

For regression coefficient β , a normal prior distribution is assigned with parameter 0 and standard deviation 1000, that is,

$$\beta_j \sim N(0,1000) \quad (2.9)$$

Following Equation 2.7, 2.8 and 2.9 and using Bayes theorem, the joint posterior distribution is given as,

$$p(\alpha, \beta | y, X) = \prod_{i=1}^n \left\{ \left[\frac{\alpha}{e^{x_i \beta}} \left(1 + \frac{y_i}{e^{x_i \beta}} \right)^{-(\alpha+1)} \right]^{\delta_i} \left[\left(1 + \frac{t_{c_i}}{e^{x_i \beta}} \right)^{-\alpha} \right]^{(1-\delta_i)} \right\} \cdot \frac{2\sigma}{\pi(\alpha^2 + \sigma^2)} \cdot \prod_{j=1}^p \left(\frac{1}{\sqrt{2\pi * 1000}} \exp \left(\frac{-\beta_j^2}{2 * 1000} \right) \right) \quad (2.10)$$

Thus, marginal posterior of α

$$p(\alpha | y, X) = \int_{-\infty}^{\infty} p(\alpha, \beta | y, X) d\beta \quad (2.11)$$

and marginal posterior of β

$$p(\beta | y, X) = \int_0^{\infty} p(\alpha, \beta | y, X) d\alpha \quad (2.12)$$

3. THE EXPONENTIAL LOMAX MODEL

A generalization to the Lomax distribution was suggested by (Abdul-Moniem, Abdel-Hameed, 2012) using Lehmann alternative type I proposed by Gupta et al. (1998). The three parameter EL pdf (for $t > 0$) is defined by

$$f(t; \nu, \alpha, \lambda) = \frac{\nu \alpha}{\lambda} \left(1 + \frac{t}{\lambda} \right)^{-(\alpha+1)} \left\{ 1 - \left(1 + \frac{t}{\lambda} \right)^{-\alpha} \right\}^{\nu-1} \quad (3.1)$$

Cumulative distribution function (cdf),

$$F(t; \nu, \alpha, \lambda) = \left\{ 1 - \left(1 + \frac{t}{\lambda} \right)^{-\alpha} \right\}^{\nu} \quad (3.2)$$

Survival function,

$$s(t; \nu, \alpha, \lambda) = 1 - \left\{ 1 - \left(1 + \frac{t}{\lambda} \right)^{-\alpha} \right\}^{\nu} \quad (3.3)$$

Hazard function,

$$h(t; \nu, \alpha, \lambda) = \frac{\frac{\nu\alpha}{\lambda} \left(1 + \frac{t}{\lambda}\right)^{-(\alpha+1)} \left\{1 - \left(1 + \frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu-1}}{1 - \left\{1 - \left(1 + \frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu}} \tag{3.4}$$

The joint posterior distribution of exponential Lomax distribution is,

$$p(\nu, \alpha, \beta | y, X) = \prod_{i=1}^n \left[\left\{ \frac{\nu\alpha}{e^{x_i\beta}} \left(1 + \frac{y_i}{e^{x_i\beta}}\right)^{-(\alpha+1)} \left\{1 - \left(1 + \frac{y_i}{e^{x_i\beta}}\right)^{-\alpha}\right\}^{\nu-1} \right\}^{\delta_i} \left\{1 - \left\{1 - \left(1 + \frac{t_{c_i}}{e^{x_i\beta}}\right)^{-\alpha}\right\}^{\nu}\right\}^{1-\delta_i} \right] \tag{3.5}$$

$$\frac{2\gamma}{\pi(\alpha^2 + \gamma^2)} \cdot \frac{2\gamma}{\pi(\alpha^2 + \gamma^2)} \cdot \prod_{j=1}^p \left(\frac{1}{\sqrt{2\pi * 1000}} \exp\left(\frac{-\beta_j^2}{2 * 1000}\right) \right)$$

Thus, marginal posterior of α

$$p(\alpha | y, X) = \int_0^{\infty} \int_{-\infty}^{\infty} p(\alpha, \nu, \beta | y, X) d\beta d\nu, \tag{3.6}$$

Marginal posterior of β

$$p(\beta | y, X) = \int_0^{\infty} \int_0^{\infty} p(\alpha, \nu, \beta | y, X) d\alpha d\nu, \tag{3.7}$$

Marginal posterior of ν

$$p(\nu | y, X) = \int_{-\infty}^{\infty} \int_0^{\infty} p(\alpha, \nu, \beta | y, X) d\beta d\alpha \tag{3.8}$$

Since, Lomax and exponential Lomax contain two and three parameters respectively; hence, the evaluation of the joint posterior density which contains censoring mechanism also, will become a very difficult job. Consequently, some rigorous computational methods are required to solve the problem. To keep this in mind Tierney and Kadane (1986) suggested the use of Laplace approximation method. The Laplace approximation is a family of asymptotic techniques used to approximate integrals (Statisticat LLC 2015).

4. LAPLACE APPROXIMATION

Laplace approximation technique was originally presented in Laplace 1774 (reprinted Stigler 1986); the most frequently cited paper on the subject is the rather more recent one by Tierney and Kadane (1986). Laplace Approximation dates back to Laplace (1774, 1814), and is used to approximate the posterior moments of integrals. A short and informal description of Laplace approximation method is: Suppose $-h(\theta)$ is a

smooth, bounded and unimodal function with a maximum at $\hat{\theta}$ where θ is a scalar and we wish to evaluate the integral

$$I = \int q(\theta) \exp(-nh(\theta)) d\theta, \quad \theta \in \Theta \quad (4.1)$$

