South African Statistical Journal
Vol. 55, No. 1, 29–44
https://doi.org/10.37920/sasj.2021.55.1.3

REGULARISATION IN DISCRETE SURVIVAL MODELS: A COMPARISON OF LASSO AND GRADIENT BOOSTING

Alphonce Bere[1]
Department of Statistics, University of Venda, Thohoyandou, South Africa
e-mail: *alphonce.bere@univen.ac.za*

Godfrey H. Sithuba
Department of Statistics, University of Venda, Thohoyandou, South Africa

Coster Mabvuu
Department of Statistics, University of Venda, Thohoyandou, South Africa

Retang Mashabela
Department of Statistics, University of Venda, Thohoyandou, South Africa

Caston Sigauke
Department of Statistics, University of Venda, Thohoyandou, South Africa

Kwabena Kyei
Department of Statistics, University of Venda, Thohoyandou, South Africa

We present the results of a simulation study performed to compare the accuracy of a lasso-type penalization method and gradient boosting in estimating the baseline hazard function and covariate parameters in discrete survival models. The mean square error results reveal that the lasso-type algorithm performs better in recovering the baseline hazard and covariate parameters. In particular, gradient boosting underestimates the sizes of the parameters and also has a high false positive rate. Similar results are obtained in an application to real-life data.

Key words: Boosting, Discrete survival model, First alcohol intake, Penalized likelihood, Variable selection.

1. Introduction

The discrete survival model expresses the hazard at time t as a function of the covariate vector $\boldsymbol{X}$ through an equation of the form

$$\lambda(t|\boldsymbol{X}_i) = F(\gamma_{0t} + \boldsymbol{X}_i^\top \boldsymbol{\beta}),$$

where F is an inverse link function and γ_{0t} is a function which captures the effect of time on the hazard; commonly referred to as the baseline hazard function.

[1] Corresponding author.

MSC2020 subject classifications. 62N86, 65C05, 62NO2.

In most applications of the discrete survival model, the number of potential explanatory variables is large, making it necessary to select a few relevant ones thereby obtaining parsimonious models which are easier to interpret. Variable selection becomes even more critical in those situations where the number of predictors exceeds the number of observations. In such cases, maximum likelihood estimates for the full model do not exist.

Traditionally, variable selection has been performed through stepwise methods. These consist of backward elimination, forward selection and stepwise selection which is a combination of forward selection and backward elimination. The results of stepwise selection procedures can be affected by very small perturbations in the data, leading to poor performance (Tutz and Schmid, 2016).

An alternative to traditional stepwise variable selection procedures are regularization methods. Among the prominent regularization techniques are penalty methods, in which variables are selected by optimizing a penalized likelihood, and boosting. The most prominent penalized likelihood variable selection method is the least absolute shrinkage and selection operator (lasso) (Tibshirani, 1996). In this method a weighted penalty that contains the sum over the absolute values of all the parameters is added to the log-likelihood function. Depending on the weight of the penalty, all or some of the coefficients of the variables are shrunk to zero. This property implies variable selection.

Boosting produces a prediction model by iteratively applying simple models (learners) and combining their solutions to produce a better prediction result. The most basic learner is the simple linear regression model. At each iteration the best performing learner is added to the model. This implies variable selection. The main weapon for variable selection in the boosting algorithm is the stopping iteration because early stopping prevents overfitting. The idea of boosting originated from the machine learning community (Freund and Schapire, 1996; Mayr et al., 2014) but was later propagated to the field of statistical modelling through the work of Friedman and others (Friedman et al., 2000; Friedman, 2001). The success of statistical boosting is attributed to an ability to incorporate automated variable selection in the fitting process, stability in high-dimensional settings with candidate predictors possibly exceeding available observations, and flexibility in terms of predictor effects that can be included in the final model (Mayr et al., 2014).

A specific feature of the discrete survival model is the baseline hazard estimates which are usually very unstable especially when the event of interest is observed at many time points. Variable selection algorithms for discrete survival models should therefore possess functionality for stabilizing baseline hazard parameters. The most convenient approach is to use ridge-type penalties because they do not result in parameters being shrunk to zero. If the baseline hazard is expanded as a sum of basis functions (for example using B-splines), a penalty based on differences between adjacent baseline hazard coefficients can be employed.

Groll and Tutz (2017) proposed a penalized likelihood method that performs efficient variable selection in discrete survival models with and without unobserved heterogeneity. The method uses a lasso-type penalty for variable selection and a ridge-type penalty for stabilization of baseline hazard parameters. An implementation of their method is available through the *glmmLasso* function of the R-package *glmmLasso* (Groll, 2011). In a simulation study the *glmmLasso* algorithm was compared with forward selection algorithms as implemented through the *glmer* function of the *lme4* package (Bates et al., 2015), the function *gamm4* in the R-package *gamm4* (Wood and Scheipl, 2013), and the *gam* function in the *mgcv* package (Wood, 2006). Comparison was also made with other lasso-based approaches as implemented in the packages *glmnet* (Friedman et al., 2010) and *penalized* (Goeman, 2010). The proposed method showed superior performance to all the alternatives. Gradient boosting was not considered in this study. In the current work, we compare the performance of *glmmLasso* and gradient boosting when used for variable selection in discrete survival models through a simulation study. Our focus will be restricted to models without unobserved heterogeneity.

2. Methods

2.1 The discrete survival model

Here we assume that the time takes the discrete values $\{1, 2, \ldots, q\}$. The discreteness may be due to time being intrinsically measured on a discrete scale or be a result of grouping continuous time into intervals. Predictions of the time to event T are assumed to be given in terms of a risk score $\kappa(\boldsymbol{X})$ where $\boldsymbol{X} = (X_1, X_2, \ldots, X_p)$ is a vector of predictor variables.

