

Análisis bayesiano para la distribución lognormal generalizada aplicada a modelos de falla con censura

Bayesian Analysis for the Generalized Lognormal Distribution Applied to Failure Time Analysis

FREDDY HERNÁNDEZ^{1,a}, OLGA CECILIA USUGA^{1,2,b}

¹DEPARTAMENTO DE ESTADÍSTICA, INSTITUTO DE MATEMÁTICA Y ESTADÍSTICA, UNIVERSIDADE DE SÃO PAULO, SÃO PAULO, BRASIL

²INGENIERÍA INDUSTRIAL, FACULTAD DE INGENIERÍAS, UNIVERSIDAD DE ANTIOQUIA, MEDELLÍN, COLOMBIA

Resumen

Existen varias versiones de la distribución lognormal en la literatura estadística y una de ellas está basada en la transformación exponencial de la distribución normal generalizada (NG). En el presente artículo se presenta el análisis Bayesiano para la distribución lognormal generalizada (logNG) considerando distribuciones a priori de Jeffreys independientes para los parámetros; así como el procedimiento para implementar el muestreador de Gibbs que permite obtener las distribuciones a posteriori de los parámetros. Los resultados obtenidos son usados para analizar modelos de tiempo de falla con datos no censurados y censurados a derecha Tipo I. El procedimiento propuesto es ilustrado usando una base de datos real relacionada con tiempos de falla de computadores.

Palabras clave: análisis de tiempo de falla, censura a derecha, distribución lognormal generalizada, inferencia bayesiana, muestreador de Gibbs.

Abstract

There are several versions of the lognormal distribution in the statistical literature, one is based in the exponential transformation of generalized normal distribution (GN). This paper presents the Bayesian analysis for the generalized lognormal distribution (logGN) considering independent non-informative Jeffreys distributions for the parameters as well as the procedure for implementing the Gibbs sampler to obtain the posterior distributions of parameters. The results are used to analyze failure time models with right-censored and uncensored data. The proposed method is illustrated using actual failure time data of computers.

Key words: Bayesian inference, Failure time analysis, Gibbs sampling, Lognormal distribution, Right censoring.

^aEstudiante de doctorado. E-mail: fhernanb@ime.usp.br

^bProfesora asistente. E-mail: ousuga@udea.edu.co

1. Introducción

En estudios de confiabilidad la distribución exponencial tiene un papel fundamental desde el punto de vista conceptual y práctico; sin embargo, algunas veces esta distribución no proporciona ajustes apropiados para modelar los datos obtenidos de un experimento, esto mismo sucede con otras distribuciones como la Weibull y Gamma; por lo tanto, una buena opción consiste en analizar los datos usando la distribución lognormal (Chen 1995). Se han obtenido buenos ajustes usando la distribución lognormal para el caso de conjuntos de datos observados y datos experimentales (Aitchison & Brown 1957) para modelar fallas en pruebas de vida (Chen & Papadopoulos 1997) y ha sido usada específicamente en el campo de la electrónica para analizar tiempos de vida de mecanismos de conducción eléctrica (Howard & Dodson 1961) y en tiempos de vida de transistores de germanio (Adam 1962).

Desde el punto de vista bayesiano para problemas de tiempos de falla la distribución lognormal de tres parámetros propuesta por Lawless (1982) fue analizada por Upadhyay & Peshwani (2001) para el caso en el que existen observaciones censuradas y no censuradas; este análisis fue realizado usando el muestreador de Gibbs, el cual es una herramienta importante para obtener la distribución a posteriori de los parámetros y que exige diseño, implementación y validación de algoritmos apropiados (Barrera & Correa 2008). Otra versión de la distribución lognormal generalizada con tres parámetros basada en la transformación exponencial de la distribución normal generalizada propuesta por Nadarajah (2005) fue estudiada por Martín & Pérez (2009) usando el muestreador de Gibbs pero sólo para el caso de observaciones sin censura.

El objetivo de este artículo consiste en extender la propuesta de análisis bayesiano presentada por Martín & Pérez (2009) usando el enfoque de Upadhyay & Peshwani (2001) para la distribución logNG propuesta por Nadarajah (2005). Se consideraron dos situaciones: la primera, cuando se tienen observaciones sin censura, y la segunda, cuando se tiene censura a derecha Tipo I. El análisis bayesiano propuesto está basado en el muestreador de Gibbs y se presenta una aplicación para una base de datos relacionada a los tiempos para que se presente la primera falla en un conjunto de computadores nuevos.

En la segunda sección de este artículo se presenta la versión de la distribución lognormal generalizada con tres parámetros a estudiar y el procedimiento para generar observaciones aleatorias de esta distribución. En la tercera sección se presenta el análisis bayesiano y los algoritmos para la implementación del muestreador de Gibbs considerando el enfoque no informativo en los casos con y sin observaciones censuradas. En la cuarta sección se presenta la aplicación del procedimiento propuesto en el artículo para analizar una base de datos real.

2. Distribución lognormal generalizada

En la literatura estadística se pueden encontrar varias versiones de la distribución lognormal propuestas por Lawless (1982), Chen (1995) y otra versión pre-

sentada en esta sección que es obtenida através de la transformación exponencial de una variable que sigue una distribución normal generalizada propuesta por Nadarajah (2005).

