

Bayesian Survival Modeling of Marshal Olkin Generalized-G family with random effects using R and STAN

SHAZIA FARHIN, FIRDOOS YOUSUF AND ATHAR ALI KHAN

•
Department of Statistics and Operations Research
Aligarh Muslim University, Aligarh-202002, India
farhinshazia@gmail.com
firdoos1990@gmail.com
atharkhan1962@gmail.com

Abstract

The purpose of this paper is to fit the Marshall-Olkin generalized-G(MOG-G) family to censored survival data with random effect in the Bayesian environment. Three special distribution based on MOG-G family are obtained, namely Marshall-Olkin generalized-exponential, Marshall-Olkin generalized-Weibull, and Marshall-Olkin generalized-Lomax. The probabilistic programming language STAN is used for the fitting of these three distribution to the survival data. STAN offers full Bayesian inference and implements via Hamiltonian Monte Carlo algorithm and No-U-Turn Sampler(NUTS) algorithm of MCMC. We compared the models with the help of leave one out cross-validation information criteria and Watanabe Akaike information criteria. Stan codes for the analysis are provided.

Keywords: Bayesian modeling, Marshall-Olkin generalized-G family, censored survival data, random effect, Leave one out information criteria, STAN

1. INTRODUCTION

In the survival analysis, researchers are using the extended version of standard distribution to analyze the lifetime data and problems related to the modeling of the aging or failure process. In this paper, we have used the Marshall-Olkin generalized-G (MOG-G) family to fit censored survival data, including the random effect. [1] proposed the MOG-G family and studied its mathematical properties along with application in the fitting of lifetime data. The Marshall Olkin distribution has been extended by using the genesis of other distributions to create a wider family of distribution. see for example, Marshall-Olkin-G family [2], Kumaraswamy marshal-Olkin family [3], beta Marshall-Olkin family [4], Beta Generalized Marshall-Olkin-G family [5], Exponentiated Marshall-Olkin family [6], The generalized Marshall-Olkin-Kumaraswamy-G family [7], The Beta generalized Marshall-Olkin Kumaraswamy-G [8], The exponentiated generalized Marshall-Olkin family [9], The Weibull Marshall-Olkin family [10].

We have considered the three models based on the MOG-G family and are fitted to the survival data. The first model is Marshall-Olkin Generalized-Exponential(MOG-E), the second is Marshall-Olkin Generalized-Weibull (MOG-W) model, and the third one is Marshall-Olkin Generalized-Lomax (MOG-L) model. The data with random effect significantly affects the distribution of the patients' survival time and accounts heterogeneity among the patients. Fitting a large number of random effects in a non-Bayesian setting requires a large amount of data. Often, the data is too small to estimate random-effects parameters reliably. However, Bayesian modeling

can be used if there is not enough data for inferential statistics. So, the above three models have been fitted to the censored survival data under the Bayesian setup in R [11] using the probabilistic programming language STAN [12], which offers full Bayesian inference. STAN uses Hamiltonian Monte Carlo (HMC) sampling [13],[14] and its extension. No-U-Turn Sampler(NUTS) [15] algorithm of MCMC for the simulation and computation of posterior estimate. HMC is a more efficient and sophisticated MCMC algorithm, and it is the combination of MCMC and deterministic simulation methods. To find the region of posterior distribution with high mass, HMC uses the gradient of the log posterior density. After that, it jumps around the posterior distribution [16]. Whether the priors are conjugate or not, the above algorithms converge at a fast rate to high dimensional target distributions as compared to other algorithms of MCMC [15].

The purpose of this paper is to fit the three models, namely MOG-E, MOG-W, and MOG-L, to the censored survival data containing random effects under the Bayesian environment using the R and STAN and select the best model for the real survival data.

2. MARSHAL-OLKIN GENERALIZED-G FAMILY

Suppose that $G(t, \psi)$ and $g(t, \psi)$ be baseline cdf and pdf of a continuous random variable T with parameter vector ψ . The cdf, pdf, survival function, and hazard function of the MOG-G family are respectively given by

$$F(t, a, \alpha, \psi) = \frac{1 - [1 - G(t, \psi)]^a}{1 - (1 - \alpha)[1 - G(t, \psi)]^a}, \quad t \in R \quad (1)$$

$$f(t, a, \alpha, \psi) = \frac{\alpha a g(t, \psi) [1 - G(t, \psi)]^{a-1}}{[1 - (1 - \alpha)[1 - G(t, \psi)]^a]^2}, \quad t \in R \quad (2)$$

$$S(t, a, \alpha, \psi) = \frac{\alpha [1 - G(t, \psi)]^a}{1 - (1 - \alpha)[1 - G(t, \psi)]^a}, \quad t \in R \quad (3)$$

$$h(t, a, \alpha, \psi) = \frac{a g(t, \psi) [G(t, \psi)]^{-1}}{1 - (1 - \alpha)[1 - G(t, \psi)]^a}, \quad t \in R \quad (4)$$

Hence forth a random variable T with pdf (2) is denoted by $T \sim \text{MOG-G}(\alpha, a, \psi)$, where α and a are two positive shape parameter.

2.1. Marshall-Olkin Generalized Exponential model

Consider T as a continous random variable follow an exponential distribution with scale parameter $\lambda > 0$, whose pdf and cdf is given by $g(t) = \frac{1}{\lambda} e^{-\frac{t}{\lambda}}$ and $G(t) = 1 - e^{-\frac{t}{\lambda}}$, $t > 0$. Then the pdf and cdf of MOG-E model are respectively given by

$$f(t) = \frac{\alpha a \frac{1}{\lambda} \exp(-a \frac{t}{\lambda})}{[1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})]^2} \quad (5)$$

$$F(t) = \frac{1 - \exp(-a \frac{t}{\lambda})}{1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})} \quad (6)$$

The survival function corresponding to Equation (6) is given as

$$S(t) = \frac{\alpha \exp(-a \frac{t}{\lambda})}{1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})} \quad (7)$$

Hazard function of the MOG-E model is written as

$$h(t) = \frac{a \frac{1}{\lambda}}{1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})} \quad (8)$$

In survival analysis, random generation of time variable from a survival model is done by putting $u = S(t)$, where U is a random variable follow $Uniform(0, 1)$. So, the generation of time variable from MOG-E model is obtained by

$$t = \frac{\lambda}{a} \log \left(\frac{\alpha}{u} + (1 - \alpha) \right) \quad (9)$$

Following the [17], the joint likelihood function for right censored data is given as

$$L = \prod_{i=0}^n Pr(t_i, \delta_i) = \prod_{i=0}^n \{h(t_i)\}^{\delta_i} S(t_i) \quad (10)$$

here δ_i is an indicator variable

$$\delta_i = \begin{cases} 0, & \text{censored} \\ 1, & \text{observed} \end{cases}$$

The likelihood function for the MOG-E survival model is given by

$$L = \prod_{i=0}^n \left\{ \frac{a \frac{1}{\lambda}}{1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})} \right\}^{\delta_i} \times \frac{\alpha \exp(-a \frac{t}{\lambda})}{1 - (1 - \alpha) \exp(-a \frac{t}{\lambda})} \quad (11)$$

2.2. Marshall-Olkin Generalized Weibull model

Let $g(t)$ and $G(t)$ be the pdf and cdf of Weibull distribution with shape parameter $\gamma > 0$ and scale parameter $\lambda > 0$. Where, $g(t) = \frac{\gamma}{\lambda^\gamma} t^{\gamma-1} e^{-(\frac{t}{\lambda})^\gamma}$ and $G(t) = 1 - e^{-(\frac{t}{\lambda})^\gamma}$, $t > 0$. Then the pdf of MOG-W model is given by

$$f(t) = \frac{\alpha a \gamma \frac{1}{\lambda} t^{\gamma-1} \exp(-a(\frac{t}{\lambda})^\gamma)}{[1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)]^2} \quad (12)$$

