem 2 suggests that the improvements in estimation accuracy, which we observe empirically in Section 5, are the result of lower variance coming from the rank restriction.

5 Simulation studies

5.1 Data generating model

We compare our method to various competitors under the assumption in (2). Specifically, for one hundred independent replications, we generate survival times under the Cox proportional hazards models for $J = 12$ distinct populations. In each setting, we generate $n_{(1)} = n_{(4)} = n_{(7)} = n_{(10)} = 100$, $n_{(2)} = n_{(5)} = n_{(8)} = n_{(11)} = 200$, and $n_{(3)} = n_{(6)} = n_{(9)} = n_{(12)} = 300$ independent survival times for each population. For each subject in our dataset, we first generate predictors $\boldsymbol{x}_{(j)i}$ as a realization of $\mathrm{N}_p(0, \boldsymbol{\Sigma})$ where $\boldsymbol{\Sigma}_{l,m} = 0.7^{|l-m|}$ for $(l, m) \in [p] \times [p]$. Given $\boldsymbol{x}_{(j)i}$, we then generate the true survival time according to the Cox proportional hazards model with Gompertz baseline hazard using $T_{(j)i} = \log\{1 - (\alpha/\zeta_{(j)}) \log(u) \exp(-\boldsymbol{x}_{(j)i}^{\top} \boldsymbol{b}_{*(j)})\}/\alpha$

14

for $i \in [n_{(j)}]$ and $j \in [J]$, where $u \sim \text{Uniform}(0, 1)$ independently for all $(j)i$ combinations. We set $\alpha = \pi/(600\sqrt{6})$ and $\zeta_{(j)} = \alpha \exp\{-0.5772 - \alpha\nu_{(j)}\}$ where $\nu_{(j)} = 2000 + 10(j - 1)$ for $j \in [J]$. These parameter values are chosen so that the generated survival times mimicked those in the TCPA data analyzed in Section 6. Under this data generating model, the baseline hazards are distinct across populations. See Bender et al. (2005) for more details about this data generating model. Given $t_{(j)1}, \ldots, t_{(j)n_{(j)}}$ for $j \in [J]$, we generate censoring times $c_{(j)i}$ as realizations of an exponential random variable with mean $q_\xi(\{t_{(j)i}\}_{i=1}^{n_{(j)}})$, where q_ξ denotes the ξth quantile of its argument. We allow ξ to vary across populations. If $n_{(j)} < 300$, we use $\xi = \tau$, whereas if $n_{(j)} \geq 300$, we use $\xi = \tau + 0.20$ with τ varying across simulation settings.

Given censoring times, we set $y_{(j)i} = \min\{c_{(j)i}, t_{(j)i}\}$ and $\delta_{(j)i} = \mathbf{1}(y_{(j)i} = t_{(j)i})$. We generate $\boldsymbol{B}_* = \boldsymbol{U}\boldsymbol{V}^\top \in \mathbb{R}^{p \times J}$ where $\boldsymbol{U} \in \mathbb{R}^{p \times r_*}$ has 20 rows randomly selected to be nonzero with each nonzero entry independent and uniformly distributed on $[-\sqrt{8}/r_*, -\sqrt{2}/r_*] \cup [\sqrt{2}/r_*, \sqrt{8}/r_*]$. The matrix $\boldsymbol{V} \in \mathbb{R}^{r_* \times J}$ is a randomly generated semi-orthogonal matrix.

In our simulation studies, we consider: (1) $r_* \in [6]$ with $p = 250$ and $\tau = 0.35$; (2) $\tau \in \{0.25, 0.35, 0.45, 0.55, 0.65\}$ with $p = 250$ and $r_* = 3$; and (3) $p \in \{100, 200, 300, 400, 500\}$ with $r_* = 3$ and $\tau = 0.35$. In each replication, we also construct validation and testing sets of size 150 and 1000, respectively, for each $j \in [J]$. We consider three performance metrics: (1) $\text{tr}\{(\widehat{\boldsymbol{B}} - \boldsymbol{B}_*)^\top \boldsymbol{\Sigma}(\widehat{\boldsymbol{B}} - \boldsymbol{B}_*)\}$, i.e., model error; (2) concordance (C-index) between linear predictors and observed survival times; and (3) Brier score evaluated at the median observed survival time. Concordance and Brier scores are averaged over the J populations. See Web Appendix C for further descriptions of these metrics.

