

# Survival Analysis with Semiparametric Proportional Hazard Models

Ziang

11/09/2019

## 1 Model Specification and Methodology

### 1.1 Types of Censoring:

In survival analysis, we are mainly dealing with the problem of right-censoring, interval-censoring and left truncation. Right-censoring is when an individual's lifetime  $T_i$  is not exactly known because the individual is still alive when the study terminates at  $C_i$ , so we are only sure about that  $T_i > C_i$  but not sure what exactly  $T_i$  is. Interval-censoring on the other hand, arises when the survival time is only known to be in an interval  $(L_i, U_i)$ , and the left truncation problem happens when some survival times are not recorded unless they are bigger than a specified start time  $t^{tr}$ , so all the data with survival times less than  $t^{tr}$  are missed.

However, Cox's semiparametric proportional hazard model cannot be handled using INLA-like algorithm when the problem of interval-censoring is present, since the trick that INLA uses to transform log-likelihood will no longer work in this case. Therefore, we will focus on the right-censoring problem right now, and come back to solve the cases of interval-censoring and left truncation later. Next, the three types of common right-censoring will be present.

Type-I right-censoring occurs when each individual's censoring time  $C_i$  is fixed and known beforehand. That means when we collect a bunch of survival times, we know whether each survival time is right-censored and when is it censored exactly. In this case, we will be able to write our original dataset  $\{T_i, C_i : i = 1, \dots, n\}$  as  $\{t_i, \delta_i : i = 1, \dots, n\}$  where:

$$t_i = \min\{T_i, C_i\}, \quad \delta_i = I(T_i \geq C_i) \quad (1)$$

This is the most common type of right-censoring, and we will focus on this type of censoring for the rest of the passage.

Type-II right-censoring occurs when we only observed the  $r$  smallest survival times in our sample. So the survival times that we can observed will be like  $t_{(1)} < t_{(2)} < \dots < t_{(r)}$ , and the other survival times will be censored so we don't know the exact numbers. In this scenario, we have a censoring time  $t_{(r)}$  that is itself random.

Lastly, independent random censoring happens when both the  $i$ th survival time  $T_i$  and the  $i$ th censoring time  $C_i$  are random variable that are independent.

### 1.2 Semiparametric Proportional Hazard Model:

Firstly, we break the time axis into  $K$  intervals with endpoints  $0 = s_0 < s_1 < \dots < s_K < \max\{t_i : i = 1, \dots, n\}$ , and assumes that the baseline hazard function is constant in each interval. i.e:  $h_0(t) = \lambda_k$  for  $t \in (s_{k-1}, s_k)$ ,  $k = 1, 2, \dots, K$

The model that we will be focusing on will be the semiparametric proportional hazard model, specified at below:

$$\begin{aligned} h(t_i) &= h_0(t_i) \exp(\beta_1 x_{i1} + \dots \beta_p x_{ip}) \\ &= \exp(\log(\lambda_k) + \beta_1 x_{i1} + \dots \beta_p x_{ip}) \quad t_i \in (s_{k-1}, s_k] \end{aligned} \quad (2)$$

Let  $\eta_{ik} = \log(\lambda_k) + \beta_1 x_{i1} + \dots \beta_p x_{ip}$ , then the log-likelihood function for the  $i^{th}$  observation  $t_i \in (s_{k-1}, s_k]$  can be written as :

$$\begin{aligned} l &= \log[f(t_i)^{\delta_i} S(t_i)^{(1-\delta_i)}] \\ &= \log[h(t_i)^{\delta_i} S(t_i)] \\ &= \delta_i \eta_{ik} - (t_i - s_k) \exp(\eta_{ik}) - \sum_{j=1}^{k-1} [(s_{j+1} - s_j) \exp(\eta_{ij})] \end{aligned} \quad (3)$$

Similarly, the full log-likelihood can be derived as:

$$l = \sum_{i=1}^n \left\{ \delta_i \eta_{ik(i)} - (t_i - s_{k(i)}) \exp(\eta_{ik(i)}) - \sum_{j=1}^{k(i)-1} [(s_{j+1} - s_j) \exp(\eta_{ij})] \right\} \quad (4)$$

I emphasize the subscript for  $k(i)$  because each survival time will correspond to a different value of  $k$ , depending on which interval the survival time lies in.

Here the INLA algorithm cannot directly be applied, because if we look at the log-likelihood of a single survival time  $\{t_i, \delta_i\}$ , we can find that it depends on more than one  $\eta$ . To use INLA, we required a conditional independent latent field together with a sparse Hessian matrix for the log-likelihood. That means we need to make sure that for a single data point, the log-likelihood should be free of terms from latent field once we condition on one of the term from the latent field.

