Published in final edited form as:

Stat Med. 2018 July 20; 37(16): 2501–2515. doi:10.1002/sim.7681.

Pathway Aggregation for Survival Prediction via Multiple Kernel Learning

Jennifer A. Sinnott[a,*] and **Tianxi Cai**[b]

[a]Department of Statistics, Ohio State University, Columbus, OH, USA

[b]Department of Biostatistics, Harvard University, Boston, MA, USA

Abstract

Attempts to predict prognosis in cancer patients using high dimensional genomic data such as gene expression in tumor tissue can be made difficult by the large number of features and the potential complexity of the relationship between features and the outcome. Integrating prior biological knowledge into risk prediction with such data by grouping genomic features into pathways and networks reduces the dimensionality of the problem and could improve prediction accuracy. Additionally, such knowledge-based models may be more biologically grounded and interpretable. Prediction could potentially be further improved by allowing for complex nonlinear pathway effects. The kernel machine framework has been proposed as an effective approach for modeling the nonlinear and interactive effects of genes in pathways for both censored and non-censored outcomes. When multiple pathways are under consideration, one may efficiently select informative pathways and aggregate their signals via multiple kernel learning (MKL), which has been proposed for prediction of non-censored outcomes. In this paper, we propose MKL methods for censored survival outcomes. We derive our approach for a general survival modeling framework with a convex objective function, and illustrate its application under the Cox proportional hazards and semiparametric accelerated failure time models. Numerical studies demonstrate that the proposed MKL-based prediction methods work well in finite sample and can potentially outperform models constructed assuming linear effects or ignoring the group knowledge. The methods are illustrated with an application to two cancer data sets.

Keywords

Accelerated Failure Time Model; Cox Proportional Hazards Model; Kernel Machines; Multiple Kernel Learning; Risk Prediction

[*]Correspondence to: Jennifer A. Sinnott, Department of Statistics, 1958 Neil Ave, Columbus, OH 43210, USA. jsinnott@stat.osu.edu.

7. Software

Software in the form of an R packages `survivalMKL` is available on request from the corresponding author (jsinnott@stat.osu.edu).

8. Supplementary Material

One Appendix is available in the supplementary material online.

1. Introduction

Studies linking disease outcomes, such as cancer recurrence and death, to large-scale genomic data, such as tumor gene expression, are rich resources for improving both our understanding of the progression of disease and our ability to predict patient prognosis. However, the number of genomic markers in such studies is often large relative to the sample size, which can make it hard to differentiate true biological relationships from noise and false positive associations. One appealing idea is to employ models that can partition the total effect into a sum of pathway-level effects. Such an approach would take advantage of the extensive biological knowledge available grouping genes into pathways and networks thought to work together — knowledge that informs collections such as the Molecular Signatures Database (mSigDB) [1]. A pathway-based approach can help with dimension reduction in two ways: first, the number of pathways tends to be smaller than the total number of genes, and second, within a pathway, genes may be correlated with each other and it may be possible to capture a pathway's effect with a smaller number of summary variables. In the presence of nonlinear effects, models that incorporate pathway information could reduce model complexity by allowing complex within-pathway effects while assuming that the signals are additive across pathways, which in turn could improve prediction. Pathway-based methods can also improve interpretability because the pathways are defined by known or hypothesized functions, which can facilitate generation of mechanistic hypotheses. Thus, interrogating pathways could be a more effective approach for understanding the biological process of the disease. Moreover, as our knowledge of biological systems continues to improve, models that integrate such information could be better for effective, reproducible risk prediction. For example, there has been evidence in certain cancers that while many individuals will have key pathways dysregulated, the actual component of the pathway that is altered may differ across individuals [2, 3].

Recently, many approaches for testing the association between a given gene set and various types of outcomes and for estimating the effects of the gene set have been proposed. In linear and logistic regression, some examples include Goeman's global test [4] which assumes linear gene effects, as well as kernel machine (KM) methods [5, 6, 7] which allow potentially nonlinear effects. KM modeling is an attractive tool for quantifying complex pathway effects because it allows for nonlinear effects without explicitly specifying the forms of those effects. To further leverage the correlation structure of the pathway, one may employ kernel principal components analysis (PCA) to reduce the dimensionality of the feature space to improve prediction [8, 9]. When the outcome of interest is subject to censoring, KM tests have been developed for the Cox proportional hazards [10, 11, 12, 13] and the semiparametric accelerated failure time (AFT) [14, 15, 16, 17, 18] models.

When multiple pathways have been identified as potentially associated with patient prognosis and interest lies in combining information across these pathways into a single prediction model, one could treat the entire collection of genes as a single pathway and employ the aforementioned procedures. However, if biological pathways are meaningful entities, this approach may not be effective since it does not incorporate the pathway structure, and it does not eliminate pathways which provide redundant information. An appealing strategy to leverage the pathway information is to allow simultaneous selection of informative pathways and estimation of their effects. When the effects are assumed linear, the group lasso (GLASSO), proposed in [19], can achieve these goals by penalizing the sum of the ℓ_2-norms of the genes in each pathway. GLASSO estimators and their variations have been further developed for various settings including censored outcomes [20, 21, 22, 23, 24, 25, 26, 27]. In the presence of nonlinear effects, multiple kernel learning (MKL) has been proposed as an effective approach to aggregate complex signals from multiple pathways [28, 29]. MKL models can incorporate the pathway information by allowing nonlinear effects within each pathway, but assume that signals are additive across pathways. Proper choices of penalization procedures under the MKL model can lead to efficient and sparse estimators for the pathway effects. Theory of MKL and the application of MKL methods to genomic data have been considered for non-censored outcomes [30, 31, 32, 33, 34]. In this paper, we propose MKL procedures to efficiently estimate the effects of multiple pathways on censored survival outcomes.

The main contribution of this paper is the extension of the MKL framework to survival models, which has not to our knowledge been previously done, with the goal of pathway selection, estimation, and aggregation to build a single pathway-based risk score that is predictive of prognosis. We describe the approach for a general objective function, and then provide specific implementation methods and software for the Cox and AFT models. The other contributions of this paper are in the solutions and strategies we propose here, in order to deal with issues that arise in implementation of MKL in survival models; we briefly note these here. First, rather than develop specific minimization algorithms for each model-specific objective function, we fit the MKL model using a quadratic approximation [35, 36]. Second, pathways are likely to have both within- and between-pathway correlation, and the pathways may overlap; we choose penalizations that behave reasonably in the face of correlation, and formulate the problem to allow for pathway overlap. Third, including many pathways and many variables within each pathway can yield models that are difficult or time-consuming to fit, and we propose several strategies for balancing the allowable model complexity and computation time. We select kernel tuning parameters marginally based on pre-existing KM pathway score tests. We use kernel PCA to reduced the number of terms in the model associated with each pathway. For the semiparametric AFT model, we replace the non-smooth rank-based objective function with a smoothed version. Implementation of the entire procedure is provided in the R package `survivalMKL`, available from the authors on request.

The rest of this paper proceeds as follows. In Section 2, we derive our methods for a general survival modeling framework with a convex objective function L_0 and propose the use of quadratic approximation for easy computation; a step-by-step outline of the method, including details of tuning, is provided in subsection 2.4. In Section 3, we illustrate these methods for (i) the Cox model, with L_0 being the log partial likelihood; and (ii) the AFT model, with L_0 being the smoothed Gehan objective function. We demonstrate the procedure under these two models in simulation (Section 4), and provide a data analysis application to two cancer gene expression data sets (Section 5). Concluding remarks are in Section 6.

2. Approach with a General Objective Function

Let T denote the survival time, **d** the $p_d \times 1$ vector of clinical covariates, and **z** the $p_z \times 1$ vector of genomic measurements. Due to censoring, for T, we observe $X = \min\{T, C\}$ and $\Delta = \mathbf{I}(T \leq C)$, where the censoring time C is assumed to be independent of T given $\mathbf{w} = (\mathbf{d}^\top, \mathbf{z}^\top)^\top$. The observed data consist of n independent and identically distributed (iid) random vectors, $\mathscr{O} = \{(X_i, \Delta_i, \mathbf{w}_i^\top) : i = 1, \ldots, n\}$. We assume that the genomic covariates **z** are grouped into M pathways, where for $m = 1, \ldots, M$; $g_m \subset \{1, \ldots, p_z\}$ denotes the indices corresponding to the m^{th} pathway; and $\mathbf{z}_{g_m}$ denotes the vector of genes belonging to the m^{th} pathway. These sets of indices overlap when pathways overlap.

2.1. Kernel Machine Modeling

Suppose the overall effect of **w** on survival can be summarized through

$$\mu(\boldsymbol{\theta}_0, \mathbf{h}) = \boldsymbol{\theta}_0^\top \mathbf{d} + h_1(\mathbf{z}_{g_1}) + \cdots + h_M(\mathbf{z}_{g_M}) = \boldsymbol{\theta}_0^\top \mathbf{d} + h_\bullet(\mathbf{z}) \qquad (1)$$

where $\boldsymbol{\theta}_0 \in \mathbb{R}^{p_d}$, $\mathbf{h} = (h_1, \ldots, h_M)$, and the $h_m(\cdot)$ are centered, smooth functions quantifying the pathway effect for the m^{th} pathway for $m = 1, \ldots, M$. We write $h_\bullet(\mathbf{z})$ as shorthand for the sum of the $h_m(\mathbf{z}_{g_m})$. Here, for simplicity, we assume the clinical covariate effects are linear, assuming that proper transformations have been applied to make this assumption reasonable; however, the method can easily accommodate nonlinear covariate effects similar to those for $\mathbf{z}_{g_m}$. We assume that each $h_m \in \mathscr{H}_{K_m}$, the Hilbert space generated by some positive definite kernel $K_m(\cdot, \cdot; \rho_m)$. A kernel K_m is a measure of similarity between two vectors of genomic measurements from two people — e.g., $\mathbf{z}_{g_m,i}$ and $\mathbf{z}_{g_m,j}$ — and may depend on a possibly unknown scaling parameter ρ_m. Different choices of kernel K yield different collections of possible functions $h(\cdot) \in \mathscr{H}_K$. For example, the *linear kernel* $K(\mathbf{z}_1, \mathbf{z}_2) = \mathbf{z}_1^\top \mathbf{z}_2$ allows linear functions $h(\mathbf{z}) = \boldsymbol{\beta}^\top \mathbf{z}$. To allow for complex nonlinear effects, one may consider the *Gaussian kernel*, defined by $K(\mathbf{z}_1, \mathbf{z}_2; \rho) = \exp(-\|\mathbf{z}_1 - \mathbf{z}_2\|^2/\rho)$; the resulting function space $\mathscr{H}_K$ is generated by the radial basis functions. For notational simplicity, we will suppress ρ from K, but will discuss selection of ρ when needed.