5.2 Competing methods

We consider various competing methods, some of which can exploit the low-rankness assumption in (2). Many of the competitors we consider estimate $\boldsymbol{B}_*$ column-by-column. The jth column of these estimators can be expressed

$$\underset{\boldsymbol{b}_{(j)} \in \mathbb{R}^p}{\arg\max} \left\{ n_{(j)}^{-1} \ell_{(j)}(\boldsymbol{b}_{(j)}) - \lambda_{(j)} \|\boldsymbol{b}_{(j)}\|_q^q \right\}, \tag{10}$$

where $\lambda_{(j)} \in (0, \infty)$ is a user specified tuning parameter for $j \in [J]$ and $q \in \{1, 2\}$. Based on (10), we consider the following alternative estimators: `Sep-Ridge`, (10) with $q = 2$ and each $\lambda_{(j)}$ chosen to minimize the deviance on the jth population validation set (i.e., separate ridge regressions used for each population); `Sep-Lasso`, (10) with $q = 1$ and $\lambda_{(j)}$ chosen to minimize the deviance on the jth population validation set; `Proj-Sep-Ridge` (resp. `Proj-Sep-Lasso`),

the nearest rank-r approximation to separate ridge-penalized (resp. lasso-penalized) Cox models. To compute `Proj-Sep-Ridge`, we first obtain (10) with $q = 2$ and each $\lambda_{(j)}$ chosen to minimize the deviance on the jth population validation set, then find the nearest rank r approximation (in squared Frobenius norm) to the estimate. The rank parameter, r, is also chosen to minimize the deviance on the validation set. The latter estimators are inspired by the "lazy" sparse reduced-rank regression estimator from Qian et al. (2020, Section 3.4.5).

We also consider `Convex-Approx`, the nuclear norm and group lasso penalized estimator of $\boldsymbol{B}_*$ described in Web Appendix A.1. Note that both `Proj-` estimators and `Convex-Approx` exploit the assumption of low-rankness in $\boldsymbol{B}_*$. Finally, we use our method with $\rho_0 = 50$ (where ρ_0 is the initializing value of ρ for the penalty method) and $\mu = 0.1$ fixed across every setting. Tuning parameters s and r are chosen to minimize the validation set deviance. We use `LR-Cox` to denote (3).

5.3 Results

We display results in Figure 1. In the first row, we display the performance of the estimators as the rank of the true regression coefficient matrix increases with τ and p held fixed at 0.35 and 250, respectively. In the three settings we consider, we see that in terms of model error, concordance, and Brier score, our method performs the best among all the competitors considered. As the rank increases, the difference in model error between our method and `Convex-Approx` begin to decrease. In terms of model error, only the performance of `LR-Cox` degrades as the rank increases. In contrast, when considering both concordance and Brier score, performance of every method (including the proposed `LR-Cox`) improves as the rank of $\boldsymbol{B}_*$ increases. This can be explained by the fact that under our data generating model, as the rank increases, the signal strength increases. Nevertheless, in each setting, our method outperforms all competitors.

In the second row of Figure 1, we display results as τ, the censoring parameter, increases with p and the rank of $\boldsymbol{B}_*$ held fixed at 250 and 3, respectively. We see the performance of all methods improve in all metrics as the proportion of censoring decreases, but our method's superior performance remains constant across the considered τ. Notably, `Proj-Sep-Lasso` only slightly outperforms `Convex-Approx` in terms of concordance, although `Convex-Approx` is better than `Proj-Sep-Lasso` in terms of Brier score and model error.

Finally, in the bottom row of Figure 1, we display results as p varies with τ and the rank of $\boldsymbol{B}_*$ held fixed at 0.35 and 3, respectively. As expected, as p grows, keeping the number of truly important variables fixed at 20, each method's performance tends to degrade. However,

16

the performance of `LR-Cox` seems to be less affected by the change in p than does the convex approximation `Convex-Approx` or any of the competing estimators. For example, in terms of Brier score, our method tends to perform similarly for all considered values of p, whereas performance of all other methods degrades as p approaches 500.

Interestingly, the difference between `Convex-Approx` and `Proj-Sep-Lasso` is relatively small in all three settings. For example, `Proj-Sep-Lasso` outperforms `Convex-Approx` in terms of concordance when the rank of $\boldsymbol{B}_*$ is less than three. This lends further evidence to the notion that `Convex-Approx` leads to overly biased estimates.