Therefore, random variable T is denoted by $T \sim \text{MOG-W}(\alpha, a, \gamma, \lambda)$. The cdf of MOG-W model is written as

$$F(t) = \frac{1 - \exp(-a(\frac{t}{\lambda})^\gamma)}{1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)} \quad (13)$$

Survival function and hazard function of the MOG-W model are given respectively

$$S(t) = \frac{\alpha \exp(-a(\frac{t}{\lambda})^\gamma)}{1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)} \quad (14)$$

$$h(t) = \frac{a \gamma \frac{1}{\lambda} t^{\gamma-1}}{1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)} \quad (15)$$

Random generation from the MOG-W model is done by the expression given below

$$t = \lambda \left[\frac{1}{a} \log \left(\frac{\alpha}{u} + (1 - \alpha) \right) \right]^{\frac{1}{\gamma}} \quad (16)$$

Using the Equation (10), the joint likelihood function for the MOG-W model based on right censored is written as

$$L = \prod_{i=0}^n \left\{ \frac{a \gamma \frac{1}{\lambda} t^{\gamma-1}}{1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)} \right\}^{\delta_i} \times \frac{\alpha \exp(-a(\frac{t}{\lambda})^\gamma)}{1 - (1 - \alpha) \exp(-a(\frac{t}{\lambda})^\gamma)} \quad (17)$$

2.3. Marshall-Olkin Generalized Lomax model

Taking Lomax distribution with parameters $\gamma > 0$ and $\lambda > 0$ having pdf $g(t) = \frac{\gamma}{\lambda} (1 + \frac{t}{\lambda})^{-(\gamma+1)}$ and cdf $G(t) = 1 - (1 + \frac{t}{\lambda})^{-\gamma}$, $t > 0$. Then the pdf and cdf of a random variable $T \sim \text{MOG-L}(\alpha, a, \gamma, \lambda)$ model are given respectively

$$f(t) = \frac{\alpha a \frac{\gamma}{\lambda} (1 + \frac{t}{\lambda})^{-1} (1 + \frac{t}{\lambda})^{-a\gamma}}{\left[1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}\right]^2} \quad (18)$$

$$F(t) = \frac{1 - (1 + \frac{t}{\lambda})^{-a\gamma}}{1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}} \quad (19)$$

Survival function of the MOG-L is given by

$$S(t) = \frac{\alpha (1 + \frac{t}{\lambda})^{-a\gamma}}{1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}} \quad (20)$$

Hazard function of the MOG-L is written as

$$h(t) = \frac{\alpha a \frac{\gamma}{\lambda} (1 + \frac{t}{\lambda})^{-1}}{1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}} \quad (21)$$

Generation of survival time from the MOG-L model is given by

$$t = \lambda \left[\left(\frac{\alpha}{u} + (1 - \alpha) \right)^{\frac{1}{a\gamma}} - 1 \right] \quad (22)$$

The joint likelihood function for the MOG-L model is written as

$$L = \prod_{i=0}^n \left\{ \frac{\alpha a \frac{\gamma}{\lambda} (1 + \frac{t}{\lambda})^{-1}}{1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}} \right\}^{\delta_i} \times \frac{\alpha (1 + \frac{t}{\lambda})^{-a\gamma}}{1 - (1 - \alpha) (1 + \frac{t}{\lambda})^{-a\gamma}} \quad (23)$$

3. KIDNEY CATHETER DATA

This dataset, originally discussed in [18]. The study concerns with the recurrence times to infection, at the point where the catheter is inserted, for kidney patients using portable dialysis equipment. The data consist of times until the first and second recurrence of kidney infection in 38 patients. Each patient has exactly two observations. Each survival time is the time until infection since the insertion of the catheter. A Catheter may be removed for reasons other than infection, in which case the observation is censored. There are about 24% censored observations in the dataset. This data set has unmeasured or 'random' effect that is an identification code of patients, which accounts heterogeneity among the patients. This data set available in the package survival [19] of R [11].

Description of kidney catheter data variables are given below:

time: time to infection in days

status: event status, 1=infection occurs or 0=censored

age: age in years

sex: 1=male, 2=female

disease: disease type(0=GN, 1=AN, 2=PKD, 3=Other)

id: identification code of the patients

3.1. Construction of data frame in R

Fitting of Bayesian models to the kidney catheter data with `stan` function requires data in a listed form, which we have created as below;

```
require(survival)
data(cancer, package="survival")
head(kidney)
y=kidney$time
x1=kidney$age
x2=kidney$sex
kidney$disease1=as.numeric(kidney$disease)
x3=kidney$GN=as.numeric(kidney$disease1==2)
x4=kidney$AN=as.numeric(kidney$disease1==3)
x5=kidney$PKD=as.numeric(kidney$disease1==4)
x=cbind(1,x1,x2,x3,x4,x5)
N=nrow(x)
M=ncol(x)
J=38
event=kidney$status
Id=as.integer(kidney$id)##identity of subject
datk=list(y=y,x=x,N=N,M=M,event=event,J=J,Id=Id)
```

4. BAYESIAN ANALYSIS OF MOG-G FAMILY

4.1. Prior Specification

For the construction of the Bayesian regression model, we need to specify a prior distribution to the parameters of the model. We have chosen half-Cauchy prior for shape and scale parameters and regularizing prior for regression coefficient.

4.1.1 Half-Cauchy prior distribution

The probability density function of half-Cauchy distribution with scale γ is given by

$$f(x) = \frac{2\gamma}{\pi(x^2 + \gamma^2)}, \quad x > 0, \gamma > 0$$

The mean and variance of half-cauchy distribution does not exist, but its mode is equal to zero. The half-cauchy distribution with scale $\gamma=25$ is nearly flat prior but not completely, the prior distribution that are not completely flat provides enough information for the numerical approximation algorithm to continue to explore the target density, the posterior distribution [20],[21]. [22] support the use half cauchy prior for scale parameter because of its excellent frequentist risk properties, and its sensible behaviour in the presence of sparsity compared to the usual conjugate alternative. [20] have also discussed the points in support of half cauchy prior.

4.1.2 Gaussian prior distribution

The probability density function of Gaussian distribution with mean μ and variance σ^2 is given by

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad -\infty < x < \infty, \sigma > 0, \mu > 0$$

In this paper, we have chosen Gaussian prior with mean 0, and standard deviation 5 for β coefficient as a regularizing prior because this prior prevent a model from getting too excited by the data that slows the rate of over excitement of model and reduce the overfitting of data to the model [23].

