Bayesian Analysis of Gamma Model with Laplace Approximation

Romana Shehla[1], Athar Ali Khan[2]

Research Scholar, Department of Statistics & Operations Research, Aligarh Muslim University, Aligarh, Uttar Pradesh, India[1]

Professor, Department of Statistics & Operations Research, Aligarh Muslim University, Aligarh, Uttar Pradesh, India [2]

Abstract: Among all the important failure time distributions, Gamma distribution is the one which is flexible, widely understood and offers a good fit to continuous, skewed reliability data. But the non availability of simple and closed forms of reliability and hazard functions of gamma model had restricted its use. This paper provides a solution to these problems in R by employing Laplace approximation for asymptotic evaluation of integrals. However, the whole demonstration is in Bayesian framework. The asymptotic results have been cross-verified with the simulation results. Real-world problems have been used for illustrative purposes.

Keywords: Bayesian, Gamma distribution, Laplace approximation, LaplacesDemon, Marginal posterior density, R.

I. INTRODUCTION

The parametric inference procedures for reliability data often involves distributions like weibull and exponential that have been widely accepted as important failure-time distributions. There are several other statistical models that can be successfully utilised for some particular types of data. The two-parameter gamma distribution is one of these probability distributions and is well suited for continuous, skewed responses. The gamma distribution fits a wide variety of failure-time data quite efficiently. Its relationship with the exponential distribution also has enhanced its importance as the sum of iid exponential random variables also follows gamma distribution. The pdf of gamma distribution is of the form

$$f(t \mid \alpha, \phi) = \frac{1}{\phi^{\alpha}\,\Gamma(\alpha)}\, t^{\alpha-1} \exp\!\left(\frac{-t}{\phi}\right) \qquad (1)$$

where $\alpha > 0$ determines the shape and $\phi > 0$, the scale of the distribution. The reliability function of the gamma model is given as

$$R(t) = \Gamma(\alpha,\, t/\phi) \qquad (2)$$

where, $\Gamma(.,.)$ is the incomplete gamma function and the corresponding hazard function is defined as

$$h(t) = \frac{1}{\phi^{\alpha}\,\Gamma(\alpha,\, t/\phi)}\, t^{\alpha-1} \exp\!\left(\frac{-t}{\phi}\right) \qquad (3)$$

Although there is a vast literature available on estimation of the gamma parameters within the classical approach, we have worked here on the Bayesian inference of the gamma parameter(s). To evaluate characteristics of posterior such as densities, means and variances, is a very tedious task. When gamma model is used as a failure-time distribution, Bayesian computations become far more difficult as it involves incomplete gamma functions. Also, in the absence of conjugate prior-likelihood pair, evaluation of posterior quantities cannot be performed in closed forms and thus requires some intensive

computational methods. As a solution to this, [1] proposed the use of Laplace's method to expectations and variances of non-positive functions.

The main aim of this paper is to demonstrate the fitting of gamma model in reliability scenario by employing Laplace's method in R [2]. The whole work is in data analytic context.

II. LAPLACE'S APPROXIMATION

The most remarkable method of evaluating integrals involved in the posterior density i.e

$$p(\theta \mid y) = \frac{p(y \mid \theta)p(\theta)}{\int p(y \mid \theta)p(\theta)\,d\theta}$$

has been pioneered by Laplace and is known as Laplace method. [3] applied this method to approximate the ratio of two integrals involved in the posterior density. Contrary to [4], this method of summarizing multivariate posterior distribution needs only upto second-order derivatives of the posterior. The whole method is based on the behaviour of the posterior about it mode. Let $h(\theta)$ be any non-negative unimodal function with mode $\hat{\theta}$. We expand $\log h(\theta)$ in a second-order Taylor series about $\hat{\theta}$ as below,

$$\log h(\theta) \approx \log h(\hat{\theta}) - (\theta - \hat{\theta})' V (\theta - \hat{\theta})\big/2$$

where,

$$V_{ij} = -\left[\frac{\partial^2}{\partial \theta_i \theta_j} \log h(\theta)\right]_{\theta=\hat{\theta}}$$