Suppose a proper convex objective function denoted by $L_0(\boldsymbol{\theta}_0, \mathbf{h})$, such as a partial likelihood function, exists for estimating the unknown parameters. When the pathways are disjoint, it is reasonable to assume the spaces $\{\mathscr{H}_{K_1}, \ldots, \mathscr{H}_{K_M}\}$ are linearly independent, in which case the overall effect of all the pathways $h_\bullet$ has a unique decomposition $h_\bullet = h_1 + \cdots + h_M$ for $h_m \in \mathscr{H}_{K_m}$, and we may obtain estimators for $(\boldsymbol{\theta}_0, \mathbf{h})$ by minimizing a penalized objective function

$$L(\boldsymbol{\theta}_0, \mathbf{h}) = L_0(\boldsymbol{\theta}_0, \mathbf{h}) + \tau^2 \sum_{m=1}^{M} \|h_m\|_{\mathscr{H}_{K_m}}^2$$

where $\|h\|_{\mathcal{H}_K}$ is the norm of h in $\mathcal{H}_K$ and τ is a tuning parameter. The $\mathcal{H}_K$-norm of h quantifies the smoothness of h, with smaller values reflecting a smoother function. To leverage pathway structure and enable pathway selection, we take a MKL approach and further penalize $L_0(\boldsymbol{\theta}_0, \mathbf{h})$ by the sum of the norms of the h_m:

$$L_0(\boldsymbol{\theta}_0, \mathbf{h}) + \tau^2 \sum_{m=1}^{M} \|h_m\|^2_{\mathscr{H}_{K_m}} + \sum_{m=1}^{M} \lambda_m \|h_m\|_{\mathscr{H}_{K_m}}. \qquad (2)$$

Each λ_m is a tuning parameter associated with the m^{th} pathway. Depending on the tuning parameters λ_m, this penalty can have the effect of setting some pathway effects to 0. This double penalty on the norms of the functions h_m produces a group adaptive elastic net type of penalty, and thus we expect it to provide good estimation and feature selection performance when those features are not independent, which is typically true of pathways [37].

When the pathways overlap, we may not be able to assume that $h_\bullet$ can be uniquely decomposed into a sum of $h_m \in \mathcal{H}_{K_m}$, particularly when using kernels with finite dimensional bases, such as the linear kernel. In this setting we may proceed as Jacob et al. [23] do when implementing GLASSO with overlapping groups. They note that when standard linear effects GLASSO is applied when the groups overlap, if one group has coefficients set to 0, all variables in that group have coefficients set to 0 even if they also belong to another group. In the context of biological pathways, this is undesirable: for example, an important gene may belong to many pathways, but if one pathway is eliminated, the effect of that gene is set to 0 in all pathways. To resolve this issue, Jacob et al. [23] introduce an extension of the GLASSO penalty that allows the total effect of a gene to be allocated across the pathways the gene belongs to, introducing into the penalty an infimum across all possible allocations to make the representation well-defined. Implementation is achieved by duplicating columns in the design matrix when genes occur in multiple pathways and applying the classical GLASSO for non-overlapping groups to this augmented design matrix. We propose to use an analogous penalty in the MKL setting, minimizing:

$$L_{\mathrm{MKL}}(\boldsymbol{\theta}_0, \mathbf{h}) = L_0(\boldsymbol{\theta}_0, \mathbf{h}) + \inf_{\tilde{\mathbf{h}} \in \mathscr{C}_{h_\bullet}} \left\{ \tau^2 \sum_{m=1}^{M} \|\tilde{h}_m\|^2_{\mathscr{H}_{K_m}} + \sum_{m=1}^{M} \lambda_m \|\tilde{h}_m\|_{\mathscr{H}_{K_m}} \right\}$$

where $C_{h_\bullet} = \{\tilde{\mathbf{h}} = (\tilde{h}_1, \ldots, \tilde{h}_m) : \tilde{h}_m \in \mathcal{H}_{K_m},$
$\mathscr{C}_{h_\bullet} = \{\tilde{\mathbf{h}} = (\tilde{h}_1, \ldots, \tilde{h}_m): \tilde{h}_m \in \mathscr{H}_{K_m}, \sum_{m=1}^{M} \tilde{h}_m(\mathbf{z}_{g_m}) \equiv h_\bullet(\mathbf{z})\}.$

In Lemma 1 in the Web Appendix in the Supplementary Material, we mimic the proof of the representer theorem [38] as given in Scholkopf and Smola [5] to argue that for any $\boldsymbol{\theta}_0$, the minimizers $(\hat{h}_1, \ldots, \hat{h}_M)$ of $L_{\mathrm{MKL}}(\boldsymbol{\theta}_0, \mathbf{h})$ take a dual form $\hat{h}_m(\mathbf{z}) = \sum_{i=1}^{n} \alpha_{mi} K_m(\mathbf{z}, \mathbf{z}_i)$. Using the dual representation, the vector $\hat{\mathbf{h}}_m = (\hat{h}_m(\mathbf{z}_1), \ldots, \hat{h}_m(\mathbf{z}_n))^\top = \mathbb{K}_m \boldsymbol{\alpha}_m$, where $\mathbb{K}_m$ is the matrix with $(i, j)^{th}$ entry $K_m(\mathbf{z}_i, \mathbf{z}_j)$, and $\|\hat{h}_m\|^2_{\mathscr{H}_{K_m}} = \alpha_m^\top \mathbb{K}_m \alpha_m$. Thus, we may rewrite L_{MKL} with some abuse of notation as a function of $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1^\top, \ldots, \boldsymbol{\alpha}_M^\top)^\top$ and $\boldsymbol{\theta}_0$:

$$L_{\text{MKL}}(\boldsymbol{\theta}_0, \boldsymbol{\alpha}) = L_0(\boldsymbol{\theta}_0, \boldsymbol{\alpha}) + \inf_{\widetilde{\boldsymbol{\alpha}} \in \mathscr{C}_{\boldsymbol{\alpha}}} \left\{ \tau^2 \sum_{m=1}^{M} \widetilde{\boldsymbol{\alpha}}_m^\top \mathbb{K}_m \widetilde{\boldsymbol{\alpha}}_m + \sum_{m=1}^{M} \lambda_m \sqrt{\widetilde{\boldsymbol{\alpha}}_m^\top \mathbb{K}_m \widetilde{\boldsymbol{\alpha}}_m} \right\} \quad (3)$$

where $\mathscr{C}_{\boldsymbol{\alpha}} = \{\widetilde{\boldsymbol{\alpha}} : \sum_{m=1}^{M} \sum_{i=1}^{n} \widetilde{\alpha}_{mi} K_m(\mathbf{z}, \mathbf{z}_i) \equiv \sum_{m=1}^{M} \sum_{i=1}^{n} \alpha_{mi} K_m(\mathbf{z}, \mathbf{z}_i)\}$. We may further rewrite this by employing a spectral decomposition for $\mathbb{K}_m$. If we let the eigenvalues and associated eigenvectors of $\mathbb{K}_m$ be $\hat{\eta}_{ml}$ and $\hat{\boldsymbol{\zeta}}_{ml}$ respectively, for $l = 1, \ldots, n$, where we assume that $\hat{\eta}_{m1} \geq \cdots \geq \hat{\eta}_{mn}$ and that the $\hat{\boldsymbol{\zeta}}_{ml}$ are orthogonal with norm 1, then we may write $\mathbb{K}_m = \widetilde{\mathbb{B}}_m \widetilde{\mathbb{B}}_m^\top$, where $\widetilde{\mathbb{B}}_m = \left(\sqrt{\hat{\eta}_{m1}} \hat{\boldsymbol{\zeta}}_{m1} \cdots \sqrt{\hat{\eta}_{mn}} \hat{\boldsymbol{\zeta}}_{mn}\right)$. Then $\hat{\mathbf{h}}_m = \widetilde{\mathbb{B}}_m \boldsymbol{\theta}_m$, where $\boldsymbol{\theta}_m = \widetilde{\mathbb{B}}_m^\top \boldsymbol{\alpha}_m$, and letting $\boldsymbol{\theta} = (\boldsymbol{\theta}_0^\top, \boldsymbol{\theta}_1^\top, \ldots, \boldsymbol{\theta}_M^\top)^\top$, $L_{\text{MKL}}(\boldsymbol{\theta}_0, \boldsymbol{\alpha})$ may be rewritten, again with some abuse of notation, as:

$$L_{\text{MKL}}(\boldsymbol{\theta}) = L_0(\boldsymbol{\theta}) + \inf_{\widetilde{\boldsymbol{\theta}} \in \mathscr{C}_{\boldsymbol{\theta}}} \left\{ \tau^2 \sum_{m=1}^{M} \widetilde{\boldsymbol{\theta}}_m^\top \widetilde{\boldsymbol{\theta}}_m + \sum_{m=1}^{M} \lambda_m \sqrt{\widetilde{\boldsymbol{\theta}}_m^\top \widetilde{\boldsymbol{\theta}}_m} \right\}. \quad (4)$$

where the infimum is once again taken over the set of $\widetilde{\boldsymbol{\theta}}$ that keep the resulting sum of functions $h_{\boldsymbol{\cdot}}$ fixed and $L_0(\boldsymbol{\theta})$ is the objective function derived under the model with design matrix $(\mathbb{D}\ \widetilde{\mathbb{B}}_1 \ldots \widetilde{\mathbb{B}}_M)$, where $\mathbb{D}$ denotes the $n \times p_d$ matrix whose rows are the $\mathbf{d}_i$.

