Statistics and Applications {ISSN 2454-7395 (online)}
Volume 22, No. 1, 2024 (New Series), pp 241–261
http://www.ssca.org.in/journal

Bayesian Inference for Glioma Data Using Generalized Burr X-G (GBX-G) Family with R and Stan

Devashish, Shazia Farhin, and Athar Ali Khan
Department of Statistics and Operations Research
Aligarh Muslim University, Aligarh - 202002, India

Received: 17 April 2023; Revised: 29 August 2023; Accepted: 03 September 2023

Abstract

Bayesian modeling of generalized distributions is currently highly appreciated due to the impressive growth in computational capabilities and software accessibility. This work attempts to fit the Bayesian inference methods for the generalized Burr X-G (GBX-G) family. On the basis of the GBX-G family, three distributions— the generalized Burr X-Weibull, the generalized Burr X-Exponential and the generalized Burr X-Lomax are analysed and fitted to censored survival data of malignant glioma patients using the probabilistic programming language STAN. In order to apply censored mechanisms throughout using STAN, codes are developed. Finally, a comparison has been made between the models through the use of Watanabe Akaike information criteria and leave one out cross-validation information criteria and conclusion has been given regarding the Bayesian model fitting of the glioma dataset.

Key words: Generalized Burr X-G (GBX-G) family; Bayesian survival modelling; Censored data; MCMC; LOOIC and WAIC methods.

1. Introduction

The time until an event occurs is the outcome variable of interest in a group of statistical techniques for data analysis called survival analysis. In the literature, there are many models that may be used to analyse lifetime data. Burr X (BX) distribution and it's generalization has been a part of research interest for survival data analysis for a long period of time. Burr (1942) introduced the Burr X (BX) distribution and later Yousof *et al.* (2017) defined Burr X-G (BX-G) family of distributions and also discussed it's application in analysing lifetime data. Also several other extended forms of the Burr X-G family were studied such as the transmuted Burr X-G (TBX-G) family and the truncated Burr X-G family of distributions that have been proposed and discussed by Al-Babtain *et al.* (2021) and Bantan *et al.* (2021) respectively. Apart from this Akhtar and Khan (2014) has conducted Bayesian analysis of generalized log-Burr family using R.

Based on the Burr X (BX) distribution, Aldahlan *et al.* (2021) created a new class of continuous distributions known as the generalised Burr X-G (GBX-G) family, studied its

Corresponding Author: Devashish
Email: devashishstatistics@gmail.com

mathematical properties, such as explicit expressions for the quantile and generating functions, ordinary and incomplete moments, order statistics, *etc.*, and provided its applications to real data sets. The GBX-G family's versatility in accommodating various forms of the hazard rate function (for details see Aldahlan *et al.* (2021)) turns into the driving force behind our work. The three GBX-G family-based models have been taken into consideration namely generalized Burr X-Exponential (GBXEx) Model, generalized Burr X-Weibull (GBXW) Model, and generalized Burr X-Lomax (GBXLx) Model in order to fit a real censored survival data named **glioma** which was initially discussed by Grana *et al.* (2002), under the Bayesian setup.

The comprehensive Bayesian inference-supporting probabilistic programming language **STAN** Carpenter *et al.* (2017) in R R Core Team (2021) is used to fit the aforementioned models. In Bayesian analysis, the computer language STAN is most typically used as a Hamiltonian Monte Carlo (HMC) sampler Duane *et al.* (1987); Brooks *et al.* (2011). Statistical models are defined using STAN. For Bayesian analysis, STAN predominantly uses the No-U-Turn sampler (NUTS) Hoffman *et al.* (2014) to obtain posterior simulation. We have also assessed and chosen the most appropriate model for the glioma data using the Watanabe-Akaike information criteria, or widely applicable information criteria (WAIC) and the Leave-One-Out information criteria (LOOIC). LOOIC and WAIC are two techniques for assessing the accuracy of pointwise out-of-sample predictions using a fitted Bayesian model and the log-likelihood evaluated at the posterior simulations of the parameter values, see Vehtari *et al.* (2017).
The article is structured as follows:
1. Explanation of PDF, CDF and hazard function for GBX-G family and all three models GBXEx, GBXW, and GBXLx of it (Section 2).
2. Explanation of the glioma data set and it's structure for STAN (Section 3).
3. Analysis under Bayesian approach by providing Likelihood, prior and posterior for all three models (Section 4).
4. Implementation and model fitting using STAN (Section 4.5).
5. Numeric as well as graphic results and interpretation of Bayesian analysis for the glioma data set (Section 4.8 - Section 4.12).
6. Bayesian Model comparison for the glioma data set (Section 5).
7. Conclusion (Section 6).

2. Generalized Burr X-G (GBX-G) family

A continuous random variable T is said to have the GBX-G family that is T $\sim$ GBX-G(α, β, η), if it has following probability density function (PDF), cumulative distribution function (CDF), survival function, and hazard function respectively, see (Aldahlan *et al.*, 2021)-

$$f_T(t,\alpha,\beta,\eta) = \frac{2\alpha\beta g(t,\eta)G(t,\eta)^{2\alpha-1}}{[1-G(t,\eta)^{\alpha}]^3} exp(-[\frac{G(t,\eta)^{\alpha}}{1-G(t,\eta)^{\alpha}}]^2) \times(1-exp(-[\frac{G(t,\eta)^{\alpha}}{1-G(t,\eta)^{\alpha}}]^2))^{\beta-1} \tag{1}$$

$$F_T(t,\alpha,\beta,\eta) = (1-exp(-[\frac{G(t,\eta)^{\alpha}}{1-G(t,\eta)^{\alpha}}]^2))^{\beta} \tag{2}$$

$$S_T(t,\alpha,\beta,\eta) = 1-(1-exp(-[\frac{G(t,\eta)^\alpha}{1-G(t,\eta)^\alpha}]^2))^\beta \tag{3}$$

$$h_T(t,\alpha,\beta,\eta) = \frac{f_T(t,\alpha,\beta,\eta)}{S_T(t,\alpha,\beta,\eta)} \tag{4}$$

Where α and β are positive shape parameters and η is parameter vector.

2.1. Generalized Burr X-exponential (GBXEx) model

Let the PDF $g(t) = \lambda e^{-\lambda t}$, for t > 0, of the exponential distribution with scale parameter λ, λ>0. Then, the probability density function (PDF), cumulative distribution function (CDF), survival function of the GBXEx model becomes

$$\begin{aligned} f(t) = \frac{2\alpha\beta\lambda e^{-\lambda t}(1-e^{-\lambda t})^{2\alpha-1}}{[1-(1-e^{-\lambda t})^\alpha]^3} exp(-[\frac{(1-e^{-\lambda t})^\alpha}{1-(1-e^{-\lambda t})^\alpha}]^2) \\ \times(1-exp(-[\frac{(1-e^{-\lambda t})^\alpha}{1-(1-e^{-\lambda t})^\alpha}]^2))^{\beta-1} \end{aligned} \tag{5}$$

$$F(t) = (1-exp(-[\frac{(1-e^{-\lambda t})^\alpha}{1-(1-e^{-\lambda t})^\alpha}]^2))^\beta \tag{6}$$

$$S(t) = 1-(1-exp(-[\frac{(1-e^{-\lambda t})^\alpha}{1-(1-e^{-\lambda t})^\alpha}]^2))^\beta \tag{7}$$

Also, using above expressions the hazard function can be obtained as-

$$h(t) = \frac{f(t)}{S(t)} \tag{8}$$

Here the random variable T will be denoted as T $\sim$ GBXEx(α, β, λ).