For any pdf that is smooth and well peaked around its point of maxima, Laplace proposed to approximate it by a normal pdf. It's a simple 2-term Taylor expansion trick on the log pdf. If $\hat{\theta}$ denotes the point of maxima of a pdf $h(\theta)$, then it is also the point of maxima of the log-pdf $q(\theta) = \log h(\theta)$ and we can write:

$$\begin{aligned} h(\theta) &\approx h(\hat{\theta}) + (\theta - \hat{\theta})h'(\hat{\theta}) + \frac{1}{2}(\theta - \hat{\theta})^2 h''(\hat{\theta}) + \dots \\ &= h(\hat{\theta}) + 0 + \frac{1}{2}(\theta - \hat{\theta})^2 h''(\hat{\theta}) + \dots \end{aligned} \quad (4.2)$$

$$q(\theta) = q(\hat{\theta}) + \frac{1}{2}(\theta - \hat{\theta})^2 q''(\hat{\theta}) + \dots$$

where $h''(\hat{\theta})$ is the Hessian of the log density evaluated at the mode. Now the integral I can be approximated by

$$I \approx (2\pi)^{1/2} n^{-1/2} \sigma \exp(-nh(\hat{\theta})) \quad (4.3)$$

Where $\sigma = \left[\frac{\partial^2 h}{\partial \theta^2} \Big|_{\hat{\theta}} \right]^{-1/2}$. Intuitively, if $\exp[-nh(\theta)]$ is very peaked about $\hat{\theta}$, then the

integral can be well approximated by the behaviour of the integrand near $\hat{\theta}$. More formally, it can be shown that

$$I = \hat{I}[1 + O(n^{-1})] \quad (4.4)$$

One general purpose of optimization algorithm for finding this mode is provided by Newton's method. Suppose one has guess at the posterior mode θ^0 . If θ^{t-1} is the estimate at the mode at the $t-1$ iteration of the algorithm, then the next iteration is given by,

$$\theta^t = \theta^{t-1} - [h''(\theta^{t-1})]^{-1} h'(\theta^{t-1}),$$

where $h'(\theta^{t-1})$ and $h''(\theta^{t-1})$ are the gradient and Hessian of the log density evaluated at the current guess at the mode. One continues these iterations until convergence. There are many alternative algorithms available for finding the posterior mode. In this paper Nelder and Mead (1965) method will be used. Since Nelder-Mead is a derivative free method and less sensitive for guess values, it works well in most of the practical situations. So, this method has been used in the paper as an argument for optimization method in `LaplaceApproximation` function.

5. SURVIVAL DATA: VETERAN'S ADMINISTRATION LUNG CANCER DATA

In this data, males with advanced inoperable lung cancer were randomized to either a standard or test chemotherapy. Only 9 of the 137 survival times were censored. The data is available in survival package and is presented in Kalbeisch and Prentice (1980, 2002). A portions of the data is analyzed by several other authors (Prentice,1973; Chen et al., 2002; Murphy et al., 1997; Bennett, 1983). In this analysis, the 137 subjects who completed the randomized portion of the trial and for whom complete covariate information was available are considered. Six covariates are available which include treatment, age, tumor cell type (adeno, small cell, squamous or large), time between initial diagnosis and enrolment in the trial, Karnofsky performance status, and prior therapy attempted (yes/no).

1. Treatment: 0 = standard, 1 = test.
2. Type of tumour: 1 = squamous, 2 = small cell, 3 = adeno, 4 = large cell.
3. Age in years.
4. Prior therapy: 0 = no, 1 = yes
5. diagtime: Time in months from diagnosis to randomization.
6. Performance status: Karnofsky performance score (100=good).

6. BAYESIAN MODELING OF LOMAX DISTRIBUTION

For Bayesian modeling of Lomax distribution on Veteran's administration lung cancer data involves the following steps:

1. Creation of lung cancer data.
2. Specification of model for Lomax distribution
3. Generation of initial values
4. Fitting of Lomax distribution using `LaplaceApproximation` function for analytic approximation and then `LaplacesDemon` function for Markov chain Monte Carlo simulation

By performing the above steps one by one, a complete posterior picture has been obtained by using two methods namely, Nelder-Mead optimization method for analytic approximation and independent Metropolis algorithm for simulation. Implementation has been made by using `LaplacesDemon` package. These steps would be discussed in the following sections.

6.1. Creation of lung cancer data

For illustrative purpose, a real survival data set called veteran that is provided with the survival package is

used . The survival data, called, veteran contains six regressor variable i.e celltype, karno, diagtime, age, prior and trt, and its vector have been defined by objects names x1; x2; x3; x4; x5 and x6, respectively, using an extraction operator \$.

```
library(LaplacesDemon)
library(survival)
data(veteran)
y<-veteran$time
x1<-veteran$karno
x2<-veteran$celltype
x3<-veteran$diagtime
x4<-veteran$age
x5<-veteran$prior
x6<-veteran$trt
censor<-veteran$status
N<-137
X<-cbind(1,x1,x2,x3,x4,x5,x6)
J<-7
```

X is called the model matrix which contains six columns of regressor variables and a column of 1's is also inserted into it as an intercept. Here $J = 7$, as X has seven columns.

```
mon.names<-c("LP", "shape")
parm.names<-
as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-
list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
y=y,censor=censor)
```

Each parameter must have a name specified in the vector `parm.names`, and parameter names must be included with the data. The object is created by using the function `as.parm.names`. The object `mon.names` is meant for the variables to be

monitored. Finally, lung cancer data in `LaplacesDemon` has been created with object name `MyData` which contains the list of `J`, `X`, `mon.names`, `parm.names`, `y`, and a vector of censored observation called `sensor`.

6.2. Model specification for Lomax distribution

To use `LaplacesDemon` package, one must specify a model. Let's consider a regression model, which is often denoted as:

$$y \sim \text{Lomax}(\alpha, \lambda)$$

$$\log(\lambda) = X\beta$$

The response variable, `y` follow Lomax distribution with parameter shape and scale, the scale parameter λ is equal to the cross product of design matrix `X` and the parameter β .

