# Genomic Meta-Feature Guided Regularized Regression for Survival Outcome

Dixin Shen
Division of Biostatistics, University of Southern California
and
Juan Pablo Lewinger
Division of Biostatistics, University of Southern California

February 25, 2021

**Abstract**

In building predictive models for genomic studies, regularized regression is a common technique as the number of genomic features is much larger than the number of samples. The application of sparse regularized regression performs feature selection while doing prediction. Associated with genomic features, such as gene expression, genetic variation, DNA methylation, there are plenty of meta-features. Some examples are functional gene sets, gene ontology annotations, knowledge of past studies. The canonical way is to modeling genomic features on phenotypic outcomes, and post hoc analysis with meta-features, like gene set enrichment analysis. However, incorporating meta-features into modeling process can potentially improve the quality of both prediction performance and feature selection.

In this paper, we extend the approach of Zeng et al. (2020) to survival outcome. The method incorporates genomic meta-features to guide the regularized Cox regression, so that each of the genomic features has its own customized penalty parameter, as opposed to one common penalty parameter for all features. With highly informative meta-features, significant features become more important/being penalized less, unrelated features become less important/heavily penalized, thereby achieving improved feature selection. The general purpose of prediction performance is also improved with the extra information. We show the benefits of the method by simulations and applications in genetic studies. Model optimization algorithm involves Laplace approximation of the likelihood function, and a majorization-minimization procedure.

## 1 Introduction

Predicting a phenotypic outcome based on genomic features is a highly active research area, with the increasing need in personalized health care to achieve the most possible benefits out of a medical intervention for a particular patient. A common technique for genomic predictive modeling is regularized regression. As the number of genomic features is typically large, thousands to millions, linear models like regression are better suited, since the data pattern is most likely linear where each feature contribute a little or none effect to the outcome. When number of features is larger than the number of samples, which is usually the case for genomics study, regularization needs to be introduced so that the model can be fit. Sparse regularized regression is a popular choice, as it not only shrinks the regression coefficients to make the model simpler, it also shrink some of the coefficients with little effects on the outcome to exactly zero, thereby performing feature selection. Typical examples are the lasso (Tibshirani, 1996) and the elastic net (Zou and Hastie, 2005). While they have similar mechanics, there is a major difference. If some of the features among which the correlations with each other are high, the lasso tends to select one of them, while the elastic net tends to select most of them and share the lasso value equally. Ridge regression (Hoerl and Kennard, 1970) is another regularization technique to cope with high dimension and collinearity of data. However, it shrinks the coefficients but not to zero, hence does not produce interpretable model.

Genomic features may have their own characteristics: grouping effect and ordering. Extensions of the lasso deal with such situations. The group lasso (Yuan and Lin, 2006) takes in the grouping information, shrinking coefficients by group. All the coefficients in one group are either zero or nonzero. Sparse group lasso (Simon et al., 2013) further allows sparsity within group. The fused lasso deals with the ordering situation, with the addition of $L_1$ terms for the differences of neighbouring coefficients, which allows sparsity in the their differences. The above extended regularization methods take into account characteristics of features, which are essentially features of the features if they are put in a data set. We call these extra characteristics of features meta-features here and after. There are plenty of such meta-features in genomics. For example, functional gene sets like hallmark (Liberzon et al., 2015), gene ontology pathways like reactome (Jassal et al., 2020) work as grouping effects; summary statistics like p-values and regression coefficients from meta-analyses work as ordering effect (p-values and regression coefficients indicate the importance of each feature). The meta-features are actual data matrices, where the samples/rows represent original features, and the columns represent meta-features. However, none of the above regularization approaches systematically utilize the meta-feature information. The group lasso assumes features are in different groups mutually exclusive. Features in multiple groups at the same time does not meet the assumption. The fused lasso takes the ordering of the features into account, but when provided with concrete information like p-values indicating exactly how important each feature are, it cannot incorporate such information.