On taking exponential of both sides, we get,

$$h(\theta) \approx h(\hat{\theta})\, e^{\frac{-1}{2}(\theta-\hat{\theta})' V(\theta-\hat{\theta})} \qquad (4)$$

Thus, if $h(\theta) = p(y \mid \theta)p(\theta)$, then equation (4) is the modal approximation of the posterior as

$$\theta \mid y \;\dot\sim\; N(\hat{\theta}, V^{-1})$$

Integration of (4) gives,

$$\int h(\theta)d\theta \approx h(\hat{\theta}) \int e^{\frac{-1}{2}(\theta-\hat{\theta})' V(\theta-\hat{\theta})}\, d\theta = \frac{h(\hat{\theta})}{\sqrt{\det(V/2\pi)}}$$

This approximate integration formula for unimodal sharp-peaked non-negative functions is known as Laplace's method.

In this paper, the whole algorithm has been implemented in R language using `LaplaceApproximation` function. This function is available in **LaplacesDemon** package introduced by Byron Hall [5]. Here are its arguments:

```
function (Model, parm, Data, Interval = 1e-06,Iterations = 100, Method = "LBFGS",
Samples = 1000,sir = TRUE, Stop.Tolerance = 1e-05)
```

where `Model` receives the model from a user-defined function. The argument `parm` requires a vector of initial values for the parameters for optimization. `Data` accepts a listed data object on which the model is to be fitted. `sir` takes a logical value to specify whether sampling importance resampling is to be implemented or not. It is implemented via `SIR` function of this package to draw independent posterior samples. The function `LaplaceApproximation` calls optim to maximize the logarithm of unnormalized joint posterior density. The default optimization algorithm is `LBFGS`. This function returns two summaries as Summary 1 and Summary 2. Summary 1 is the laplace approximation of the joint posterior density and Summary 2 presents the posterior means, sd and quantiles of all the parameters based on monte carlo samples.

III. SIMULATION

Another alternative to evaluate complex integrals is to resort to simulation technique. In the simplest cases, direct simulation and rejection sampling can be used. In complex situations, we apply Monte carlo and MCMC methods. Here, the function `LaplacesDemon` serves our purpose. The performance of the laplace approximation has been investigated through the simulation study. With the optimal initial values and estimated covariance matrix obtained from Laplace approximation, `LaplacesDemon` function maximizes the logarithm of unnormalised joint posterior density with MCMC. Arguments of this function are:

```
function (Model, Data, Initial.Values, Covar = NULL, Iterations =1e+05, Status =
1000, Thinning = 100, Algorithm = "RWM", Specs = NULL, ...)
```

where `Model` receives the same user-defined model, `Data` stands for the listed data object. `Initial.Values` accepts a vector of initial values supplied by `LaplaceApproximation`. `Covar` receives a $d \times d$ proposal covariance matrix (where d is the number of parameters) as returned by the function `LaplaceApproximation`. `Iterations` specifies the number of iterations that Laplace's Demon will update the parameters searching for target distribution. Algorithm reports the specific MCMC algorithm used for estimation and defaults to Random-Walk Metropolis algorithm.

IV. DATA ANALYSIS

In this section, we analyse a data from [6] to illustrate our methodology. The data involves failure times (in minutes) for a sample of 15 electronic components in an accelerated life test.

1.4, 5.1, 6.3, 10.8, 12.1, 18.5, 19.7, 22.2, 23.0, 30.6, 37.3, 46.3, 53.9, 59.8, 66.2

We now look for the posterior estimates of the parameters when the gamma model is fitted to the above data. The firstmost requirement for the Bayesian fitting is the definition of the likelihood. Here, we have likelihood as:

$$L = \prod_{i=1}^{n} f(t_i)$$

and thus our log-likelihood becomes

$$\log L = \sum_{i=1}^{n} (-\alpha \log \phi - \log \Gamma \alpha + (\alpha - 1) \log t_i - \frac{t_i}{\phi})$$

The model is

$$y \sim G(\alpha, \phi)$$

where,

$$\phi = \exp(X\beta)$$

is the log link function.