Random number generation - For random number generation of time variable from any survival model we will equate Survival function, S(t) to u, where U is a Uniform(0,1) variate and solve this equation for the value of t. Gelman *et al.* (2013) explained this method to generate the random numbers. Farhin *et al.* (2022) used this method recently.
The random number generation from GBXEx model is obtained by -

$$t = \frac{-1}{\lambda} log(1-(\frac{(-log(1-(1-u)^{1/\beta}))^{1/2}}{1+(-log(1-(1-u)^{1/\beta}))^{1/2}})^{1/\alpha}) \tag{9}$$

2.2. Generalized Burr X-Weibull (GBXW) model

Let the PDF $g(t) = a\lambda x^{\lambda-1}e^{-ax^\lambda}$, for t > 0, of the Weibull distribution with parameters λ and a, λ>0, a>0. Thus, the probability density function (PDF), cumulative distribution function (CDF), survival function of the GBXW model becomes

$$\begin{aligned} f(t) = \frac{2\alpha\beta a\lambda t^{\lambda-1}e^{-at^\lambda}(1-e^{-at^\lambda})^{2\alpha-1}}{[1-(1-e^{-at^\lambda})^\alpha]^3} exp(-[\frac{(1-e^{-at^\lambda})^\alpha}{1-(1-e^{-at^\lambda})^\alpha}]^2) \\ \times(1-exp(-[\frac{(1-e^{-at^\lambda})^\alpha}{1-(1-e^{-at^\lambda})^\alpha}]^2))^{\beta-1} \end{aligned} \tag{10}$$

$$F(t) = (1 - exp(-[\frac{(1 - e^{-at^{\lambda}})^{\alpha}}{1 - (1 - e^{-at^{\lambda}})^{\alpha}}]^2))^{\beta} \tag{11}$$

$$S(t) = 1 - (1 - exp(-[\frac{(1 - e^{-at^{\lambda}})^{\alpha}}{1 - (1 - e^{-at^{\lambda}})^{\alpha}}]^2))^{\beta} \tag{12}$$

Also, using above expressions the hazard function can be obtained as-

$$h(t) = \frac{f(t)}{S(t)} \tag{13}$$

Here the random variable T will be denoted as T $\sim$ GBXW(α, β, a, λ).

Also, the generation of random numbers from GBXW model is obtained by -

$$t = \lambda \times (-log(1 - (\frac{(-log(1 - (1 - u)^{1/\beta}))^{1/2}}{1 + (-log(1 - (1 - u)^{1/\beta}))^{1/2}})^{1/\alpha}))^{1/a} \tag{14}$$

2.3. Generalized Burr X-Lomax (GBXLx) model

Let the PDF $g(t) = a/\lambda[1 + t/\lambda]^{-a-1}$, for t $>$ 0, of the Lomax distribution with parameters λ and a, λ>0, a>0. Thus, the probability density function (PDF), cumulative distribution function (CDF), survival function of the GBXLx model is given by

$$f(t) = \frac{2\alpha\beta a/\lambda[1 + t/\lambda]^{-a-1}(1 - [1 + t/\lambda]^{-a})^{2\alpha-1}}{[1 - (1 - [1 + t/\lambda]^{-a})^{\alpha}]^3} exp(-[\frac{(1 - [1 + t/\lambda]^{-a})^{\alpha}}{1 - (1 - [1 + t/\lambda]^{-a})^{\alpha}}]^2) \times (1 - exp(-[\frac{(1 - [1 + t/\lambda]^{-a})^{\alpha}}{1 - (1 - [1 + t/\lambda]^{-a})^{\alpha}}]^2))^{\beta-1} \tag{15}$$

$$F(t) = (1 - exp(-[\frac{(1 - [1 + t/\lambda]^{-a})^{\alpha}}{1 - (1 - [1 + t/\lambda]^{-a})^{\alpha}}]^2))^{\beta} \tag{16}$$

$$S(t) = 1 - (1 - exp(-[\frac{(1 - [1 + t/\lambda]^{-a})^{\alpha}}{1 - (1 - [1 + t/\lambda]^{-a})^{\alpha}}]^2))^{\beta} \tag{17}$$

Also, using above expressions the hazard function can be obtained as-

$$h(t) = \frac{f(t)}{S(t)} \tag{18}$$

Here the random variable T will be denoted as T $\sim$ GBXLx(α, β, a, λ). The generation of random numbers from GBXLx model is obtained by-

$$t = \lambda \times ((1 - (\frac{((1 - (1 - u)^{1/\beta}))^{1/2}}{1 + (-log(1 - (1 - u)^{1/\beta}))^{1/2}})^{1/\alpha})^{-1/a} - 1) \tag{19}$$

3. Data: malignant glioma pilot study

On malignant glioma patients receiving pretargeted adjuvant radioimmunotherapy with yttrium-90-biotin, Grana *et al.* (2002) did a non-randomized pilot research and evaluated overall survival and the time to relapse. In this study, 37 high-grade glioma patients, 17 with grade III glioma and 20 with glioblastoma (GBM) were enrolled in a controlled open non-randomized study. Among them, 19 patients were treated with adjuvant radioimmunotherapy (RIT) and 18 were represented as the Control group. The survival time for each patient alongwith other helpful information such as gender, histology, age, *etc.* had been recorded. There are 14 censored observations out of 37 in the dataset.
This complete data set can be accessed through the R R Core Team (2021) package **coin** Zeileis *et al.* (2008) with the name **glioma**.
The discription of variables of glioma data set are given below:
no.: patient number.
age: patient age in years.
sex: a factor indicating patient's gender with levels "M" for Male and "F" for Female.
histology: a factor with levels "Grade3" (grade III glioma) and "GBM" (grade IV or glioblastoma).
group: a factor with levels "RIT"(radioimmunotherapy) and "Control".
event: censoring status indicator: FALSE for right-censored values and TRUE otherwise.
time: survival time in months.

3.1. Data creation for computation in stan

The model matrix x, a number of predictors M, and details of the censoring and response variable are needed for data production. N is the number of observations that are stated. Censoring is considered, with 0 being censored values and 1 denoting uncensored values. Finally, a listed form of data named 'datg' is created by combining all of these operations.

```
library(coin)
library(survival)
data("glioma")
glioma
help(glioma)
head(glioma)
y=glioma$time
x1=glioma$age
x2=as.numeric(glioma$sex)
#0=Female, 1=Male
x2=as.numeric(x2==2)
x3=as.numeric(glioma$histology)
#0=GBM, 1=Grade3
x3=as.numeric(x3==2)
x4=as.numeric(glioma$group)
#0=Control, 1=RIT
x4=as.numeric(x4==2)
```

```
#0=censored, 1=observed
censor=as.numeric(glioma$event)
x=cbind(1,x1,x2,x3,x4)
N=nrow(x)
M=ncol(x)
datg=list(y=y,censor=censor,x=x,N=N,M=M)
```

4. Analysis using Bayesian mechanism

In Bayesian analysis, in accordance with Bayes Theorem, we look for the posterior distribution which is the exact parameter distribution, by combining likelihood or data with the prior distribution of the parameter. The likelihood of the data and the prior distribution or the prior belief about the model's parameters must be established before the Bayesian regression model can be built.