One way of utilizing the meta-features is modeling with original features, and performing gene set enrichment analysis (Subramanian et al., 2005) in hindsight. However, incorporating meta-features into modeling process can potentially improve both prediction performance and the quality of feature selection. Weaver et al. (reference) and Shen et al. (reference) incorporate the meta-features in a hierarchical modeling setup. The outcomes are regressed on original features, assuming quantitative outcome,

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

where $\boldsymbol{Y}$ is the length $n$ outcome vector, $\boldsymbol{X}$ is the $n \times p$ data matrix, $\boldsymbol{\beta}$ is the length $p$ feature coefficient vector to be estimated in the model. Then the feature coefficients $\boldsymbol{\beta}$ are regressed on meta-features,

$$\boldsymbol{\beta} = \boldsymbol{Z}\boldsymbol{\alpha} + \boldsymbol{\gamma}$$

where $\boldsymbol{Z}$ is the $p \times q$ meta-feature matrix, $\boldsymbol{\alpha}$ is coefficients vector for meta-features. To integrate both level of features into modeling process, an objective function is formed as below

$$\min_{\boldsymbol{\beta},\boldsymbol{\alpha}} \frac{1}{2n}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \frac{\lambda_1}{2}\|\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{\alpha}\|_2^2 + \lambda_2\|\boldsymbol{\alpha}\|_1$$

The original feature data $\boldsymbol{X}$ and meta-feature data $\boldsymbol{Z}$ are fitted through two least squares above. The additional $L_1$ term of meta-feature coefficients $\boldsymbol{\alpha}$ is to control model complexity and meta-feature selection. This integration method incorporates meta-features linearly, emphasizing on meta-feature selection. It was shown to improve the prediction performance considerably with high quality meta-features. Zeng et al. (2020) developed another method for integrating meta-features $\boldsymbol{Z}$ for quantitative and binary outcomes, in a non-linear way, such that each of the original features has its own customized penalty parameter, as opposed to one common penalty parameter for all features. The customized penalty parameters potentially allow more accurate feature selection. In this paper, we extend this approach to survival outcomes.

# 2 Methods

## 2.1 Model setup and notations

Starting with the survival model setup, let the outcome data be $(\boldsymbol{y}, \boldsymbol{\delta})$ where $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ is observed time, $\delta = (\delta_1, \delta_2, \ldots, \delta_2)$ is censoring status. If $\delta_i = 1 (i = 1, 2, \ldots, n)$, event happened, $y_i$ is event time; if $\delta_i = 0$, event did not happen in experimental period, $y_i$ is censoring time. There are $p$ features for the n instances/samples, data matrix $\boldsymbol{X}$ with dimension $n \times p$ stores the feature values, i.e., $\boldsymbol{x}_i$ is a vector of feature values for instance $i$ $(x_{i1}, x_{i2}, \ldots, x_{ip})$. Associated with the features, there are q meta-features. A $p \times q$ matrix $\boldsymbol{Z}$ stores the meta-feature values for the $p$ original features, i.e., $\boldsymbol{z}_j (j = 1, 2, \ldots, p)$ is a vector of meta-feature values for feature $j$ $(z_{j1}, z_{j2}, \ldots, z_{jq})$. The common choice of regression method is Cox's proportional hazards model (Cox, 1972). It assumes hazard functions are proportional at the same time point, which allows model fitting without knowing explicit form of baseline hazard function, and only depends on the order in which events occur, not on the exact time of occurrence. To illustrate, let $t_1 < t_2 < \cdots < t_l < \cdots < t_m$ be the the unique event times arranged in increasing order, and $D_l = \{i : \delta_i = 1, y_i = t_l\}$ is the set of instances experienced event at time $t_l$. Let $\boldsymbol{\beta}$ be a length $p$ vector for the regression coefficients for features. The partial likelihood function, $L(\boldsymbol{\beta})$, takes the form

$$L(\boldsymbol{\beta}) = \prod_{l=1}^{m} \frac{e^{\sum_{i \in D_l} \boldsymbol{x}_i^T \boldsymbol{\beta}}}{(\sum_{i \in R_l} e^{\boldsymbol{x}_i^T \boldsymbol{\beta}})^{d_l}}$$

where $R_l = \{i : y_i \geq t_l\}$ is the risk set at event time $t_l$, i.e., the set of all instances who have not experienced the event and are uncensored just prior to time $t_l$; $d_l = |D_l|$ is the number of events at time $t_l$. $L(\boldsymbol{\beta})$ is Breslow's adjustment of partial likelihood (Breslow, 1972). It deals with ties in each event time ($d_l > 1$: more than one instance experienced event at a particular event time). When there are no ties ($d_l = 1$), $L(\boldsymbol{\beta})$ automatically reduces to Cox's partial likelihood. We can see that neither hazard functions nor times are involved in the function, only the order of event times matters.