The censoring time is denoted by $C \in \{1, 2, \ldots, q\}$ and is assumed to be independent of T conditional on κ, i.e. censoring at random. For data which is right-censored, the duration of observation is $\overline{T} = \min(T, C)$. The random variable $\delta := I(T \leq C)$ indicates whether $\overline{T}$ is right-censored ($\delta = 0$) or not ($\delta = 1$).

We focus on modelling the discrete hazard function

$$\lambda(t|\kappa) = P(T = t \mid T \geq t, \kappa), \tag{1}$$

which gives the conditional probability of an event at time t given that the subject has survived up to time t. The corresponding survival function

$$S(t|\kappa) = P(T > t \mid T \geq t, \kappa) = \prod_{i=1}^{n} (1 - \lambda(t|\kappa(\boldsymbol{X}_i)))$$

gives the probability that the event occurs later than time t.

We are particularly interested in expressing the hazard as a function of covariates through an equation of the form

$$\lambda(t|\boldsymbol{X}_i) = F(\gamma_{0t} + \boldsymbol{X}_i^T \boldsymbol{\beta}),$$

where F is an inverse link function and γ_{0t} is a function which captures the effect of time on the hazard; commonly referred to as the baseline hazard function.

The risk score, $\boldsymbol{\kappa}_{it} = \boldsymbol{\gamma}_{\mathbf{0t}} + \boldsymbol{X}_i^\top \boldsymbol{\beta}$ is made up of intercepts $\gamma_{0t}, t = 1, 2, \ldots, q-1$, and a vector of regression coefficients $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)^\top$ which is independent of t. The hazard function in (1) is adequately determined by $F(\cdot)$ and the coefficients $\gamma_{01}, \gamma_{02}, \ldots, \gamma_{0,q-1}, \boldsymbol{\beta}^\top$, hence there is no need for an intercept at $t = q$. In this study, the cumulative distribution function of the logistic distribution was used for $F(\cdot)$. This gives a logistic discrete hazard model of the form

$$\lambda(t \mid \boldsymbol{X}_i) = \frac{\exp(\gamma_{0t} + \boldsymbol{X}_i^\top \boldsymbol{\beta})}{1 + \exp(\gamma_{0t} + \boldsymbol{X}_i^\top \boldsymbol{\beta})}.$$

Alternatives to the CDF of the logistic distribution include the CDFs of the Gompertz and standard normal distributions (Tutz and Schmid, 2016).

2.2 Penalized likelihood variable selection

Under random censoring, the probability of observing (t_i, d_i) is given by

$$L_i = P(T_i = t_i)^{\delta_i} P(T_i > t_i)^{1-\delta_i} P(C_i \geq t_i)^{\delta_i} P(C_i = t_i)^{1-\delta_i}.$$

Assuming non-informative censoring, i.e. the C_i are independent of the parameters that determine the risk of an event at any given time, we can separate the factor $c_i = P(C_i \geq t_i)^{\delta_i} P(C_i = t_i)^{1-\delta_i}$ to get the simpler equation

$$L_i = c_i P(T_i = t_i)^{\delta_i} P(T_i > t_i)^{1-\delta_i}.$$

If we define the sequence $(y_{i1}, \dots, y_{it}) = (0, \dots, 0, 1)$ for a non-censored observation ($\delta = 1$) and $(y_{i1}, \dots, y_{it}) = (0, \dots, 0, 0)$ for a censored observation, the likelihood (omitting c_i) can be expressed as

$$L_i = \prod_{s=1}^{t_i} \lambda(s|\boldsymbol{X}_i)^{y_{is}} (1 - \lambda(s|\boldsymbol{X}_i)^{1-y_{is}}.$$

Consequently, the log-likelihood for the whole sample is proportional to

$$l = \sum_{i=1}^{n} \sum_{s=1}^{t_i} y_{is} \log\lambda(s|\boldsymbol{X}_i) + (1 - y_{is}) \log(1 - \lambda(s|\boldsymbol{X}_i)). \tag{2}$$

which is the log-likelihood of binary observations $(y_{11}, \dots, y_{1,t_1}, y_{21}, \dots, y_{n,t_n})$ from the model $P(y_{ij} = 1/\boldsymbol{X}_i) = F(\boldsymbol{X}_i^\top \boldsymbol{\beta})$.

Variable selection can then be obtained by penalizing the log-likelihood. The most famous implementation of this is the lasso where a penalty of the form $\vartheta \sum_{j=1}^{p} |\beta_j|$ is added to equation (2) (Tibshirani, 1996). The penalty term enforces variable selection. The strength of the selection is determined by the magnitude of ϑ. When $\vartheta = 0$ all the variables are permitted into the model whereas as $\vartheta \longrightarrow \infty$, all the coefficients are shrunk to zero.