4.1. Likelihood

Following Collett (2015), The right censored data can be formulated using the following joint likelihood function-

$$L = \prod_{i=1}^{n} h(t_i)^{\gamma_i} S(t_i) \tag{20}$$

And, the log-likelihood can be re-written as an alternative to the above form as-

$$logL = \sum_{i=1}^{n} (\gamma_i (logh(t_i) + logS(t_i))) \tag{21}$$

Here γ_i is the censoring indicator such that $\gamma = 1$ if the observation is not censored and $\gamma = 0$ if the observation is censored. To obtain the likelihood of GBXEx, GBXW and GBXLx survival models, the survival function $S(t_i)$ and the hazard function $h(t_i)$ of GBXEx, GBXW and GBXLx models respectively can be sustituted in the equation 20.

4.2. Modeling information

Following Lawless (2011), We have introduced covariates using the log link function in order to construct a regression model.

$$log(\lambda_i) = b_1 + b_2 x_{i1} + b_3 x_{i2} + b_4 x_{i3} + b_5 x_{i4} \tag{22}$$

$$\lambda_i = exp(b_1 + b_2 x_{i1} + b_3 x_{i2} + b_4 x_{i3} + b_5 x_{i4}) \tag{23}$$

$$\lambda_i = exp(x_i b) \tag{24}$$

Here $b = [b_1, b_2, b_3, b_4, b_5]$ are regression coefficients and x_i's are covariates of the data set discussed in Section 3. In particular, b_1 is the intercept, b_2 is the coefficient of covariate (x_1) of Age, b_3 is the coefficient of covariate (x_2) of Sex, b_4 is the coefficient of covariate (x_3) of Histology and b_5 is the coefficient of covariate (x_4) of Group.
In **STAN** the *transformed parameter* block of the stan model code contains above regression model. These stan codes are discussed in Section 4.5.

4.3. Prior

A prior probability distribution for parameters of the model needs to be specified before building the Bayesian regression model. In this article, for shape and scale parameters, we have chosen a half-Cauchy prior, and for the regression coefficient, a regularizing prior. We have opted for the Normal prior with mean 0, and standard deviation 5 for regression coefficient as a regularizing prior. Regularizing prior reduces the rate of learning from the data and prevents a model from becoming overexcited by it. Notably, it reduces the overfitting of the model to the data.
The half-Cauchy distribution with scale parameter 25, used as a noninformative prior distribution for shape parameter. Taking 25 as the value of the scale parameter, the half-Cauchy distribution becomes almost flat. Gelman (2006) support the use of half-Cauchy or uniform prior for the regression coefficents. Khan and Khan (2018) and Farhin and Khan (2023) explained the use of Gaussian prior for the regression coefficient and half-Cauchy prior for the shape as well as the scale in detail.

4.4. Posterior

Here the Bayes Theorem is used to obtain the joint posterior distribution of parameter $\theta = (\alpha, \beta, a, b) = (\alpha, \beta, a, b_0, b_1, ..., b_p)$ given the data as

$$P(\theta|t, X) \propto L(t|\theta, X)P(\theta) \tag{25}$$

Taking X as the matrix of covariates and assuming the parameters as independent, we have

$$P(\alpha, \beta, a, b|t, X) \propto P(t|\alpha, \beta, a, b, X)P(\alpha)P(\beta)P(a)P(b) \tag{26}$$

Hence the joint posterior distribution of GBXEx Model, GBXW Model and GBXLx Model can be obtained by sustituting priors and the likelihood of the corresponding models in Equation 26.

4.4.1. Posterior density of GBXEx model

$$\begin{aligned}
&P(\alpha, \beta, b|t, X) \propto P(t|\alpha, \beta, b, X)P(\alpha)P(\beta)P(b)\\
&\propto \prod_{i=1}^{n} \left\{ \frac{2\alpha\beta e^{x_i b} e^{-e^{x_i b}t}(1 - e^{-e^{x_i b}t_i})^{2\alpha-1}}{[1 - (1 - e^{-e^{x_i b}t_i})^{\alpha}]^3} exp(-[\frac{(1 - e^{-e^{x_i b}t_i})^{\alpha}}{1 - (1 - e^{-e^{x_i b}t_i})^{\alpha}}]^2) \right\}^{\gamma_i}\\
&\times \left\{ (1 - exp(-[\frac{(1 - e^{-e^{x_i b}t_i})^{\alpha}}{1 - (1 - e^{-e^{x_i b}t_i})^{\alpha}}]^2))^{\beta-1} \right\}^{\gamma_i}\\
&\times \left\{ 1 - (1 - exp(-[\frac{(1 - e^{-e^{x_i b}t_i})^{\alpha}}{1 - (1 - e^{-e^{x_i b}t_i})^{\alpha}}]^2))^{\beta} \right\}^{1-\gamma_i}\\
&\times \frac{2 \times 25}{\pi(\alpha^2 + 25^2)} \times \frac{2 \times 25}{\pi(\beta^2 + 25^2)} \times \prod_{j=0}^{J} \frac{1}{5\sqrt{2\pi}} exp\left(-\frac{1}{2 \times 25} b_j^2\right)
\end{aligned} \tag{27}$$

4.4.2. Posterior density of GBXW model

$$
\begin{aligned}
&P(\alpha,\beta,a,b|t,X) \propto P(t|\alpha,\beta,a,b,X)P(\alpha)P(\beta)P(a)P(b)\\
&\propto \prod_{i=1}^{n}\left\{\frac{2\alpha\beta a e^{x_i b}t_i{}^{e^{x_i b}-1}e^{-at_i{}^{e^{x_i b}}}(1-e^{-at_i{}^{e^{x_i b}}})^{2\alpha-1}}{[1-(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}]^3}exp(-[\frac{(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}{1-(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}]^2)\right\}^{\gamma_i}\\
&\times\left\{(1-exp(-[\frac{(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}{1-(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}]^2))^{\beta-1}\right\}^{\gamma_i}\\
&\times\left\{1-(1-exp(-[\frac{(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}{1-(1-e^{-at_i{}^{e^{x_i b}}})^{\alpha}}]^2))^{\beta}\right\}^{1-\gamma_i}\\
&\times\frac{2\times 25}{\pi(\alpha^2+25^2)}\times\frac{2\times 25}{\pi(\beta^2+25^2)}\times\frac{2\times 25}{\pi(a^2+25^2)}\times\prod_{j=0}^{J}\frac{1}{5\sqrt{2\pi}}exp\left(-\frac{1}{2\times 25}b_j^2\right) \qquad (28)
\end{aligned}
$$