5.4 Additional simulation studies

In the Web Appendix, we provide additional simulation study results. In Web Appendix F.1, we compare (3) to an alternative estimator which uses a sample size-weighted version of $\mathcal{L}$. In Web Appendix F.2, we assess the sensitivity of (3) to the choice of rank. Unsurprisingly, we see that overspecification of the rank has only a slight effect relative to underspecification. In Web Appendix F.3, we consider comprehensive simulation studies under three alternative data generating models. Specifically, we consider settings where some factors are not shared by subsets of the populations and settings where some predictors are relevant for only a subset of the populations. Finally, in Web Appendix F.4, we present mean squared estimation error results under the data generating models from Section 5.1.

6 Pan-cancer survival analysis with protein expression

6.1 Data processing

Finally, we perform the pan-cancer integrative survival analysis that motivated our proposed method. The data we analyze is from The Cancer Proteome Atlas (TCPA), detailed in Li et al. (2013). These data consist of clinical information and protein expression measurements from reverse-phase protein arrays for patients with many distinct types of cancer. In our analysis, we use data from all cancer types which had at least 30 patients with recorded failure times and 30 with censored failure times. Kaplan-Meier survival curves for each of the 18 cancer types are displayed in Web Figure 1. The normalized expression (level 4 from Pan-Can 32 from the TCPA database) of the $p = 210$ proteins with no missing expression in the cancer types we analyzed were used as the predictors in our analysis.
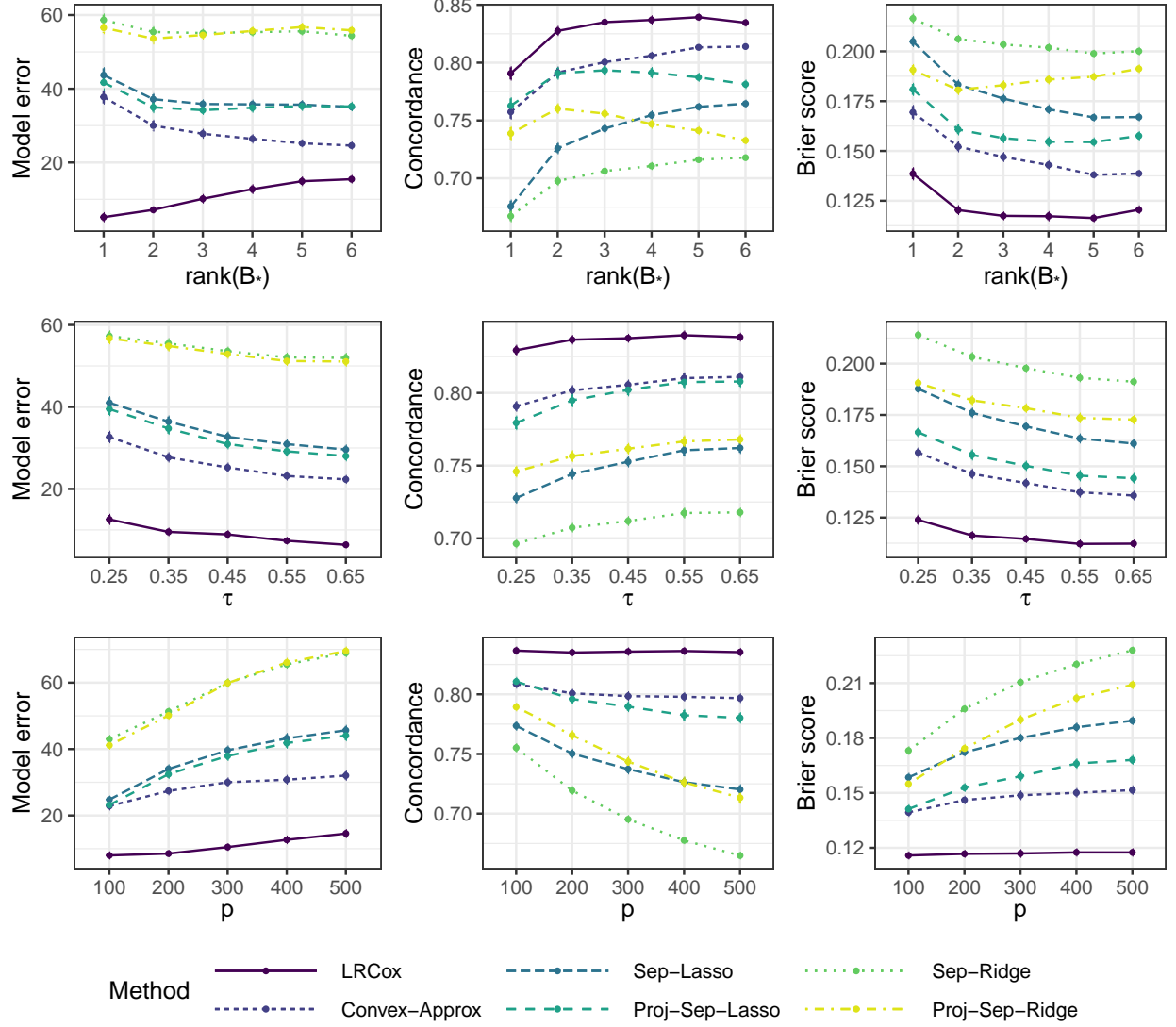
Figure 1: Averages (plus and minus two standard errors) for each of the six methods over 100 independent replications under the models described in Section 5.1 with (top row) $(p, \tau) = (250, 0.35)$, (middle row) $(p, r_*) = (250, 3)$, and (bottom row) $(r_*, \tau) = (3, 0.35)$. This figure appears in color in the electronic version of this article.

