

# Regularized AFT models

Anuj Khare

This has been adapted from the Survival package vignette.

## Model definition

**Inputs:**  $X_{n \times p}$ ,  $T_{n \times 2}$ , density  $f \dots$

**Parameters:** Coefficients  $\beta = [\beta_0, \beta_1, \dots, \beta_p]^T$ , Scale  $\sigma$   
where,  $\beta_0$  is the intercept.

The type of censoring is determined as follows:

$$\begin{cases} \text{Left censoring} & \text{if: } -\infty = \underline{t}_i, \bar{t}_i < \infty \\ \text{Right censoring} & \text{if: } -\infty < \underline{t}_i, \bar{t}_i = \infty \\ \text{Interval censoring} & \text{if: } -\infty < \underline{t}_i \neq \bar{t}_i < \infty \\ \text{No censoring} & \text{if: } -\infty < \underline{t}_i = \bar{t}_i < \infty \end{cases} \quad (1)$$

Define the transformed output,

$$y_i = \text{trans}(T_i) \quad (2)$$

where *trans* depends on the distribution, and is log for the log-gaussian, log-logistic distributions, and so on.

The model is defined by the equation:

$$y = X\beta + \sigma\epsilon \quad (3)$$

where,  $\epsilon \sim f$ . Thus,

$$e_i = \frac{y_i - x_i^T \beta}{\sigma} \sim f \quad (4)$$

We define the elastic net (L1 + L2) penalty as follows:

$$\lambda P_\alpha(\beta) = \lambda(\alpha \|\beta\|_1 + 1/2(1 - \alpha) \|\beta\|_2^2) \quad (5)$$

Our objective is to maximize the penalized, scaled log likelihood:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmax}} \left( \frac{1}{n} l(\beta) - \lambda P_\alpha(\beta) \right) \quad (6)$$

## Likelihood

For calculating likelihood, in the observations with no censoring, the pdf is used, and in censored observations, the cdf is used. Hence, the likelihood is given as:

$$lik = \left( \prod_{exact} f(e_i)/\sigma \right) \left( \prod_{right} 1 - F(e_i) \right) \left( \prod_{left} F(e_i) \right) \left( \prod_{interval} F(e_i^u) - F(e_i^l) \right) \quad (7)$$

"Exact", "left", "right", and "interval" refer to uncensored, left censored, right censored and interval censored observations respectively, and  $F$  is the cdf of the distribution.  $e_i^u$ , and  $e_i^l$  are upper and lower endpoints for interval censored data.

Hence the log likelihood is given as:

$$l(\beta) = \sum_{exact} g_1(e_i) - \log(\sigma) + \sum_{right} g_2(e_i) + \sum_{left} g_3(e_i) + \sum_{interval} g_4(e_i^l, e_i^u) \quad (8)$$

$$g_1 = \log(f), g_2 = \log(1 - F), g_3 = \log(F), g_4(e_i^l, e_i^u) = \log(F(e_i^u) - F(e_i^l)).$$

## Score and Hessian

Derivatives of the LL with respect to the regression parameters are:

$$\frac{\partial l(\beta)}{\partial \beta_j} = \sum_{i=1}^n \frac{\partial g}{\partial \eta_i} \frac{\partial \eta_i}{\partial \beta_j} = \sum_{i=1}^n x_{ij} \frac{\partial g}{\partial \eta_i} \quad (9)$$

$$\frac{\partial^2 l(\beta)}{\partial \beta_j \partial \beta_k} = \sum_{i=1}^n x_{ij} x_{ik} \frac{\partial^2 g}{\partial \eta_i^2} \quad (10)$$

where  $\eta_i = x_i^T \beta$  is the vector of linear predictors.

Define  $\mu_i = \frac{\partial g}{\partial \eta_i}$ , where  $g$  is one of  $g_1$  to  $g_4$  depending on type of censoring in the  $i^{th}$  observation, and  $\mu = [\mu_1, \dots, \mu_n]^T$ . Then, partial derivative of log-likelihood is given as:

$$\frac{\partial l(\beta)}{\partial \beta_j} = \sum_{i=1}^n x_{ij} \mu_i \quad (11)$$

Hence, the score (gradient of log likelihood) is given as:

$$S = \nabla_{\beta} l(\beta) = X^T \mu = \sum_{i=1}^n \mu_i \bar{x}_i \quad (12)$$

The hessian can be written as:

$$H = \sum_{i=1}^n \bar{x}_i \bar{x}_i^T \frac{\partial^2 g}{\partial \eta_i^2} = \sum_{i=1}^n \bar{x}_i \bar{x}_i^T w_i \quad (13)$$

Define  $W = \text{diag}(w_1, \dots, w_n)$ .

$$H = X^T W X \quad (14)$$

## Taylor approximation for log-likelihood

A 2-step Taylor series centered at  $\tilde{\beta}$  is given as:

We use Newton's algorithm to find MLE for the AFT model. The Newton update is as follows:

$$\begin{aligned} \beta &= \tilde{\beta} + H^{-1} \tilde{S} \\ &= \tilde{\beta} + (X^T \tilde{W} X)^{-1} X^T \tilde{\mu} \\ &= (X^T \tilde{W} X)^{-1} ((X^T \tilde{W} X) \tilde{\beta} + X^T \tilde{\mu}) \\ &= (X^T \tilde{W} X)^{-1} X^T (\tilde{W} X \tilde{\beta} + \tilde{\mu}) \\ &= (X^T \tilde{W} X)^{-1} X^T (\tilde{W} X \tilde{\beta} + \tilde{\mu}) \end{aligned} \quad (15)$$

where, define the working response  $\tilde{z} = X \tilde{\beta} + \tilde{W}^{-1} \tilde{\mu}$ . Here, the tilde denotes that the respective values are evaluated using the parameters from the previous step.

## Coordinate descent

Hence, at each step we are solving a penalized weighted least squares problem, which is a minimizer of (using the scaled approximate log-likelihood):

$$M = \frac{1}{2n} \sum_{i=1}^n \tilde{w}_i (\tilde{z}_i - \bar{x}_i^T \beta)^2 + \lambda P_{\alpha}(\beta) \quad (16)$$

The subderivative of the optimization objective is given as:

$$\frac{\partial M}{\partial \beta_k} = \frac{1}{n} \sum_{i=1}^n -\tilde{w}_i x_{ik} (\tilde{z}_i - \bar{x}_i^T \beta) + \lambda \alpha \text{sgn}(\beta_k) + \lambda(1 - \alpha) \beta_k \quad (17)$$

where,  $\text{sgn}(\beta_k)$  is 1 if  $\beta_k > 1$ , -1 if  $\beta_k < 0$  and 0 if  $\beta_k = 0$ .

Using the subderivative, three cases of solutions for  $\beta_k$  may be obtained.

The solution is given by:

$$\hat{\beta}_k = \frac{S\left(-\frac{1}{n} \sum_{i=1}^n \tilde{w}_i x_{ik} \left[\tilde{z}_i - \sum_{j \neq k} x_{ij} \beta_j\right], \lambda \alpha\right)}{-\frac{1}{n} \sum_{i=1}^p \tilde{w}_i x_{ik}^2 + \lambda(1 - \alpha)} \quad (18)$$

where,  $w_i$  and  $z_i$  are given in 13 and 16 respectively, and  $S$  is the soft thresholding operator given as:

$$S(x, \lambda) = \text{sgn}(x)(|x| - \lambda)_+ \quad (19)$$

The intercept is not regularized, and hence can be calculated as:

$$\hat{\beta}_0 = \frac{-\frac{1}{n} \sum_{i=1}^n \tilde{w}_i \left[\tilde{z}_i - \sum_{j \neq 0} x_{ij} \beta_j\right]}{-\frac{1}{n} \sum_{i=1}^p \tilde{w}_i} \quad (20)$$

The coordinate descent algorithm works by cycling through each  $\beta_j$  in turn, keeping the others constant, and using the above estimate to calculate the optimal value  $\hat{\beta}_j$ .

After each update cycle for  $\beta$ , the scale parameter  $\sigma$  is updated once using a Newton step:

$$\sigma_{new} = \sigma_{old} - \left(\frac{\partial^2 l(\sigma)}{\partial \sigma^2}\right)^{-1} \left(\frac{\partial l(\sigma)}{\partial \sigma}\right) \quad (21)$$

This is repeated until convergence of both  $\beta$  and  $\sigma$ . Note that we have ignored the off-diagonal entries in the Hessian for the scale parameter.

## Pathwise solution

This section is borrowed from section 2.3 of [3]. The iregnet function will return solutions for an entire path of values of  $\lambda$ , for a fixed  $\alpha$ . We begin with  $\lambda$  sufficiently large to set the solution  $\beta = 0$ , and decrease  $\lambda$  until we arrive near the unregularized solution. The solutions for each value of  $\lambda$  are used as the initial estimates of  $\beta$  for the next  $\lambda$  value. This is known as warm starting, and makes the algorithm efficient and stable. To choose initial value of  $\lambda$ , we use Equation 18, and notice that for  $\frac{1}{n} \sum_{i=1}^n w_i(0) x_{ij} z(0)_i < \alpha \lambda$  for all  $j$ , then  $\beta = 0$  minimizes the objective 6. Thus,

$$\lambda_{max} = \max_j \frac{1}{n\alpha} \sum_{i=1}^n w_i(0) x_{ij} z(0)_i \quad (22)$$

We will set  $\lambda_{min} = \epsilon \lambda_{max}$  , and compute solutions over a grid of  $m$  values, where  $\lambda_j = \lambda_{max}(\lambda_{min}/\lambda_{max})^{j/m}$  for  $j = 0, \dots, m$ .

