Suitable Algorithm Associated with a Parameterization for the Three-Parameter Log-Normal Distribution

Suitable Algorithm Associated with a Parameterization for the Three-parameter Lognormal Distribution

Yoshio Komori
Department of Systems Design and Informatics
Kyushu Institute of Technology
Iizuka 820-8502, Japan
komori@ces.kyutech.ac.jp

Keywords
Extended lognormal distribution, Local maximum likelihood, Bisection method
Mathematics Subject Classification
62A10, 62F10

Abstract

Associated with a parameterization for the three-parameter lognormal distribution, algorithm was proposed by Komori and Hirose (2004), which can find a local maximum likelihood (ML) estimate surely if it exists. Nevertheless, by Vera and Díaz-García (2008) it was shown that performance in finding a local ML estimate deteriorated by adopting the parameterization only and using other algorithm. In the present paper, it will be shown that Komori and Hirose's algorithm should be used for the parameterization. This work will also give MATLAB codes as a useful tool for the parameter estimation of the distribution.

1 Introduction

Parameter estimation methods for the three-parameter lognormal distribution have been studied by many researchers. Many of such studies are introduced in Vera and Díaz-García (2008).

The probability density function is given by

$$f(x;\alpha,\beta,\gamma) \stackrel{\text{def}}{=} \frac{1}{\sqrt{2\pi}(x-\alpha)\beta} \exp\left[-\frac{\{\ln((x-\alpha)/\gamma)\}^2}{2\beta^2}\right]$$
$$(x>\alpha,\ \beta>0,\ \gamma>0),$$

where x is a variable and α, β and γ are parameters. When x_i $(1 \le i \le n)$ are independent observations, we have the likelihood function: $L(\alpha,\beta,\gamma) \stackrel{\text{def}}{=} \prod_{i=1}^{n} f(x_i;\alpha,\beta,\gamma)$. For the observations, $x_1 > x_2 \ge \cdots \ge x_{n-1} > x_n$ will be assumed in the sequel without loss of generality.

Noting that a random variable $\ln(X-\alpha)$ is normally distributed, we can see that $L(\alpha,\beta,\gamma)$ achieves its maximum at a point $(\alpha,\hat{\beta}(\alpha),\hat{\gamma}(\alpha))$ for a given $\alpha < x_n$, where

$$\hat{\beta}(\alpha) \stackrel{\text{def}}{=} \sqrt{\frac{1}{n}\sum_{i=1}^{n}\{\ln(x_i-\alpha)-\ln\hat{\gamma}(\alpha)\}^2}, \qquad \hat{\gamma}(\alpha) \stackrel{\text{def}}{=} \exp\left[\frac{1}{n}\sum_{i=1}^{n}\ln(x_i-\alpha)\right].$$

It is, however, known that $L(\alpha,\hat{\beta}(\alpha),\hat{\gamma}(\alpha)) \to \infty$ as $\alpha \to x_n - 0$. Hence, instead of a maximum likelihood (ML) estimate in the usual meaning, a local ML estimate is considered, which makes $L(\alpha,\hat{\beta}(\alpha),\hat{\gamma}(\alpha))$ maximum under the condition $x_n - \alpha > \delta$ for a small $\delta > 0$ (Hill, 1963).

In order to find the local ML estimate of α, one possible way is to display $L(\alpha,\hat{\beta}(\alpha),\hat{\gamma}(\alpha))$, but it may have difficulties in such cases that the shape of $L(\alpha,\hat{\beta}(\alpha),\hat{\gamma}(\alpha))$ is complicated or the range of α to search through is too

wide (Cheng and Iles, 1990; Hill, 1963; Johnson et al., 1994). Also when an iterative solver such as Newton's method is used, similar or other difficulties can happen (Komori and Hirose, 2004). That is why many researchers tackled this estimation problem.

On the other hand, in order to avoid such difficulties Munro and Wixley (1970) have proposed a parameterization for the three-parameter estimation, where local ML estimates for α, β and γ are sought independently and simultaneously. In the sequel we will simply call a triplet of them a local ML estimate. Their parameterization is given by $\alpha \stackrel{\text{def}}{=} \mu - \sigma/\lambda$, $\beta \stackrel{\text{def}}{=} \lambda$ and $\gamma \stackrel{\text{def}}{=} \sigma/\lambda$ and it leads to

$$\begin{aligned} & f(x; \mu - \sigma/\lambda, \lambda, \sigma/\lambda) \\ & = \frac{1}{\sqrt{2\pi}\{\sigma + \lambda(x-\mu)\}} \exp\left[-\frac{\{\ln(\sigma + \lambda(x-\mu)) - \ln\sigma\}^2}{2\lambda^2} \right]. \end{aligned} \tag{1}$$

This can permit λ to be negative. We call it the extended lognormal distribution, in which $\lambda \neq 0$ and $\sigma > 0$. The parameterization is much helpful to improve the convergency of many iterative methods (Eastham et al., 1987; Hirose, 1997). It is, however, still probable that methods fail to find a local ML estimate. For example, see Subsection 3.3 in Komori and Hirose (2001). In addition, when they cannot find a local ML estimate, it is unclear whether it exists or not.

