Nepal Journal of Mathematical Sciences (NJMS)
ISSN: **2738-9928 (*online*), 2738-9812 (*print*)**
Vol. 3, No. 2, 2022 (August): 39-58
DOI: 10.3126/njmathsci.v3i2.49202

Research Article
Received Date: March 3, 2022
Accepted Date: August 2, 2022
Published Date: August 30,2022

A Bayesian Analysis and Estimation of Weibull Inverse Rayleigh Distribution Using HMC Method

Laxmi Prasad Sapkota
Department of Statistics, Tribhuvan University, Tribhuvan Multiple Campus, Palpa, Nepal
Email*: laxmisapkota75@gmail.com*

Abstract: *Under the Bayesian environment we have analyzed the Weibull inverse Rayleigh model. The parameters of the model are estimated and predicted through posterior samples which are generated using Markov Chain Monte Carlo (MCMC) technique. The concern model is fitted using Stan software (a probabilistic programming language), utilizing the Hamiltonian Monte Carlo (HMC) algorithm and its adaptive variant the No-U-turn sampler (NUTS). For the illustration, we have considered a real data set and performed Bayesian analysis numerically and graphically using weakly Gamma informative priors. The posterior predictive check is also carried out to accesses the predictability of the model. The tools and methods used in this article are under the Bayesian approach which is implemented in R statistical programming language.*

Keywords: *Hamiltonian Monte Carlo, Inverse Rayleigh distribution, Gamma prior, Credible interval, Bayesian estimation.*

1. Introduction

The inverse Rayleigh (IR) distribution is a flexible model and plays a vital role in modeling reliability datasets, which was developed by Treyer [1]. The distribution function (CDF) and density function (PDF) of IR distribution with a parameter λ, can be expressed as

$$G(x;\lambda) = \exp\left(-\lambda / x^2\right) \; ; x > 0, \; \lambda > 0 \text{ and} \tag{1}$$

$$g(x;\lambda) = 2\lambda x^{-3} \exp\left(-\lambda / x^2\right) \; ; x > 0, \; \lambda > 0 \text{ respectively.} \tag{2}$$

Voda [39] has studied the properties and estimated the parameters of the IR distribution. Similarly, Gharraph [19] has provided a detailed closed-form illustration for the arithmetic mean, harmonic mean, median, geometric mean, and mode of IR distribution.

Using various generating techniques, the extension of IR distribution has been provided by different researchers, some of them are, the study of acceptance sampling method based on the IR distribution introduced by Rosaiah & Kantam [33]. Further IR distribution was used in estimation and prediction depending on the lower record values by Soliman et al. [34]. Ahmad et al. [2] have defined the transmuted IR distribution. Khan [22] has developed an extension of IR distribution that can have a bathtub hazard function called modified inverse Rayleigh distribution with parameters α and β. Further, Khan [23] has introduced another extension of IR distribution having three parameters named transmuted modified IR distribution. The Kumaraswamy exponentiated IR distribution was presented by Ul Haq [37]. Fatima and Ahmad [10] have studied the weighted IR distribution. Fatima et al. [11] have studied several structural properties of exponentiated generalized IR distribution. Another extension of IR distribution has been studied by Elgarhy & Alrajhi [9] named odd frechet IR distribution with upside-

down or j-shaped or right-skewed hazard rate function. Mohammed and Yahia [27] have presented type II Topp Leone IR distribution and Almarashi et al. [4] presented another extension of IR distribution using a half-logistic generating family of distribution that can be used for left-skewed lifetime data. In this study, we have considered a flexible model defined by Ogunsanya et al. [31] named Weibull Inverse Rayleigh (WIR) distribution, and the authors have adopted the classical approach and Lindley approximation approach for parameter estimation. We have performed a full Bayesian approach under MCMC along with the Hamiltonian Monte Carlo (HMC) algorithm and its adaptive variant the No-U-turn sampler (NUTS) for the study of this model. The main objective of this work is to analyze the Weibull inverse Rayleigh model and its predicting capability under the Bayesian environment.

2. WIR Model

The CDF and PDF of the WIR distribution are

$$F\left(x;\alpha,\beta,\lambda\right)=1-\exp\left[-\beta\left\{\frac{\exp\left(-\alpha/x^2\right)}{1-\exp\left(-\alpha/x^2\right)}\right\}^{\lambda}\right];x>0,\ (\alpha,\beta,\lambda)>0 \text{ and} \tag{3}$$

$$f\left(x;\alpha,\beta,\lambda\right)=2\alpha\beta\lambda x^{-3}\left\{e^{\left(-\frac{\alpha}{x^2}\right)}\right\}^{\lambda}\left\{1-e^{\left(-\frac{\alpha}{x^2}\right)}\right\}^{-(\lambda+1)}\exp\left[-\beta\left\{\frac{\exp\left(-\frac{\alpha}{x^2}\right)}{1-\exp\left(-\frac{\alpha}{x^2}\right)}\right\}^{\lambda}\right];x>0 \tag{4}$$

respectively.

2.1. Survival/Reliability Function

Let $X>0$ is random variable follows the WIR distribution, and then the survival function is

$$S\left(x\right)=\exp\left[-\beta\left\{\frac{\exp\left(-\alpha/x^2\right)}{1-\exp\left(-\alpha/x^2\right)}\right\}^{\lambda}\right];x>0,\ (\alpha,\beta,\lambda)>0\ . \tag{5}$$

2.2. Hazard Rate Function (HRF)

The HRF of WIR distribution having parameters (α,β,λ) is

$$h\left(x\right)=2\alpha\beta\lambda x^{-3}\left\{\exp\left(-\frac{\alpha}{x^2}\right)\right\}^{\lambda}\left\{1-\exp\left(-\frac{\alpha}{x^2}\right)\right\}^{-(\lambda+1)};x>0,\ (\alpha,\beta,\lambda)>0\,. \tag{6}$$

The HRF of WIR distribution can have different forms like monotonically increasing, increasing-decreasing, j-shaped, or inverted bathtub (Figure 2) according to different values of α *and* λ keeping β as constant. Similarly, we can obtain the cumulative hazard function of WIR distribution as

$$H\left(x\right)=\int_0^x h(y)dy=\beta\left\{\frac{\exp\left(-\alpha/x^2\right)}{1-\exp\left(-\alpha/x^2\right)}\right\}^{\lambda} \tag{7}$$

and failure rate average (FRA) is

$$FRA(x)=\frac{H\left(x\right)}{x}=\frac{\beta}{x}\left\{\frac{\exp\left(-\alpha/x^2\right)}{1-\exp\left(-\alpha/x^2\right)}\right\}^{\lambda}\ . \tag{8}$$

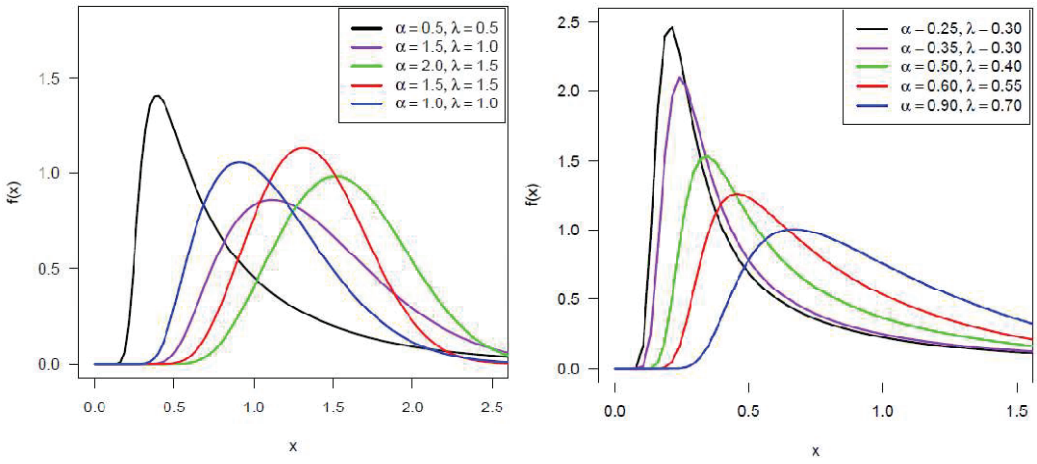

Figure 1: *The graphs of PDF of WIR distribution for α and λ keeping β as constant.*

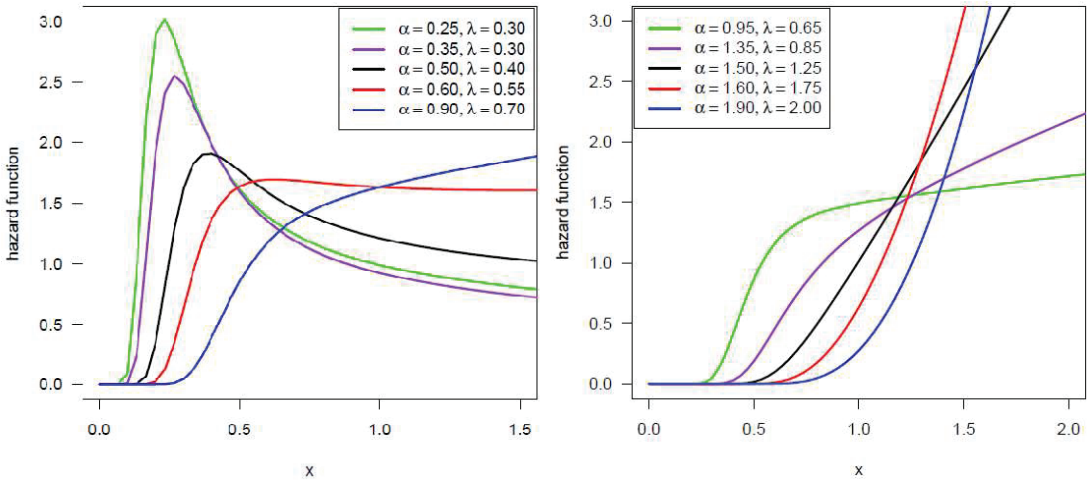

Figure 2: *The graphs of HRF of WIR distribution for α and λ keeping β as constant.*