4.4.3. Posterior density of GBXLx model

$$
\begin{aligned}
&P(\alpha,\beta,a,b|t,X) \propto P(t|\alpha,\beta,a,b,X)P(\alpha)P(\beta)P(a)P(b)\\
&\propto \prod_{i=1}^{n}\left\{\frac{2\alpha\beta a/e^{x_i b}[1+t_i/e^{x_i b}]^{-a-1}(1-[1+t_i/e^{x_i b}]^{-a})^{2\alpha-1}}{[1-(1-[1+t_i/e^{x_i b}]^{-a})\alpha]^3}\right\}^{\gamma_i}\\
&\times\left\{exp(-[\frac{(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}{1-(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}]^2)\times(1-exp(-[\frac{(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}{1-(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}]^2))^{\beta-1}\right\}^{\gamma_i}\\
&\times\left\{1-(1-exp(-[\frac{(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}{1-(1-[1+t_i/e^{x_i b}]^{-a})^{\alpha}}]^2))^{\beta}\right\}^{1-\gamma_i}\\
&\times\frac{2\times 25}{\pi(\alpha^2+25^2)}\times\frac{2\times 25}{\pi(\beta^2+25^2)}\times\frac{2\times 25}{\pi(a^2+25^2)}\times\prod_{j=0}^{J}\frac{1}{5\sqrt{2\pi}}exp\left(-\frac{1}{2\times 25}b_j^2\right) \qquad (29)
\end{aligned}
$$

Now, to find marginal posterior distribution we need to solve a a high-dimensional integral over all model parameters. Since it is difficult to derive the normalised joint posterior distribution and the marginal distributions of the parameters analytically, we approximate these integrals using Markov chain Monte Carlo (MCMC) methods. Thus, the estimates and other pertinent findings are achieved using the MCMC simulation approach with the aid of STAN.

4.5. Implementation using stan

STAN incorporates the use of the no-U-turn sampler (NUTS), an adaptive variant of Hamiltonian Monte Carlo (HMC) sampling, to efficiently simulate from the posterior distribution. HMC, which extends the capabilities of the Metropolis algorithm, is particularly advantageous in high-dimensional models due to its improved effectiveness and speed. While

Bayesian inference using Gibbs sampling (BUGS) is a commonly used approach, it often faces challenges when confronted with large datasets or complex models, leading to lengthy computation times or even failure to provide solutions. STAN, on the other hand, excels in handling such scenarios, offering faster computations and requiring a reduced number of iterations to achieve convergence when compared to BUGS (See Ashraf-Ul-Alam and Khan (2021) and Gelman *et al.* (2013)).
In R, to execute the STAN code, the package `rstan` is necessary. A Stan programme has six code blocks that are used for Bayesian modelling. Each block accommodates a list of instructions for distinct tasks. These blocks are - Data block, Transformed data block, Parameter block, Transformed parameter block, Model block, and Generated quantities block. In the Appendix, the stan codes with all these blocks for GBXEx, GBXW, and GBXLx models are provided.

4.6. Fitting the model using stan

The package `rstan` has a function named *stan* which is used to fit all the models based on GBX-G family. STAN uses C^{++} compiler for sampling from the posterior distrbution of the model parameters. All necessory codes for the numeric and graphical illustrations are provided in upcoming sub sections.

4.6.1. Bayesian data visualization: key plots for analysis

Graphical summary is an improtant part for analysis to assess the model convergence and communicate posterior related findings effectively. In this study, four plots are used namely- Traceplot, Caterpiller plot, Posterior predictive density plot and Posterior density plot. The traceplot provides a visual assessment of Markov Chain Monte Carlo (MCMC) convergence and it can be seen by comparing several Markov chains in a single plot. The Caterpiller plot shows the credible intervals or the quantiles for various parameters of the model and can be used to see the statistical significance of various coeffcients of the model. Posterior predictive checks (PPD plots) allow us to evaluate how well the model fits the observed data by comparing the density of the predicted values produced using the posterior predictive distribution of the specified model to the observed data. The posterior density plot is a graphical representation of the posterior distribution of a parameter and is constructed using simulated draws of the parameter from the posterior distribution. The R packages `Bayesplot` Gabry *et al.* (2019) and `ggplot2` Wickham and Wickham (2016) with `rstan` are used to create these plots in this paper.

4.7. Running the GBXEx model using stan

```
#calling rstan package
require(rstan)
#fitting the model
GBXE=stan(model_code = MGXE,data=datg,iter=4000,chains = 4)
print(GBXE)
```

4.8. Output summary and interpretaion

The results of Bayesian model fitting of GBXEx model are provided in Table 1. Also graphs are provided for summaries of posterior density and model convergence. The coeffcients b[2] of age (x1) and b[3] of sex (x2) are negative which shows that the older patients tends to have less survival probability then younger ones and the chances of survival for female are greater than male. The coeffcient b[5] of group is positive which indicates the chances of survival for patients who recieved the radioimmunotherapy (RIT) are greater than patients of control group which is a clear indication of positive impact of radioimmunotherapy on glioma patients. We can also see that the coeffcient b[4] of histology is positive which indicates patients with Grade 3 glioma have higher survival rates than those with glioblastoma (GBM). Also, after observing the estimates summary, it can be observed that the 95% credible intervals of b[4] and b[5] do not contain a value of zero, so the effect of coefficients of histology (GBM and Grade3) and group (Control and RIT) is statistically significant. Additionally, the summary table conatins the posterior estimates mean and se_mean, the standard deviation (sd), and the credible interval. Apart from this, the numerical summary table also contain the n_eff that is the effective number of samples which is a measure of the number of independent samples from the posterior distribution and the Rhat or the potential scale reduction factor, see (Gelman *et al.*, 2013), which is a quantitative criterion to assess convergence to the target distribution. In general n_eff > 100 and Rhat < 1.1 is acceptable for appropriate parameter estimates and model convergence, see (Gelman *et al.*, 2013). We can discern that the Rhat values for all parameters of the GBXEx model fall within an acceptable range, signifying successful convergence of the Markov chains to the desired distribution. The effective sample size is also reasonable. Using the **Bayesplot** package Gabry *et al.* (2019) , posterior predictive density (PPD) charts are created to visually assess the model. Posterior predictive density charts (Figure 2) depicts that the GBXEx model is consistent with the current data. Trace plots are also provided (Figure 1) to indicate the convergence of MCMC algorithm. Also, Figure 1 displays the caterpillar plot, wherein a vertical line appears at the value zero. Notably, the credible intervals of coefficients b[4] and b[5] lie on the right-hand side of the line, indicating that these intervals do not encompass the value of zero. This finding serves as evidence of the statistical significance of these coefficients.

