Deutsches Krebsforschungszentrum
Praktikum

Final Project Report

Nida Murad

Cox Regression incorporating Sparse-Group Lasso penalization

Github: `https://github.com/nida612/DKFZ_praktikum`

Heidelberg University

April 29, 2023

Contents

1 Survival Analysis

1.1 Introduction & Background

Survival analysis is the phrase used to describe the analysis of data in the form of times from a well-defined time origin until the occurrence of some particular event or end-point. In medical research, the time origin will often correspond to the recruitment of an individual into an experimental study, such as a clinical trial to compare two or more treatments. This in turn may coincide with the diagnosis of a particular condition, the commencement of a treatment regimen or the occurrence of some adverse event. If the end- point is the death of a patient, the resulting data are literally survival times. However, data of a similar form can be obtained when the end-point is not fatal, such as the relief of pain, or the recurrence of symptoms. In this case, the observations are often referred to as time to event data, and there exist several methods for analysing such survival data[1]

1.1.1 Special features of survival data

The reasons why survival data are not amenable to standard statistical procedures used in data analysis, is that survival data are generally not symmetrically distributed. This difficulty could be resolved by first transforming the data to give a more symmetric distribution, for example by taking logarithms. However, a more satisfactory approach is to adopt an alternative distributional model for the original data.

Another main feature of survival data that renders standard methods inappropriate is that survival times are frequently censored. The survival time of an individual is said to be censored when the end-point of interest has not been observed for that individual. This may be because the data from a study are to be analysed at a point in time when some individuals are still alive. Alternatively, the survival status of an individual at the time of the analysis might not be known because that individual has been lost to follow-up.

Another form of censoring is left censoring, which is encountered when the actual survival time of an individual is less than that observed.[1].In each of these situations, a patient who entered a study at time to dies at time to+t. However, t is unknown. If the individual was last known to be alive at time to +c, the time e is called a censored survival time. This censoring occurs after the individual has been entered into a study, that is, to the right of the last known survival time, and is therefore known as right censoring.

1.2 Modelling Survival Data

In simple linear regression we examine the relation between a response variable and a single predictor. The model is

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

where i is the observation number, $i = 1, \ldots, n$, Y is the response, x is the predictor, β_0, β_1 are regression coefficients, called intercept and slope respectively, ε is an error term. Estimates of β_0 and β_1 (denoted by $\hat{\beta}_0$ and $\hat{\beta}_1$ respectively) are determined by a least squares approach. Note that $\hat{\beta}_1$ is the increase in Y corresponding to a 1-unit increase in x. Kaplan-Meier curves and logrank tests are examples of univariate analysis. They describe the survival according to one factor under investigation, but ignore the impact of any others.

Additionally, Kaplan-Meier curves and logrank tests are useful only when the predictor variable is categorical (e.g.: treatment A vs treatment B; males vs females). They don't work easily for quantitative predictors such as gene expression, weight, or age.

An alternative method is the Cox proportional hazards regression analysis[18], which works for both quantitative predictor variables and for categorical variables. Furthermore, the Cox regression model extends survival analysis methods to assess simultaneously the effect of several risk factors on survival time.

1.2.1 Penalised Cox proportional hazard model

The Cox proportional-hazards model [8] is essentially a regression model commonly used statistical in medical research for investigating the association between the survival time of patients and one or more predictor variables. It is common to model survival times through

the hazard function. The Cox proportional hazards model is given by,

$$h(t|x) = h_0(t)exp(\beta' x) \tag{1}$$

where t is the time, x the vector of covariates, β the vector of regression coefficients and $h_0(t)$ is the so-called baseline hazard function, i.e. the hazard function for x=0. As model (1) is formulated through the hazard function, the simulation of appropriate survival times for this model is not straightforward. One important issue in simulation studies regarding regression models is the knowledge of the true regression coefficients. This does not present a problem in a linear regression model, where the simulated variables are directly connected with the pre-specified regression coefficients. However, in the Cox model, the effect of the covariates have to be translated from the hazards to the survival times, because the usual software packages for Cox models require the individual survival time data, not the hazard function. The translation of the regression coefficients from hazard to survival time is easy if the baseline hazard function is constant, i.e. the survival times are exponentially distributed. This may be the reason why most simulation studies regarding the Cox model consider only the exponential distribution.[9]