The Bayesian framework requires the specification of prior distributions for the parameters. Here, we stick to subjectivity and thus introduce weakly informative priors for the parameters. Priors for the α and ϕ are taken to be normal and half Cauchy densities

$$\beta_j \sim N(0, 10000), \qquad j = 1, 2, \ldots\ldots, J$$
$$\alpha \sim HC(25)$$

A. Creation of Data

Data y is entered into R using the concatenation function. Then intercept is introduced into the design matrix with a vector of 1's created with function `matrix`. The codes are as follows:

```
X<-matrix(1,nrow=length(y))
N<-length(y)
J<-ncol(X)
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,N=N,X=X,mon.names=mon.names,parm.names=parm.names,y=y)
```
MyData is a listed data object with all the specifications required for the analysis.

B. Model Specification

Positively restricted parameters of the gamma model is made to range over the entire real line with the logarithmic link function. The model with the gamma error is specified with the following set of code lines.

```
Model<-function(parm,Data)
{
###Parameters###
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
###Log(Prior densities)###
beta.prior<-sum(dnormv(beta,0,10000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
scale<-exp(tcrossprod(Data$X,t(beta)))
###Log-Likelihood###
LL<-sum(dgamma(Data$y,shape,scale=scale,log=T))
###Log-Posterior###
LP<-LL+sum(beta.prior)+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
yhat=rgamma(15,shape,scale=scale),parm=parm)
return(Modelout)
}
```

Initial values for the parameters can be taken as zero if proper prior knowledge is not at hand. Another way of assigning initial values to all the parameters of the model is to use the function `GIV` of the same package. Here we have generated initial values with the `GIV` function.

```
###Initial Values###
```

```
Initial.Values<-GIV(Model,MyData,n=1000)
```

Finally, the fitting is done with the `LaplaceApproximation` function using the following commands

```
###Fitting with LaplaceApproximation###
set.seed(1)
M1<-LaplaceApproximation(Model,Initial.Values,MyData,Iterations=15000)
```

Output summary has been tabulated in Table 1 and Table 2.

Table 1   Asymptotic estimates of mode and modal standard deviations for the parameters $\log \phi$ and $\log \alpha$

	Mode	SD	LB	UB
beta	2.9507	0.39	2.16	3.74
log.shape	0.3651	0.33	-0.29	1.03

Table 2   Posterior mean values of the parameters along with the quantiles as obtained by sampling importance resampling technique

	Mean	SD	LB	Median	UB
beta	3.0376	0.39	2.35	3.05	4.16
log.shape	0.2716	0.34	-0.37	0.28	0.87
Deviance	130.539	2.045	128.40	129.85	135.51

V.   SIMULATION STUDY

We implement Random-Walk Metropolis algorithm to simulate from the joint posterior distribution. A proposed future estimate called proposal of the joint target density is calculated and then the ratio of proposed to the current joint posterior density called α is calculated. The proposed state is accepted over the current state if the proposed state is an improvement over the current state. If the proposed state is rejected, then current state is repeated so that another proposal may be estimated at the next iteration. Here, the proposal distribution is multivariate normal. The whole process has been implemented using `LaplacesDemon` function. This function is however assisted by the function `LaplaceApproximation`. The R codes are as follows:

```
Initial.Values<-as.initial.values(M1)
###Fitting with LaplacesDemon###
M11<-LaplacesDemon(Model,MyData,Initial.Values)
```

Output has been reported in Table 3.

Table 3   Summarizes marginal posterior distributions of the parameters, deviance calculated only on the stationary samples

	Mean	SD	LB	Median	UB
beta	3.0623	0.42	2.31	3.03	3.95
log.shape	0.2701	0.33	-0.41	0.28	0.91
Deviance	130.449	2.05	128.41	129.85	136.06