For computation, because the internal infimum in (3) is taken over $\widetilde{\boldsymbol{\theta}}$ which keep $L_0(\boldsymbol{\theta})$ fixed, we may proceed as Jacob et al. [23] and directly minimize:

$$L_{\text{MKL}}(\boldsymbol{\theta}) = L_\tau(\boldsymbol{\theta}) + \sum_{m=1}^{M} \lambda_m \sqrt{\boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m}, \quad L_\tau(\boldsymbol{\theta}) = L_0(\boldsymbol{\theta}) + \tau^2 \sum_{m=1}^{M} \boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m. \quad (5)$$

2.2. Kernel PCA

One motivation for using a KM approach is to gain power to detect signal in the data by leveraging the correlation structure; however, each $\boldsymbol{\alpha}_m$ in (3) has n components, so by using KMs, we have introduced a large number of parameters, which may cause a loss in power. Reparametrizing in terms of the $\boldsymbol{\theta}_m$ does not fix this problem because each $\boldsymbol{\theta}_m$ also has n components, but this parametrization leads to a natural strategy for dimension reduction, known as kernel PCA [39, 40]. Specifically, each space $\mathscr{H}_{K_m}$ is spanned by a set of orthonormal eigenfunctions $\zeta_{ml}(\cdot)$ and any function $h_m \in \mathscr{H}_{K_m}$ may be written in its primal representation, $h_m(\mathbf{z}) = \sum_{l=1}^{\mathscr{J}_m} \theta_{ml} \sqrt{\eta_{ml}} \zeta_{ml}(\mathbf{z})$, where $\eta_{m1} \geq \eta_{m2} \geq \cdots$ are the eigenvalues corresponding to $\zeta_{m1}, \zeta_{m2}, \ldots$, and $\mathscr{J}_m$ may be infinity. If the eigenvalues decay quickly,

$h_m(\mathbf{z})$ can be well-approximated by a truncated sum $h_{r_m}(\mathbf{z}) = \sum_{l=1}^{r_m} \theta_{ml}\sqrt{\eta_{ml}}\zeta_{ml}(\mathbf{z})$ for some reasonably small r_m. Moreover, it has been shown that the eigenvalues and eigenvectors of the kernel matrix $\mathbb{K}_m$ calculated on the observed data can be used to consistently estimate the underlying true eigenvalues and eigenfunctions of $\mathcal{H}_{K_m}$ (evaluated at the data points) [39, 40]. Thus, by estimating the coefficients $\boldsymbol{\theta}_m$, we are estimating h_m in its approximate primal form, and when the eigenvalues decay quickly, we may not lose much information by estimating h_m using only the first r_m eigenvectors of the kernel matrix $\mathbb{K}_m$, where r_m is the smallest number for which $\sum_{i=1}^{r_m} \hat{\eta}_{mi} / \sum_{i=1}^{n} \hat{\eta}_{mi} \geq \mathfrak{p}$ for a prespecified fraction $\mathfrak{p}$. Ideally, the included eigenvectors encode aspects of maximal variability in the data, while the excluded eigenvectors capture noise.

We propose to use this kernel PCA approximation in our method in order to greatly reduce the number of unknown parameters being estimated, and hence improve both the estimation and computational efficiency. Writing the truncated matrices $\widetilde{\mathbb{B}}_{mr_m} = \left(\sqrt{\hat{\eta}_{m1}}\hat{\boldsymbol{\zeta}}_{m1} \cdots \sqrt{\hat{\eta}_{mr_m}}\hat{\boldsymbol{\zeta}}_{mr_m}\right)$, we replace $\mathbb{K}_m$ by $\mathbb{K}_{mr_m} = \widetilde{\mathbb{B}}_{mr_m}\widetilde{\mathbb{B}}_{mr_m}^{\top}$ in (3). For notational simplicity, we will proceed, with our final abuse of notation, writing the objective function as $L_{\mathrm{MKL}}(\boldsymbol{\theta})$ and the parameter as $\boldsymbol{\theta}$, but we keep in mind that each $\widetilde{\mathbb{B}}_m$ may be replaced by its approximation $\widetilde{\mathbb{B}}_{mr_m}$, and that the associated $\boldsymbol{\theta}_m$ may be length r_m; note that we recover $\widetilde{\mathbb{B}}_m$ and $\mathbb{K}_m$ by taking $\mathfrak{p} = 1$, so that the kernel PCA formulation is in fact simply more general.

Once the model is fit, we may calculate a risk score for a new patient with covariates $\mathbf{w}_0 = (\mathbf{d}_0^{\top}, \mathbf{z}_0^{\top})^{\top}$ using the Nyström approximation method [41] which essentially uses the kernel to "project" these new covariate values onto the basis functions estimated in kernel PCA. Specifically, the risk score for the future subject is

$$\hat{\mu}_i(\hat{\boldsymbol{\theta}}) = \hat{\boldsymbol{\theta}}_0^{\top}\mathbf{d}_0 + \sum_{m=1}^{M}\sum_{\ell=1}^{r_m} \hat{\theta}_{m\ell}\mathbb{K}_{m\mathbf{z}_0}^{\top}\hat{\boldsymbol{\zeta}}_{ml}\hat{\eta}_{m\ell}^{-\frac{1}{2}} \qquad (6)$$

where $\mathbf{K}_{m\mathbf{z}_0} = (K_m(\mathbf{z}_0, \mathbf{z}_1), \ldots, K_m(\mathbf{z}_0, \mathbf{z}_n))^{\top}$ and $\hat{\theta}_{m\ell}$ is the ℓ^{th} entry of the vector $\hat{\boldsymbol{\theta}}_m$.

2.3. Least Squares Approximation

The penalty in (4) is equivalent to the GLASSO penalty — or more precisely, the group elastic net penalty. The equivalence between MKL and the GLASSO is described in detail in Bach [32], and fitting the MKL model with the linear kernel and no kernel PCA is a reparametrization of the GLASSO. Methods for fitting a model with the GLASSO penalty have been worked out for linear and logistic regression [19, 21, 23]. Rather than developing the machinery to minimize (4) for specific functions $L_0(\boldsymbol{\theta})$, we propose to approximate $L_{\tau}(\boldsymbol{\theta})$ via a quadratic approximation similar to those proposed in Wang and Leng [35] and Zhang and Lu [36]. Specifically, we may show that minimizing $L_{\mathrm{MKL}}(\boldsymbol{\theta})$ is equivalent to minimizing $Q(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{\theta} - \widetilde{\boldsymbol{\theta}})^\top \ddot{L}_\tau(\widetilde{\boldsymbol{\theta}})(\boldsymbol{\theta} - \widetilde{\boldsymbol{\theta}}) + \sum_{m=1}^{M} \lambda_m \sqrt{\boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m}$, where $\hat{\boldsymbol{\theta}} = \text{argmin}_{\boldsymbol{\theta}} L_\tau(\boldsymbol{\theta})$, for $L_\tau(\boldsymbol{\theta})$ defined in Equation (5), and $\ddot{L}_\tau(\boldsymbol{\theta}) = \partial^2 L_\tau(\boldsymbol{\theta}) / \partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^\top$. This minimization may be done using existing software for the GLASSO for linear regression applied to pseudodata: letting $\ddot{L}_\tau(\widetilde{\boldsymbol{\theta}}) = \widetilde{X}_{\text{pseudo}}^\top \widetilde{X}_{\text{pseudo}}$, and $\widetilde{Y}_{\text{pseudo}} = \widetilde{X}_{\text{pseudo}} \widetilde{\boldsymbol{\theta}}$, we have $Q(\boldsymbol{\theta}) = \frac{1}{2}(\widetilde{X}_{\text{pseudo}}\boldsymbol{\theta} - \widetilde{Y}_{\text{pseudo}})^\top (\widetilde{X}_{\text{pseudo}}\boldsymbol{\theta} - \widetilde{Y}_{\text{pseudo}}) + \sum_{m=1}^{M} \lambda_m \sqrt{\boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m}$, a standard least squares formulation. Minimizing $Q(\boldsymbol{\theta})$ will result in an estimator $\hat{\boldsymbol{\theta}}$ where some pathways may have all coefficients set to 0. We could then use $\hat{\boldsymbol{\theta}}$ as our estimate of $\boldsymbol{\theta}$, or we could iterate the procedure, restricting the data to the retained pathways to re-estimate $\widetilde{\boldsymbol{\theta}}$, and then repeating the least squares approximation, potentially repeating until the collection of pathways has stabilized.

2.4. Step-by-Step Description of the Survival MKL Procedure

In this section, we describe in detail the steps of our proposed procedure, including information about how we choose tuning parameters. We assume that we have a data set with survival outcomes, genomic measurements, and (perhaps) clinical covariates to be included in the model; we call this the training data. We may also have a validation data set with the same structure, to be used to evaluate our survival model. We select a model of interest – methods for the Cox model and the AFT model are described in Section 3 and supported in the R package `survivalMKL`. We identify a collection of pathways of interest: these would preferably be candidate pathways thought to be relevant for a particular disease based on prior disease studies or basic science research. In the absence of such prior information, a pathway database such as the Biocarta database, available on mSigDB, could be used [1]. We also select a kernel of interest to model each pathway. Researchers may use subject matter knowledge to decide on which kernel best captures similarity in their data, or be guided by the scope of signals they wish to consider (e.g., linear or nonlinear). Here, since we focus on gene expression data, we recommend using the Gaussian kernel, which through its tuning parameter can be used to flexibly capture different sorts of linear and nonlinear effects. Other types of genomic data have other natural kernel choices – for example, the identity-by-state kernel has been used frequently for genotype data.

We next provide a step-by-step description of our proposed survival MKL prediction procedure.

