

Bayesian Heart Failure Survival Analysis with Parametric Cox Proportional Hazard Regression

Hanning Su
Student ID: hs3375

May 21, 2023

1 Introduction

In this project, we conduct bayesian analysis with Weibull baseline Cox proportional hazard model.

1.1 Model derivation

Suppose for $i \in 1, \dots, n$, \mathbf{Z}_i is covariate vector for subject i, and $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{ip})^T$, we let

$$T_i | \mathbf{Z}_i \stackrel{iid}{\sim} \text{Weibull}(\alpha, \lambda_i)$$

with hazard:

$$h(t | \mathbf{Z}_i) = \lambda_i \alpha t^{\alpha-1}$$

, where λ_i depends on \mathbf{Z}_i .

Let $\mu_i = -\sigma \log \lambda_i$ and $\sigma = 1/\alpha$, then

$$Y_i = \log T_i = \mu_i + \sigma W_i$$

, where $\mu_i := \tilde{\mu} + \gamma^T \mathbf{Z}_i$ and $W \stackrel{iid}{\sim}$ standard Gumbel(minimum) with pdf $f_{W_i}(w) = e^{w-e^w}$. Thus, we have

$$Y_i = \tilde{\mu} + \gamma^T \mathbf{Z}_i + \sigma W_i$$

, where $\gamma = (\gamma_1, \dots, \gamma_p)^T$.

This leads to a proportional hazard model for T_i with a Weibull baseline hazard (Klein and Moeschberger, 2003):

$$h(t | \mathbf{Z}_i) = \alpha t^{\alpha-1} \tilde{\lambda} \exp(\beta^T \mathbf{Z}_i)$$

, where $\alpha = 1/\sigma$, $\tilde{\lambda} = \exp(-\tilde{\mu}/\sigma)$, $\beta = (\beta_1, \dots, \beta_p)^T$, and $\beta_j = -\gamma_j/\sigma$.

If we let $\beta_0 = -\tilde{\mu}/\sigma$, we can write (corresponding to Brilleman et al. 2020):

$$h(t | \mathbf{Z}_i) = \alpha t^{\alpha-1} \exp(\beta_0 + \beta_1 Z_{i1} + \beta_2 Z_{i2} + \dots + \beta_p Z_{ip}) \quad (*)$$

We use (*) to conduct our Bayesian analysis.

Since the dataset we adopted has right censoring, in order to use Stan and analyze survival curves later, we need to derive the explicit form of CDF and pdf and other characterizing functions of T_i corresponding to (*):

Cumulative hazard:

$$H(t|\mathbf{Z}_i) = \int_0^t \alpha r^{\alpha-1} \exp(\beta_0 + \beta^T \mathbf{Z}_i) dr = t^\alpha \exp(\beta_0 + \beta^T \mathbf{Z}_i)$$

Survival function:

$$S(t|\mathbf{Z}_i) = \exp(-H(t|\mathbf{Z}_i)) = \exp(-t^\alpha \exp(\beta_0 + \beta^T \mathbf{Z}_i))$$

CDF:

$$F(t|\mathbf{Z}_i) = 1 - \exp(-t^\alpha \exp(\beta_0 + \beta^T \mathbf{Z}_i))$$

pdf:

$$f(t|\mathbf{Z}_i) = \alpha t^{\alpha-1} \exp(\beta_0 + \beta^T \mathbf{Z}_i) \exp(-t^\alpha \exp(\beta_0 + \beta^T \mathbf{Z}_i))$$

For diffuse priors, we use the options mentioned by Brillemann et al. (2020):

$$\beta_1, \beta_2, \dots, \beta_p \sim N(0, 20)$$

$$\beta_0 \sim \text{Cauchy}(0, 20)$$

$$\alpha \sim \text{Half-normal}(0, 10)$$

For concentrated priors, we will run a frequentist Cox regression and design based on the result. We will elaborate on this in later section.

1.2 Research questions

1. Finding out the significant factors that significantly impact the hazard of mortality of a heart failure patient.
2. We want to draw posterior (mean overlayed on top on all the posterior possible) survival curves for "typical" patients (discussed in detail later) at different levels of ejection fraction(EF) (i.e. $EF \leq 30$, $30 < EF \leq 45$ and $EF > 45$) based on the posterior samples of $\beta_0, \beta_1, \dots, \beta_p$ and α .
3. We have two pairs of typical patients, i.e. typical male and typical female, typical smoker and typical non-smoker. Based on posterior predicted survival times T , We want to find out the probabilities that a typical male outlives a typical female and that a typical smoking patient outlives a typical non-smoking patient. For further comparison, we also report the predicted mean survival times for all the typical individuals.

We emphasize EF in our analysis. Here is some further information about this variable by Ahmed et al. (2017): "Ejection fraction is an important measurement of how well one's heart is pumping and is used to help classify heart failure and guide treatment. The EF is also found to be significant correlate of deaths among heart failure patients from Cox regression for present sample."

1.3 Dataset introduction

I direct quote the paper of Ahmed et al. (2017) "Current study is based on 299 patients of heart failure comprising of 105 women and 194 men. All the patients were more than 40 years old, having left ventricular systolic dysfunction and falling in NYHA class III and IV. Follow up time was 4–285 days with an average of 130 days. Disease was diagnosed by cardiac echo report or notes written by physician. Age, serum sodium, serum creatinine, gender, smoking, Blood Pressure (BP), Ejection Fraction (EF), anemia, platelets, Creatinine Phosphokinase (CPK) and diabetes were considered as potential variables explaining mortality caused by CHD. Age, serum sodium and CPK are continuous variables whereas EF, serum creatinine and platelets were taken as categorical variables. EF was divided into three levels (i.e. $EF \leq 30$, $30 < EF \leq 45$ and $EF > 45$) and platelets was also divided into three level on the basis of quartiles. Serum creatinine greater than its normal level (1.5) is an indicator of renal dysfunction. Its effect on mortality was studied as creatinine > 1.5 vs ≤ 1.5 . Anemia in patients was assessed by their haematocrit level. Following McClellan et al. [4] the patients with haematocrit less than 36 (minimum normal level of haematocrit) were taken as anemic. The information related to risk factors were taken from blood reports while smoking status and blood pressure were taken from physician's notes."

2 Model fitting with diffuse prior and analysis

```
Heart <- read.csv("Heart_cox.csv") [,-c(1,12)]  
  
N_censored <- sum(Heart$Event == 0)  
N_uncensored <- sum(Heart$Event == 1)  
  
Times_censored <- Heart[Heart$Event == 0, ]$TIME  
Times_uncensored <- Heart[Heart$Event == 1, ]$TIME  
  
Z_censored <- as.matrix(Heart[Heart$Event == 0, ] [,-c(1,2)])  
Z_uncensored <- as.matrix(Heart[Heart$Event == 1, ] [,-c(1,2)])  
  
N_predictors <- dim(Z_censored)[2]  
  
scale_alpha <- 10  
scale_beta <- 20
```

We define

$$\mathbf{x} = (\text{Gender}, \text{Smoking}, \text{Diabetes}, \text{BP}, \text{Anaemia}, \text{Age}, \text{Sodium}, \text{Creatinine}, \log_{-}\text{CPK}, \\ \text{PleteletsleqQ1}, \text{PleteletsgeqQ3})^T$$

In the following section, we want to find some "typical" patients. To be specific, we want to find 4 sets (indexed by 1, 2, 3, and 4) of covaraites representing typical male, female,

smoking, and non-smoking patients respectively.

For a continuous variable, by typical, we mean it takes an average value. For categorical variable, by typical, we mean it takes on the label of the majority class.

```
Heart_factor <- Heart
Heart_factor$Gender <- as.factor(Heart$Gender)
Heart_factor$Smoking <- as.factor(Heart$Smoking)
Heart_factor$Diabetes <- as.factor(Heart$Diabetes)
Heart_factor$BP <- as.factor(Heart$BP)
Heart_factor$Anaemia <- as.factor(Heart$Anaemia)
Heart_factor$PleteletsleqQ1 <- as.factor(Heart$PleteletsleqQ1)
Heart_factor$PleteletsgeqQ3 <- as.factor(Heart$PleteletsgeqQ3)

# Gender = 1 corresponds to being male (#male = 194)
summary(Heart_factor[Heart$Gender == 1,] [,-c(1,2,3,11,12)])
```

	Smoking	Diabetes	BP	Anaemia	Age	Sodium
##	0:102	0:124	0:133	0:117	Min. :40.00	Min. :113.0
##	1: 92	1: 70	1: 61	1: 77	1st Qu.:52.00	1st Qu.:134.0
##					Median :60.00	Median :137.0
##					Mean :61.41	Mean :136.5
##					3rd Qu.:70.00	3rd Qu.:139.0
##					Max. :95.00	Max. :148.0
##	Creatinine		log_CPK		PleteletsleqQ1	PleteletsgeqQ3
##	Min. :0.600		Min. :3.135		0:137	0:150
##	1st Qu.:1.000		1st Qu.:4.751		1: 57	1: 44
##	Median :1.100		Median :5.517			
##	Mean :1.399		Mean :5.686			
##	3rd Qu.:1.400		3rd Qu.:6.366			
##	Max. :9.400		Max. :8.970			

```
# Gender = 0 corresponds to being female (#male = 105)
summary(Heart_factor[Heart$Gender == 0,] [,-c(1,2,3,11,12)])
```