Estimation of parameters in the discrete survival model is complicated by the presence of the baseline hazard parameters in γ_{0t}. Estimates of these tend to be very unstable even for small sample sizes. This necessitates the addition of a second penalty term of the form $K(\gamma_{0t})$ to the log-likelihood. The penalized approximate log-likelihood used has the form

$$l = \sum_{i=1}^{n} \sum_{s=1}^{t_i} y_{is} \log\lambda(s \mid \boldsymbol{X}_i) + (1 - y_{is}) \log(1 - \lambda(s|\boldsymbol{X}_i)) - \vartheta \sum_{j=1}^{p} |\beta_j| - \vartheta_b K(\gamma_{0t}). \tag{3}$$

The additional tuning parameter helps to stabilize estimates of the baseline hazard parameters. A ridge-type penalty of the form

$$K(\gamma_{0t}) = \sum_{t=1}^{q} \gamma_{0t}^2$$

is used if the baseline is expressed as a sum of dummy variables representing time periods at which observations are made. Alternatively if the baseline hazard is expressed as a sum of basis functions (Efron, 1988; Fahrmeir, 1994; Möst et al., 2016), a difference penalty can be used. The latter approach is considered in this study.

To obtain a smooth baseline hazard use was made of a B-spline basis of order three and a second order difference penalty of the form

$$K_\alpha = \sum_{j=\alpha+1}^{q} (\Delta^\alpha \gamma_{0j})^2.$$

Here Δ is the difference operator on adjacent B-spline coefficients, i.e. $\Delta\gamma_{oj} = \gamma_{0j} - \gamma_{0,j-1}, \Delta^2\gamma_{0j} = \Delta(\gamma_{0j} - \gamma_{0,j-1}) = \gamma_{0j} - 2\gamma_{0,j-1} + \gamma_{0,j+2}$.

The tuning parameters ϑ_b and ϑ can be determined using information criteria such as AIC and BIC. Alternatively cross-validation can be used. Through simulations, Groll and Tutz (2017) however established that it is not necessary to select both parameters using either cross-validation or information criteria. They recommend an approach where ϑ is carefully selected using information criteria or cross-validation and a moderate value is used for ϑ_b. In line with Groll and Tutz (2017) wherein small values of ϑ_b are recommended, ϑ_b was set at 15. The results do not differ substantially when ϑ_b is set at 10 or 20. The BIC was used as criterion for selecting ϑ. The process involved creating a vector of possible values of ϑ, fitting a model using each value and noting the value that gives the minimum value of the BIC. To enhance fast convergence at each value of ϑ, the parameter estimates of the previous fit were used as starting values for the next fit. With this approach it is essential to make sure that the ϑ sequence starts at a value big enough such that all covariates are shrunk to zero.

Estimation of the model (3) can be done using appropriate binary logistic regression software after constructing an appropriate design matrix depending on the formulation of the baseline hazard. Details can be found in Tutz and Schmid (2016).

There are many R functions that can be used for penalized likelihood variable selection in discrete survival models. Prominent choices are *glmnet* (Friedman et al., 2010), *grplasso* (Meier et al., 2008), *grpreg* (Breheny and Jian, 2015), *penalized* (Goeman, 2010) and *glmmLasso* (Groll and Tutz, 2017). The first three of these functions do not have capability for separate penalization of the baseline hazard parameters. The first and the fourth have been shown to have weaker performance as compared to *glmmLasso* (Groll and Tutz, 2017). The *glmmLasso* function was therefore selected for this study. The *glmmLasso* algorithm is a gradient ascent algorithm which incorporates variable selection by L1-penalized estimation. In a final re-estimation step a model that includes only the variables corresponding to the non-zero fixed effects is fitted by simple Fisher scoring. In its implementation the *glmmLasso* function permits for the inclusion of a second penalty to stabilize the baseline hazard estimates. The *glmmLasso* function was primarily designed for variable selection in discrete survival models including unobserved heterogeneity but was also shown to exhibit superior performance in models without unobserved heterogeneity (Groll and Tutz, 2017).

2.3 Boosting

Gradient boosting is an example of an ensemble learning method wherein many models (weak learners) are combined together to create a more powerful model (strong learner) which gives better predictions. The models are fitted in series with the aim of reducing errors sequentially, i.e. each sequential model tries to reduce the errors (residuals) that are observed after the fitting of the previous model. Here again we assume we have a data set containing the values of an outcome variable y, a vector of values of predictor variables $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)$, and a loss function $\varrho(y, h(\boldsymbol{x}))$ that gives the magnitude of the deviation between the outcome variable and a prediction function $h(x)$. In gradient boosting the aim is to estimate the optimal prediction function h^* which minimizes the expected loss

$$h^* = \text{argmin}_h E_{Y,\boldsymbol{X}}[\varrho(y, h(\boldsymbol{x}))].$$

In practice, the theoretical mean $E_{Y,\boldsymbol{X}}[\varrho(y, h(\boldsymbol{x}))]$ is unknown. Instead one only has the data set $(y_i, \boldsymbol{x}_i, i = 1, \ldots, n)$. For this reason, the optimal prediction function is obtained by minimising the empirical risk

$$\mathfrak{R} = \frac{1}{n}\sum_{i=1}^{n} \varrho(y_i, h(\boldsymbol{x}_i)).$$

If the response variable is not normally distributed, ϱ is usually defined as the negative probability density function of the response distribution. In that case, if the distribution of the response belongs to the exponential family of distributions, the empirical risk is equivalent to the negative log-likelihood function of the generalized linear model:

$$\mathfrak{R} = -\frac{1}{n}\sum_{i=1}^{n} \phi(y_i, h(\boldsymbol{x}_i)),$$

where ϕ represents the probability density function of y. A convenient choice for $\mathfrak{R}$ for discrete survival models is the negative of (2).

We begin with an n-dimensional vector of starting values (representing the starting values for the predicted values of the n observations in the data) and update this vector sequentially as set out in the steps below.

1. Initialize the n-dimensional vector $\hat{\boldsymbol{h}}^0 = (\hat{h}_1^{[0]}, \ldots, \hat{h}_n^{[0]})^\top$ with starting values.