With the simulated values of the parameters with `SIR` and those with the `LaplacesDemon` function, marginal posterior density plots are made by using `plot` function and are shown in Figure 1. Also, normal approximation of the parameter distributions have been graphed over it.
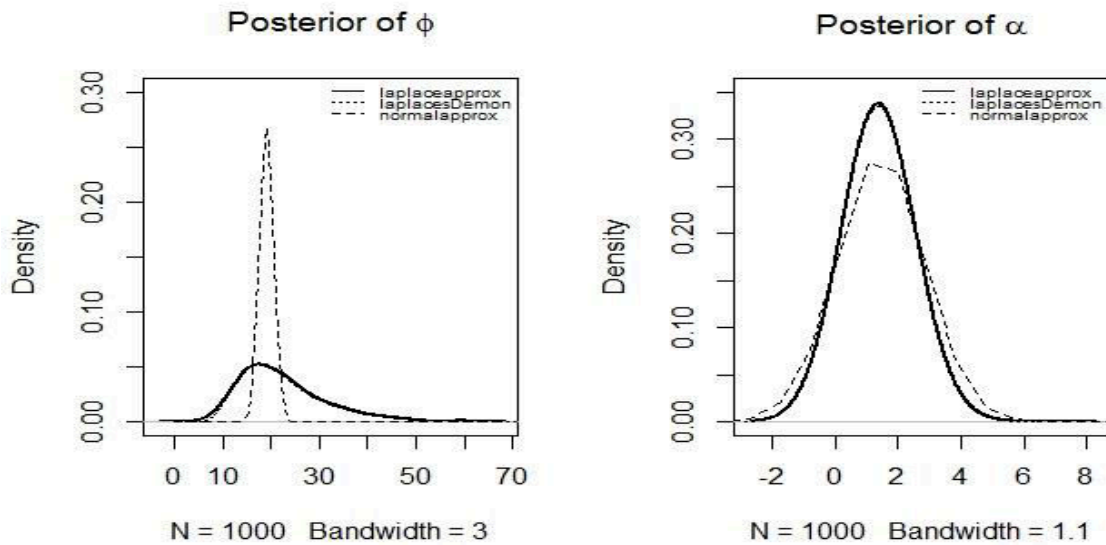

Fig. 1: Left panel of the figure depicts the marginal posterior density of scale parameter and that of the shape parameter is in the right panel of the figure.

The reliability function and hazard function curves have been shown in Figure 2. The reliability of the electronic components are decreasing with the passage of time. The increasing hazard rate specifies the way in which the instantaneous probability of failure of the components is changing with the time.
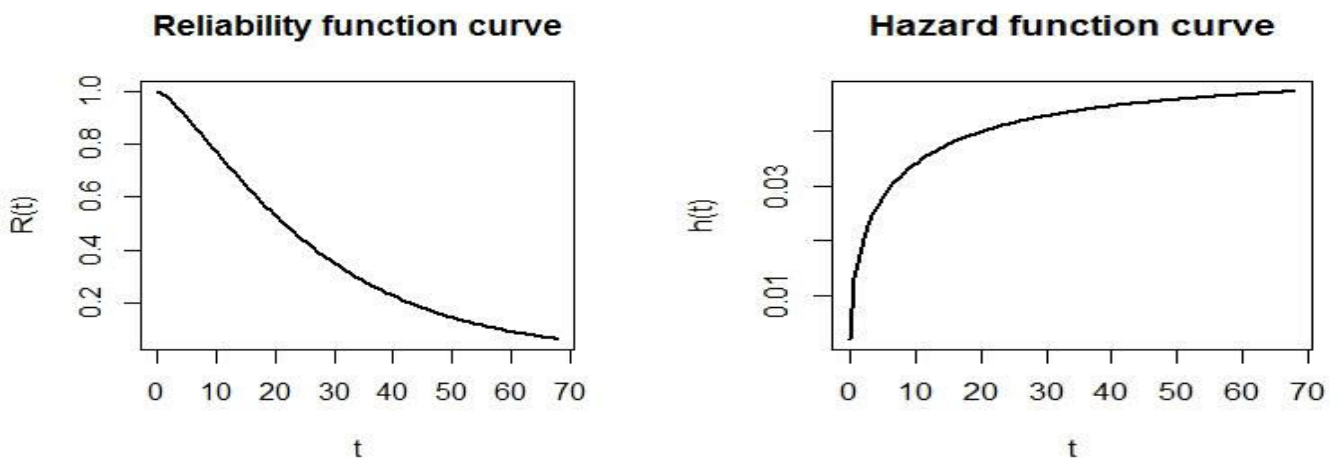

Fig. 2: Reliability trend for the electronic components is being shown by the left panel of the figure while the right panel shows an increasing hazard functions.