`LaplacesApproximation` deterministically maximizes the logarithm of the unnormalized joint posterior density as specified in the `Model` function. In Bayesian inference, the logarithm of the unnormalized joint posterior density is proportional to the sum of the log-likelihood and logarithm of the prior densities:

$$\log[p(\theta | y)] \propto \log[p(y | \theta)] + \log[p(\theta)]$$

During each iteration `LaplacesApproximation` passes two arguments to `Model`: `parm` and `Data`. These arguments are specified in the beginning of the function (i.e first line of `Model`). After defining parameters of distribution, the next step is the assigning of prior to them. To obtain log-likelihood, we need density function and survival function which are defined as `f1` and `s1`, respectively. Then, the `Model` function is evaluated and the logarithm of the unnormalized joint posterior density is calculated as `LP`. This function returns an object called `Modelout`, which is a list of the objects, log-posterior (`LP`), deviance = $-2 * \text{log-likelihood}$ (`Dev`), monitored variable (`Monitor`), `yhat` and `parm`.

```
Model<-function(parm, Data)
{
#Parameters
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta, 0, sqrt(1000), log=T))
shape.prior<-dhalfcauchy(shape, 20, log=T)
# Loglikelihood
mu<-tcrossprod(beta, Data$X)
```

```

scale<-exp(mu)
f1<-log(shape)-log(scale)-(shape+1)*log(1+y/scale)
s1<--shape*log(1+y/scale)
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)

## Log-posterior
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=
r1omax(length(y),shape,scale),parm=parm)
return(Modelout)
}

```

6.3 Initial values

LaplaceApproximation requires a vector of initial values for the parameters. Each initial value is a starting point for the estimation of a parameter. Here, all initial values are set to zero and LaplaceApproximation function will optimize initial values using Nelder-Mead method.

```
Initial.Values<-c(rep(0,J),log(1))
```

However, we recommend better initial values obtain from fitting multiple regression model using logarithm of response variable.

```
Initial.Values<-
c(coef(lm(log(y)~x1+as.numeric(x2)+x3+x4+as.numeric(x5)+x
6)),
log(1))
```

The performance of these initial values in terms of convergence could be seen in Section 6.5 through trace plots reported in Figure 2 and 3.

6.3. Fitting of data using LaplaceApproximation function

The LaplaceApproximation function deterministically maximizes the logarithm of the unnormalized joint posterior density with one of several optimization algorithms. The goal of Laplace approximation is to estimate the posterior mode and variance of each parameter. Here, an output object called M1 will be created as a result of using the LaplaceApproximation function. The object M1 gives two summaries, summary1 is obtained by Nelder-mead method and summary2 is obtained by

sampling importance resampling method.

```
M1<-
LaplaceApproximation(Model,Initial.Values,Data=MyData,
Method="NM",Iterations=100000)
```

6.4. Summarization of output

The first part of the Table 1 summarizes the point-estimated posterior modes. Uncertainty around the posterior mode is estimated from the asymptotic covariance matrix. Rows are parameters i.e celltype, karno, diagtime, age, prior and treatment. The following columns are included: Mode, SD (Standard Deviation), LB (Lower Bound), and UB (Upper Bound). The bounds constitute a 95% credible interval. The second part of the Table 1 summarizes the posterior samples drawn with sampling importance resampling (SIR) when sir=TRUE, given the point-estimated posterior mode and the covariance matrix obtain from LaplaceApproximation. Again rows are parameters. The following columns are included: Mean, SD (Standard Deviation), LB (Lower Bound), and UB (Upper Bound). The bounds constitute a 95% credible interval.

Table 1. Posterior summaries of the lung cancer data using the function LaplaceApproximation, which is based on asymptotic approximation theory, and posterior summary due to sampling importance resampling method respectively, using the same function.

Optimization- Nelder and Mead				
	Mode	SD	LB	UB
Intercept	4.659	1.231	2.198	7.121
Karno	0.037	0.005	0.026	0.048
Celltype	-0.103	0.088	-0.279	0.072
Diagtime	-0.001	0.009	-0.018	0.017
Age	0.003	0.010	-0.017	0.023
Prior therapy	0.009	0.023	-0.038	0.056
Treatment	-0.153	0.202	-0.557	0.251
log.shape	1.967	0.682	0.602	3.332

Simulation- Sampling Importance Resampling

	Mean	SD	LB	UB
Intercept	4.601	1.221	2.305	7.127
Karno	0.042	0.011	0.037	0.058
Cell type	-0.099	0.095	-0.281	0.089
Diagnosis time	0.002	0.013	-0.022	0.021
Age	0.004	0.011	-0.025	0.028
Prior therapy	0.012	0.024	-0.048	0.061
Treatment	-0.161	0.213	-0.557	0.264
shape	8.651	6.755	2.443	29.858

In Figure 2 convergence starts from 1000th iteration whereas in Figure 3 algorithm converges at 600th iteration. Hence, it shows 40% increase in the efficiency when initial values are obtained by using function lm as compared to the initial values taken as zero.

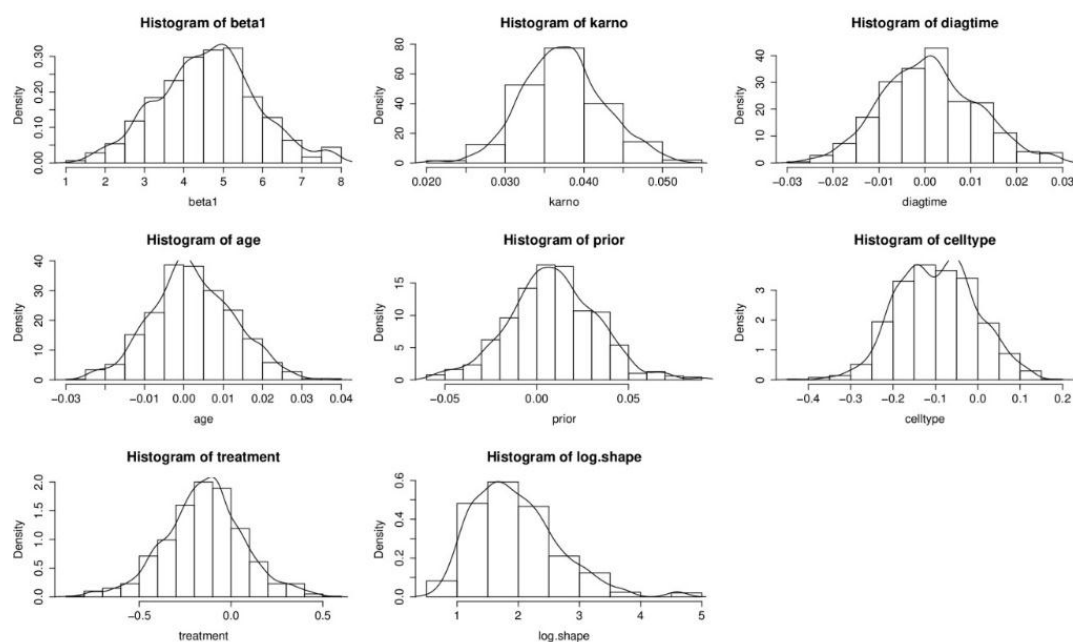


Figure 1. Histograms and posterior densities of all the parameters and regression coefficients betas for Lomax distribution.

