

Bayesian Analysis of Type II Generalized Topp–Leone Accelerated Failure Time Models Using R and Stan

DEVASHISH AND ATHAR ALI KHAN



Department of Statistics and Operations Research
Aligarh Muslim University, Aligarh-202002, India
devashishstatistics@gmail.com
atharkhan1962@gmail.com

Abstract

With a Bayesian framework, the current study intends to fit the Type II generalized Topp–Leone-G (TIIGTL-G) model as an accelerated failure time (AFT) model to censored survival data. In this paper, we have obtained and analysed three AFT models using Type II Generalized Topp-Leone (TIIGTL) distribution as generator and considering Weibull, Exponential, and Log-Logistic as a baseline distribution. The fitting of these models to the censored survival data is done with the help of R and STAN. A comparison of these two models is conducted, and the best model is chosen using the Bayesian model evaluation criteria LOOIC and WAIC.

Keywords: Type II generalized Topp–Leone G Model, Bayesian Survival Modelling, Censored data, Leave one out information criteria, STAN

1. INTRODUCTION

[1] proposed the Type II generalised Topp-Leone-G (TIIGTL-G) family of distributions, which uses the Topp-Leone random variable as a generator, and investigated its mathematical properties and how they were used to fit lifetime data. Research analysts are evaluating lifetime data and issues with modelling the survival process using the extended form of the standard distribution in the survival study. It has been shown that the Bayesian paradigm is instrumental in analyzing survival models in many real-world contexts. [2] set up and analysed Topp-Leone exponential distribution, Topp-Leone exponentiated exponential distribution and Topp-Leone exponentiated extension distribution using Bayesian approach. Also, [3] fitted the Weibull, Topp-Leone-Weibull (TL-W), and Generalized Topp-Leone-Weibull (GTL-W) survival models as accelerated failure time models using Bayesian approach and have shown that TL-W AFT model is the most suitable model for fitting a censored data (tumor data). Recently, [4] analysed and compared three accelerated failure time models—Weibull, log-normal, and log-logistic under Bayesian framework.

In this article, We have fitted a censored survival data using TIIGTL-G model as an accelerated failure time (AFT) model. The aforementioned models were fitted using the full Bayesian inference-supporting probabilistic programming language STAN [5] in R. The programming language Stan is used to define statistical models, and in Bayesian analysis, it is most frequently employed as a Hamiltonian Monte Carlo (HMC) sampler [6, 7]. STAN primarily uses the No-U-Turn sampler (NUTS) [8] to obtain posterior simulation for Bayesian analysis. Thus, we have also evaluated and selected the best model using Leave-One-Out information criteria (LOOIC) and Watanabe-Akaike information criteria or widely applicable information criteria (WAIC) for the diet data. Using a fitted Bayesian model and the log-likelihood assessed at the posterior simulations of the parameter

values, LOO and WAIC are two methods for evaluating the precision of pointwise out-of-sample predictions [9]. Thus, in this article, we have conducted a Bayesian analysis of TIIGTL-Weibull AFT, TIIGTL-Exponential AFT, and TIIGTL-Log-logistic AFT models by presenting summaries of the posterior densities in both numerical and graphical form by using R and Stan.

2. TYPE II GENERALIZED TOPP-LEONE-G (TIIGTL-G) FAMILY

Let a continuous random variable T with baseline cdf and pdf $G(t, \phi)$ and $g(t, \phi)$ respectively with parameter vector ϕ . The cumulative distribution function (cdf), probability density function (pdf), survival function, and hazard function of the TIIGTL-G family are respectively given by

$$F_T(t, c, d, \phi) = 1 - (1 - G(t, \phi)^{2d})^c \tag{1}$$

$$f_T(t, c, d, \phi) = 2cdg(t, \phi)[G(t, \phi)]^{2d-1}(1 - G(t, \phi)^{2d})^c \tag{2}$$

$$S_T(t, c, d, \phi) = 1 - F_T(t, c, d, \phi) = (1 - G(t, \phi)^{2d})^c \tag{3}$$

$$h_T(t, c, d, \phi) = f_T(t, c, d, \phi) / S_T(t, c, d, \phi) \tag{4}$$

Thus, the random variable T with the pdf given in Equation 2 will be denoted as $T \sim TIIGTL - G(c, d, \phi)$ where c,d are two shape parameters and ϕ is parameter vector of baseline distribution. Also, random number generation from the survival model is accomplished by equating $F(t)$ and v , where V has *Uniform*(0,1) distribution. Thus,

$$F(t) = v \tag{5}$$

$$1 - (1 - G(t)^{2d})^c = v \tag{6}$$

then we have,

$$G(t) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{7}$$

For any baseline cdf G(t), this is the TIIGTL-G model's general expression for producing random numbers.

3. ACCELERATED FAILURE TIME (AFT) MODELS

It has been noted in statistical literature that many models have been created for assessing survival data or life time data. The Cox Proportional Hazard (PH) model is the most well-liked of them all. When examining survival data, the Accelerated Failure Time (AFT) model can be thought of as a good substitute for the Cox PH model [10]. AFT models are parametric models that take into account the linear regression of the logarithm of the survival time T on a variety of covariates. They are used to investigate the impact of a covariate on how quickly or slowly the survival process advances [3]. According to the AFT model, covariates and failure time have a direct relationship [11]. If number of covariates x_1, x_2, \dots, x_p have an impact on survival time T then we can write the AFT model as:

$$\log(T) = \beta_0 + \sum_{k=1}^L \beta_k x_k + \sigma e = \mathbf{x}'\beta + \sigma e \tag{8}$$

where $\beta_k, k = 1, 2, \dots, L$ are the coefficients of regression, σ is a scale parameter such that $\sigma > 0$ and e is the random error with a specified probability distribution.

3.1. Weibull AFT model

Let survival time T follows Weibull distribution with scale and shape parameter λ and α respectively. Then the probability density function, cumulative distribution function, and survival function of Weibull distribution are provided as follows [12]:

$$g(t|\alpha, \lambda) = (\alpha/\lambda)(t/\lambda)^{\alpha-1} \exp(-(t/\lambda)^\alpha) \tag{9}$$

$$G(t|\alpha, \lambda) = 1 - \exp(-(t/\lambda)^\alpha) \tag{10}$$

$$S(t|\alpha, \lambda) = \exp(-(t/\lambda)^\alpha) \tag{11}$$

hence, we can write $T \sim Weibull(\alpha, \lambda)$. Now, Let a random variable ϵ has a standard extreme value distribution with density function $g(e) = \exp(e - \exp(e))$ and survival function $S(e) = \exp(-\exp(e))$ substituting $e = (\log t - \mathbf{x}'\beta)/\sigma$ from the Equation 8 in the extreme value distribution and then the Weibull AFT model is obtained and we can write it as $T \sim Weibull(1/\sigma, \exp(\mathbf{x}'\beta))$.