VI. FAILURE TIME REGRESSION MODELING

A model with explanatory variables (covariates) can sometimes best describe the heterogeneity in a population. It explains or predicts why some units survive a long time whereas others fail quickly. The main objective behind regression modeling is to exploit the relationship between failure-time and the explanatory variables. One of the way to create a regression model is to allow the model parameter(s) to depend on covariates. We illustrate this with insulating fluid failure-time data discussed by [7]. The data consists of breakdown times (in minutes) of 76 specimens of type of an electrical insulating fluid which were tested at seven voltage levels, $v = 26, 28, 30, 32, 34, 36$ kilovolts. The main objective of this study was to draw a relationship between time to breakdown and voltage stress.

A. Data creation
We create the data y in a vector form as below:

```
y<-
c(5.79,1579.52,2323.7,68.85,426.07,110.29,108.29,1067.6,17.05,22.66,21.02,175.88,
139.07,144.12,20.46,43.40,194.90,47.3,7.74,0.40,82.85,9.88,89.29,215.10,2.75,0.79,
15.93,3.91,0.27,0.69,100.58,27.80,13.95,53.24,0.96,4.15,0.19,0.78,8.01,31.75,7.35,
6.50,
8.27,33.91,32.52,3.16,4.85,2.78,4.67,1.31,12.06,36.71,72.89,1.97,0.59,2.58,1.69,2.
71,25.50,0.35,0.99,3.99,3.67,2.07,0.96,5.35,2.90,13.77,0.47,0.73,1.40,0.74,0.39,1.
13,
0.09,2.38)
voltage<-
c(rep(26,3),rep(28,5),rep(30,11),rep(32,15),rep(34,19),rep(36,15),rep(38,8))
```

Engineering background suggests a power law model in this case with

$$\log \phi(x) = \beta_0 + \beta_1 x$$

where,

$$x = \log(\text{cov ariate})$$

Thus, we make a logarithmic transformation of voltage levels.

```
x<-log(voltage)
J<-2
N<-length(y)
X<-cbind(1,x)
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,N=N,X=X,mon.names=mon.names,parm.names=parm.names,y=y)
```

B. Model Specification
Within this subsection, we specify the weakly informative prior distributions for the regression coefficients and the shape parameter along with the log-likelihood and the log-posterior.

```
Model<-function(parm,Data)
{
###Parameters###
beta<-parm[1:Data$J]
```

```
shape<-exp(parm[Data$J+1])
###Log(Prior-densities)###
beta.prior<-dnormv(beta,0,10000,log=T)
shape.prior<-dhalfcauchy(shape,25,log=T)
scale<-exp(tcrossprod(Data$X,t(beta)))
###Log-Likelihood###
LL<-sum(dgamma(Data$y,shape,scale=scale,log=TRUE))
###Log-Posterior###
LP<-LL+sum(beta.prior)+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
yhat=rgamma(76,shape,scale),parm=parm)
return(Modelout)
}
```

With the specification of initial values for the parameters, we finally fit the model to the data with the `LaplaceApproximation` function.

```
M2<-LaplaceApproximation(Model,Initial.Values,MyData,Iterations=40000)
```

Results have been shown in Table 4 and Table 5.

Table 4   Posterior modes and sd of the gamma parameters based on laplace approximation

	Mean	SD	LB	UB
beta[1]	64.5446	5.35	53.83	75.25
beta[2]	-17.4938	1.53	-20.55	-14.43
log.shape	-0.3678	0.14	-0.643	-0.091

Table 5   Posterior mean values obtained via `SIR` function

	Mean	SD	LB	Median	UB
beta[1]	65.4317	5.36	54.99	65.46	75.35
beta[2]	-17.7339	1.53	-20.58	-17.75	-14.77
log.shape	-0.3900	0.14	-0.67	-0.39	-0.12
Deviance	606.2026	2.30	603.46	605.55	611.79

The same data has been studied through simulations. Here, the algorithm used is Adaptive-Mixture Metropolis. First, AMM updates a scatter matrix which is based on the cumulative current parameters and the cumulative associated outer-products, and these are used to generate a multivariate normal proposal. However, the proposal is a mixture. The two mixture components are adaptive multivariate and static/symmetric univariate proposals. At each iteration, the mixture is determined with a mixture weight. The mixture weight must be in the interval (0, 1], and it defaults to 0.05, as in [8]. A lower weight is associated with more adaptive multivariate proposals and a higher value of the mixture weight is associated with more static/symmetric univariate proposals. `LaplacesDemon` function has been used for this purpose. The R codes for it are as follows:

```
Initial.Values <- as.initial.values(M2)
M22<- LaplacesDemon(Model, Data=MyData, Initial.Values,
Iterations=6e+05, Status=277777, Thinning=600,
Algorithm="AMM", Specs=list(Adaptive=588, B=NULL, Periodicity=117647,
w=0.05))
```

The parameter estimates are reported in Table 6.