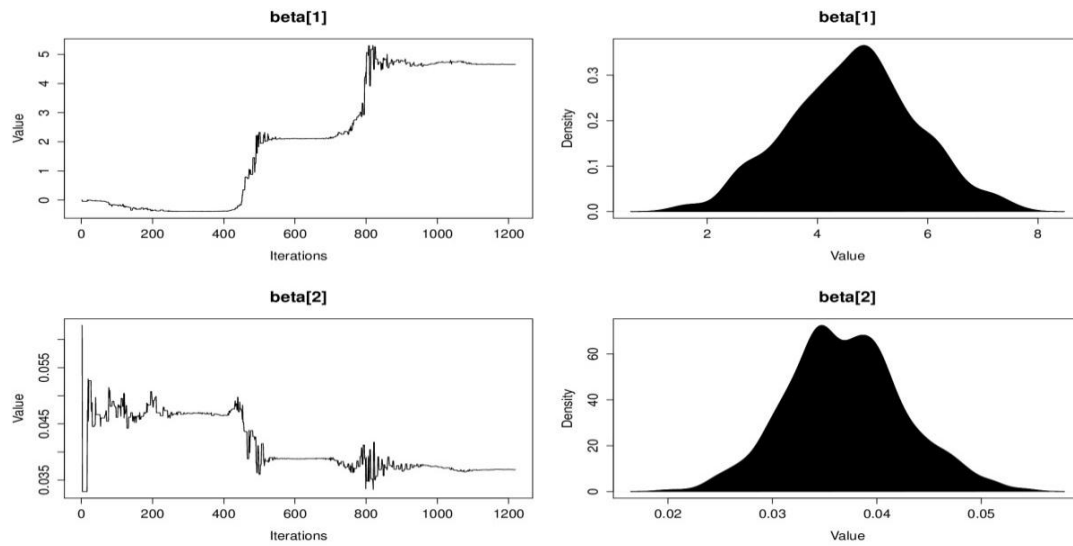


Figure 2. Trace and posterior density plots of Lomax distribution at initial values zero for all the parameters.

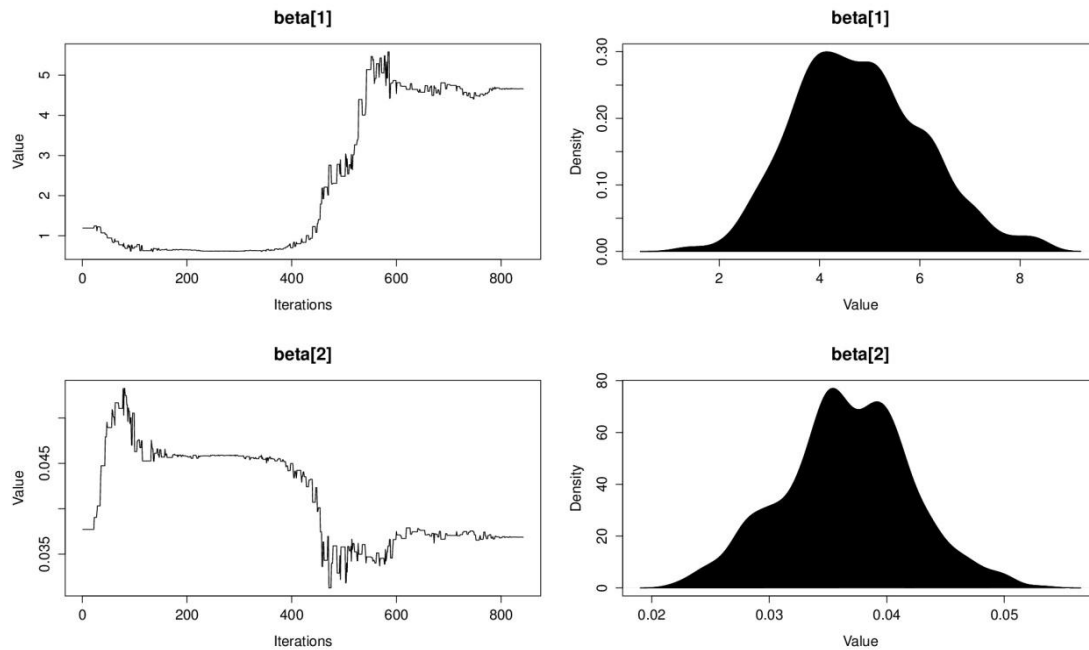


Figure 3. Trace and posterior density plots of Lomax distribution at initial values obtain from fitting multiple regression model using logarithm of response variable.

6.5. Fitting of data using LaplacesDemon function

Now, we have to explore the same veteran data using function LaplacesDemon. The LaplacesDemon function is the main function of LaplacesDemon package. It

maximizes the logarithm of the unnormalized joint posterior density with MCMC and provides samples of the marginal posterior distributions, deviance, and other monitored variables. In `LaplacesDemon` function there is an argument called `Algorithm`, here the algorithm used for simulation from joint posterior distribution is independent-Metropolis algorithm. Multivariate normal has been treated as a proposal distribution $q(\theta)$. Here, the proposal distribution does not depend on the previous state of the chain. The IM algorithm is efficient when the proposal is a good approximation of the target posterior distribution. Good independent proposal densities can be based on LaplaceApproximation (Tierney and Kadane, 1986; Tierney et al., 1989; Erkanli, 1994). Thus, a generally successful proposal can be obtained by a multi-variate normal distribution with mean equal to the posterior mode and precision matrix.