3.1.1 TIIGTL-W AFT model

The Type two generalized Topp-Leone-Weibull (TIIGTL-W) AFT model is obtained by considering weibull AFT model as the baseline model G and substituting it in the TIIGTL-G model. Thus, the cdf, pdf, survival function, and hazard function of the TIIGTL-W AFT model are respectively given by

$$F(t|\Omega, \mathbf{x}) = 1 - (1 - G(t)^{2d})^c \tag{12}$$

$$f(t|\Omega, \mathbf{x}) = 2abg(t)[G(t)]^{2d-1}(1 - G(t)^{2d})^c \tag{13}$$

$$S(t|\Omega, \mathbf{x}) = (1 - G(t)^{2d})^c \tag{14}$$

$$h(t|\Omega, \mathbf{x}) = f(t|\Omega, \mathbf{x})/S(t|\Omega, \mathbf{x}) \tag{15}$$

Where $t > 0$, $g(t)$ and $G(t)$ are the pdf and cdf of Weibull AFT model. $\Omega = (c, d, \sigma, \beta)$, c, d and α are shape parameters and λ is scale parameter. Also $\sigma = 1/\alpha, \lambda = \exp(\mathbf{x}'\beta)$ from the AFT model and we have $T \sim TIIGTL - W(c, d, 1/\sigma, \exp(\mathbf{x}'\beta))$. Now, for random number generation from TIIGTL-W we proceed as follows, Let $V \sim Uniform(0, 1)$. Then from Equation 7 we have

$$G(t) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{16}$$

$$1 - \exp(-(t/\lambda)^\alpha) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{17}$$

then we get,

$$t = \exp(\mathbf{x}'\beta) [-\log(1 - (1 - (1 - v)^{1/c})^{1/2d})]^\sigma \tag{18}$$

This is the TIIGTL-W AFT model's general expression for producing random numbers, where $\lambda = \exp(\mathbf{x}'\beta)$ and $\sigma = 1/\alpha$.

3.2. Exponential AFT model

Let survival time T follows Exponential distribution with inversescale or rate parameter $\lambda > 0$ Then the probability density function, cumulative distribution function, and survival function of Exponential distribution are provided as follows [12]:

$$g(t|\alpha, \lambda) = 1 - \exp(-\lambda t) \tag{19}$$

$$G(t|\alpha, \lambda) = \lambda \exp(-\lambda t) \tag{20}$$

$$S(t|\alpha, \lambda) = \exp(-\lambda t) \tag{21}$$

hence, we can write $T \sim Exp(\lambda)$. Now, Let a random variable ϵ has a standard extreme value distribution with density function $g(e) = \exp(e - \exp(e))$ and survival function $S(e) = \exp(-\exp(e))$. Considering $\sigma = 1$ substituting $e = (\log t - \mathbf{x}'\beta)$ from the Equation 8 in the extreme value distribution and then the Exponential AFT model is obtained and we can write it as $T \sim Exp(\exp(-\mathbf{x}'\beta))$.

3.2.1 TIIGTL-E AFT model

The Type two generalized Topp-Leone-Exponential (TIIGTL-E) AFT model is obtained by considering exponential AFT model as the baseline model G and substituting it in the TIIGTL-G model. Thus, the cdf, pdf, survival function, and hazard function of the TIIGTL-E AFT model are respectively given by

$$F(t|\Omega, \mathbf{x}) = 1 - (1 - G(t)^{2d})^c \tag{22}$$

$$f(t|\Omega, \mathbf{x}) = 2cdg(t)[G(t)]^{2d-1}(1 - G(t)^{2d})^c \tag{23}$$

$$S(t|\Omega, \mathbf{x}) = (1 - G(t)^{2d})^c \tag{24}$$

$$h(t|\Omega, \mathbf{x}) = f(t|\Omega, x)/S(t|\Omega, x) \tag{25}$$

Where $t > 0$, $g(t)$ and $G(t)$ are the pdf and cdf of Exponential AFT model. $\Omega = (c, d, \beta)$, c, d are shape parameters and λ is inversescale parameter. Also $\lambda = \exp(-\mathbf{x}'\beta)$ from the AFT model and we have $T \sim TIIGTL - E(c, d, \exp(-\mathbf{x}'\beta))$. Now, for random number generation from TIIGTL-E we proceed as follows, Let $V \sim Uniform(0, 1)$. Then from Equation 7 we have

$$G(t) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{26}$$

$$1 - \exp(-\lambda t) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{27}$$

then we get,

$$t = (-\exp(\mathbf{x}'\beta))\log[1 - (1 - (1 - v)^{1/c})^{1/2d}] \tag{28}$$

This is the TIIGTL-E AFT model's general expression for producing random numbers, where $\lambda = \exp(\mathbf{x}'\beta)$.

3.3. Log Logistic AFT model

Let survival time T follows Log Logistic distribution with scale and shape parameter λ and α respectively. Then the probability density function, cumulative distribution function, and survival function of Log Logistic distribution are provided as follows [12]:

$$g(t|\alpha, \lambda) = (\alpha/\lambda)(t/\lambda)^{\alpha-1}(1 + (t/\lambda)^\alpha)^{-2} \tag{29}$$

$$G(t|\alpha, \lambda) = 1 - (1 + (t/\lambda)^\alpha)^{-1} \tag{30}$$

$$S(t|\alpha, \lambda) = (1 + (t/\lambda)^\alpha)^{-1} \tag{31}$$

hence, we can write $T \sim LL(\alpha, \lambda)$. Now, Let a random variable e has a standard logistic value distribution with density function $g(e) = \exp(e)(1 - \exp(e))^{-2}$ and survival function $S(e) = (1 - \exp(e))^{-1}$ substituting $e = (\log t - \mathbf{x}'\beta)/\sigma$ from the Equation 8 in the extreme value distribution and then the Log Logistic AFT model is obtained and we can write it as $T \sim LL(1/\sigma, \exp(\mathbf{x}'\beta))$.