2. Specify a set of base learners $g_1(x_1), \ldots, g_p(x_p)$. Set the iteration counter m to 0.

3. Increase m to $m + 1$.

4. Obtain values of the negative gradient $-\frac{\partial \mathfrak{R}}{\partial h}$ at $\left(y_i, \hat{h}_i^{[m-1]}\right)$, $i = 1, \ldots, n$ to obtain the negative gradient vector

$$\begin{aligned}\boldsymbol{W}^{[m]} &= \left(W_i^{[m]}\right)_{i=1,\ldots,n} \\ &= \left(-\frac{\partial \mathfrak{R}}{\partial h}\right)_{i=1,\ldots,n}.\end{aligned}$$

5. Fit the negative gradient vector $\boldsymbol{W}^{[m]}$ seperately to every base learner:

$$\boldsymbol{W}^{[m]} \xrightarrow{\text{base learner}} \hat{g}_j^{[m]}(x_l), l = 1, \ldots, p.$$

6. Select the component l^* that best fits the negative gradient vector

$$l^* = \underset{1\le j\le p}{\text{argmin}} \sum_{i=1}^{n} \left(w_i^{[m]} - \hat{g}_j^{[m]}\right)^2.$$

7. Update the predictor $\hat{h}$ with this component

$$\hat{h}^{[m]}(\cdot) = \hat{h}^{[m-1]}(\cdot) + \omega.\hat{g}_{l^*}^{[m]}(x_{l^*}),$$

where ω is a small step length.

8. Iterate steps (2) to (7) until $m = m_{stop}$.

The most basic base learner is the simple linear regression model, but any classical linear least squares regression model can be used as a base learner. Smoothing splines can also be used in cases where the predictor variables have a non-linear effect on the response (Friedman, 2001). In every boosting iteration the best performing component of $\boldsymbol{X}$ is included in the final model since only the best-performing base-learner is added to the model. This leads to data-driven variable selection during the model estimation.

The main tuning parameter of boosting algorithms is the stopping iteration m_{stop}. Early stopping prevents overfitting and improves prediction accuracy. The shrinkage of estimates and variable selection is controlled by m_{stop}. The selection of m_{stop} therefore provides the trade-off between bias and variance. Large values lead to complex models with high variance and low bias, whereas small values lead to more shrinkage and reduced variance (Mayr et al., 2012). In the current work, the value of m_{stop} was selected using 25-fold cross-validation.

In R software the boosting algorithm can be implemented via the *gamboost* function from the *mboost* package (Hothorn et al., 2018). The package also has functionality for stabilizing baseline hazard parameter estimates.

3. Simulation study

3.1 The data generating mechanism

We follow the data generating scheme in (Groll and Tutz, 2017). The underlying process is given by $\boldsymbol{\kappa}_{it} = \boldsymbol{\gamma}_{\mathbf{0t}} + \boldsymbol{X}_i^\top \boldsymbol{\beta}$ as described before. The baseline hazard is given by

$$\boldsymbol{\gamma}_{\mathbf{0t}} = 2h_\Gamma(t-2) - 2.3,$$

where $h_\Gamma(t)$ is the density of the gamma distribution. The shape and scale parameters were chosen to be 5 and 1, respectively. The covariate effects were given by $\beta_1 = 5, \beta_2 = 6, \beta_3 = -3.5, \beta_4 = -3.5, \beta_5 = -4$ and $\beta_j = 0$ for $j = 6, \dots, p$. We set the number of variables, p at $p = 20, 50, 100$. The sample size n was set at 100. We considered a case where the covariates were categorical (binary) and a case where they were continuous. For the categorical case, binary data was generated with the proportions of zeros (reference level) and ones being roughly equal. For the continuous case, samples for each random variable were drawn from the uniform distribution within the $[0, 1]$ interval. The censoring probability π was set at 0.05. For each subject $i = 1, \dots, 100$, the following simulation scheme was used. For each time point $t = 1, \dots, q$,

1. Generate the Bernoulli response variable y_{it} with success probability $\lambda(t \mid \boldsymbol{x}_i)$;

2. **(a)** If $y_{it} = 1$, stop and set the event time as $T_i = t$;

(b) else generate a censoring variable S from the Bernoulli distribution with success probability π. If $S = 1$, stop and set $T_i = t$, else set $T_i = t + 1$.

Table 1. Results for MSE_{β} and MSE_{γ} for glmmLasso and gradient boosting (standard errors in brackets) with categorical covariates.

	MSE_{β}		MSE_{γ}	
p	**glmmLasso**	**boosting**	**glmmLasso**	**boosting**
20	10.49(22.97)	44.02(4.89)	14.54(21.38)	46.71(6.26)
50	14.47(22.56)	51.24(6.18)	22.91(35.06)	51.86(3.98)
100	13.58(9.73)	56.33(6.26)	37.04(35.13)	53.32(3.59)

Table 2. False negatives (F.N.) and false positives (F.P.) for glmmLasso and gradient boosting using categorical covariates.