Una variable aleatoria X sigue distribución lognormal generalizada con tres parámetros si la transformación $Y = \log X$ sigue una distribución normal generalizada. La función de densidad para una variable logNG con parámetros μ , σ y s es dada por

$$f(x | \mu, \sigma, s) = \frac{s}{2x\sigma\Gamma(1/s)} \exp\left(-\left|\frac{\log x - \mu}{\sigma}\right|^s\right) \quad (1)$$

donde $x > 0$, $-\infty < \mu < +\infty$, $\sigma > 0$, $s \geq 1$ y $\Gamma(\cdot)$ corresponde a la función Gamma.

Otras distribuciones como la lognormal y la log-Laplace se obtienen a partir de la expresión (1), tomando $s = 2$ y cambiando σ por $\sqrt{2}\sigma$ se obtiene la distribución lognormal y cuando $s = 1$ se obtiene la distribución log-Laplace. Martín & Pérez (2009) afirman que las densidades de la logNG con $s \in (1, 2) \cup (2, 3)$ son apropiadas para el ajuste de datos en los cuales la lognormal no genera ajustes satisfactorios. Una característica importante de la familia logNG es que todas las densidades están concentradas a la izquierda (véase figura 1), y cuando X tiende infinito, la densidad disminuye lo cual es apropiado para modelar variables asociadas a tiempos de vida.

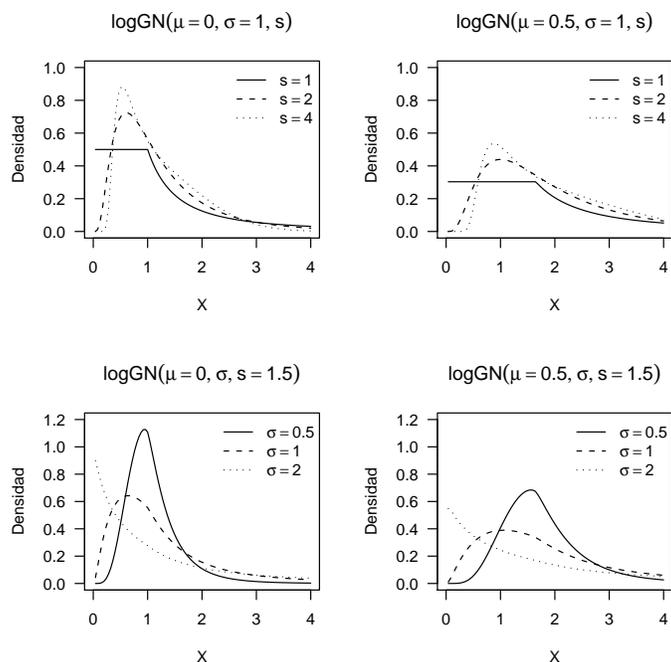


FIGURA 1: Densidades para la familia logNG

La siguiente proposición fue presentada por Martín & Pérez (2009) y muestra la relación entre la distribución Gamma y la logNG y, por tanto, un método para generar variables aleatorias de una distribución logNG.

Proposición 1. Sean U y X dos variables aleatorias tales que $U \sim \text{Gamma}(\alpha = 1 + 1/s, \gamma = 1)$ y $f(x | u) = \mathcal{I}[\exp(\mu - \sigma u^{1/s}) < x < \exp(\mu + \sigma u^{1/s})] / (2x\sigma u^{1/s})$ entonces $X \sim \text{logNG}(\mu, \sigma, s)$.

donde $\mathcal{I}[\cdot]$ corresponde a la función indicadora.

El proceso propuesto por Martín & Pérez (2009) basado en la proposición anterior para generar observaciones de una distribución logNG(μ, σ, s) consta de los siguientes tres pasos.

1. Generar $U \sim \text{Gamma}(\alpha = 1 + 1/s, \gamma = 1)$
2. Generar $V \sim \text{Unif}(0, 1)$
3. Hacer $X = \exp(\sigma U^{1/s} V + \mu)$

La función de sobrevivencia definida como $S(x) = P(X > x)$ corresponde a la probabilidad de que un individuo sobreviva más allá del tiempo x y en el contexto de fallas de equipos $S(x)$ es llamada función de confiabilidad (Klein & Moeschberger 2003); para la distribución logNG, la función de sobrevivencia está dada por:

$$S(x) = \begin{cases} 1 - \frac{\Gamma\left(\frac{1}{s}, \left(\frac{\mu - \log(x)}{\sigma}\right)^s\right)}{2\Gamma(1/s)} & \text{si } 0 < x \leq \exp(\mu) \\ \frac{\Gamma\left(\frac{1}{s}, \left(\frac{\log(x) - \mu}{\sigma}\right)^s\right)}{2\Gamma(1/s)} & \text{si } x > \exp(\mu) \end{cases}$$

donde $\Gamma(\cdot, \cdot)$ denota la función Gamma incompleta.