$$H(\hat{\theta}) = \left(- \frac{\partial^2 \log p(\theta | y)}{\partial \theta_i \partial \theta_j} \Big|_{\theta = \hat{\theta}} \right)$$

That is, minus the second derivative matrix of the log-posterior density

$$\log p(\theta | y) = \log p(y | \theta) + \log p(\theta)$$

Evaluated at the posterior mode $\hat{\theta}$. Consequently, an efficient proposal is given by,

$$q(\theta) = N(\hat{\theta}, [H(\hat{\theta})]^{-1})$$

The posterior mode can be evaluated by some of the efficient methods provided in LaplaceApproximation with object M1. Among the optimization methods the performance of Nelder-Mead (1965) seems to be the best. Thus object M1 is created by making use of LaplaceApproximation with the choice of optimization method of Nelder-Mead ‘‘N-M’’. When low information prior is used, then an adequate proposal can be obtained by setting the mean equal to the corresponding maximum-likelihood estimator (MLE) and the precision equal to its observed Fisher information matrix.

The acceptance probability, when proposing a transition from θ to $\hat{\theta}$, is given by

$$\alpha = \min \left(1, \frac{p(\theta' | y)q(\theta)}{p(\theta | y)q(\theta')} \right)$$

Which can be expressed as,

$$\alpha = \min \left(1, \frac{w(\theta')}{w(\theta)} \right)$$

where $w(\theta) = p(\theta | y) / q(\theta)$ is the ratio between the target and the proposal distribution and is equivalent to the importance weight used in importance sampling (Ntzroufras, 2009). This theory is implemented in `LaplacesDemon` with object name M2. M2 is an object of class `demonoid`, which means that since it has been assigned a customized class, other functions have been custom-designed to work with it. This M2 object contains an argument, called `Covar`, the covariance matrix may be input from the `LaplaceApproximation` function M1. Another argument is `Specs`, which

accepts the list of specifications for the MCMC algorithm.

```
Initial.Values<-as.initial.values (M1)

M2<-LaplacesDemon (Model, Data=MyData, Initial.Values,

Covar=M1$Covar, Iterations=2000, Status=F, Thinning=1,

Algorithm="IM", Specs=list (mu=M1$Summary1 [1:length (Initial
.Values), 1]))

M2
```

6.6. Output by simulations

The `LaplacesDemon` function also generates two posterior summaries. `Summary1` gives the the marginal posterior distributions of the parameters, deviance, and monitored variables. The following summary statistics are included: mean, standard deviation, MCSE (Monte Carlo Standard Error), ESS which is the effective sample size due to autocorrelation, and finally the 2.5%, 50%, and 97.5% quantiles are reported in Table 2. MCSE is essentially a standard deviation around the marginal posterior mean that is due to uncertainty associated with using MCMC. `Summary2` is identical to `Summary1`, except that it is calculated only on the stationary samples and it ensures that convergence has been reached its equilibrium distribution.

Table 2. Simulated posterior summary of lung cancer data by independent Metropolis algorithm under the assumption of Lomax model.

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	4.529	0.475	0.012	2000.00	3.655	4.576	5.576
Karno	0.042	0.001	0.0023	2000.00	0.034	0.042	0.044
Cell type	-0.101	0.043	0.001	2000.00	-0.175	-0.101	-0.034
Diagnosis time	-0.002	0.003	0.002	2000.00	-0.013	-0.001	0.014
Age	0.003	0.001	0.002	2000.00	-0.001	0.001	0.034
Prior therapy	0.021	0.011	0.001	2000.00	-0.013	0.011	0.034
Treatment	-0.167	0.083	0.003	2000.00	-0.321	-0.174	-0.023
shape	7.585	2.086	0.051	2000.00	4.373	7.284	12.285

The convergence of the algorithm can also be determined by MCSE, Table 2 shows very small values of this error which indicates that we have calculated the quantity of interest with high precision. From Figure 4, convergence can also be monitored through the trace and autocorrelation plots. Trace plots in the leftmost panel are very much convincing in terms of convergence as all generated values within a parallel zone. Monitoring autocorrelation is also very useful as it is evident from rightmost panel of the Figure 4 that the low values indicate fast convergence.