To solve this puzzle, we will utilize a data ‘‘augmentation’’ trick to transform the log-likelihood of a single data point into the form that INLA likes. Notice that if we are looking at a random variable  $X_i$  that follows a poisson distribution with mean  $(t_i - s_k) \exp(\eta_{ik})$ , then the log-likelihood corresponding to a single data point  $\{X_i = 0\}$  will be:

$$\begin{aligned} l &= \log \left( P(X_i = 0 | \lambda = (t_i - s_k) \exp(\eta_{ik})) \right) \\ &= 0 \times \ln((t_i - s_k) \exp(\eta_{ik})) - (t_i - s_k) \exp(\eta_{ik}) - \ln(0!) \\ &= -(t_i - s_k) \exp(\eta_{ik}) \end{aligned} \quad (5)$$

Similarly, when  $X_i = 1$ , the log-likelihood of this single data point is:

$$\begin{aligned} l &= \log \left( P(X_i = 1 | \lambda = (t_i - s_k) \exp(\eta_{ik})) \right) \\ &= 1 \times \ln((t_i - s_k) \exp(\eta_{ik})) - (t_i - s_k) \exp(\eta_{ik}) - \ln(1!) \\ &= \ln(t_i - s_k) + \eta_{ik} - (t_i - s_k) \exp(\eta_{ik}) \\ &\propto \eta_{ik} - (t_i - s_k) \exp(\eta_{ik}) \end{aligned} \quad (6)$$

Here we can basically ignore the term  $\ln(t_i - s_k)$  as it does not depend on any term from the latent field. So when we later take derivative, this term will just disappear which means it won't affect our  $C$  matrix.

We showed that the first two terms of the log-likelihood of a single data point  $\{t_i, \delta_i\}$  can be viewed as the log-likelihood of a single data point  $X_i \sim \text{Poisson}(\lambda = (t_i - s_k) \exp(\eta_{ik}))$  being 0 when  $\delta_i = 0$  and being 1 when  $\delta_i = 1$ .

Next step will be to figure out a similar way to deal with the last term in equation (3). Notice that for a Poisson random variable  $Y_j$  with mean  $(s_{j+1} - s_j) \exp(\eta_{ij})$ , the log-likelihood for observing it being 0 will be:

$$\begin{aligned} l &= \log \left( P(Y_j = 0 | \lambda = (s_{j+1} - s_j) \exp(\eta_{ij})) \right) \\ &= -(s_{j+1} - s_j) \exp(\eta_{ij}) \end{aligned} \quad (7)$$

Similarly, if we gather a sample of  $\{Y_{i_1} = 0, Y_{i_2} = 0, \dots, Y_{i_k} = 0\}$  where each  $Y_{i_j} \sim \text{Poisson}(\lambda = (s_{j+1} - s_j)\exp(\eta_{ij}))$  is independent of others, then the log-likelihood of this sample will simply be the sum of log-likelihood of each term due to independence, which sums to be  $\sum_{j=1}^{k-1} (s_{j+1} - s_j)\exp(\eta_{ij})$ , that is exactly what we want.

Putting these two pieces information together, which means if we have a sample being  $\{X_i = \delta_i, Y_{i_1} = 0, Y_{i_2} = 0, \dots, Y_{i_k} = 0\}$ , and all the terms in this sample being mutually independent, then the log-likelihood of this sample will just be the log-likelihood of the single data point  $\{t_i, \delta_i\}$ . Doing this for all the data points  $\{t_i, \delta_i | i = 1, \dots, n\}$ . We retrieve the original log-likelihood from the log-likelihood of a sample of  $\sum_{i=1}^n k(i)$  number of independent, but non-identical Poisson random variables. In other words, we augment our original dataset  $\{t_i, \delta_i | i = 1, \dots, n\}$  into a huge dataset  $\{x_i, y_{i_1}, y_{i_2}, \dots, y_{i_{k(i)}} | i = 1, 2, \dots, n\}$ , where all the terms in this new dataset are mutually independent. This is the cure for our problem since the log-likelihood of each term from this new ‘‘augmented’’ dataset, will only depend on the latent field through one  $\eta$ .