2.3. Quantile Function of WIR distribution

The quantile function is the inverse function of CDF which can be expressed as

$$Q(u) = \sqrt{\alpha\left\{\ln\left[1+\left\{\frac{\ln(1-u)}{-\beta}\right\}^{-1/\lambda}\right]\right\}^{-1}} \quad ; 0<u<1. \tag{9}$$

Let $V \sim U(0,1)$, then the random deviate of X is calculated by using

$$x = \sqrt{\alpha\left\{\ln\left[1+\left\{\frac{\ln(1-v)}{-\beta}\right\}^{-1/\lambda}\right]\right\}^{-1}} \quad ; 0<v<1. \tag{10}$$

2.4. Median, upper and lower quartiles of the WIR distribution

The median of the WIR distribution can be obtained as

$$Median = \sqrt{\alpha\left\{\ln\left[1+\left\{\frac{\ln 0.5}{-\beta}\right\}^{-1/\lambda}\right]\right\}^{-1}}.$$

The lower and upper quartiles can be calculated as

$$Q_1 = \sqrt{\alpha\left\{\ln\left[1+\left\{\frac{\ln 0.75}{-\beta}\right\}^{-1/\lambda}\right]\right\}^{-1}} \text{ and } Q_3 = \sqrt{\alpha\left\{\ln\left[1+\left\{\frac{\ln 0.25}{-\beta}\right\}^{-1/\lambda}\right]\right\}^{-1}} .$$

3. Model Formulation using the Bayesian approach

In a classical approach, we generally assume the parameters $\theta=(\alpha,\beta,\lambda)$ (for our study) as a constant and the goal is to analyze the distribution of the observed data set given θ using the likelihood of the data sample. But in the Bayesian approach, the parameter θ is considered as a random variable where as the observed data set is taken as constant. In this type of modeling, prior information is used to support our assumption about the parameters of the distribution [17]. In Bayesian modeling, the posterior distribution function is obtained by multiplying the prior distribution function and the likelihood function of the model under consideration for more detail see [25]. For Bayesian inference, we need the following elements

- The probability distribution function: $f(x/\alpha,\beta,\lambda)$
- Prior distribution: $p(\alpha,\beta,\lambda)$
- Likelihood $p(data/\theta)$
- Data: $\underline{x}=(x_1,...,x_n)$

3.1. Prior Distribution $p(\theta)$

In Bayesian inference, a prior distribution (simply called prior) is the unconditional probability distribution that is used to express our beliefs about the true value of the parameters before the data is taken into account. Mathematically a prior is a weight given to each parameter value which is used in the numerator of Bayes' rule to multiply the likelihood. Bayes' rule is actually only a technique to revise our initial beliefs with regard to observation or data. The term $p(\alpha,\beta,\lambda)$ denotes the probability distribution which represents our pre-data beliefs depending upon the different values of the parameters $\theta=(\alpha,\beta,\lambda)$ of our model. In this article, we have taken the weakly-informative Gamma prior for the parameters $\theta=(\alpha,\beta,\lambda)$ as

$\alpha\sim G(a_1,b_1)$, $\beta\sim G(a_2,b_2)$ and $\lambda\sim G(a_3,b_3)$ where $(a_1=0.5,\ b_1=0.5)$, $(a_2=0.1,\ b_2=0.1)$ and $(a_3=0.1,\ b_3=0.1)$ respectively. This Gamma prior is the most commonly used weak prior on variance which is nearly flat (Fig. 3). The prior distributions can be written as

$$p(\alpha)=\frac{b_1^{a_1}}{\overline{|a_1}}\alpha^{a_1-1}\exp(-b_1\alpha);\ \ \alpha>0,\ (a_1,b_1)>0 .$$

$$p(\beta)=\frac{b_2^{a_2}}{\overline{|a_2}}\beta^{a_2-1}\exp(-b_2\beta);\ \ \beta>0,\ (a_2,b_2)>0 .$$

$$p(\lambda)=\frac{b_3^{a_3}}{\overline{|a_3}}\lambda^{a_3-1}\exp(-b_3\lambda);\ \ \lambda>0,\ (a_3,b_3)>0 .$$

3.2. Likelihood $p(data/\theta)$

The likelihood function provides us the probability of obtaining a particular data conditioned on model parameters. Given a set of data $\underline{x}=(x_1,...,x_n)$, the likelihood function can be computed as

$$L(\alpha,\beta,\lambda/\underline{x})=(2\alpha\beta\lambda)^n\prod_{i=1}^{n}\left[x_i^{-3}\left\{I(x_i)\right\}^{\lambda}\left\{1-I(x_i)\right\}^{-(\lambda+1)}\right]\exp\left[-\beta\sum_{i=1}^{n}\left\{\frac{I(x_i)}{1-I(x_i)}\right\}^{\lambda}\right]$$

where $I(x_i) = \exp\left(-\frac{\alpha}{x_i^2}\right)$.

3.3. Posterior distribution $p(\theta / data)$

The posterior probability distribution is the probability distribution of a parameter θ under study, considered as a random variable, conditional on the facts taken from a survey or experiment. Let $p(\alpha, \beta, \lambda / \underline{x})$ denote the posterior distribution then, according to Bayes' rule

$$p(\alpha, \beta, \lambda / \underline{x}) \propto L(\alpha, \beta, \lambda / \underline{x}) \times p(\alpha, \beta, \lambda).$$

In the Bayesian inference technique, we use Bayes' rule to estimate probability distribution called posterior distribution which can be obtained as

$$p(\theta / data) = \frac{p(data / \theta)\, p(\theta)}{p(data)}$$

$$p(\theta / data) \propto p(data / \theta)\, p(\theta)$$

Hence posterior distribution is defined as

$$p(\alpha, \beta, \lambda / \underline{x}) = L(\alpha, \beta, \lambda / \underline{x})\, p(\alpha)\, p(\beta)\, p(\lambda)$$

$$p(\alpha, \beta, \lambda / \underline{x}) = (2\alpha\beta\lambda)^n \prod_{i=1}^{n}\left[x_i^{-3}\left\{I(x_i)\right\}^{\lambda}\left\{1 - I(x_i)\right\}^{-(\lambda+1)}\right] \exp\left[-\beta\sum_{i=1}^{n}\left\{\frac{I(x_i)}{1-I(x_i)}\right\}^{\lambda}\right] \times$$

$$\frac{b_1^{a_1}}{\lceil a_1}\alpha^{a_1-1}\exp(-b_1\alpha)\frac{b_2^{a_2}}{\lceil a_2}\beta^{a_2-1}\exp(-b_2\beta)\frac{b_3^{a_3}}{\lceil a_3}\lambda^{a_3-1}\exp(-b_3\lambda)$$

$$p(\alpha, \beta, \lambda / \underline{x}) \propto \alpha^{n+a_1-1}\beta^{n+a_2-1}\lambda^{n+a_3-1}\prod_{i=1}^{n}\left[x_i^{-3}\left\{I(x_i)\right\}^{\lambda}\left\{1 - I(x_i)\right\}^{-(\lambda+1)}\right] \times$$

$$\exp\left[-\beta\sum_{i=1}^{n}\left\{\frac{I(x_i)}{1-I(x_i)}\right\}^{\lambda} - b_1\alpha - b_2\beta - b_3\lambda\right]$$

All the information needed for Bayesian analyses is contained in the posterior distribution and the aim is to compute the numeric as well as graphic summaries of it through integration. But the posterior distribution is quite complicated and could not draw any inferences. Hence we purpose an alternative technique known as the simulation technique. This technique is based on Markov Chain Monte Carlo (MCMC) method. Using MCMC we can generate samples by running an expertly created Markov Chain that ultimately converges to the objective distribution i.e. posterior distribution $p(\alpha, \beta, \lambda / \underline{x})$.

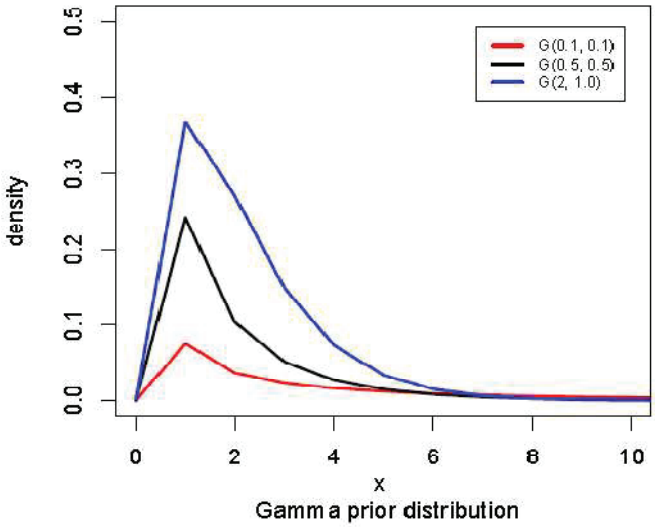

Figure 3: *Graph of Gamma prior for various values of the parameters.*

There are many different techniques to construct such chains some of them are, the Gibbs sampler [18] and [13] are special cases of the general framework of [26, 20]. In this article we have implemented MCMC algorithms through Stan (a probabilistic programming language) [35], the Hamiltonian Monte Carlo (HMC) algorithm and its adaptive variant the no-U-turn sampler (NUTS) for more detail see [21, 7]. Recently, Chaudhary & Kumar [8] have presented the Bayesian estimate of Gompertz extension distribution having three parameters. Also, Alizadeh et al. [3] discussed the technique for estimating the model parameters of the odd log-logistic Lindley-G family of distribution.