6.2 Results

We use five-fold cross-validation to select r and s for model fitting. For each $k \in [5]$, we fit the model using all the data outside the kth fold with $r \in [10]$, $s \in \{10, 12, 14, \ldots, 40\}$, $\mu = 50$, and $\rho_0 = 50$. In Web Figure 2, we display a heatmap of the cross-validated linear predictor scores (see equation (13) of the Web Appendix). Models with rank less than four all performed poorly, as did models with rank greater than seven and $s \geq 26$. The minimum
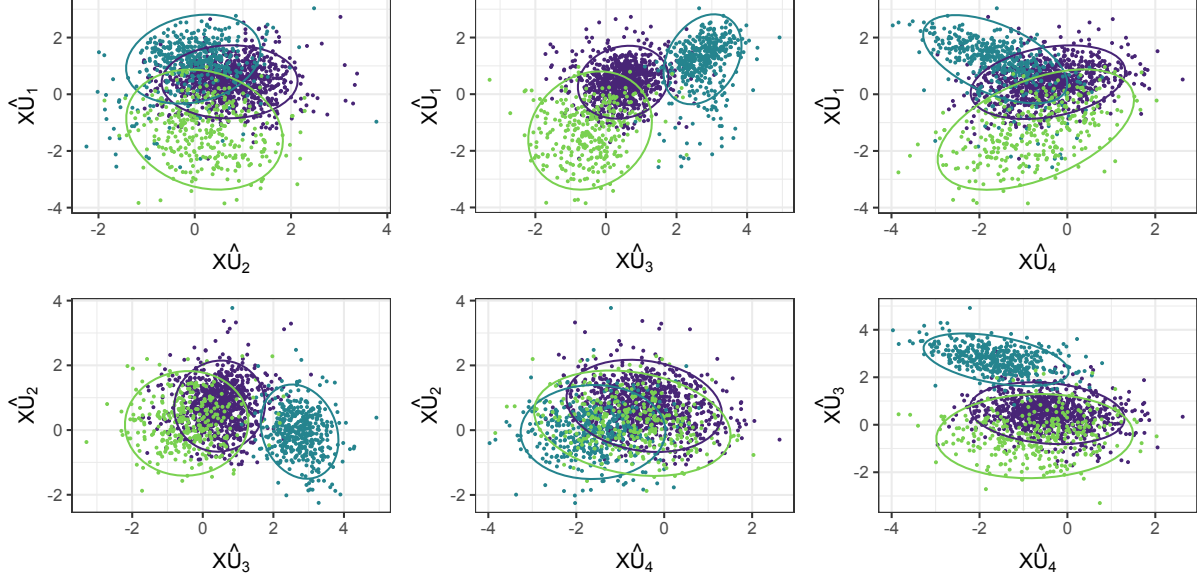
Figure 2: Pairwise estimated factors for three cancer types: BRCA (purple), LGG (blue), and LUSC (green). Normal ellipses, included for improved visualization, were computed using `stat_ellipse` in `ggplot2`. This figure appears in color in the electronic version of this article: the mentioned colors refer to that version.

overall cross-validated linear predictor score is achieved by $s = 20$ and $r = 6$.

We refit the model to the entire dataset using $s = 20$ and $r = 6$. This model assumes there are $r = 6$ linear combinations of the proteins and each linear combination consists of the same $s = 20$ proteins. Compared to the full model with $pJ = 3780$ coefficients to be estimated, there are $(J + s - r)r = 192$ parameters to be estimated in our model. See Web Appendix I for an explanation of the number of parameters in reduced-rank models.