1.2.2 Simulating Survival Times

The survival function of the Cox proportional hazards model (1) is given by,

$$S(t|x) = exp[-H_0(t)exp(\beta' x)] \tag{2}$$

where,

$$H_0(t) = \int_0^t h_0(u)du \tag{3}$$

is the cumulative baseline hazard function [10].
If $h_0(t) > 0$ for all t, then H_0 can be inverted and the survival time T of the Cox model (1) can be expressed as,

$$T = H_0^{-1}[-log(U)exp(-\beta' x)] \tag{4}$$

where U is a random variable with $U \sim U[0;1]$. By applying formula (4), uniformly distributed random numbers can be transformed into survival times following a specific Cox model. It is just required to insert the inverse of an appropriate cumulative baseline hazard function into equation (4). Hence, the survival time data for Cox models with exponentially (Cox–exponential model) distributed survival times can be generated as,

$$T = -\frac{log(U)}{\lambda exp(\beta' x)} \tag{5}$$

and having a baseline hazard rate modelled in a flexible parametric way dependent on the covariates x instead of (1),

$$h(t|x) = \lambda exp(\beta' x) \tag{6}$$

2 Variable Selection/Regularisation Methods

The standard linear model (or the ordinary least squares method) performs poorly in a situation, where you have a large multivariate data set containing a number of variables superior to the number of samples. A better alternative is the penalized regression allowing to create a regression model that is penalized, for having too many variables in the model, by adding a constraint in the equation. This is also known as shrinkage or regularization methods. The consequence of imposing this penalty, is to reduce (i.e. shrink) the coefficient values towards zero. This allows the less contributive variables to have a coefficient close to zero or equal zero. [17]

2.1 Lasso

Consider a sample consisting of $\mathbb{N}$ cases, each of which consists of p covariates and a single outcome. Let y_i be the outcome and $x_i = (x_1, x_2, ..., x_p)_i^T$ be the covariate vector for the i^{th} observation. Then the objective of lasso is to solve,

$$\min_{\beta_0,\beta} \sum_{n=1}^{N} (y_i - \beta_0 - x_i^T \beta)^2 \tag{7}$$

$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t$$

Here β_0 is the constant coefficient, $\beta = (\beta_1, \beta_2, \beta_3, ..., \beta_p)^T$ is the coefficient vector, and t is a pre-specified free parameter that determines the degree of regularization.[2]

Letting X be the covariate matrix, so that $X_{ij} = (x_i)_j$ and x_i^T is the i^{th} row of X, the expression can be written more compactly as,

$$\min_{\beta_0,\beta} ||y - \beta_0 - X\beta||_2^2 \tag{8}$$

$$\text{subject to } ||\beta||_1 \leq t$$

It can be expressed as the following Lagrangian form,

$$\min_{\beta \in \mathbb{R}^p} ||y - X\beta||_2^2 + \lambda ||\beta||_1 \tag{9}$$

where the exact relationship between t and λ is data dependent[2]. Lasso variants have been created in order to remedy limitations of the original technique and to make the method more useful for particular problems.

2.1.1 Group Lasso Regression

The estimates of group lasso have the attractive property of being invariant under group-wise orthogonal reparameterizations. In (10), it finds a solution with few nonzero entries in β. Suppose, further, that our predictor variables were divided into m different groups and rather than just sparsity in β we would like a solution that uses only a few of the groups. Yuan and Lin (2007) proposed the group-lasso criterion for this problem; the problem is,[4] [5]

$$\min_{\beta} \frac{1}{2} |||y - \sum_{l=1}^{m} X^{(l)} \beta^{(l)}||_2^2 + \lambda \sum_{l=1}^{m} \sqrt{p_l} ||\beta^{(l)}||_2 \tag{10}$$

where $X^{(l)}$ is the sub matrix of X with columns corresponding to the predictors in group l, $\beta^{(l)}$ the coefficient vector of that group, and p_l is the length of $\beta^{(l)}$. If the size of each group is 1, this gives us exactly the regular lasso solution. This criterion exploits the non differentiability of $||\beta^{(l)}||_2$ at $\beta^{(l)} = 0$, setting groups of coefficients to exactly 0. The sparsity of the solution is determined by the magnitude of the tuning parameter λ. [5] There were still certain drawbacks with this approach like,

a) The method does not yield sparsity within a group. If a group of parameters is non-zero, they will all be non-zero

b) Yuan & Lin's algorithm [4] assumed that submatrices in each group are always orthonormal.

Consequently, in order to combat the aforementioned problems and to gain both sparsity of groups and within each group, the method of Sparse-Group Lasso was proposed.