	Smoking	Diabetes	BP	Anaemia	Age	Sodium
##	0:101	0:50	0:61	0:53	Min. :40.00	Min. :116.0
##	1: 4	1:55	1:44	1:52	1st Qu.:50.00	1st Qu.:135.0
##					Median :60.00	Median :137.0
##					Mean :59.78	Mean :136.8
##					3rd Qu.:66.00	3rd Qu.:140.0
##					Max. :95.00	Max. :146.0
##	Creatinine		log_CPK		PleteletsleqQ1	PleteletsgeqQ3
##	Min. :0.500		Min. :3.951		0:87	0:74
##	1st Qu.:0.900		1st Qu.:4.820		1:18	1:31
##	Median :1.000		Median :5.521			

```

##  Mean    :1.384   Mean    :5.612
##  3rd Qu.:1.400   3rd Qu.:6.366
##  Max.    :9.000   Max.    :8.285

# Gender = 1 corresponds to smoker (#Smoking = 96)
summary(Heart_factor[Heart$Smoking == 1,] [,-c(1,2,4,11,12)])
```

	Gender	Diabetes	BP	Anaemia	Age	Sodium	Creatinine
##	0: 4	0:66	0:66	0:62	Min. :40.00	Min. :113.0	Min. :0.600
##	1:92	1:30	1:30	1:34	1st Qu.:52.00	1st Qu.:135.0	1st Qu.:0.900
##					Median :60.00	Median :137.0	Median :1.100
##					Mean :61.16	Mean :136.7	Mean :1.353
##					3rd Qu.:70.00	3rd Qu.:139.0	3rd Qu.:1.300
##					Max. :90.00	Max. :145.0	Max. :9.400
##	log_CPK	PleteletsleqQ1	PleteletsgeqQ3				
##	Min. :3.850	0:69	0:69				
##	1st Qu.:4.796	1:27	1:27				
##	Median :5.403						
##	Mean :5.619						
##	3rd Qu.:6.366						
##	Max. :8.680						


```
# Gender = 0 corresponds to being female (#non-smoking = 203)
summary(Heart_factor[Heart$Smoking == 0,] [,-c(1,2,4,11,12)])
```

	Gender	Diabetes	BP	Anaemia	Age	Sodium
##	0:101	0:108	0:128	0:108	Min. :40.00	Min. :116.0
##	1:102	1: 95	1: 75	1: 95	1st Qu.:50.00	1st Qu.:134.0
##					Median :60.00	Median :137.0
##					Mean :60.68	Mean :136.6
##					3rd Qu.:68.50	3rd Qu.:140.0
##					Max. :95.00	Max. :148.0
##	Creatinine	log_CPK	PleteletsleqQ1	PleteletsgeqQ3		
##	Min. :0.500	Min. :3.135	0:155	0:155		
##	1st Qu.:0.900	1st Qu.:4.745	1: 48	1: 48		
##	Median :1.100	Median :5.638				
##	Mean :1.413	Mean :5.679				
##	3rd Qu.:1.500	3rd Qu.:6.366				
##	Max. :9.000	Max. :8.970				

Our "typical" patients are represented below:

$$\mathbf{x}_{\text{typ1}} = (1, 0, 0, 0, 0, 61.41, 136.5, 1.399, 5.686, 0, 0)^T$$

$$\mathbf{x}_{\text{typ2}} = (0, 0, 1, 0, 0, 59.78, 136.8, 1.384, 5.612, 0, 0)^T$$

$$\mathbf{x}_{\text{typ3}} = (1, 1, 0, 0, 0, 61.16, 136.7, 1.353, 5.619, 0, 0)^T$$

$$\mathbf{x}_{typ4} = (1, 0, 0, 0, 0, 60.08, 136.6, 1.413, 5.679, 0, 0)^T$$

```
Z_typical <- matrix(c(1,0,0,0,0,61.41, 136.5, 1.399,0,0,5.686,0,0,
                      1,0,0,0,0,61.41, 136.5, 1.399,1,0,5.686,0,0,
                      1,0,0,0,0,61.41, 136.5, 1.399,0,1,5.686,0,0,
                      0,0,1,0,0,59.78, 136.8, 1.384,0,0,5.612,0,0,
                      0,0,1,0,0,59.78, 136.8, 1.384,1,0,5.612,0,0,
                      0,0,1,0,0,59.78, 136.8, 1.384,0,1,5.612,0,0,
                      1,1,0,0,0,61.16, 136.7, 1.353,0,0,5.619,0,0,
                      1,1,0,0,0,61.16, 136.7, 1.353,1,0,5.619,0,0,
                      1,1,0,0,0,61.16, 136.7, 1.353,0,1,5.619,0,0,
                      1,0,0,0,0,60.68, 136.6, 1.413,0,0,5.679,0,0,
                      1,0,0,0,0,60.68, 136.6, 1.413,1,0,5.679,0,0,
                      1,0,0,0,0,60.68, 136.6, 1.413,0,1,5.679,0,0),
                     byrow = TRUE, nrow = 12, ncol= 13)
```

```
colnames(Z_typical) <- colnames(Z_censored)
```

```
stan_code <-
"functions{
  real myweibull_lpdf(real x, real eta, real alpha, real beta_0) {
    return log(alpha) + (alpha-1)*log(x) + (beta_0 + eta) -
      (x^alpha) * exp(beta_0 + eta);
  }

  real myweibull_lccdf(real x, real eta, real alpha, real beta_0){
    return -(x^alpha) * exp(eta + beta_0);
  }
}

data {
  int<lower=1> N_censored; // Number of censored individuals
  int<lower=1> N_uncensored; // Number of uncensored individuals
  int<lower=1> N_typical; // Number of typical individuals
  int<lower=0> N_predictors;
  vector<lower=0>[N_censored] Observed_times_censored;
  vector<lower=0>[N_uncensored] Observed_times_uncensored;
  matrix[N_censored, N_predictors] Z_censored;
  matrix[N_uncensored, N_predictors] Z_uncensored;
  matrix[N_typical, N_predictors] Z_typical;
  real scale_alpha; //prior sd on alpha
  real scale_beta; // prior sds on beta
}
```

```

parameters {
    real<lower=0> alpha; // =1/sigma sigma is the scale paramter of log weibull
    vector[N_predictors] beta; // regression coefficients beta vector
    real beta_0; //intercept
}

transformed parameters {
    vector[N_predictors] exp_beta;
    vector[N_predictors] gamma;
    real mu_tilda;
    real sigma;

    exp_beta = exp(beta);
    gamma = -beta/alpha;
    mu_tilda = -beta_0 / alpha;
    sigma = 1/alpha;
}

model {
    real eta;
    row_vector[N_predictors] Z;
    real x;

    // priors
    alpha ~ normal(0, scale_alpha); // half normal
    beta_0 ~ cauchy(0, scale_beta);
    beta ~ normal(0, scale_beta);

    for (i in 1:N_censored){
        x = Observed_times_censored[i];
        Z = Z_censored[i];
        eta = Z * beta;
        target += myweibull_lccdf(x|eta, alpha, beta_0);
    }

    for (i in 1:N_uncensored){
        x = Observed_times_uncensored[i];
        Z = Z_uncensored[i];
        eta = Z * beta;
        target += myweibull_lpdf(x|eta, alpha, beta_0);
    }
}

generated quantities{
    vector[N_typical] times_typical_sampled;
}

```

```

for(i in 1:N_typical) {
  times_typical_sampled[i] = exp(mu_tilda + Z_typical[i]*gamma -
  sigma *gumbel_rng(0,1));
}
}"

```

```

library(rstan)

stan_code <-
[2123 chars quoted with ''']

write(stan_code, "Project.stan")

model = stan_model("Project.stan")

```

```

scale_alpha <- 10
scale_beta <- 20

model_data = list(
  N_censored = N_censored,
  N_uncensored = N_uncensored,
  N_predictors = N_predictors,
  N_typical = 12,
  Observed_times_censored = Times_censored,
  Observed_times_uncensored = Times_uncensored,
  Z_censored = Z_censored,
  Z_uncensored = Z_uncensored,
  Z_typical = Z_typical,
  scale_alpha = 10,
  scale_beta = 20
)

iter= 10000
mod_fit = sampling(model, data = model_data, warmup = floor(iter/2),
                    iter = iter, thin = 5, chains=5)

summary_mod_fit <- summary(mod_fit)$summary

params <- extract(mod_fit)

save(params, file = "Stan_pos_sample.RData")

save(summary_mod_fit, file = "Stan_pos_summary.RData")

```

We check convergence of parameters, with traceplot.