First, we display the estimated factors to show how they separate distinct types of cancer. Specifically, taking the singular value decomposition $\widehat{\boldsymbol{B}} = \widehat{\boldsymbol{U}}\widehat{\boldsymbol{D}}\widehat{\boldsymbol{R}}^\top$ where $\widehat{\boldsymbol{U}} = (\widehat{\boldsymbol{u}}_1, \ldots, \widehat{\boldsymbol{u}}_6) \in \mathbb{R}^{p \times 6}$, $\boldsymbol{x}^\top \widehat{\boldsymbol{u}}_k$ can be interpreted as the kth estimated factor for a subject with protein expression $\boldsymbol{x} \in \mathbb{R}^p$. One can think of these factors as the underlying low-dimensional predictors, and can in turn think of $\widehat{\boldsymbol{V}}^\top = \widehat{\boldsymbol{D}}\widehat{\boldsymbol{R}}^\top$ as the regression coefficient matrix for these low-dimensional predictors. In Figure 2, we display all pairwise factors for three cancer types: breast cancer (BRCA, purple), low grade glioma (LGG, blue), and lung squamous cell carcinoma (LUSC, green). We see that the first and third factors nearly entirely separate the three cancer types. In addition, we see that BRCA and LUSC are entirely separated in three of the six plots.

In Figure 3, we also display the matrix $\widehat{\boldsymbol{D}}\widehat{\boldsymbol{R}}^\top$. Since $\widehat{\boldsymbol{D}}\widehat{\boldsymbol{R}}^\top$ can be interpreted as the

19
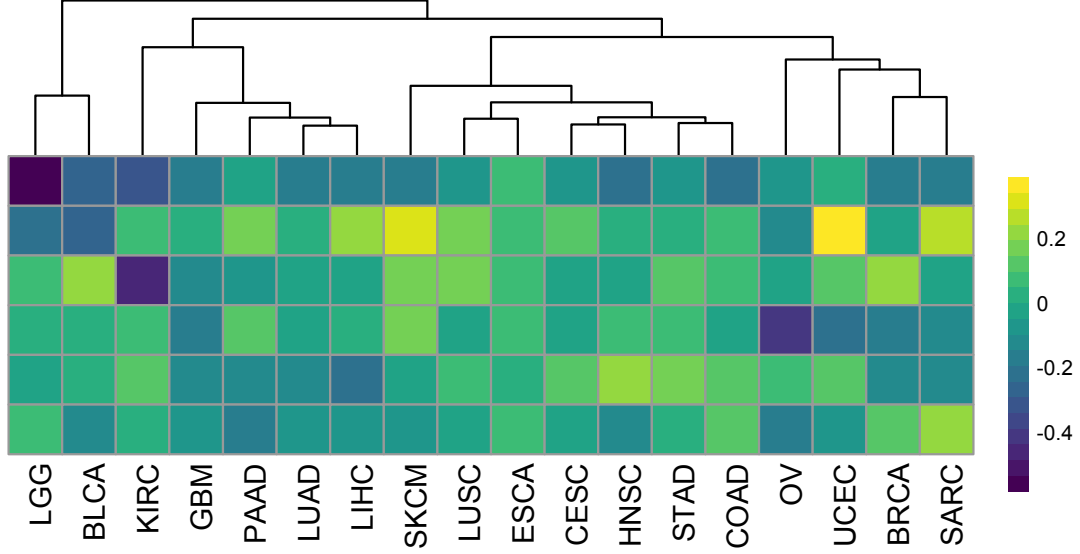
Figure 3: A heatmap of $\widehat{\boldsymbol{D}}\widehat{\boldsymbol{R}}^{\top}$ based on the fitted model described in Section 6.2. Cancer types were sorted by hierarchical clustering. This figure appears in color in the electronic version of this article.

matrix of regression coefficients corresponding to the low-dimensional factors, we can compare coefficients across cancer types. For example, KIRC has a negative coefficient for the third factor, whereas BRCA has a positive coefficient. Similarly, BLCA has a negative coefficient for the second factor, whereas UCEC has a relatively large positive coefficient.

The proteins selected by our method can be found in Table 1. Many of the identified proteins are known to play a role in cancer biology. For example, c-Kit, a tyrosine kinase receptor, is often found in higher amounts on the surface of cancer cells. Recent studies have shown that c-Kit is expressed in aggressive cancers, on circulating tumor cells, and in recurrent and resistant tumors (Foster et al., 2018). Similarly, there is evidence that FASN, which encodes fatty acid synthase, is a metabolic oncogene that plays a central role in tumor progression and survival (Flavin et al., 2010). In addition, Annexin A1 is known to inhibit innate immune cells and promote T-cell activation (Perretti and D'acquisto, 2009), and Paxillin is believed to play a role in tumorigenesis and metastasis (Deakin et al., 2012).