3.3.1 TIIGTL-LL AFT model

The Type two generalized Topp-Leone-log-logistic (TIIGTL-LL) AFT model is obtained by considering log-logistic AFT model as the baseline model G and substituting it in the TIIGTL-G model. Thus, the cdf, pdf, survival function, and hazard function of the TIIGTL-W AFT model are respectively given by

$$F(t|\Omega, \mathbf{x}) = 1 - (1 - G(t)^{2d})^c \tag{32}$$

$$f(t|\Omega, \mathbf{x}) = 2cdg(t)[G(t)]^{2d-1}(1 - G(t)^{2d})^c \tag{33}$$

$$S(t|\Omega, \mathbf{x}) = (1 - G(t)^{2d})^c \tag{34}$$

$$h(t|\Omega, \mathbf{x}) = f(t|\Omega, x)/S(t|\Omega, x) \tag{35}$$

Where $t > 0$, $g(t)$ and $G(t)$ are the pdf and cdf of Log-logistic AFT model. $\Omega = (c, d, \sigma, \beta)$, c, d and α are shape parameters and λ is scale parameter. Also $\sigma = 1/\alpha, \lambda = \exp(\mathbf{x}'\beta)$ from the AFT

model and we have $T \sim TIIGTL - LL(c, d, 1/\sigma, \exp(\mathbf{x}'\beta))$. Now, for random number generation from TIIGTL-LL we proceed as follows, Let $V \sim Uniform(0, 1)$. Then from Equation 7 we have

$$G(t) = [1 - (1 - v)^{1/c}]^{1/2d} \tag{36}$$

$$1 - (1 + (t/\lambda)^\alpha)^{-1} = [1 - (1 - v)^{1/c}]^{1/2d} \tag{37}$$

then we get,

$$t = \exp(\mathbf{x}'\beta)[(1 - (1 - (1 - v)^{1/c})^{1/2d})^{-1} - 1]^\sigma \tag{38}$$

This is the TIIGTL-LL AFT model's general expression for producing random numbers, where $\lambda = \exp(\mathbf{x}'\beta)$ and $\sigma = 1/\alpha$.

4. DIET DATA

90 homogenous rats of the same species, age, and environmental conditions were separated into three groups and fed with low, saturated, and unsaturated fat diets, respectively, as reported by [13]. Each rat's foot pad received an identical dosage of tumour cells. 200 days of observation of the rats revealed the growth of a tumour as the event. Several of the rats got tumours, but several did not. Survival time is defined as the amount of time without a tumour or the amount of time before one develops one. The survival times of the tumor-free animals are marked with stars and treated as censored. As a result, the data is correctly suppressed, as shown in the Table 1. The primary objective of this study is to compare the three diets' tumor-preventing capacities in rats.

Table 1: Tumor-free duration (days) of 90 rats on three different diets (* indicates censored)

Low Fat (30 rats)	Saturated Fat (30 rats)	Unsaturated Fat (30 rats)
140 87 200*	124 96 81	112 63 66
177 56 200*	58 142 133	68 63 94
50 66 200*	56 86 165	84 77 101
65 73 200*	68 75 170*	109 91 105
86 119 200*	79 117 200*	153 91 108
153 140* 200*	89 98 200*	143 66 112
181 200* 200*	107 105 200*	60 70 115
191 200* 200*	86 126 200*	70 77 126
77 200* 200*	142 43 200*	98 63 161
84 200* 200*	110 46 200*	164 66 178

4.1. Data Structure for computation in R

We have produced the data in a listed form necessary for fitting Bayesian models to the data using *stan* function.

```

y = survival times (Tumor-free time in days)
y <- c(140,177,50,65,86,153,181,191,77,84,87,56,66,73,119,140,200,200,
200,200,200,200,200,200,200,200,200,200,200,200,124,58,56,68,79,89,107,
86,142,110,96,142,86,75,117,98,105,126,43,46,81,133,165,170,200,200,200,
200,200,200,112,68,84,109,153,143,60,70,98,164,63,63,77,91,91,66,70,77,
63,66,66,94,101,105,108, 112,115,126,161,178 )
event=1 if tumor is developed or zero if it is censored
event <- c(rep(1,15),rep(0,15),rep(1,23),rep(0,7),rep(1,30))
Low-Fat is considered as reference category
x1 = 1 if saturated fat is applied and 0 otherwise
x1 <- c(rep(0,30),rep(1,30),rep(0,30))
    
```

```
x2 = 1 if unsaturated fat is applied and 0 otherwise
x2 <- c(rep(0,30),rep(0,30),rep(1,30))
x = cbind(1,x1,x2)
N = nrow(x)
M = ncol(x)
datt = list(y=y, event=event,x=x,N=N,M=M)
```

5. BAYESIAN ANALYSIS

In Bayesian analysis, following Bayes Theorem, we look for the exact parameter distributions known as the posterior distribution by fusing the prior distribution of parameter with the data or likelihood. We must define a prior distribution for the model’s parameters and likelihood of the data before building the Bayesian regression model.

5.1. Likelihood

Following the [14], the joint likelihood function for right censored data is given as

$$L = \prod_{i=1}^n h(t_i)^{\gamma_i} S(t_i) \tag{39}$$

Also as an alternative to the likelihood, the log-likelihood can be written as

$$\log L = \sum_{i=1}^n (\gamma_i (\log h(t_i) + \log S(t_i))) \tag{40}$$

here γ_i is an indicator variable such that $\gamma = 0$ if the observed value is censored and $\gamma = 1$ if the observed value is failed (recorded). In equation 39 we can substitute the hazard function $h(t_i)$ and survival function $S(t_i)$ of TIIGTL-W AFT, TIIGTL-E AFT and TIIGTL-LL AFT models in order to get the likelihood of TIIGTL-W AFT, TIIGTL-E AFT and TIIGTL-LL AFT survival models respectively.

5.2. Prior

A prior distribution must be specified for the model’s parameters in order to build a Bayesian regression model. Two prior types—the student t prior and the normal prior, are used by the researchers in the remaining sections of this work. Student t distribution is used for the priors of shape and scale parameters and Normal distribution is used as a prior for the regression coefficients. These priors are weakly informative priors and are discussed briefly by [3].