1. **Optional marginal pathway screening.** If a small number of candidate pathways (e.g., 5–20) have been identified in previous studies, no preliminary pathway screening needs to be done. However, if the number of candidate pathways is moderate or large (e.g., over 30) or a pathway database is used, and if it is reasonable to hypothesize that the number of pathways related to survival is much smaller than the total number of pathways under consideration, pathways should be screened for their potential to improve risk prediction. This screening step can both improve computation time of the method and risk prediction accuracy of the final model. The screening can be performed based on KM tests that provide a p-value for the association between a given pathway and the outcome. The p-value threshold for significance should not be overly

stringent and could be based on the nominal p-value, or based on maintaining a desired false discovery rate [42]. In our numerical studies when 30 pathways are under consideration, and in our data examples where 32 and 217 pathways are under consideration, we perform this step and we retain any pathway with nominal p-value < 0.05.

2. **Building the design matrix.** For each pathway, we construct the kernel matrix $\mathbb{K}_m = \mathbb{K}_m(\rho_m)$ (where selection of ρ_m is described below) and perform kernel PCA: we find $\tilde{\mathbb{B}}_m$ such that $\mathbb{K}_m = \tilde{\mathbb{B}}_m \tilde{\mathbb{B}}_m^\top$, and then truncate $\tilde{\mathbb{B}}_m$ to $\tilde{\mathbb{B}}_{mr_m}$, where r_m is the smallest number of eigenvectors whose eigenvalues account for $\mathfrak{p}$ of the total eigenvalue sum; we find $\mathfrak{p} = 0.90$ provides a nice balance of dimension reduction and relevant feature retention. Then, we construct the design matrix by column concatenation: $\tilde{\mathbb{W}} = (\mathscr{D}\ \tilde{\mathbb{B}}_{1r_1}\ \tilde{\mathbb{B}}_{2r_2} \cdots \tilde{\mathbb{B}}_{Mr_M}) = (\mathbb{D}\ \tilde{\mathbb{B}})$

3. **Finding a preliminary ℓ_2-penalized coefficient estimate.** We next find a preliminary estimate $\tilde{\boldsymbol{\theta}}$ that relates the design matrix $\tilde{\mathbb{W}}$ to survival by minimizing $\mathscr{L}_\tau(\boldsymbol{\theta})$. The least squares approximation method relies on a preliminary consistent estimate of the coefficient vector, so in practice, rather than tune τ, we have found it sufficient to choose a value of τ that is large enough that the model fits without overshrinking the coefficients too much towards the null. We accomplish this by requiring the effective degrees of freedom to be a particular value – we find that $\widehat{\mathrm{df}}(\tau) = 0.9 \min\{\sum_{i=1}^{n} \Delta_i, p_z\}$ works well in practice, where $\widehat{\mathrm{df}}(\tau) = p_D + \mathrm{tr}\, \tilde{\mathbb{B}}^\top \tilde{\mathbb{B}} \left(\tilde{\mathbb{B}}^\top \tilde{\mathbb{B}} + n\tau^2 I\right)^{-1}$.

4. **One-Step MKL fit.** We use the preliminary estimator $\tilde{\boldsymbol{\theta}}$ to construct $Q(\boldsymbol{\theta})$ defined in Section 2.3, and minimize it with respect to $\boldsymbol{\theta}$.We set the pathway-specific tuning parameters λ_m in $Q(\boldsymbol{\theta})$ to be $\lambda_m = \lambda \frac{\sqrt{r_m}}{\|\tilde{\boldsymbol{\theta}}_m\|_2}$, where $\|\boldsymbol{\theta}\|_2 = \sqrt{\boldsymbol{\theta}^\top \boldsymbol{\theta}}$.

   This choice of λ_m mimics the choice typically used in adaptive procedures, such as the AENet, wherein feature-specific tuning parameters proportional to the inverse of a consistent estimator have been shown to give variable selection and estimation procedures desirable properties such as the oracle property [37, 43]. We find that λ is best chosen by (bootstrap) cross-validation (CV), using the unpenalized objective function $\mathscr{L}_0$ as a model fit criterion. After λ is selected, we can find $\hat{\boldsymbol{\theta}}_{\mathrm{MKL}} = \mathrm{argmin}_{\boldsymbol{\theta}} Q(\boldsymbol{\theta})$, which will have zero entries for the coefficients associated with any pathways that were eliminated by the choice of λ. At this point, we can stop: we have estimated the model coefficients, and can use them to estimate a risk score for a new patients through (6).

5. **Optional Iteration of Steps 3 and 4.** We also considered an iterative strategy, in which we exclude the pathways that were eliminated by the MKL step, and perform Steps 3 and 4 again, until the number of pathways retained is the same across two iterations.

Step 1 above employs a marginal KM score test for screening. KM score tests under the Cox and AFT models are available [12, 18]. Each test relates survival to a single pathway –

specifically, to a signal of the form $\boldsymbol{\theta}_0^\top \mathbf{d} + h_m(\mathbf{z}_{g_m})$– and tests $H_0 : h_m = 0$, where $h_m \in \mathcal{H}_{K_m}$.

For a particular kernel tuning parameter ρ_m, each test is based on a score statistic – say, $\hat{\mathcal{Q}}_m(\rho_m)$. The distribution of $\hat{\mathcal{Q}}_m(\rho_m)$ under the null is approximated by perturbation resampling; thus, we could calculate, say, $\widehat{\mathcal{Q}}_m^{*(1)}(\rho_m), \ldots, \widehat{\mathcal{Q}}_m^{*(B)}(\rho_m)$ for some large number B to approximate the null distribution of $\hat{\mathcal{Q}}_m(\rho_m)$. The p-value for the test is ultimately estimated by $\hat{p}_m(\rho_m) = \#\left\{\widehat{\mathcal{Q}}_m^{*(b)}(\rho_m) \geq \widehat{\mathcal{Q}}_m^{\text{obs}}(\rho_m)\right\}/B$. To test across a range of kernel tuning parameters ρ_m, we can rely on a supremum statistic — for instance, $\hat{S}_m = \sup_{\rho_m} \hat{Q}_m(\rho_m)$. To find its null distribution, we use the same perturbation resamples across the range of ρ_m, and calculate $\widehat{S}_m^{*(b)} = \sup_{\rho_m} \widehat{\mathcal{Q}}_m^{*(b)}(\rho_m)$, to get an approximate null distribution $\widehat{S}_m^{*(1)}, \ldots, \widehat{S}_m^{*(B)}$ to use to calculate p-values $\hat{p}_m$. Thus, for kernels that require tuning, these tests produce both a p-value for each tuning parameter ρ under consideration, which can be used to determine the values of ρ for which the data evinces the most evidence against the null, as well as a single summary p-value for the pathway that adjusts for the step of searching across the range of ρ. To determine the range of ρ_m, we choose the lower and upper bound such that the associated number of eigenvectors included to capture $\mathfrak{p} = 0.90$ of the eigenvalues, r_m, is between 2 and $\max\left\{\sqrt{\sum_{i=1}^n \Delta_i}, \sqrt{p_m}\right\}$, where p_m is the number of genes in pathway m, which we find to be sufficiently flexible.

3. MKL under the Cox and AFT Models

3.1. Cox PH Model

The Cox PH KM model with M pathways assumes:

$$\lambda_i(t) = \lambda_0(t)\exp\{\boldsymbol{\theta}_0^\top \mathbf{d}_i + h_1(\mathbf{z}_{g_1,i}) + \cdots + h_M(\mathbf{z}_{g_M,i})\},\ i = 1, \ldots, n$$

where $\lambda_i(t)$ is the hazard that person i has an event at time t given their covariates $\mathbf{w}_i$ and $\lambda_0(t)$ is a common baseline hazard function. Defining the usual counting and at risk processes $N_i(t) = \Delta_i \mathbf{I}(X_i \leq t)$ and $Y_i(t) = \mathbf{I}(X_i \geq t)$, we can let $L_\tau(\boldsymbol{\theta})$ be the penalized log partial likelihood function:

$$L_\tau(\boldsymbol{\theta}) = \sum_{i=1}^n \int \left[\boldsymbol{\theta}^\top \tilde{\mathbf{w}}_i - \log\left\{S^{(0)}(\boldsymbol{\theta}, s)\right\}\right] dN_i(s) + \tau^2 \sum_{m=1}^M \boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m .$$

where the $\tilde{\mathbf{w}}_i$ are the rows of the (transformed) design matrix $\widetilde{\mathbb{W}}$, $S^{(k)}(\boldsymbol{\theta}, s) = \sum_{l=1}^n Y_l(s)\exp(\boldsymbol{\theta}^\top \tilde{\mathbf{w}}_l)\tilde{\mathbf{w}}_l^{\otimes k}$, and for any vector $\boldsymbol{a}$, $\boldsymbol{a}^{\otimes 0} = 1$, $\boldsymbol{a}^{\otimes 1} = \boldsymbol{a}$ and $\boldsymbol{a}^{\otimes 2} = \boldsymbol{a}\boldsymbol{a}^\top$. Then

$$\dot{L}_\tau(\boldsymbol{\theta}) = \sum_{i=1}^n \int \left\{\tilde{\mathbf{w}}_i - \frac{S^{(1)}(\boldsymbol{\theta}, s)}{S^{(0)}(\boldsymbol{\theta}, s)}\right\} dN_i(s) + 2\tau^2 \mathbb{G}\boldsymbol{\theta}$$

and

$$\ddot{L}_\tau(\boldsymbol{\theta}) = \sum_{i=1}^{n} \int \left\{ \frac{S^{(2)}(\boldsymbol{\theta}, s)}{S^{(0)}(\boldsymbol{\theta}, s)} - \frac{S^{(1)}(\boldsymbol{\theta}, s)^{\otimes 2}}{S^{(0)}(\boldsymbol{\theta}, s)^2} \right\} dN_i(s) + 2\tau^2 \mathbb{G}$$

where $\mathbb{G}$ is the diagonal matrix whose first p_d diagonal entries are 0 and whose remaining diagonal entries are 1. KM testing for individual pathways can be done using the method developed in [12].