To assess whether our method yields better fitted models than competitors on rare cancer types, we also perform a leave-one-out cross-validation analysis. In terms of both concordance and linear predictor scores, our method performs better than `Sep-Lasso`, `Sep-Ridge`, and separate elastic net estimators on four of the five cancer types we considered. In the one cancer type where a competitor performed better, none of the methods had concordance higher than

Table 1: Proteins with nonzero coefficient estimates using the full dataset with $(r, s) = (6, 20)$ chosen by five-fold cross-validation minimizing the cross-validated linear predictor score.

x4E-BP1_pT37_T46	c-Kit	Caspase-7_cleavedD198	Caveolin-1
Gab2	HSP70	IGFBP2	MAPK_pT202_Y204
PAI-1	Paxillin	FASN	MYH11
TFRC	EPPK1	Acetyl-a-Tubulin-Lys40	Annexin-1
EGFR	NF-kB-p65_pS536	NDRG1_pT346	p16Ink4a

Table 2: Average concordance (and standard errors) for each of the four considered methods across 1000 independent replications in each of the four datasets. Bolded cells are those with largest median among the four methods.

Cancer	LR-Cox-DR	Sep-Ridge	Sep-Lasso	Sep-En
KIRP	0.575 (0.007)	**0.672** (0.007)	0.464 (0.003)	0.497 (0.006)
THCA	**0.545** (0.008)	0.528 (0.009)	0.456 (0.004)	0.460 (0.007)
READ	**0.529** (0.009)	0.524 (0.009)	0.490 (0.005)	0.534 (0.008)
MESO	**0.618** (0.006)	0.551 (0.006)	0.526 (0.004)	0.544 (0.005)

0.5, which corresponds to randomly guessing the linear predictor ordering. Additional details can be found in Web Appendix G.

6.3 External validation of factors

To further investigate whether our method identifies a useful set of pan-cancer proteomic factors for survival, we use our estimated factors to model survival in other cancer types. Specifically, we focus on four rare cancer types from TCPA which were excluded from our analysis in Section 6.2. These are cancer types that have at least 50 subjects, and both 15 failure and censoring times: kidney renal papillary cell carcinoma (KIRP), thyroid carcinoma (THCA), rectum adenocarcinoma (READ), and mesothelioma (MESO), which have sample sizes $208, 372, 130,$ and 61, respectively.

For 1000 independent replications, we randomly split each dataset into a training set (90%) and testing set (10%). Recall that in these data, predictors consist of $p = 210$ proteins. We fit a Cox proportional hazards model to the training data using three methods: `Sep-Lasso`, `Sep-Ridge`, and a version of (10) with elastic net penalty (`Sep-En`). For each method, tuning parameters are chosen by five-fold cross-validation on the training set. We obtain the estimated linear predictors on the testing set based on the fitted model using the tuning parameter which minimized partial likelihood deviance (the default in `glmnet`).

The fourth method we considered, `LR-Cox-DR`, relies on our dimension-reduced fitted model from Section 6.2. Specifically, letting $\widehat{\boldsymbol{U}}$ be the left singular vectors of the estimate of $\boldsymbol{B}_*$ from before, we first set $\widetilde{\boldsymbol{x}}_i = \widehat{\boldsymbol{U}}^\top \boldsymbol{x}_i \in \mathbb{R}^6$ (where $\boldsymbol{x}_i$ is the ith training subject's predictors) and fit a standard Cox proportional hazards model with $\widetilde{\boldsymbol{x}}_i$ as predictors. We do this separately for each of the four cancer types. Then, we transform testing set predictors using the same $\widehat{\boldsymbol{U}}$ and obtain the estimated linear predictor based on the fitted Cox model. It is important to emphasize that the estimate $\widehat{\boldsymbol{U}}$ came from datasets entirely separate from those we consider here. Specifically, $\widehat{\boldsymbol{U}}$ corresponds to the estimated factors from the 18 cancer types analyzed in Section 6.2 which did not include KIRP, THCA, READ, or MESO.

In each replication and cancer type, we measure the concordance between the estimated linear predictor and the true survival outcomes (with appropriate adjustments for censoring). We report averages in Table 2. We see that our method, denoted `LR-Cox-DR` in Table 2, performs as well or better than competitors in three of the four external datasets. Ridge regression performs best in one dataset (KIRP), and in another (READ), `LR-Cox-DR`, `Sep-En`, and `Sep-Ridge` were not significantly different. When considering only methods which perform variable selection, `LR-Cox-DR` performs as well or better than `Sep-Lasso` and `Sep-En` in all four datasets. Together, these results suggest that our estimated factor weights may generalize to other cancer types reasonably well.

