Survival Analysis with Applications in Medicine: Take-home examination

2025-03-19

A. Weibull regression models

Q1. For a proportional hazards model with $S_0(t)$ that is a Weibull distribution, show that survival $S(t)$ is also from a Weibull distribution.

Given a proportional hazards model with survival function $S(t|x) = S_0(t)^{\exp(\beta x)}$

for time t and a given covariate x, $S_0(t)$ is the baseline survival function, and β is the log hazard ratio.

We have the Weibull survival function $S_0(t) = \exp(-\lambda t^k)$ for a scale parameter λ and a shape parameter k.

Substituting $S_0(t)$ in the proportional hazards model, we get the survival function as

$S(t|x) = \left[\exp(-\lambda t^k)\right]^{\exp(\beta x)}$

$\Rightarrow S(t|x) = \exp(-\lambda t^k \exp(\beta x))$

which is in form $S(t|x) = \exp(-\tilde{\lambda}_a t^k)$ where $\tilde{\lambda}_a = \lambda \exp(\beta x)$ is the new scale parameter for the same shape parameter k.

This means that the survival function $S(t|x)$ for a proportional hazards model is also from a Weibull distribution.

Here, $S(t) = E_X[S(t \mid X)]$, which will also be from a Weibull distribution.

Q2. For an accelerated failure time model with $S_0(t)$ that is a Weibull distribution, show that survival $S(t)$ is also from a Weibull distribution.

Given an accelerated failure time model with survival function $S(t|x) = S_0(t \exp(-\tilde{\beta} x))$

for time t and a given covariate x, $S_0(t)$ is the baseline survival function, and $\tilde{\beta}$ is the log time ratio.

Substituting $S_0(t)$ in the accelerated failure time model, we get the survival function as

$S(t|x) = \exp(-\lambda \big(t \exp(-\tilde{\beta} x)\big)^k)$

$\Rightarrow S(t|x) = \exp(-\lambda t^k \exp(-k\tilde{\beta} x))$

which is in form $S(t|x) = \exp(-\tilde{\lambda}_b t^k)$ where $\tilde{\lambda}_b = \lambda \exp(-k\tilde{\beta} x)$ is the new scale parameter for the same shape parameter k.

This means that the survival function $S(t|x)$ for an accelerated failure time model is also from a Weibull distribution.

Similarly, $S(t) = E_X[S(t \mid X)]$, which will also be from a Weibull distribution.

Q3. What is the relationship between β and $\tilde{\beta}$ if both models have a Weibull baseline survival function?

For the proportional hazards model, we have $S(t|x) = \exp(-\lambda t^k \exp(\beta x))$.

For the accelerated failure time model, we have $S(t|x) = \exp(-\lambda t^k \exp(-k\tilde{\beta}x))$.

Comparing the two models, we get $\exp(\beta x) = \exp(-k\tilde{\beta}x)$.

Taking the natural logarithm of both sides, we get $\beta x = -k\tilde{\beta}x$.

Therefore, the relationship between β and $\tilde{\beta}$ is $\beta = -k\tilde{\beta}$ if both models have a Weibull baseline survival function.

B: Interval-censored likelihood

For a data tuple (t_i, u_i, v_i) where t_i is the (left truncated) delayed entry time, and the event is observed in the interval $(u_i, v_i]$ for an individual i.

Q1a. Express the log-likelihood in terms of Survival function $S(t)$ at time t:

The Likelihood for the interval-censored data: $u_i < T \leq v_i$ for an entry time t_i is given by:

$L_i = \frac{S(u_i)-S(v_i)}{S(t_i)}$

where $S(t)$ is the Survival function.

Hence, the log-likelihood in terms of Survival function $S(t)$ at time t is given by $\log(S(u_i) - S(v_i)) - \log(S(t_i))$.

Q1b. Express the log-likelihood in terms of the hazard function $h(t)$ at time t:

Now, for the derivation of the log-likelihood in terms of the hazard function $h(t)$ at time t, we need to express the Survival function $S(t)$ in terms of the hazard function $h(t)$.

We know that, the survival function $S(t)$ is given by $-log(S(t)) = H(t)$,

where $H(t)$ is the cumulative hazard function, and $H(t) = \int_0^t h(u)du$ for the hazard function $h(t)$.