3.2. AFT model

The AFT-KM model with M pathways assumes:

$$\log T_i = \boldsymbol{\theta}_0^\top \mathbf{d}_i + h_1(\mathbf{z}_{g_1, i}) + \cdots + h_M(\mathbf{z}_{g_M, i}) + E_i, \; i = 1, \ldots, n \quad (7)$$

where E_i is an iid error term independent of $\mathbf{w}_i = (\mathbf{d}_i,^\top \mathbf{z}_i^\top)^\top$ with completely unspecified distribution. For this model, we can let $L_\tau(\boldsymbol{\theta})$ be the penalized Gehan objective function:

$$L_\tau(\boldsymbol{\theta}) = n^{-2} \sum_{i=1}^{n} \sum_{j=1}^{n} \Delta_i \mid \widetilde{e}_j(\boldsymbol{\theta}) - \widetilde{e}_i(\boldsymbol{\theta}) \mid_+ + \tau^2 \sum_{m=1}^{M} \boldsymbol{\theta}_m^\top \boldsymbol{\theta}_m \quad (8)$$

where $\tilde{e}_i(\boldsymbol{\theta}) = \log X_i - \boldsymbol{\theta}^\top \tilde{\mathbf{w}}_i$. Unfortunately, this objective function is not twice differentiable, so our procedure does not directly apply. To remedy this, we perform a smoothing step, following reasoning similar to that in Brown and Wang [44]. Specifically, the gradient of $L_\tau(\boldsymbol{\theta})$ is:

$$\dot{L}_\tau(\boldsymbol{\theta}) = n^{-2} \sum_{i=1}^{n} \sum_{j=1}^{n} \Delta_i(\tilde{\mathbf{w}}_i - \tilde{\mathbf{w}}_j)\mathbf{I}\{\widetilde{e}_j(\boldsymbol{\theta}) - \widetilde{e}_i(\boldsymbol{\theta}) > 0\} + 2\tau^2 \mathbb{G}\boldsymbol{\theta} \quad (9)$$

where $\mathbb{G}$ is as defined above. The function $\dot{L}$ has jumps because of the indicator function $\mathbf{I}(\cdot)$, so to smooth $\dot{L}_\tau$ we could replace $\mathbf{I}\{\tilde{e}_j(\boldsymbol{\theta}) - \tilde{e}_i(\boldsymbol{\theta}) > 0\}$ by $\Phi\left\{\frac{e_j(\boldsymbol{\theta}) - e_i(\boldsymbol{\theta})}{\sigma_n}\right\}$ where Φ is some continuous cdf and σ_n is a bandwidth parameter. Here, we will take Φ to be the standard normal cdf, and use as a bandwidth $\sigma_n = \text{s.d.}\{e_j(\widetilde{\boldsymbol{\theta}})\} \times n^{-\frac{1}{3}}$. We choose this bandwidth with under-smoothing to eliminate the potential bias induced by smoothing [45]. Our smoothed version of $\dot{L}_\tau$ is $\dot{L}_\tau^{sm}$:

$$\dot{L}_{\tau}^{sm}(\boldsymbol{\theta}) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}\Delta_i(\widetilde{\mathbf{w}}_i - \widetilde{\mathbf{w}}_j)\Phi\left\{\frac{e_j(\boldsymbol{\theta}) - e_i(\boldsymbol{\theta})}{\sigma_n}\right\} + 2\tau^2\mathbb{G}\boldsymbol{\theta} \quad (10)$$

We can now take a further derivative:

$$\ddot{L}_{\tau}^{sm}(\boldsymbol{\theta}) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\Delta_i}{\sigma_n}\phi\left\{\frac{\widetilde{e}_j(\boldsymbol{\theta}) - \widetilde{e}_i(\boldsymbol{\theta})}{\sigma_n}\right\}(\widetilde{\mathbf{w}}_i - \widetilde{\mathbf{w}}_j)(\widetilde{\mathbf{w}}_i - \widetilde{\mathbf{w}}_j)^{\top} + 2\tau^2\mathbb{G}. \quad (11)$$

Finally, we can check that $\dot{L}_{\tau}^{sm}(\boldsymbol{\theta})$ is the gradient of the smoothed Gehan objective function defined by:

$$L_{\tau}^{sm}(\boldsymbol{\theta}) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}\Delta_i\left[\{e_j(\boldsymbol{\theta}) - e_i(\boldsymbol{\theta})\}\Phi\left\{\frac{e_j(\boldsymbol{\theta}) - e_i(\boldsymbol{\theta})}{\sigma_n}\right\} + \sigma_n\phi\left\{\frac{e_j(\boldsymbol{\theta}) - e_i(\boldsymbol{\theta})}{\sigma_n}\right\}\right] + \tau^2\sum_{m=1}^{M}\boldsymbol{\theta}_m^{\top}\boldsymbol{\theta}_m.$$

KM testing for individual pathways can be done using methods in [18].

4. Simulation Studies

To assess the performance of the proposed procedure, we conducted simulation studies. All simulations are based on the tumor gene expression available from 522 ovarian cancer patients from TCGA, as accessible through the `curatedOvarianData` R package [46, 47]. In each simulation, patients were randomly partitioned into a training set of 372 and a validation set of 150. Gene expression values were used directly, so all between- and within-pathway correlation structures reflect those patterns in actual data. Survival outcomes were generated by simulation, according to models detailed below.

We considered settings with either 10 or 30 initial pathways; these pathways were selected from the Biocarta pathway database to represent features of that data set (with respect to pathway sizes and overlap). Three pathways were selected to be causal. For the set of 10 pathways, pathway size ranged from 7 to 39 (median 17.5); the causal pathways were of size 7, 16, and 23, and among the 168 genes involved in these pathways, 143 belonged to only one pathway, 20 belonged to two pathways, and 5 belonged to three. For the set of 30 pathways, pathway size ranged from 9 to 36 (median 16); the causal pathways were of size 12, 18, and 28; and among the 320 genes involved in these pathways, 222 were in one pathway, 58 were in two, and the remaining 40 genes were in three or more pathways (up to nine pathways).

We generated data according to the model

$$\log T = h(\mathbf{z}_{g_1}, c_1) + h(\mathbf{z}_{g_2}, c_2) + h(\mathbf{z}_{g_3}, c_3) + E,$$

where E was generated from the extreme value distribution; for this choice of error, both the Cox and AFT models hold. The signal function h was selected as either the linear function $h(\mathbf{z}, c) = c\mathbf{1}^\top\mathbf{z}$ or the nonlinear function $h(\mathbf{z},c) = c\|\mathbf{z}\|$. The constant c was selected to achieve a certain signal strength and to allocate that signal evenly across the genes in the pathway: that is, for a pathway with p_m genes in it, we set $c_m = \sqrt{c_0^2/p_m}$. For weak linear signal, we set $c_0 = 0.5$; for strong linear signal, $c_0 = 1$; for weak nonlinear signal, $c_0 = 3$; and for strong nonlinear signal, $c_0 = 4$. We generated censoring variables from a uniform distribution on $[0, \tau]$, where τ was selected in each simulation setting to produce approximately 50% censoring.

The methods we compare are:

- The MKL method, as outlined in Section 2.4, under each of the Cox and AFT models, using each of the Gaussian and linear kernels.
- The Cox AENet, with preliminary coefficients estimated using an ℓ_2 penalization analogous to that done in Step 4 of the MKL procedure. The ℓ_1 tuning parameter is estimated using cross-validation. The AENet assumes (sparse) linear effects and does not account for group structure; It is known to perform effective variable selection and estimation even in the presence of correlation among predictors.
- The Cox 1 KM method, which groups all the genes in the pathways together into a single pathway, and summarizes their signal using a single kernel. For this method we considered both the linear and Gaussian kernels; all tuning parameters ($\mathfrak{p}$, ρ, and τ) are selected analogously to Steps 2 and 4 of the MKL procedure.

Finally, we note that in the 10 pathway/168 gene setting, we apply the method without any pathway screening (i.e., we skip Step 1 in Section 2.4. In the 30 pathway/320 gene setting, we screen the pathways as described there, retaining those with nominal $p < 0.05$ for the MKL methods. We do the same screening for individual genes for the AENet method, retaining genes with individual marginal $p < 0.05$.

For each method, we built the model in the training data, and evaluated its prediction accuracy in the validation data by a C-statistic

$$C_{t_0} = P\left\{\hat{\boldsymbol{\theta}}_0^\top \mathbf{d}_i + \hat{\mathbf{h}}(\mathbf{z}_i) > \hat{\boldsymbol{\theta}}_0^\top \mathbf{d}_j + \hat{\mathbf{h}}(\mathbf{z}_j) \,\middle|\, T_i > T_j, T_j < t_0\right\},$$

which captures how well the order of the associated risk predictions corresponds to the order of the true survival times during a pre-specified follow-up period $(0, t_0)$. The time t_0 was pre-selected as the 70th percentile of X. We estimate C_{t_0} using the nonparametric estimator proposed in Uno et al. [48]. Under each simulation configuration and each method, we present the average C-statistic across simulations; the variability in that C-statistic is quantified by the standard deviation across simulations. For the MKL methods, we also present the average number of pathways selected and the variability in that metric across simulations. Results are based on 100 simulations run in each setting. Results for the linear settings are presented in Figure 1; results for the nonlinear settings are presented in Figure 2.

We will begin by commenting on performance aspects of the MKL methods; we will subsequently compare their performance to other existing methods. First, in all settings, the MKL step outperforms the initial ℓ_2-penalized fit (labeled as "Preliminary"). This suggests that the MKL step of eliminating some unnecessary or redundant pathways improves the model, even when pathways are first screening marginally. Second, the model fit with one MKL step and the model fit by iterating the MKL fit until the number of pathways converges produce models with similar performance. In fact, iterating can sometimes reduce prediction performance in the validation data. We suspect that this is because iteratively re-estimating coefficients based on previous coefficient estimates could potentially proliferate errors in estimation or be more plagued by over-fitting than a single MKL fit. For example, this problem occurs most consistently with the AFT model fit using the Gaussian kernel – we suspect this is because estimation under the AFT is more variable (compared with the Cox model) at a given sample size, and because, particularly with a kernel with a tuning parameter and with iteration, the model is tuned and re-tuned using the training data and may be slightly over-fit. Thus, we recommend using the one-step fit – although iterating can reduce the number of pathways, it never improves prediction accuracy dramatically, and in some circumstances can make it worse.