These two problems have been overcome with algorithm and another parameterization proposed by Komori and Hirose (2004). That is, the combination of them makes it possible to judge whether a local ML estimate exists or not, and to find it surely if it exists.

Vera and Díaz-García (2008) have proposed a global simulated annealing

(SA) optimization heuristic for the parameterizations mentioned above as well as Wingo's parameterization (Wingo, 1984). In their simulation, however, successful rates in finding a local ML estimate are low for Komori and Hirose's parameterization. This will be shown in the present paper.

The paper is organized as follows. In Section 2, we will briefly introduce our parameterization and algorithm. In Section 3, we will give simulation studies and discussion. Lastly, we will give concluding remarks.

2 Komori and Hirose's parameterization and algorithm

The substitution of $\tau \stackrel{\text{def}}{=} \sigma - \lambda\mu$ and $s \stackrel{\text{def}}{=} \ln\sigma$ into (1) yields

$$f(x; -\tau/\lambda, \lambda, \mathrm{e}^s/\lambda) = \frac{1}{\sqrt{2\pi}(\lambda x + \tau)} \exp\left[-\frac{\{\ln(\lambda x + \tau) - s\}^2}{2\lambda^2}\right] \qquad (\lambda \neq 0).$$

By arranging $\ln \bar{L}(\lambda, \tau, s) \stackrel{\text{def}}{=} \sum_{i=1}^n \ln f(x_i; -\tau/\lambda, \lambda, \mathrm{e}^s/\lambda)$, we obtain

$$\begin{aligned}\ln \bar{L}(\lambda, \tau, s) = &-\frac{n}{2\lambda^2}\left\{s - \frac{1}{n}\sum_{i=1}^n \ln(\lambda x_i + \tau)\right\}^2 - n\ln\sqrt{2\pi} \\ &+\frac{1}{2n\lambda^2}\left\{\sum_{i=1}^n \ln(\lambda x_i + \tau)\right\}^2 - \frac{1}{2\lambda^2}\sum_{i=1}^n \{\ln(\lambda x_i + \tau)\}^2 \\ &-\sum_{i=1}^n \ln(\lambda x_i + \tau).\end{aligned}$$

The first term has the maximum value 0 when $s = (1/n)\sum_{i=1}^n \ln(\lambda x_i + \tau)$. Hence, all we need to do is to maximize the following function:

$$F(\lambda, \tau) \stackrel{\text{def}}{=} \frac{1}{2n\lambda^2}\left\{\sum_{i=1}^n \ln(\lambda x_i + \tau)\right\}^2 - \frac{1}{2\lambda^2}\sum_{i=1}^n \{\ln(\lambda x_i + \tau)\}^2 - \sum_{i=1}^n \ln(\lambda x_i + \tau).$$

In order to achieve it, Komori and Hirose (2004) have proved the following theorem.

Theorem 2.1 *Let us define* $\tau_U^+(\lambda)$ *and* $\tau_U^-(\lambda)$ *by*

$$\tau_U^+(\lambda) \stackrel{\text{def}}{=} -\lambda x_n \left(\frac{1 - \bar{x}/x_n \mathrm{e}^{-\lambda^2}}{1 - \mathrm{e}^{-\lambda^2}} \right) \quad \textit{for } \lambda > 0,$$

$$\tau_U^-(\lambda) \stackrel{\text{def}}{=} -\lambda x_1 \left(\frac{1 - \bar{x}/x_1 \mathrm{e}^{-\lambda^2}}{1 - \mathrm{e}^{-\lambda^2}} \right) \quad \textit{for } \lambda < 0,$$

where $\bar{x}$ *stands for the arithmetic mean of data, that is,* $(1/n)\sum_{i=1}^n x_i$*. Then, the following statements hold.*

1)

$$\frac{\partial F}{\partial \tau}(\lambda, \tau_U^+(\lambda)) < 0 \quad \textit{for } \lambda > 0, \qquad \frac{\partial F}{\partial \tau}(\lambda, \tau_U^-(\lambda)) < 0 \quad \textit{for } \lambda < 0.$$