	F.N.		**F.P.**	
p	**glmmLasso**	**boosting**	**glmmLasso**	**boosting**
20	0.00	0.00	0.33	10.38
50	0.04	0.00	0.66	20.08
100	0.10	0.00	2.8	26.71

3.2 Performance evaluation measures

We use the squared errors for the parameter vector $\boldsymbol{\beta}$ and the baseline hazard $\boldsymbol{\gamma}_0 = \left(\gamma_{0t}, \ldots, \gamma_{0q}\right)$ to compare the performance of the two methods. The corresponding squared errors, given by

$$\mathrm{MSE}_{\boldsymbol{\beta}} = \sum_{i=1}^{p} (\beta_i - \hat{\beta}_i)^2$$

and

$$\mathrm{MSE}_{\gamma} = \sum_{t=1}^{q} (\gamma_{0j} - \hat{\gamma}_{0j})^2,$$

are averaged over 100 simulations. False positive rates (FPR) and False Negative Rates (FNR) are also considered for each run. A false positive is a case where a parameter that is truly zero is estimated as non-zero. Conversely, a false negative refers to a non-zero parameter being estimated as zero. The false positive and false negative rates for each simulation run were also averaged over 100 simulation runs. The results are given in Tables 1 to 4.

It is clear from the tables that *glmmLasso* recovers both the covariate parameters and the baseline hazard function better than gradient boosting. It is also evident that gradient boosting gives more false positives as compared to *glmmLasso*. The results are similar between models with continuous covariates and those with discrete ones.

Table 3. Results for MSE_β and MSE_γ for glmmLasso and gradient boosting (standard errors in brackets) with continuous covariates.

	MSE_β		MSE_γ	
p	**glmmLasso**	**boosting**	**glmmLasso**	**boosting**
20	6.38(7.06)	43.02(29.86)	18.04(27.98)	379.53(107.70)
50	13.45(21.27)	33.75(28.04)	30.99(57.45)	416.33(63.60)
100	13.45(21.27)	37.93(37.75)	30.99(57.45)	430.96(49.57)

Table 4. False negatives (F.N.) and false positives (F.P.) for glmmLasso and gradient boosting using continuous covariates.

	F.N.		**F.P.**	
p	**glmmLasso**	**boosting**	**glmmLasso**	**boosting**
20	0.03	0	0.32	9.53
50	0.11	0	0.84	18.83
100	0.11	0.01	0.84	27.28

Figure 1 gives box plots which reveal the distribution of the parameter estimates obtained over 100 simulations from the glmmLasso and gradient boosting. From the figure, we can see that the gradient boosting parameter estimates have smaller magnitude compared to the glmmLasso estimates. In the next section we apply the two methods to a real life data set which was collected with the aim of establishing the factors that determine the timing of first alcohol intake among students at tertiary institutions.

4. Application

4.1 The data

The data were collected from students from two tertiary institutions in Thohoyandou, South Africa in 2017. One is a university and the other is a vocational training college (VCT). A total of 745 students completed a self-administered questionnaire. The questionnaire sought information on the demographic and socio-economic characteristics of respondents as reflected in Table 5. The questions corresponding to items 7 to 17 in the table sought to determine if the respondent had ever experienced the stated event between the ages from 1 to 15 years. The response variable was the age at first alcohol intake. Respondents who had never consumed alcohol were censored at their age as of the time of the survey.

Table 5 shows that the sample was evenly balanced in terms of gender and childhood place of residence, but dominated by students from the university. The majority of the respondents had never been married. Judging from the low family incomes, large numbers of siblings, and a fairly large proportion of parents with no education, we can conclude that the socio-economic status of the

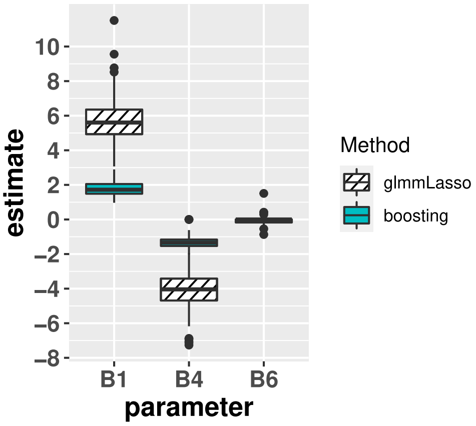

Figure 1. Box plots of estimates of selected parameters for the model with binary covariates, $n = 100$, $p = 50$.

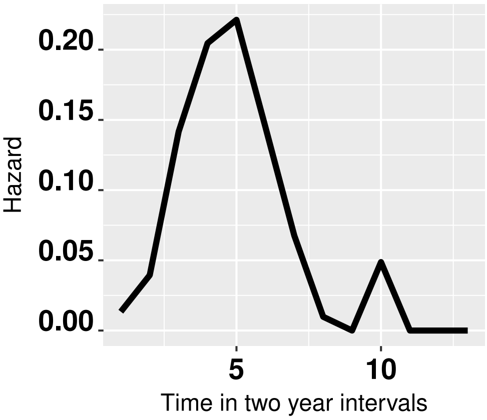

Figure 2. Empirical baseline hazard function.

majority of respondents cannot be classified as high. The most common experiences between the ages of one and 15 years were having drinking peers (43%), having a confiding relationship with one adult (43%), stress (89%) having parents who abused alcohol/other drugs (47%) and being subjected to parental discipline (59%). The use of other drugs/cigarettes was not very common (about 15%).

Figure 2 gives the raw life table baseline hazard estimate. We can see that the hazard of alcohol consumption initiation peaks around the age of 18 years. There is a suggestion of a secondary peak around the age of 28 years, probably due to those postgraduate respondents who started consuming alcohol after completing the first degree and obtaining the first job around the age of 25 years.

4.2 Variable selection results

From Table 6, we can see that the six variables selected by the *glmmLasso* functions were also selected by *gamboost*, albeit with much attenuated parameter estimates. Besides the six, *gamboost* also selects other variables. Most of the corresponding parameter estimates are however very small, indicating that these variables are not very influential in the model. The results are in agreement with those obtained in the simulation. Figure 3 shows that the optimal value of m for the boosting algorithm was $m_{stop} = 1289$.