4.2. Model Specification

Following the [24] to build a regression model, we have introduced covariates including random intercept through the log link function i.e.

$$\begin{aligned} \log(\lambda_i) &= \beta_1 + w_{[subj_i]} + \beta_2 x_{i1} + \beta_3 x_{i2} + \beta_4 x_{i3} + \beta_5 x_{i4} + \beta_6 x_{i5} \\ \lambda_i &= \exp(\beta_1 + w_{[subj_i]} + \beta_2 x_{i1} + \beta_3 x_{i2} + \beta_4 x_{i3} + \beta_5 x_{i4} + \beta_6 x_{i5}) \end{aligned}$$

or,

$$\lambda_i = \exp(w_{[subj_i]} + x_i \beta)$$

where, $w_{[subj_i]}$ is the variability accounted by subject or patients called as the random intercept, $w \sim N(0, \sigma_w)$, and $\beta \sim N(0, \sigma = 5)$

4.2.1 Posterior density of MOG-E

By using bayes theorem, the joint posterior distribution is given as

$$P(a, \alpha, \beta | X, t) \propto L(t | a, \alpha, \beta, X) \times P(a) \times P(\alpha) \times P(\beta) \quad (24)$$

$$\begin{aligned} P(a, \alpha, \beta | X, t) &\propto \prod_{i=0}^n \left\{ \frac{a \frac{1}{\exp(w_{[subj_i]} + x_i \beta)}}{1 - (1 - \alpha) \exp(-a \frac{t}{\exp(w_{[subj_i]} + x_i \beta)})} \right\}^{\delta_i} \times \frac{\alpha \exp(-a \frac{t}{\exp(w_{[subj_i]} + x_i \beta)})}{1 - (1 - \alpha) \exp(-a \frac{t}{\exp(w_{[subj_i]} + x_i \beta)})} \\ &\times \frac{2 \times 25}{\pi(a^2 + 25^2)} \times \frac{2 \times 25}{\pi(\alpha^2 + 25^2)} \times \frac{1}{\sigma_w \sqrt{2\pi}} \exp\left(-\frac{w_i^2}{2\sigma_w^2}\right) \times \prod_{j=0}^J \frac{1}{5\sqrt{2\pi}} \exp\left(-\frac{1}{2 \times 25} \beta_j^2\right) \end{aligned} \quad (25)$$

4.2.2 Posterior density of MOG-W

By using bayes theorem, the joint posterior distribution is given as

$$P(a, \alpha, \gamma, \beta | X, t) \propto L(t | a, \alpha, \gamma, \beta, X) \times P(a) \times P(\alpha) \times P(\gamma) \times P(\beta) \quad (26)$$

$$\begin{aligned} P(a, \alpha, \gamma, \beta | X, t) &\propto \prod_{i=0}^n \left\{ \frac{a \gamma \frac{1}{\exp(w_{[subj_i]} + x_i \beta)} t^{\gamma-1}}{1 - (1 - \alpha) \exp(-a (\frac{t}{\exp(w_{[subj_i]} + x_i \beta)})^\gamma)} \right\}^{\delta_i} \\ &\times \frac{\alpha \exp(-a (\frac{t}{\exp(w_{[subj_i]} + x_i \beta)})^\gamma)}{1 - (1 - \alpha) \exp(-a (\frac{t}{\exp(w_{[subj_i]} + x_i \beta)})^\gamma)} \times \frac{2 \times 25}{\pi(a^2 + 25^2)} \times \frac{2 \times 25}{\pi(\alpha^2 + 25^2)} \\ &\times \frac{2 \times 25}{\pi(\gamma^2 + 25^2)} \times \frac{1}{\sigma_w \sqrt{2\pi}} \exp\left(-\frac{w_i^2}{2\sigma_w^2}\right) \times \prod_{j=0}^J \frac{1}{5\sqrt{2\pi}} \exp\left(-\frac{1}{2 \times 25} \beta_j^2\right) \end{aligned} \quad (27)$$

4.2.3 Posterior density of MOG-L

By using bayes theorem, the joint posterior distribution is given as

$$P(a, \alpha, \gamma, \beta | X, t) \propto L(t | a, \alpha, \gamma, \beta, X) \times P(a) \times P(\alpha) \times P(\gamma) \times P(\beta) \quad (28)$$

$$\begin{aligned}
 P(a, \alpha, \gamma, \beta | X, t) &\propto \prod_{i=0}^n \left\{ \frac{\alpha a \frac{\gamma}{\exp(w_{[subj_i]} + x_i \beta)} \left(1 + \frac{t}{\exp(w_{[subj_i]} + x_i \beta)}\right)^{-1}}{1 - (1 - \alpha) \left(1 + \frac{t}{\exp(w_{[subj_i]} + x_i \beta)}\right)^{-a\gamma}} \right\}^{\delta_i} \\
 &\times \frac{\alpha \left(1 + \frac{t}{\exp(w_{[subj_i]} + x_i \beta)}\right)^{-a\gamma}}{1 - (1 - \alpha) \left(1 + \frac{t}{\exp(w_{[subj_i]} + x_i \beta)}\right)^{-a\gamma}} \times \frac{2 \times 25}{\pi(a^2 + 25^2)} \times \frac{2 \times 25}{\pi(\alpha^2 + 25^2)} \\
 &\times \frac{2 \times 25}{\pi(\gamma^2 + 25^2)} \times \frac{1}{\sigma_w \sqrt{2\pi}} \exp\left(-\frac{w_i^2}{2\sigma_w^2}\right) \times \prod_{j=0}^J \frac{1}{5\sqrt{2\pi}} \exp\left(-\frac{1}{2 \times 25} \beta_j^2\right)
 \end{aligned} \tag{29}$$

4.3. Implementation using Stan

Bayesian modeling of MOG-G family in STAN language includes the creation of blocks: functions block, data block, transformed data block, parameters block, transformed parameters block, model block, and generated quantities block. To run STAN code in R requires package rstan that is an interface of R and STAN.

4.3.1 Stan code for MOG-E model

```

modelMOGE="functions{
vector log_moegs(vector t, real a, real alpha, vector lambda){
vector[num_elements(t)]log_moegs;
for(i in 1:num_elements(t)){
log_moegs[i]=log(alpha)-a*t[i]/lambda[i]-log(1-(1-alpha)*exp(-a*t[i]/lambda[i]));
}
return log_moegs;
}
vector log_moegh(vector t, real a, real alpha, vector lambda){
vector[num_elements(t)]log_moegh;
for(i in 1:num_elements(t)){
log_moegh[i]=log(a)-log(lambda[i])-log(1-(1-alpha)*exp(-a*t[i]/lambda[i]));
}
return log_moegh;
}
real surv_MOEG_lpdf(vector t, vector d, real a, real alpha, vector lambda){
vector[num_elements(t)] llikmoeg;
real prob;
llikmoeg=d .* log_moegh(t,a,alpha,lambda)+log_moegs(t,a,alpha,lambda);
prob=sum(llikmoeg);
return prob;
}}
data{
int N;
vector<lower=0>[N] y;
vector<lower=0,upper=1>[N] event;
int M;
matrix[N,M] x;
int<lower=1>J;
int<lower=1,upper=J>Id[N];
}
parameters{

```

```

real<lower=0>a;
vector[M] beta;
real<lower=0> alpha;
vector[J] w;
real<lower=0>sigma_w;
}
transformed parameters{
vector[N] linpred;
vector<lower=0>[N] lambda;
linpred=x*beta;
for(i in 1:N){
lambda[i]=exp(w[Id[i]]+linpred[i]);
}
}
model{
target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=cauchy_lpdf(a|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=normal_lpdf(beta|0,5);
target+=normal_lpdf(w|0,sigma_w);
target+=cauchy_lpdf(sigma_w|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=surv_MOEG_lpdf(y|event,a,alpha,lambda);
}
generated quantities{
vector[N] log_lik;
vector[N] yrepmoeg;
real dev;
dev=0;
for(n in 1:N) log_lik[n]=event[n]*(log(a)-log(lambda[n])-log(1-(1-alpha)*exp(-a*y[n]/
lambda[n]))) +log(alpha)-a*y[n]/lambda[n]-log(1-(1-alpha)*exp(-a*y[n]/lambda[n]));
{real u;
u=uniform_rng(0,1);
for(n in 1:N) yrepmoeg[n]=(lambda[n]/a)*log(alpha/u+(1-alpha));
}
dev=dev+(-2)*surv_MOEG_lpdf(y|event,a,alpha,lambda);
}
}