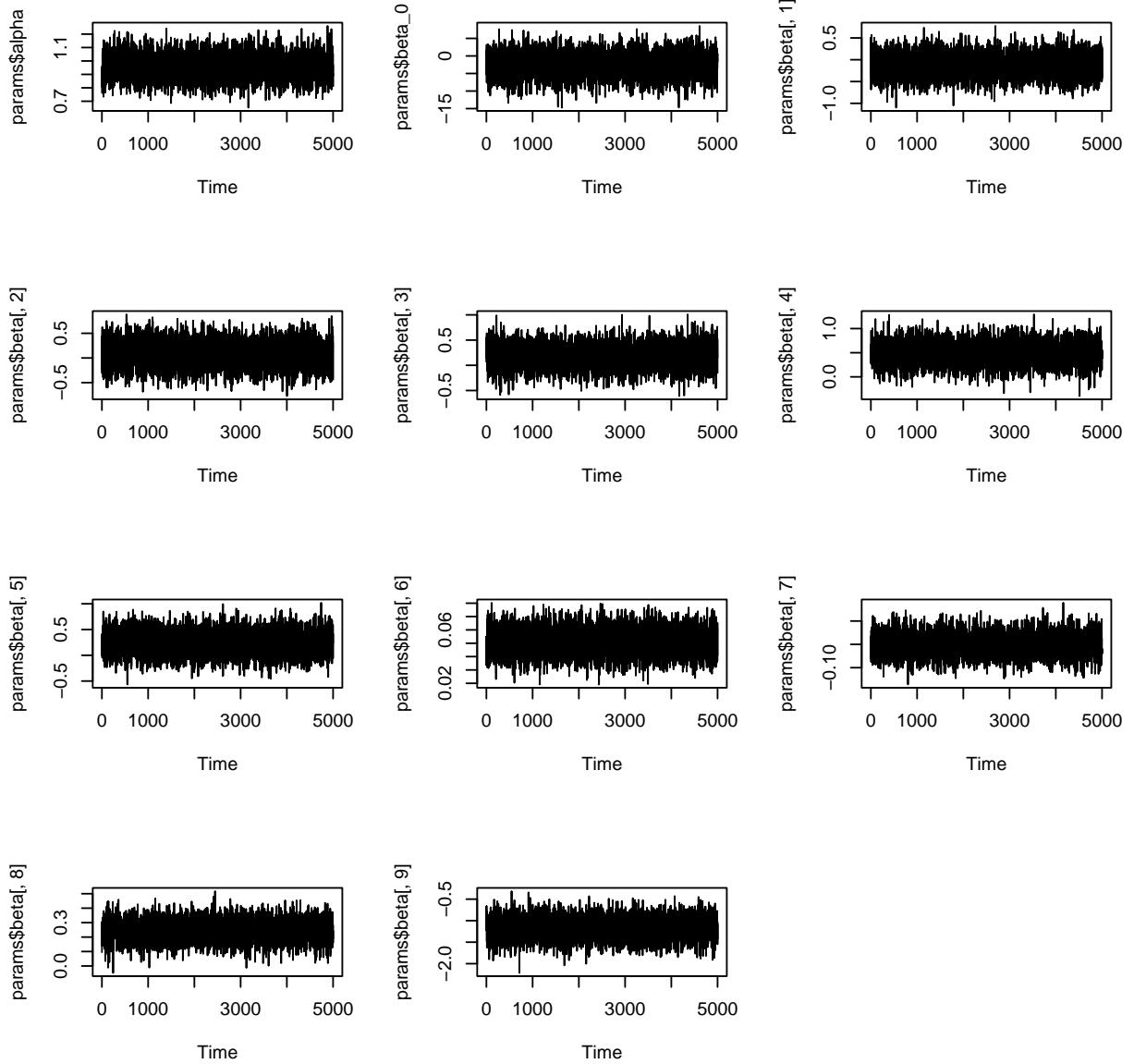
```
load("Stan_pos_sample.RData")

par(mfrow = c(4, 3))
# trace plot for alpha
ts.plot(params$alpha)

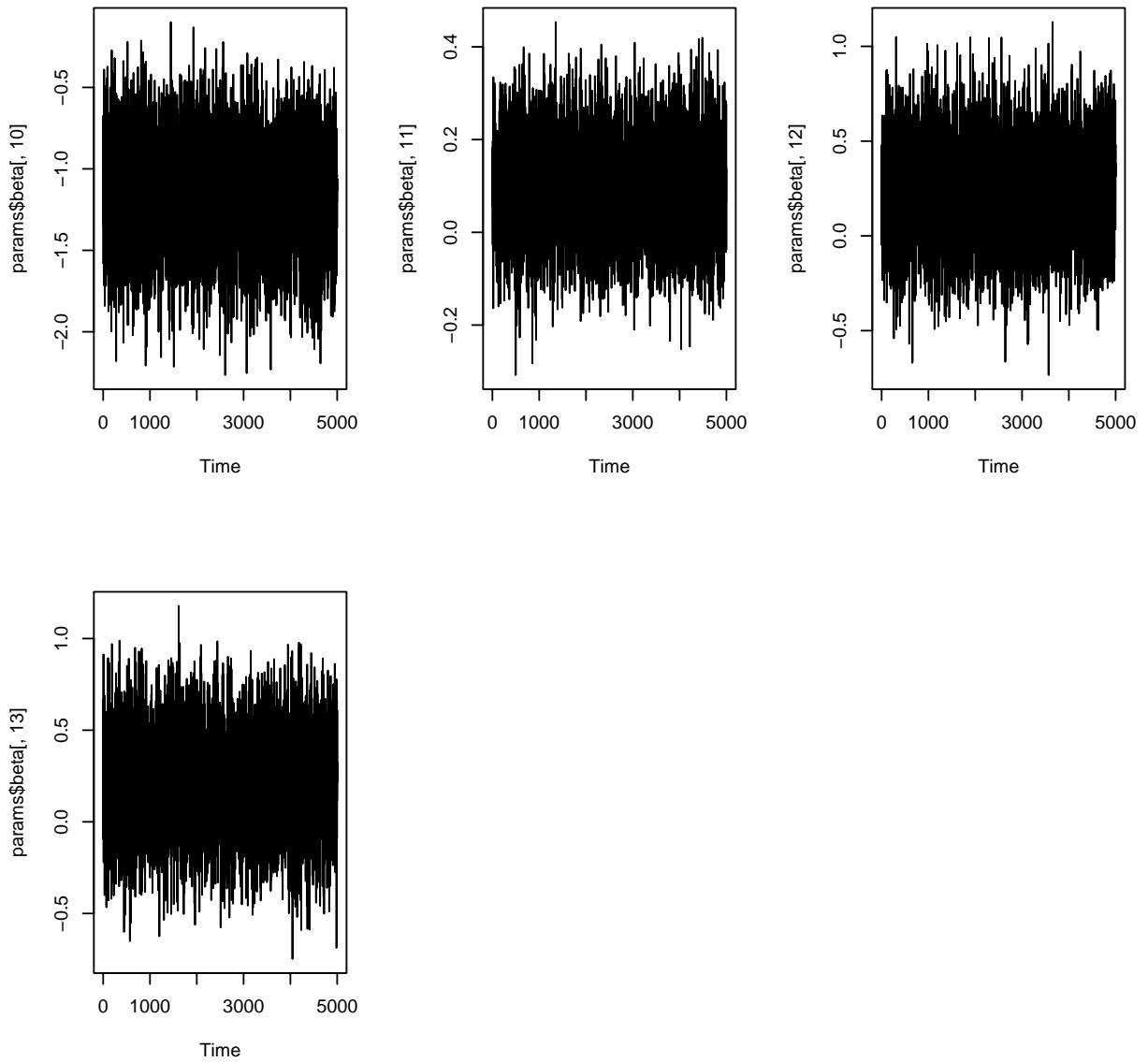
# trace plot for beta_0
ts.plot(params$beta_0)

# trace plot for betas for covariates
ts.plot(params$beta[,1])
ts.plot(params$beta[,2])
ts.plot(params$beta[,3])
ts.plot(params$beta[,4])
ts.plot(params$beta[,5])
ts.plot(params$beta[,6])
ts.plot(params$beta[,7])
ts.plot(params$beta[,8])
ts.plot(params$beta[,9])

par(mfrow = c(2, 3))
```



```
ts.plot(params$beta[, 10])
ts.plot(params$beta[, 11])
ts.plot(params$beta[, 12])
ts.plot(params$beta[, 13])
```



All parameters that we assume priors seems to have convergent posterior samples.

```

load("Stan_pos_summary.RData")
round(summary_mod_fit[1:15, -c(2, 3, 5, 7)], 4)

##          mean    2.5%    50%   97.5%    n_eff    Rhat
## alpha     0.9526  0.7908  0.9500  1.1253 4959.784 0.9999
## beta[1]   -0.1325 -0.6200 -0.1291  0.3520 5116.908 0.9998
## beta[2]   0.0801 -0.4051  0.0796  0.5632 4937.099 1.0001
## beta[3]   0.1774 -0.2526  0.1808  0.6091 5044.885 1.0001
## beta[4]   0.4923  0.0516  0.4925  0.9251 4836.286 0.9996
## beta[5]   0.2490 -0.1815  0.2513  0.6747 5140.638 1.0002

```

```

## beta[6]  0.0511  0.0327  0.0512  0.0696 4980.526 0.9992
## beta[7] -0.0525 -0.0944 -0.0530 -0.0060 4747.204 0.9998
## beta[8]  0.2483  0.1083  0.2505  0.3789 4610.782 0.9997
## beta[9]  -1.1644 -1.6498 -1.1553 -0.7198 4678.269 0.9998
## beta[10] -1.1465 -1.7881 -1.1324 -0.5493 5056.654 0.9998
## beta[11]  0.0971 -0.1089  0.0971  0.2966 4886.874 0.9997
## beta[12]  0.2271 -0.2682  0.2266  0.7268 4866.904 0.9999
## beta[13]  0.2127 -0.3185  0.2149  0.7330 5076.096 0.9996
## beta_0   -2.5302 -8.9018 -2.4512  3.6287 4730.711 0.9997

```

\hat{R} for all parameters are close to 1, and effective sample sizes are very high.

```

quantile(params$beta[,1], c(0.025, 0.975)) #Gender

##      2.5%      97.5%
## -0.6200159  0.3520263

quantile(exp(params$beta[,1]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.5379359 1.4219459

quantile(params$beta[,2], c(0.025, 0.975)) #Smoking

##      2.5%      97.5%
## -0.4050532  0.5632015

quantile(exp(params$beta[,2]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.6669413 1.7562863

quantile(params$beta[,3], c(0.025, 0.975)) #Diabetes

##      2.5%      97.5%
## -0.2526089  0.6090605

quantile(exp(params$beta[,3]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.7767717 1.8387032

quantile(params$beta[,4], c(0.025, 0.975)) #BP

##      2.5%      97.5%
## 0.0515987 0.9250559