Table 6   Posterior mean values along with the 0.025, 0.5, 0.975 quantiles based on stationary samples

	Mean	SD	LB	Median	UB
beta[1]	64.4673	5.20	55.27	63.70	75.46
beta[2]	-17.4600	1.48	-20.55	-17.23	-14.71
log.shape	-0.3942	0.14	-0.67	-0.39	-0.11
Deviance	606.1653	2.44	603.40	605.54	612.39

Based on the simulated values, marginal posterior densities of all the three parameters have been represented in Figure 3. The classical interpretation of density of parameter(s) estimates have been shown on the same plot.
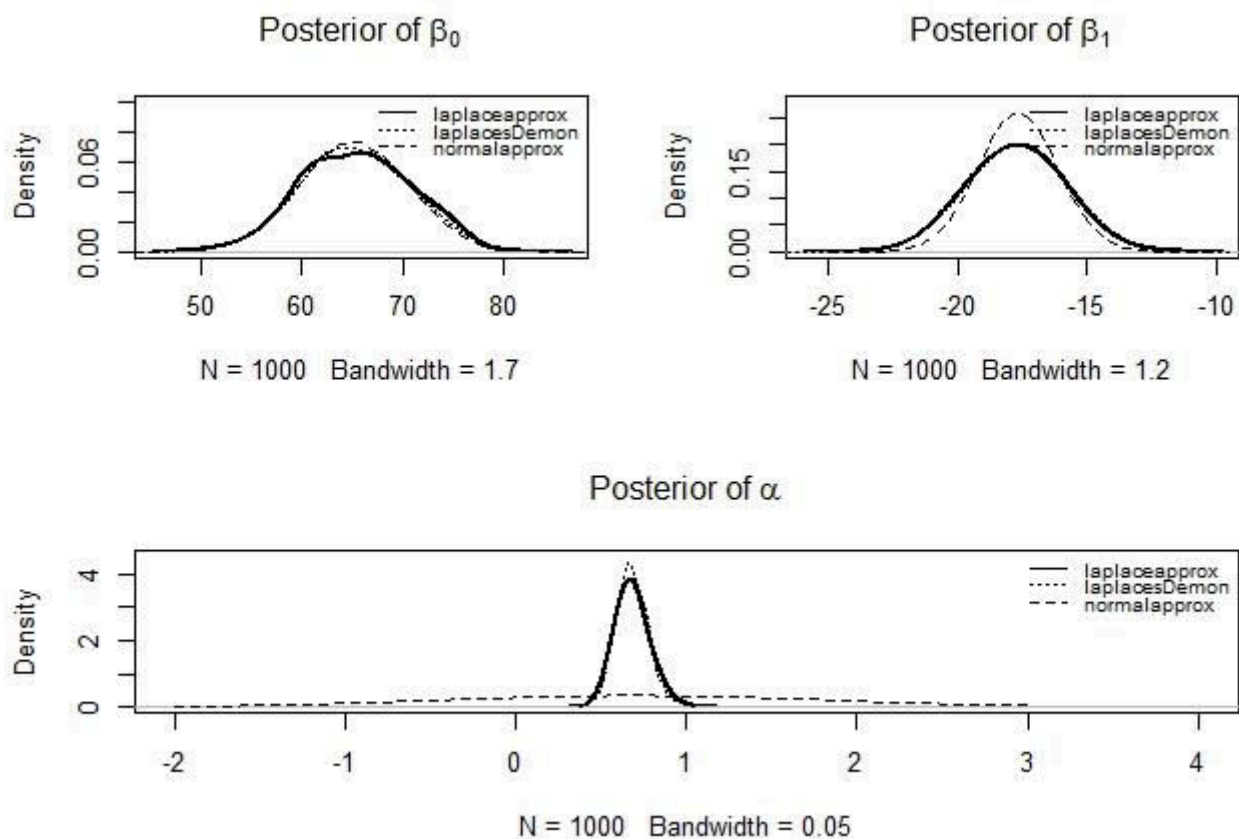