5.3. Posterior

The Bayes Theorem can be used to determine the joint posterior distribution of parameter $\Omega = (c, d, \sigma, \beta) = (c, d, \sigma, \beta_0, \beta_1, \dots, \beta_p)$ given data as

$$P(\Omega|t, X) \propto L(\Omega|t, X)P(\Omega) \tag{41}$$

$$P(\Omega|t, X) \propto L(\Omega|t, X)P(c)P(d)P(\sigma)P(\beta) \tag{42}$$

Here parameters are assumed to be independent and X is the matrix of covariates. Hence we can obtain the joint posterior distribution of TIIGTL-W AFT Model, TIIGTL-E AFT Model and TIIGTL-LL AFT Model by substituting the likelihood and priors of corresponding models in equation 42. Because it is challenging to determine the marginal distributions of the parameters and the normalised joint posterior distribution analytically, the estimates and other relevant results are obtained using the Markov chain Monte Carlo (MCMC) simulation technique.

5.4. Implementation using Stan

The package `rstan` is necessary to run STAN code in R. For the Bayesian modeling, there are several blocks in Stan such as Data and Transformed data block, Parameter, and Transformed parameter block, Generated quantities block, etc. Following are the stan codes containing all these blocks for all three models discussed in this article.

5.4.1 Stan code for TIIGTL-W AFT model

```

stancode_ttgtlw = "
functions{
// defines the log survival
vector log_S (vector t,real shape1,real shape2,
real shape3,vector scale){
vector[num_elements(t)] log_S ;
for (i in 1:num_elements(t)){
log_S[i] = log(((1-((weibull_cdf(t[i],shape3,
scale[i]))^(2*shape2)))^(shape1)))));
}
return log_S;
}
//defines the log hazard
vector log_h (vector t,real shape1,real shape2,
real shape3,vector scale){
vector[num_elements(t)] log_h ;
vector[num_elements(t)] ls ;
ls = log_S(t,shape1,shape2,shape3,scale) ;
for (i in 1:num_elements(t)){
log_h[i] = (log(2)+log(shape1)+log(shape2)+
weibull_lpdf(t[i]|shape3,scale[i]))+
(((2*shape2)-1)*weibull_lcdf(t[i]|shape3,scale[i]))+
((shape1-1)*(log(1-(weibull_cdf(t[i],shape3,
scale[i]))^(2*shape2)))))) - ls[i];
}
return log_h;
}
//defines the log-likelihood for right censored data
real surv_ttgtlw_lpdf(vector t,vector d,real shape1,
real shape2,real shape3,vector scale){
vector[num_elements(t)] log_lik;
real prob;
log_lik = d .* log_h(t,shape1,shape2,shape3,scale)+
log_S(t,shape1,shape2,shape3,scale);
prob = sum(log_lik);
return prob;
}
}
//data block
data{
int N; // number of observations
vector <lower=0> [N] y;// observed times
vector <lower=0,upper=1> [N] event;//censoring(1=obs.,
// 0=cens.)
int M; // number of covariates

```

```

matrix[N,M] x;//model matrix (N rows, M columns)
}
//parameters block
parameters{
vector [M] beta;//coef.in the linear predictor
real<lower=0> shape1;// shape parameter
real<lower=0> shape2;// shape parameter
real<lower=0> sigma;//scale parameter sigma=1/shape3
}
// transformed parameters block
transformed parameters{
vector[N] linpred;
vector[N] mu;
linpred = x*beta; //linear predictor
for (i in 1:N){
mu[i] = exp(linpred[i]);
}
}
// model block
model{
shape1 ~ student_t(5,0,10) T[0, ];//prior for shape1
shape2 ~ student_t(5,0,10) T[0, ];//prior for shape2
sigma ~ student_t(2,0,10) T[0, ];//prior for sigma
beta ~ normal(0,10);//prior for reg. coefficients
y ~ surv_ttgtlw(event,shape1,shape2,1/sigma,mu);
//model for the data
}
// generated quantities block
generated quantities{
vector[N] y_rep;//posterior predictive value
vector[N] log_lik;//log-likelihood
{ for(n in 1:N){
log_lik[n] = ((log(2)+log(shape1)+log(shape2))+
weibull_lpdf(y[n]|1/sigma,exp(x[n,]*beta))+
(((2*shape2)-1)*weibull_lcdf(y[n]|1/sigma,
exp(x[n,]*beta))))+(shape1-1)*
(log(1-(weibull_cdf(y[n],1/sigma,exp(x[n,]*beta)))^
(2*shape2))))-log(((1-((weibull_cdf(y[n],1/sigma,
exp(x[n,]*beta)))^(2*shape2))))^(shape1))))*event[n])+
(log(((1-((weibull_cdf(y[n],1/sigma,exp(x[n,]*beta)))^(2*shape2))))
^(shape1)))));}
}
{real u;
u=uniform_rng(0,1);
for (n in 1:N){
y_rep[n] = (exp(x[n,]*beta))*(-log((1-(1-((1-u)^(1/shape1)))
^(1/(2*shape2))))^(sigma)));}
}
}
"

```

5.4.2 Stan code for TIIGTL-E AFT model

```

stancode_ttgtle = "

```



```

functions{
// defines the log survival
vector log_S (vector t,real shape1,real shape2,vector inversescale){
vector[num_elements(t)] log_S ;
for (i in 1:num_elements(t)){
log_S[i] = log(((1-((exponential_cdf(t[i],
inversescale[i]))^(2*shape2)))^(shape1)))));
}
return log_S;
}
//defines the log hazard
vector log_h (vector t,real shape1,real shape2,vector inversescale){
vector[num_elements(t)] log_h ;
vector[num_elements(t)] ls ;
ls = log_S(t,shape1,shape2,inversescale) ;
for (i in 1:num_elements(t)){
log_h[i] = (log(2)+log(shape1)+log(shape2)+
exponential_lpdf(t[i]|inversescale[i])+
(((2*shape2)-1)*exponential_lcdf
(t[i]|inversescale[i]))+
((shape1-1)*(log(1-(exponential_cdf
(t[i],inversescale[i]))^(2*shape2)))))) - ls[i];
}
return log_h;
}
//defines the log-likelihood for right censored data
real surv_ttgtle_lpdf(vector t,vector d,real shape1,
real shape2,vector inversescale){
vector[num_elements(t)] log_lik;
real prob;
log_lik = d .* log_h(t,shape1,shape2,inversescale)+
log_S(t,shape1,shape2,inversescale);
prob = sum(log_lik);
return prob;
}
}
//data block
data{
int N; // number of observations
vector <lower=0> [N] y;// observed times
vector <lower=0,upper=1> [N] event;//censoring(1=obs.,
// 0=cens.)
int M; // number of covariates
matrix[N,M] x;//model matrix (N rows, M columns)
}
//parameters block
parameters{
vector [M] beta;//coef.in the linear predictor
real<lower=0> shape1;// shape parameter
real<lower=0> shape2;// shape parameter
}
// transformed parameters block
transformed parameters{