```

```

quantile(exp(params$beta[,4]), c(0.025, 0.975))

##      2.5%    97.5%
## 1.052953 2.522009

quantile(params$beta[,5], c(0.025, 0.975)) #Anaemia

##      2.5%    97.5%
## -0.1814727 0.6747432

quantile(exp(params$beta[,5]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.834041 1.963529

quantile(params$beta[,6], c(0.025, 0.975)) #Age

##      2.5%    97.5%
## 0.03267963 0.06959313

quantile(exp(params$beta[,6]), c(0.025, 0.975))

##      2.5%    97.5%
## 1.033219 1.072072

quantile(params$beta[,7], c(0.025, 0.975)) #Sodium

##      2.5%    97.5%
## -0.094370555 -0.006043202

quantile(exp(params$beta[,7]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.9099455 0.9939750

quantile(params$beta[,8], c(0.025, 0.975)) #Creatinine

##      2.5%    97.5%
## 0.1082525 0.3788930

quantile(exp(params$beta[,8]), c(0.025, 0.975))

##      2.5%    97.5%
## 1.114329 1.460667

quantile(params$beta[,9], c(0.025, 0.975)) #EF30to45

```

```

##      2.5%    97.5%
## -1.649800 -0.719754

quantile(exp(params$beta[,9]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.1920884 0.4868720

quantile(params$beta[,10], c(0.025, 0.975)) #EF $\geq$ 45

##      2.5%    97.5%
## -1.7881092 -0.5493113

quantile(exp(params$beta[,10]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.1672762 0.5773473

quantile(params$beta[,11], c(0.025, 0.975)) #log_CPK

##      2.5%    97.5%
## -0.1088695 0.2966409

quantile(exp(params$beta[,11]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.8968475 1.3453321

quantile(params$beta[,12], c(0.025, 0.975)) #Plelets $\leq$ Q1

##      2.5%    97.5%
## -0.2681925 0.7267952

quantile(exp(params$beta[,12]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.7647605 2.0684411

quantile(params$beta[,13], c(0.025, 0.975)) #Plelets $\geq$ Q3

##      2.5%    97.5%
## -0.3184857 0.7329521

quantile(exp(params$beta[,13]), c(0.025, 0.975))

##      2.5%    97.5%
## 0.7272495 2.0812156

```

From 95% posterior quantile, we conclude that blood pressure (BP), age, Serum sodium (sodium), Serum creatinine (creatinine), indicator of ejection fraction (EF) between 30 to 45 (EF30t45), and indicator of ejection fraction (EF) greater or equal 45 (EFgeq45) are significant factor that impact the survival experience of heart failure patients.

With 95% confidence, regarding heart failure patients, we can summarize the marginal effects of our covariates below. Having high blood pressure will increase mortality hazard by 1.053 to 2.522 times. Being 1 year older will increase mortality hazard by 1.033 to 1.072 times. One more unit of sodium in blood will decrease mortality hazard by 0.91 to 0.99 times. One more unit of creatinine in blood will increase mortality hazard by 1.11 to 1.46 times. Having EF between 30 and 45 (as poosed to lower than 30 will decrease mortality by 0.19 to 0.487 times. Having EF greater than 45 (as poosed to lower than 30 will decrease mortality by between 0.17 to 0.58 times.

```
typical1_leq30 <- c(1,0,0,0,0,61.41, 136.5, 1.399,0,0,5.686,0,0)
typical1_30to45 <- c(1,0,0,0,0,61.41, 136.5, 1.399,1,0,5.686,0,0)
typical1_geq45 <- c(1,0,0,0,0,61.41, 136.5, 1.399,0,1,5.686,0,0)
library(yarrr)

par(mfrow=c(1,2))
plot( range(Heart$TIME),range(c(0,1)),type="n",xlab="time",
      ylab="S(t|Z)", main = "Typical male")
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical1_leq30))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("steelblue1", trans.val = .9))
}
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical1_30to45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("orangered", trans.val = 0.8))
}
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical1_geq45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
```

```

    col=transparent("yellow3", trans.val = .97))
}

alpha_pos_mean <- mean(params$alpha)
beta0_pos_mean <- mean(params$beta_0)
beta_pos_mean <- apply(params$beta, 2, mean)
curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical1_leq30))),
  from = 0, to = 300, add = TRUE, lwd=2, col="blue4")

curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical1_30to45))),
  from = 0, to = 300, add = TRUE, lwd=2, col="red")

curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical1_geq45))),
  from = 0, to = 300, add = TRUE, lwd=2, col="gold1")

typical2_leq30 <- c(0,0,1,0,0,59.78,136.8,1.384,0,0,5.612,0,0)
typical2_30to45 <- c(0,0,1,0,0,59.78,136.8,1.384,1,0,5.612,0,0)
typical2_geq45 <- c(0,0,1,0,0,59.78,136.8,1.384,0,1,5.612,0,0)

plot( range(Heart$TIME),range(c(0,1)),type="n",xlab="time",
      ylab="S(t|Z)", main = "Typical female")
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical2_leq30))),
    from = 0, to = 300, add = TRUE, lwd=0.5,
    col=transparent("steelblue1", trans.val = .9))
}

for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical2_30to45))),
    from = 0, to = 300, add = TRUE, lwd=0.5,
    col=transparent("orangered", trans.val = 0.8))
}

for(i in 1:5000) {
  alpha <- params$alpha[i]

```

```

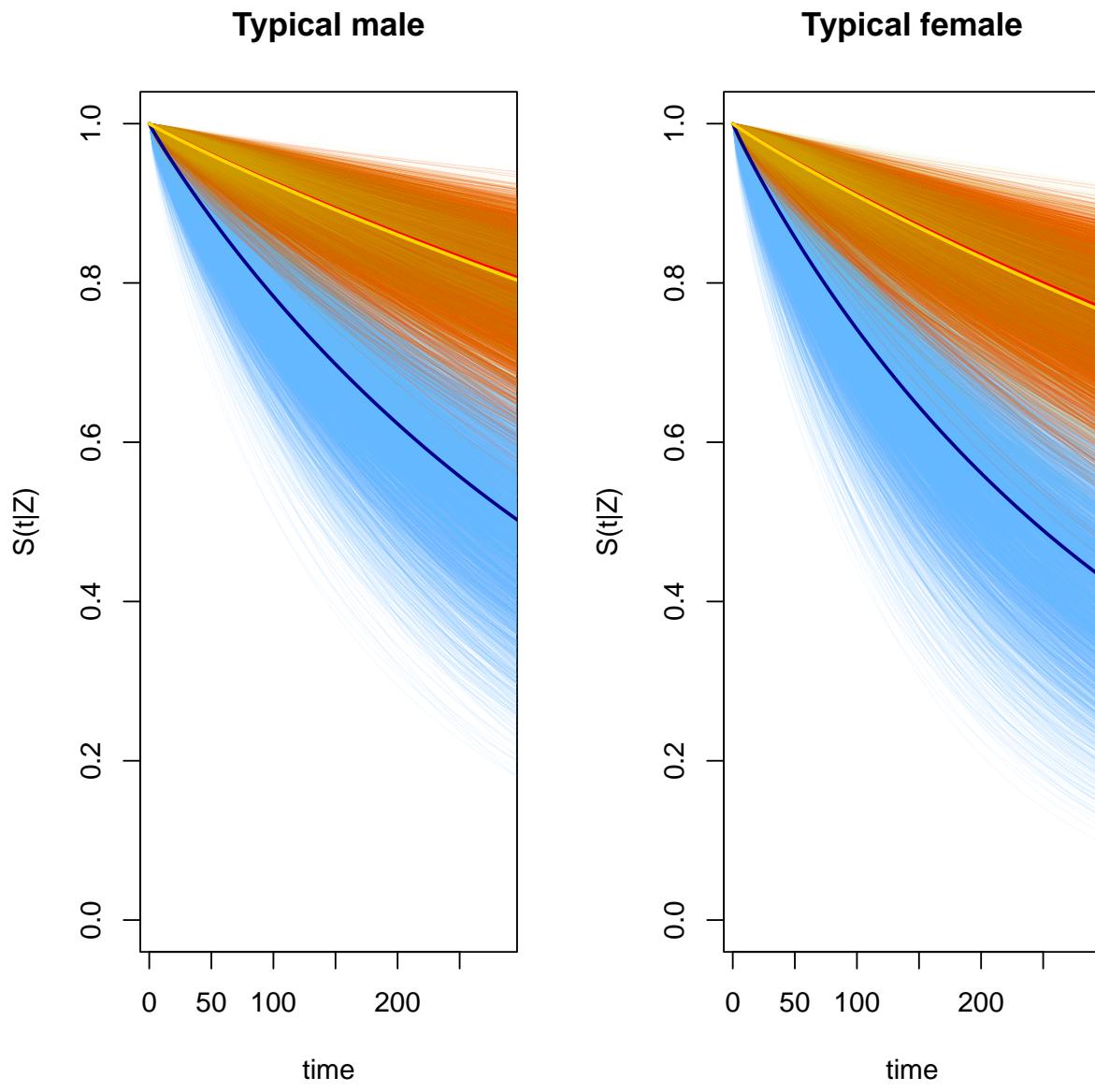
beta0 <- params$beta_0[i]
beta <- params$beta[i,]
curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical2_geq45))),
      from = 0, to = 300, add = TRUE, lwd=0.5,
      col=transparent("yellow3", trans.val = .97))
}

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical2_leq30))),
      from = 0, to = 300, add = TRUE, lwd=2, col="blue4")

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical2_30to45))),
      from = 0, to = 300, add = TRUE, lwd=2, col="red")

curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical2_geq45))),
      from = 0, to = 300,add = TRUE, lwd=2, col="gold1")

```



Note that in all plot, we use blue for $\text{EF} \leq 30$, red for $30 < \text{EF} \leq 45$ and yellow for $\text{EF} > 45$

For both typical male and female patients, EF lower than 30 has lowest survival probabilities. Other two levels of EF have nearly identical survival experiences.