Table 1: Posterior estimates results of GBXEx model parameters

Parameters	mean	se_mean	sd	2.5%	25%	50%	97.5%	n_eff	Rhat
b[1]	3.691	0.057	1.203	2.246	2.882	3.354	6.925	439	1.005
b[2]	-0.013	0.000	0.009	-0.031	-0.019	-0.013	0.006	899	1.002
b[3]	-0.284	0.008	0.245	-0.759	-0.444	-0.291	0.227	875	1.000
b[4]	0.962	0.010	0.307	0.431	0.758	0.940	1.627	967	1.003
b[5]	1.209	0.007	0.247	0.789	1.039	1.183	1.756	1418	1.001
alpha	4.652	0.155	4.223	0.122	0.972	3.576	14.582	747	1.004
beta	1.255	0.133	2.628	0.105	0.183	0.303	10.175	391	1.009

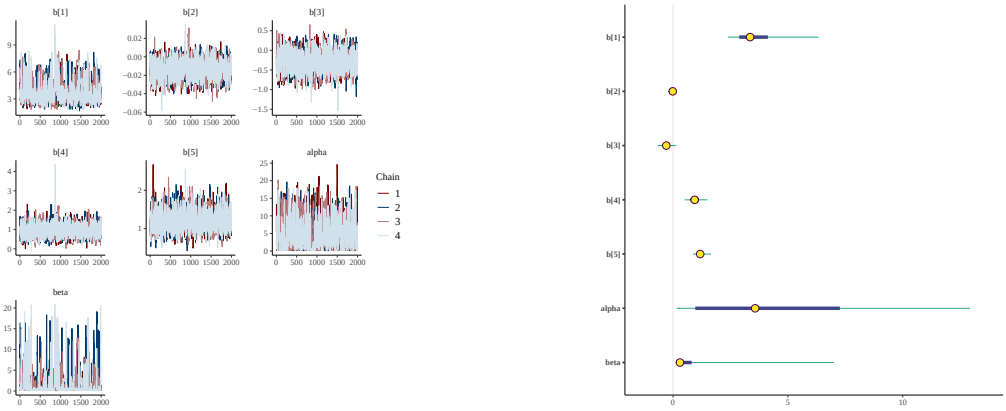

Figure 1: (i) Traceplot for GBXEx model, four chains were displayed in two separate runs; combining the two chains successfully indicates that MCMC algorithm has converged to the target joint posterior distribution. (ii) Caterpillar plot for GBXEx model.

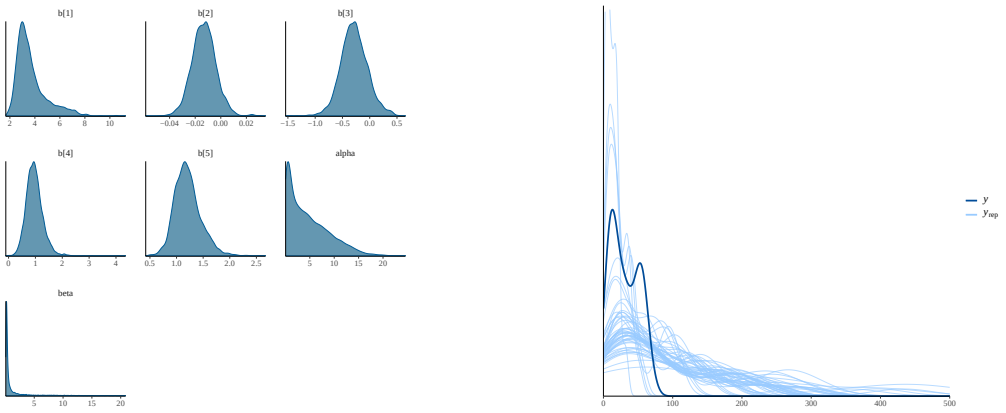

Figure 2: (i) Posterior density plot for GBXEx model. (ii) Posterior predictive density plot of the GBXEx model to assess the convergence of model. The GBXEx's posterior predictive density adequately fits the data, according to the PPD plot.

4.9. Running the GBXW model using stan

```
GBXW=stan(model_code = MGXW,data=datg,iter=4000,chains = 4, init = "random")
print(GBXW)
```

4.10. Output summary and interpretaion

The results of Bayesian model fitting of GBXW model are provided in Table 2. The Rhat values for model parameters are < 1.1, which depicts that the Markov chain converges to the desired distribution. And, the n_eff is more than 100 for all parameters of the model. The PPD chart (Figure 4) of the GBXW model indicates a good fit of the posterior predictive density with the data. It can also be seen that the coeffcients b[2] of age (x1) and b[3] of sex (x2) are negative and the coeffcients b[4] of histology (x3) and b[5] of group (x4) are positive. From the numeric summary of posterior estimates (Table 2) and the caterpillar plot (Figure 3), it is observed that the 95% credible intervals do not encompass the value

of zero for the coefficients of histology and group, which serves as evidence of the statistical significance of these coefficients.

Table 2: Posterior estimates results of GBXW model parameters

Parameters	mean	se_mean	sd	2.5%	25%	50%	97.5%	n_eff	Rhat
b[1]	0.500	0.112	3.827	-7.926	-1.967	0.971	6.993	1177	1.007
b[2]	-0.013	0.000	0.010	-0.032	-0.019	-0.013	0.007	2105	1.000
b[3]	-0.162	0.007	0.277	-0.696	-0.348	-0.166	0.372	1781	1.001
b[4]	0.966	0.006	0.302	0.417	0.763	0.948	1.612	2861	1.000
b[5]	1.208	0.005	0.254	0.756	1.036	1.190	1.760	2775	1.001
alpha	12.997	0.948	41.793	0.057	1.319	4.608	69.750	1945	1.003
beta	7.361	0.567	20.603	0.134	0.470	1.461	53.536	1319	1.003
a	0.859	0.072	1.671	0.086	0.168	0.292	6.352	539	1.010

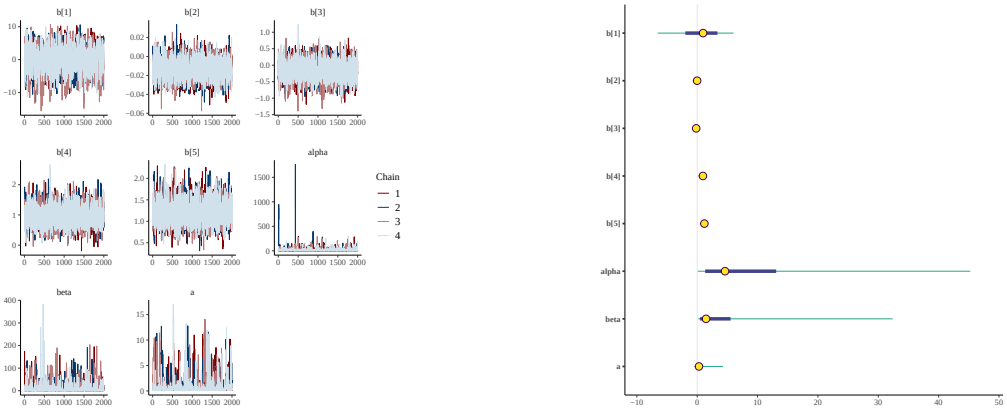

Figure 3: (i) Traceplot for GBXW model, four chains were displayed in two separate runs; combining the two chains successfully indicates that MCMC algorithm has converged to the target joint posterior distribution. (ii) Caterpillar plot for GBXW model.