```

```

vector[N] linpred;
vector[N] mu;
linpred = -x*beta; //linear predictor
for (i in 1:N){
mu[i] = exp(linpred[i]);
}
}
// model block
model{
shape1 ~ student_t(5,0,10) T[0,]; //prior for shape1
shape2 ~ student_t(5,0,10) T[0,]; //prior for shape2
beta ~ normal(0,10); //prior for reg. coefficients
y ~ surv_ttgtle(event,shape1,shape2,mu);
//model for the data
}
// generated quantities block
generated quantities{
vector[N] y_rep; //posterior predictive value
vector[N] log_lik; //log-likelihood
{ for(n in 1:N){
log_lik[n] = ((log(2)+log(shape1)+log(shape2)+
exponential_lpdf(y[n] | exp(-(x[n,]*beta)))+
(((2*shape2)-1)*exponential_lcdf(y[n] | exp(-(x[n,]*
beta)))))+((shape1-1)*
(log(1-(exponential_cdf(y[n],exp(-(x[n,]*
beta))))))^
(2*shape2))))-(log(((1-((exponential_cdf(y[n],
exp(-(x[n,]*beta))))^(2*shape2))))^(shape1))))*event[n])+
(log(((1-((exponential_cdf(y[n],
exp(-(x[n,]*beta))))^(2*shape2))))^(shape1)))));}
}
{real u;
u=uniform_rng(0,1);
for (n in 1:N){
y_rep[n] = (exp(x[n,]*beta))*(-log(1-(1-((1-u)^(1/shape1)))
^(1/(2*shape2)))));}
}
}
"

```

5.4.3 Stan code for TIIGTL-LL AFT model

```

stancode_ttgtlll = "
functions{
// defines the log survival
vector log_S (vector t,real shape1,real shape2,
real shape3,vector scale){
vector[num_elements(t)] log_S ;
for (i in 1:num_elements(t)){
log_S[i] = log((1-(((1+(t[i]/scale[i]))^(-shape3))^(-1))^(2*shape2)))
^(shape1));
}
return log_S;
}
}

```

```

//defines the log hazard
vector log_h (vector t,real shape1,real shape2,
real shape3,vector scale){
vector[num_elements(t)] log_h ;
vector[num_elements(t)] ls ;
ls = log_S(t,shape1,shape2,shape3,scale) ;
for (i in 1:num_elements(t)){
log_h[i] = (log(2)+log(shape1)+log(shape2)+
(log(shape3)-(shape3)*log(scale[i]))+(shape3-1)*
log(t[i])-2*log(1+(t[i]/scale[i])^(shape3))))+
(((2*shape2)-1)*log(((1+(t[i]/scale[i])^(-shape3))^(-1))))+
((shape1-1)*(log(1-((1+(t[i]/scale[i])^(-shape3))^(-1))^(2*shape2))))))
-ls[i];
}
return log_h;
}

//defines the log-likelihood for right censored data
real surv_ttgtil1_lpdf(vector t,vector d,real shape1,
real shape2,real shape3,vector scale){
vector[num_elements(t)] log_lik;
real prob;
log_lik = d .* log_h(t,shape1,shape2,shape3,scale)+
log_S(t,shape1,shape2,shape3,scale);
prob = sum(log_lik);
return prob;
}
}

//data block
data{
int N; // number of observations
vector <lower=0> [N] y;// observed times
vector <lower=0,upper=1> [N] event;//censoring(1=obs.,
// 0=cens.)
int M; // number of covariates
matrix[N,M] x;//model matrix (N rows, M columns)
}

//parameters block
parameters{
vector [M] beta;//coef.in the linear predictor
real<lower=0> shape1;// shape parameter
real<lower=0> shape2;// shape parameter
real<lower=0> sigma;//scale parameter sigma=1/shape3
}

// transformed parameters block
transformed parameters{
vector[N] linpred;
vector[N] mu;
linpred = x*beta; //linear predictor
for (i in 1:N){
mu[i] = exp(linpred[i]);
}
}

// model block

```

```

model{
shape1 ~ student_t(5,0,10) T[0, ];//prior for shape1
shape2 ~ student_t(5,0,10) T[0, ];//prior for shape2
sigma ~ student_t(2,0,10) T[0, ];//prior for sigma
beta ~ normal(0,10);//prior for reg. coefficients
y ~ surv_ttgttl1(event,shape1,shape2,1/sigma,mu);
//model for the data
}
// generated quantities block
generated quantities{
vector[N] y_rep;//posterior predictive value
vector[N] log_lik;//log-likelihood
{ for(n in 1:N){
log_lik[n] = ((log(2)+log(shape1)+log(shape2)+(log(1/sigma)-
(1/sigma)*(x[n,]*beta)+((1/sigma)-1)*log(y[n]))-
2*log(1+(y[n]/exp(x[n,]*beta))^(1/sigma)))+
(((2*shape2)-1)*log(((1+(y[n]/exp(x[n,]*beta))^(1/sigma))^(1/sigma)))+
((shape1-1)*(log(1-(((1+(y[n]/exp(x[n,]*beta))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma)))-
(log(((1-(((1+(y[n]/exp(x[n,]*beta))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma)))*event[n]))+
(log((1-(((1+(y[n]/exp(x[n,]*beta))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma))^(1/sigma));}
}
{real u;
u=uniform_rng(0,1);
for (n in 1:N){
y_rep[n] = (exp(x[n,]*beta))*((((1-(1-((1-u)^(1/shape1))))^(1/(2*shape2))))
^(-1))-1)^(sigma);}
}
}
"

```

5.5. Model fitting with Stan

The function *stan* from the package **rstan** is used for the fitting of all three models based on TIIGTL-G family. All relevant codes for the numeric as well as graphical summary are attached in upcoming sub sections.