We add regularization to Cox regression to control model complexity. Denote the log of partial likelihood as $\ell(\boldsymbol{\beta})$,

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ -\ell(\boldsymbol{\beta}) + \lambda \left[ \frac{1}{2}(1 - c)\|\beta\|_2^2 + c\|\boldsymbol{\beta}\|_1 \right] \right\}. \tag{1}$$

The regularization function covers lasso, elastic net, ridge penalties, i.e., $c = 1$ represents the lasso,

$c = 0$ represents ridge, and $0 < c < 1$ represents the elastic net. When $0 < c \leq 1$, it is sparse regularization which shrink some coefficients to exactly zero, producing interpretale model. The regularization has a universal penalty parameter $\lambda$ for all the features. This ignores underlying characteristics of features assuming each of them are equally important by applying the same amount of penalty. Our idea is to incorporate informative meta-features which might indicate the importance of the original features, giving each of them a unique penalty parameter $\lambda_j$. To incorporate meta-features to $\lambda_j$, first form a linear combination of $\boldsymbol{z_j}$ for feature $j$, $\boldsymbol{\alpha}$ is the weight vector of length $q$; then give it a non-linear function by expenentiating it

$$\begin{aligned} &\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ -\ell(\boldsymbol{\beta}) + \sum_{j=1}^{p} \lambda_j \left[ \frac{1}{2}(1-c)\beta_j^2 + c|\beta_j| \right] \right\}, \\ &\lambda_j = e^{\boldsymbol{z_j}^T \boldsymbol{\alpha}}. \end{aligned} \tag{2}$$

## 2.2 Model fitting

The standard regularized Cox proportional hazards model, equation (1), is fitted with pathwise coordinate descent (Simon et al., 2011). As the universal penalty parameter $\lambda$ is a hyper-parameter, the algorithm constructs a $\lambda$ path to tune via cross-validation. The proposed model, equation (2), has $p$ $\lambda$'s decided by weights $\boldsymbol{\alpha}$, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_p) = e^{\boldsymbol{Z\alpha}}$, it is impossible to tune them. Instead, we estimate the weights $\boldsymbol{\alpha}$ first to get the values of $\boldsymbol{\lambda}$. With known $\boldsymbol{\lambda}$, we can fit the model via coordinate descent.

### 2.2.1 Empirical Bayes objective function for estimation of $\boldsymbol{\alpha}$

We need an objective function to optimize $\boldsymbol{\alpha}$. Since the regularized regression has a natural Bayesian interpretation, we apply empirical Bayes estimation of hyper-parameters in random effects model, which is maximizing marginal likelihood in terms of hyper-parameter $\boldsymbol{\alpha}$ obtained by integrating out random effects $\boldsymbol{\beta}$. Based on the Bayesian elsatic net (Li et al., 2010), equation (2) has the interpretation

$$f(\boldsymbol{Y}|\boldsymbol{\beta}; \boldsymbol{X}) = L(\boldsymbol{\beta}), \tag{3}$$

$$\pi(\beta_j; \boldsymbol{\alpha}) \propto exp \left\{ -\lambda_j \left[ \frac{1}{2}(1-c)\beta_j^2 + c|\beta_j| \right] \right\}. \tag{4}$$

With the likelihood (3) and prior distribution (4), we construct the joint distribution of $\boldsymbol{Y}$ and $\boldsymbol{\beta}$, and integrate out $\boldsymbol{\beta}$, so to get the marginal likelihood of $\boldsymbol{Y}$,