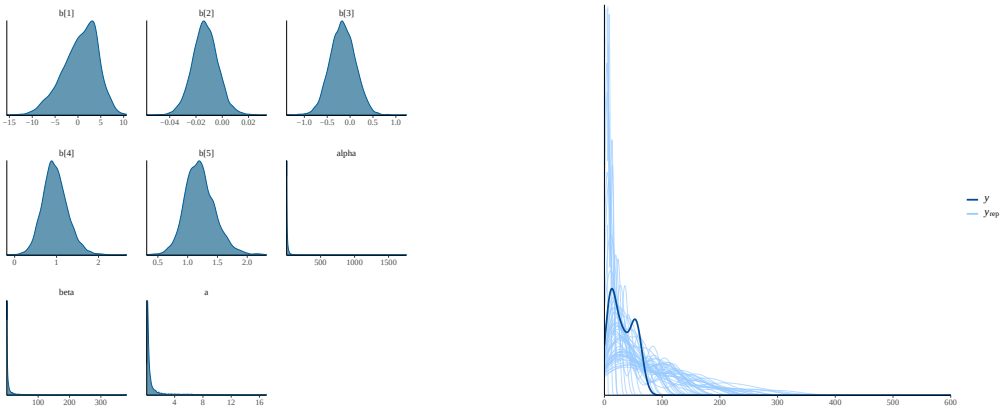

Figure 4: (i) Posterior density plot for GBXW model. (ii) Posterior predictive density plot of the GBXW model to assess the convergence of model. The GBXW's posterior predictive density adequately fits the data, according to the PPD plot.

4.11. Running the GBXLx model using stan

```
GBXL=stan(model_code = MGXL,data=datg,iter=4000,chains = 4)
print(GBXL)
```

4.12. Output summary and interpretaion

The results of Bayesian model fitting of GBXLx model are provided in Table 3. The Rhat values for model parameters are < 1.1, which depicts that the Markov chain converges to the desired distribution. And, the n_eff is more than 100 for all parameters of the model. The PPD chart (Figure 6) for the GBXLx model indicates a good fit of the posterior predictive density with the data. It can also be seen that the coeffcients b[2] of age (x1) and b[3] of sex (x2) are negative and the coeffcients b[4] of histology (x3) and b[5] of group (x4) are positive. From the numeric summary of posterior estimates (Table 3) and the caterpillar plot (Figure 5), it is observed that the 95% credible intervals do not encompass the value of zero for the coefficients of histology and group, which serves as evidence of the statistical significance of these coefficients.

Table 3: Posterior estimates results of GBXLx model parameters

Parameters	mean	se_mean	sd	2.5%	25%	50%	97.5%	n_eff	Rhat
b[1]	3.916	0.251	4.042	-4.775	1.203	5.100	9.957	259	1.016
b[2]	-0.013	0.000	0.009	-0.031	-0.019	-0.014	0.004	1453	1.002
b[3]	-0.203	0.009	0.254	-0.692	-0.372	-0.204	0.304	815	1.004
b[4]	0.966	0.007	0.286	0.459	0.770	0.951	1.594	1541	1.005
b[5]	1.194	0.006	0.241	0.773	1.034	1.173	1.723	1655	1.002
alpha	12.687	1.891	45.047	0.129	1.830	5.370	63.206	568	1.007
beta	2.067	0.228	5.376	0.118	0.221	0.440	15.137	556	1.004
a	26.023	2.067	95.486	0.274	1.062	6.787	152.409	2134	1.001

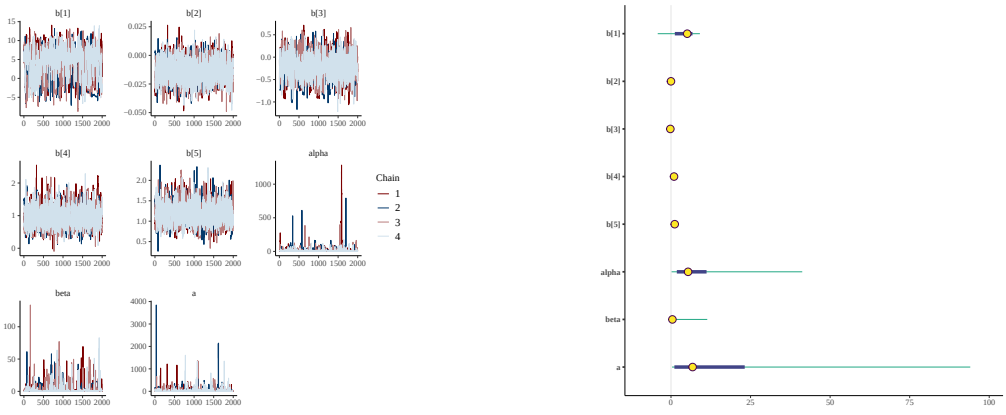

Figure 5: (i) Traceplot for GBXLx model, four chains were displayed in two separate runs; combining the two chains successfully indicates that MCMC algorithm has converged to the target joint posterior distribution. (ii) Caterpillar plot for GBXLx model.

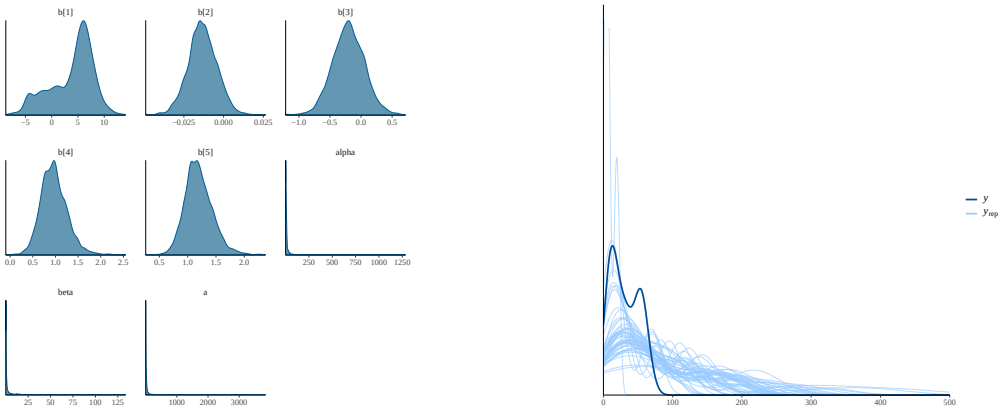

Figure 6: (i) Posterior density plot for GBXLx model. (ii) Posterior predictive density plot of the GBXLx model to assess the convergence of model. The GBXLx's posterior predictive density adequately fits the data, according to the PPD plot.

5. Model comparison with Bayesian criteria

For the purpose of comparing the fitted models, we use criteria for model evaluation and selection like the Leave One Out cross-validation Information Criteria (LOOIC) and the Watanabe Akaike Information Criteria (WAIC) both of which are methods for estimating pointwise out-of-sample prediction accuracy from a fitted Bayesian model Watanabe and Opper (2010); Vehtari *et al.* (2018). There are simpler estimates of predictive accuracy such as Akaike Information Criterion (AIC) and Deviance Information Criterion (DIC) but LOOIC and WAIC are better as instead of using only point estimates both LOOIC and WAIC use the pointwise log-likelihood of the full Bayesian posterior distribution. LOOIC and WAIC are more advantageous as they account for model complexity more effectively and offer fully Bayesian model comparison (See Vehtari *et al.* (2017)). LOOIC and WAIC quantifies the predictive accuracy of a model by estimating the expected log pointwise predictive density and in Stan, the generated quantities block computes these values. After fitting the model through STAN, the LOOIC and WAIC values are obtained by utilizing an R package `loo` (see Vehtari *et al.* (2018)) by calculating the log-likelihood assessed using posterior simulations of the parameters. A better model fit is indicated by a lower value for these selection criterias. Recently Ashraf-Ul-Alam and Khan (2021) and AbuJarad *et al.* (2022) used LOOIC and WAIC as the basis of comparision of the Bayesian survival models.