4. Methods

4.1. Hamiltonian Monte Carlo (HMC) method

HMC is computationally a bit costly as compared to Metropolis and Gibbs sampling but its proposals are much more efficient [16]. As a result, HMC doesn't require as many samples to explore the posterior distribution. To conduct the HMC algorithm, we require the following five things

i) A function U gives the minus-log probability of the data at the current position (parameter values). U is just the log posterior.
ii) A gradient function U_{grad} computes the slope of the minus-log probability at the current position.
iii) A step size epsilon (ε)
iv) Number of leapfrog steps *'L'*
v) An initial position $q_{current}$

The complete HMC algorithm can be performed under the following steps

i) Select a random starting location θ_0 from some initial proposal distribution.
ii) For $t = 1, 2, ..., n$ this, we have to conduct the following
   - Generate a random initial momentum from a proposal distribution (For eg. $X \sim N(\mu, \Sigma)$)
   - Perform the leapfrog algorithm to solve for the path of the particle moving over the minus-log of posterior density space for a time period T.
   - After an amount of time T elapsed, record the momentum of the particle 'm' and its position in parameter space θ_{t+1} .
   - Calculate

$$r = \frac{p(x/\theta_{t+1})p(\theta_{t+1})}{p(x/\theta_t)p(\theta_t)}\frac{q(m')}{q(m)}$$

   - Generate $u \sim U(0,1)$. If $r > u$ the move to the θ_{t+1} otherwise remain θ_t s at.

4.2. No-U-Turn Sampler (NUTS)

NUTS engine routinely selects a suitable value for leapfrog step L in every iteration in order to maximize the distance at every L and control the random walk behavior. Consider θ_1 and θ_0 be the current position and initial position of a particle and D be half of the distance between the positions θ_1 and θ_0 at each leapfrog step. The aim is to run leapfrog steps until θ_1 start to move backward towards θ_0, which is achieved using the following algorithm, where leapfrog steps are run until the derivative of D with respect to time becomes less than 0.

$$\frac{\partial D}{\partial t} = \frac{\partial}{\partial t}\left[\frac{1}{2}(\theta_1 - \theta_0)^T(\theta_1 - \theta_0)\right] = (\theta_1 - \theta_0)^T p < 0 .$$

However, this algorithm doesn't assure convergence or reversibility to the target distribution. The NUTS solves this type of problem by performing a doubling method for slice sampling [29]. To generate the samples using NUTS, we have to follow the following steps

i) Set the initial value of $\varepsilon, \theta, \delta, \mu, \gamma, j_0$ and κ .

ii) Create momentum p from the standard normal distribution $\mathrm{P} \sim N(0,\mathrm{I})$.

iii) Create auxiliary variable μ from the uniform distribution

$$\mu \sim Uniform\left(0, \exp\left(\log f(\theta) - \frac{1}{2}\mathrm{P}^T M^{-1}\mathrm{P}\right)\right).$$

iv) Generate C by using the doubling method with transition kernel T, where C is a subset of (θ_0, p) states.

v) Accept the proposal (θ_1, p_1) with probability α_j at the j[th] iteration.

vi) Update ε_j by dual averaging.

vii) Repeat steps (1) to (6). Since step (6) is repeated only during the warm-up phase.

Where α_j = actual acceptance probability for the j[th] iteration.

δ = desired average acceptance probability

μ = freely chosen point that is iterated ε_j shrivel towards.

j_0 = free parameter that dampens early exploration.

γ = free parameter that controls the amount of shriveling towards μ.

For more detail about NUTS see [21, 30].

5. Data Analysis

5.1 Data set: **Tensile strength:** We have considered a real dataset for analysis to illustrate the intended methodology. The dataset is of 65 observations of failure stresses of single carbon fibers of length 50 mm on the data of tensile strength. The data was used by Bader & Priest [5] and also used by Muhammad & Liu [28]. The descriptive statistics of the dataset are presented in Table 1.

5.2 An exploratory study of the dataset

The main aim of the exploratory analysis of the data is to explore more information about the data. The latest statistical tools for data analysis incorporate exploratory data analysis.

Table 1: Summary statistics of the dataset

Min.	Q_1	Median	Mean	Q_3	Max.	Skewness	Kurtosis
1.339	1.931	2.272	2.244	2.558	3.174	0.0223	2.5499

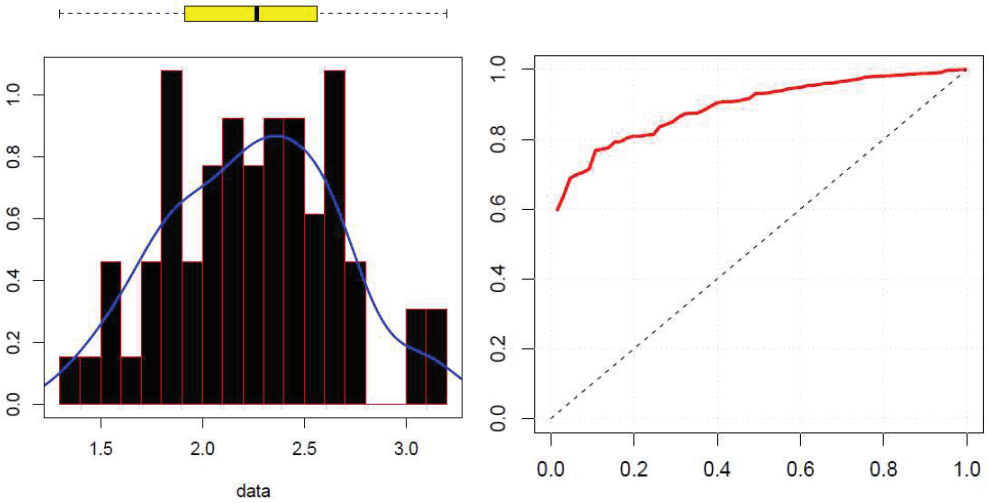

Figure 4: *Histogram with density and box plot (left panel) and TTT plot (right panel).*

Data analysis through exploratory method is a set of techniques to display and summarization of the data:

- Presenting the data in a chart and graph that demonstrates general patterns and remarkable observations (bar plot, box plot, density curve, histogram, etc.).
- Calculating descriptive information that summarizes the explicit aspect of data (central tendency and dispersion).

The basic exploratory data analysis technique is applied to study the data the tensile strength and results are displayed respectively in Table 1 and Fig. 4 (left panel). To get the nature of the shape of the hazard function, we have plotted the total-time-on test (TTT) plot (Fig.4, right panel). We have plotted the TTT plot defined by Aarset [1] of an empirical version of the scaled TTT conversion of the data set. We have observed that the curve of the TTT plot is concave; hence the hazard function is increasing.

The following steps are required for empirical modeling:

i) Developing the full probability model and estimation of its parameters;
ii) Model evaluation and validation; and
iii) Model selection.

Efficient modeling requires an excellent understanding of the characteristics of diverse types of models. The parameters of the concerned model are estimated using the maximum likelihood (ML) estimation technique. To evaluate the validity of the model, we calculate the Kolmogorov-Smirnov (KS) distance between the fitted and empirical distribution function where the parameters are estimated by the ML estimation method. The probability–probability (PP) plot and quantile-quantile (QQ) plot are used to check the suitability of the proposed model.

5.3. Calculation of MLE

The parameters (α, β, λ) of the proposed distribution are estimated for the tensile strength of a real dataset using the `optim()` function under the R programming software [32] and for more detail see [24]. The MLEs of WIR with corresponding standard errors of the estimate are presented in Table 2.

Table 2: MLE and SE for α, β *and* λ of WIR distribution

Parameter	MLE	SE
alpha	4.9585	3.8750
beta	1.8833	3.9701
lambda	1.9317	0.6448

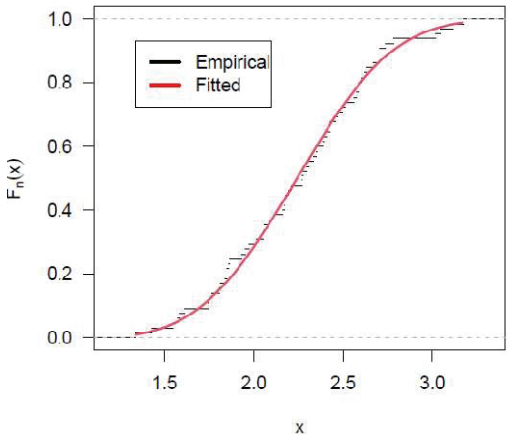

Figure 5: *The graph of the fitted and empirical distribution functions of the WIR distribution.*

5.4. Model Validation

To evaluate the validity of the model, we have calculated the KS distance 0.0623 between the fitted and empirical distribution function and its corresponding p-value is 0.9626, where the parameters are estimated by the ML estimation method. Since $p - value > 0.05$ indicates agreement with null hypothesis of good fit. We have also displayed the fitted and empirical distribution function in Fig.5, and it is obvious that the WIR distribution provides an excellent fit to the data set under consideration. Further, we have plotted the Q-Q and P-P plots to show the validity of the model in Fig. 6. From these graphs, we observed that the WIR distribution fits the data very well.