2.1.2 Sparse Group Lasso: Basic form

It is possible to extend the group lasso to the so-called sparse group lasso, which can select individual covariates within a group, by adding an additional ℓ^1 penalty to each group subspace[6].

While the group lasso gives a sparse set of groups, if it includes a group in the model then all coefficients in the group will be nonzero. Sometimes we would like both group wise sparsity and within group sparsity. Group wise sparsity refers to the number of groups with at least one nonzero coefficient, and within group sparsity refers to the number of nonzero coefficients within each nonzero group.

We choose $\hat{\beta}$ to minimize,

$$\frac{1}{2n} |||y - \sum_{l=1}^{m} X^{(l)} \beta^{(l)}||_2^2 + (1-\alpha)\lambda \sum_{l=1}^{m} \sqrt{p_l} ||\beta^{(l)}||_2 + \alpha\lambda ||\beta^{(l)}||_1 \tag{11}$$

where $\alpha \in [0, 1]$ denotes the tuning parameter and yields a convex combination of the lasso and group-lasso penalties. $\alpha = 0$ gives the group-lasso fit, $\alpha = 1$ gives the lasso fit. One might note that this looks very similar to the elastic net penalty [7] It differs because the $||\cdot||_2$ penalty is not differentiable at 0, so some groups are completely zeroed out. However, within each nonzero group it gives an elastic net fit (though with the $||\cdot||^2$ penalty parameter a function of the optimal $||\beta(k)||^2$)[5].

The optimal solution to the proposed criterion (11) is characterized by the subgradient equations. For a group $k, \bar{\beta}^{(k)}$ satisfies

$$\frac{1}{n} X^{(k)T} \left(y - \sum_{l=1}^{m} X^{(l)} \hat{\beta}^{(l)} \right) = (1-\alpha)\lambda\mu + \alpha\lambda v$$

where μ and v are subgradients of $\left\|\hat{\beta}^{(k)}\right\|_2$ and $\left\|\hat{\beta}^{(k)}\right\|_1$ respectively, evaluated at $\hat{\beta}^{(k)}$. Then, it can be shown that the subgradient equations can be satisfied with $\beta^{(k)} = 0$ if

$$\left\|S\left(X^{(k)T} r(-k)/n, \alpha\lambda\right)\right\|_2 \leq (1-\alpha)\lambda$$

where $r(-k)$ is the partial residual of y, subtracted from all other group fits except group k; and $S(\cdot)$ is the coordinate-wise soft thresholding operator [3].

2.1.3 Sparse Group Lasso: Methodology and Algorithm

Friedman et al. proposed pathwise coordinate gradient descent, using accelerated generalized descent with backtracking within each group. Since, the criteria is the sum of a convex differential function (the loss), and a separable penalty (between groups) the advantage of this method is that Blockwise Coordinate Gradient Descent is guaranteed to converge to the global optimum.

Additionally, instead of fixing the regularization parameter the authors proposed a method of finding a pathwise solution for various λ values, and by implementing warm-starts along each pathwise iteration the method is made efficient. This helps solve the problem of selecting optimal tuning parameters. The important techniques employed in the generalized gradient descent procedure to aid in faster convergence are Nesterov's Momentum and Backtracking.

Nesterov's momentum smooths updates by taking a step in the direction of the previous accumulated gradient then corrects the velocity based on this step. The previous gradients are accumulated using an exponential weighted moving average of the previous gradients [13]. This procedure reduces the stochasticity in the gradient update steps by dampening the updates when the optimization space is narrow (non-optimal) and increasing the magnitude of the updates when the conditions are optimal [13]. Back-tracking is a method for finding the optimal step size. This optimization method maximizes the step size in the direction of steepest descent [14]. Together these methods help the algorithm converge the optima faster.

Algorithm:

The algorithm consists of two loops, one over each of the groups and another over the parameters within each group. Blockwise-descent is used over the groups, and to solve within each group accelerated generalized gradient descent is employed [5] The algorithm is outlined as follows:

a) **Outer loop**: Cyclically iterate over the groups; at each group k to minimize over, consider the other group coefficients as fixed.

b) Check if the group's coefficients are exactly 0. If not, enter inner loop:

c) **Inner loop**: Start with $\beta^{(k,l)} = \theta^{(k,l)} = \beta_0^{(k)}$, step size $t = 1$, and counter $l = 1$. Unit convergence repeat:

(a) Update the gradient g by $g = \nabla l\left(r_{(-k)}, \beta^{(k,l)}\right)$

(b) Optimize step size by iterating $t = 0.8 * t$ until

$$l\left(U\left(\beta^{(k,l)}, t\right)\right) \leq l\left(\beta^{(k,l)}\right) + g^T \Delta_{(l,t)} + \frac{1}{2t}\left\|\Delta_{(l,t)}\right\|_2^2$$

(c) Update the center via a Nesterov step by

$$\beta^{(k,l+1)} \leftarrow \theta^{(k,l)} + \frac{l}{l+3}\left(\theta^{(k,l+1)} - \theta^{(k,l)}\right)$$

(d) Set $l = l + 1$
Note: $U(\beta_0, t)$ is the update rule, $\Delta_{(l,t)} = U\left(\beta^{(k,l)}, t\right) - \beta^{(k,l)}$

In the above algorithm the outer loop is optimizing over the groups using (block) coordinate-wise gradient descent, while the inner loop uses generalised gradient descent within each of the non-zero groupings. Additionally, we see at steps (b) backtracking being implemented, and at step (d) Nesterov's Momentum being applied.

Pathwise Solutions:

The algorithm also includes an additional step as mentioned before, it finds the pathwise solutions for a range of λ values. Since, iteratively fitting models over a grid of α and λ values is computationally impractical [5]. The algorithm instead fixes the mixing parameter α, and computes solutions for a path of λ values using warm starts. The procedure is as follows:

a) Start with large values of λ to set $\hat{\beta} = 0$ and decrease λ from there

b) Use the previous solution for the algorithm at the next λ value along the path. [Warm-starts]

c) Since α is fixed, the objective is a piece wise quadratic in λ

d) Find the smallest λ_l for each group that sets that group's coefficients to 0

- Thus begin the path search with: $\lambda^{max} = max_i \lambda_i$
- The exact value at which the first coefficient enters the model.
- Set λ^{min} to be a small fraction of λ^{max} [default 0.1]

This method is efficient for finding optimal λ values for solution, however, optimal α value is still problem specific.

2.2 Elastic Net Regression

Elastic net regularization adds an additional ridge regression-like penalty that improves performance when the number of predictors is larger than the sample size, allows the method to select strongly correlated variables together, and improves overall prediction accuracy [7]

When p > n (the number of covariates is greater than the sample size) lasso can select only n covariates (even when more are associated with the outcome) and it tends to select one covariate from any set of highly correlated covariates. Additionally, even when n > p, ridge regression tends to perform better given strongly correlated covariates.

The elastic net extends lasso (10) by adding an additional l^2 penalty term giving,

$$\underset{\beta \in \mathbb{R}^p}{min} ||y - X\beta||_2^2 + \lambda_1 ||\beta||_1 + \lambda_2 ||\beta||_2^2 \tag{12}$$

which is equivalent to solving,

$$\underset{\beta_0, \beta}{min} \sum_{n=1}^{N} ||y_i - \beta_0 - X\beta||_2^2 \tag{13}$$

$$\text{subject to } (1-\alpha)||\beta||_1 + \alpha||\beta||_2^2 \leq t$$

where $\alpha = \frac{\lambda_1}{\lambda_1 + \lambda_2}$

3 Numerical Analysis and Simulation

3.1 Sparse group lasso as a variable selector

3.1.1 Aims and Estimands

With reference to the simulations performed in the paper [5], the authors conducted simulations to examine covariate selection and predictive accuracy of sparse-group lasso penalization. One might also be interested in comparing the Elastic Net to the Sparse-group lasso for variable selection on simulated data. Our main aim is to find the correctly identified proportion of true non zero coefficients from the generative groups that matches the number of non zero coefficients from the results generated by Sparse Group Lasso and Elastic Net regularisation methods.

3.1.2 Data-generating mechanisms

The columns of X represent iid Gaussian distributed random variables where the columns are created and merged in batches of the group size to keep each group normally distributed and the response y was constructed as,

$$\sum_{l=1}^{g} X^{(l)} \beta^{(l)} + \sigma\epsilon \tag{14}$$

where $\epsilon \sim \mathcal{N}(0, I)$ and $\beta^{(l)} = (1, 2, ..., 5, 0, ..., 0)^T$ for l = 1,2,...,g and σ set so that signal-to-noise ratio was 2. Additionally, number of generative groups, g, varied from 1 to 3 changing the amount of the sparsity 5, 10, or 15 corresponding to g = 1, 2, 3. For example,

