

Mimic Dikta's simulation

2020-03-30

Dikta proposed another semi-parametric estimator with consideration of covariates. The estimator is derivatived from the estimator generated by Stute (1993), instead of the product limit estimator from Kaplan Meier.

The Stute estimator (which estimates $E(\phi(x, t))$):

$$F_0(x, t) = P(X \leq x, Z \leq t) = \sum_{i=1}^n W_i \phi(x, t) \quad (1)$$

If we set $\phi(x, t) = I_{(X_i \leq x, Z_i \leq t)}$, it leads to

$$F_0(x, t) = \sum_{i=1}^n W_i I_{(X_i \leq x, Z_i \leq t)} = \sum_{i=1}^n \left\{ \frac{\delta_i}{n-i+1} \left[\prod_{j=1}^{i-1} \left(1 - \frac{1}{n-j+1} \right)^{\delta_i} \right] \right\} I_{(X_i \leq x, Z_i \leq t)} \quad (2)$$

where $W_i = \frac{\delta_i}{n-i+1} \prod_{j=1}^{i-1} \left(1 - \frac{1}{n-j+1} \right)^{\delta_i}$.

Dikta replaced δ_i with $m(t, x)$ and got

$$S^{D3} = \sum_{i=1}^n \left\{ \frac{m(Z_i, X_i)}{n-i+1} \left[\prod_{j=1}^{i-1} \left(1 - \frac{m(Z_i, X_i)}{n-j+1} \right) \right] \right\} I_{(X_i \leq x, Z_i \leq t)} \quad (3)$$

Therefore, the esimators of Cox PH model and of Dikta's model are actually estimating different things, i.e.

Cox

$$S^{cox}(t|X = x) = P(T > t|X = x)$$

Dikta

$$S^{D3} = P(T < t, X < x) = P(T < t, X_1 < x_1, \dots, X_p < x_p)$$

Therefore, to calculate $P(T < t, X < x)$ from $P(T < t|X = x)$, we then need some transformations.

Since,

$$\begin{aligned} \int_0^t f_{T,X}(t, x) dt &= \int_0^t f_{T|X}(t|x) f_X(x) dt \\ &= f_X(x) \int_0^t f_{T|X}(t|x) dt \\ &= f_X(x) P(T < T|X = x) \end{aligned}$$

then

- $P(T < t|X = x) = \frac{1}{f_X(x)} \int_0^t f_{T,X}(t, x) dt$
- $f_{T,X}(t, x) = f_X(x) \frac{\partial}{\partial t} P(T < t|X = x)$
- $F_{T,X}(t, x) = \int_0^x \int_0^t f_{T,X}(t, x) dt dx$
- $= \int_0^x \int_0^t f_X(x) \frac{\partial}{\partial t} P(T < t|X = x) dt dx$
- $= \int_0^x f_X(x) P(T < t|X = x) dx$

Then for example, if we have one dimension covariate $X \sim N(0, 1)$. The conditional distribution

$$\begin{aligned}
P(T > t|X = x) &= S(t|x) = S_0(t)^{\exp(x)} = \exp(-t \times \exp(x)) \\
P(T < t|X = x) &= 1 - P(T > t|X = x) = 1 - \exp(-t \times \exp(x)) \\
F_{T,X}(t, x) &= \int_0^x f_X(x) P(T < t|X = x) dx \\
&= \int_0^x \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2}) (1 - \exp(-t \times \exp(x))) dx \\
&= \Phi(x) - 0.5 - \int_0^x \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2} - t \exp(x)) dx
\end{aligned}$$

which does not have a close form.

Simulation settings in Dikta's paper

In Dikta's paper, he assume that $\psi(u, v) = I(u < x, v < y)$ for a given (x, y) point. Therefore, the Kaplan-Meier-biased estimator estimates $P(T < t, X < x)$.