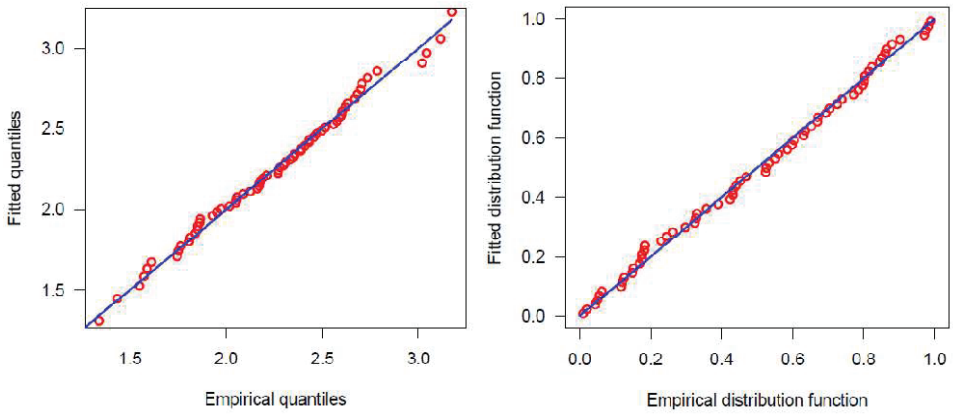

Figure 6: *The Q-Q plot (left) and P-P plot (right) of the WIR distribution*.

6. Bayesian Analysis of the WIR model

For the Bayesian analysis of the model WIR, we have used the latest Bayesian analysis software called Stan a high-level programming language that uses NUTS which is a variant of HMC simulation [21]. We have run the Stan using the algorithm HMC and engine NUTS having 4 chains for 4000 iterations. By default, Stan generates 2000 warm-up samples and 2000 real samples for a chain which are used for inferences. Stan scripts are presented in the Appendix.

6.1 Convergence and efficiency diagnostics for NUTS/ HMC and Markov Chains

In the convergence diagnostic, we monitor the NUTS/ HMC and MCMC as

i) **NUTS/ HMC**: here we study the information about divergence, energy, tree -depth, step-size, and acceptance statistic.

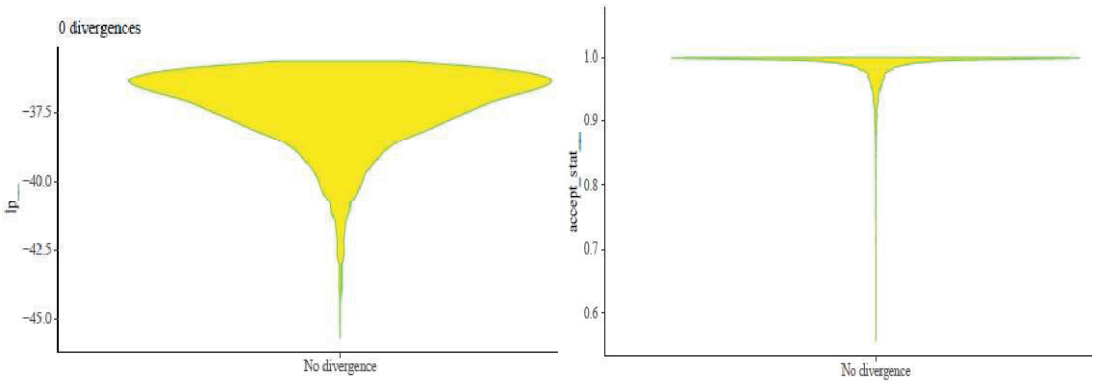

Figure 7: *Graphs of the divergent transition status of the sampling algorithm v/s the log-posterior (left panel) and acceptance statistic (right panel) for all chains.*

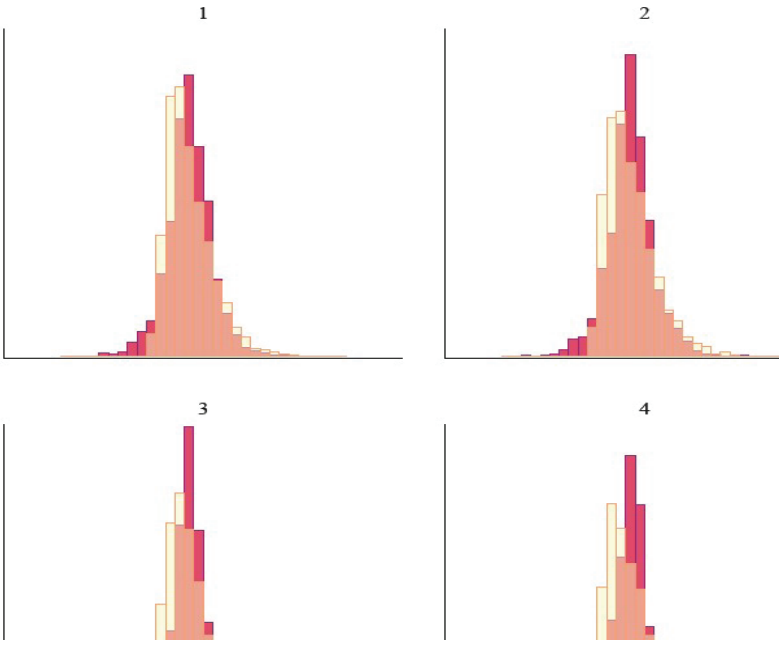

Figure 8: *Histograms of π_E $\pi_{\Delta E}$ and for all 4 chains.*

Fig. 8 is the plots of the overlaid histograms of the energy transition distribution $\pi_{\Delta E}$ and the marginal energy distribution π_E for all 4 chains and indicate that the histograms that seem well-matched and the Hamiltonian Monte Carlo has performed robustly.

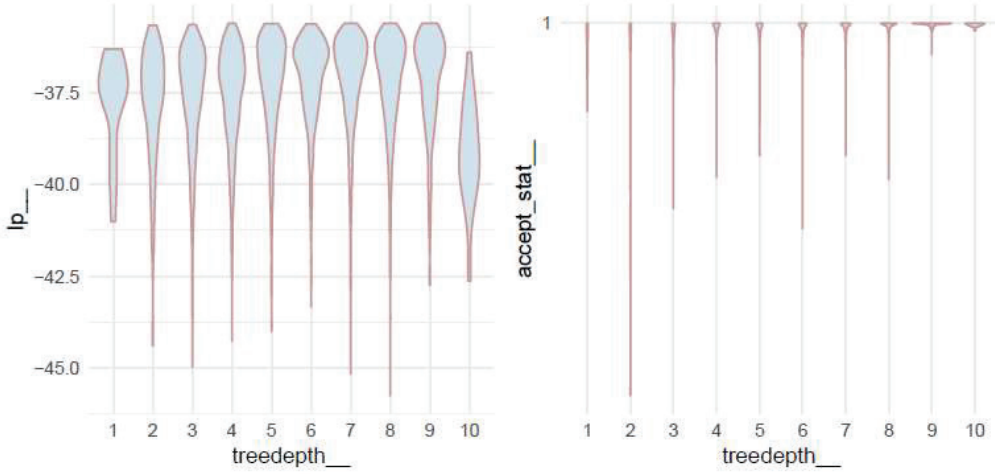

Figure 9: *Graph of tree-depth against the log posterior (left panel) and acceptance statistic (right panel) for all chains.*

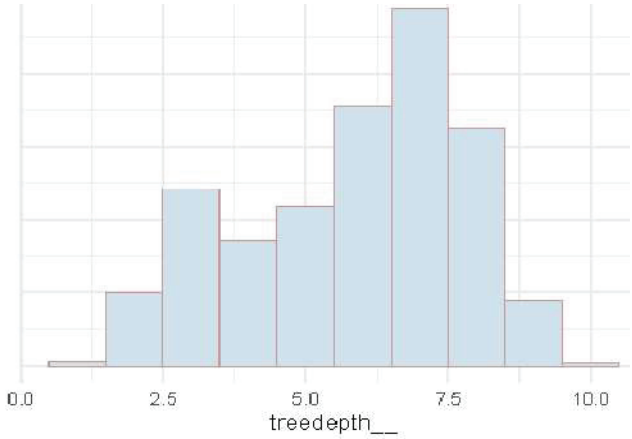

Figure 10: *Histogram of tree-depth.*

Fig. 9 and 10 indicate that the sampling algorithm is efficient. Fig. 11 indicates the full exploration of the log-posterior and high acceptance rates for step sizes and all chains.

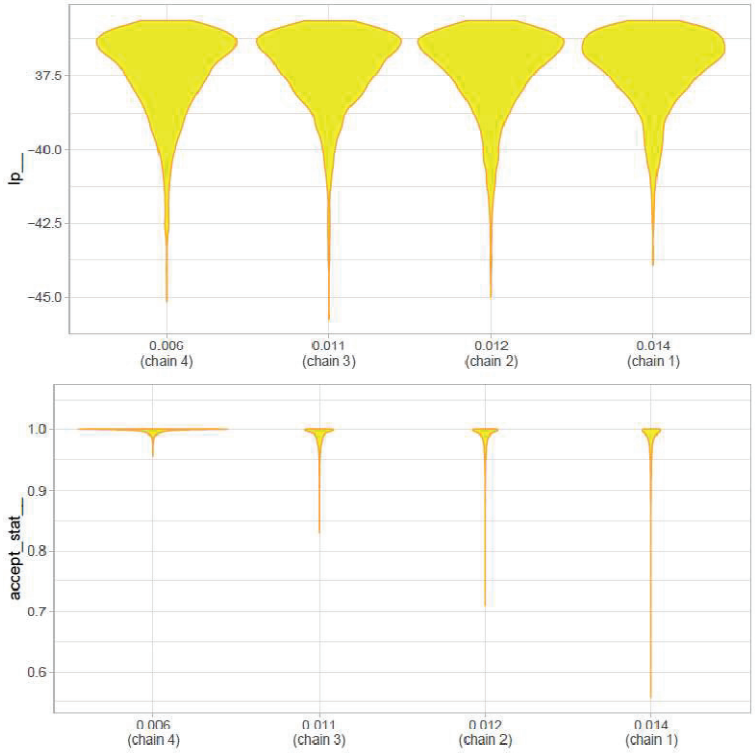

Figure 11: *Integrator step size per chain along the x-axis against the log-posterior (top panel) and acceptance statistic (bottom panel) of the sampling algorithm.*