La función de riesgo definida como $r(x) = f(x)/S(x)$ se conoce en sobrevivencia como tasa condicional de falla (Klein & Moeschberger 2003) y para la distribución logNG es dada por:

$$r(x) = \begin{cases} \frac{s \exp\left(-\left(\frac{\mu - \log(x)}{\sigma}\right)^s\right)}{x\sigma\left(2\Gamma(1/s) - \Gamma\left(\frac{1}{s}, \left(\frac{\mu - \log(x)}{\sigma}\right)^s\right)\right)} & \text{si } 0 < x \leq \exp(\mu) \\ \frac{s \exp\left(-\left(\frac{\log(x) - \mu}{\sigma}\right)^s\right)}{x\sigma\Gamma\left(\frac{1}{s}, \left(\frac{\log(x) - \mu}{\sigma}\right)^s\right)} & \text{si } x > \exp(\mu) \end{cases}$$

En la figura 2 se presentan las funciones de riesgo para dos densidades particulares de la distribución logNG. Para ver propiedades de la función de sobrevivencia y riesgo, véase Gupta & Lvin (2005).

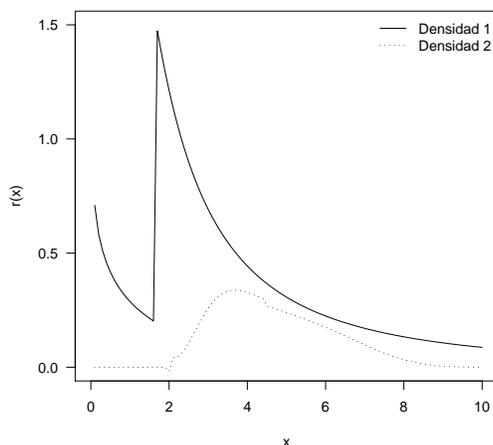


FIGURA 2: Función de riesgo. Densidad 1 ($\mu = 0.5$, $\sigma = 2$, $s = 1.5$) y Densidad 2 ($\mu = 1.5$, $\sigma = 0.5$, $s = 3$)

3. Análisis bayesiano

Según Robert (2001), el principal propósito de la teoría estadística es que a partir de un modelo estadístico basado en las observaciones recolectadas de un fenómeno aleatorio se puedan obtener inferencias sobre la distribución de probabilidad asociada al fenómeno estudiado. El modelo estadístico bayesiano está compuesto de dos elementos, el primer elemento corresponde al modelo estadístico paramétrico dado por $f(x | \theta)$ donde x corresponde a la información obtenida de los datos y θ al parámetro de la distribución asociada al fenómeno; el segundo elemento $\pi(\theta)$ corresponde a la distribución a priori para el parámetro. Estos dos elementos combinados dan lugar al modelo bayesiano $\pi(\theta | x)$ dado por $\pi(\theta | x) \propto f(x | \theta)\pi(\theta)$. La novedad de la aproximación bayesiana es que pone las causas (observaciones) y los efectos (parámetros) en el mismo nivel considerando para ambos distribuciones de probabilidad (Robert 2001).

Berger (1985) asegura que el enfoque bayesiano ofrece la posibilidad de incluir en el modelo la opinión de especialistas por medio de la distribución a priori en el proceso de inferencia. Diversos trabajos relacionados con la distribución lognormal tradicional han usado a priori de Jeffreys, Padgett & Johnson (1983), Upadhyay & Peshwani (2001, 2003, 2008). Martín & Pérez (2009) propusieron la utilización de la distribución a priori no informativa de Jeffreys para la distribución logNG. Portela & Gómez-Villegas (2004) sugieren usar distribuciones a priori independientes para cada uno de los parámetros en las distribuciones de la familia de distribuciones Exponencial Potencia a la cual pertenece la logNG. En este trabajo se consideran distribuciones a priori de Jeffreys independientes para los parámetros, este tipo de

distribuciones no informativas son útiles cuando las opiniones de los especialistas o conocimientos previos difieren (Gelman, Stern & Rubin 2004).

3.1. Enfoque sin censura

En virtud de la proposición 1 es posible escribir la distribución para logNG como la distribución marginal para x de $f(x | \mu, \sigma, s, u)f(u | s)$, donde

$$f(x | \mu, \sigma, s, u) = \frac{1}{2x\sigma u^{1/s}} \mathcal{I} \left[\exp(\mu - \sigma u^{1/s}) < x < \exp(\mu + \sigma u^{1/s}) \right] \quad (2)$$

$$f(u | s) = \text{Gamma}(1 + 1/s, 1) \quad (3)$$

Así la función de verosimilitud para los parámetros μ, σ, s y $\mathbf{u} = (u_1, u_2, \dots, u_n)'$ dada una muestra aleatoria $\mathbf{x} = (x_1, x_2, \dots, x_n)'$ es dada por

$$\begin{aligned} L(\mu, \sigma, s, \mathbf{u} | \mathbf{x}) &= \prod_{i=1}^n f(x_i | \mu, \sigma, s, u_i) f(u_i | s) \\ &= \frac{s^n}{(2\sigma)^n \Gamma^n(1/s)} \prod_{i=1}^n \frac{e^{-u_i}}{x_i} \mathcal{I} \left[\exp(\mu - \sigma u_i^{1/s}) < x_i < \exp(\mu + \sigma u_i^{1/s}) \right] \end{aligned} \quad (4)$$