4.3 Model Evaluation

Discrete survival models can be evaluated using goodness-of-fit measures and/or measures of predictive accuracy. The former methods provide information on how well the fitted values agree with the corresponding observed proportions while the latter are used to assess how well the model performs in predicting survival of future observations. The R-package *discSurv* (Welchowski and Schmid, 2019) was used to estimate the model evaluation measures considered in this work.

Adjusted deviance residuals (Tutz and Schmid, 2016) were used for evaluating goodness-of-fit. For a well-fitting model, the adjusted deviance residuals should be approximately normally distributed.

Table 5. Independent Variables considered for the analysis of time to first alcohol intake.

Item	Variable name	Category	Frequency(%)
1	Gender	Male	396(53.15)
		Female	349(46.85)
2	Ethnicity	Black	729(97.90)
		Other race	16(2.09)
3	Family monthly Income (rand)	< 1000	134(17.99)
		$1000 - 4000$	405(54.36)
		> 4000	206(27.65)
4	Siblings	$0 - 3$	348(46.71)
		$4 - 6$	354(47.52)
		≥ 7	43(5.77)
5	Parent’s qualification	None	314(42.15)
		Matric	154(20.67)
		Above matric	277(37.18)
6	Childhood residence	Rural/farmland	404(54.23)
		Urban	341(45.77)
7	Other drugs	Yes	108(14.50)
		No	637(85.50)
8	Drinking Peers	Yes	318(42.68)
		No	427(57.32)
9	Physical abuse	Yes	60 (8.05)
		No	685(91.95)
10	Sexual Abuse	Yes	12(1.61)
		No	733(98.39)
11	Negative life events	Yes	217(29.13)
		No	528(70.87)
12	Stress	Yes	663(88.99)
		No	82(11.00)
13	Parental drug/alcohol abuse	Yes	344(46.17)
		No	401(53.83)
14	Subjected to parental discipline?	Yes	441(59.19)
		No	304(40.81)
15	Family dependent on social welfare?	Yes	480(64.43)
		No	265(35.57)
16	Dysfunctional family	Yes	254(34.09)
		No	491(65.91)
17	Relation with adult	Yes	423(56.78)
		No	322(43.22)

Table 6. Parameter estimates obtained using glmmLasso and gamboost.

Item	Variable name	Category	glmmLasso	gamboost
1	Gender	Female	0.00	0.00
		Male	0.38	0.12
2	Ethnicity	Black	0.00	0.00
		Other Race	-	-
3	Family monthly Income (rand)	< 1000	0.00	0.00
		1000 – 4000	-	-0.14
		> 4000	-	0.00
4	Siblings	0 – 3	0.00	0.00
		4 – 6	-	-0.01
		≥ 7	-	0.07
5	Parent’s qualification	None	0.00	0.00
		Matric	-	0.01
		Above matric	-	-0.16
6	Childhood residence	Rural/farmland	0.00	0.00
		Urban	-	0.04
7	Other drugs	No	0.00	0.00
		Yes	0.67	0.34
8	Drinking Peers	No	0.00	0.00
			0.34	0.13
9	Physical abuse	No	0.00	0.00
		Yes	0.56	0.20
10	Sexual Abuse	No	0.00	0.00
		Yes	-	0.10
11	Negative life events	No	0.00	0.00
		Yes	-	0.08
12	Stress	No		0.00
		Yes		-0.15
13	Parental drug/alcohol abuse	No	0.00	0.00
		Yes	0.35	0.12
14	Subjected to parental discipline?	No	0.00	0.00
		Yes	-	-0.13
15	Family dependent on social welfare?	No	0.00	0.00
		No	-	-
16	Dysfunctional family	No	0.00	0.00
		No	-	-
17	Relation with adult	No	0.00	0.00
		Yes	-	-0.07