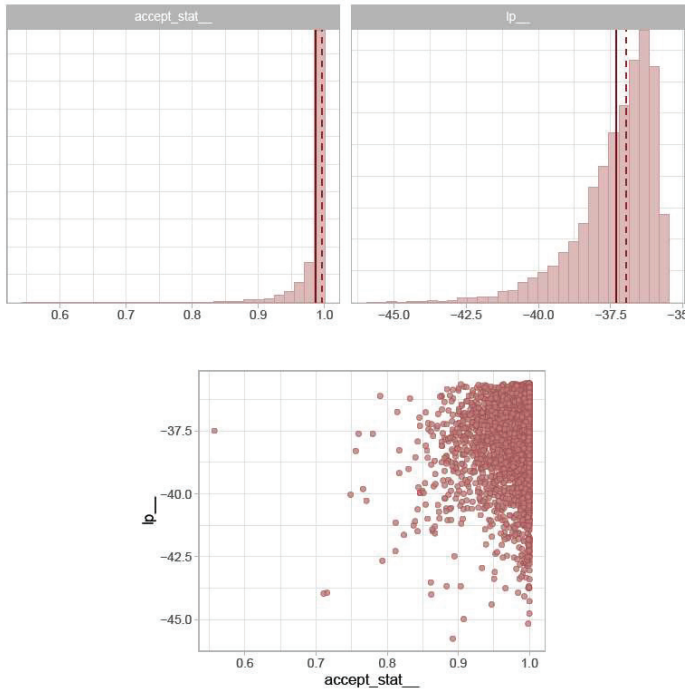

Figure 12: *The log-posterior (top right panel), acceptance statistic (top left panel), and, the acceptance statistic (x-axis bottom panel) v/s the log-posterior (y-axis bottom panel) for all chains.*

Fig. 12 are the plots of the acceptance statistic (top left panel), the log-posterior (top right panel), and the acceptance statistic against the log-posterior (bottom panel) for all chains. The vertical lines (solid)

indicate the mean and (dashed) indicate the median respectively and Fig. 12 indicates that sampling is efficient and fully explored the posterior space for detail see [6].

ii) MCMC: The MCMC draws can be monitored by plotting the following graphs

- Autocorrelation plots
- Rank plots
- Trace plots
- Ergodic mean plots
- Pairs plot

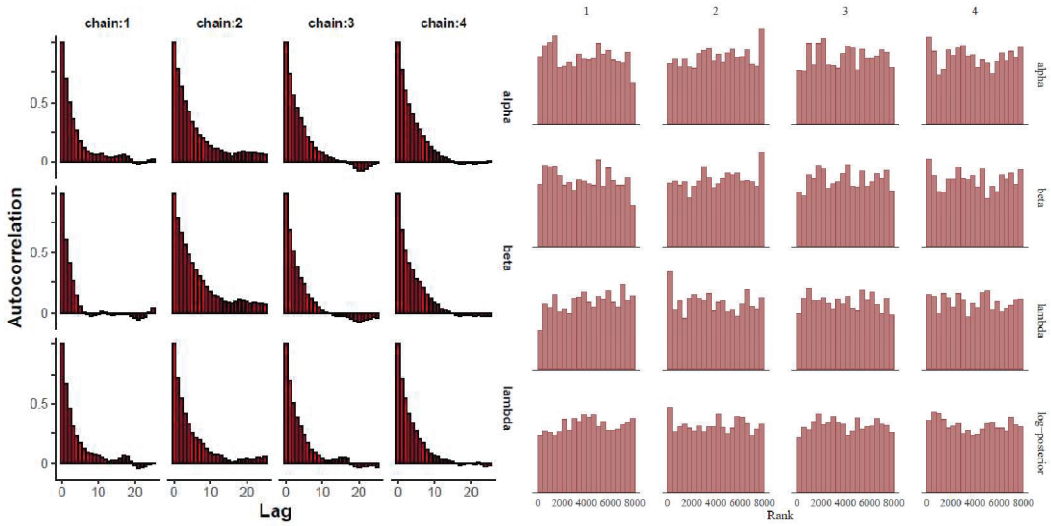

Figure 13: *Autocorrelation plots of the parameters α, β and λ for all chains (left panel) and Rank histograms of alpha, beta, lambda and log-posterior (right panel).*

To investigate the dependency of samples drawn from MCMC simulation we have plotted the autocorrelation plots for α, β *and* λ all chains. The autocorrelation expresses that there is no dependence between the samples of a Monte Carlo simulation (Fig. 13, left panel). We have displayed the rank plots of α, β, λ and log-posterior in Fig.13 (right panel) histograms of rank visualize the values from the chains mixed together in terms of ranking nearly following a uniform distribution See [38] for details.

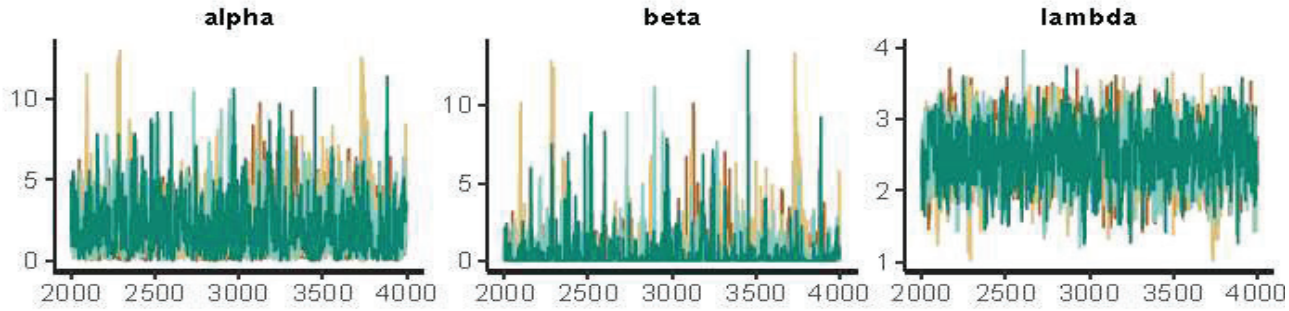

Figure 14: *Trace plot of the parameters α, β and λ for all chains.*

In general, we look for three possessions in the trace plots: (1) good mixing, (2) stationarity, and (3) convergence. Good mixing implies that the chain quickly explores the full posterior region. It doesn't slowly wander, but rather rapidly zig-zags around, as a good Hamiltonian chain should. Stationarity indicates the path of each chain staying within the same high-probability portion of the posterior distribution. Another way to imagine this is that the average value of the chain is relatively stable from start to end. Convergence represents that independent, multiple chains attach around the same area of high probability.

Fig. 14 shows the trace plots for α, β *and* λ are well mixing and convergence for all four chains. Trace plots present a visual mode to inspect sampling performance and judge mixing across chains.

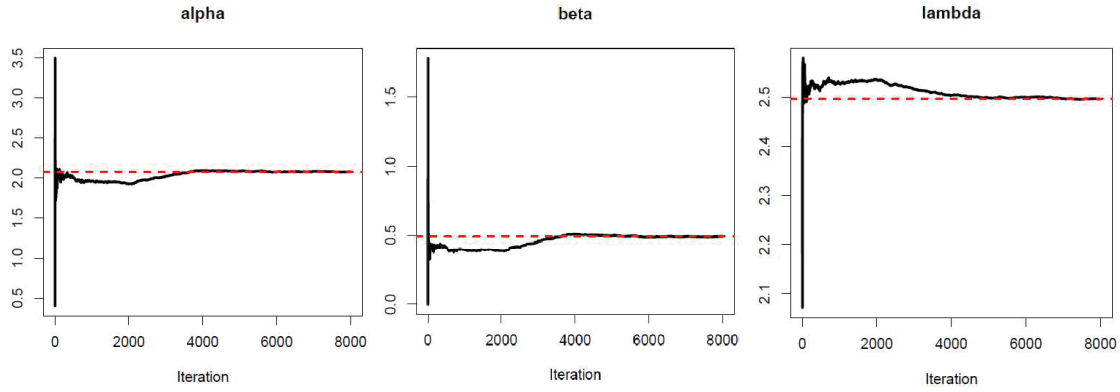

Figure 15: *The Ergodic mean plot for* α, β *and* λ.

The Ergodic mean is computed as the average of all values of the samples for all chains corresponding iterations. Fig. 15 indicates that all chains are convergent smoothly around the mean value. Fig. 16 is a pairs plot of MCMC draws of α, β, λ and log-posterior. Diagonal histograms represent the univariate marginal posteriors whereas the off-diagonal scatter plots represent the bivariate plots. We have also presented a detailed numerical summary of the HMC and NUTS algorithm in Table 3 and statistics related to the efficiency of MCMC sampling in Table 4.