Therefore, the log-likelihood in terms of the hazard function $h(t)$ at time t is:

$\log L_i = \log(S(u_i) - S(v_i)) - \log(S(t_i))$

$\Rightarrow \log L_i = \log(exp(-H(u_i)) - exp(-H(v_i))) - \log(exp(-H(t_i)))$

$\Rightarrow \log L_i = \log(exp(-H(v_i)) * (exp(H(v_i) - H(u_i)) - 1)) + H(t_i)$

$\Rightarrow \log L_i = \log(exp(-H(v_i))) + \log(exp(H(v_i) - H(u_i)) - 1) + H(t_i)$

$\Rightarrow \log L_i = H(t_i) - H(v_i) + \log(exp(H(v_i) - H(u_i)) - 1)$

$\Rightarrow \log L_i = \int_0^{t_i} h(t)dt - \int_0^{v_i} h(t)dt + \log(exp(\int_{u_i}^{v_i} h(t)dt) - 1)$

$\Rightarrow \log L_i = -\int_{t_i}^{v_i} h(t)dt + \log(exp(\int_{u_i}^{v_i} h(t)dt) - 1)$

Hence, the log-likelihood in terms of the hazard function $h(t)$ at time t is given by $-\int_{t_i}^{v_i} h(t)dt + \log(exp(\int_{u_i}^{v_i} h(t)dt) - 1)$.

Q2. Can you express these data using the Surv function from the survival package? If so, show an example; if not, explain why.

Yes, we can express the interval-censored data using the `Surv` function from the `survival` package.

The `Surv` function is used to create a survival object that represents the survival time of an individual. It takes the form `Surv(time, event)` where `time` is the survival time and `event` is the event indicator.

For interval-censored data, we can use the `Surv` function as `Surv(time, time2, type = "interval2")` where `time` is the start of the interval, `time2` is the end of the interval and type `interval2` is used to indicate interval-censored data effectively.

Although, the `Surv` function doesn't support left truncation directly, we can filter out the left truncated data by taking the maximum of the entry time and the left truncation time.

Here is an example of how to express left-truncated interval-censored data using the `Surv` function:

```r
library(survival)

# sample data for the given data tuple structure (t_i, u_i, v_i)
d = data.frame(left_truncation_time = c(1,1,3,3,3),
               entry_time = c(0,0,2,2,2),
               exit_time = 1:5,
               event = c(1,0,1,0,1))

# filter out left truncated data
d$entry_time = pmax(d$entry_time, d$left_truncation_time)

# interval-censored data
with(d, Surv(entry_time, exit_time, type="interval2"))
```

```
## [1] 1      [1, 2] 3      [3, 4] [3, 5]
```

C: Truncated distributions

For a continuous random variable T, we have the survival function $S(t) = P(T > t)$.

Let the quantile function $Q(p)$ such that $Q(p) = S^{-1}(1-p) \Rightarrow 1 - p = S(Q(p)) = P(T > Q(p))$.

Assume that we have a truncated distribution with survival function $P(T > t|T > t_0)$ for left truncation time t_0.

Q1. Express the survival function for the truncated distribution in terms of the survival function for T.

The survival function for the truncated distribution is given by:

$P(T > t|T > t_0)$

$\Rightarrow \frac{P(T>t,T>t_0)}{P(T>t_0)}$ by definition of conditional probability.

$\Rightarrow \frac{P(T>t)}{P(T>t_0)}$ because $T > t, T > t_0$ implies $T > t$ as t_0 is the left truncation time.

Since, $P(T > t) = S(t)$ and $P(T > t_0) = S(t_0)$ by the survival function for T,

the survival function for the truncated distribution is given by $\frac{S(t)}{S(t_0)}$.

Q2. For the truncated distribution, what is the quantile function $Q(p|t_0)$ that solves $P(T > t|T > t_0) = 1 - p$ for t in terms of the survival and quantile functions for T at quantile (probability) p?

Given $P(T > t|T > t_0) = 1 - p$,

we know that $P(T > t|T > t_0) = \frac{S(t)}{S(t_0)} = 1 - p$.

Therefore, the quantile function $Q(p|t_0) = t$ that solves $P(T > t|T > t_0) = 1 - p$:

$S(t) = (1 - p)S(t_0)$

$\Rightarrow t = S^{-1}((1 - p)S(t_0))$

$\Rightarrow Q(p|t_0) = S^{-1}((1 - p)S(t_0))$.

To solve for $S^{-1}((1 - p)S(t_0))$, we know that $Q(p) = S^{-1}(1 - p) \rightarrow Q(1 - p) = S^{-1}(p)$.