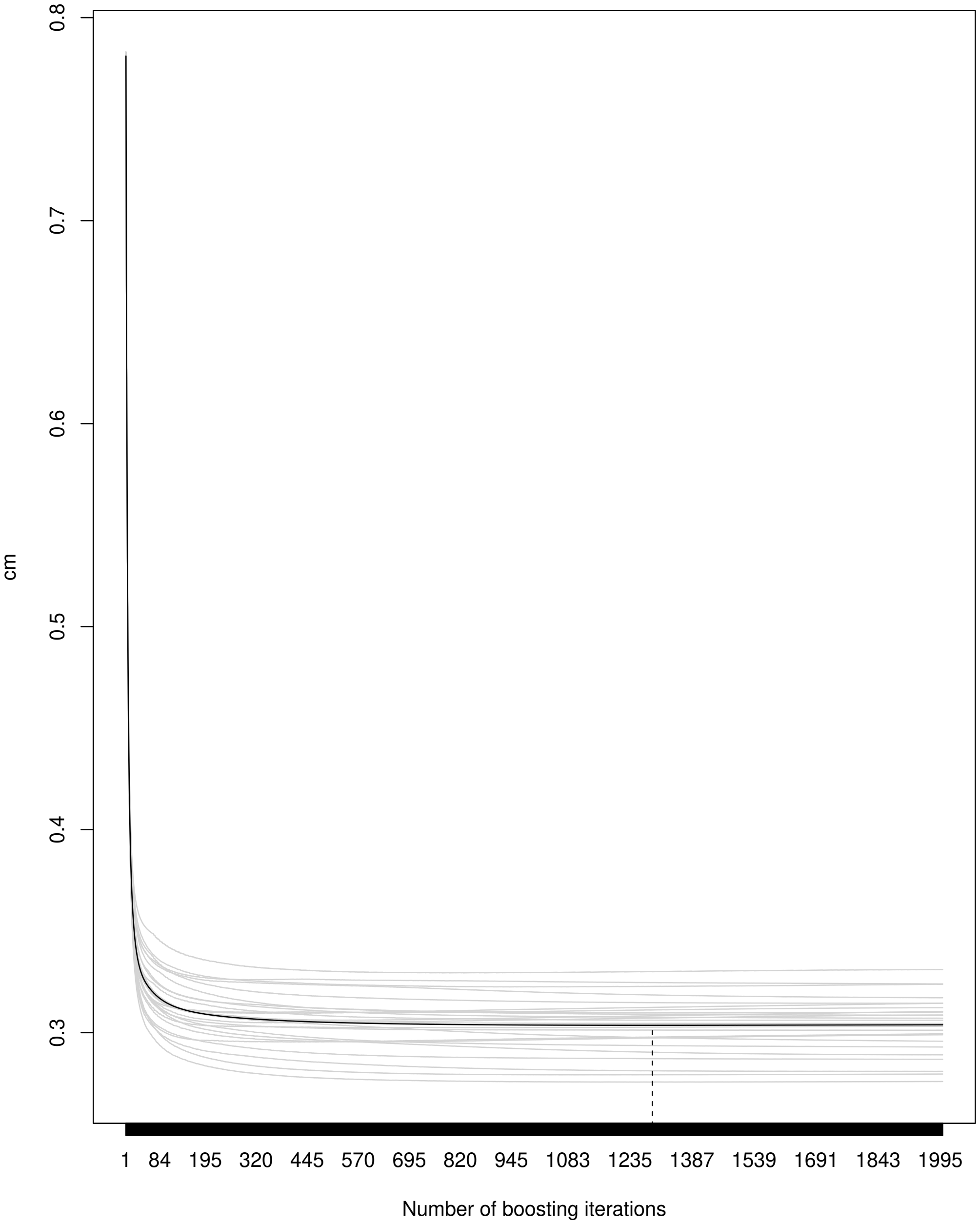

Figure 3. Cross-validated predictive risk with 25-fold bootstrapping.

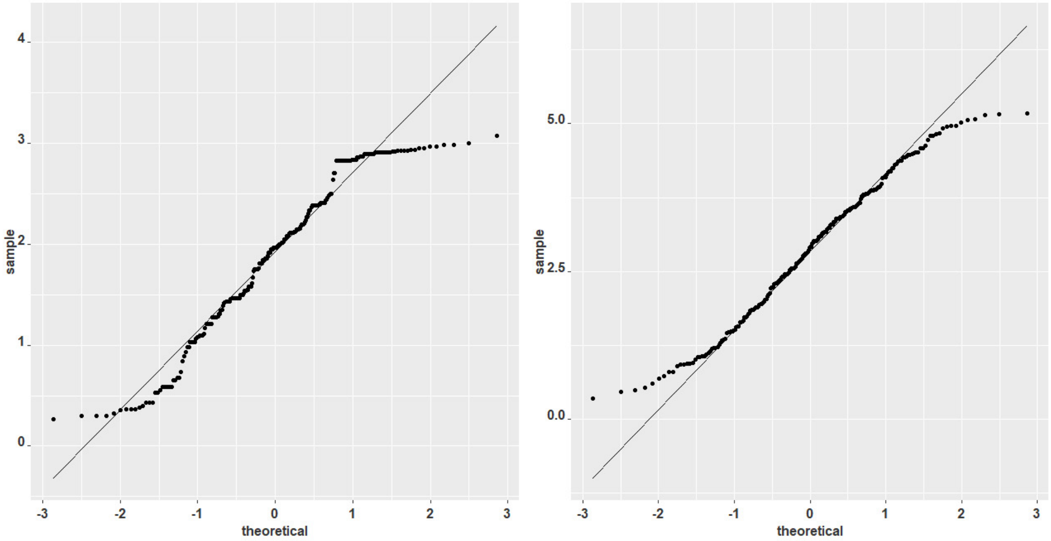

Figure 4. QQ-plots of the deviance residuals for glmmLasso (left panel) and gamboost (right panel).