- $\beta = (1, 2, 3, 4, 5, 0, 0, \ldots, 0)^T$ for $g = 1$
- $\beta = (1, 2, 3, 4, 5, 0, 0...0, 1, 2, 3, 4, 5, 0, \ldots, 0)^T$ for $g = 2$
- $\beta = (1, 2, 3, 4, 5, 0, 0...0, 1, 2, 3, 4, 5, 0, \ldots, 0, 1, 2, 3, 4, 5, \ldots)^T$ for $g = 3$

We chose penalty parameters for the sparse-group lasso as $\alpha = 0.95$ so that the number of nonzero coefficients chosen in the fits matched the true number of nonzero coefficients in the generative model in Equation (14).

For each of the three generative groups, the parameter settings were as follows:

Simulation 1: $n = 60, m = 10, p = 100$
Simulation 2: $n = 70, m = 50, p = 200$
Simulations 3: $n = 150, m = 100, p = 1200$
Simulations 4: $n = 200, m = 200, p = 2000$

Note: n is number of observations, p is the number of covariates and m is number of groups.

3.1.3 Methods

The number of true non-zero coefficients from the generative groups that corresponds to the number of non-zero coefficients in the results produced by the Sparse Group Lasso (section 2.1.2) and Elastic Net (section 2.2) regularization methods is what we are primarily looking for. This is implemented with the help of glmnet and SGL package.

To calculate the proportion of correctly identified coefficients we first find number of nonzero coefficients for all lambdas, then find minimum lambda where the number of nonzero coefficients chosen in the fit match the true number of nonzero coefficients in the generative model which finally leads to obtaining the correct proportion. However, instead of conducting 10 trial of repeated simulations in [5] , only a single trial of simulations was conducted for each parameter settings. We then simulated our covariate matrix X with different numbers of covariates, observations, and groups.

3.1.4 Performance measures

We would like to note that in some trials we were unable to make the sparse-group lasso select exactly the true number of nonzero coefficients (due to the grouping effects). In these cases, we allowed the sparse-group lasso to select extra variables (as few as it could manage); however, when calculating the proportion of correct nonzero coefficient identifications, we used the total number of variables selected in our denominator, for example, if the sparse-group lasso selected nine variables in the five true variable case, it would be unable to get a

proportion better than $5/9 = 0.56$. While not ideal, we find no reason to believe that this would bias our results in favor of the sparse-group lasso.

	Number of Groups in Generative Model		
Models	1 group	2 groups	3 groups
	$n = 60, m = 10, p = 100$		
SGL	1	0.73	0.60
Elastic Net	0.8	0.6	0.467
	$n = 70, m = 50, p = 200$		
SGL	0.8	0.73	0.73
Elastic Net	0.8	0.7	0.563
	$n = 150, m = 100, p = 1200$		
SGL	0.833	0.8	0.533
Elastic Net	0.8	0.7	0.64
	$n = 200, m = 200, p = 2000$		
SGL	0.8	0.73	0.60
Elastic Net	0.8	0.8	0.44

Table 1: Using different models varying in numbers of covariates, observations, and groups to find the proportions of correct nonzero coefficients

For both SGL and Elastic Net we can observe that, as the number of generative groups increases, proportions of correct nonzero coefficient identifications decreases. Additionally we can also observe that SGL performs better most of the times when identifying the correct number of proportions.

3.2 Frequentist Sparse Group Lasso for Cox Model

3.2.1 Aim and Estimands

In the frequentist implementations of lasso, sparse group lasso and elastic net, the regression parameter β is estimated for a pre-specified value of the penalty parameter $\lambda > 0$. Cross validation is performed to decide which penalty parameter to choose out of a range of λs by using the negative cross validated partial log-likelihood function as the loss function[15]. Here we use an approximation[16] where each of the cross validation folds contributes to the cross validated log-likelihood which is approximated by subtracting the log-partial likelihood evaluated on the non left out data from that evaluated on the full data. This is implemented in the glmnet and SGL package, which we will use to determine the frequentist lasso, sparse group lasso and elastic net estimates in our simulation studies respectively. We then compare these with the oracle model to get a baseline comparison.