The Cox model generally slightly out-performs the AFT model with slightly higher prediction accuracy based on fewer included pathways. The Gaussian kernel performs similarly to the linear kernel when the signal is linear and outperforms the linear kernel when the signal is nonlinear. For instance, under the Cox MKL method, when the signal is weak and linear, the linear and Gaussian kernels produce nearly identical C-statistics; when the signal is strong and linear, the linear kernel produces C-statistics of 80 and 86 when the number of pathways is 10 and 30, while the Gaussian kernel produces C-statistics of 79 and 85, only a slight deflation. This is due to the fact that the Gaussian kernel approximates the linear kernel when ρ is large. When the signal is nonlinear, the Gaussian kernel's C-statistics beat the linear kernel's C-statistics by between 9 and 16 points. Thus, unless prior knowledge points to a preference for the AFT model or the linear kernel, we recommend for the MKL method using the Cox model with a Gaussian kernel summarizing each pathway effect.

Next, we compare the MKL method (focusing on the Cox One Step MKL with Gaussian kernel) to other competing methods (the AENet and the 1 KM methods). When the signal is linear, the AENet and the overall linear kernel models both perform quite well. The Cox One Step MKL beats them slightly when there are 10 pathways; this reflects a key benefit of the MKL approach — that it can take advantage of the grouping information to efficiently remove entire pathways of non-informative markers. When there are 30 pathways and the MKL and AENet methods are used after preliminary marginal screens, the AENet, as well as the overall linear KM method, beat the Cox MKL method slightly – though the differences noted here are at most 1 point. Interestingly, the overall Gaussian kernel underperforms in this setting, coming in at least a few points below the other methods. When the signal is nonlinear, the Cox MKL with Gaussian kernel handily beats the AENet and the

Overall Linear kernel methods, and continues to outperform the overall Gaussian kernel method by a similar margin as in the the linear setting. This demonstrates the potential for gain when leveraging pathway structure when the signal does arise from a small subset of specific pathways.

5. Data Examples

5.1. Data Example 1: Ovarian Cancer Gene Expression Study

Ovarian cancer is the tenth most common cancer among U.S. women, and the fifth leading cause of cancer death [49]. Many studies have sought to identify genes whose tumor expression is predictive of prognosis. We focus on the data set assembled for TCGA, as accessible through the R package `curatedOvarianData`, which has served as the basis for the simulations presented earlier [46, 47]. The outcome was progression-free survival, which was available for 522 individuals, with 372 selected at random for inclusion in the training data and 150 for inclusion in the validation data. In the validation data, models were assessed with the C-statistic for recurrence or death up through t_0 = 5 years; this metric assesses how well the model can order the patients' survival time during the first five years [48]. In the random partition used, 68% (67%) of women experienced progression or death in the training (validation) sets. Based on our findings from the simulation studies, we focus the KM-based analyses on the Gaussian kernel.

We approached this analysis agnostically, using the 217 pathways in the Biocarta database which were of sizes 6–85 genes (median 17 genes); a total of 1178 genes were involved in these pathways. Separately for the Cox and AFT models, we implemented the approach outlined in Section 2.4, performing a preliminary marginal test for each pathway based on B =10,000 perturbations, retaining those pathways with nominal p-value < 0.05, and then performing the MKL procedure. The results are presented in the upper panel of Table 1. When applying the MKL method under the Cox model, 44 pathways had marginal p-value < 0.05, and 16 of these pathways were retained after the MKL step, comprising 211 genes; the C-statistic in the validation data was 61% (95% confidence interval [CI]: 54–67 %). The MKL method under the AFT model produced a smaller model with discrimination of 57% (95% CI: 50–64%) in the validation data. We compared these results with models that do not use information about the potential pathway structure of the gene expression values. A Cox model fit using a single kernel to summarize all 1178 genes had discrimination of 55% (95% CI 48–65%). For the Cox model fit with the AENet penalization, we first screened genes individually and found 99 genes with nominal p-value less than 0.05. These were allowed to enter the model fit with AENet penalization, which produced a final model with 49 genes that had a C-statistic in the validation data of 53% (95% CI 44–63%).

Although all CIs quoted here overlap, these results suggest that the Cox MKL method performs quite well relative to other methods. The improvement in C-statistic of KM-based methods over the AENet could indicate that the pathways contain many genes with small effects rather than a few genes with large effects, which could be more easily picked up by the AENet. The MKL methods have the added benefit of being potentially more interpretable than a single KM model, if the selected pathways are meaningful to scientists; they also include fewer genes, which could be helpful in downstream development of tools with clinical applicability. The pathways selected by the MKL Cox model are provided in the Web Appendix.

5.2. Data Example 2: Breast Cancer Gene Expression Study

In breast cancer, gene expression signatures have begun to be used to help doctors better predict risk of recurrence among patients, but a majority of early-stage breast cancer patients are given adjuvant therapy in addition to local treatment, even though this additional therapy has some negative side-effects and is believed to benefit only a small fraction of these patients [50]. Identifying more predictive biomarkers of aggressive tumors could help doctors better identify patients who could potentially avoid adjuvant therapy. We sought to build models predicting recurrence-free survival. We trained each model among 286 lymph node negative patients from the study described in Wang et al. [51]. A total of 107 deaths or recurrences were observed, so that 63% of observations were censored. We evaluated model performance in an independent set of 119 lymph node negative patients with gene expression assessed on the same chip [52]. The validation set had 27 deaths or recurrences, and thus 77% censoring. We assessed the accuracy of each model in the validation data using the C-statistic for recurrence up through $t_0 = 5$ years and calculated 95% CIs [48], and all KM-based methods were implemented with the Gaussian kernel.

Here, we consider the 32 candidate pathways considered individually in [18]; these pathways ranged in size from 7 to 238 genes, with a median size of 23.5; a total of 788 genes were involved in these pathways. We performed the approach described in Section 2.4, performing a preliminary marginal test for each pathway based on B =10,000 perturbations, retaining those pathways with nominal p-value < 0.05, and then performing the MKL procedure separately for the Cox and AFT models. The results are presented in the lower panel of Table 1. We compared results with methods using only a single kernel to summarize all 788 genes under the Cox model, and the AENet-penalized Cox model, fit among genes with marginal p-value less than 0.05.

The top performing methods in this analysis were the MKL methods under either the Cox or the AFT model: for the Cox model, 22 pathways had marginal p-value < 0.05, 17 of these were retained after the MKL step comprising 517 genes, and the C-statistic in the validation data was 72% (95% CI: 60–84 %). For the AFT model, 23 pathways had marginal p-value < 0.05, but none were removed in the MKL step, resulting in a model based on 616 genes with a C-statistic of 72% (95% CI: 61–84%). The Cox model using a single Gaussian kernel, ignoring the pathway structure, had a C-statistic of 70% (95% CI: 55–85%), while the AENet-penalized model, which included 61 genes selected from the 131 genes with marginal p-value less than 0.05, had a C-statistic of 67% (95% CI: 54–80%).

As in the Ovarian cancer example, the CIs quoted here overlap; however, the results again suggest that the MKL method under the Cox model with the Gaussian kernel produces a more interpretable model with potentially better prediction performance than other methods, based on a smaller number of genes. The pathways selected by this model are provided in the Web Appendix.

6. Discussion

In this paper, we proposed KM-based procedures to build risk prediction models by selecting and combining information from multiple pathways. By taking a pathway-based approach, we take advantage of pre-existing biological knowledge and privilege groups of genes believed to work together. By working with flexible kernels such as the Gaussian kernel, we can capture both linear and nonlinear effects well. In settings where disease progression operates through dysregulation of key pathways, building a pathway-level risk prediction model could be more meaningful and interpretable than gene-level models. Such models could also be more predictive of prognosis if the pathways are composed of many genes with small effects; if different subsets of genes in a pathway are dysregulated in different groups of patients; or if there are indeed substantial complex, nonlinear effects of genes on outcome. On the other hand, in settings where the underlying signal is sparse, with only a few genes associated with survival, gene-level models assuming linear effects, such as the AENet, may perform better. It would be interesting to extend our proposed procedure to allow for feature selection under the KM framework, such as that discussed in Allen [53]. Such an extension could potentially gain some parsimony benefits enjoyed by methods such as the hierarchical lasso [25], while allowing for potentially nonlinear, complex effects.

The model as currently formulated assumes that effects are additive across pathways, but it would be interesting to extend the method to incorporate interactions between pathways via approaches such as tensor kernel regression [54, 55]. Additionally, the current implementation focuses on modeling pathway effects using the Gaussian kernel which we find works well in practice for gene expression data, but clever kernel choices for other types of biological data and effective approaches for optimal kernel selection are key areas of future research.

We should note that the proposed method can be somewhat computationally intensive. To construct the design matrix $\tilde{\mathbb{W}}$, KM score tests for each pathway are run to evaluate pathway significance and select tuning parameters (as needed); because the score tests rely on resampling, this step can take time, though it is highly parallelizable since the computations for each pathway can be carried out independently. After the design matrix is constructed, the MKL penalized fit is found with tuning parameters selected by cross-validation. This can also take time in settings where many pathways are included and many eigenvectors are selected to represent each pathway, but the cross-validation steps can also be parallelized to reduce runtime. For reference, on a 2017 Macbook Pro laptop, using *no* parallelization, under the simulation setting with nonlinear signal from 3 of 10 pathways, it took 2.4 minutes to build the design matrix for the MKL procedure under the Cox model with Gaussian kernel. That matrix ultimately included 130 columns, and the final MKL fit took an additional 42 seconds to run (including tuning parameter selection by cross-validation). By comparison, the AENet penalized Cox model took only 3 seconds to fit. In this setting, the MKL method produces a much better model, but it is certainly the case that incorporating the pathway structure and allowing for potentially nonlinear pathway effects estimated in a data-adaptive way comes at an increased computational cost.