### 1.3 Methodology for Approximation:

Here I will present how the Bayesian approximation can be carried out using an INLA-type of algorithm. Firstly, to make the covariance matrix of the joint gaussian latent field non-singular, and to simplify the Hessian matrix that we are going to derive later, we will assume that for each  $\eta_{ij}$ , a normal random noise  $\epsilon_{ij}$  is added. We assume that  $\epsilon_{ij} \sim N(0, \sigma_\epsilon^2)$  being mutually independent across different  $i$  and  $j$ . In other words, we will write  $\eta_{ij} = \log(\lambda_k) + \beta_1 x_{i1} + \dots + \beta_p x_{ip} + U_i + \eta_{ij}$ , where  $U_i$  is any random effect that we believe exists in the context of the study.

Then, the latent field can be denoted as:

$$\tilde{W} = (\eta_{11}, \eta_{12}, \dots, \eta_{1k}, \eta_{2k}, \dots, \eta_{nk}, \beta_1, \dots, \beta_p, \log(\lambda_1), \dots, \log(\lambda_k))^T \quad (8)$$

Besides assume that  $\tilde{W}$  is a GMRF, we also assume that  $\log(\lambda_{k+1}) - \log(\lambda_k)$  follows  $N(0, \tau^{-1})$ , a RW1 model. So we will just use  $\tilde{\theta}$  to denote the hyperparameter vector that determines the precision matrix of our latent field.

Now, let's derive the negated Hessian matrix of the log-likelihood with respect to the latent field. To do that, let's first consider the log-likelihood consider only one survival time  $\{t_i, \delta_i\}$  where  $t_i \in (s_{k(i)-1}, s_{k(i)})$ . In this case, the log-likelihood for this data point will be:

$$l = \delta_i \eta_{ik(i)} - (t_i - s_{k(i)}) \exp(\eta_{ik(i)}) - \sum_{j=1}^{k(i)-1} [(s_{j+1} - s_j) \exp(\eta_{ij})] \quad (9)$$

The derivative with respect to  $\eta_{ik(i)}$  will be

$$\frac{\partial l}{\partial \eta_{ik(i)}} = \delta_i - (t_i - s_{k(i)}) \exp(\eta_{ik(i)}) \quad (10)$$

That means the negated second derivative will be:

$$-\frac{\partial^2 l}{\partial \eta_{ik(i)}^2} = (t_i - s_{k(i)}) \exp(\eta_{ik(i)}) \quad (11)$$

For first and negated second derivatives with  $\eta_{ij}$  where  $j < k(i)$ , we have:

$$\begin{aligned} \frac{\partial l}{\partial \eta_{ij}} &= -(s_{j+1} - s_j) \exp(\eta_{ij}) \\ -\frac{\partial^2 l}{\partial \eta_{ij}^2} &= (s_{j+1} - s_j) \exp(\eta_{ij}) \end{aligned} \quad (12)$$

Apparently, for  $\eta_{ij}$  where  $j > k_{(i)}$ , we have the second derivatives of log-likelihood being 0's. Combine them together, we know that the negated Hessian matrix for the log-likelihood of  $\{t_i, \delta_i\}$ ,  $H_i$  will be:

$$\begin{bmatrix} (s_2 - s_1)exp(\eta_{i1}) & 0 & 0 & \dots & \dots & \dots & 0 \\ 0 & (s_3 - s_2)exp(\eta_{i2}) & 0 & \ddots & & & \\ 0 & 0 & \ddots & & \ddots & & \\ \vdots & \dots & \dots & (s_{k_{(i)}} - s_{k_{(i)}-1})exp(\eta_{i(k_{(i)}-1)}) & \ddots & \vdots & \\ \vdots & & \ddots & 0 & (t_i - s_{k_{(i)}})exp(\eta_{ik_{(i)}}) & \dots & \vdots \\ \vdots & & & \ddots & & \ddots & \vdots \\ \vdots & & & & \dots & \dots & \vdots \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \quad (13)$$

This is a very sparse matrix with only diagonal terms.

Repeating this procedure for the rest data points, using the property of independence, we can get the negated Hessian matrix  $H$  for the full log-likelihood will be:

$$H = \begin{pmatrix} H_1 & 0 & 0 & \dots & & \\ 0 & H_2 & 0 & \dots & & \\ & \dots & \ddots & & & \\ & & & H_n & \dots & \vdots \\ & & & & \ddots & \vdots \\ & & & & & 0 \end{pmatrix} \quad (14)$$

Here we build a block diagonal matrix  $H$  using each block  $H_i$  obtained from above procedures. The negated Hessian matrix is very sparse, which is exactly what we want it to be. Then, we will try to derive the precision matrix  $Q$  of the latent field (To be continued).