$$\begin{aligned}\ln f(\boldsymbol{Y};\boldsymbol{\alpha}) &= \int_{\boldsymbol{\beta}\in\mathbb{R}^p} \ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha}) d\boldsymbol{\beta} \\ &= \int_{\boldsymbol{\beta}\in\mathbb{R}^p} \left[\ln f(\boldsymbol{Y}|\boldsymbol{\beta};\boldsymbol{X}) + \ln \pi(\boldsymbol{\beta};\boldsymbol{\alpha})\right] d\boldsymbol{\beta} \\ &= \int_{\boldsymbol{\beta}\in\mathbb{R}^p} \left\{ \sum_{l=1}^{m} \left[ \sum_{i\in D_l} \boldsymbol{x}_i^T\boldsymbol{\beta} - d_l \ln(\sum_{i\in R_l} e^{\boldsymbol{x}_i^T\boldsymbol{\beta}}) \right] - \sum_{j=1}^{p} \lambda_j \left[ \frac{1}{2}(1-c)\beta_j^2 + c|\beta_j| \right] + \text{const} \right\} d\boldsymbol{\beta}\end{aligned}$$

This integral does not have a closed form expression because the elastic net prior is not a conjugate prior for the likelihood. We propose two approximation methods: first approximate the elastic net prior to a normal prior, then apply Laplace approximation. To approximate the elastic net prior, we follow Zeng et al. (2021) (reference),

$$\pi(\beta_j;\boldsymbol{\alpha}) = N(0, \frac{2}{2\lambda_j(1-c)+c^2\lambda_j^2}). \tag{5}$$

Equation (5) gives a similar variance to that of the elastic net prior. The joint distribution, $\ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha})$, then takes the form

$$\begin{aligned}\ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha}) &= \sum_{l=1}^{m} \left[ \sum_{i\in D_l} \boldsymbol{x}_i^T\boldsymbol{\beta} - d_l \ln(\sum_{i\in R_l} e^{\boldsymbol{x}_i^T\boldsymbol{\beta}}) \right] - \sum_{j=1}^{p} \frac{1}{2} v_j \beta_j^2 + \text{const}, \\ v_j &= \frac{2\lambda_j(1-c)+c^2\lambda_j^2}{2}.\end{aligned} \tag{6}$$

This is essentially a ridge regularized Cox regression with customized penalty vector. For Laplace approximation of the marginal likelihood/model evidence, we elaborate the details. Consider a Taylor series of $\ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha})$ at the stationary point $\widetilde{\boldsymbol{\beta}}$, where $\nabla \ln f(\boldsymbol{Y},\widetilde{\boldsymbol{\beta}};\boldsymbol{\alpha}) = 0$,

$$\ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha}) \approx \ln f(\boldsymbol{Y},\widetilde{\boldsymbol{\beta}};\boldsymbol{\alpha}) - \frac{1}{2}(\boldsymbol{\beta}-\widetilde{\boldsymbol{\beta}})^T \boldsymbol{H} (\boldsymbol{\beta}-\widetilde{\boldsymbol{\beta}}).$$

$\widetilde{\boldsymbol{\beta}}$ is the solution of a ridge regularized Cox regression as already stated, it can be computed using **R** language *glmnet* package (Simon et al., 2011), with known $\boldsymbol{\alpha}$. $\boldsymbol{H}$ is the Hessian matrix,

$$\begin{aligned}\boldsymbol{H} &= -\nabla\nabla \ln f(\boldsymbol{Y},\boldsymbol{\beta};\boldsymbol{\alpha})|_{\boldsymbol{\beta}=\widetilde{\boldsymbol{\beta}}} \\ &\approx \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X} + \boldsymbol{V}\end{aligned}$$

where $\boldsymbol{V} = \mathrm{diag}[v_1, \ldots, v_p]$, $\boldsymbol{W}$ is a diagonal matrix with elements

$$\boldsymbol{W}_{ii} = \sum_{l \in C_i} \frac{d_l e^{\boldsymbol{x}_i^T \boldsymbol{\beta}}}{\sum_{k \in R_l} e^{\boldsymbol{x}_k^T \boldsymbol{\beta}}} - \sum_{l \in C_i} \frac{d_l (e^{\boldsymbol{x}_i^T \boldsymbol{\beta}})^2}{(\sum_{k \in R_l} e^{\boldsymbol{x}_k^T \boldsymbol{\beta}})^2}.$$