5.5.1 Fitting of TIIGTL-W AFT model

```

require(survival)
betaw = solve(crossprod(x),crossprod(x,log(y)))
betaw = c(betaw)
TTGTLWAFT <- stan(model_code = stancode_ttgtlw,data=datt,
init=list(list(beta=betaw),list(beta=betaw)),iter=5000,chains=2)

```

Output and graphics Summarization: Table 2 contains the results obtained after fitting the TIIGTL-W AFT model to the diet data set. The coefficients β_2 of saturated fat (x_1) and β_3 of unsaturated fat (x_2) are negative which indicate that both x_1 and x_2 expedite the tumor development process, consequently, survival time (time to develop a tumor) will be shorter. From the summary results and from the caterpillar plot (Figure 1b), it is seen that the 95% credible intervals do not contain a value of zero for the coefficients of the diets, so the coefficients are statistically significant. Additionally, we can see the posterior estimates (mean and se_mean), the standard deviation (sd), and the credible interval. Also we can observe the n_eff (rough estimate

of the effective sample size), and the Rhat, also known as the potential scale reduction factor [15], which calculates the Markov chain’s convergence to the target distribution. According to [15] the allowable range of n_{eff} is greater than 100 and Rhat values less than 1.1. We can observe Rhat values for all parameters of the TIIGTL-W AFT model is less than 1.1, this indicates that the Monte Carlo error is tolerable, the Markov chains reach to the target distribution, and the effective sample size is appropriate.. Trace plots are also attached (Figure 1a) as indicator of convergence of MCMC algorithm. Using the **Bayesplot** package, posterior predictive density (PPD) charts are used to visually evaluate the model. Posterior predictive density (Figure 2a) graphs shows that the TIIGTL-W AFT model is consistent with the current data.

Table 2: Summary of Posterior estimates of TIIGTL-W AFT model parameters

parametrs	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	2.825	0.037	1.416	-0.168	0.398	5.402	1440	1.002
beta[2]	-0.390	0.003	0.157	-0.695	-0.648	-0.086	2353	1.000
beta[3]	-0.658	0.004	0.161	-0.980	-0.930	-0.345	2049	1.000
shape1	9.569	0.154	8.365	0.524	0.936	31.259	2953	1.001
shape2	13.829	0.253	10.081	2.275	2.945	40.073	1586	1.001
sigma	2.726	0.025	0.956	1.044	1.241	4.739	1494	1.002

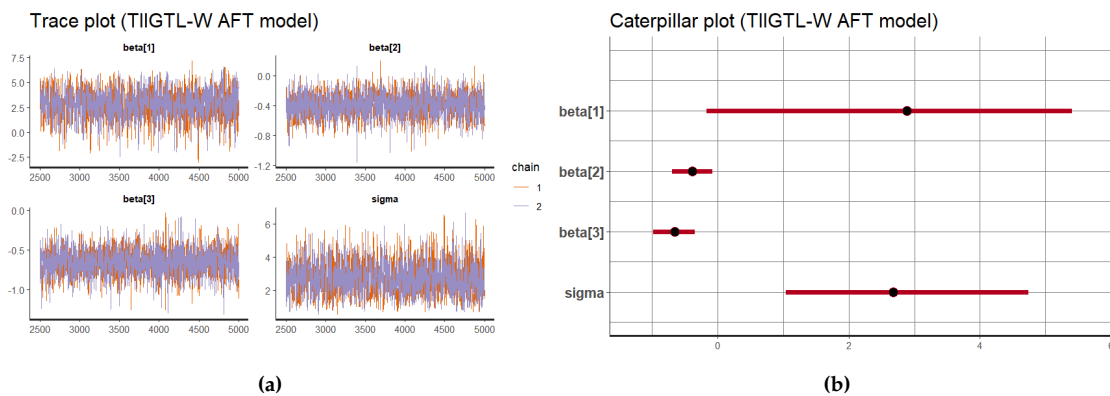


Figure 1: (a) Traceplot for TIIGTL-W AFT model, In two separate runs, two chains were displayed; combining the two chains successfully indicates that MCMC algorithm has converged to the target joint posterior distribution. (b) Caterpillar plot for TIIGTL-W AFT model

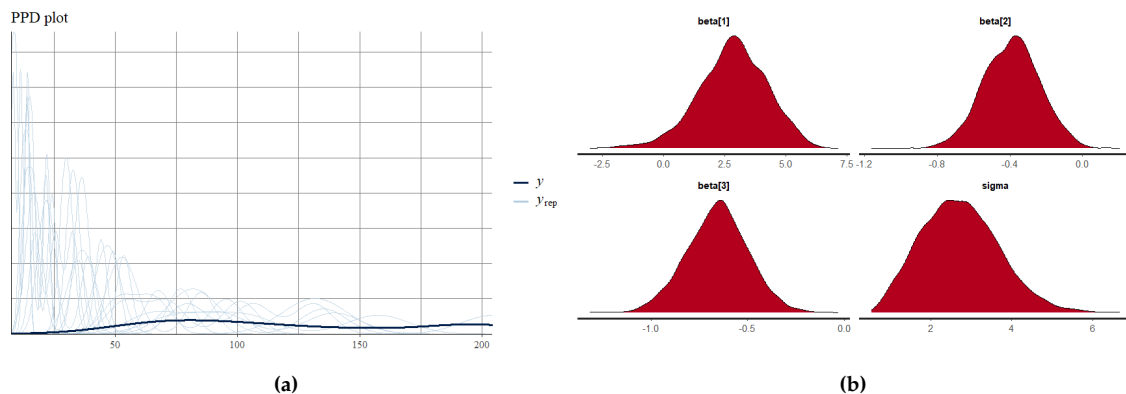


Figure 2: (a) Posterior predictive density (PPD) plot of the TIIGTL-W AFT model to check model convergence. The TIIGTL-W AFT model’s posterior predictive density adequately fits the data, according to the PPD plot (b) Posterior density plot for TIIGTL-W AFT model

5.5.2 Fitting of TIIGTL-E AFT model

```
TTGTLEAFT <- stan(model_code = stancode_ttgtle,data=datt,
init=list(list(beta=betae),list(beta=betae)),iter=5000,chains=2)
```

Output and graphics Summarization: From Table 3 we can observe that the coefficients beta[2] of saturated fat (x1) and beta[3] of unsaturated fat (x2) are negative and the Rhat of the TIIGTL-E AFT model parameters are less than 1.1, which shows Markov chain converges to the target distribution. Also, n_eff is greater than 100. From the caterpillar plot (Figure 3b), it is seen that the 95% credible intervals do not contain a value of zero for the coefficients of the diets, so the coefficients are statistically significant. The PPD plot (Figure 4a) of the TIIGTL-E AFT model indicates that the posterior predictive density matched the data well.