Consider a one dimensional covariate, X , with distribution $X \sim N(0.5, 1.5^2)$. Consider the lifetime given the covariate has the model:

$$P(T < t|X = x) = 1 - \exp(-t \exp(-\rho x))$$

where $\rho = 1$ or $\rho = 3$. The censoring time follows the model

$$P(C > t) = \exp(-t)$$

Therefore, the $F_{T,X}(t, x)$ function can be written as:

$$\begin{aligned}
F_{T,X}(t, x) &= \int_0^x f_X(x) P(T < t|X = x) dx \\
&= \int_0^x \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(x - \mu)^2}{2\sigma^2}) (1 - \exp(-t \exp(-\rho x))) dx
\end{aligned}$$

And the $m()$ function is

$$m(x, t) = 1/(1 + \exp(-\rho x))$$

The author just looked at three special points, $(x, t) = (1.5, 0.1)$, $(x, t) = (1.5, 0.4)$, $(x, t) = (1.5, 1.3)$. We may calculate the $F_{T,X}(t, x)$ with those settings.

```

fxy = function(x,t){
  res = 0; h = 0.00001
  for(i in seq(0,x,h)){
    temp = 1/ (sqrt(2*pi)*sigma) *
      exp(-(i-mu)^2/(2*sigma^2)) *
      (1-exp(- t*exp(rho*i)))
    res = res + temp * h
  }
  res
  return(res)
}

```

	x = 1.5; rho = 1			x = 1.5; rho = 3		
	t = 0.1	t = 0.4	t = 1.3	t = 0.1	t = 0.4	t = 1.3
In Dikta paper	0.0917	0.2746	0.4967	0.2272	0.3446	0.4385
Mine calculation	0.0754	0.2148	0.3430	0.2195	0.3172	0.3665

However, the results are not quite similar.

For the calculation of Kaplan-Meier-based estimator:

For the calculation of Dikta's semiparameter estimator:

The Stute estimator (which estimates $E(\phi(x, t))$):

$$F_0(x, t) = \sum_{i=1}^n \left\{ \frac{\delta_i}{n-i+1} \left[\prod_{j=1}^{i-1} \left(1 - \frac{1}{n-j+1} \right)^{\delta_j} \right] \right\} I_{(X_i \leq x, Z_i \leq t)} \quad (4)$$

```

km_x = function(t, x, data){
  n = dim(data)[1]
  temp = 0
  for(i in 1:n){
    if(data$time[i] < t & data$x[i] <= x){
      # print(i)
      w = 1
      if(i == 1){
        w = w * data$status[i]/(n - i + 1)
        temp = temp + w
      }else{
        for(j in 1:(i - 1)){
          w = w * (1 - data$status[j]/(n - j + 1))
        }
        w = w * data$status[i]/(n - i + 1)
        temp = temp + w
      }
    }
  }
}

```

```

    }
  }
}
return(temp)
}

```

Dikta replaced δ_i with $m(t, x)$ and got

$$S^D = \sum_{i=1}^n \left\{ \frac{m(Z_i, X_i)}{n - i + 1} \left[\prod_{j=1}^{i-1} \left(1 - \frac{m(Z_i, X_i)}{n - j + 1} \right) \right] \right\} I_{(X_i \leq x, Z_i \leq t)} \quad (5)$$

```

dikta_x = function(t, x, data){
  n = dim(data)[1]
  temp = 0
  for(i in 1:n){
    if(data$time[i] < t & data$x[i] <= x){
      # print(i)
      w = 1
      if(i == 1){
        w = w * mm(data$x[i], data$time[i])/(n - i + 1)
        temp = temp + w
      }else{
        for(j in 1:(i - 1)){
          w = w * (1 - mm(data$x[j], data$time[j])/(n - j + 1))
        }
        w = w * mm(data$x[i], data$time[i])/(n - i + 1)
        temp = temp + w
      }
    }
  }
  return(temp)
}

# m function
mm = function(x,y){
  v = r*x + log(c) - log(a) + (c-a)*log(y)
  temp = 1/(1 + exp(v))
  return(temp)
}

```

Results

Kaplan-Meier-biased			Dikta semiparametric		
Bias	SD	MSE	Bias	SD	MSE

$t = 0.1$	0.0487	0.0148	0.0026	-0.0065	0.0080	0.0001
$t = 0.4$	0.0916	0.0236	0.0089	0.0139	0.0155	0.0004
$t = 1.3$	0.1756	0.0310	0.0318	0.1174	0.0224	0.0143
