贝叶斯生存分析

蔡苗 *miao.cai@slu.edu*

2018-11-06

1 指数回归

2 韦伯回归

2.1 简单韦伯回归 - 老鼠生存的例子

```
N_uncensored <- 65L
N_censored <- 15L
M <- 4
group_uncensored <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4)
group_censored <- c(1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4)
t_uncensored <- c(12, 17, 21, 25, 11, 26, 27, 30, 13, 12, 21, 20, 23, 25, 23,
29, 35, 31, 36, 32, 27, 23, 12, 18, 38, 29, 30, 32, 25, 30, 37,
27, 22, 26, 28, 19, 15, 12, 35, 35, 10, 22, 18, 12, 31, 24, 37,
29, 27, 18, 22, 13, 18, 29, 28, 16, 22, 26, 19, 17, 28, 26, 12,
17, 26)
censor_time <- c(40, 40, 40, 40, 40, 40, 40, 40, 10, 24, 40, 40, 20, 29, 10)
```

```
library(rstan)
library(shinystan)

mice_weibull_stan = '
data {
  int<lower=0> N_uncensored;
  int<lower=0> N_censored;
  int<lower=0> M;
  int<lower=1,upper=M> group_uncensored[N_uncensored];
  int<lower=1,upper=M> group_censored[N_censored];
  real<lower=0> censor_time[N_censored];
```

```
  real<lower=0> t_uncensored[N_uncensored];
}

parameters {
  real<lower=0> r;
  real beta[M];
  real<lower=1> t2_censored[N_censored];
}

model {
  r ~ exponential(0.001);
  beta ~ normal(0, 100);
  for (n in 1:N_uncensored) {
    t_uncensored[n] ~ weibull(r, exp(-beta[group_uncensored[n]] / r));
  }
  for (n in 1:N_censored) {
    t2_censored[n] ~ weibull(r, exp(-beta[group_censored[n]] / r) / censor_time[n]);
  }
}

generated quantities {
  real median[M];
  real pos_control;
  real test_sub;
  real veh_control;

  for (m in 1:M)
    median[m] = pow(log(2) * exp(-beta[m]), 1/r);

  veh_control = beta[2] - beta[1];
  test_sub    = beta[3] - beta[1];
  pos_control = beta[4] - beta[1];
}
'

datmice_weibull = list(N_uncensored,
               N_censored,
               M,
               group_uncensored,
```

```
                group_censored,
                censor_time,
                t_uncensored)

miceWeibullfit <-
  stan(
  model_code = mice_weibull_stan,
  model_name = "mice_weibull_stan",
  data = datmice_weibull,
  iter = 3000,
  warmup = 1000,
  chains = 1
  )

launch_shinystan(miceWeibullfit)
```