Table 3: Summary of Posterior estimates of TIIGTL-E AFT model parameters

parametrs	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	4.507	0.034	1.000	2.950	3.063	6.360	889	1.001
beta[2]	-0.362	0.004	0.158	-0.671	-0.619	-0.059	1672	1.001
beta[3]	-0.615	0.005	0.185	-0.968	-0.914	-0.245	1231	1.001
shape1	3.206	0.131	5.022	0.168	0.192	18.503	1473	1.000
shape2	4.755	0.163	5.236	1.211	1.296	19.390	1036	1.002

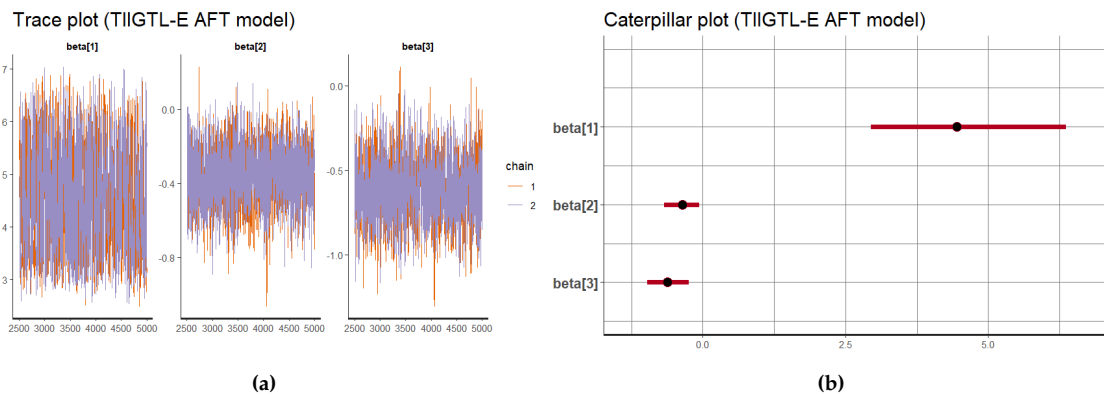


Figure 3: (a) Traceplot for TIIGTL-E AFT model parameters (b) Caterpillar plot for the TIIGTL-E AFT model

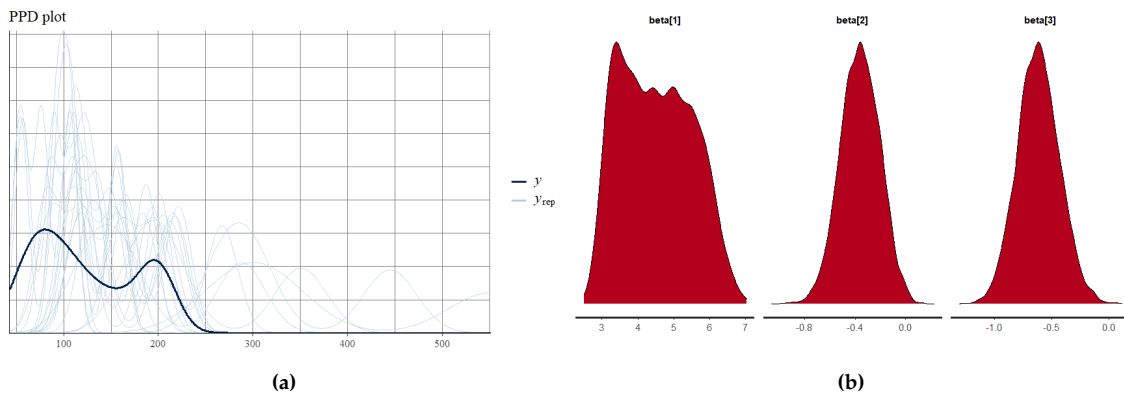


Figure 4: (a) The posterior predictive density (PPD) plot of the TIIGTL-E AFT model (b) Posterior density plot TIIGTL-E AFT model parameters

5.5.3 Fitting of TIIGTL-LL AFT model

```
TIIGTLAFT <- stan(model_code = stancode_ttgctl11,,data=datt,
init=list(list(beta=beta11),list(beta=beta11)),iter=5000,chains=2)
```

Output and Graphics Summarization: From Table 4 we can observe that the coefficients beta[2] of saturated fat (x1) and beta[3] of unsaturated fat (x2) are negative and the Rhat of the TIIGTL-LL AFT model parameters are less than 1.1, which shows Markov chain converges to the target distribution. Also, n_eff is greater than 100. From the caterpillar plot (Figure 5b), it is seen that the 95% credible intervals do not contain a value of zero for the coefficients of the diets, so the coefficients are statistically significant. The PPD plot (Figure 6a) of the TIIGTL-LL AFT model indicates that the posterior predictive density matched the data well.

Table 4: Summary of Posterior estimates of TIIGTL-LL AFT model parameters

parametrs	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	2.951	0.025	1.141	0.713	1.108	5.304	2017	1.000
beta[2]	-0.354	0.004	0.157	-0.674	-0.618	-0.039	1987	1.000
beta[3]	-0.575	0.004	0.172	-0.913	-0.850	-0.234	1651	1.001
shape1	11.323	0.176	8.929	1.188	1.829	34.794	2566	1.000
shape2	11.555	0.167	8.733	1.568	2.175	33.203	2738	1.000
sigma	1.145	0.007	0.309	0.514	0.616	1.758	1713	1.000

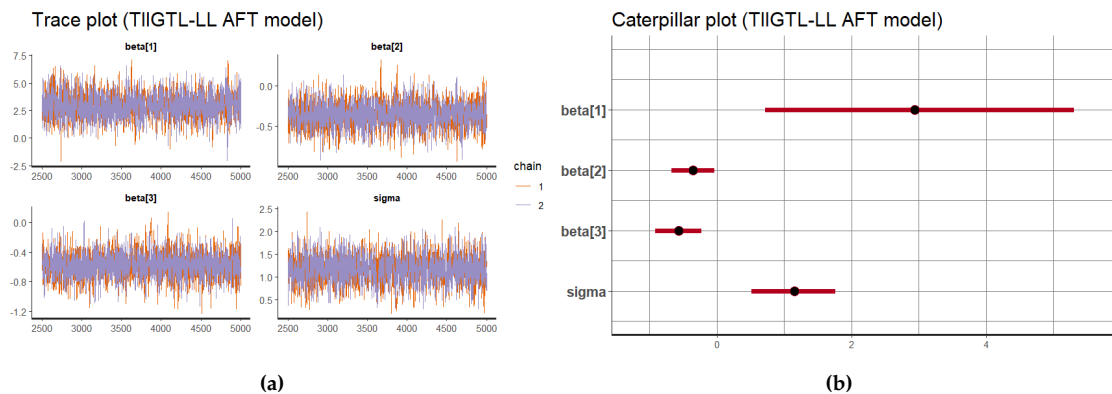


Figure 5: (a) Traceplot of TIIGTL-LL AFT model parameters (b) Caterpillar plot for TIIGTL-LL AFT model