7   Discussion

There are two directions we plan to explore in future research. First, our estimator relies on a computational approach using distance-to-set penalties. Instead, one could compute (3) using an iterative hard thresholding algorithm (Jain et al., 2014). This could be more efficient than our approach but would require iteratively projecting onto the intersection of the set of row-sparse and low-rank matrices, which is nontrivial. Second, we plan to extend the proposed methodology to models which do not assume proportional hazards, e.g., the accelerated failure time (AFT) model. Recent advances in computation for semiparametric AFT models in high dimensions (Suder and Molstad, 2022) may be useful for such extensions.

Acknowledgements

Data availability statement

The data that support the findings in this paper are openly available from The Cancer Proteome Atlas at https://tcpaportal.org/tcpa/ (Li et al., 2013, 2017).

References

Akbani, R., Ng, P. K. S., Werner, H. M., Shahmoradgoli, M., Zhang, F., Ju, Z., Liu, W., Yang, J.-Y., Yoshihara, K., and Li, J. (2014). A pan-cancer proteomic perspective on the cancer genome atlas. *Nature Communications* **5,** 1–15.

Anderson, T. W. (1999). Asymptotic distribution of the reduced rank regression estimator under general conditions. *Annals of Statistics* **27,** 1141–1154.

Baladandayuthapani, V., Talluri, R., Ji, Y., Coombes, K. R., Lu, Y., Hennessy, B. T., Davies, M. A., and Mallick, B. K. (2014). Bayesian sparse graphical models for classification with application to protein expression data. *Annals of Applied Statistics* **8,** 1443.

Bender, R., Augustin, T., and Blettner, M. (2005). Generating survival times to simulate Cox proportional hazards models. *Statistics in Medicine* **24,** 1713–1723.

Chen, L. and Huang, J. Z. (2012). Sparse reduced-rank regression for simultaneous dimension reduction and variable selection. *Journal of the American Statistical Association* **107,** 1533–1545.

Dai, B. and Breheny, P. (2019). Cross validation approaches for penalized cox regression. https://arxiv.org/abs/1905.10432.

Deakin, N. O., Pignatelli, J., and Turner, C. E. (2012). Diverse roles for the paxillin family of proteins in cancer. *Genes and Cancer* **3,** 362–370.

Fiocco, M., Putter, H., and Van Houwelingen, J. (2005). Reduced rank proportional hazards model for competing risks. *Biostatistics* **6,** 465–478.

Flavin, R., Peluso, S., Nguyen, P. L., and Loda, M. (2010). Fatty acid synthase as a potential therapeutic target in cancer. *Future Oncology* **6,** 551–562.

Foster, B. M., Zaidi, D., Young, T. R., Mobley, M. E., and Kerr, B. A. (2018). CD117/c-Kit in cancer stem cell-mediated progression and therapeutic resistance. *Biomedicines* **6,** 31.

Hieronymus, H., Murali, R., Tin, A., Yadav, K., Abida, W., Moller, H., Berney, D., Scher, H., Carver, B., and Scardino, P. (2018). Tumor copy number alteration burden is a pan-cancer prognostic factor associated with recurrence and death. *Elife* **7,** e37294.

Hjort, N. L. and Pollard, D. (2011). Asymptotics for minimisers of convex processes. https://arxiv.org/abs/1107.3806.

Hong, W., Gu, Y., Guan, R., Xie, D., Zhou, H., and Yu, M. (2020). Pan-cancer analysis of the casp gene family in relation to survival, tumor-infiltrating immune cells and therapeutic targets. *Genomics* **112,** 4304–4315.

Huang, Y., Zhang, Q., Zhang, S., Huang, J., and Ma, S. (2017). Promoting similarity of sparsity structures in integrative analysis with penalization. *Journal of the American Statistical Association* **112,** 342–350. PMID: 30100648.

Jain, P., Tewari, A., and Kar, P. (2014). On iterative hard thresholding methods for high-dimensional M-estimation. *Advances in Neural Information Processing Systems* **27,** 685–693.

Keys, K. L., Zhou, H., and Lange, K. (2019). Proximal distance algorithms: Theory and practice. *Journal of Machine Learning Research* **20,** 1–38.

Lange, K. (2016). *MM Optimization Algorithms*, volume 147. SIAM.