Table 4: LOOIC and WAIC values for GBXEx, GBXLx, and GBXW models

Model	LOOIC	WAIC
GBXEx	190.9	190.2
GBXLx	191.8	191.2
GBXW	192.8	192.4

We can observe from Table 4 that the GBXEx model's LOOIC and WAIC values are the lowest of the three, demonstrating that it exhibits superior performance as a survival model when compared to the other models applied to the glioma data.

6. Conclusion

In present study, Bayesian paradigm was applied to the analysis of a censored survival data using the GBXG family. The `Rstan` package of R is used to implement the simulation and analytical approximation techniques. The covariates Histology and Group are significant, and Markov chains for all models converge to the target distribution. The GBXEx model stands out as the most suitable option for fitting the glioma data, as evidenced by thorough comparisons of posterior predictive density plots, LOOIC, and WAIC. Additionally, patients who received radioimmunotherapy (RIT) had higher survival rates than individuals in the control group. Compared to patients with glioblastoma (GBM), patients with Grade 3 glioma had higher survival chances.

Acknowledgements

We sincerely appreciate the Editors' advice and support. We are very appreciative of the reviewer's insightful remarks and recommendations to generously add numerous helpful references.

References

AbuJarad, M. H., AbuJarad, E. S., and Khan, A. A. (2022). Bayesian survival analysis of type I general exponential distributions. *Annals of Data Science*, **9**, 347–367.

Akhtar, M. T. and Khan, A. A. (2014). Bayesian analysis of generalized log-Burr family with R. *SpringerPlus*, **3**, 1–10.

Al-Babtain, A. A., Elbatal, I., Al-Mofleh, H., Gemeay, A. M., Afify, A. Z., and Sarg, A. M. (2021). The flexible Burr XG family: properties, inference, and applications in engineering science. *Symmetry*, **13**, 474.

Aldahlan, M. A., Khalil, M. G., and Afify, A. Z. (2021). A new generalized family of distributions for lifetime data. *Journal of Modern Applied Statistical Methods*, **19**, 6.

Ashraf-Ul-Alam, M. and Khan, A. A. (2021). Generalized Topp-Leone-Weibull AFT modelling: a Bayesian analysis with MCMC tools using R and stan. *Austrian Journal of Statistics*, **50**, 52–76.

Bantan, R. A., Chesneau, C., Jamal, F., Elbatal, I., and Elgarhy, M. (2021). The truncated burr XG family of distributions: properties and applications to actuarial and financial data. *Entropy*, **23**, 1088.

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2011). *Handbook of Markov Chain Monte Carlo*. CRC press.

Burr, I. W. (1942). Cumulative frequency functions. *The Annals of Mathematical Statistics*, **13**, 215–232.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P., and Riddell, A. (2017). Stan: a probabilistic programming language. *Journal of Statistical Software*, **76**.

Collett, D. (2015). *Modelling Survival Data in Medical Research*. CRC press, third edition.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics Letters B*, **195**, 216–222.

Farhin, S. and Khan, A. A. (2023). Bayesian survival analysis of Rayleigh-X family with time varying covariate. *Applied Mathematics E-Notes*, **23**, 124–145.

Farhin, S., Yousuf, F., and Khan, A. A. (2022). Bayesian survival modeling of Marshal Olkin generalized-G family with random effects using R and stan. *Reliability: Theory & Applications*, **17**, 422–440.

Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., and Gelman, A. (2019). Visualization in Bayesian workflow. *Journal of the Royal Statistical Society Series A: Statistics in Society*, **182**, 389–402.

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper). *Bayesian Analysis*, **1**, 515–534.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. CRC press.

Grana, C., Chinol, M., Robertson, C., Mazzetta, C., Bartolomei, M., De Cicco, C., Fiorenza, M., Gatti, M., Caliceti, P., and Paganelli, G. (2002). Pretargeted adjuvant radioimmunotherapy with yttrium-90-biotin in malignant glioma patients: a pilot study. *British Journal of Cancer*, **86**, 207–212.

Hoffman, M. D., Gelman, A., et al. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, **15**, 1593–1623.

Khan, N. and Khan, A. A. (2018). Bayesian analysis of Topp-Leone generalized exponential distribution. *Austrian Journal of Statistics*, **47**, 1–15.

Lawless, J. F. (2011). *Statistical Models and Methods for Lifetime Data*. John Wiley & Sons.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Vehtari, A., Gabry, J., Yao, Y., and Gelman, A. (2018). loo: Efficient leave-one-out cross-validation and WAIC for Bayesian models. *R Package Version*, **2**, 1003.

Vehtari, A., Gelman, A., and Gabry, J. (2017). Practical bayesian model evaluation using leave-one-out cross-validation and waic. *Statistics and Computing*, **27**, 1413–1432.

Watanabe, S. and Opper, M. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, **11**.

Wickham, H. and Wickham, H. (2016). *Data Analysis*. Springer.

Yousof, H. M., Afify, A. Z., Hamedani, G., and Aryal, G. R. (2017). The burr X generator of distributions for lifetime data. *Journal of Statistical Theory and Applications*, **16**, 288–305.

Zeileis, A., Wiel, M. A., Hornik, K., and Hothorn, T. (2008). Implementing a class of permutation tests: the coin package. *Journal of Statistical Software*, **28**, 1–23.

APPENDIX

A. Stan code for GBXE model

```
MGXE="functions{
 real gbxe_lpdf(real t, real alpha, real beta, real lambda){
  real log_fe;
  log_fe=log(2)+log(alpha)+log(beta)+exponential_lpdf(t|lambda)+
  (2*alpha-1)*exponential_lcdf(t|lambda)-3*log
  (1-(exponential_cdf(t,lambda))^alpha)-
  ((exponential_cdf(t,lambda))^alpha/(1-(exponential_cdf(t,lambda))
  ^alpha))^2 + (beta-1)*log(1-exp(-((exponential_cdf(t,lambda))^alpha
  /(1-(exponential_cdf(t,lambda))^alpha))^2));
  return log_fe;
 }
 real gbxe_lccdf(real t, real alpha, real beta, real lambda){
  real log_ccfe;
  log_ccfe=log(1-(1-exp(-((exponential_cdf(t,lambda))^alpha/
  (1-(exponential_cdf(t,lambda))^alpha))^2))^beta);
  return log_ccfe;
 }
 real surv_gbxe_lpdf(vector t, vector d, real alpha,
 real beta, vector lambda){vector[num_elements(t)] llk_gbxe;
  real prob;
  for(i in 1:num_elements(t)){
   llk_gbxe[i]=log_mix(d[i],gbxe_lpdf(t[i]|alpha,beta,lambda[i]),
   gbxe_lccdf(t[i]|alpha,beta,lambda[i]));
  }
  prob=sum(llk_gbxe);
  return prob;
 }
}
data{
 int N;
 vector<lower=0>[N] y;
 vector<lower=0,upper=1>[N] censor;
 int M;
 matrix[N,M] x;
}
parameters{
 vector[M] b;
 real<lower=0> alpha;
 real<lower=0> beta;
}
transformed parameters{
 vector[N] linpred;
 vector<lower=0>[N] lambda;
 linpred=x*b;
```

```
 for(i in 1:N){
  lambda[i]=exp(-linpred[i]);
 }
}
model{
 //priors
 target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=cauchy_lpdf(beta|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=normal_lpdf(b|0,5);
 //liklihood
 target+=surv_gbxe_lpdf(y|censor,alpha,beta,lambda);
}
generated quantities{
 vector[N] log_lik;
 vector[N] yrepgbxe;
 for(n in 1:N) log_lik[n]=log_mix(censor[n],
 gbxe_lpdf(y[n]|alpha,beta,lambda[n]),
 gbxe_lccdf(y[n]|alpha,beta,lambda[n]));
 {real u;
  u=uniform_rng(0,1);
  for(n in 1:N) yrepgbxe[n]=-1/lambda[n]*
  log(1-(((-log(1-(1-u)^(1/beta)))^(1/2))/
  (1+(-log(1-(1-u)^(1/beta)))^(1/2)))^(1/alpha));
 }
}"
```