3.2.2 Data-generating mechanisms

Survival data are simulated according to (4) to follow Cox-Exponential survival model. The scale parameter is chosen such that the survival probability at 12 time units (called

months) are 0.5. Censoring is generated assuming uniform data entries in the first 36 months with a follow up time of 72 months. The survival times are then simulated as follows,

$$C^* \sim U(0, 36) + 72 \tag{15}$$

$$T^* \sim -\frac{log(U)}{\lambda exp(\beta' x)} \tag{16}$$

With n=200 independent observations, we have generated test and train data sets having p=30 penalized("genomic") variables. For each observation all genomic variables are independently N(0,1) distributed keeping the correlation structure similar to Lee et al. (2011) where we applied an alternative simulation scheme with pairwise correlations between any genomic variables x_j and x_k $\text{cor}(x_j, x_k) = \rho^{|j-k|}, j \neq k$, and $\rho = 0.5$. After data generation, all genomic variables in the input data matrix X_g are scaled and centered to have zero mean and unit variance[22].

Additionally, their corresponding survival times are simulated according to the Cox-Exponential model (4), with the following regression coefficients

$$\beta_X = (0.75, -0.75, 0.5, -0.5, 0.25, -0.25, 0, .., 0)$$

. All variables X_i with regression coefficients $\beta_i = 0$ are considered to be unrelated to the survival times.

3.2.3 Methods

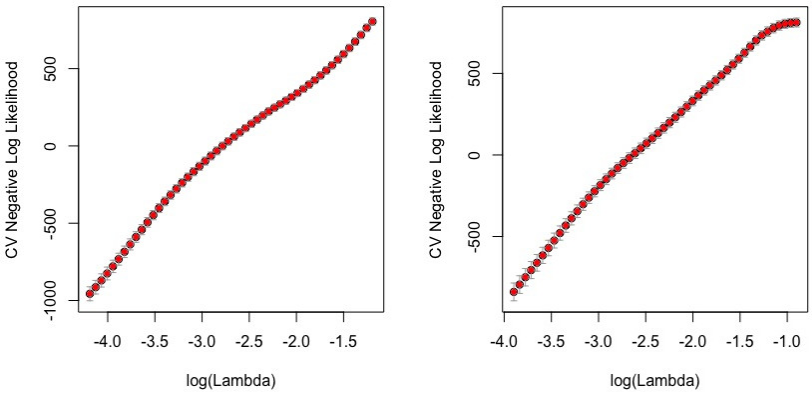

Figure 1: k fold cross validation for lasso(left) and elastic net(right)

We then fit the models with the training data accordingly. For sparse group lasso (section 2.1.2), we fit and cross-validate a regularized generalized cox model (section 1.2.1) via penalized maximum likelihood. The model is fit for a path of values of the penalty parameter, and a parameter value is chosen by cross-validation. We implement $\alpha = 0.95$ ($\sim$ almost lasso) and $\alpha = 0.05$ ($\sim$ almost group lasso) to observe the outcome of the grouping effect.

Similarly, for lasso (section 2.1) and elastic net (section 2.2), we do a k-fold cross-validation for glmnet, which produces a plot (figure 1), and returns a value for lambda. We then compare the sparse group lasso, lasso, elastic net method with the oracle cox regression model.

3.2.4 Performance measures

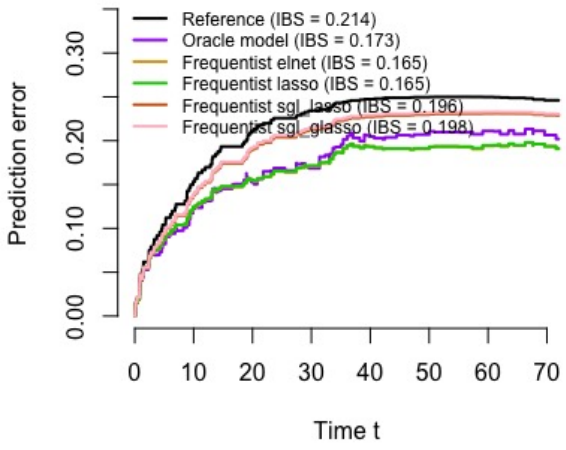

Figure 2: prediction error curves for different models

As observed, we can see that frequentist elastic net and lasso outperforms the rest, even the oracle surprisingly.