Considerando el siguiente conjunto de distribuciones a priori de Jeffreys independientes para los parámetros

$$\pi(\mu) \propto 1, \quad \pi(\sigma) \propto \frac{1}{\sigma}, \quad \pi(s) \propto \frac{1}{s}$$

La distribución a posteriori obtenida es dada por

$$\pi(\mu, \sigma, s, \mathbf{u} | \mathbf{x}) \propto \frac{s^{n-1}}{\sigma^{n+1} \Gamma^n(1/s)} \prod_{i=1}^n \frac{e^{-u_i}}{x_i} \mathcal{I} \left[\exp(\mu - \sigma u_i^{1/s}) < x_i < \exp(\mu + \sigma u_i^{1/s}) \right] \quad (5)$$

En el caso de modelos multiparamétricos la distribución a posteriori conjunta no siempre presenta una forma conocida y, por lo, tanto es difícil obtener muestras aleatorias; sin embargo, es posible que a partir de las distribuciones a posteriori marginales de cada uno de los parámetros se puedan obtener muestras aleatorias con mayor facilidad. En estos casos, una aproximación a la distribución a posteriori conjunta puede realizarse usando el muestreador de Gibbs, consiste en un algoritmo iterativo para construir una secuencia dependiente de valores para los parámetros que convergen a los parámetros de la distribución a posteriori conjunta estudiada (Hoff 2009). El conjunto de distribuciones a posteriori marginal para cada uno de los parámetros de la distribución a posteriori conjunta de la expresión (5) es dada

por:

$$\pi(\mu \mid \sigma, s, \mathbf{u}, \mathbf{x}) \propto 1, \quad \max_i \left\{ \log(x_i) - \sigma u_i^{1/s} \right\} < \mu < \min_i \left\{ \log(x_i) + \sigma u_i^{1/s} \right\} \quad (6)$$

$$\pi(\sigma \mid \mu, s, \mathbf{u}, \mathbf{x}) \propto \frac{1}{\sigma^{n+1}}, \quad \sigma > \max_i \left\{ \frac{|\mu - \log(x_i)|}{u_i^{1/s}} \right\} \quad (7)$$

$$\pi(s \mid \sigma, \mu, \mathbf{u}, \mathbf{x}) \propto \frac{s^{n-1}}{\Gamma^n(1/s)}, \quad s < \frac{\log(u_i)}{\log \left| \frac{\log(x_i) - \mu}{\sigma} \right|} \quad (8)$$

$$\pi(u_i \mid \sigma, s, \mu, \mathbf{x}) \propto e^{-u_i}, \quad u_i > \left(\frac{|\log(x_i) - \mu|}{\sigma} \right)^s, \quad i = 1, 2, \dots, n \quad (9)$$

Para generar observaciones aleatorias de las distribuciones a posteriori anteriores es importante identificar los núcleos característicos, la densidad (6) corresponde a la distribución uniforme, en la densidad (7) se tiene el núcleo de la distribución Pareto y en la densidad (9) se tiene el núcleo de la exponencial, mientras que para la densidad (8) es necesario usar el método de rechazo para obtener observaciones aleatorias (véase Gamerman & Lopes 2006).

3.2. Enfoque con censura

Supongamos que la muestra aleatoria \mathbf{x} obtenida tiene observaciones con censura a derecha Tipo I, es decir, existen r observaciones $\mathbf{x}_1 = (x_1, x_2, \dots, x_r)'$ de las n en las cuales fueron observados tiempos de falla, mientras que $\mathbf{x}_2 = (x_{r+1}, x_{r+2}, \dots, x_n)'$ corresponde a las $n - r$ observaciones con censura, donde $\mathbf{x} = \mathbf{x}_1 \cup \mathbf{x}_2$. Así la función de verosimilitud dada en (4) ahora se define como

$$L(\mu, \sigma, s, \mathbf{u} \mid \mathbf{x}) = \prod_{i=1}^r f(x_i \mid \mu, \sigma, s, u_i) \prod_{j=r+1}^n \int_{x_j}^{\infty} f(x_j \mid \mu, \sigma, s, u_j) dx_j \quad (10)$$

Considerando las mismas distribuciones a priori de la sección anterior, la distribución a posteriori es ahora dada por

$$\pi(\mu, \sigma, s, \mathbf{u} \mid \mathbf{x}_1, \mathbf{x}_2) \propto P_1 P_2 \quad (11)$$

donde

$$P_1 = \frac{s^{r-1}}{\sigma^{r+1} \Gamma^r(1/s)} \prod_{i=1}^r \frac{e^{-u_i}}{x_i} \mathcal{I} \left[\exp(\mu - \sigma u_i^{1/s}) < x_i < \exp(\mu + \sigma u_i^{1/s}) \right]$$

$$P_2 = \prod_{j=r+1}^n \int_{x_j}^{\infty} f(x_j \mid \mu, \sigma, s, u_j) dx_j$$