Table 3: Informational statistic of NUTS/HMC for convergence of chains

Chain	accept_stat	stepsize	treedepth	nleapfrog	divergent	energy
All chains	0.99	0.01	5.85	160.83	0.00	38.82
chain1	0.98	0.01	5.61	120.48	0.00	38.73
chain2	0.98	0.01	5.57	123.12	0.00	38.87
chain3	0.99	0.01	5.63	129.32	0.00	38.78
chain4	1.00	0.01	6.59	270.39	0.00	38.92

Table 4: Informational statistic of MCMC for convergence of chains

Parameters	Rhat	n_eff / N	se_mean / sd	n_eff	Bulk_ESS	Tail_ESS
alpha	1	0.12	0.03	934	908	1099
beta	1.01	0.11	0.03	857	913	1127
lambda	1	0.14	0.03	1127	1155	925
log-posterior	1	0.14	0.03	1093	1314	1450

6.2 Posterior Analysis:

a) Numerical Summary

Using the `stan()` function in R-Software we have estimated the posterior density of the fitted WIR model. We have presented the posterior summary for all merged chains in Table 5. Table 5 includes estimates of the posterior mean, standard error of the mean, and posterior standard deviation (sd). Quantiles for the median (50%) and 95% posterior interval (2.5%, 97.5%) are also presented. The remaining last two columns n_eff (number of effective sample size) and no parameters have an effective sample size for estimating the posterior mean less than 10% of the total sample size indicating that the samples are efficient and Rhat (estimated potential scale reduction statistic) provides the analysis of sampling and its efficiency here Rhat is less than 1.1 indicates convergence of chains.

Table 5: Output summary of posterior samples for the WIR model

Parameters	mean	se_mean	sd	2.5%	50%	97.5%	n_eff	Rhat
alpha	2.08	0.06	1.82	0.04	1.64	6.62	933.52	1.00
beta	0.49	0.04	1.16	0.00	0.06	3.80	855.40	1.01
lambda	2.50	0.01	0.43	1.59	2.53	3.26	1125.39	1.00
Log-likelihood	-37.32	0.04	1.36	-40.84	-36.97	-35.75	1090.30	1.00

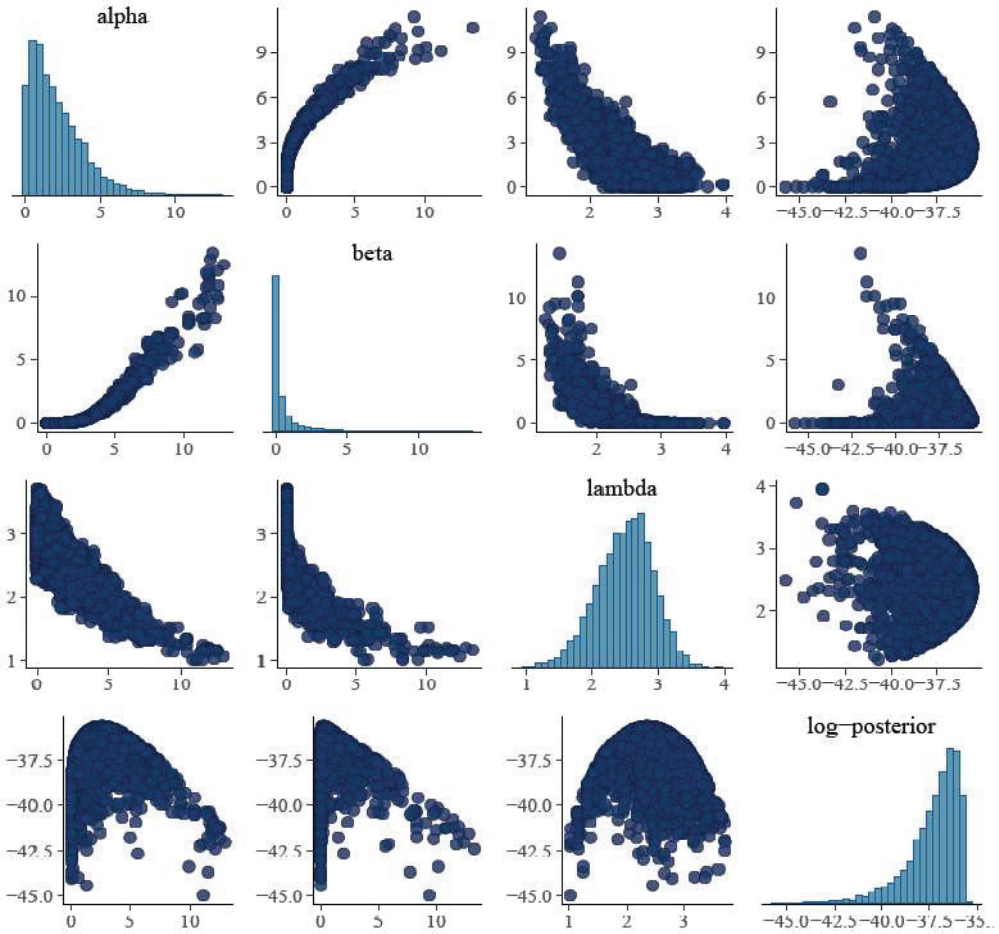

Figure 16: *Pairs plot of alpha, beta, lambda and log-posterior.*

b) Visual Summary

The visual summary of posterior distribution can be presented using various graphs like histograms, boxplots, caterpillar plots, density plots etc. We have plotted histogram and kernel density estimate with all chains for alpha (Fig.17), beta (Fig.18) and lambda (Fig.19). These graphical presentations provide approximately complete information about the posterior samples regarding the parameters. Over all posterior samples of size 8000 were used to construct these graphs. Generally, histograms are useful to know the information about the nature in the tails, skewness, kurtosis, outliers and the existence of multi-modal behavior. Fig.18 represents the histogram with 95% credible interval (green line) (left panel) and kernel density estimates along with 4 chains (right panel) for the parameter alpha. Similar graphs for parameters beta and lambda are also constructed and presented in Fig.19 and Fig.20 respectively. It is observed that the parameter lambda is nearly symmetric whereas alpha and beta are positively skewed.

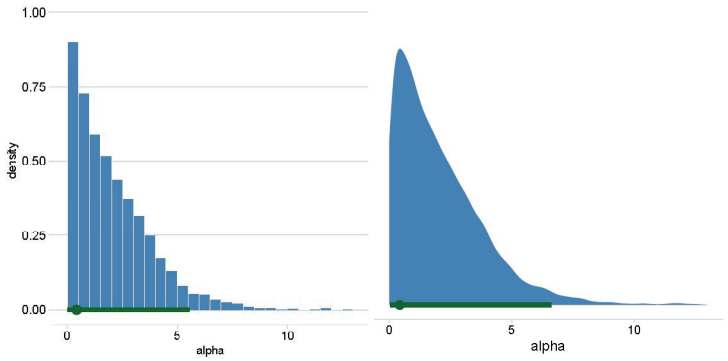

Figure 17*: Histogram with 95% credible interval (green mark with mode point) (left panel) and kernel density estimates with 95% credible interval (green mark with mode point) (right panel) for the parameter alpha.*

Also, we have constructed the caterpillar plot which is a well-linked plot in Bayesian analysis for summarizing the quantiles of posterior samples. In Fig.20 caterpillar plot is displayed with a 90% credible interval (blue line) and 99% credible interval (red line) (left panel) and the point estimate is median by default.

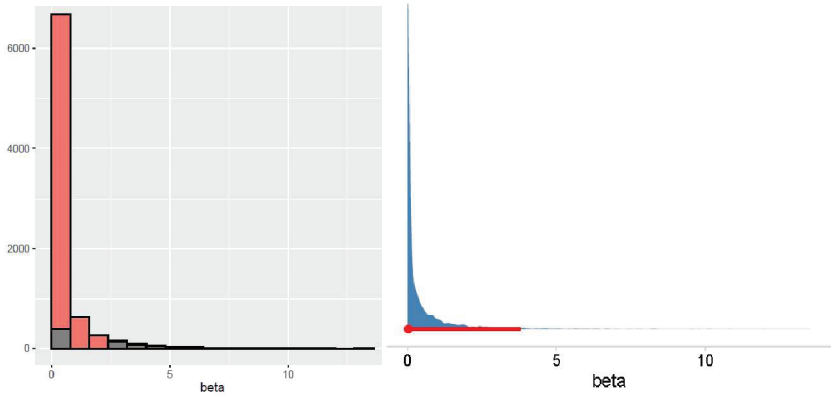

Figure 18: *Histogram with 90% credible interval (red area) (left panel) and kernel density estimates with 95% credible interval (red mark with mode point) (right panel) of parameter beta.*

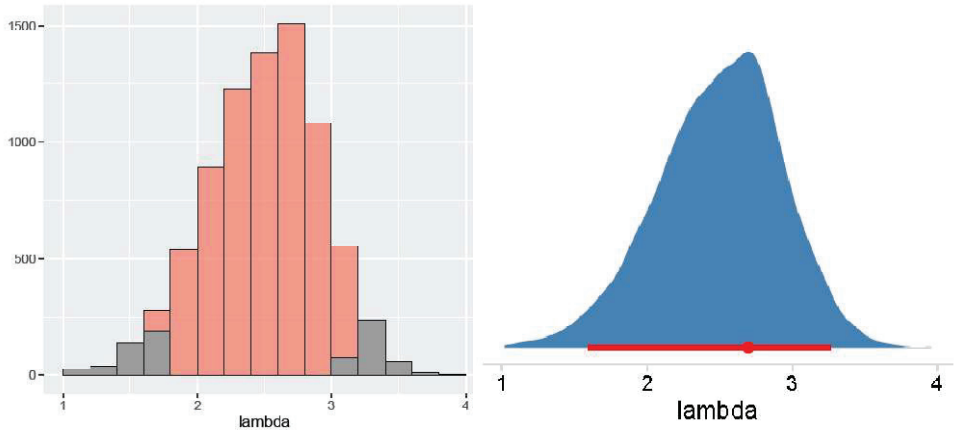

Figure 19: *Histogram with 90% credible interval (red area) (left panel) and kernel density estimates with 95% credible interval (red mark with mode point) (right panel) of parameter lambda.*