Fig. 3: Marginal posterior densities of regression coefficients and shape parameter**.**

However, the expected mean time until the first failure of insulating fluid under different voltage levels is decreasing at a steep rate and has been shown in Figure 4.
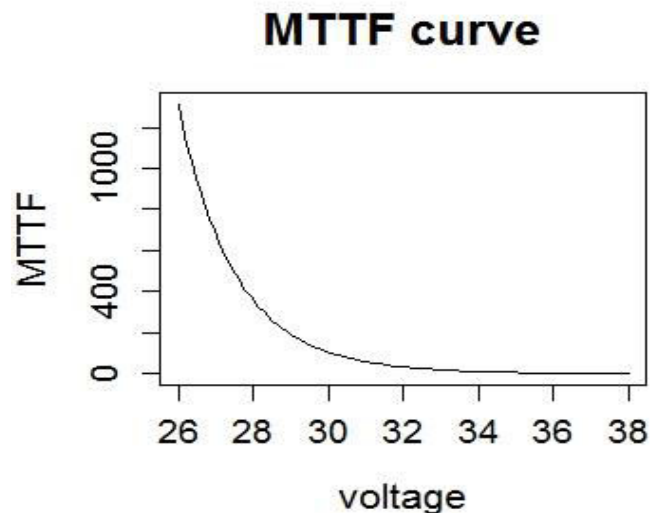
Fig. 4: Trend of mean time to failure of electrical insulating fluid example
under the gamma model.

VII.   CONCLUDING REMARKS

This paper introduces the contributed R package LaplacesDemon that facilitates high dimensional Bayesian inference. It has been written entirely in R and has a remarkable provision for user-defined probability models. The main body of the article comprises of written and distributed codes for gamma modeling of real datasets with Laplace approximation. It has been found that Laplace approximation is a suitable alternative to the high-dimensional integrations and R provides a simple function namely `LaplaceApproximation` to implement it. Also, we have attempted to provide a true picture of the marginal posteriors in contrast to the one provided by the asymptotic theory of frequentist approach. It is evident from the results that Bayesian methods based on non-informative (or weakly-informative) priors are equally compatible to the classical methods of inference.

REFERENCES
[1]. L. Tierney, R. E. Kass, J. B. Kadane, "Fully exponential Laplace approximations to expectations and variances of non-positive functions", Journal of the American Statistical Association, Vol. 84, No. 407, pp. 710-716, 1989.
[2]. R Development Core Team, "R: A language and environment for statistical computing", R Foundation for Statistical Computing, Vienna, Austria, 2012.
[3]. L. Tierney, J. B. Kadane, "Accurate approximations for posterior moments and marginal densities", Journal of the American Statistical Association, Vol. 81, No. 393, pp. 82-86, 1986.
[4]. D. V. Lindley, "Approximate Bayesian methods", in Bayesian Statistics, eds. J. M. Bernardo. M. H. Degroot, D. V. Lindley, and A. M. F. Smith, Valencia, Spain: University Press, pp. 223-245, 1980.
[5]. Statisticat, LLC, LaplacesDemon: "Complete environment for Bayesian Inference", R package version 13.03.04, URL http://www.bayesian-inference.com, 2013.
[6]. J. F. Lawless, "Statistical Models and Methods for Lifetime Data", Wiley, New York, 1982.
[7]. W. B. Nelson, "Statistical methods for accelerated lifetest data- the inverse power law model", General Electric Co. Technical Report 71-C-011. Schenectady, New York, 1970a.
[8]. Roberts G, Rosenthal J, "Examples of Adaptive MCMC", Computational Statistics and Data Analysis, 18, 349-367, 2009.
[9]. J. F. Lawless, "Statistical Models and Methods for Lifetime Data", 2nd ed.,Wiley, New York, 2003.