## 2 Example from Diabetics Dataset

Firstly, I will use the dataset “diabetics” to demonstrate the equivalence between “coxph” approach in survival package and inla approach.

Let us briefly view the structure of “diabetics”:

### 2.1 Dataset:

```
head(as_tibble(diabetic))
```

```
## # A tibble: 6 x 8
##   id laser age eye   trt risk time status
##   <int> <fct> <int> <fct> <int> <int> <dbl> <int>
## 1     5 argon  28 left     0     9  46.2     0
## 2     5 argon  28 right    1     9  46.2     0
## 3    14 xenon  12 left     1     8  42.5     0
## 4    14 xenon  12 right    0     6  31.3     1
## 5    16 xenon   9 left     1    11  42.3     0
## 6    16 xenon   9 right    0    11  42.3     0
```

We can see that those survival times are right-censored. We will fit a cox proportional hazard model with piece-wise constant baseline hazard, assuming that each individual will have the same baseline hazard function. The variable ID will be treated as a random effect(so called frailty). The variables “age”, “eye”, “trt” and “laser” will be included as fixed effects.

## 2.2 Survival: coxph

```
diabetic.CoxPh <- coxph(Surv(time, status)~age + eye + trt + laser + frailty.gaussian(id), data = diabetic)
summary(diabetic.CoxPh)
```

```
## Call:
## coxph(formula = Surv(time, status) ~ age + eye + trt + laser +
##       frailty.gaussian(id), data = diabetic)
##
##      n= 394, number of events= 155
##
##              coef          se(coef) se2          Chisq  DF      p
## age              0.009548  0.01323  0.009879   0.52  1.00  4.7e-01
## eyeright         0.483005  0.17501  0.168693   7.62  1.00  5.8e-03
## trt             -1.007507  0.17930  0.174315  31.57  1.00  1.9e-08
## laserargon      -0.182388  0.39471  0.293566   0.21  1.00  6.4e-01
## frailty.gaussian(id)                                131.35 79.63 2.4e-04
##
##          exp(coef) exp(-coef) lower .95 upper .95
## age             1.0096    0.9905    0.9837    1.0361
## eyeright        1.6209    0.6169    1.1503    2.2842
## trt              0.3651    2.7388    0.2569    0.5189
## laserargon      0.8333    1.2001    0.3844    1.8062
##
## Iterations: 6 outer, 24 Newton-Raphson
##      Variance of random effect= 0.9447455
## Degrees of freedom for terms=  0.6  0.9  0.9  0.6 79.6
## Concordance= 0.867 (se = 0.867 )
## Likelihood ratio test= 228 on 82.62 df,  p=2e-15
```

From the output above, it can be seen that variables “age” and “eyeright” have positive association with the rate of occurrence of visual loss, and variables “trt” and whether using “argon” type of laser are negatively associated with the rate. All the fixed effects that we included in this study have significant effects for the risk of visual loss.

## 2.3 Bayesian: INLA

```
formula = inla.surv(time, status) ~ age + eye + trt + laser + f(id, model = "iid")
diabetic.INLA <- inla(formula, control.compute = list(dic = TRUE), family = "coxph",
                      data = diabetic, control.hazard=list(model="rw2", n.intervals=20))
diabetic.INLA$summary.fixed
```

```
##              mean          sd 0.025quant      0.5quant 0.975quant
## (Intercept) -4.763726793 0.2499475 -5.27243977 -4.757708156 -4.2888292
## age          0.007159506 0.0108562 -0.01385358  0.007023392  0.0290777
## eyeright     0.385771765 0.1764002  0.04932060  0.382228898  0.7428639
## trt          -0.867268284 0.1924494 -1.26424348 -0.860395035 -0.5083653
```

```
## laserargon -0.118800233 0.3229562 -0.77667890 -0.111786522 0.4986221
##               mode          kld
## (Intercept) -4.746083656 3.873797e-06
## age          0.006833516 8.319500e-06
## eyeright     0.375244209 6.176833e-06
## trt          -0.845994482 1.479641e-05
## laserargon -0.098991670 3.897219e-06
```

It seems like these two approaches are similar enough. Though in the classic frequentist approach, the effects of age and using argon-type laser are significant, but inla gives insignificant results (the 95% credible interval contains zero).