## Algorithm

The algorithm to be followed for fitting the distribution is:

```

Transform output variable  $y$  using log transformation ;
Calculate  $\lambda_{max}$  using equation 22, and set  $\tilde{\beta} = 0, \tilde{\eta} = 0$  ;
Calculate  $\lambda_{min}$  and a grid of  $m$   $\lambda$  values ;
foreach  $\lambda_j$  in  $j = m, \dots, 0$  do
    repeat
        Compute  $\tilde{w}_i$  and  $\tilde{z}_i$  ;
        Find  $\hat{\beta}$  by solving the penalized weighted least square problem
            defined in equation 6 using coordinate descent ;
        Set  $\tilde{\beta} = \hat{\beta}$  ;
    until convergence of  $\hat{\beta}$ ;
    Set  $\tilde{\beta} = \hat{\beta}$  ,  $\tilde{\eta} = X\tilde{\beta}$  ;
end

```

**Algorithm 1:** Overall optimization algorithm

## Scale parameter

So far, I have ignored the  $\sigma$  parameter from the calculations and equations. This is only reasonable if we treat  $\sigma$  as fixed. However, in other cases,  $\sigma$  needs to be estimated along with the parameters  $\beta$ , by using the derivatives as listed below.

## Derivatives

Iterations are done with respect to  $\log(\sigma)$  to prevent numerical underflow.

$$\begin{aligned}
\frac{\partial g_1}{\partial \eta} &= -\frac{1}{\sigma} \left[ \frac{f'(z)}{f(z)} \right] \\
\frac{\partial g_4}{\partial \eta} &= -\frac{1}{\sigma} \left[ \frac{f(z^u) - f(z^l)}{F(z^u) - F(z^l)} \right] \\
\frac{\partial^2 g_1}{\partial \eta^2} &= -\frac{1}{\sigma^2} \left[ \frac{f''(z)}{f(z)} \right] - (\partial g_1 / \partial \eta) \\
\frac{\partial^2 g_4}{\partial \eta^2} &= -\frac{1}{\sigma^2} \left[ \frac{f'(z^u) - f'(z^l)}{F(z^u) - F(z^l)} \right] - (\partial g_4 / \partial \eta)^2 \\
\frac{\partial g_1}{\partial \log \sigma} &= -\left[ \frac{zf'(z)}{f(z)} \right] \\
\frac{\partial g_4}{\partial \log \sigma} &= -\left[ \frac{z^u f(z^u) - z^l f(z^l)}{F(z^u) - F(z^l)} \right] \\
\frac{\partial^2 g_1}{\partial (\log \sigma)^2} &= \left[ \frac{z^2 f''(z) + zf'(z)}{f(z)} \right] - (\partial g_1 / \partial \log \sigma)^2 \\
\frac{\partial^2 g_4}{\partial (\log \sigma)^2} &= \left[ \frac{(z^u)^2 f'(z^u) - (z^l)^2 f'(z^l)}{F(z^u) - F(z^l)} \right] - (\partial g_1 / \partial \log \sigma)(1 + \partial g_1 / \partial \log \sigma) \\
\frac{\partial^2 g_1}{\partial \eta \partial \log \sigma} &= \left[ \frac{zf''(z)}{\sigma f(z)} \right] - (\partial g_1 / \partial \eta)(1 + \partial g_1 / \partial \log \sigma) \\
\frac{\partial^2 g_4}{\partial \eta \partial \log \sigma} &= \left[ \frac{z^u f'(z^u) - z^l f'(z^l)}{\sigma [F(z^u) - F(z^l)]} \right] - (\partial g_4 / \partial \eta)(1 + \partial g_4 / \partial \log \sigma)
\end{aligned} \tag{23}$$

Derivatives for  $g_2$  can be obtained by setting  $z_u$  to  $\inf$  in the equations for  $g_4$ , and similarly for  $g_3$ .

The distribution specific values of  $f(z)$ , etc. are omitted.

## Subgradient of Cost

The cost to be minimized is the negative of the penalized, scaled log-likelihood:

$$J(\beta) = \left( -\frac{1}{n}l(\beta) + \lambda P_{\alpha}(\beta) \right) \quad (24)$$

$$\hat{\beta} = \operatorname{argmin}_{\beta} \left( -\frac{1}{n}l(\beta) + \lambda P_{\alpha}(\beta) \right) \quad (25)$$

The subderivative of the cost is given as:

$$\nabla_{\beta} J = -\frac{1}{n}S(\beta) + \lambda \alpha \operatorname{sgn}(\beta) + \lambda(1 - \alpha)\beta \quad (26)$$

where,  $\operatorname{sgn}(\beta)$  is calculated element-wise on the vector.  $S$  is the score as given in 12.

The closeness of the degree 1, 2, and inf norms of the subderivate to zero can be used as a mteric for judging the optimality of the obtained solutions.

# Bibliography

- [1] Survival - Terry M Therneau
- [2] Machine Learning: A Probabilistic Perspective - Kevin Murphy
- [3] Regularization Paths for Cox's Proportional Hazards Model via Coordinate Descent - Simon, Friedman, Hastie, Tibshirani
- [4] AFT - TD Hocking