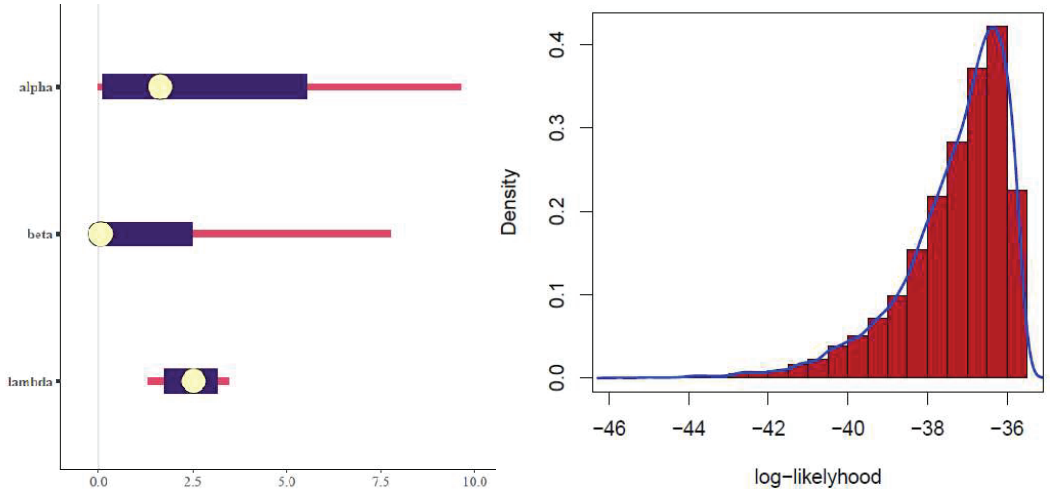

Figure 20: *Caterpillar plot with 90% credible interval (blue line) (left panel) and point estimate is median and histogram of posterior log-likelihood (right panel).*

6.3 Model Compatibility:

Posterior predictive checks (PPCs)

A usual way to access the fit attained by a Bayesian model is to observe how good the predictions can be created from the model that agrees with the observed data [14] and [15]. If our model is capable to fit the

data then it should generate data that are quite similar to the observed data.The data which are used for PPCs we can generate them by simulating the posterior predictive distribution (PPD). The R package **bayesplot** presents different plotting functions for visual posterior predictive checking*;* using observed data and simulated data from the PPD we can generate these graphical displays [12].

The PPD is the distribution of the outcome variable implied by a probability model after utilizing the observed data y (a vector of length N=65) to update our beliefs about unknown model parameters $\theta=(\alpha,\beta,\lambda)$. The PPD for observation y_{rep} can be expressed as $p(\tilde{y}/y)=\int p(\theta/y)\,p(\tilde{y}/\theta)\,d\theta$

For every simulation (draw) $s=1,\ldots,S$ of the parameters from the posterior distribution $\theta(s)\sim p(\theta\mid y)$, we have generated a vector of N outcomes $\tilde{y}^{(s)}$ using the PPD by simulating from the data model conditional on parameters $\theta^{(s)}$. The outcome is a $S\times N$ (8000×65) matrix of draws $\tilde{y}$. We have denoted the resulting simulation matrix by y_{rep}, this matrix as the replications of the observed data y rather than predictions for future observations. To attain further clarity on our decision for the study of the posterior predictive checks we have taken the smallest, middle and largest, i.e. ($y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$) replicated observations. Functions for carrying out a broad variety of graphical model checks depend on comparing observed data to draws from the posterior (or prior) predictive distribution and comparing the empirical distribution of the data y to the distributions of simulated/replicated data y_{rep} from the PPD.

To analyze the predicting capacity of posterior samples we have presented the visual summaries such as histogram (Fig. 21), box and whiskers plot (Fig. 22, left panel) are displayed for y and $y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$. Empirical CDF estimates of each dataset (row) y_{rep} are plotted with the distribution of y (dark) is displayed in (Fig. 23, right panel). We have also displayed an interval plot (Fig. 23, left panel) which plots intervals as vertical bars (inner 80% and outer 90% HPD) with points indicating y_{rep} medians and darker points indicating observed y values. Histograms of the predictive errors computed from y and $y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$ are displayed in (Fig. 23, right panel).

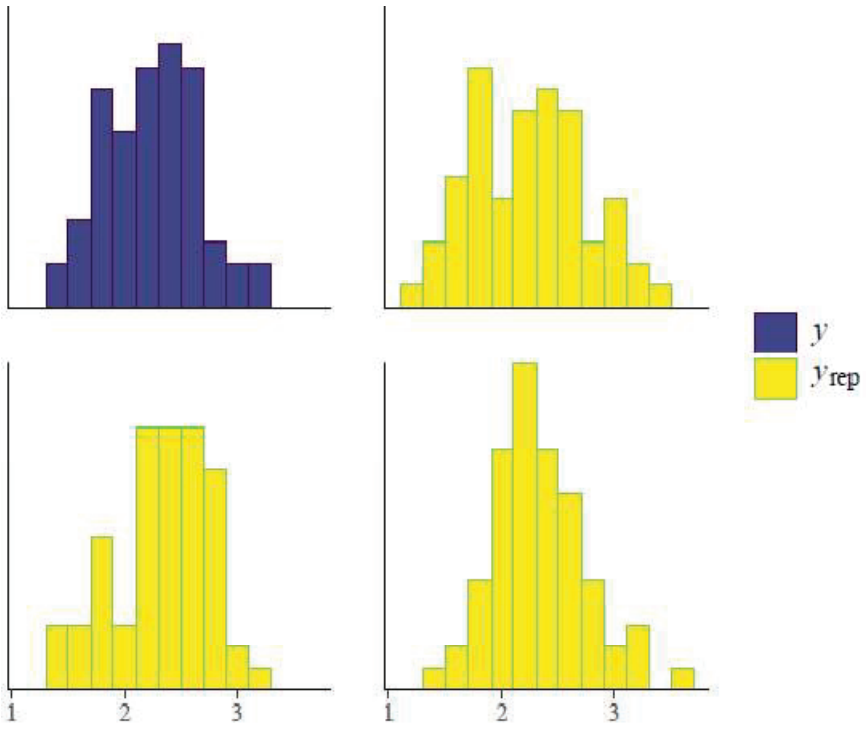

Figure 21: *Histograms of observed data (y) and replicated data* $y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$.

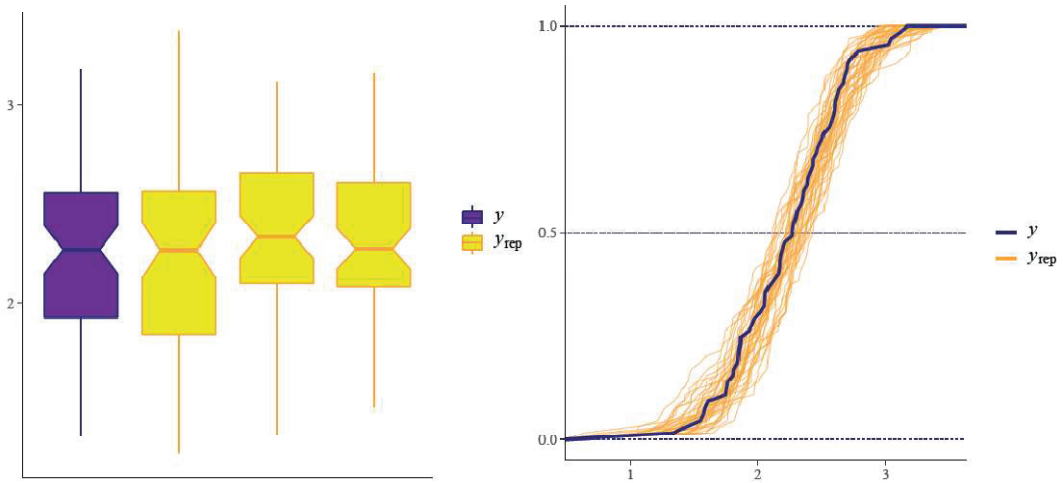

Figure 22: *Boxplot of observed data (y) and replicated data* $y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$ *(left panel) and estimated CDF of observed data (y) and replicated data (right panel).*

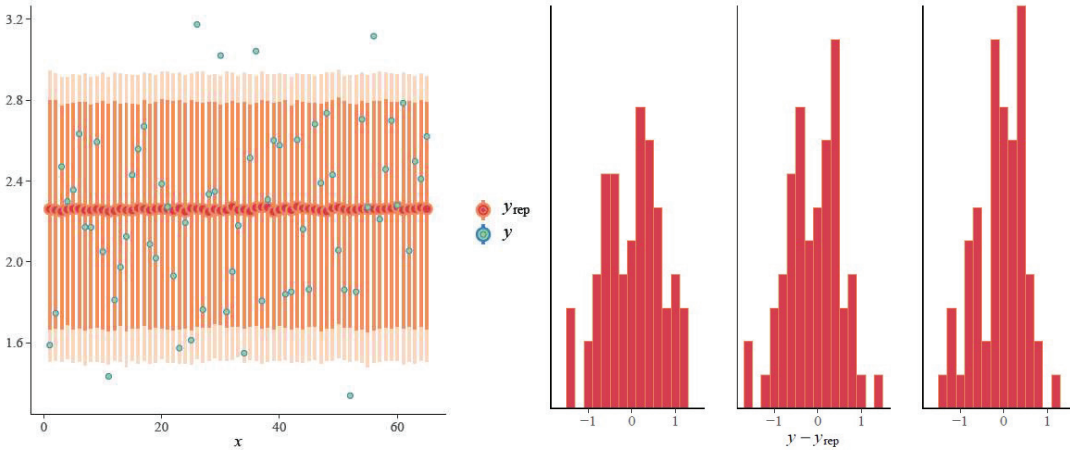

Figure 23: *Interval plot (left panel) 80% HPD interval (dark vertical bars) and 90% HPD interval (light vertical bars)* y_{rep} *with median points vs. data points y (green) and Histogram of the predictive errors computed from y and* $y_{rep}[1]$, $y_{rep}[32]$ *and* $y_{rep}[65]$ *(right panel).*

Conclusion

Using Stan software whose Markov chain Monte Carlo (MCMC) techniques are based on No-U-Turn sampler (NUTS) which is an adaptive variant of Hamiltonian Monte Carlo (HMC); a more robust and efficient sampler, we have conducted a Bayesian analysis of WIR distribution. We have presented the numerical as well as graphical analysis of the WIR model and found that all chains are well mixed and conversed. Further, we have estimated the parameters of the model and performed posterior predictive checks and found that the underlying model can be used to generate reliable samples. The developed techniques are applied to an observed data set thus we can apply these for a full Bayesian analysis of the WIR model using these Bayesian techniques.