### 3 Example from Bladder Dataset:

Next, we will study the two approaches on the dataset “bladder1”:

#### 3.1 Dataset:

```
data <- as_tibble(bladder1)
data <- select(data, -c(rsize, rtumor, enum))
data <- data %>% mutate(censored = status==0)
for (i in 1:length(data$censored)) {
  if(data$censored[i]) data$censored[i] <- 1
  else data$censored[i] <- 0
}
data <- data %>% mutate(times = stop-start)
head(data)
```

```
## # A tibble: 6 x 10
##       id treatment number  size recur start  stop status censored times
##   <int> <fct>      <int> <int> <int> <int> <int> <dbl>    <dbl> <int>
## 1     1 placebo         1     1     0     0     0     3         0     0
## 2     2 placebo         1     3     0     0     1     3         0     1
## 3     3 placebo         2     1     0     0     4     0         1     4
## 4     4 placebo         1     1     0     0     7     0         1     7
## 5     5 placebo         5     1     0     0    10     3         0    10
## 6     6 placebo         4     1     1     0     6     1         0     6
```

```
tail(data)
```

```
## # A tibble: 6 x 10
##       id treatment number  size recur start  stop status censored times
##   <int> <fct>      <int> <int> <int> <int> <int> <dbl>    <dbl> <int>
## 1   115 thiotepa         4     1     3    24    47     1         0    23
## 2   115 thiotepa         4     1     3    47    50     0         1     3
## 3   116 thiotepa         3     4     0     0    54     0         1    54
## 4   117 thiotepa         2     1     1     0    38     1         0    38
## 5   117 thiotepa         2     1     1    38    54     0         1    16
## 6   118 thiotepa         1     3     0     0    59     3         0    59
```

Here the variable “id” specifies different individuals, and should be treated as a random effect. The variable “start” and “stop” denote the start time and end time of each time interval. It seems like a interval censoring problem but the start time is known before hand, so we can treat it as a regular type-I right censoring.

In this study, a nonzero value of “status” can be death from bladder disease, death from other reason or recurrence. Here we will just view all of these situations as “occurrence” for simplicity. We will include “number”, “size” and “recur” as fixed effects in this study.

### 3.2 Survival:coxph

```
bladder.CoxPh <- coxph(Surv(times, censored) ~ number + size + recur + frailty.gaussian(id), data = data,
summary(bladder.CoxPh)
```

```
## Call:
## coxph(formula = Surv(times, censored) ~ number + size + recur +
##       frailty.gaussian(id), data = data)
##
##      n= 294, number of events= 76
##
##              coef          se(coef) se2      Chisq DF    p
## number           0.070722  0.07808  0.07805  0.82  1.00  0.37
## size            -0.077520  0.07250  0.07243  1.14  1.00  0.28
## recur           -0.009121  0.06595  0.06593  0.02  1.00  0.89
## frailty.gaussian(id)                                0.07  0.09  0.53
##
##      exp(coef) exp(-coef) lower .95 upper .95
## number    1.0733    0.9317    0.9210    1.251
## size      0.9254    1.0806    0.8028    1.067
## recur     0.9909    1.0092    0.8708    1.128
##
## Iterations: 10 outer, 45 Newton-Raphson
##      Variance of random effect= 0.001302083
## Degrees of freedom for terms= 1.0 1.0 1.0 0.1
## Concordance= 0.571 (se = 0.571 )
## Likelihood ratio test= 2.01 on 3.09 df,    p=0.6
```

Fitting this model using the traditional partial likelihood approach gives insignificant results for all the fixed effects. But we will still proceed to check what will happen if we fit it using a Bayesian approach.

### 3.3 Bayesian: INLA

```
formula = inla.surv(times, censored) ~ number + size + recur + f(id, model = "iid")
bladder.INLA <- inla(formula, control.compute = list(dic = TRUE), family = "coxph",
                     data = data, control.hazard=list(model="rw2", n.intervals=20))
bladder.INLA$summary.fixed
```

```
##              mean          sd 0.025quant    0.5quant  0.975quant
## (Intercept) -3.215860744 0.24119932 -3.69834764 -3.212779112 -2.75090997
## number       0.072456595 0.07733310 -0.08906334  0.075832434  0.21490330
## size        -0.056131419 0.06805192 -0.19699962 -0.053601172  0.07027963
## recur       -0.006008487 0.06521873 -0.13954475 -0.004063216  0.11656202
##              mode          kld
## (Intercept) -3.2066709495 5.562343e-08
## number       0.0825998635 2.719313e-06
## size        -0.0485569185 2.050413e-06
## recur       -0.0002009621 1.277043e-06
```

Indeed, the two results seem pretty similar in general. In both cases, we can see that there are no apparent relationships between those fixed effects and the rate of occurrence of bladder cancer's recurrence, or death.