B. Stan code for GBXW model

```
MGXW="functions{
 real gbxw_lpdf(real t, real alpha, real beta,real a, real lambda){
  real log_fw;
  log_fw=log(2)+log(alpha)+log(beta)+weibull_lpdf(t|a,lambda)+(2*alpha-1)
  *weibull_lcdf(t|a,lambda)-3*log(1-(weibull_cdf(t,a,lambda))^alpha)-
  ((weibull_cdf(t,a,lambda))^alpha/(1-(weibull_cdf(t,a,lambda))^alpha))^2
  +(beta-1)*log(1-exp(-((weibull_cdf(t,a,lambda))^alpha/
  (1-(weibull_cdf(t,a,lambda))^alpha))^2));
  return log_fw;
 }
 real gbxw_lccdf(real t, real alpha, real beta,real a, real lambda){
  real log_ccfw;
  log_ccfw=log(1-(1-exp(-((weibull_cdf(t,a,lambda))^alpha/
  (1-(weibull_cdf(t,a,lambda))^alpha))^2))^beta);
  return log_ccfw;
 }
 real surv_gbxw_lpdf(vector t, vector d, real alpha, real beta,real a,
 vector lambda){vector[num_elements(t)] llk_gbxw;
```

```
  real prob;
  for(i in 1:num_elements(t)){
   llk_gbxw[i]=log_mix(d[i],gbxw_lpdf(t[i]|alpha,beta,a,lambda[i]),
   gbxw_lccdf(t[i]|alpha,beta,a,lambda[i]));
  }
  prob=sum(llk_gbxw);
  return prob;
 }
}
data{
 int N;
 vector<lower=0>[N] y;
 vector<lower=0,upper=1>[N] censor;
 int M;
 matrix[N,M] x;
}
parameters{
 vector[M] b;
 real<lower=0> alpha;
 real<lower=0> beta;
 real<lower=0> a;
}
transformed parameters{
 vector[N] linpred;
 vector<lower=0>[N] lambda;
 linpred=x*b;
 for(i in 1:N){
  lambda[i]=exp(linpred[i]);
 }
}
model{
 //priors
 target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=cauchy_lpdf(beta|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=cauchy_lpdf(a|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=normal_lpdf(b|0,5);
 //liklihood
 target+=surv_gbxw_lpdf(y|censor,alpha,beta,a,lambda);
}
generated quantities{
 vector[N] log_lik;
 vector[N] yrepgbxw;
 for(n in 1:N) log_lik[n]=log_mix(censor[n],
 gbxw_lpdf(y[n]|alpha,beta,a,lambda[n]),
 gbxw_lccdf(y[n]|alpha,beta,a,lambda[n]));
 {real u;
  u=uniform_rng(0,1);
```

```
  for(n in 1:N) yrepgbxw[n]=lambda[n]*
  (-log(1-(((-log(1-(1-u)^(1/beta)))^(1/2))/
  (1+(-log(1-(1-u)^(1/beta)))^(1/2)))^(1/alpha)))^(1/a);
 }
}"
```

C. Stan code for GBXLx model

```
MGXL="functions{
 real gbxl_lpdf(real t, real alpha, real beta,real a, real lambda){
  real log_fl;
  log_fl=log(2)+log(alpha)+log(beta)+pareto_type_2_lpdf(t|0,lambda,a)
  +(2*alpha-1)*pareto_type_2_lcdf(t|0,lambda,a)-
  3*log(1-(pareto_type_2_cdf(t,0,lambda,a))^alpha)-
  ((pareto_type_2_cdf(t,0,lambda,a))^alpha/
  (1-(pareto_type_2_cdf(t,0,lambda,a))^alpha))^2+(beta-1)*
  log(1-exp(-((pareto_type_2_cdf(t,0,lambda,a))^alpha/
  (1-(pareto_type_2_cdf(t,0,lambda,a))^alpha))^2));
  return log_fl;
 }
 real gbxl_lccdf(real t, real alpha, real beta,real a, real lambda){
  real log_ccfl;
  log_ccfl=log(1-(1-exp(-((pareto_type_2_cdf(t,0,lambda,a))^alpha/
  (1-(pareto_type_2_cdf(t,0,lambda,a))^alpha))^2))^beta);
  return log_ccfl;
 }
 real surv_gbxl_lpdf(vector t, vector d, real alpha, real beta,
 real a, vector lambda){
  vector[num_elements(t)] llk_gbxl;
  real prob;
  for(i in 1:num_elements(t)){
   llk_gbxl[i]=log_mix(d[i],gbxl_lpdf(t[i]|alpha,beta,a,lambda[i]),
   gbxl_lccdf(t[i]|alpha,beta,a,lambda[i]));
  }
  prob=sum(llk_gbxl);
  return prob;
 }
}
data{
 int N;
 vector<lower=0>[N] y;
 vector<lower=0,upper=1>[N] censor;
 int M;
 matrix[N,M] x;
}
parameters{
```

```
 vector[M] b;
 real<lower=0> alpha;
 real<lower=0> beta;
 real<lower=0> a;
}
transformed parameters{
 vector[N] linpred;
 vector<lower=0>[N] lambda;
 linpred=x*b;
 for(i in 1:N){
  lambda[i]=exp(linpred[i]);
 }
}
model{
 //priors
 target+=cauchy_lpdf(alpha|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=cauchy_lpdf(beta|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=cauchy_lpdf(a|0,25)- 1 * cauchy_lccdf(0|0,25);
 target+=normal_lpdf(b|0,5);
 //liklihood
 target+=surv_gbxl_lpdf(y|censor,alpha,beta,a,lambda);
}
generated quantities{
 vector[N] log_lik;
 vector[N] yrepgbxl;
 for(n in 1:N) log_lik[n]=log_mix(censor[n],
 gbxl_lpdf(y[n]|alpha,beta,a,lambda[n]),
 gbxl_lccdf(y[n]|alpha,beta,a,lambda[n]));
 {real u;
  u=uniform_rng(0,1);
  for(n in 1:N) yrepgbxl[n]=lambda[n]*((1-
  (((-log(1-(1-u)^(1/beta)))^(1/2))/
  (1+(-log(1-(1-u)^(1/beta)))^(1/2)))^(1/alpha))^(-1/a)-1);
 }
}"
```