```

4.3.2 Stan code for MOG-W model

```

modelMOGW="functions{
vector log_mogws(vector t, real a, real alpha,real gamma, vector lambda){
vector[num_elements(t)]log_mogws;
for(i in 1:num_elements(t)){
log_mogws[i]=log(alpha)-a*(t[i]/lambda[i])^(gamma)-log(1-(1-alpha)
*exp(-a*(t[i]/lambda[i])^(gamma)));
}
return log_mogws;
}
vector log_mogwh(vector t, real a, real alpha, real gamma, vector lambda){
vector[num_elements(t)]log_mogwh;
for(i in 1:num_elements(t)){
log_mogwh[i]=log(a)+log(gamma)-gamma*log(lambda[i])+(gamma-1)*log(t[i])
-log(1-(1-alpha)*exp(-a*(t[i]/lambda[i])^(gamma)));
}
return log_mogwh;
}
}

```



```
real surv_MOGW_lpdf(vector t, vector d, real a, real alpha,real gamma,vector lambda){
vector[num_elements(t)] llikmogw;
real prob;
llikmogw=d .* log_mogwh(t,a,alpha,gamma,lambda)+log_mogws(t,a,alpha,gamma,lambda);
prob=sum(llikmogw);
return prob;
}}
data{
int N;
vector<lower=0>[N] y;
vector<lower=0,upper=1>[N] event;
int M;
matrix[N,M] x;
int<lower=1>J;
int<lower=1,upper=J>Id[N];
}
parameters{
real<lower=0>a;
vector[M] beta;
real<lower=0> alpha;
real<lower=0> gamma;
vector[J] w;
real<lower=0>sigma_w;
}
transformed parameters{
vector[N] linpred;
vector<lower=0>[N] lambda;
linpred=x*beta;
for(i in 1:N){
lambda[i]=exp(w[Id[i]]+linpred[i]);
}
}
model{
target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=cauchy_lpdf(a|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=cauchy_lpdf(gamma|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=normal_lpdf(beta|0,5);
target+=normal_lpdf(w|0,sigma_w);
target+=cauchy_lpdf(sigma_w|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=surv_MOGW_lpdf(y|event,a,alpha,gamma,lambda);
}
generated quantities{
vector[N] log_lik;
vector[N] yrepmogw;
real dev;
dev=0;
for(n in 1:N) log_lik[n]=event[n]*(log(a)+log(gamma)-gamma*log(lambda[n])+(gamma-1)*
log(y[n])-log(1-(1-alpha)*exp(-a*(y[n]/lambda[n])^(gamma))))+log(alpha)
-a*(y[n]/lambda[n])^(gamma)-log(1-(1-alpha)*exp(-a*(y[n]/lambda[n])^(gamma)));
{real u;
u=uniform_rng(0,1);
for(n in 1:N) yrepmogw[n]=lambda[n]*((1/a)*log(alpha/u+(1-alpha)))^(1/gamma);
}
dev=dev+(-2)*surv_MOGW_lpdf(y|event,a,alpha,gamma,lambda);
}
```

}"

4.3.3 Stan code for MOG-L model

```
modelMOGL="functions{
vector log_mogls(vector t, real a, real alpha,real gamma, vector lambda){
vector[num_elements(t)]log_mogls;
for(i in 1:num_elements(t)){
log_mogls[i]=log(alpha)-a*gamma*log(1+t[i]/lambda[i])-log(1-(1-alpha)
*(1+t[i]/lambda[i])^(-a*gamma));
}
return log_mogls;
}
vector log_moglh(vector t, real a, real alpha, real gamma, vector lambda){
vector[num_elements(t)]log_moglh;
for(i in 1:num_elements(t)){
log_moglh[i]=log(a)+log(gamma)-log(lambda[i])-log(1+t[i]/lambda[i])-
log(1-(1-alpha)*(1+t[i]/lambda[i])^(-a*gamma));
}
return log_moglh;
}
real surv_MOGL_lpdf(vector t, vector d, real a, real alpha,real gamma,vector lambda){
vector[num_elements(t)] llikmog1;
real prob;
llikmog1=d .* log_moglh(t,a,alpha,gamma,lambda)+log_mogls(t,a,alpha,gamma,lambda);
prob=sum(llikmog1);
return prob;
}}
data{
int N;
vector<lower=0>[N] y;
vector<lower=0,upper=1>[N] event;
int M;
matrix[N,M] x;
int<lower=1>J;
int<lower=1,upper=J>Id[N];
}
parameters{
real<lower=0>a;
vector[M] beta;
real<lower=0> alpha;
real<lower=0> gamma;
vector[J] w;
real<lower=0>sigma_w;
}
transformed parameters{
vector[N] linpred;
vector<lower=0>[N] lambda;
linpred=x*beta;
for(i in 1:N){
lambda[i]=exp(w[Id[i]]+linpred[i]);
}
}
model{
target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
```

```

target+=cauchy_lpdf(a|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=cauchy_lpdf(gamma|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=normal_lpdf(beta|0,5);
target+=normal_lpdf(w|0,sigma_w);
target+=cauchy_lpdf(sigma_w|0,25)- 1 * cauchy_lccdf(0|0,25);
target+=surv_MOGL_lpdf(y|event,a,alpha,gamma,lambda);
}
generated quantities{
vector[N] log_lik;
vector[N] yrepmogl;
real dev;
dev=0;
for(n in 1:N) log_lik[n]=event[n]*(log(a)+log(gamma)-log(lambda[n]))-log(1+y[n]/lambda[n])
-log(1-(1-alpha)*(1+y[n]/lambda[n])^(-a*gamma)))+log(alpha)-a*gamma*log(1+y[n]/lambda[n])
-log(1-(1-alpha)*(1+y[n]/lambda[n])^(-a*gamma));
{real u;
u=uniform_rng(0,1);
for(n in 1:N) yrepmogl[n]=lambda[n]*((alpha/u+(1-alpha))^(1/(a*gamma))-1);
}
dev=dev+(-2)*surv_MOGL_lpdf(y|event,a,alpha,gamma,lambda);
}"
    
```

4.4. Fitting with Stan

To fit the survival models based on MOG-G family, the function `stan` is used, and list `datk` of data pass into the function `stan`. STAN used C++ compiler to samples the posterior distribution of the model parameters, including random intercepts w_j for each patient J . To get summary of result, the function `print` is used.

4.4.1 Fitting of MOG-E model

```

MOGE=stan(model_code = modelMOGE,data=datk,iter=5000,chains = 2)
print(MOGE)
    
```

Summarizing Output: After fitting of MOG-E survival model to the kidney data set, we get the results in tabular form are given in Table 1. It contains posterior estimates, standard deviation, credible interval, n_{eff} (crude estimate of effective sample size), and \hat{R} called as potential scale reduction factor [16], which estimate the convergence of Markov chain to the target distribution. Besides \hat{R} , Traceplot also shows the convergence of the Markov chain. According to [16] the acceptable limit of n_{eff} is >100 and \hat{R} values lower than 1.1. \hat{R} for all parameters of the MOG-E model is close to 1, which means Markov chains converge to the target distribution, the Monte Carlo error is acceptable, and the effective sample size is reasonable. Here, we can see that the posterior estimate of parameters β_1 (Intercept) is 4.440, and β_3 (Sex) is 1.678 are statistically significant as 95% credible interval (CI) does not contains 0 respectively. The positive value of β_3 inferred that the male patients have more chance to get infected at the place where a catheter is inserted than the female patients. The posterior estimate of parameters β_2 (Age) is -0.002, β_4 (AN) is -0.116, β_5 (GN) is -0.544 and β_6 (PKD) is 0.906 are not statistically significant as corresponding CI includes 0.