3.2.4.1 Brier score To measure the prediction accuracy a time dependent Brier score for survival data [12] which at a specific time point $t > 0$ for an individual k is defined as the mean squared error between the observed event status and the predicted survival probability. The empirical Brier score BS (t) as a function of time $t > 0$ is therefore defined by

$$\mathrm{BS}(t) = \frac{1}{n}\sum_{k=1}^{n}\left[\frac{\hat{S}\left(t \mid x_k\right)^2 I\left(t_k \leq t \wedge \delta_k = 1\right)}{\hat{G}\left(t_k\right)} + \frac{\left(1-\hat{S}\left(t \mid x_k\right)\right)^2 I\left(t_k > t\right)}{\hat{G}(t)}\right],$$

with individual survival time t_k, event status (also called censoring indicator) δ_k and estimated survival probability $\hat{S}\left(t \mid x_k\right)$ at time t (Graf et al., 1999). $\hat{G}(t)$ denotes the Kaplan-Meier estimate of the censoring distribution, which is based on the observations $(t_k; 1-\delta_k)$, with $I(\cdot)$ denoting the indicator function. As a summary measure for the time interval $[0, t^*]$ we use the integrated Brier score IBS (t^*) (Graf et al., 1999)

$$\mathrm{IBS}\left(t^*\right) = \frac{1}{t^*}\int_0^{t^*} \mathrm{BS}(t)dt$$

The lower the Brier scores are, the better the prediction performance of a model is deemed to be. A model can only be considered to have any prognostic value at all, if the Brier score function over time (also called prediction error curve) is lower than the prediction error curve obtained with the simple Kaplan-Meier estimator, which does not contain any covariate information at all.

Table 2: Integrated Brier Scores

Model	**IBS[0;time=72]**
Reference	0.214
Oracle	0.173
Frequentist lasso	0.165
Frequentist elastic net	0.165
Frequentist sgl($\alpha = 0.95$)	0.196
Frequentist sgl($\alpha = 0.05$)	0.180

3.2.4.2 Concordance index: In addition to studying the prediction performance of the prognostic models by means of the integrated Brier scores, we also compare the discriminative power of the models in terms of a time-dependent concordance index over time interval [0, 1] (c-index(t))[21], which is the frequency of concordant pairs among all pairs of subjects[22]. The larger the c-index, the better the discriminatory power of a model. The intuition behind Harrell's C-index is as follows. We have n patients with their covariate information $X_1, \ldots, X_p$ and a "time-to-event" response T. For patient i, our risk model assigns a risk score η_i. If our risk model is any good, patients who had shorter times-to-disease should have higher risk scores. Boiling this intuition down to two patients: the patient with the higher risk score should have a shorter time-to-disease. We can compute the C-index in the following way: For every pair of patients i and j (with $i \neq j$), look at their risk scores and times-to-event.

- If both T_i and T_j are not censored, then we can observe when both patients got the disease. We say that the pair (i, j) is a concordant pair if $\eta_i > \eta_j$ and $T_i < T_j$, and it is a discordant pair if $\eta_i > \eta_j$ and $T_i > T_j$.
- If both T_i and T_j are censored, then we don't know who got the disease first (if at all), so we don't consider this pair in the computation.
- If one of T_i and T_j is censored, we only observe one disease. Let's say we observe patient i getting disease at time T_i, and that T_j is censored. (The same logic holds for the reverse situation.)
  - If $T_j < T_i$, then we don't know for sure who got the disease first, so we don't consider this pair in the computation.
  - If $T_j > T_i$, then we know for sure that patient i got the disease first. Hence, (i, j) is a concordant pair if $\eta_i > \eta_j$, and is a discordant pair if $\eta_i < \eta_j$.

Harrell's C-index is simply

$$c = \frac{\#\text{ concordant pairs}}{\#\text{ concordant pairs } + \#\text{ discordant pairs}}.$$

The logic above can be expressed succinctly in a formula [20]:

$$c = \frac{\sum_{i \neq j} 1\{\eta_i < \eta_j\} 1\{T_i > T_j\} d_j}{\sum_{i \neq j} 1\{T_i > T_j\} d_j}$$

A practically meaningful interpretation of the C-index, however, may present several difficulties and pitfalls. Specifically, we identify two main issues:

- The C-index remains implicitly, and subtly, dependent on time
- Its relationship with the number of subjects whose risk was incorrectly predicted is not straightforward.

Failure to consider these two aspects may introduce undesirable and unwanted biases in the evaluation process, and even result in the selection of a sub-optimal model [23]

Table 3: Estimated C-index in % at time=107.4

Model	**AppCindex**	**Pairs**	**Concordant**
Oracle	67.4	39680	26764
Frequentist lasso	67.5	39680	26772
Frequentist elastic net	67.6	39680	26821
Frequentist sgl($\alpha = 0.95$)	61.1	39680	24248
Frequentist sgl($\alpha = 0.05$)	64.8	39680	25718

The larger the c-index, the better the discriminatory power of a model. In that case, elastic net outperforms them all, the frequestist lasso is close as well. Again, surprisingly both perform better than the oracle.