The Hessian is an approximation because $W$ is in fact a full matrix with high computational cost. We only use diagonal elements to speed up computation without much loss of accuracy. For greater details, refer to Shen en al. (2001) (reference). Now that we see $f(\boldsymbol{Y}, \boldsymbol{\beta}; \boldsymbol{\alpha})$'s Taylor approximation has a multivariate normal form with mean $\widetilde{\boldsymbol{\beta}}$, variance $\boldsymbol{H}^{-1}$, integrating out $\boldsymbol{\beta}$ returns the normalizing constant.

$$\begin{aligned} -\ln f(\boldsymbol{Y}; \boldsymbol{\alpha}) &\approx -\ln f(\boldsymbol{Y} | \widetilde{\boldsymbol{\beta}}; \boldsymbol{X}) - \ln \pi(\widetilde{\boldsymbol{\beta}}; \boldsymbol{\alpha}) - \frac{p}{2} \ln 2\pi + \frac{1}{2} \ln |\boldsymbol{H}| \\ &= -\ln |\boldsymbol{V}| + \widetilde{\boldsymbol{\beta}}^T \boldsymbol{V} \widetilde{\boldsymbol{\beta}} + \ln |\boldsymbol{H}| + \mathrm{const} \end{aligned} \tag{7}$$

The approximate negative log marginal likelihood, equation (7), is the objective function we are going to minimize with respect to $\boldsymbol{\alpha}$.

### 2.2.2 Objective function optimization

The objective function, equation (7), is nonconvex. In particular, it can decomposed as difference of two convex functions. $g(\boldsymbol{\alpha}) := -\ln |\boldsymbol{V}| + \widetilde{\beta_j}^T \boldsymbol{V} \widetilde{\beta_j}$ is convex in $\boldsymbol{\alpha}$, where as $h(\boldsymbol{\alpha}) := \ln |\boldsymbol{H}|$ is concave. This makes it a proper candidate to apply difference of convex functions algorithm (DCA) (Le Thi et al., 2015). The principle idea of DCA is to approximate the nonconvex objective function by a sequence of convex ones: at each iteration of the sequence, approximate the concave part by its affine majorization, i.e., the supporting hyperplane obtained by calculating its gradient, or subgradients if not differentiable, and minimize the resulting convex approximation. Note that it is also an application of majorization-minimization algorithm (Hunter and Lange, 2004). The affine approximation of the concave part is the majorization step, which forms a surface lying above the objective function, and is tangent to it, i.e, at the current estimation of the target parameter, the majorization equals to the objective function. This ensures the majorization is a tight upperbound for the objective. Minimizing the convex upperbound is the minimization step. The DCA algorithm for the marginal likelihood, $-\ln f(\boldsymbol{Y}; \boldsymbol{\alpha})$:

1. Initialize $\boldsymbol{\alpha}$ with $\widetilde{\boldsymbol{\alpha}} \in \mathbb{R}^q$.

2. Majorization:

- calculate the gradient at current estimation $\widetilde{\boldsymbol{\alpha}}$,

$$\boldsymbol{\theta} = \nabla_{\boldsymbol{\alpha}} \ln |\boldsymbol{H}| = \mathrm{diag}[\boldsymbol{H}^{-1}]$$

- form the convex upperbound,

$$\begin{aligned} u(\boldsymbol{\alpha}) &= +g(\boldsymbol{\alpha}) + h(\widetilde{\boldsymbol{\alpha}}) + \boldsymbol{\theta}^T(\boldsymbol{\alpha} - \widetilde{\boldsymbol{\alpha}}) \\ &= -\ln |\boldsymbol{V}| + \widetilde{\boldsymbol{\beta}}^T \boldsymbol{V} \widetilde{\boldsymbol{\beta}} + \boldsymbol{\theta}^T \boldsymbol{\alpha} + \mathrm{const} \end{aligned}$$