Table 1: Posterior summary of MOG-E model parameters

parametrs	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	4.440	0.033	1.728	0.833	4.468	7.612	2699	1.001
beta[2]	-0.002	0.000	0.013	-0.029	-0.002	0.025	2584	1.000
beta[3]	1.678	0.007	0.387	0.917	1.678	2.427	3309	1.000
beta[4]	-0.116	0.009	0.483	-1.085	-0.112	0.836	2678	1.000
beta[5]	-0.544	0.009	0.484	-1.533	-0.533	0.397	2777	1.000
beta[6]	0.906	0.014	0.717	-0.493	0.910	2.317	2456	1.000
a	129.344	78.878	2544.490	0.671	18.762	275.779	1041	1.002
alpha	2.825	0.059	2.186	0.534	2.225	8.820	1397	1.001
sigma_w	0.651	0.012	0.221	0.195	0.660	1.091	367	1.003

Graphical Analysis

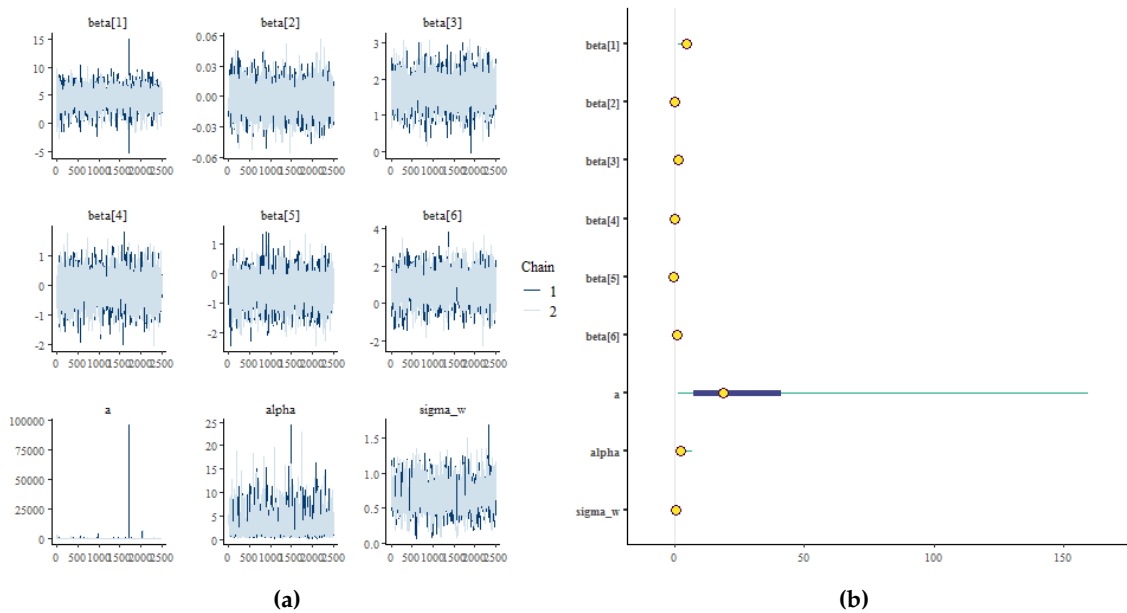


Figure 1: (a) Traceplot of MOG-E model parameters, two chains were run depicted in different color and mixing of two chains is good means Markov chains converge to the target distribution, and (b) Caterpillar plot of the MOG-E model parameters.

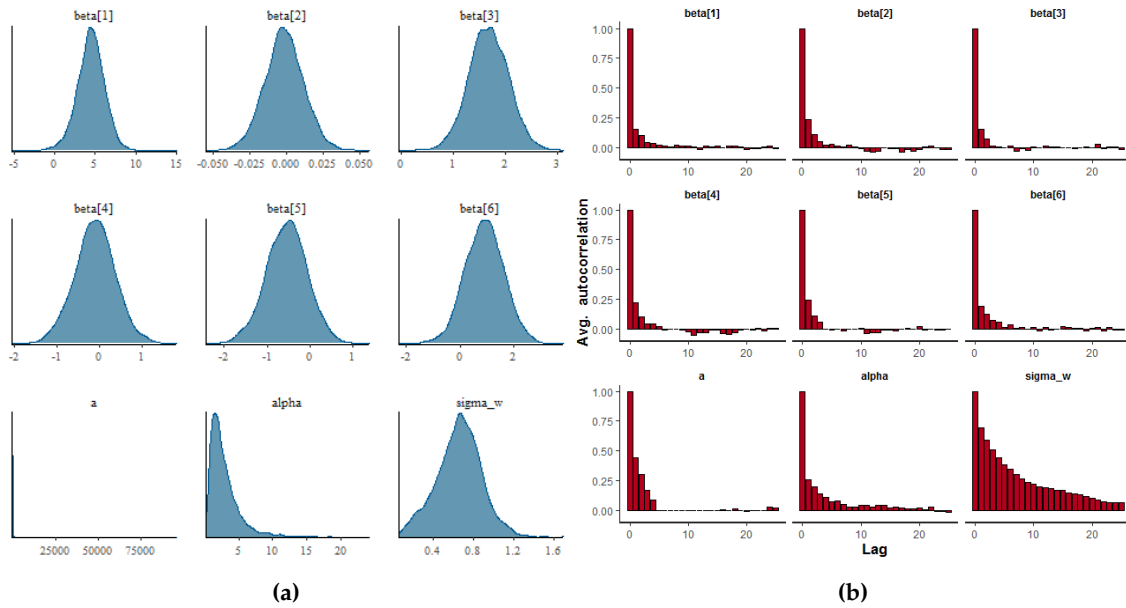


Figure 2: (a) Posterior density plot MOG-E model parameters, (b) Autocorrelation plot of MOG-E model parameters, after 20 lag autocorrelation declining towards zero.

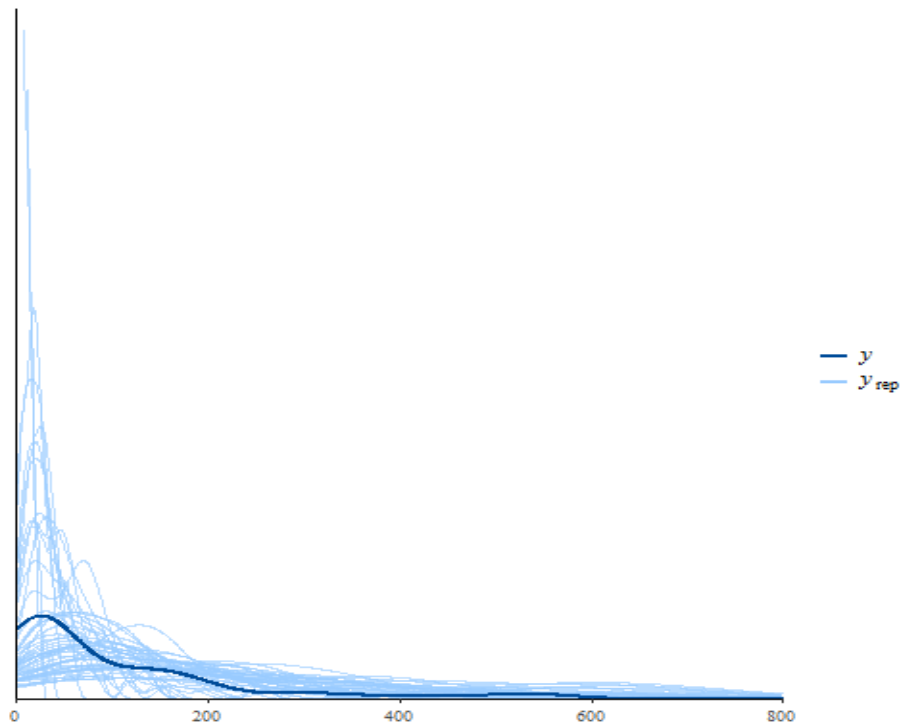


Figure 3: The posterior predictive density (PPD) plot of the MOG-E model is done by plotting the data y and then overlaying the density of the predicted values y_{rep} , which are generated from the posterior predictive distribution of the given model. PPD plot of the MOG-E model shows that the posterior predictive density fits the data well.