El cálculo de la distribución a posteriori dada en (11) es complicado por causa de P_2 debido a las observaciones censuradas; una manera de tratar este problema, el cual es el objetivo del presente trabajo, consiste en considerar \mathbf{x}_2 , el conjunto

de observaciones con censura, como desconocidos e incluirlo en el muestreador de Gibbs (Upadhyay & Peshwani 2001). Las distribuciones a posteriori para los parámetros de la logNG y para las observaciones con censura son entonces dadas por

$$\pi(\mu, \sigma, s, \mathbf{u} \mid \mathbf{x}_1, \mathbf{x}_2) = \pi(\mu, \sigma, s, \mathbf{u} \mid \mathbf{x}) \quad (12)$$

$$\pi(\mathbf{x}_2 \mid \mu, \sigma, s, \mathbf{u}, \mathbf{x}_1) = \pi(\mathbf{x}_2 \mid \mu, \sigma, s, \mathbf{u}) \quad (13)$$

El enfoque propuesto tiene dos etapas: la primera consiste en incluir el conjunto de observaciones con censura como conocido (suponiendo valores iguales a la censura) en el muestreador de Gibbs para calcular la distribución a posteriori de los parámetros de la logNG usando la expresión (12), la cual corresponde a la distribución a posteriori de los parámetros de la logNG, sin considerar censura de la expresión (5). La segunda etapa consiste en que una vez que son actualizados los valores de los parámetros de la distribución logNG se generan nuevos valores para las observaciones censuradas usando la expresión (13) que se reduce a generar $n - r$ observaciones independientes x de una distribución logNG truncada de tal forma que $x > x_j$ para $j = r + 1, \dots, n$.

3.3. Sistemas de generación

A continuación se presenta el procedimiento usado para la aplicación del muestreador de Gibbs al problema estudiado.

3.3.1. Sin censura

1. Generar valores iniciales para μ , σ , s y u_i .
2. Generar una observación para μ según la distribución uniforme dada en (6).
3. Generar una observación para σ según la distribución Pareto dada en (7).
4. Generar una observación para s según la distribución dada en (8) usando el método de muestreo por rechazo implementado en el paquete Runuran de R Development Core Team (2010).
5. Generar u_i según la distribución exponencial dada en (9).

Una vez actualizados los valores de los parámetros μ , σ , s y u_i repetir los pasos 2 al 5 y continuar el proceso.

3.3.2. Con censura

Cuando la base de datos contiene observaciones censuradas, el procedimiento para la aplicación del muestreador de Gibbs es en esencia similar a los pasos de la subsección anterior, pero con algunos cambios. En la primera iteración es necesario considerar temporalmente las r observaciones censuradas como no censuradas, y

el valor de la observación corresponde al valor de la censura, esto se hace así para generar valores iniciales de los parámetros usando toda la información disponible. Después se aplican los pasos 2 al 5 más un sexto paso adicional que consiste en generar r observaciones provenientes de una distribución logNG truncada en los valores de censura. Después volver al paso 2 y continuar el proceso iterativo.

4. Aplicación

La aplicación se realizó usando la base de datos presentada por Barrera & Correa (2008) obtenida del departamento de apoyo técnico de la Universidad Nacional de Colombia, la cual contiene información sobre el tiempo para la presencia de la primera falla en un conjunto de 72 computadores nuevos con iguales características, comprados todos en la misma fecha y observados hasta transcurridos 66 meses.

En la tabla 1 se presentan los datos. Se observa que de los 72 computadores solo 17 fallaron antes de terminar el horizonte de observación mientras que los 55 datos restantes denotados por 66^+ indican que estos computadores no presentaron fallas durante el horizonte del estudio.

TABLA 1: Tiempos para la primera falla de 72 computadores (meses).

14.07	17.80	19.43	21.33	24.60	28.97	29.63	33.73	37.60	37.67	40.87	52.40
53.97	60.57	64.27	65.43	65.43	66 ⁺						
66 ⁺											
66 ⁺											
66 ⁺											

Se realizaron dos análisis a la base de datos. En el primero se consideró la distribución logNG para los tiempos de falla y luego aplicando los pasos de la sección 3.3.1 con diez mil iteraciones fueron obtenidas las distribuciones marginal a posteriori de los 3 parámetros. En el segundo análisis se consideró la distribución lognormal y nuevamente aplicando los pasos de la sección 3.3.2, pero teniendo en cuenta que la distribución lognormal es un caso particular de la logNG cuando $s = 2$ y $\sigma = \sqrt{2}\sigma$, así fueron nuevamente obtenidas las distribuciones marginal a posteriori.

En la figura 3 se presentan los resultados obtenidos del muestreador de Gibbs, en la parte izquierda se tienen las distribuciones marginales para el caso de la distribución logNG, mientras que en la parte derecha están las distribuciones marginales con la distribución lognormal; además, se incluyeron las regiones de mayor densidad (High Density Regions (HDR)) a un nivel de confianza del 90 % las cuales se calcularon con el paquete *hdrnde* de R Development Core Team (2010). Se observa que las distribuciones y las regiones de mayor densidad para μ son similares en ambos casos, se observa también que la distribución para S está concentrada cerca del valor de 2 y la región de mayor densidad incluye este valor de 2.