Appendix

Stan scripts of the model WIR for the Bayesian analysis

```
Functions {real Weibull.IR_lpdf(real y, real alpha, real beta, real lambda){
return   log(2*alpha*beta*lambda*y^(3)*((exp(alpha/y^2))^lambda)*(1-exp(-
alpha/y^2))^(-1-lambda)* exp(-beta*((exp(- alpha/y^2))/(1-exp(-
alpha/y^2)))^lambda));
  }

}

data{
  int N;
  real y[N];
}
parameters{
  real <lower=0> alpha;
  real <lower=0> beta;
  real <lower=0> lambda;
}
model{
 for(i in 1 : N){
    y[i]~ Weibull.IR(alpha, beta, lambda);// likelihood
  }
  alpha~ gamma(0.5, 0.5);// priors
  beta~ gamma(0.1, 0.1);
  lambda~ gamma(0.1, 0.1);
}

generated quantities{
  vector [N] yrep;
  for(i in 1 : N){
    yrep[i]=  Weibull.IR_rng(alpha, beta, lambda);
  }
}
```

Data Creation in R software

```
y = c(1.589, 1.746, 2.471, 2.299, 2.356, 2.633, 2.172, 2.171, 2.593, 2.051,
1.434, 1.812, 1.974, 2.125, 2.430, 2.558, 2.670, 2.088, 2.019, 2.386, 2.272,
1.931, 1.574, 2.194, 1.613, 3.174, 1.764, 2.335, 2.349, 3.020, 1.753, 1.952,
2.180, 1.549, 2.514, 3.042, 1.807, 2.308, 2.601, 2.577, 1.840, 1.852, 2.604,
2.162, 1.864, 2.682, 2.390, 2.735, 2.431, 2.058, 1.862, 1.339, 1.852, 2.705,
2.270, 3.116, 2.211, 2.458, 2.699, 2.280, 2.785, 2.055, 2.497, 2.410, 2.620).
N = length(y)
Data = list(y=y, N=N).
```

References

[1] Aarset, M. V. **(1987)**. How to identify bathtub hazard rate. *IEEE Transactions Reliability*, **36**, 106-108.

[2] Ahmad, A., Ahmad, S. P., & Ahmed, A. **(2014)**. Transmuted inverse Rayleigh distribution: A generalization of the inverse Rayleigh distribution. *Mathematical Theory and Modeling*, ***4***(7), 90-98.

[3] Alizadeh, M., Afify, A.Z., Eliwa, M.S., & Ali, S. **(2020)**. The odd log-logistic Lindley-G family of distributions: properties, Bayesian and nonBayesian estimation with applications. *Comput Stat* **35**, 281–308.

[4] Almarashi, A. M., Badr, M. M., Elgarhy, M., Jamal, F., & Chesneau, C. **(2020)**. Statistical inference of the Half-Logistic inverse Rayleigh distribution. *Entropy*, ***22***(4), 449.

[5] Bader, M. G., & Priest, A. M. **(1982)**. Statistical aspects of fibre and bundle strength in hybrid composites. *Progress in science and engineering of composites*, 1129-1136.

[6] Betancourt, M., **(2017)**. A Conceptual Introduction to Hamiltonian Monte Carlo. arXiv preprint arXiv:1701.02434.

[7] Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M.A., Guo, J., Li, P. & Riddell, A. **(2016)**. Stan: A probabilistic programming language. *Journal of Statistical Software,* **20**, 1-37.

[8] Chaudhary, A. K., & Kumar, V. **(2020)**. A Bayesian Estimation and Prediction of Gompertz Extension Distribution Using the MCMC Method. *Nepal Journal of Science and Technology*, ***19***(1), 142-160.

[9] Elgarhy, M., & Alrajhi, S. **(2019)**. The odd Fréchet inverse Rayleigh distribution: Statistical properties and applications. *J. Nonlinear Sci. Appl*, ***12***, 291-299.

[10] Fatima, K., & Ahmad, S. P. **(2017)**. Weighted inverse Rayleigh distribution. *International journal of Statistics and systems*, ***12***(1), 119-137.

[11] Fatima, K., Naqash, S., & Ahmad, S. P. **(2018)**. Exponentiated Generalized Inverse Rayleigh Distribution with Applications in Medical Sciences. *Pak. J. Statist*, ***34***(5), 425-439.

[12] Gabry, J., Simpson, D., Vehtari, A., Betancourt, M. and Gelman, A. **(2019)**, Visualization in Bayesian workflow. *J. R. Stat. Soc. A*, **182**, 389-402. :10.1111/rssa.12378. (journal version, arXiv preprint, code on GitHub).

[13] Gelfand, A. E. & Smith, A. F. M. **(1990)**. Sampling based approach to calculating marginal densities. *Journal of the American Statistical Association*. **85**, 398–409.

[14] Gelman, A. **(2003)**. A Bayesian Formulation of Exploratory Data Analysis and Goodness-of-fit Testing. *International Statistical Review*. **71**(2), 369-382.

[15] Gelman, A., Carlin, J., Stern, H., & Rubin, D. **(2004)**. *Bayesian Data Analysis*. Second Edition. London, Chapman & Hall.

[16] Gelman, A., Lee, D. & Guo, J. **(2015)**. Stan: A probabilistic programming language for Bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, **40**(5), 530-543.

[17] Gelman, Andrew, J. B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, & Donald B. Rubin. **(2013)**. *Bayesian Data Analysis*. Third. London: Chapman &Hall/CRC Press.

[18] Geman, S. & Geman, D. **(1984)**. Stochastic Relaxation. Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **6**, 721–741.

[19] Gharraph, M. K. **(1993)**. Comparison of estimators of location measures of an inverse Rayleigh distribution. *Egypt Stat. J., **37,*** 295–309.

[20] Hastings, W. K. **(1970)**. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika,* **57**, 97 –109.

[21] Hofman, M. D., & Gelman, A. **(2014)**. The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J Mach Learn Res*, **15**, 1593–1623.

[22] Khan, M. S. **(2014)**. Modified inverse Rayleigh distribution. *International Journal of Computer Applications*, ***87***(13), 28-33.

[23] Khan, M. S., & King, R. **(2015)**. Transmuted modified inverse Rayleigh distribution. *Austrian Journal of Statistics*, ***44***(3), 17-29.

[24] Lambert, B. **(2018)**. *A student's guide to Bayesian statistics*. Sage.

[25] McElreath, R. **(2018)**. *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.

[26] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. **(1953)**. Equations of state calculations by fast computing machines. *Journal Chemical Physics,* **21**, 1087–1091.

[27] Mohammed, H. F., & Yahia, N. **(2019)**. On type II Topp-Leone inverse Rayleigh distribution. *Appl. Math. Sci*, ***13***, 607-615.

[28] Muhammad, M., & Liu, L. **(2019)**. A New Extension of the Generalized Half Logistic Distribution with Applications to Real Data. *Entropy*, **21**(4), 339.

[29] Neal, R. M. **(2003)**. Slice sampling. *Ann Stat*, **31**, 705–741.

[30] Nishio, M., & Arakawa, A. **(2019)**. Performance of Hamiltonian Monte Carlo and No-U-Turn Sampler for estimating genetic parameters and breeding values. *Genetics Selection Evolution*, ***51***(1), 1-12.

[31] Ogunsanya, A. S., Akarawak, E. E. E., & Ekum, M. I. **(2021)**. A New Three-Parameter Weibull Inverse Rayleigh Distribution: Theoretical Development and Applications. *Mathematics and Statistics*, ***9***, 249-272.

[32] R Core Team **(2022)**. *R: A language and environment for statistical computing. R Foundation for Statistical Computing,* Vienna, Austria.

[33] Rosaiah, K., & Kantam, R. R. L. **(2005)**. Acceptance sampling based on the inverse Rayleigh distribution. *Economic Quality Control*, ***20***(2), 277-286.

[34] Soliman, A., Amin, E. A., & Abd-El Aziz, A. A. **(2010)**. Estimation and prediction from inverse Rayleigh distribution based on lower record values. *Applied Mathematical Sciences*, ***4***(62), 3057-3066.

[35] Stan Development Team, **(2021)**. Stan users guide, version 2.28; *https://mc-stan.org/docs/2_18/reference-manual/index.html.* Accessed 28 Oct 2021.

[36] Treyer, V. N. **(1964)**. Doklady Acad, Nauk, Belorus, U.S.S.R.

[37] Ul Haq, M. A. **(2016)**. Kumaraswamy exponentiated inverse Rayleigh distribution. *Math. Theo. Model*, ***6***(3), 93-104.URL https://www.R-project.org/.

[38] Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., & Bürkner, P. C. **(2019)**. Rank-normalization, folding, and localization: An improved $\widehat {R} $ for assessing convergence of MCMC. *arXiv preprint arXiv:1903.08008*.

[39] Voda, V.G. **(1972)**. On the inverse Rayleigh distributed random variable. *Rep. Stat. Appl. Res, **19***, 13–21.

□□