

Regularized AFT models

Anuj Khare

This has been adapted from the Survival package vignette.

Model definition

A standard AFT model is defined as follows:

$$\log(T_i) = x_i^T \beta + \sigma \epsilon_i \quad (1)$$

Where x_i are the covariates, Y_i is the observed time (output). $\epsilon_i \sim f$, where f is the probability density.

Here on, we assume that σ is fixed, and is ignored. We let y_i be the transformed data vector obtained by taking log of T_i . Hence, we have:

$$e_i = \frac{y_i - x_i^T \beta}{\sigma} \sim f \quad (2)$$

For interval regression with censored data, we are given time intervals $\{\underline{t}_i, \bar{t}_i\}$ and covariates x_i for $i = 1 : n$, where \underline{t}_i may be *inf* (left censoring) and \bar{t}_i may be *inf* (right censoring).

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:

$$l = \left(\prod_{exact} f(e_i) / \sigma \right) \left(\prod_{right} 1 - F(e_i) \right) \left(\prod_{left} F(e_i) \right) \left(\prod_{interval} F(e_i^u) - F(e_i^l) \right) \quad (3)$$

where, "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:

$$LL = \sum_{exact} g_1(e_i) - \log(\sigma) + \sum_{right} g_2(e_i) + \sum_{left} g_3(e_i) + \sum_{interval} g_4(e_i^l, e_i^u) \quad (4)$$

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

Score and Hessian

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

$$\frac{\partial LL}{\partial \beta_j} = \sum_{i=1}^n \frac{\partial g}{\partial \eta_i} \frac{\partial \eta_i}{\partial \beta_j} = \sum_{i=1}^n x_{ij} \frac{\partial g}{\partial \eta_i} \quad (5)$$

$$\frac{\partial^2 LL}{\partial \beta_j \partial \beta_k} = \sum_{i=1}^n x_{ij} x_{ik} \frac{\partial^2 g}{\partial \eta_i^2} \quad (6)$$

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:

$$\frac{\partial l(\beta)}{\partial \beta_j} = \sum_{i=1}^n x_{ij} \mu_i \quad (7)$$

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

$$S = \nabla_{\beta} LL(\beta) = X^T \mu = \sum_{i=1}^n \mu_i \bar{x}_i \quad (8)$$

The hessian can be written as:

$$H = \sum_{i=1}^n \bar{x}_i \bar{x}_i^T \frac{\partial^2 g}{\partial \eta_i^2} = \sum_{i=1}^n \bar{x}_i \bar{x}_i^T w_i \quad (9)$$

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

$$H = X^T W X \quad (10)$$

IRLS

We use Newton's algorithm to find MLE for the AFT model, using negative loglikelihood (NLL). The Newton update is as follows:

$$\begin{aligned} \beta &= \tilde{\beta} - H^{-1} \tilde{S} \\ &= \tilde{\beta} - (X^T \tilde{W} X)^{-1} X^T \tilde{\mu} \\ &= (X^T \tilde{W} X)^{-1} ((X^T \tilde{W} X) \tilde{\beta} - X^T \tilde{\mu}) \\ &= (X^T \tilde{W} X)^{-1} X^T (\tilde{W} X \tilde{\beta} - \tilde{\mu}) \\ &= (X^T \tilde{W} X)^{-1} X^T (\tilde{W} X \tilde{\beta} - \tilde{\mu}) \end{aligned} \quad (11)$$

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.

Hence, at each step we are solving a weighted least squares problem, which is a minimizer of:

$$\sum_{i=1}^n \tilde{w}_i (\tilde{z}_i - \tilde{x}_i^T \beta)^2 \quad (12)$$

This algorithm is the iteratively reweighted least squares (IRLS), since at each iteration we solve a weighted least squares problem.

Elastic net penalty and coordinate descent

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

$$\lambda P_\alpha(\beta) = \lambda(\alpha \|\beta\|_1 + 1/2(1 - \alpha) \|\beta\|_2^2) \quad (13)$$

Adding the elastic net (L1 + L2) penalty, we get the following penalized weighted least squares objective:

$$M = \sum_{i=1}^n \tilde{w}_i (\tilde{z}_i - \tilde{x}_i^T \beta)^2 + \lambda P_\alpha(\beta) \quad (14)$$

The subderivative of the optimization objective is given as:

$$\frac{\partial M}{\partial \beta_k} = \sum_{i=1}^n \tilde{w}_i x_{ik} (\tilde{z}_i - \tilde{x}_i^T \beta) + \lambda \alpha \operatorname{sgn}(\beta_k) + \lambda(1 - \alpha) \beta_k \quad (15)$$

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

Using the subderivative, three cases of solutions for β_k may be obtained.