4.4.2 Fitting of MOG-W model

```
MOGW=stan(model_code = modelMOGW,data=datk,iter=5000,chains = 2)
```

print(MOGW)

Summarizing Output: It is an evident from Table 2 that the Rhat of the MOG-W model parameters are close to 1, which indicates Markov chain converges to the target distribution and effective sample size is enough to get conversion.

Table 2: Posterior summary of MOG-W model parameters

parametrs	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	4.289	0.053	2.252	-0.936	4.468	8.381	1807	1.000
beta[2]	-0.001	0.000	0.014	-0.028	-0.001	0.027	1892	1.001
beta[3]	1.688	0.007	0.391	0.930	1.686	2.451	2978	1.000
beta[4]	-0.164	0.010	0.479	-1.130	-0.155	0.744	2102	1.002
beta[5]	-0.586	0.011	0.489	-1.582	-0.577	0.350	1935	1.001
beta[6]	0.841	0.020	0.742	-0.625	0.858	2.323	1383	1.002
a	38.413	2.699	169.093	0.786	15.061	185.706	3926	1.000
alpha	72.737	12.350	726.957	0.109	7.801	335.523	3465	1.001
gamma	0.800	0.010	0.357	0.289	0.744	1.562	1332	1.000
sigma_w	0.622	0.013	0.227	0.175	0.630	1.063	288	1.002

Graphical Analysis

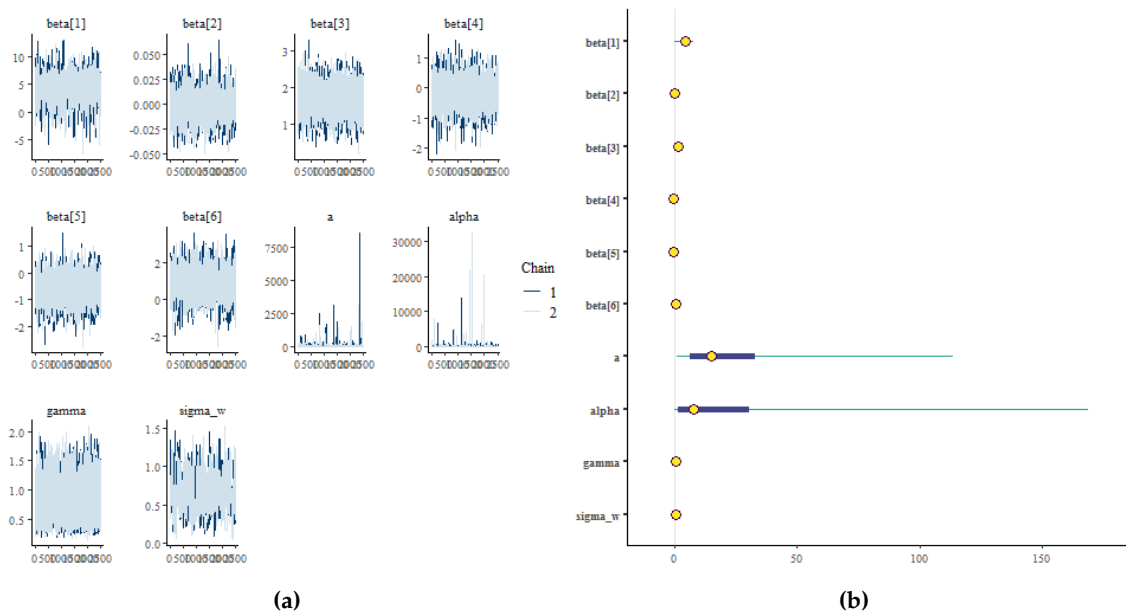


Figure 4: (a) Traceplot of MOG-W model parameters, two chains were run depicted in different color and (b) caterpillar plot of the MOG-W model parameters.

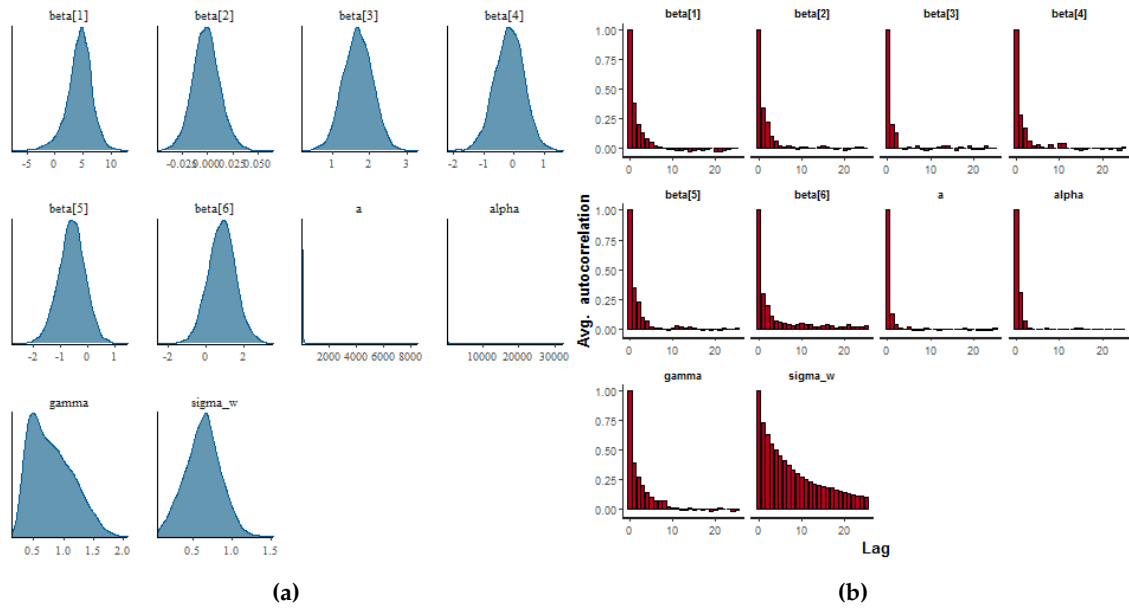


Figure 5: (a) Posterior density plot MOG-W model parameters, (b) Autocorrelation plot of MOG-W model parameters, after 20 lag autocorrelation declining towards zero.