Curiously, typical male patients have evidently higher survival probabilities at all levels of EF, suggesting female heart failure patients might experience more mortality risk.

```
typical3_leq30 <- c(1,1,0,0,0,61.16, 136.7, 1.353,0,0,5.619,0,0)
typical3_30to45 <- c(1,1,0,0,0,61.16, 136.7, 1.353,1,0,5.619,0,0)
typical3_geq45 <- c(1,1,0,0,0,61.16, 136.7, 1.353,0,1,5.619,0,0)
```

```

par(mfrow=c(1,2))
plot( range(Heart$TIME),range(c(0,1)),type="n",xlab="time",
      ylab="S(t|Z)", main = "Typical smoker")
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical3_leq30))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("steelblue1", trans.val = .9))
}

for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical3_30to45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("orangered", trans.val = 0.8))
}

for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical3_geq45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("yellow3", trans.val = .97))
}

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical3_leq30))),
  from = 0, to = 300, add = TRUE, lwd=2, col="blue4")

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical3_30to45))),
  from = 0, to = 300, add = TRUE, lwd=2, col="red")

curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
  as.numeric(beta_pos_mean%*%typical3_geq45))),
  from = 0, to = 300, add = TRUE, lwd=2, col="gold1")

typical4_leq30 <- c(1,0,0,0,0,60.68, 136.6, 1.413,0,0,5.679,0,0)
typical4_30to45 <- c(1,0,0,0,0,60.68, 136.6, 1.413,1,0,5.679,0,0)

```

```

typical4_geq45 <- c(1,0,0,0,0,60.68, 136.6, 1.413,0,1,5.679,0,0)

plot( range(Heart$TIME),range(c(0,1)),type="n",xlab="time",
      ylab="S(t|Z)", main="Typical non-smoker")
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[i,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical4_leq30))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("steelblue1", trans.val = .9))
}

for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[i,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical4_30to45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("orangered", trans.val = 0.8))
}

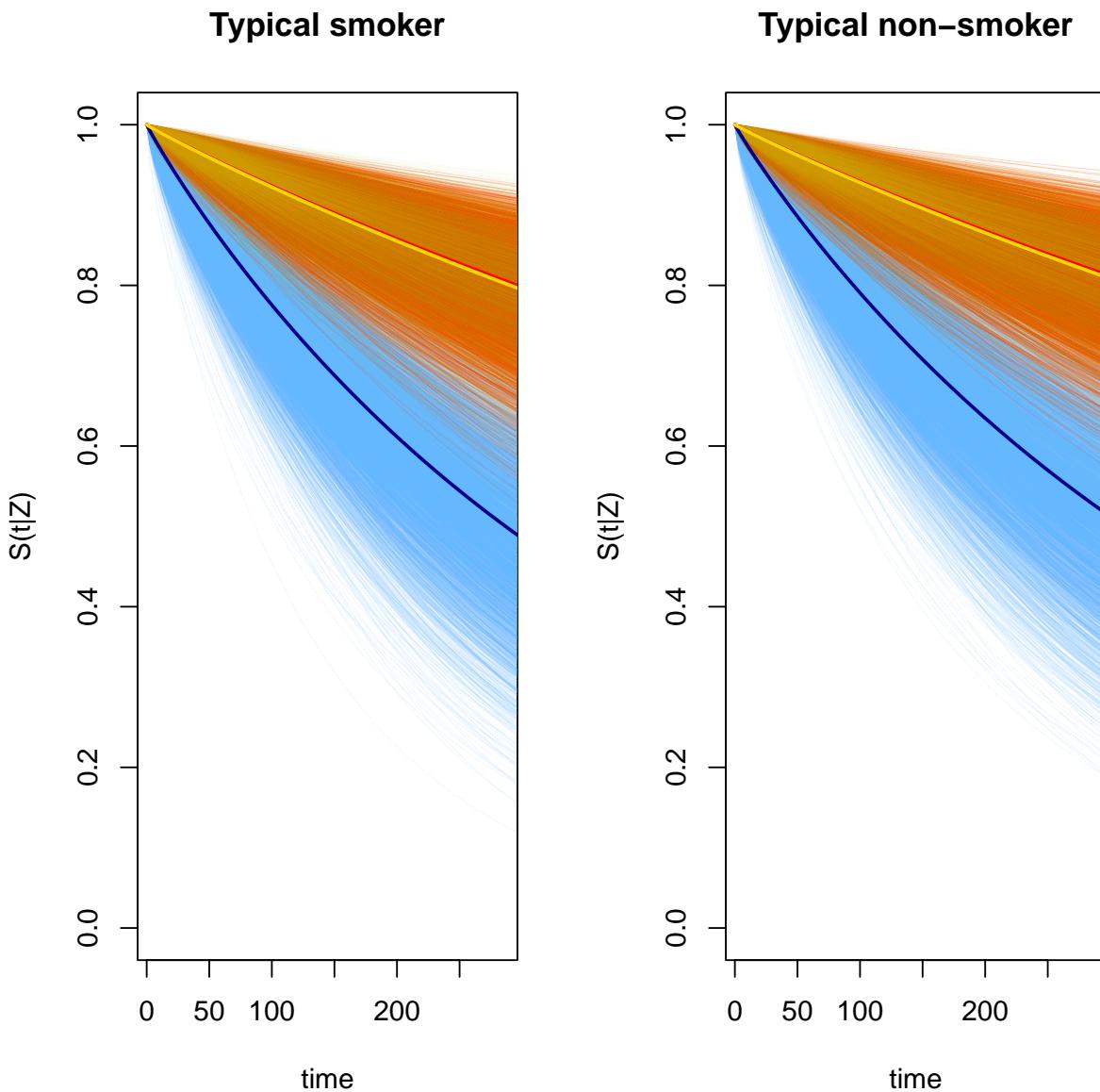
for(i in 1:5000) {
  alpha <- params$alpha[i]
  beta0 <- params$beta_0[i]
  beta <- params$beta[i,]
  curve(exp(-(x^alpha) * exp(beta0 + as.numeric(beta%*%typical4_geq45))),
        from = 0, to = 300, add = TRUE, lwd=0.5,
        col=transparent("yellow3", trans.val = .97))
}

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical4_leq30))),
      from = 0, to = 300, add = TRUE, lwd=2, col="blue4")

curve(exp(-(x^alpha_pos_mean ) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical4_30to45))),
      from = 0, to = 300, add = TRUE, lwd=2, col="red")

curve(exp(-(x^alpha_pos_mean) * exp(beta0_pos_mean +
      as.numeric(beta_pos_mean%*%typical4_geq45))),
      from = 0, to = 300,add = TRUE, lwd=2, col="gold1")

```



For both typical smoking and non-smoking patients, EF lower than 30 has lowest survival probabilities. Other two levels of EF have nearly identical survival experiences.

As expected, typical smoking patients have slightly (but still observable) lower survival probabilities at all levels of EF.

We calculate predicted probabilities that a typical male patient outlives a typical female patient for 3 categories of ejection fraction.

```
# EF less equal 30, typical male vs female
mean(params$times_typical_sampled[,1] > params$times_typical_sampled[,4])
## [1] 0.5448
```

```

# EF between 30 and 45, typical male vs female
mean(params$times_typical_sampled[,2] > params$times_typical_sampled[,5])

## [1] 0.5448

# EF greater equal 45, typical male vs female
mean(params$times_typical_sampled[,3] > params$times_typical_sampled[,6])

## [1] 0.5558

```

Typical male patient has more than 50% chance outlives typical female patient for all 3 categories of EF.

Then, we calculate predicted probabilities that a typical male patient outlives a typical female patient for 3 categories of ejection fraction.

```

# EF less equal 30, typical smoking vs non-smoking
mean(params$times_typical_sampled[,7] > params$times_typical_sampled[,10])

## [1] 0.4912

# EF between 30 and 45, smoking vs non-smoking
mean(params$times_typical_sampled[,8] > params$times_typical_sampled[,11])

## [1] 0.4876

# EF greater equal 45, smoking vs non-smoking
mean(params$times_typical_sampled[,9] > params$times_typical_sampled[,12])

## [1] 0.4862

```

A typical non-smoking patient has slightly over 50% chance to outlive a typical smoking patient for all 3 categories of EF.