Li, J., Akbani, R., Zhao, W., Lu, Y., Weinstein, J. N., Mills, G. B., and Liang, H. (2017). Explore, visualize, and analyze functional cancer proteomic data using the cancer proteome atlas. *Cancer Research* **77,** e51–e54.

Li, J., Lu, Y., Akbani, R., Ju, Z., Roebuck, P. L., Liu, W., Yang, J.-Y., Broom, B. M., Verhaak, R. G., and Kane, D. W. (2013). TCPA: a resource for cancer functional proteomics data. *Nature Methods* **10,** 1046–1047.

Li, K.-C., Wang, J.-L., and Chen, C.-H. (1999). Dimension reduction for censored regression data. *The Annals of Statistics* **27,** 1–23.

Li, L. and Li, H. (2004). Dimension reduction methods for microarrays with application to censored survival data. *Bioinformatics* **20,** 3406–3412.

Liu, J., Huang, J., Zhang, Y., Lan, Q., Rothman, N., Zheng, T., and Ma, S. (2014). Integrative analysis of prognosis data on multiple cancer subtypes. *Biometrics* **70,** 480–488.

Maity, A. K., Bhattacharya, A., Mallick, B. K., and Baladandayuthapani, V. (2020). Bayesian data integration and variable selection for pan-cancer survival prediction using protein expression data. *Biometrics* **76,** 316–325.

Molstad, A. J. and Patra, R. K. (2022). R code for reproducing results from "Dimsension reduction for integrative survival analysis", https://github.com/ajmolstad/integrativecox.

Nocedal, J. and Wright, S. (2006). *Numerical optimization.* Springer Science & Business Media.

Perperoglou, A., Cessie, S. l., and van Houwelingen, H. C. (2006). Reduced-rank hazard regression for modelling non-proportional hazards. *Statistics in Medicine* **25,** 2831–2845.

Perretti, M. and D'acquisto, F. (2009). Annexin A1 and glucocorticoids as effectors of the resolution of inflammation. *Nature Reviews Immunology* **9,** 62–70.

Qian, J., Tanigawa, Y., Li, R., Tibshirani, R., Rivas, M. A., and Hastie, T. (2020). Large-scale sparse regression for multiple responses with applications to UK biobank. *BioRxiv* .

Shankavaram, U. T., Reinhold, W. C., Nishizuka, S., Major, S., Morita, D., Chary, K. K., Reimers, M. A., Scherf, U., Kahn, A., and Dolginow, D. (2007). Transcript and protein expression profiles of the nci-60 cancer cell panel: an integromic microarray study. *Molecular Cancer Therapeutics* **6,** 820–832.

She, Y. (2017). Selective factor extraction in high dimensions. *Biometrika* **104,** 97–110.

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

Stoica, P. and Viberg, M. (1996). Maximum likelihood parameter and rank estimation in reduced-rank multivariate linear regressions. *IEEE Transactions on Signal Processing* **44,** 3069–3078.

Suder, P. M. and Molstad, A. J. (2022). Scalable algorithms for semiparametric accelerated failure time models in high dimensions. *Statistics in Medicine* **41,** 933–949.

Tang, L., Zhou, L., and Song, P. X. (2019). Fusion learning algorithm to combine partially heterogeneous Cox models. *Computational Statistics* **34,** 395–414.

TCPA Data Portal (Accessed August, 2020). https://tcpaportal.org/tcpa/download.html [tcga-pancan32-l4.zip].

Van der Vaart, A. (2002). Semiparametric statistics. In *Lectures on probability theory and statistics*, volume 1781 of *Lecture Notes in Math.*, pages 331–457. Springer.

Wang, L., Li, Y., Zhou, J., Zhu, D., and Ye, J. (2017). Multi-task survival analysis. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 485–494.

Xu, J., Chi, E., and Lange, K. (2017). Generalized linear model regression under distance-to-set penalties. In *Advances in Neural Information Processing Systems*, pages 1385–1395.

Zhang, Q., Zhang, S., Liu, J., Huang, J., and Ma, S. (2016). Penalized integrative analysis under the accelerated failure time model. *Statistica Sinica* **26,** 493–508.

Zhao, Q., Shi, X., Huang, J., Liu, J., Li, Y., and Ma, S. (2015). Integrative analysis of '-omics' data using penalty functions. *Wiley Interdisciplinary Reviews: Computational Statistics* **7,** 99–108.

Supporting Information

Web appendices, figures, and tables referenced in Sections 2, 4, 5, and 6 are available with this paper at the Biometrics website on Wiley Online Library. Code to reproduce all simulation results are available for download at `github.com/ajmolstad/IntegrativeCox`.