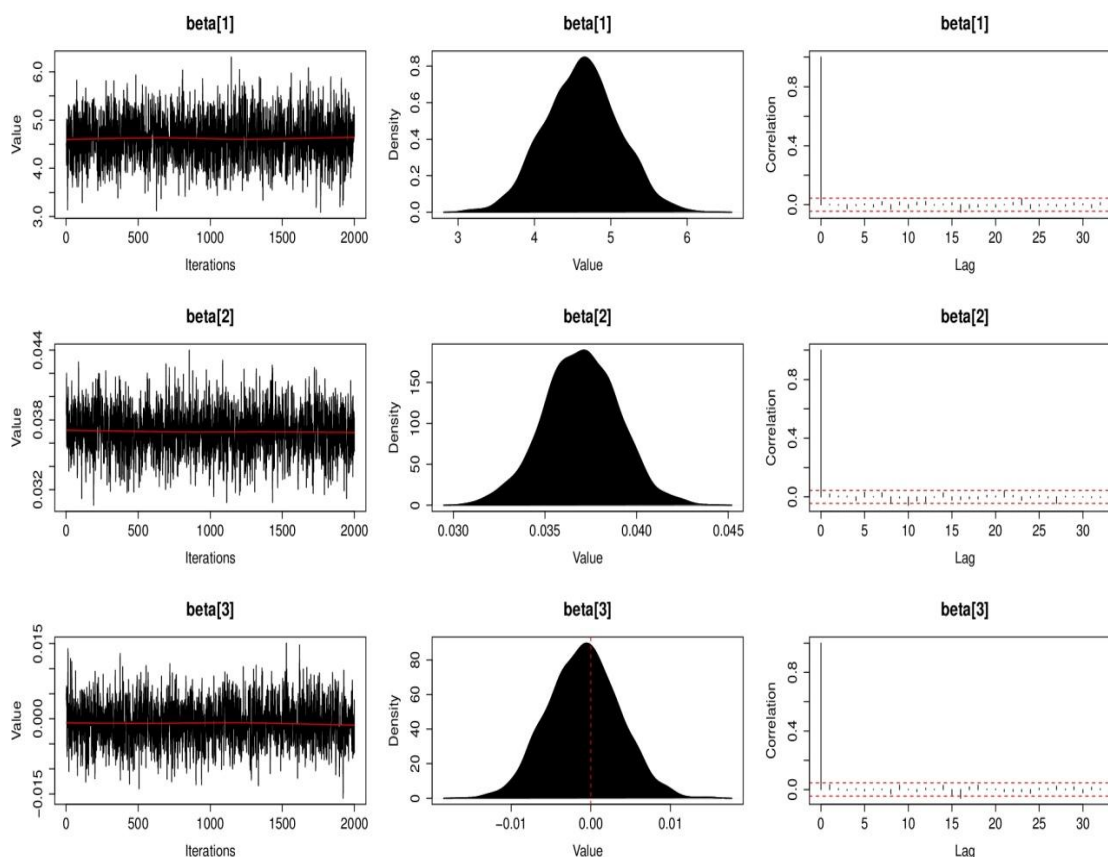


Figure 4. Simulated posterior density plots of the parameter of Lomax distribution. The leftmost is the trace plot, the middlemost is the density plot and the rightmost is the auto correlation plot, showing low autocorrelation at different lags.

7. BAYESIAN MODELING OF EXPONENTIAL LOMAX DISTRIBUTION

7.1. Creation of lung cancer data for exponential Lomax distribution

In this section, data for exponential Lomax has been created with object name `MyData` which contains the list of vectors, that are, model matrix X , survival time vector y , monitoring variables `mon.names`, list of parameters of the model `parm.names`, vector of censored observations `ensor`. R commands for the creation of veteran's lung

cancer data for exponential Lomax distribution are described below,

```
y<-veteran$time
```

```
x1<-veteran$karno
```

```
x2<-veteran$celltype
```

```
x3<-veteran$diagtime
```

```
x4<-veteran$age
```

```
x5<-veteran$prior
```

```
x6<-veteran$trt
```

```
sensor<-veteran$status
```

```
N<-137
```

```
X<-cbind(1,x1,x2,x3,x4,x5,x6)
```

```
J<-7
```

```
mon.names<-c("LP","shape1","shape2")
```

```
parm.names<-  
as.parm.names(list(beta=rep(0,J),log.shape1=0,  
log.shape2=0))
```

```
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=  
parm.names,y=y,sensor=sensor)
```

7.2. Specification of model for exponential Lomax distribution

Let's consider a regression model, which can be written as:

$$y \sim EL(\nu, \alpha, \lambda)$$

where, ν and α are the two shape parameters and λ is the scale parameter.

Also,

$$\log(\lambda) = X\beta$$

Prior for ν ,

$$\nu \sim \text{half-cauchy}(25)$$

Prior for α ,

$$\alpha \sim \text{half-cauchy}(25)$$

Prior for β ,

$$\beta \sim N(0,1000)$$

All these parameters and their priors have been defined in the function called `Model`. For exponential Lomax distribution there is no distribution function available in this package, so we will define density and survival function of EL distribution with object name `f1` and `s1`, respectively. The R command for the model specification of exponential Lomax distribution is given below:

```
Model<-function(parm,Data)
{
#Parameters
beta<-parm[1:Data$J]
shape1<-exp(parm[Data$J+1])
shape2<-exp(parm[Data$J+2])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
shape1.prior<-dhalfcauchy(shape1,20,log=T)
shape2.prior<-dhalfcauchy(shape2,20,log=T)
```

```

# Loglikelihood

mu<-tcrossprod(beta,Data$X)

scale<-exp(mu)

f1<-log(shape1)+log(shape2)-log(scale)-
(shape2+1)*log(1+y/scale)+(shape1-1)*
log(1-(1+y/scale)^(-shape2))

s1<-log(1-(1-(1+y/scale)^(-shape2))^shape1)

LL<-censor*f1+(1-censor)*s1

LL<-sum(LL)

## Log-posterior

LP<-LL+beta.prior+shape1.prior+shape2.prior

Modelout<-list(LP=LP,Dev=-
2*LL,Monitor=c(LP,shape1,shape2),

yhat=rexplomax(length(y),shape1,shape2,scale),

parm=parm)

return(Modelout)
}

```

7.3. Fitting of the data using function LaplaceApproximation

Let us fit the model using LaplaceApproximation with the option of Nelder-Mead (1965) method of optimization as,

```

Initial.Values<-c(coef(lm(log(y)~x1+as.numeric(x2)+x3+x4+
as.numeric(x5)+x6)),log(1),log(1))

M1<-
LaplaceApproximation(Model,Initial.Values,Data=MyData,

Samples=5000,Method="NM",Iterations=10000)

```

7.4. Output summary

The output obtained by `LaplaceApproximation` is being reported in Table 3. This contains the posterior mode, posterior mean, posterior sd, 2:5% and 97:5% quantiles.

Table 3. Posterior summaries of lung cancer data by `LaplaceApproximation` function, giving two summaries first from Nelder-mead method and second is from sampling importance resampling under the assumption of exponential Lomax model.

	Optimization- Nelder and Mead Method			
	Mode	SD	LB	UB
Intercept	2.077	1.219	-0.362	4.516
Karno	0.040	0.005	0.030	0.049
Cell type	-0.010	0.091	-0.191	0.172
Diagnosis time	0.002	0.009	-0.017	0.021
Age	0.008	0.010	-0.011	0.028
Prior therapy	0.004	0.022	-0.040	0.047
Treatment	-0.095	0.190	-0.476	0.286
log.shape1	0.576	0.230	0.117	1.036
log.shape2	0.902	0.356	0.190	1.614

	Simulation-Sampling Importance Resampling			
	Mean	SD	LB	UB
Intercept	2.131	1.302	-0.420	4.501
Karno	0.040	0.005	0.033	0.046
Cell type	-0.011	0.100	-0.194	0.196
Diagnosis time	0.001	0.014	-0.023	0.025
Age	0.011	0.010	-0.014	0.035
Prior	0.001	0.023	-0.056	0.055
Treatment	-0.115	0.206	-0.509	0.268
shape1	2.026	0.537	1.285	3.488
shape2	2.615	1.042	1.371	5.565