3. Minimization: $\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \{u(\boldsymbol{\alpha})\}$.

4. Set $\widetilde{\boldsymbol{\alpha}} = \hat{\boldsymbol{\alpha}}$.

5. Repeat step 2-4 until convergence of $\hat{\boldsymbol{\alpha}}$.

The minimization of $u(\boldsymbol{\alpha})$ can be processed with standard first order method like gradient descent, or second order method like Newton-Raphson. We show the gradient and Hessian here,

$$\begin{aligned} \nabla_{\boldsymbol{\alpha}} u(\boldsymbol{\alpha}) &= \boldsymbol{Z}^T \left[ (-\frac{1}{\boldsymbol{v}} + \widetilde{\boldsymbol{\beta}}^2 + \boldsymbol{\theta})((1-c)\boldsymbol{\lambda} + c^2\boldsymbol{\lambda}^2) \right], \\ \nabla\nabla_{\boldsymbol{\alpha}} u(\boldsymbol{\alpha}) &= \boldsymbol{Z}^T \mathrm{diag} \left[ \frac{\boldsymbol{\lambda}^2}{\boldsymbol{v}^2}(1 - c + c^2\boldsymbol{\lambda})^2 + (-\frac{1}{\boldsymbol{v}} + \widetilde{\boldsymbol{\beta}}^2 + \boldsymbol{\theta})\boldsymbol{\lambda}(1 - c + 2c^2\boldsymbol{\lambda}) \right] \boldsymbol{Z}. \end{aligned}$$

## 2.3 Summary

We incorporate the meta-features into the penalty parameter of regularized Cox proportional hazards model, as a log-linear function, to give each feature a unique penalty parameter depending on the meta-features. We then apply Bayesian interpretation of regularized regression to obtain the marginal likelihood function, as the objective function to optimize with respect to the introduced meta-feature weights $\boldsymbol{\alpha}$, thereby estimating the customized penalty parameter vector $\boldsymbol{\lambda}$. The nonconvex objective function can be decomposed to a difference of two convex functions, which can be solved with difference of convex functions algorithm. With estimated $\boldsymbol{\alpha}$, we can plug values of penalty parameters into the regularized Cox regression. The model fitting procedure is

1. Initialize $\boldsymbol{\alpha}$ with $\widetilde{\boldsymbol{\alpha}}$.

2. Repeat, until convergence of $\hat{\boldsymbol{\alpha}}$.

(a) Laplace approximation of marginal likelihood with known $\widetilde{\boldsymbol{\alpha}}$, section 2.2.1,

- Approximate the elastic net prior with a normal prior, equation (6),
- Calculate $\widetilde{\boldsymbol{\beta}}$ and $\boldsymbol{H}$.

(b) Optimize Laplace approximation of marginal likelihood, equation (7), get solution $\hat{\boldsymbol{\alpha}}$, with DCA described in section 2.2.2.

(c) Set $\widetilde{\boldsymbol{\alpha}} = \hat{\boldsymbol{\alpha}}$.

3. Calculate customized penalty vector $\boldsymbol{\lambda} = e^{\boldsymbol{Z}\hat{\boldsymbol{\alpha}}}$.

4. Fit regularized Cox regression, equation (2), with $\boldsymbol{\lambda}$.

# 3 Simulations

# 4 Applications

# 5 Discussion

# References

Chubing Zeng, Duncan Campbell Thomas, and Juan Pablo Lewinger. Incorporating prior knowledge into regularized regression. *Bioinformatics*, 09 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btaa776. URL `https://doi.org/10.1093/bioinformatics/btaa776`. btaa776.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of computational and graphical statistics*, 22(2):231–245, 2013.

Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P Mesirov, and Pablo Tamayo. The molecular signatures database hallmark gene set collection. *Cell systems*, 1(6):417–425, 2015.

Bijay Jassal, Lisa Matthews, Guilherme Viteri, Chuqiao Gong, Pascual Lorente, Antonio Fabregat, Konstantinos Sidiropoulos, Justin Cook, Marc Gillespie, Robin Haw, et al. The reactome pathway knowledgebase. *Nucleic acids research*, 48(D1):D498–D503, 2020.

Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

Norman E Breslow. Contribution to discussion of paper by dr cox. *J. Roy. Statist. Soc., Ser. B*, 34:216–217, 1972.

Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of statistical software*, 39(5):1, 2011.

Qing Li, Nan Lin, et al. The bayesian elastic net. *Bayesian analysis*, 5(1):151–170, 2010.

Hoai An Le Thi, T Pham Dinh, Hoai Minh Le, and Xuan Thanh Vo. Dc approximation approaches for sparse optimization. *European Journal of Operational Research*, 244(1):26–46, 2015.

David R Hunter and Kenneth Lange. A tutorial on mm algorithms. *The American Statistician*, 58(1):30–37, 2004.