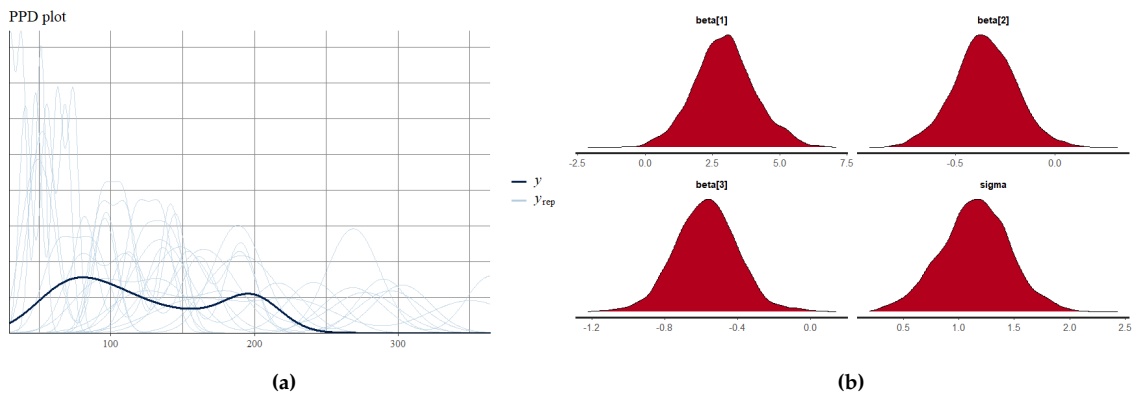


Figure 6: (a) The posterior predictive density (PPD) plot of the TIIGTL-LL AFT model (b) Posterior density plot for TIIGTL-LL AFT model

5.6. Bayesian model Comparison

We take into account model evaluation and selection standards such as Watanabe Akaike Information Criteria (WAIC) and Leave One Out cross-validation Information Criteria (LOOIC) ([16],[17]) in order to compare the fitted models. In R, **loo** package [17] is used to obtain LOOIC and WAIC by using the log-likelihood evaluated at the posterior simulations of the parameters after fitting the model through STAN. The lower value of these selection strategies, however, denotes a better model fit.

Table 5: LOOIC and WAIC values for all models.

Model	LOOIC	WAIC
TIIGTL-E AFT	1026.4	1026.3
TIIGTL-W AFT	1024.5	1024.5
TIIGTL-LL AFT	1015.0	982.8

From Table 5, we can see that the LOOIC and WAIC value of the TIIGTL-LL AFT model is lowest among the three, which shows in comparison to other models for diet data, the TIIGTL-LL AFT model is a superior survival model.

5.7. Conclusion

In a Bayesian framework, the Weibull, Exponential, and Log-Logistic Accelerated Failure Time models for the diet data are fitted using the Type II Generalized Topp–Leone distribution. Diet coefficients for each model have statistical significance. The posterior predictive density (PPD) plots for the TIIGTL-W AFT, TIIGTL-E AFT, and TIIGTL-LL AFT models were used to calculate the posterior predictive check. The replicated data sets are derived from the same model as the original data set, and all are sufficient models for projecting the future value, as seen in the PPD plot where the data y and replicated data set y_{rep} exhibit the same behaviour and share a similar appearance. TIIGTL-LL AFT model fits the censored diet data better than the other models, according to comparisons of posterior predictive density plots, LOOIC and WAIC.

REFERENCES

- [1] Hassan, A. S., Elgarhy, M., and Ahmad, Z. (2019). Type ii generalized topp-leone family of distributions: Properties and applications. *Journal of data science*, 17(4).
- [2] AbuJarad, M. H. and Khan, A. A. (2018). Bayesian survival analysis of topp-leone generalized family with stan. *Int J Stat Appl*, 8(5):274-290.
- [3] Ashraf-Ul-Alam, M. and Khan, A. A. (2021b). Generalized topp-leone-weibull aft modelling: A bayesian analysis with mcmc tools using r and stan. *Austrian Journal of Statistics*, 50(5):52-76.
- [4] Ashraf-Ul-Alam, M. and Khan, A. A. (2021a). Comparison of accelerated failure time models: A bayesian study on head and neck cancer data. *Journal of Statistics Applications & Probability*, 10(3):715-738.
- [5] Stan, V. (2017). *Stan modeling language: User’s guide and reference manual*. Stan Development Team, mc-stan. org.
- [6] Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics letters B*, 195(2):216-222.
- [7] Neal, R. M., Brooks, S., Gelman, A., Jones, G., Meng, X., et al. (2011). *Handbook of markov chain monte carlo*. Press C, editor, 22011.
- [8] Hoffman, M. D., Gelman, A., et al. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593-1623.
- [9] Vehtari, A., Gelman, A., and Gabry, J. (2017). Practical bayesian model evaluation using leave-one-out cross-validation and waic. *Statistics and computing*, 27(5):1413-1432.

- [10] Saikia, R. and Barman, M. P. (2017). A review on accelerated failure time models. *Int J Stat Syst*, 12(2):311-322.
- [11] Kalbfleisch, J. D. and Prentice, R. L. (2011). *The statistical analysis of failure time data*. John Wiley & Sons.
- [12] Lawless, J. F. (2011). *Statistical models and methods for lifetime data*. John Wiley & Sons.
- [13] Lee, E. T. and Wang, J. W. (2013). *Statistical Methods for Survival Data Analysis*. John Wiley & Sons.
- [14] Collett, D. (2015). *Modelling survival data in medical research*. CRC press.
- [15] Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). Models for robust inference. *Bayesian Data Analysis*. 3rd edn, Chapman & Hall/CRC, Boca Raton, FL, pages 435-446.
- [16] Watanabe, S. and Opper, M. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of machine learning research*, 11(12).
- [17] Vehtari, A., Gabry, J., Yao, Y., and Gelman, A. (2018). loo: Efficient leave-one-out cross-validation and waic for bayesian models. *R package version*, 2(0):1003.