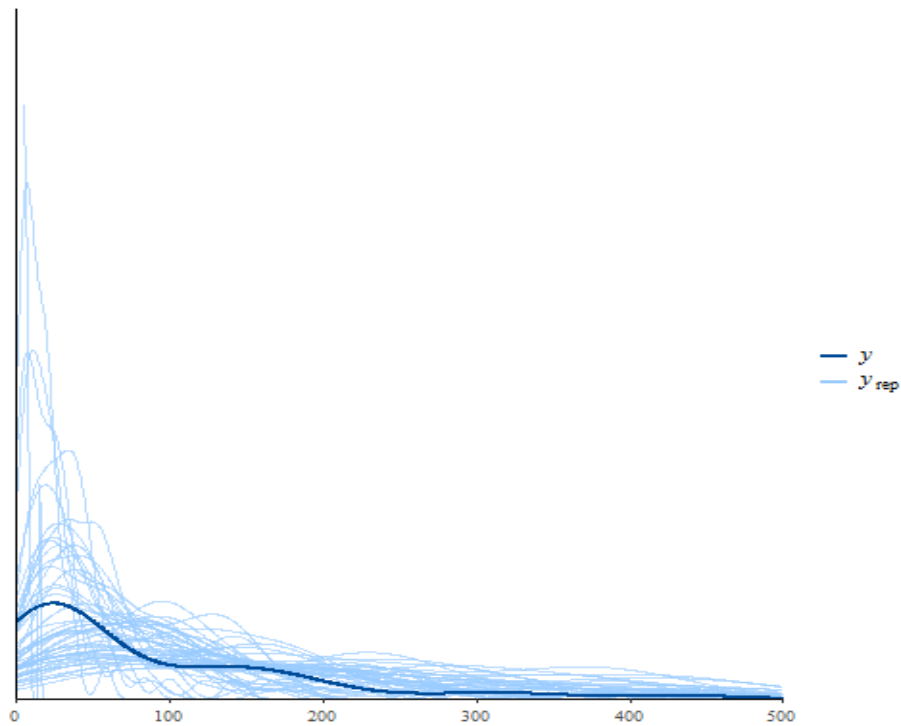


Figure 6: PPD plot of the MOG-W model shows that the posterior predictive density fits the data well and good compatibility of the model to the data.

4.4.3 Fitting of MOG-L model

```
MOGW=stan(model_code = modelMOGW,data=datk,iter=5000,chains = 2)
print(MOGW)
```

Table 3: Posterior summary of MOG-L model parameters

parametr	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
beta[1]	5.180	0.115	3.164	-0.916	5.636	10.534	753	1.012
beta[2]	-0.002	0.000	0.013	-0.027	-0.002	0.025	2500	1.000
beta[3]	1.693	0.007	0.377	0.944	1.700	2.433	3111	1.000
beta[4]	-0.153	0.010	0.494	-1.146	-0.161	0.839	2577	1.000
beta[5]	-0.562	0.009	0.470	-1.478	-0.567	0.394	2478	1.000
beta[6]	0.869	0.014	0.713	-0.530	0.867	2.274	2501	1.000
a	41.617	9.152	641.724	0.119	9.765	182.772	4916	1.000
alpha	11.755	1.010	41.049	0.639	3.193	82.591	1653	1.004
gamma	32.510	2.419	163.682	0.129	10.641	152.084	4580	1.000
sigma_w	0.611	0.014	0.226	0.201	0.612	1.058	253	1.013

Graphical Analysis

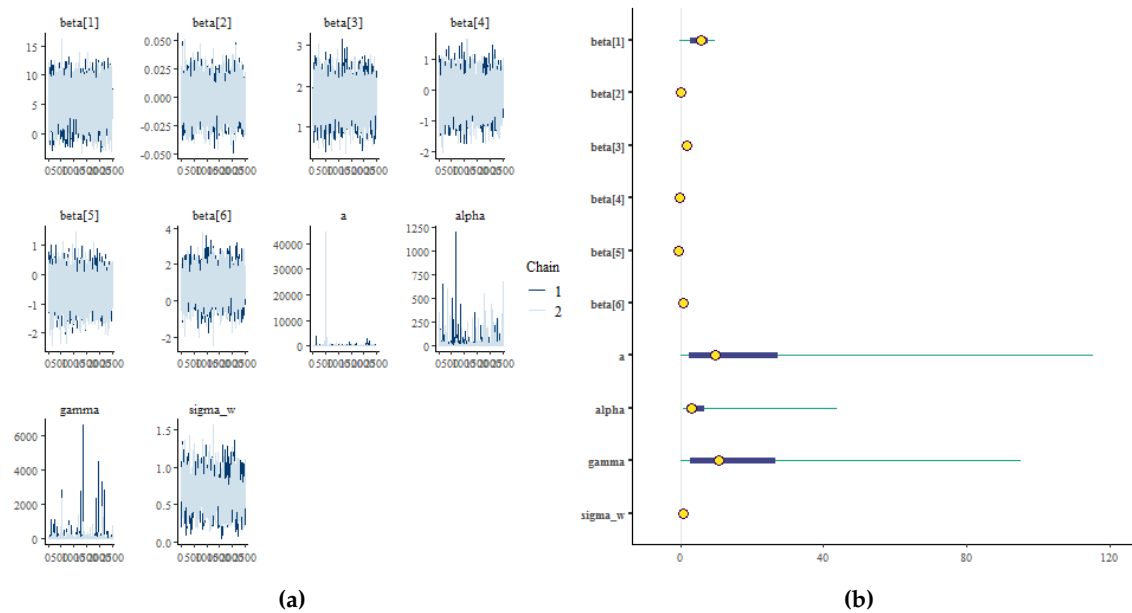


Figure 7: (a) Traceplot of the MOG-L model parameters, two chains were run depicted in different color and (b) caterpillar plot of the MOG-L model parameters.

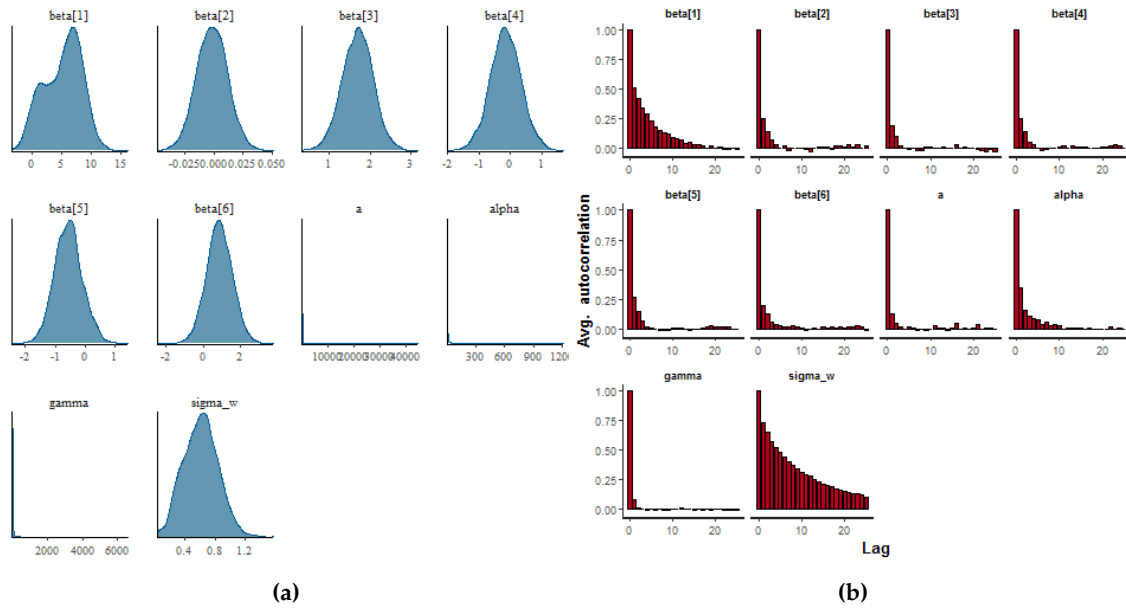


Figure 8: (a) Posterior density plot the MOG-L model parameters, (b) Autocorrelation plot of the MOG-L model parameters, after 20 lag autocorrelation declining towards zero.