The solution is given by:

$$\hat{\beta}_k = \frac{S \left(\sum_{i=1}^n \tilde{w}_i x_{ik} \left[\tilde{z}_i - \sum_{j \neq k} x_{ij} \beta_j \right], \lambda \alpha \right)}{\sum_{i=1}^n \tilde{w}_i x_{ik}^2 + \lambda(1 - \alpha)} \quad (16)$$

where, S is the soft thresholding operator, and w_i and z_i are given in 9 and 12 respectively.

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

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 λ , for a fixed α . We begin with λ sufficiently large to set the solution $\beta = 0$, and decrease λ until we arrive near the unregularized solution. The solutions for each value of λ are used as the initial estimates of β for the next λ value. This is known as warm starting, and makes the algorithm efficient and stable. To choose initial value of λ , we use Equation 16, 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 14. Thus,

$$\lambda_{max} = \max_j \frac{1}{n\alpha} \sum_{i=1}^n w_i(0)x_{ij}z(0)_i \quad (17)$$

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:

```

Calculate  $\lambda_{max}$  using 17 ;
foreach  $\lambda_j$  in  $j = 0, \dots, m$  do
    Initialise  $\tilde{\beta}$  using solution  $\hat{\beta}$  from previous  $\lambda$ ;
    repeat
        Compute  $\tilde{w}_i$  and  $\tilde{z}_i$  ;
        Find  $\hat{\beta}$  by solving the penalized weighted least square problem
            defined in 14 using coordinate descent ;
        Set  $\tilde{\beta} = \hat{\beta}$  ;
    until convergence of  $\hat{\beta}$ ;
end
```

Algorithm 1: Overall optimization algorithm

Scale parameter

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

Derivatives

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

$$\begin{aligned}
\frac{\partial g_1}{\partial \eta} &= -\frac{1}{\sigma} \left[\frac{f'(z)}{f(z)} \right] \\
\frac{\partial g_4}{\partial \eta} &= -\frac{1}{\sigma} \left[\frac{f(z^u) - f(z^l)}{F(z^u) - F(z^l)} \right] \\
\frac{\partial^2 g_1}{\partial \eta^2} &= -\frac{1}{\sigma^2} \left[\frac{f''(z)}{f(z)} \right] - (\partial g_1 / \partial \eta) \\
\frac{\partial^2 g_4}{\partial \eta^2} &= -\frac{1}{\sigma^2} \left[\frac{f'(z^u) - f'(z^l)}{F(z^u) - F(z^l)} \right] - (\partial g_4 / \partial \eta)^2 \\
\frac{\partial g_1}{\partial \log \sigma} &= -\left[\frac{zf'(z)}{f(z)} \right] \\
\frac{\partial g_4}{\partial \log \sigma} &= -\left[\frac{z^u f(z^u) - z^l f(z^l)}{F(z^u) - F(z^l)} \right] \\
\frac{\partial^2 g_1}{\partial (\log \sigma)^2} &= \left[\frac{z^2 f''(z) + zf'(z)}{f(z)} \right] - (\partial g_1 / \partial \log \sigma)^2 \\
\frac{\partial^2 g_4}{\partial (\log \sigma)^2} &= \left[\frac{(z^u)^2 f'(z^u) - (z^l)^2 f'(z^l)}{F(z^u) - F(z^l)} \right] - (\partial g_1 / \partial \log \sigma)(1 + \partial g_1 / \partial \log \sigma) \\
\frac{\partial^2 g_1}{\partial \eta \partial \log \sigma} &= \left[\frac{zf''(z)}{\sigma f(z)} \right] - (\partial g_1 / \partial \eta)(1 + \partial g_1 / \partial \log \sigma) \\
\frac{\partial^2 g_4}{\partial \eta \partial \log \sigma} &= \left[\frac{z^u f'(z^u) - z^l f'(z^l)}{\sigma [F(z^u) - F(z^l)]} \right] - (\partial g_4 / \partial \eta)(1 + \partial g_4 / \partial \log \sigma)
\end{aligned} \tag{18}$$

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

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