Hence, for a y, we can say that $S^{-1}(y) = Q(1 - y)$,

$\Rightarrow Q(p|t_0) = S^{-1}((1 - p)S(t_0)) = Q(1 - (1 - p)S(t_0))$.

$\Rightarrow Q(p|t_0) = Q(1 - (1 - p)S(t_0))$.

i.e., the p-th quantile of left truncated distribution is the $(1 - (1 - p)S(t_0))$-th quantile of the original distribution.

Q3. Using this algorithm, write, run and report on R code to calculate the 0.4 quantile from a truncated log-normal distribution where $T \sim LogNormal(\mu = 1, \sigma^2 = 1.2^2)$ for a log-normal distribution with mean μ and standard deviation σ on the log scale for $t_0 = 2$.

We can derive the quantile function $Q(p = 0.4|t_0 = 2)$ as follows:

$Q(p|t_0) = Q(1 - (1 - p)S(t_0))$.

To derive $S(t_0)$, we know that $S(t_0) = P(T > t_0) = 1 - P(T \leq t_0) = 1 - F(t_0)$,

where $F(t)$ is the cumulative distribution function for T.

For a log-normal distribution of T, we can compute $S(t_0)$ using `plnorm` function in R.

Then, we can substitute the $S(t_0)$ value back to $Q(1 - (1 - p)S(t_0))$ and compute the quantile using `qlnorm` function in R for the probability $p = 0.4$.

```r
#' @param p is the probability
#' @param meanlog mean on the log scale
#' @param sdlog standard deviation on the log scale
#' #' @param t0 left truncation time
#' @return the quantile from a truncated log-normal distribution

f = function(p, meanlog, sdlog, t0) {
  S2 = 1 - plnorm(t0, meanlog, sdlog)
  Q = qlnorm(1 - (1-p)*S2, meanlog, sdlog)
  return(Q)
}

# function call
f(p = 0.4, meanlog = 1, sdlog = 1.2, t0 = 2)
```

```
## [1] 4.171994
```

Hence, the 0.4 quantile from a truncated log-normal distribution with $T \sim LogNormal(\mu = 1, \sigma^2 = 1.2^2)$ for $t_0 = 2$ is approximately 4.172.

Q4. Check your value of the 0.4 quantile from the truncated log-normal distribution in question C3 by given random sampling code.

Given `R` code to return a vector of random numbers t sampled from a truncated log-normal distribution:

```r
#' @param n the number of random numbers
#' @param meanlog mean on the log scale
#' @param sdlog sd on the log scale
#' @param t0 left truncation time(s)
#' @return vector of random numbers drawn from a truncated log-normal distribution
rtrunc_lnorm = function(n, meanlog, sdlog, t0) {
    y = rlnorm(n, meanlog, sdlog)
    while (any(y<t0))
        y[y<t0] = rlnorm(n, meanlog, sdlog)[y<t0]
    y
}
```

Now, let's try to compute the 0.4 quantile for the above random numbers using the `quantile` function in R for large sample sizes of $n = 10,000$ and $n = 100,000$.

```r
# set seed for reproducibility
set.seed(123)

# generate 10,000 random numbers from truncated log-normal distribution
y = rtrunc_lnorm(n = 10000, meanlog = 1, sdlog = 1.2, t0 = 2)

# calculate the 0.4 quantile
quantile(y, 0.4)
```

```
##      40%
## 4.149261
```

```r
# generate 100,000 random numbers from truncated log-normal distribution
y = rtrunc_lnorm(n = 100000, meanlog = 1, sdlog = 1.2, t0 = 2)

# calculate the 0.4 quantile
quantile(y, 0.4)
```

```
##      40%
## 4.164862
```

We can see that the 0.4 quantile is approximately 4.149 and 4.165 by randomly sampling t values from a truncated log-normal distribution of T for $n = 10,000$ and $n = 100,000$ respectively. This is very close to the value of 4.172 obtained from the algorithm in question C3.

D: Cox's partial likelihood with a time-varying effects

E: Data analysis of a randomised controlled trial for hormonal treatment of breast cancer patients in Germany

---

F: Analysis plan for a randomised controlled trial

I am declaring that I have used generative artificial intelligence (GAI) to assist me in completing this assignment in the form of Github co-pilot to auto-complete my explanations.

This assignment took me approximately 7 hours to complete.