Our method is built on the premise that integrating prior biological knowledge about pathways or other gene groupings is useful. However, it is unclear to what extent pathways in existing databases are correctly specified or misspecified, nor is it clear what method of defining a gene set is best for an application like predicting risk from pathway gene expression in tumors. Pathway collections available in mSigDB include those based on position on the chromosome; gene ontology; observed changes when known cancer genes are perturbed experimentally; and curation from the literature. We have focused on collections of pathways in this last category, but it's possible a different method of grouping genes might be more useful for risk prediction. Depending on which collection of pathways are used, certain genes may belong to no pathways. Such genes may either be excluded from consideration altogether, or included as pathways of size one. Better understanding of the relevant annotations and best practices for pathway building require further research.

When the goal is risk prediction, inclusion of relevant clinical covariates is important. In our formulation of the MKL method, clinical covariates may be included as linear effects in the risk score, which is likely sufficient for covariates such as age and gender. However, for other variables, such as treatment, this formulation may not be sufficient. For example, it may be the case that the effect of a biological pathway on survival may differ for patients undergoing different treatment regimes. Our model can be extended to allow for this possibility by including a treatment effect, a pathway effect, and a treatment-pathway interaction effect. How best to fit such a model when many pathways are under consideration is a direction of future research.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

JAS was supported by the National Institutes of Health (NIH) grant T32 CA09001 and the A. David Mazzone Career Development Award. TC was supported by the NIH grants R01 GM079330, R01 HL089778, and U54 H6007963.

References

1. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci U S A. Oct 15; 2005 102(43):545–50. DOI: 10.1073/pnas.0506580102
2. McLendon R, Friedman A, Bigner D, Van Meir EG, Brat DJ, Mastrogianakis GM, Olson JJ, Mikkelsen T, Lehman N, Aldape K, et al. Comprehensive genomic characterization defines human glioblastoma genes and core pathways. Nature. 2008; 455(7216):1061–1068. [PubMed: 18772890]
3. Jones S, Zhang X, Parsons DW, Lin JCH, Leary RJ, Angenendt P, Mankoo P, Carter H, Kamiyama H, Jimeno A, et al. Core signaling pathways in human pancreatic cancers revealed by global genomic analyses. Science. 2008; 321(5897):1801–1806. [PubMed: 18772397]
4. Goeman JJ, Van De Geer SA, De Kort F, Van Houwelingen HC. A global test for groups of genes: testing association with a clinical outcome. Bioinformatics. 2004; 20(1):93–99. [PubMed: 14693814]
5. Scholkopf, B., Smola, A. Learning with kernels. MIT Press; Cambridge, Mass: 2002.

6. Liu D, Lin X, Ghosh D. Semiparametric regression of multidimensional genetic pathway data: least-squares kernel machines and linear mixed models. Biometrics. Dec; 2007 63(4):1079–88. DOI: 10.1111/j.1541-0420.2007.00799.x [PubMed: 18078480]
7. Liu D, Ghosh D, Lin X. Estimation and testing for the effect of a genetic pathway on a disease outcome using logistic kernel machine regression via logistic mixed models. BMC Bioinformatics. 2008; 9:292.doi: 10.1186/1471-2105-9-292 [PubMed: 18577223]
8. Mika S, Schölkopf B, Smola A, Müller K, Scholz M, Rätsch G. Kernel pca and de-noising in feature spaces. Advances in neural information processing systems. 1999; 11(1):536–542.
9. Schölkopf B, Smola A, Müller K. Nonlinear component analysis as a kernel eigenvalue problem. Neural computation. 1998; 10(5):1299–1319.
10. Cox DR. Regression models and life tables. J R Statist Soc B. 1972; 34:187–220.
11. Li H, Luan Y. Kernel cox regression models for linking gene expression profiles to censored survival data. Pac Symp Biocomput. 2003:65–76. [PubMed: 12603018]
12. Cai T, Tonini G, Lin X. Kernel machine approach to testing the significance of multiple genetic markers for risk prediction. Biometrics. 2011; 67(3):975–986. [PubMed: 21281275]
13. Neykov M, Hejblum BP, Sinnott JA. Kernel machine score test for pathway analysis in the presence of semi-competing risks. Statistical methods in medical research. 2016:427. 0962280216653.
14. Tsiatis A. Estimating regression parameters using linear rank tests for censored data. The Annals of Statistics. 1990:354–372.
15. Ritov Y. Estimation in a linear regression model with censored data. The Annals of Statistics. 1990:303–328.
16. Jin Z, Lin D, Wei L, Ying Z. Rank-based inference for the accelerated failure time model. Biometrika. 2003; 90(2):341.
17. Liu Z, Chen D, Tan M, Jiang F, Gartenhaus RB. Kernel based methods for accelerated failure time model with ultra-high dimensional data. BMC Bioinformatics. 2010; 11:606.doi: 10.1186/1471-2105-11-606 [PubMed: 21176134]
18. Sinnott JA, Cai T. Omnibus risk assessment via accelerated failure time kernel machine modeling. Biometrics. 2013; 69(4):861–873. [PubMed: 24328713]
19. Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2006; 68(1):49–67.
20. Wang H, Leng C. A note on adaptive group lasso. Computational Statistics & Data Analysis. 2008; 52(12):5277–5286.
21. Meier L, Van De Geer S, Bühlmann P. The group lasso for logistic regression. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2008; 70(1):53–71.
22. Nardi Y, Rinaldo A, et al. On the asymptotic properties of the group lasso estimator for linear models. Electronic Journal of Statistics. 2008; 2:605–633.
23. Jacob, L., Obozinski, G., Vert, J. Group lasso with overlap and graph lasso. Proceedings of the 26th Annual International Conference on Machine Learning; ACM; 2009. p. 433-440.
24. Luan Y, Li H. Group additive regression models for genomic data analysis. Biostatistics. 2008; 9(1):100–113. [PubMed: 17513311]
25. Wang S, Nan B, Zhu N, Zhu J. Hierarchically penalized cox regression with grouped variables. Biometrika. 2009; 96(2):307–322.
26. Wu TT, Wang S. Doubly regularized cox regression for high-dimensional survival data with group structures. Statistics and Its Interface. 2013; 6:175–186.
27. Eng KH, Wang S, Bradley WH, Rader JS, Kendziorski C. Pathway index models for construction of patient-specific risk profiles. Statistics in medicine. 2013; 32(9):1524–1535. [PubMed: 23074142]
28. Bach, F., Lanckriet, G., Jordan, M. Multiple kernel learning, conic duality, and the smo algorithm. Proceedings of the twenty-first international conference on Machine learning; ACM; 2004. p. 6
29. Lanckriet G, Cristianini N, Bartlett P, Ghaoui L, Jordan M. Learning the kernel matrix with semidefinite programming. The Journal of Machine Learning Research. 2004; 5:27–72.

30. Lanckriet, G., Deng, M., Cristianini, N., Jordan, M., Noble, W., et al. Proceedings of the Pacific Symposium on Biocomputing. Vol. 9. World Scientific Singapore; 2004. Kernel-based data fusion and its application to protein function prediction in yeast; p. 2
31. Sonnenburg S, Rätsch G, Schäfer C, Schölkopf B. Large scale multiple kernel learning. The Journal of Machine Learning Research. 2006; 7:1531–1565.
32. Bach F. Consistency of the group lasso and multiple kernel learning. The Journal of Machine Learning Research. 2008; 9:1179–1225.
33. Koltchinskii V, Yuan M, et al. Sparsity in multiple kernel learning. The Annals of Statistics. 2010; 38(6):3660–3695.
34. Seoane JA, Day IN, Gaunt TR, Campbell C. A pathway-based data integration framework for prediction of disease progression. Bioinformatics. 2014; 30(6):838–845. [PubMed: 24162466]
35. Wang H, Leng C. Unified lasso estimation by least squares approximation. Journal of the American Statistical Association. 2007; 102(479):1039–1048.
36. Zhang H, Lu W. Adaptive lasso for cox’s proportional hazards model. Biometrika. 2007; 94(3): 691–703.
37. Zou H, Hastie T. Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2005; 67(2):301–320.
38. Kimeldorf G, Wahba G. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. The Annals of Mathematical Statistics. 1970; 41(2):495–502.
39. Koltchinskii V, Giné E. Random matrix approximation of spectra of integral operators. Bernoulli. 2000; 6(1):113–167.
40. Braun, M. PhD Thesis. University of Bonn; 2005. Spectral properties of the kernel matrix and their application to kernel methods in machine learning.
41. Rasmussen, C., Williams, C. Gaussian processes for machine learning 2006. Vol. 38. The MIT Press; Cambridge, MA, USA: 2006. p. 715-719.
42. Benjamini Y, Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the royal statistical society. Series B (Methodological). 1995:289–300.
43. Zou H, Zhang HH. On the adaptive elastic-net with a diverging number of parameters. Annals of statistics. 2009; 37(4):1733. [PubMed: 20445770]
44. Brown B, Wang Y. Induced smoothing for rank regression with censored survival times. Statistics in medicine. 2007; 26(4):828–836. [PubMed: 16646005]
45. van der Vaart A. Weak convergence of smoothed empirical processes. Scandinavian Journal of Statistics. 1994:501–504.
46. Cancer Genome Atlas Research Network. Integrated genomic analyses of ovarian carcinoma. Nature. 2011; 474(7353):609–615. [PubMed: 21720365]
47. Ganzfried BF, Riester M, Haibe-Kains B, Risch T, Tyekucheva S, Jazic I, Wang XV, Ahmadifar M, Birrer M, Parmigiani G, et al. curatedovariandata: Clinically annotated data for the ovarian cancer transcriptome. Database. 2013; 2013 URL http://database.oxfordjournals.org/content/2013/bat013.abstract. doi: 10.1093/database/bat013
48. Uno H, Cai T, Pencina MJ, D’Agostino RB, Wei LJ. On the c-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. Stat Med. May; 2011 30(10): 1105–17. DOI: 10.1002/sim.4154 [PubMed: 21484848]
49. US Cancer Statistics Working Group. Technical Report. Department of Health and Human Services, Centers for Disease Control and Prevention, and National Cancer Institute; 2017. United states cancer statistics: 1999–2014 incidence and mortality web-based report. https://www.cdc.gov/cancer/ovarian/statistics/index.htm
50. Reis-Filho J, Pusztai L. Gene expression profiling in breast cancer: classification, prognostication, and prediction. The Lancet. 2011; 378(9805):1812–1823.
51. Wang Y, Klijn J, Zhang Y, Sieuwerts A, Look M, Yang F, Talantov D, Timmermans M, Meijer-van Gelder M, Yu J, et al. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. The Lancet. 2005; 365(9460):671–679.