At last, for further comparison, we also report the predicted survival times for all the typical individuals.

```

## EF less equal 30, typical male vs female
apply(params$times_typical_sampled[,c(1:12)], 2, mean)[1]

## [1] 473.2103

apply(params$times_typical_sampled[,c(1:12)], 2, mean)[4]

## [1] 390.7328

# EF between 30 and 45, typical male vs female
apply(params$times_typical_sampled[,c(1:12)], 2, mean)[2]

```

```

## [1] 1690.6

apply(params$times_typical_sampled[,c(1:12)], 2, mean) [5]

## [1] 1348.754

# EF greater equal 45, typical male vs female
apply(params$times_typical_sampled[,c(1:12)], 2, mean) [3]

## [1] 1705.656

apply(params$times_typical_sampled[,c(1:12)], 2, mean) [6]

## [1] 1400.342

```

```

# EF less equal 30, typical smoking vs non-smoking
apply(params$times_typical_sampled[,c(1:12)], 2, mean) [7]

## [1] 467.055

apply(params$times_typical_sampled[,c(1:12)], 2, mean) [10]

## [1] 492.2218

# EF between 30 and 45, smoking vs non-smoking
apply(params$times_typical_sampled[,c(1:12)], 2, mean) [8]

## [1] 1578.322

apply(params$times_typical_sampled[,c(1:12)], 2, mean) [11]

## [1] 1708.513

# EF greater equal 45, smoking vs non-smoking
apply(params$times_typical_sampled[,c(1:12)], 2, mean) [9]

## [1] 1666.361

apply(params$times_typical_sampled[,c(1:12)], 2, mean) [12]

## [1] 1738.563

```

3 Model fitting with concentrated prior and analysis

We want to rerun our analysis with more concentrated prior, we base our concentrated prior by running the semiparametric cox regression (no baseline hazard assumed). If the weibull baseline hazard assumption is correct (or at least reasonable), the frequentist estimates of less restrictive semiparametric Cox regression will serve as a meaningful source of prior information for the bayesian analysis based on a more restrictive parametric Cox model.

```
Heart_cox <- read.csv("Heart_cox.csv")
library(survival)
cox_fit <- coxph(Surv(TIME, Event)~Gender + Smoking + Diabetes + BP + Anaemia +
+ Age + Sodium + Creatinine + EF30to45 + EFgeq45 + log_CPK +
PleteletsleqQ1 + PleteletsgeqQ3, data=Heart_cox)

summary(cox_fit)$coef

##                                     coef   exp(coef)    se(coef)      z   Pr(>|z|)
## Gender          -0.14342735 0.8663837 0.249313436 -0.5752893 5.650957e-01
## Smoking         0.09851931 1.1035357 0.250074233  0.3939603 6.936104e-01
## Diabetes        0.17988203 1.1970761 0.222877153  0.8070905 4.196144e-01
## BP              0.47338108 1.6054131 0.218392664  2.1675686 3.019153e-02
## Anaemia         0.22399319 1.2510625 0.218976863  1.0229080 3.063513e-01
## Age             0.04904792 1.0502707 0.009448571  5.1910409 2.091217e-07
## Sodium          -0.05294300 0.9484341 0.023650380 -2.2385689 2.518398e-02
## Creatinine       0.25736873 1.2935220 0.070762023  3.6371025 2.757222e-04
## EF30to45        -1.10760470 0.3303493 0.238048634 -4.6528505 3.273777e-06
## EFgeq45         -1.08006645 0.3395730 0.319293120 -3.3826800 7.178222e-04
## log_CPK          0.09310614 1.0975782 0.101208827  0.9199409 3.576036e-01
## PleteletsleqQ1   0.23484315 1.2647104 0.256968703  0.9138979 3.607705e-01
## PleteletsgeqQ3   0.22846800 1.2566733 0.265348686  0.8610105 3.892323e-01

scale_beta <- summary(cox_fit)$coef[,3] * 10
mean_beta <- summary(cox_fit)$coef[,1]
```

We will choose to set the prior standard deviation of each coefficient to be 10 times the se(coef) (a lot more information than unit information prior but a lot less than full information prior based on all 299 subjects) reported by frequentist estimation. We keep using diffuse priors for auxiliary parameters, i.e. β_0 and α .

```
stan_code <-
"functions{
  real myweibull_lpdf(real x, real eta, real alpha, real beta_0) {
    return log(alpha) + (alpha-1)*log(x) + (beta_0 + eta) -
    (x^alpha) * exp(beta_0 + eta);
}
```

```

real myweibull_lccdf(real x, real eta, real alpha, real beta_0){
    return -(x^alpha) * exp(eta + beta_0);
}
}

data {
    int<lower=1> N_censored; // Number of censored individuals
    int<lower=1> N_uncensored; // Number of uncensored individuals
    int<lower=1> N_typical; // Number of typical individuals
    int<lower=0> N_predictors;
    vector<lower=0>[N_censored] Observed_times_censored;
    vector<lower=0>[N_uncensored] Observed_times_uncensored;
    matrix[N_censored, N_predictors] Z_censored;
    matrix[N_uncensored, N_predictors] Z_uncensored;
    matrix[N_typical, N_predictors] Z_typical;
    real scale_alpha; //prior sd on alpha
    vector[N_predictors] scale_beta; // prior sds on beta
    vector[N_predictors] mean_beta; // prior means on beta
    real scale_beta0; //prior sd on beta0
}

parameters {
    real<lower=0> alpha; // =1/sigma sigma is the scale parameter of log weibull
    vector[N_predictors] beta; // regression coefficients beta vector
    real beta_0; //intercept
}

transformed parameters {
    vector[N_predictors] exp_beta;
    vector[N_predictors] gamma;
    real mu_tilda;
    real sigma;

    exp_beta = exp(beta);
    gamma = -beta/alpha;
    mu_tilda = -beta_0 / alpha;
    sigma = 1/alpha;
}

model {
    real eta;
    row_vector[N_predictors] Z;
    real x;

    // priors

```

```

alpha ~ normal(0, scale_alpha); // half normal
beta_0 ~ cauchy(0, scale_beta);

for (i in 1:N_predictors) {
  beta[i] ~ normal(mean_beta[i], scale_beta[i]);
}

for (i in 1:N_censored){
  x = Observed_times_censored[i];
  Z = Z_censored[i];
  eta = Z * beta;
  target += myweibull_lccdf(x|eta, alpha, beta_0);
}

for (i in 1:N_uncensored){
  x = Observed_times_uncensored[i];
  Z = Z_uncensored[i];
  eta = Z * beta;
  target += myweibull_lpdf(x|eta, alpha, beta_0);
}

generated quantities{
vector[N_typical] times_typical_sampled;

for(i in 1:N_typical) {
  times_typical_sampled[i] = exp(mu_tilda + Z_typical[i]*gamma -
  sigma *gumbel_rng(0,1));
}
}"