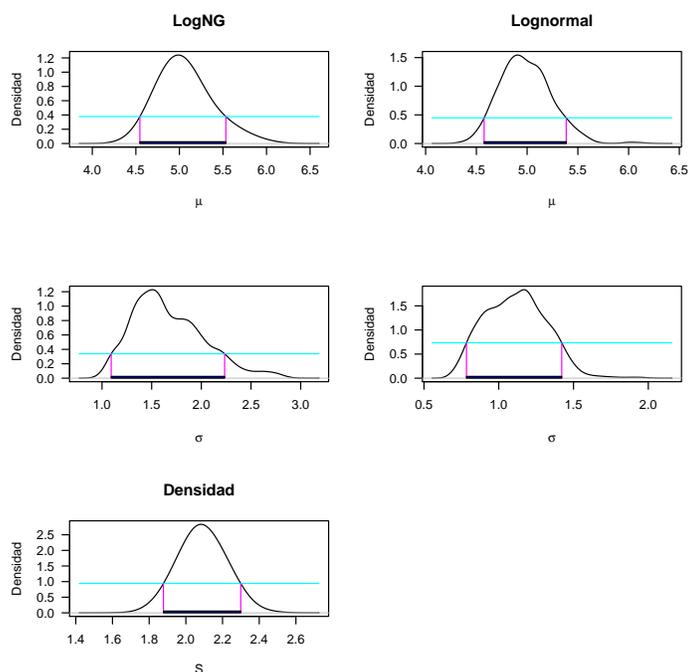


FIGURA 3: Distribución marginal a posteriori para los parámetros con HDR de 90 %.

En la tabla 2 se presentan las modas para las distribuciones marginales de los parámetros de la figura 3 en los casos considerados. Se observa que las modas de μ y σ son cercanas para ambos casos y la moda para S está cercana al valor de 2.

TABLA 2: Moda para las distribuciones marginales

	μ	σ	s
logNG	4.99	1.51	2.08
log	4.91	1.17	-

En la figura 4 se presentan dos distribuciones predictivas, una de ellas corresponde al caso de la distribución logNG y la otra a la distribución lognormal como posibles distribuciones para el tiempo de falla de los computadores. En la tabla 3 se presentan algunos cuantiles de interés para las distribuciones predictivas de la figura 4. De la figura se observa que existe una gran similitud entre las dos distribuciones, ya que las curvas son bastante cercanas entre sí; además, los percentiles son cercanos, especialmente el percentil 25.

Se usó el criterio de información de desvianza (CID) propuesto por Spiegelhalter, Best, Carlin & van der Linde (2002), el cual es un criterio de comparación de modelos bayesianos, el modelo con el menor CID es elegido como el modelo que mejor predice el conjunto de datos. Al calcular el CID para la aplicación se

encontró que $CID_{logNG} = 916$ y $CID_{lognormal} = 941$. Esto indica que el modelo más apropiado corresponde al modelo logNG.

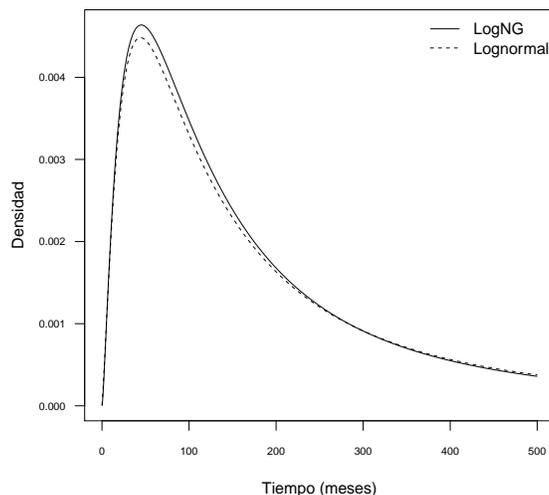


FIGURA 4: Distribución predictiva.

TABLA 3: Percentiles para las distribuciones predictivas.

Percentil	25	50	75
logNG	69.9	150.4	338.0
log	68.0	142.0	306.3

5. Conclusiones

En el artículo se ha propuesto la metodología bayesiana para usar la distribución lognormal generalizada con tres parámetros con el objetivo de estudiar modelos de falla con censura a derecha. Se consideraron a priori de Jeffreys independientes para los parámetros y se implementó el muestreador de Gibbs para obtener las distribuciones a posteriori de los parámetros. El procedimiento se ilustró usando una base de datos real correspondiente a los tiempos de falla de computadores, teniendo la base de datos observaciones con censura a derecha. Se consideraron las distribuciones lognormal y lognormal generalizada como posibles distribuciones para los tiempos de falla.

Como posible trabajo futuro se pueden generar nuevos procedimientos considerando otras distribuciones a priori para los parámetros, así como otro tipo de distribuciones para los tiempos de falla. La comparación del desempeño de

los procedimientos propuestos se podría realizar por medio del escore logarítmico propuesto por Bernardo (1979) mediante simulaciones.

Agradecimientos

Agradecemos a los revisores anónimos por sus comentarios que permitieron mejorar el presente trabajo.