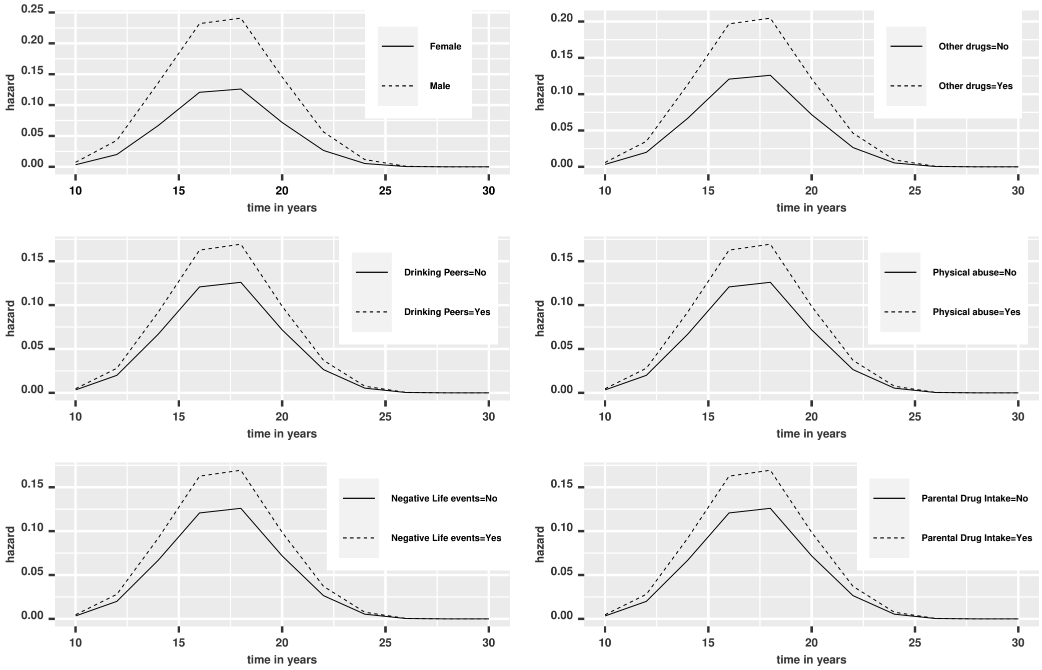

Figure 5. Hazards of alcohol intake initiation estimated using *glmmLasso*.

Figure 4 gives a quantile-quantile plot of the adjusted deviance residuals for the three models considered here. The plots are largely straight lines though there is suggestion of deviations from normality.

Predictive performance of the models was evaluated through the use of the C-index (Schmid et al., 2018). The C index should exceed 0.5 if κ predicts better than chance. The C indices for *glmmLasso* and *gamboost* are 0.75 and 0.63, respectively, showing fairly high discriminating power for both models.

Figure 5 shows the estimated hazards of alcohol initiation stratified by each of the six variables selected using the *glmmLasso* algorithms. The graphs in the figure were obtained using *glmmLasso* which exhibited the highest discriminating power. As expected from Table 6 the most influential variable is the use of other drugs.

5. Conclusion

A simulation study has been conducted to compare the accuracy of lasso as implemented in *glmmLasso* and gradient boosting in recovering the baseline hazard and covariate parameters in discrete survival models. Categorical (binary) and continuous covariates were considered. The results show that lasso gives smaller MSE's for both the baseline hazard and covariate estimates. Gradient boosting underestimates the parameter sizes and also gives a high false positive rate. Similar results are obtained when the two methods are applied to a real-life data set.

References

BATES, D., MÄCHLER, M., BOLKER, B., AND WALKER, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, **67**, 1–48.

BREHENY, P. AND JIAN, J. (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, **25**, 173–187.

EFRON, B. (1988). Logistic regression, survival analysis, and the Kaplan–Meier curve. *Journal of the American Statistical Association*, **83**, 414–425.

FAHRMEIR, L. (1994). Dynamic modelling and penalized likelihood estimation for discrete time survival data. *Biometrika*, **81**, 317–330.

FREUND, Y. AND SCHAPIRE, R. E. (1996). Experiments with a new boosting algorithm. *In Proceedings of the Thirteenth Conference on Machine Learning, Bari, Italy*. 148–156.

FRIEDMAN, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, **29**, 1189–1232.

FRIEDMAN, J. H., HASTIE, T., AND TIBSHIRANI, R. (2000). Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, **28**, 337–407.

FRIEDMAN, J. H., HASTIE, T., AND TIBSHIRANI, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, **33**, 1–22.

GOEMAN, J. J. (2010). L1 penalized estimation in the Cox proportional hazards model. *Biometrical Journal*, **52**, 70–84.

GROLL, A. (2011). *glmmLasso: Variable selection for generalized linear mixed models by L1-*

penalized estimation. R package version 1.2.3.
URL: *http://CRAN.R-project.org/package=glmmLasso*

Groll, A. and Tutz, G. (2017). Variable selection in discrete survival models including heterogeneity. *Lifetime Data Analysis*, **23**, 305–338.

Hothorn, T., Buehlmann, P., Kneib, T., Schmid, M., and Hofner, B. (2018). *mboost: Model-Based Boosting*. R package version 2.9-1.
URL: *https://CRAN.R-project.org/package=mboost*

Mayr, A., Binder, H., Gefeller, O., and Schmid, M. (2014). The evolution of boosting algorithms. From machine learning to statistical modelling. *Methods of Information in Medicine*, **53**, 419–427.

Mayr, A., Hofner, B., and Schmid, M. (2012). The importance of knowing when to stop. A sequential stopping rule for component-wise gradient boosting. *Methods of Information in Medicine*, **51**, 178–186.

Meier, L., Geer, S. V. D., and Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B*, **70**, 53–71.

Möst, S., Pössnecker, W., and Tutz, G. (2016). Variable selection for discrete competing risks models. *Quality & Quantity*, **50**, 1589–1610.

Schmid, M., Tutz, G., and Welchowski, T. (2018). Discrimination measures for discrete time-to-event predictions. *Econometrics and Statistics*, **7**, 153–164.

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

Tutz, G. and Schmid, M. (2016). *Modelling Discrete Time-to-Event Data*. Springer, Zurich.

Welchowski, T. and Schmid, M. (2019). *discSurv: Discrete time survival analysis*. R package version 1.4.1.
URL: *https://CRAN.R-project.org/package=discSurv*

Wood, S. N. (2006). *Generalized Additive Models: An Introduction with R*. 2nd edition. CRC Press, Boca Raton, FL.

Wood, S. N. and Scheipl, F. (2013). *gamm4: Generalized additive mixed models using mgcv and lme4*. R package version 0.2-2.
URL: *http://CRAN.R-project.org/package=gamm4*

Manuscript received 2020-02-27, revised 2020-08-03, accepted 2020-10-18.