```

We check convergence of parameters, with traceplot.

```

load("Stan_pos_sample2.RData")
params <- params2

par(mfrow = c(4, 3))
# trace plot for alpha
ts.plot(params$alpha)

# trace plot for beta_0
ts.plot(params$beta_0)

# trace plot for betas for covariates
ts.plot(params$beta[,1])
ts.plot(params$beta[,2])

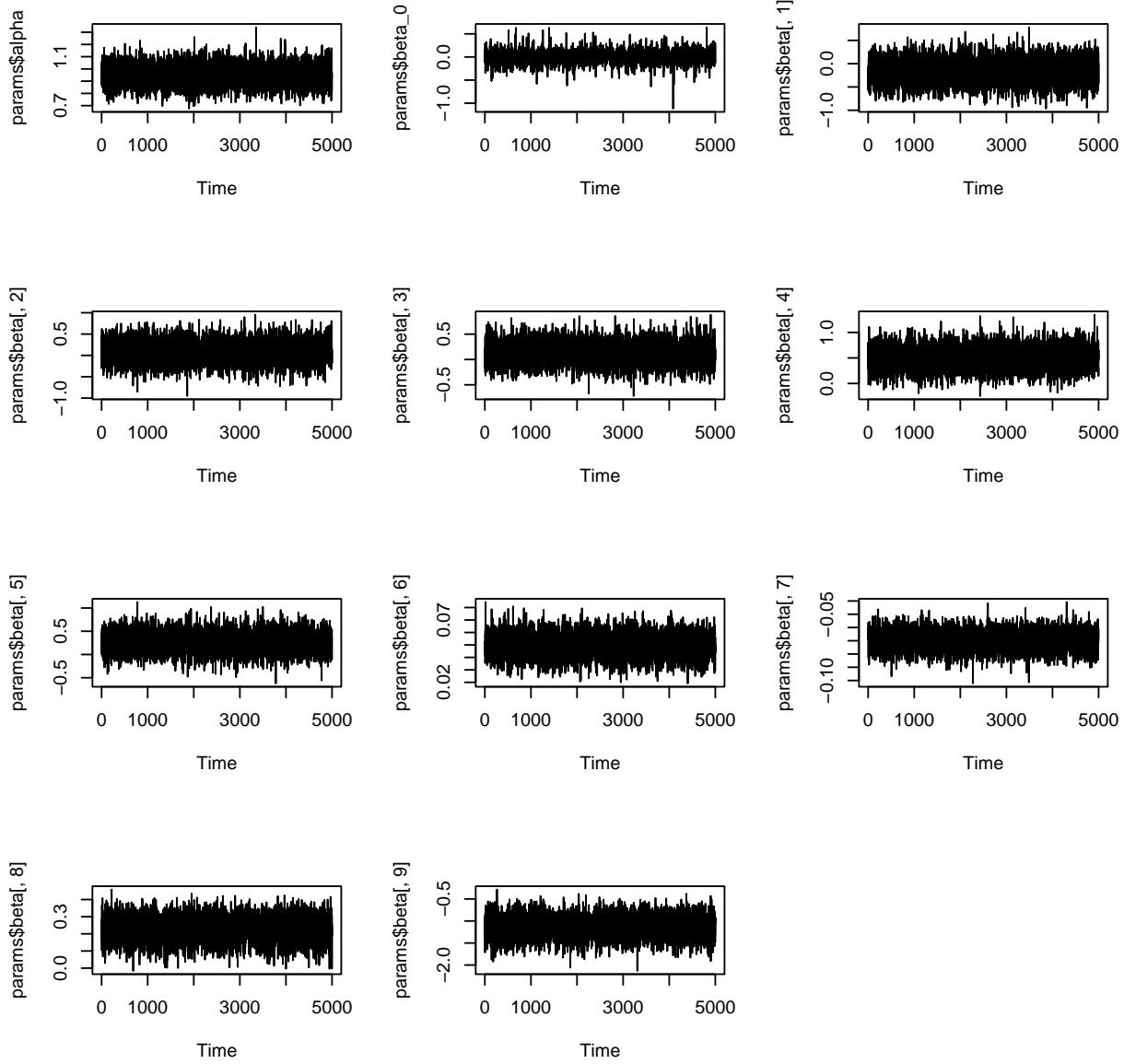
```

```

ts.plot(params$beta[,3])
ts.plot(params$beta[,4])
ts.plot(params$beta[,5])
ts.plot(params$beta[,6])
ts.plot(params$beta[,7])
ts.plot(params$beta[,8])
ts.plot(params$beta[,9])

par(mfrow = c(2, 3))

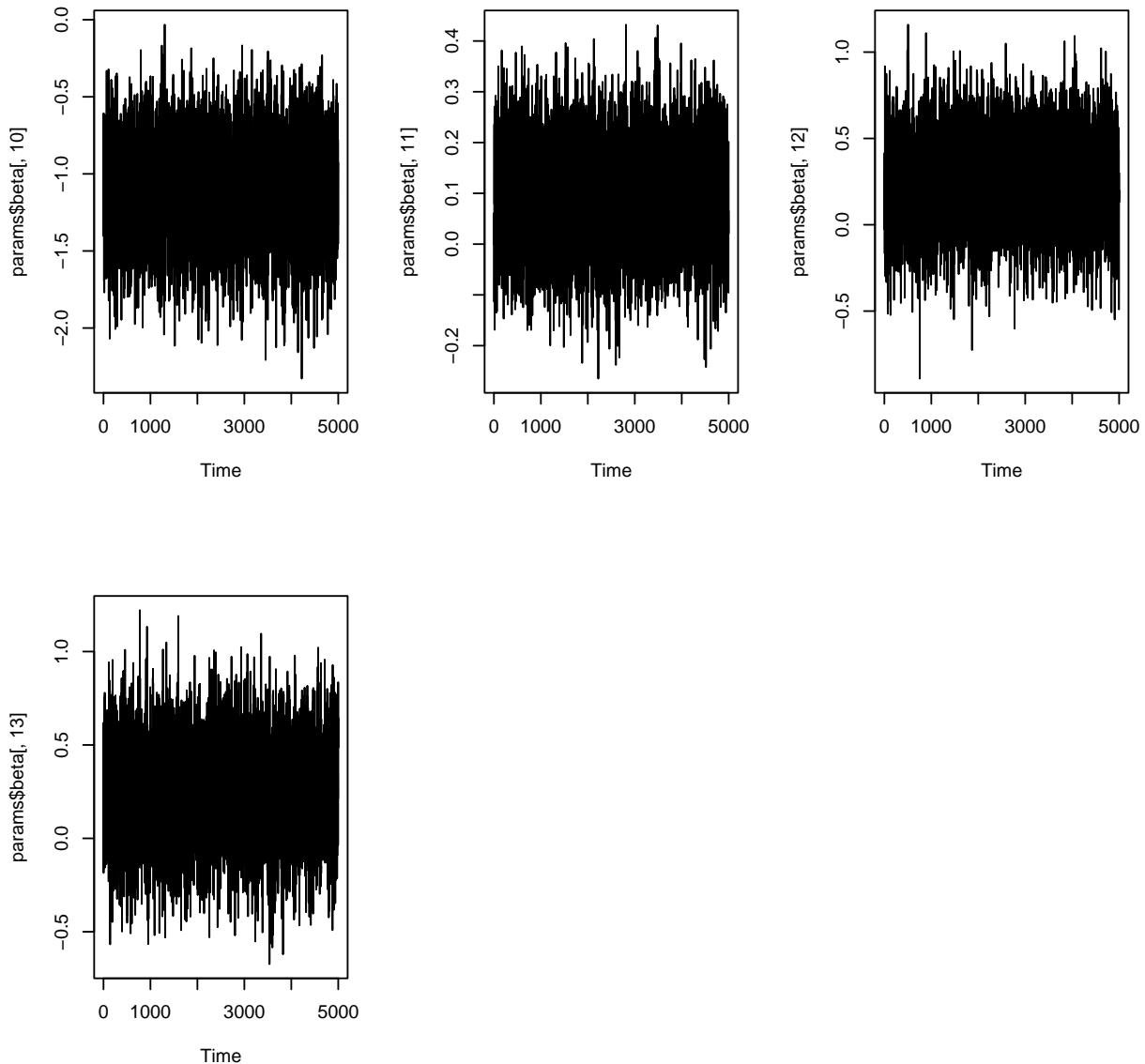
```



```

ts.plot(params$beta[, 10])
ts.plot(params$beta[, 11])
ts.plot(params$beta[, 12])
ts.plot(params$beta[, 13])

```



All parameters that we assume priors seems to have convergent posterior samples.

	mean	2.5%	50%	97.5%	n_eff	Rhat
## alpha	0.9470	0.7896	0.9455	1.1113	4715.849	0.9993
## beta[1]	-0.1560	-0.6394	-0.1551	0.3214	4882.845	1.0008
## beta[2]	0.0884	-0.3984	0.0908	0.5761	4702.954	1.0010
## beta[3]	0.1339	-0.2888	0.1339	0.5408	5173.268	1.0004

```

## beta[4]    0.5057  0.0785  0.5065  0.9174 4964.206 1.0000
## beta[5]    0.2613 -0.1640  0.2580  0.6931 5147.355 1.0006
## beta[6]    0.0496  0.0312  0.0496  0.0671 4661.028 1.0000
## beta[7]   -0.0698 -0.0843 -0.0699 -0.0554 4880.421 0.9996
## beta[8]    0.2297  0.0835  0.2323  0.3567 4969.063 0.9996
## beta[9]   -1.1433 -1.6157 -1.1424 -0.6709 4972.120 0.9997
## beta[10]   -1.1094 -1.7408 -1.0996 -0.4981 5027.476 1.0002
## beta[11]   0.0922 -0.0999  0.0942  0.2845 5368.042 1.0000
## beta[12]   0.2514 -0.2634  0.2526  0.7456 5119.584 0.9998
## beta[13]   0.2432 -0.2781  0.2430  0.7552 5047.793 1.0001
## beta_0   -0.0031 -0.2335 -0.0021  0.2270 4995.440 1.0003

```

\hat{R} for all parameters are close to 1, and effective sample sizes are very high.

```

quantile(params$beta[,1], c(0.025, 0.975)) #Gender

##      2.5%      97.5%
## -0.6393792  0.3214060

quantile(exp(params$beta[,1]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.5276198  1.3790653

quantile(params$beta[,2], c(0.025, 0.975)) #Smoking

##      2.5%      97.5%
## -0.3984307  0.5760796

quantile(exp(params$beta[,2]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.6713728  1.7790501

quantile(params$beta[,3], c(0.025, 0.975)) #Diabetes

##      2.5%      97.5%
## -0.2888270  0.5408325

quantile(exp(params$beta[,3]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.7491418  1.7174360

quantile(params$beta[,4], c(0.025, 0.975)) #BP

```

```

##          2.5%      97.5%
## 0.07849585 0.91735132

quantile(exp(params$beta[,4]), c(0.025, 0.975))

##          2.5%      97.5%
## 1.081659 2.502653

quantile(params$beta[,5], c(0.025, 0.975)) #Anaemia

##          2.5%      97.5%
## -0.1640228 0.6930939

quantile(exp(params$beta[,5]), c(0.025, 0.975))

##          2.5%      97.5%
## 0.8487227 1.9998935

quantile(params$beta[,6], c(0.025, 0.975)) #Age

##          2.5%      97.5%
## 0.03120222 0.06713759

quantile(exp(params$beta[,6]), c(0.025, 0.975))

##          2.5%      97.5%
## 1.031694 1.069443

quantile(params$beta[,7], c(0.025, 0.975)) #Sodium

##          2.5%      97.5%
## -0.08433669 -0.05542151

quantile(exp(params$beta[,7]), c(0.025, 0.975))

##          2.5%      97.5%
## 0.9191217 0.9460863

quantile(params$beta[,8], c(0.025, 0.975)) #Creatinine

##          2.5%      97.5%
## 0.08346112 0.35669526

quantile(exp(params$beta[,8]), c(0.025, 0.975))

##          2.5%      97.5%
## 1.087043 1.428600