2.2 多水平韦伯回归

```
NP <-
38
N_uc <-
58
N_rc <-
18
t_uc <-
c(8, 23, 22, 447, 30, 24, 7, 511, 53, 15, 7, 141, 96, 536, 17,
185, 292, 15, 152, 402, 13, 39, 12, 132, 34, 2, 130, 27, 152,
190, 119, 63, 16, 28, 318, 12, 245, 9, 30, 196, 154, 333, 38,
177, 114, 562, 66, 40, 201, 156, 30, 25, 26, 58, 43, 30, 8, 78
)
t_rc <-
c(149, 22, 113, 5, 54, 6, 13, 8, 70, 25, 4, 159, 108, 24, 46,
5, 16, 8)
age_uc <-
c(28, 48, 32, 31, 10, 16, 51, 55, 69, 51, 44, 34, 35, 17, 60,
60, 43, 44, 46, 30, 62, 42, 43, 10, 52, 53, 54, 56, 57, 44, 22,
60, 28, 32, 32, 10, 17, 51, 56, 69, 52, 44, 35, 60, 44, 47, 63,
43, 58, 10, 52, 53, 54, 56, 51, 57, 22, 52)
```

```
age_rc <-
c(42, 53, 57, 50, 42, 52, 48, 34, 42, 17, 60, 53, 44, 30, 43,
45, 42, 60)
sex_uc <-
c(0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1,
1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0,
1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1)
sex_rc <-
c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0)
patient_uc <-
c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15,
16, 17, 18, 20, 21, 22, 23, 24, 25, 27, 28, 29, 30,
31, 33, 34, 35, 38, 1, 3, 4, 5, 6, 7, 8, 9, 10,
11, 13, 17, 18, 21, 23, 25, 26, 27, 28, 29, 30, 31,
32, 33, 35, 37)
patient_rc <-
c(14, 19, 26, 32, 36, 37, 2, 12, 14, 15, 16, 19,
20, 22, 24, 34, 36, 38)
disease_uc <-
c(1, 2, 1, 1, 1, 1, 2, 2, 3, 2, 3, 1, 3, 1, 3, 1, 1, 1, 4, 1,
3, 3, 3, 2, 3, 2, 2, 3, 4, 2, 1, 4, 1, 1, 1, 1, 1, 2, 2, 3, 2,
3, 3, 1, 1, 4, 3, 3, 3, 2, 3, 2, 2, 3, 3, 4, 1, 4)
disease_rc <-
c(3, 2, 3, 3, 1, 4, 2, 1, 3, 1, 3, 2, 1, 1, 3, 2, 1, 4)
```

```
kidney_weibull_stan = '
data {
  int<lower=0> NP;
  int<lower=0> N_uc;
  int<lower=0> N_rc;
  real<lower=0> t_uc[N_uc];
  real<lower=0> t_rc[N_rc];
  int disease_uc[N_uc];
  int disease_rc[N_rc];
  int patient_uc[N_uc];
  int patient_rc[N_rc];
  int sex_uc[N_uc];
  int sex_rc[N_rc];
  int age_uc[N_uc];
  int age_rc[N_rc];
```

```
}
parameters {
  real alpha;
  real beta_age;
  real beta_sex;
  real beta_disease2;
  real beta_disease3;
  real beta_disease4;
  real<lower=0> r;
  real<lower=0> tau;
  real b[NP];
}

transformed parameters {
  real sigma;
  real yabeta_disease[4];
  yabeta_disease[1] <- 0;
  yabeta_disease[2] <- beta_disease2;
  yabeta_disease[3] <- beta_disease3;
  yabeta_disease[4] <- beta_disease4;
  sigma <- sqrt(1 / tau);
}

model {
  alpha ~ normal(0, 100);
  beta_age ~ normal(0, 100);
  beta_sex ~ normal(0, 100);
  beta_disease2 ~ normal(0, 100);
  beta_disease3 ~ normal(0, 100);
  beta_disease4 ~ normal(0, 100);

  tau ~ gamma(1.0E-3, 1.0E-3);
  r ~ gamma(1, 1.0E-3);

  for (i in 1:NP) b[i] ~ normal(0, sigma);
  for (i in 1:N_uc) {
    t_uc[i] ~ weibull(r, exp(-(alpha + beta_age * age_uc[i] + beta_sex * sex_uc[i] +
                               yabeta_disease[disease_uc[i]] + b[patient_uc[i]]) / r));
  }
```

```
  for (i in 1:N_rc) {
    1 ~ bernoulli(exp(-pow(t_rc[i] / exp(-(alpha + beta_age * age_rc[i] + beta_sex * sex_rc[i] + y
    //TODO: try the weibull_cdf
    // 0 ~ bernoulli(weibull_cdf(t_rc[i], exp(-(alpha + beta_age * age_rc[i] + beta_sex * sex_rc[i
    // + yabeta_disease[disease_rc[i]] + b[patient_rc[i]]) / r), r));
  }
}

generated quantities {
}
'

datkidney_weibull = list(N_uncensored,
                N_censored,
                M,
                group_uncensored,
                group_censored,
                censor_time,
                t_uncensored)

kidneyWeibullfit <-
  stan(
  model_code = kidney_weibull_stan,
  model_name = "kidney_weibull_stan",
  data = datkidney_weibull,
  iter = 3000,
  warmup = 1000,
  chains = 1
  )
```

3 Cox回归