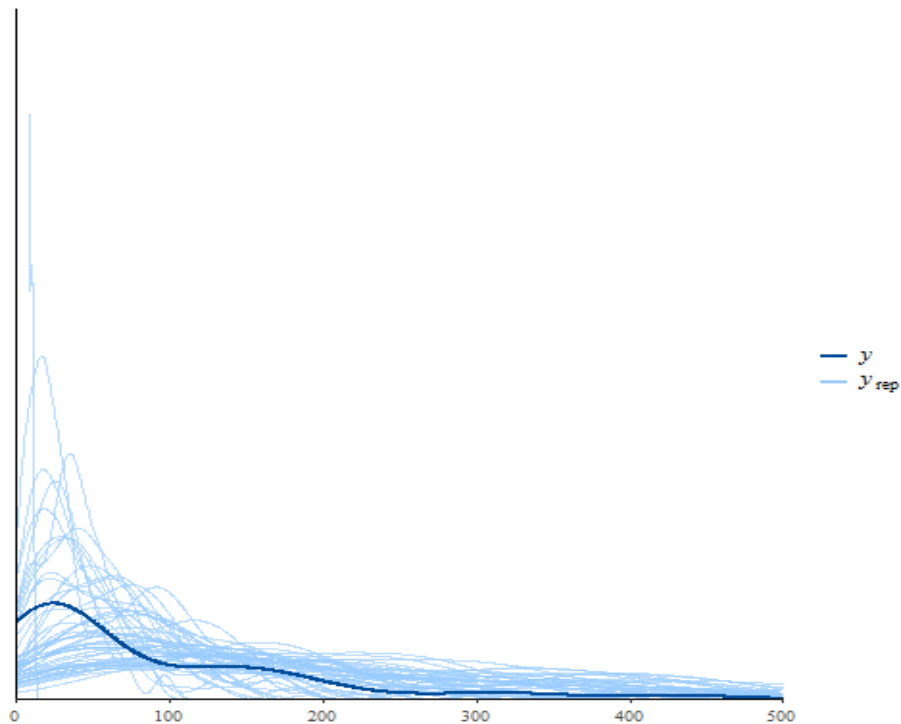


Figure 9: PPD plot of the MOG-L model shows that the posterior predictive density fits the data well, and the model is compatible with the given data.

4.5. Bayesian model Comparison

In order to compare the fitted models, we consider the model selection criteria like Watanabe Akaike information criteria [25] and leave one out cross-validation information criteria (LOOIC). However, the lower value of these selection methods indicates a better model fit. The WAIC and

LOOIC are two fully Bayesian selection methods than the others information criteria. They are methods for estimating pointwise out of sample prediction accuracy from a fitted Bayesian model using the log-likelihood evaluated at the posterior simulations of the parameters [16]. Although WAIC is asymptotically equal to LOOIC, it is preferable to use LOOIC because of its robustness in finite cases with weak priors or influential observation.

Table 4: WAIC and LOOIC values for all models.

Model	No of parameters	WAIC	LOOIC
MOG-E	3	667.9	673.3
MOG-W	4	671.1	675.8
MOG-L	4	670.5	674.0

From Table 4, it is evident that the value of WAIC and LOOIC of the MOG-E model is less than the MOG-W and MOG-L, which means the MOG-E model is a better survival model as compared to other models for Kidney catheter data.

5. DISCUSSION AND CONCLUSION

In this paper, the MOG-G family are fitted to the real survival data includes random effect in Bayesian setup, which is implemented by STAN language using package `rstan` of R. For all models, the Markov chains converges to the target distribution, and covariate Sex is significant. The Posterior predictive check has been computed using the posterior predictive density plot for the MOG-E, MOG-W, and MOG-L models. We have seen in the PPD plot, the data y and replicated data set y^{rep} are showing the same behavior and looks similar which means the replicated data sets are coming from the same model as the given data set, and all are the adequate model for predicting the future value. Upon comparison with the results obtained through LOOIC and WAIC, it can be concluded that the MOG-E model fits the data better than MOG-W and MOG-L.

REFERENCES

- [1] Yousof, H. M., Afify, A. Z., Nadarajah, S., Hamedani, G., and Aryal, G. R. (2018). The marshall-olkin generalized-g family of distributions with applications. *Statistica*, 78(3):273-295.
- [2] Marshall, A. W. and Olkin, I. (1997). A new method for adding a parameter to a family of distributions with application to the exponential and weibull families. *Biometrika*, 84(3):641-652.
- [3] Alizadeh, M., Tahir, M., Cordeiro, G. M., Mansoor, M., Zubair, M., and Hamedani, G. (2015). The kumaraswamy marshal-olkin family of distributions. *Journal of the Egyptian Mathematical Society*, 23(3):546-557.
- [4] Alizadeh, M., Cordeiro, G. M., Brito, E. d., and B Demeétrio, C. G. (2015). The beta marshall-olkin family of distributions. *Journal of Statistical Distributions and Applications*, 2(1):1-18.
- [5] Handique, L. and Chakraborty, S. (2016). The beta generalized marshall-olkin-g family of distributions. arXiv preprint arXiv:1608.05985.
- [6] B Dias, C. R., Cordeiro, G. M., Alizadeh, M., Diniz Marinho, P. R., and Campos Coelho, H. F. (2016). Exponentiated marshall-olkin family of distributions. *Journal of Statistical Distributions and Applications*, 3(1):1-21.
- [7] Chakraborty, S. and Handique, L. (2017). The generalized marshall-olkin-kumaraswamy-g family of distributions. *Journal of data Science*, 15(3):391-422.
- [8] Handique, L. and Chakraborty, S. (2017). The beta generalized marshall-olkin kumaraswamy-g family of distributions with applications. *Int. J. Agricult. Stat. Sci*, 13(2):721-733.

- [9] Handique, L., Chakraborty, S., and de Andrade, T. A. (2019). The exponentiated generalized marshallolkin family of distribution: its properties and applications. *Annals of Data Science*, 6(3):391-411.
- [10] Korkmaz, M. C., Cordeiro, G. M., Yousof, H. M., Pescim, R. R., Afify, A. Z., and Nadarajah, S. (2019). The weibull marshallolkin family: Regression model and application to censored data. *Communications in Statistics-Theory and Methods*, 48(16):4171-4194.
- [11] R Core Team (2021). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.
- [12] Team, S. D. et al. (2017). Stan modeling language users guide and reference manual. *Technical report*.
- [13] Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics letters B*, 195(2):216-222.
- [14] Neal, R. M. et al. (2011). Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2.
- [15] Hoffman, M. D. and Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593-1623.
- [16] Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian data analysis*. Chapman and Hall/CRC.
- [17] Collett, D. (2015). *Modelling survival data in medical research*. CRC press.
- [18] McGilchrist, C. and Aisbett, C. (1991). Regression with frailty in survival analysis. *Biometrics*, 461-466.
- [19] Therneau, T. M. and Lumley, T. (2014). Package 'survival'. *Survival analysis* Published on CRAN, 2:3.
- [20] Khan, N. and Khan, A. A. (2018). Bayesian analysis of topp-leone generalized exponential distribution. *Austrian Journal of Statistics*, 47(4):1-15.
- [21] Akhtar, M. T. and Khan, A. A. (2014). Bayesian analysis of generalized log-burr family with R. *SpringerPlus*, 3(1):185.
- [22] Gelman, A. et al. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian analysis*, 1(3):515-534.
- [23] McElreath, R. (2018). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.
- [24] Lawless, J. F. (2011). *Statistical models and methods for lifetime data*. John Wiley and Sons.
- [25] Watanabe, S. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(12):3571:3594.