```

```

quantile(params$beta[,9], c(0.025, 0.975)) #EF30to45

##      2.5%      97.5%
## -1.6157067 -0.6709069

quantile(exp(params$beta[,9]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.1987502 0.5112447

quantile(params$beta[,10], c(0.025, 0.975)) #EFgeq45

##      2.5%      97.5%
## -1.7408362 -0.4980553

quantile(exp(params$beta[,10]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.1753737 0.6077113

quantile(params$beta[,11], c(0.025, 0.975)) #log_CPK

##      2.5%      97.5%
## -0.09994845 0.28453056

quantile(exp(params$beta[,11]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.9048841 1.3291379

quantile(params$beta[,12], c(0.025, 0.975)) #PleteletsleqQ1

##      2.5%      97.5%
## -0.2633502 0.7455563

quantile(exp(params$beta[,12]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.7684727 2.1076136

quantile(params$beta[,13], c(0.025, 0.975)) #PleteletsgeqQ3

##      2.5%      97.5%
## -0.2781307 0.7551631

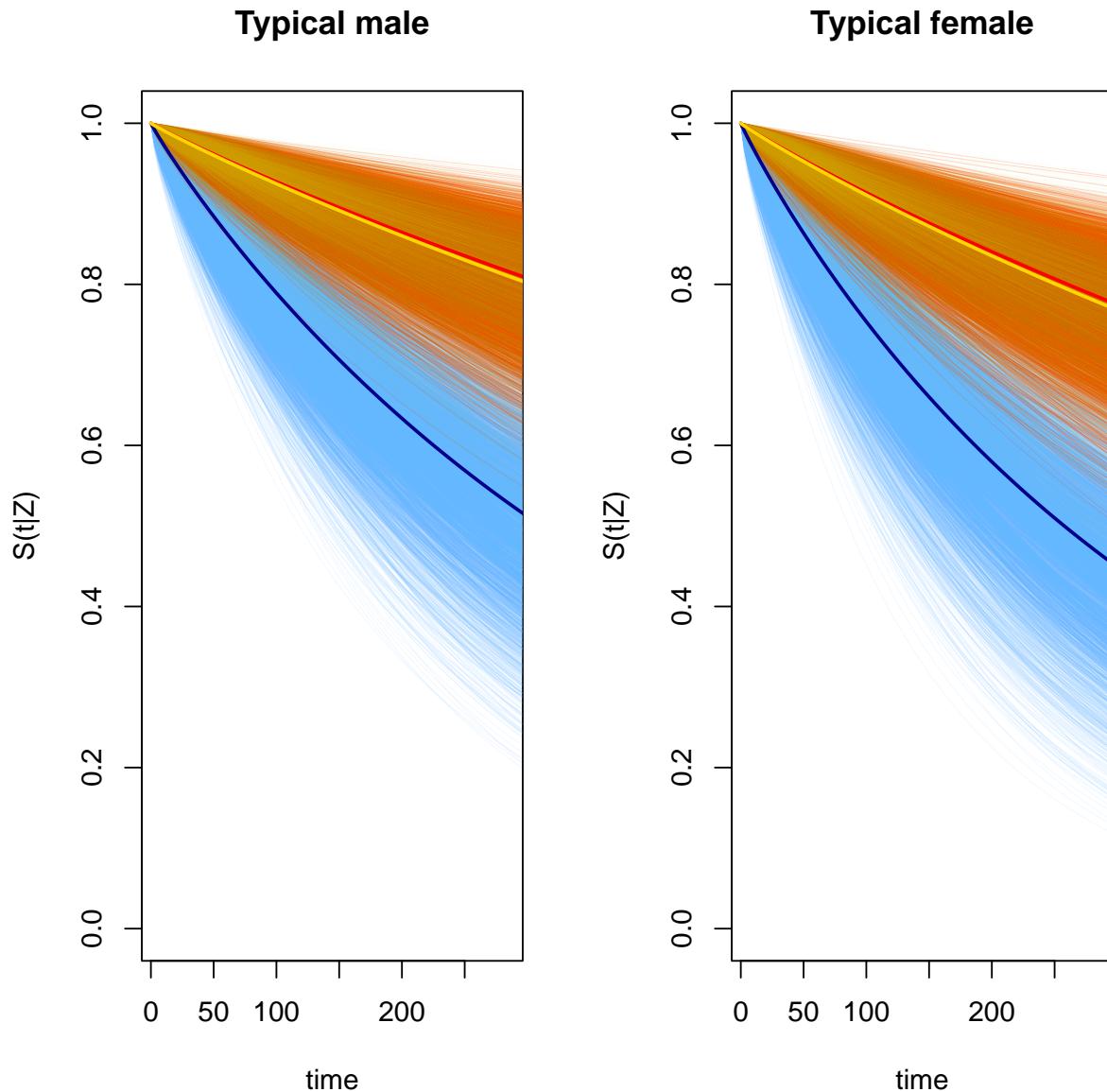
quantile(exp(params$beta[,13]), c(0.025, 0.975))

##      2.5%      97.5%
## 0.7571978 2.1279586

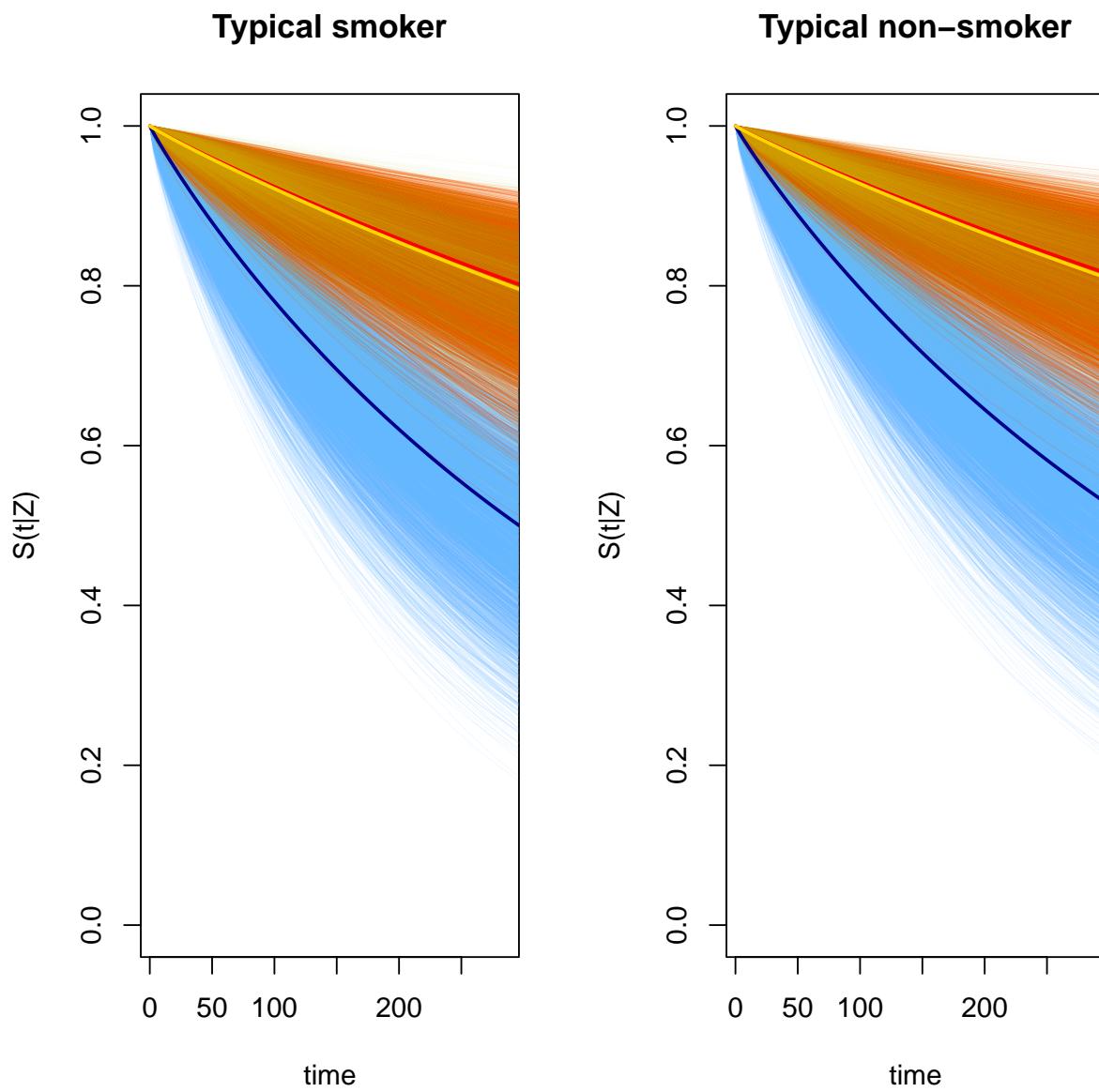
```

The result here agrees with the result with diffuse prior From 95% posterior quantile, we conclude that blood pressure (BP), age, Serum sodium (sodium), Serum creatinine (creatinine), indicator of ejection fraction (EF) between 30 to 45 (EF30t45), and indicator of ejection fraction (EF) greater or equal 45 (EFgeq45) are significant factor that impact the survival experience of heart failure patients.

Interval interpretations about $\exp(\text{coefficients})$ can be done as before.



Our previous conclusions about survival curves for typical male vs female stand.



Our previous conclusions about survival curves for typical smoker vs non-smoker stand.

We calculate predicted probabilities that a typical male patient outlives a typical female patient for 3 categories of EF.

```
## [1] 0.534
## [1] 0.534
## [1] 0.532
```

We calculate predicted probabilities that a typical smoking patient outlives a typical non-smoking patient for 3 categories of EF.

```
## [1] 0.4808  
## [1] 0.4762  
## [1] 0.479
```

At last, for further comparison, we also report the predicted survival times for all the typical individuals.

```
## [1] 488.069  
## [1] 414.1226  
## [1] 1672.067  
## [1] 1420.713  
## [1] 1684.522  
## [1] 1443.168
```

```
## [1] 479.095  
## [1] 520.4877  
## [1] 1654.623  
## [1] 1787.767  
## [1] 1677.555  
## [1] 1821.031
```

The result is similar as before.

The conclusion based on these probabilities essentially agree with previous analysis with diffuse prior.

4 Future directions

In this project, following the paper of Ahmed et al. (2017), we do not model interactions of covariates, so some of the consistencies in our conclusions are from our modelling assumptions rather than the reality. Therefore, our conclusions should be considered an initial insight rather than a well established result. It is importance to check the existence of interactions, take them into account and compare with our current result, but this is left as something to be done in the future.

5 References

Ahmad, Tanvir, et al. "Survival analysis of heart failure patients: A case study." PloS one 12.7 (2017): e0181001.

Brilleman, Samuel L., et al. "Bayesian survival analysis using the rstanarm R package." arXiv preprint arXiv:2002.09633 (2020).

Klein, John P., and Melvin L. Moeschberger. Survival analysis: techniques for censored and truncated data. Vol. 1230. New York: Springer, 2003.