7.5. Fitting of the data using LaplacesDemon function

Let us fit the model for the same data using LaplacesDemon with the option of independent Metropolis algorithm of simulation. The output obtained by simulation is reported in Table 4 and the posterior density plots are reported in Figure 6. There are six regressors but only plots of two regressors are reported (Karno= $\beta[2]$, Cell type= $\beta[3]$ and $\beta[1]$ is the intercept). In Figure 6 all the three rows are considered as $\beta[1]$, $\beta[2]$ and $\beta[3]$, respectively. These plots show the well mixing of the chain and low auto-correlation shows fast convergence of algorithm for all the variables.

```
Initial.Values<-as.initial.values(M1)
```

```
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
```

```
Covar=M1$Covar, Iterations=2000, Status=F, Thinning=1,
```

```
Algorithm="IM", Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
```

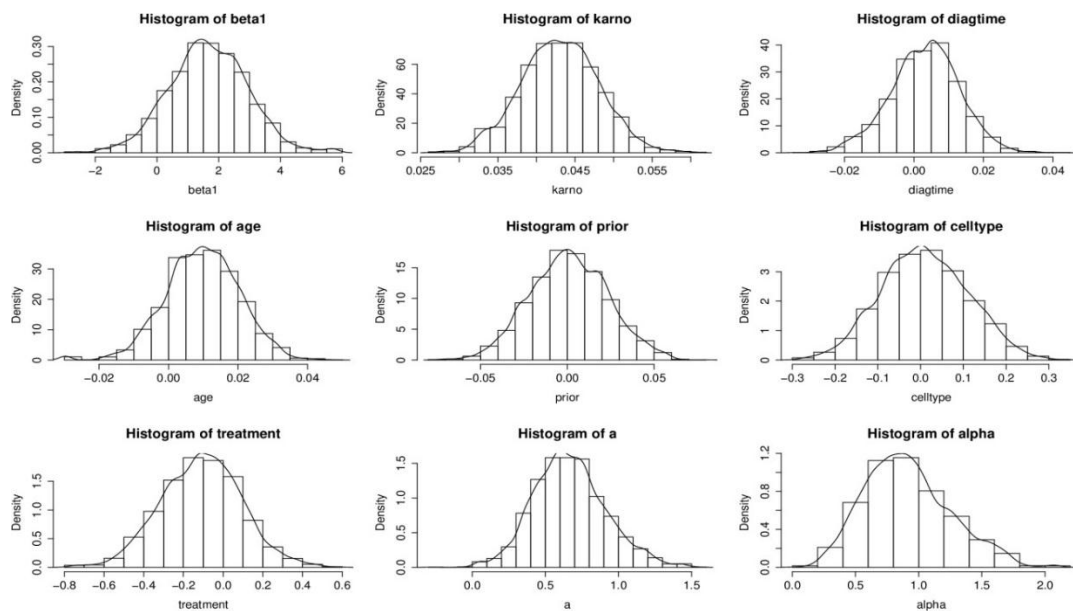
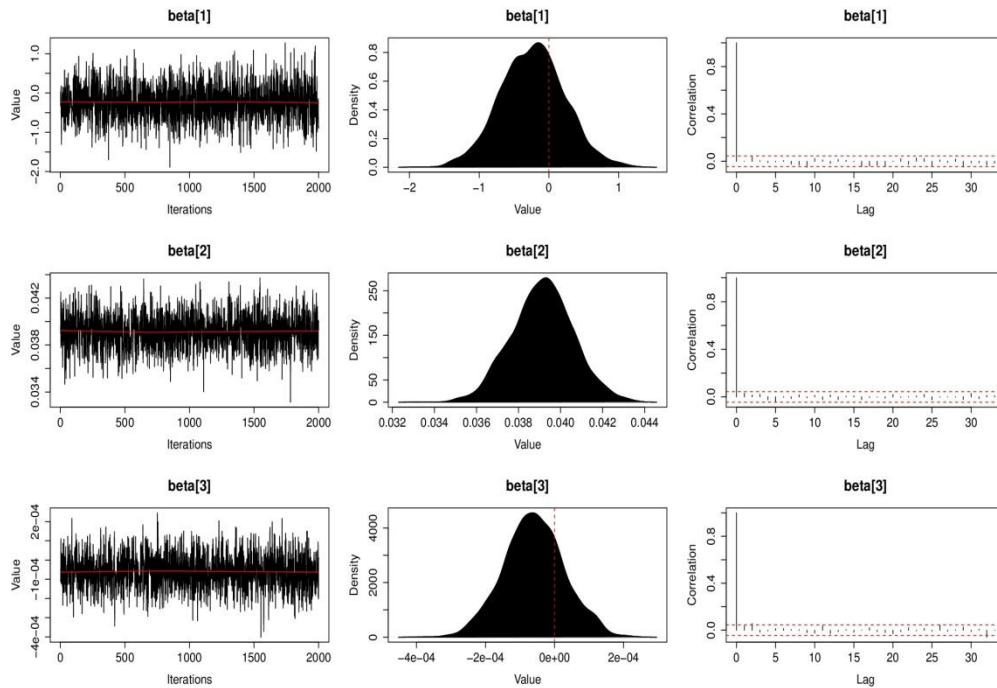


Figure 5. Histogram and posterior densities of all the parameters and regression coefficients under the assumption of exponential Lomax distribution.

Table 4. Simulated posterior summaries obtained by independent Metropolis algorithm.

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	1.285	0.462	0.012	2000.00	0.363	1.309	2.176
Karno	0.039	0.001	0.002	1641.23	0.041	0.044	0.053
Cell type	0.034	0.041	0.003	2000.00	-0.052	0.031	0.109
Diagnosis time	0.002	0.001	0.003	2000.00	-0.005	0.008	0.016
Age	0.01	0.001	0.002	2000.00	0.012	0.016	0.024
Prior therapy	0.004	0.010	0.004	2000.00	-0.018	0.004	0.028
Treatment	-0.092	0.080	0.001	1543.30	-0.241	-0.095	0.078
shape1	2.071	0.203	0.001	1788.54	1.714	2.055	2.492
shape2	2.383	0.300	0.011	2000.00	1.841	2.364	3.028

**Figure 6.** Simulated posterior density plots of the parameters of the distribution. The leftmost is the trace plot, the middlemost is the density plots and the rightmost is the auto correlation plots, showing low autocorrelation at different lags.