[Recibido: agosto de 2010 — Aceptado: febrero de 2011]

Referencias

- Adam, J. (1962), 'Failure time distribution estimation', *Semiconductor Reliability* **2**, 41–52.
- Aitchison, J. & Brown, J. (1957), *The Lognormal Distribution*, Cambridge University Press, United Kingdom.
- Barrera, C. & Correa, J. (2008), 'Distribución predictiva bayesiana para modelos de pruebas de vida vía MCMC', *Revista Colombiana de Estadística* **31**(2), 145–155.
- Berger, J. (1985), *Statistical Decision Theory and Bayesian Analysis*, Springer, New York.
- Bernardo, J. (1979), 'Expected information as expected utility', *Annals of Statistics* **7**(3), 686–690.
- Chen, G. (1995), 'Generalized log-normal distributions with reliability application', *Computational Statistics and Data Analysis* **19**(3), 309–319.
- Chen, K. & Papadopoulos, A. (1997), 'Shortest Bayes credibility intervals for the lognormal failure model', *Microelectron Reliability* **37**(12), 1859–1863.
- Gamerman, D. & Lopes, H. (2006), *Markov Chain Monte Carlo Stochastic Simulation for Bayesian Inference*, Chapman and Hall/CRC, Boca Raton.
- Gelman, A., Stern, J. & Rubin, H. (2004), *Bayesian Data Analysis*, Chapman & Hall-CRC.
- Gupta, R. & Lvin, S. (2005), 'Reliability functions of generalized log-normal model', *Mathematical and Computer Modelling* **42**, 939–946.
- Hoff, P. (2009), *A First Course in Bayesian Statistical Methods*, Springer, New York.
- Howard, B. & Dodson, G. (1961), 'High stress aging to failure of semiconductor devices', *Proc. Seventh National Symposium on Reliability and Quality Control* pp. 201–207.

- Klein, J. & Moeschberger, M. (2003), *Survival Analysis: Techniques for Censored and Truncated Data*, Springer-Verlag, New York.
- Lawless, J. F. (1982), *Statistical Models and Methods for Lifetime Data*, Wiley, New York.
- Martín, J. & Pérez, C. (2009), 'Bayesian analysis of a generalized lognormal distribution', *Computational Statistics and Data Analysis* **53**, 1377–1387.
- Nadarajah, S. (2005), 'A generalized normal distribution', *Journal of Applied Statistics* **37**(7), 685–694.
- Padgett, W. & Johnson, M. (1983), 'Lower bounds on reliability in the lognormal distribution', *The Canadian Journal of Statistics - La Revue Canadienne de Statistique* **11**(2), 137–147.
- Portela, J. & Gómez-Villegas, M. (2004), 'Implementation of a robust Bayesian method', *Journal of Statistical Computation and Simulation* **74**(4), 235–248.
- R Development Core Team (2010), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
*<http://www.R-project.org>
- Robert, C. (2001), *The Bayesian Choice*, second edn, Springer, New York.
- Spiegelhalter, D., Best, N., Carlin, B. & van der Linde, A. (2002), 'Bayesian measures of model complexity and fit', *Journal of The Royal Statistical Society* **64**, 583–639.
- Upadhyay, S. & Peshwani, M. (2001), 'Full posterior analysis of three parameter Lognormal distribution using Gibbs sampler', *Journal of Statistical Computation and Simulation* **71**(3), 215–230.
- Upadhyay, S. & Peshwani, M. (2003), 'Choice between Weibull and lognormal models: A simulation based Bayesian study', *Communications in Statistics - Theory and Methods* **32**(2), 381–405.
- Upadhyay, S. & Peshwani, M. (2008), 'Posterior analysis of lognormal regressions models using the Gibbs sampler', *Statistical Papers* **49**, 59–85.