52. Sotiriou C, Wirapati P, Loi S, Harris A, Fox S, Smeds J, Nordgren H, Farmer P, Praz V, Haibe-Kains B, et al. Gene expression profiling in breast cancer: understanding the molecular basis of histologic grade to improve prognosis. Journal of the National Cancer Institute. 2006; 98(4):262. [PubMed: 16478745]

53. Allen GI. Automatic feature selection via weighted kernels and regularization. Journal of Computational and Graphical Statistics. 2013; 22(2):284–299.

54. Hardoon DR, Shawe-Taylor J. Decomposing the tensor kernel support vector machine for neuroscience data with structured labels. Machine Learning. 2010; 79(1–2):29–46.

55. Lee, YK., Beng Jin Teoh, A., Toh, KA. Joint kernel collaborative representation on tensor manifold for face recognition. Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on; IEEE; 2014. p. 6245-6249.

56. Shen Y, Cai T. Identifying predictive markers for personalized treatment selection. Biometrics. 2016; 72(4):1017–1025. [PubMed: 26999054]

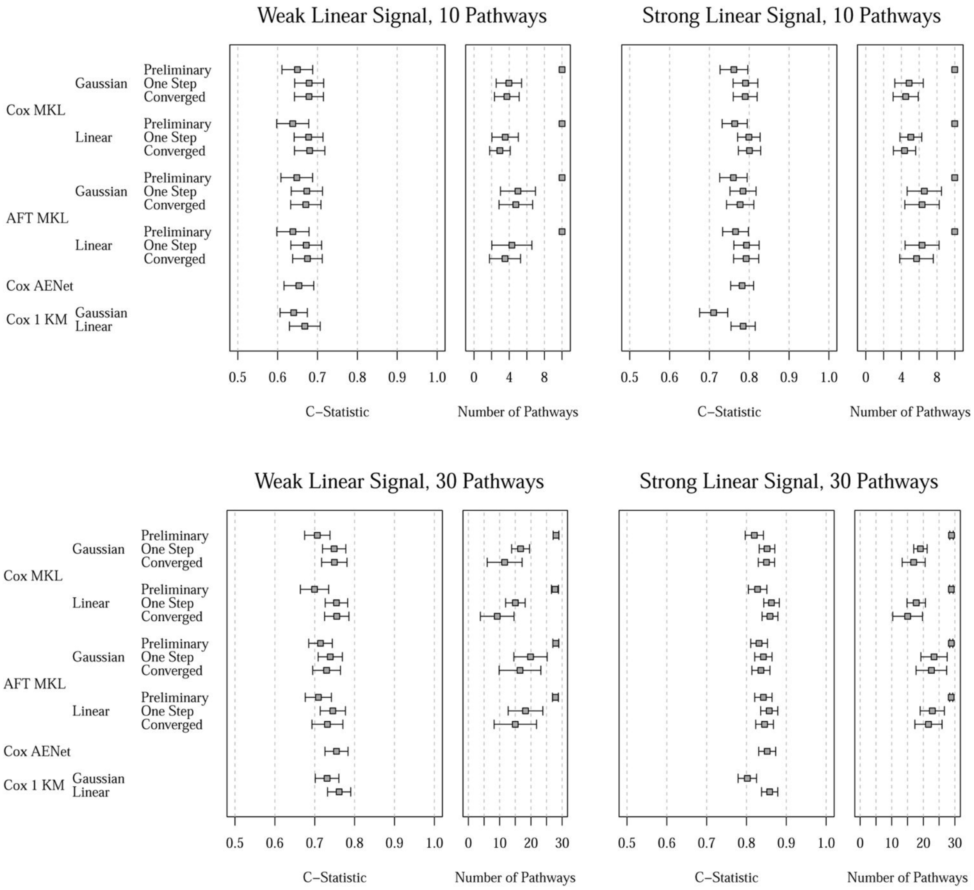

Figure 1.

Presented are results for the simulation studies described in Section 4, in which the relationships between the three causal pathways and survival are *linear*. The upper panel assumes 10 pathways to start, and there is no screening of pathways or genes; the lower panel assumes 30 pathways to start, and begins by eliminating pathways with p-value > 0.05 (MKL methods) or genes with p-value > 0.05 (Cox AENet method). The left and right panels differ based on the strength of the pathway signals. Presented are the estimated C-statistics in a validation set, as well as the number of pathways included in the models for all MKL fits. Also compared are the Cox AENet fit, and KM fits that ignore pathway structure (labeled Cox 1 KM). Those KM-based models estimate a single $h \in \mathcal{H}_K$ based on all the genes that make up the pathways. For the MKL methods, “Preliminary” refers to the initial ℓ_2-penalized fit with all pathways (that survived screening, if there was a screening step); “One Step” refers to a single MKL step in which pathway selection and estimation is done; and “Converged” refers to the final model selected if the MKL step is repeated until the number of pathways retained stabilizes.

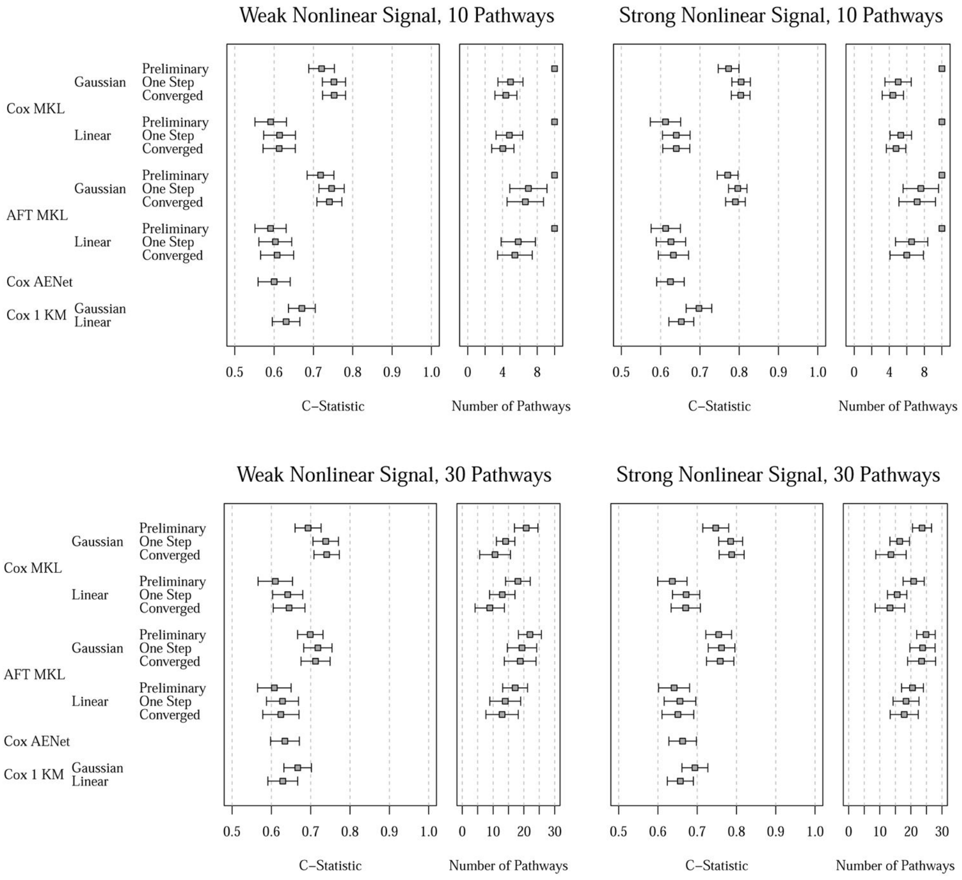

Figure 2.

Presented are results for the simulation studies described in Section 4, in which the relationships between the three causal pathways and survival are *nonlinear*. All methods and metrics are the same as those in Figure 1.

Table 1

Results in real data examples. Methods compared are the MKL method using the Gaussian kernel under the Cox and AFT models; a single KM method ignoring the pathway structure using the Gaussian kernel under the Cox model; and a Cox model with AENet penalty. Presented are the number of pathways or genes that screen positive with nominal marginal $p < 0.05$; the number of pathways or genes included in the final model; and the estimated *C*-statistics in the validation set.

Ovarian Cancer Data Set — 217 Initial Pathways from Biocarta

Method	**Model**	**Screen Positive**	**Final Model**	**C (Validation)**
MKL	Cox	44 pathways	16 pathways/211 genes	61 (54, 67)
	AFT	31 pathways	7 pathways/111 genes	57 (50, 64)
1 KM	Cox		1178 genes	55 (48, 65)
AEnet	Cox	99 genes	49 genes	53 (44, 63)

Breast Cancer Data Set — 32 Initial Candidate Pathways

Method	Model	Screen Positive	Final Model	C (Validation)
MKL	Cox	22 pathways	17 pathways/517 genes	72 (60, 84)
	AFT	23 pathways	23 pathways/616 genes	72 (61, 84)
1 KM	Cox		788 genes	70 (55, 85)
AEnet	Cox	131 genes	61 genes	67 (54, 80)