8. MODEL COMPARISON

Model selection is the task of choosing appropriate model from a set of candidate models. Here, Table 5 clearly shows that exponential Lomax is the appropriate model for veteran data as it has minimum value of DIC and deviance as compared to Lomax. DIC and deviance are very good criteria of model comparison as they have the potential to provide powerful comparison of complex models.

Table 5. Model comparison of Lomax and exponential Lomax models for the lung cancer data. It is evident from this table that exponential Lomax fits much better than Lomax.

Models	Deviance	DIC
Lomax	1446.665	1448.316
Exponential Lomax	1403.089	1405.332

9. CONCLUSION

In this paper, Bayesian approach has been employed to model the real survival data under the assumption of Lomax and its extended forms namely, exponential Lomax distribution. These distributions have been used as a Bayesian model to fit the survival data. This paper includes the derivation of joint and marginal posterior densities of these two models. Asymptotic approximation and simulation methods, the two most important techniques have been implemented to solve the high-dimensional integrations. These two methods have been implemented using the functions of `LaplacesDemon` package. The function `LaplaceApproximation` is the main function for the purpose of optimization in Bayesian scenario whereas `LaplacesDemon` is the function which is meant for implementation of Markov chain Monte Carlo simulation tools. The central part of the chapter has been composed of the description of R code. After Bayesian modeling of these distributions, the last step is to compare the goodness of fit of the models through the values of DIC and deviance, as per recommendation of Gelman et al., (2004) deviance is the best criteria for model selection. Following Table 5, it could be noticed that the value of DIC and deviance for exponential Lomax distribution is the least value followed by Lomax. Hence, it could be concluded that the EL is highly competitive in the sense of fitting real survival data.

The code developed in R can be used in other areas of regression modeling besides in the field of survival, because of their general nature and paradigm. Finally, Bayesian approach is more suitable even if sample size is small and it can be used very effectively in the modeling of survival data wherein non Guassian model like Weibull, Lomax and exponential Lomax commonly fit. Undoubtedly, R enhances the astonishing vigour of Bayesian approach.

REFERENCES

- [1] Abdul-Moniem, I.B. and Abdel-Hameed, H.F.(2012) On exponentiated Lomax distribution, *International Journal of Mathematical Archive* 3, 2144-2150.
- [2] Bennett, S. (1983). Log-logistic regression models for survival data. *Applied Statistics* 32, 165-171.
- [3] Campbell, G. and Ratnaparkhi, M. V. (1993). An application of Lomax distributions in receiver operating characteristic (ROC) curve analysis, *Communications in Statistics-Theory Methods* 22, 1681-1697.
- [4] Chahkandi, M. and Ganjali, M. (2009). On some lifetime distributions with decreasing failure rate. *Computational Statistics and Data Analysis*. 53, 4433-4440.
- [5] Chen, K., Jin, Z., and Ying, Z. (2002). Semiparametric analysis of transformation models with censored data. *Biometrika* 89, 659-668.
- [6] Corbellini, A., Crosato, L., Ganugi, P and Mazzoli, M. (2007). Fitting Pareto II distributions on rm size: Statistical methodology and economic puzzles. Paper presented at the International Conference on Applied Stochastic Models and Data Analysis, Chania, Crete.
- [7] Cordeiro, G. M., Ortega, E. M. M. and Popovi'c, B. V. (2013). The gamma-Lomax distribution. *Journal of Statistical computation and Simulation*. doi:10.1080/00949655.2013.822869.
- [8] Erkanli, A. (1994). Laplace approximations for posterior expectation when the model occurs at the boundary of the parameter space, *Journal of the American Statistical Association* 89, 205-258.
- [9] Gelman, A., Carlin, J.B., Stern, H.S., and Rubin, D.B. (2004). *Bayesian Data Analysis, Second Edition*. Chapman and Hall/CRC, Boca Raton, FL.
- [10] Ghitany, M. E., AL-Awadhi, F. A and Alkhalafan, L. A. (2007). Marshall-Olkin extended Lomax distribution and its applications to censored data. *Communications in Statistics-Theory and Methods* 36, 1855-1866.
- [11] Glanzel, W. (2008). On some new bibliometric applications of statistics related to the h-index. *Scientometrics* 77, 187-196.
- [12] Gupta, R. C., Gupta, P. L., and Gupta, R. D. (1998). Modeling failure time data by Lehmann alternatives, *Communications in Statistics-Theory and Methods* 27, 887-904.
- [13] Gupta, R. C., Ghitany, M. E. and Al-Mutairi, D. K. (2010). Estimation of reliability from Marshall-Olkin extended Lomax distributions. *Journal of Statistical Computation and Simulation* 80, 937-947.
- [14] Harris, C. M. (1968). The Pareto distribution as a queue service discipline. *Operations Research* 16, 307-313.

- [15] Hassan, A. S. and Al-Ghamdi, A. S. (2009). Optimum step stress accelerated life testing for Lomax distribution. *Journal of Applied Sciences Research* 5, 2153-2164.
- [16] Holland, O., Golaup, A. and Aghvami, A. H. (2006). Traffic characteristics of aggregated module downloads for mobile terminal reconfiguration. *IEE proceedings on Communications*. 135, 683-690.
- [17] Kalbeisch, J. D., and Prentice, R. L. (1980). *The Statistical Analysis of Failure Time Data*. New York: John Wiley and Sons.
- [18] Kalbeisch, J. D., and Prentice, R. L. (2002). *The Statistical Analysis of Failure Time Data*, second edition, New York: John Wiley and Sons.
- [19] Laplace, P. (1774). *Memoire sur la Probabilite des Causes par les Evenements*. *l'Academie Royale des Sciences* 6, 621-656. English translation by S.M. Stigler in 1986 as *Memoir on the Probability of the causes of events in Statistical Science* 1, 364-378.