Apéndice

A continuación se presenta el código usado en R Development Core Team (2010) para la implementación del muestreador de Gibbs.

```

#!----- LIBRERIAS REQUERIDAS -----
require(hbmem)
require(Runuran)
require(VGAM)
require(hdrcde)
#!----- FUNCIONES -----
#! fdp para la lognormal generalizada con los parametros mu sigma e s
flog=function(x,mu,sigma,s) s*exp(-abs((log(x)-mu)/sigma)^s)/(2*x*sigma*gamma(1/s))
#!----- Generador de observaciones aleatoria logGN truncadas -----
rlogGN = function(n,MU,SIGMA,S,minimo=0,center=50) {
  flog = function(x,mu,sigma,s) s*exp(-abs((log(x)-mu)/sigma)^s)/(2*x*sigma*gamma(1/s))
  logflog = function(x,mu,sigma,s) log(s/(2*x*sigma*gamma(1/s)))+-abs((log(x)-mu)/sigma)^s
  a = function(x) flog(x,mu=MU,sigma=SIGMA,s=S) # f(x)
  b = function(x) logflog(x,mu=MU,sigma=SIGMA,s=S) # log f(x)
  generador = pinv.new(pdf=a,lb=minimo,ub=Inf, islog=FALSE, center=center)
  ur(generador,n) }
#!-----
rexp.trunc=function(n=1,rate,mini) {
  x = urexp(n=n, rate=1/rate, lb=mini, ub=Inf)
  x }
#!----- Funciones para muestrear de las a posteriori, caso logNG -----
rpost.mu=function(sigma,s,u,x) { # u y x son vectores
  l.inf = max(log(x)-sigma*u^(1/s))
  l.sup = min(log(x)+sigma*u^(1/s))
  if (l.inf<=l.sup) {
    rmu=runif(n=1,min=l.inf,max=l.sup)
  }
  else rmu=runif(n=1,min=l.sup,max=l.inf)
  rmu }
#!-----
rpost.sigma=function(mu,s,u,x) { # u y x son vectores
  n=length(x)
  minimo = max(abs(mu-log(x))/u^(1/s))
  rpareto(n=1, shape=n, location=minimo) }
#!-----
rpost.u = function(x,mu,sigma,s) {
  mini = (abs(log(x)-mu)/sigma)^s
  n=length(x)
  muestra=NULL
  for (i in 1:n) muestra[i]=urexp(n=1,rate=1,lb=mini[i],ub=Inf)
  muestra }
#!-----
rpost.s = function(sigma,mu,u,x) { # u y x son vectores
  N = length(x)
  limite = log(u)/log(abs(mu-log(x))/sigma) # calcula los limites para s
  s.1 = which(limite>=1) # obtiene los índices donde el limite es >=1
  mini = min(c(1,s.1))
  maxi = max(s.1)
  pdf.s = function(x,n=N) { x^(n-1)/(gamma(x))^n }
  gen.s = tdr.new(pdf=pdf.s,n=N,lb=1,ub=10)
  ur(gen.s,n=1) }
#!----- Funciones para muestrear de las a posteriori, caso lognormal -----
rpost.mu.l=function(sigma,u,x) { # u y x son vectores
  l.inf = max(log(x)-sigma*(2*u)^(1/2))
  l.sup = min(log(x)+sigma*(2*u)^(1/2))
  if (l.inf<=l.sup) {
    rmu=runif(n=1,min=l.inf,max=l.sup)
  }
  else rmu=runif(n=1,min=l.sup,max=l.inf)
  rmu }
#!-----
rpost.sigma.l=function(mu,u,x) { # u y x son vectores
  n=length(x)

```

```

minimo = max(abs(mu-log(x))/(2*u)^(1/2))
rpareto(n=1, shape=n, location=minimo) }
#!-----
rpost.u.l = function(x,mu,sigma) {
mini = (abs(log(x)-mu)/(sqrt(2)*sigma))^2
n=length(x)
muestra=NULL
for (i in 1:n) muestra[i]=urexp(n=1,rate=1,lb=mini[i],ub=Inf)
muestra }
#!-----
#!----- Análisis de los datos con censura considerando logNG -----
#!-----
dsc=c(14.07, 17.80, 19.43, 21.33, 24.60, 28.97, 29.63, 33.73, 37.60, 37.67,
      40.87, 52.40, 53.97, 60.57, 64.27, 65.43, 65.43)
dcc=rep(x=66,times=55) ; datos = c(dsc,dcc)
#!-----
#! Definiendo los valores iniciales de los parametros
n=length(dsc) ; n.censu=length(dcc)
mu0=3.5 ; sigma0=1 ; s0=20 ; x=datos
Mu=Sigma=S=NULL ; Mu[1]=mu0 ; Sigma[1]=sigma0 ; S[1]=s0
#!-----
for (i in 2:10000) {
x[18:72] =rlogN(n=n.censu,MU=Mu[i-1],SIGMA=Sigma[i-1],S=S[i-1],minimo=66,center=80)
u.temporal=rpost.u(x=x,mu=Mu[i-1],sigma=Sigma[i-1],s=S[i-1])
Mu[i] =rpost.mu(sigma=Sigma[i-1],s=S[i-1],u=u.temporal,x=x)
Sigma[i] =rpost.sigma(mu=Mu[i],s=S[i-1],u=u.temporal,x=x)
S[i] =rpost.s(sigma=Sigma[i],mu=Mu[i],u=u.temporal,x=x) }
#!-----
parametros_logNG = data.frame( Mu=Mu[-(1:50)] , Sigma=Sigma[-(1:50)] , S=S[-(1:50)] )
#!-----
#!----- Análisis de los datos con censura considerando logN -----
#!-----
#! Definiendo los valores iniciales de los parametros
Mu=Sigma=NULL ; Mu[1]=mu0 ; Sigma[1]=sigma0
#!-----
for (i in 2:10000) {
x[18:72] =urlnorm(n=n.censu , meanlog=Mu[i-1] , sdlog=Sigma[i-1] , lb=66 , ub=Inf)
u.temporal=rpost.u.l(x=x,mu=Mu[i-1],sigma=Sigma[i-1])
Mu[i] =rpost.mu.l(sigma=Sigma[i-1],u=u.temporal,x=x)
Sigma[i] =rpost.sigma.l(mu=Mu[i],u=u.temporal,x=x) }
#!-----
parametros_logN = data.frame( Mu=Mu[-(1:50)] , Sigma=Sigma[-(1:50)] )
#!-----
Titulo = c('Mu', 'Sigma', 'S') ; par(mfrow=c(2,3))
for (i in 1:3) hdr.den(parametros_logNG[,i],xlab=Titulo[i],main='logNG',prob=90)
for (i in 1:2) hdr.den(parametros_logN[,i] ,xlab=Titulo[i],main='logN' ,prob=90)
#!-----

```