References

[1] APA. Collett, D. (2014). Modelling survival data in medical research (3rd ed.).

[2] Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the lasso". Journal of the Royal Statistical Society. Series B (methodological). Wiley. 58 (1): 267–88.

[3] Tibshirani, Robert (1997). "The lasso Method for Variable Selection in the Cox Model". Statistics in Medicine. 16 (4): 385–395. CiteSeerX 10.1.1.411.8024. doi:10.1002/(SICI)1097-0258(19970228)16:4¡385::AID-SIM380¿3.0.CO;2-3. PMID 9044528.

[4] Yuan, Ming; Lin, Yi (2006). "Model Selection and Estimation in Regression with Grouped Variables". Journal of the Royal Statistical Society. Series B (statistical

Methodology). Wiley. 68 (1): 49–67. doi:10.1111/j.1467-9868.2005.00532.x. JSTOR 3647556. S2CID 6162124.

[5] Simon, Noah and Friedman, Jerome and Hastie, Trevor and Tibshirani, Robert. (2013). A Sparse-Group Lasso. Journal of Computational and Graphical Statistics. 22.10.1080/10618600.2012.681250.

[6] Puig, Arnau Tibau, Ami Wiesel, and Alfred O. Hero III. "A Multidimensional Shrinkage-Thresholding Operator". Proceedings of the 15th workshop on Statistical Signal Processing, SSP'09, IEEE, pp. 113–116.

[7] Zou, Hui; Hastie, Trevor (2005). "Regularization and Variable Selection via the Elastic Net". Journal of the Royal Statistical Society. Series B (statistical Methodology). Wiley. 67 (2): 301–20. doi:10.1111/j.1467-9868.2005.00503.x. JSTOR 3647580. S2CID 122419596.

[8] Cox DR. Regression models and life tables (with discussion). Journal of the Royal Statistical Society, Series B 1972; 34:187–220.

[9] STATISTICS IN MEDICINE Statist. Med. 2005; 24:1713–1723, 2005 -Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/sim.2059

[10] Kalbfleisch JD, Prentice RL. The Statistical Analysis of Failure Time Data. Wiley: New York, 1980.

[11] https://vincentarelbundock.github.io/Rdatasets/csv/survival/cancer.csv

[12] Brier, G.W. (1950) Verification of Forecasts Expressed in Terms of Probability. Monthly Weather Review, 78, 1-3. http://dx.doi.org/10.1175/1520-0493(1950)078¡0001:VOFEIT¿2.0.CO;2

[13] Aleksandar Botev, Guy Lever, and David Barber. Nesterov's accelerated gradient and momentum as approximations to regularised update descent, 2016.

[14] Tuyen Trung Truong and Tuan Hang Nguyen. Backtracking gradient descent method for general c1 functions, with applications to deep learning, 2018.

[15] Verweij 1993, https://doi.org/10.1002/sim.4780122407 and Van Houwelingen 1994, https://doi.org/10.1002/sim.4780132307

[16] Van Houwelingen 2006, https://doi.org/10.1002/sim.2353

[17] Bruce, Peter, and Andrew Bruce. 2017. Practical Statistics for Data Scientists. O'Reilly Media

[18] David Cox: Regression models and life tables. Journal of the Royal Statistical Society B, 34 (1972), p. 187–220. JSTOR:2985181.

[19] A. Ziegler, S. Lange R. Bender: Survival time analysis. The Cox Regression. German Medical Weekly, 132(S 01) (2007), e42–e44. doi:10.1055/s-2007-959039

[20] On the use of Harrell's C for clinical risk prediction via random survival forests arXiv:1507.03092

[21] Harrell Jr, F. E. et al. (1982). Evaluating the yield of medical tests.

[22] Zucknick M, Saadati M, Benner A. Nonidentical twins: Comparison of frequentist and Bayesian lasso for Cox models. Biom J. 2015 Nov;57(6):959-81. doi: 10.1002/bimj.201400160. Epub 2015 Sep 29. PMID: 26417963.

[23] title = A practical perspective on the concordance index for the evaluation and selection of prognostic time-to-event models, doi = https://doi.org/10.1016/j.jbi.2020.103496, url = https://www.sciencedirect.com/science/article/pii/S1532046420301246, author = Enrico Longato and Martina Vettoretti and Barbara Di Camillo,