2)

$$\lim_{\tau \to -\lambda x_n + 0} \frac{\partial F}{\partial \tau}(\lambda, \tau) = +\infty \textit{ for } \lambda > 0, \qquad \lim_{\tau \to -\lambda x_1 + 0} \frac{\partial F}{\partial \tau}(\lambda, \tau) = +\infty \textit{ for } \lambda < 0.$$

3) If a point (λ, τ) *satisfies* $\dfrac{\partial^2 F}{\partial \tau^2}(\lambda, \tau) = 0$*, then* $\dfrac{\partial F}{\partial \tau}(\lambda, \tau) < 0$ *holds for the point.*

4) $\lim_{\lambda \to +\infty} F(\lambda, \tau_U^+(\lambda)) = +\infty$, $\lim_{\lambda \to -\infty} F(\lambda, \tau_U^-(\lambda)) = +\infty$.

5) $F_0(\tau) \stackrel{\text{def}}{=} \lim_{\lambda \to \pm 0} F(\lambda, \tau)$ *achieves the relative maximum when*

$$\tau = \tau^* \stackrel{\text{def}}{=} \frac{1}{n} \sqrt{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (x_i - x_j)^2}.$$

Based on the theorem, our algorithm is given and it provides the profile of $F(\lambda, \tau)$ for $\lambda > 0$ and $\lambda < 0$, respectively.

For $\lambda > 0$, the profile is sought by the following procedure. Here, τ, λ, τ_{min} and τ_{max} stands for variables, whereas ε_0, ε_1, ε_2, λ^+_{max} and $\triangle\lambda$ are preassigned positive constants.

1) Set $\tau = \tau^*$ and $\lambda = \varepsilon_0$ for a small preassigned constant $\varepsilon_0 > 0$.

2) For a sufficient large preassigned constant λ^+_{max}, if $\lambda > \lambda^+_{max}$, go to 8). Otherwise, set $\tau_{min} = -\lambda x_n$ and $\tau_{max} = \tau^+_U(\lambda)$.

3) Make sure that $\tau_{min} < \tau < \tau_{max}$ is satisfied. If it is not satisfied, go to 8).

4) If $\dfrac{\partial F}{\partial \tau}(\lambda, \tau) > 0$, set $\tau_{min} = \tau$. Otherwise, set $\tau_{max} = \tau$.

5) If $\dfrac{\partial F}{\partial \tau}(\lambda, (\tau_{min} + \tau_{max})/2) > 0$, set $\tau_{min} = (\tau_{min} + \tau_{max})/2$. Otherwise, set $\tau_{max} = (\tau_{min} + \tau_{max})/2$.

6) If $(\tau_{max} - \tau_{min})/|\tau_{max}| > \varepsilon_1$, go to 5). Otherwise, set $\tau = \tau_{max}$.

7) If $\left|\dfrac{\partial F}{\partial \tau}(\lambda, \tau)\right| < \varepsilon_2$, then record $(\lambda, \tau, F(\lambda, \tau))$, set $\lambda = \lambda + \triangle\lambda$ and go to 2).

8) End the procedure.

Step 1) comes from Statement 5) in the theorem. Step 2) comes from the fact that for each $\lambda > 0$ there exists the unique solution, say $\tau_0(\lambda)$, of $\frac{\partial F}{\partial \tau}(\lambda, \tau) = 0$ in $(-\lambda x_n, \tau^+_U(\lambda))$. In fact, from Statements 1) and 2), we can

see that there exists a solution $\tau_0(\lambda)$. Because

$$\frac{\partial F}{\partial \tau}(\lambda, \tau) < \frac{1}{\lambda^2}\left(\sum_{i=1}^{n} \frac{1}{\lambda x_i + \tau}\right)\left(\ln \frac{\lambda \bar{x} + \tau}{\lambda x_n + \tau} - \lambda^2\right)$$

holds for $\lambda > 0$ (Komori and Hirose, 2004), $\tau_0(\lambda)$ must be smaller than $\tau_U^+(\lambda)$ for which the expression in the right-hand side vanishes. Furthermore, because of Statements 1) and 3), $\frac{\partial F}{\partial \tau}(\lambda, \tau) < 0$ holds for $\tau \in (\tau_0(\lambda), \tau_U^+(\lambda))$. Thus, $\tau_0(\lambda)$ is unique. Step 3) confirms $\tau_{min} < \tau < \tau_{max}$ to start the bisection method with respect to τ. The violation of it means that the preassigned positive constant ε_0 or $\triangle\lambda$ is too large. Step 4) helps to shorten the interval to search through before the bisection method starts. Steps 5) and 6) indicate the bisection method with respect to τ. Using $\{(\lambda, F(\lambda, \tau))\}$ in the records in Step 7), we can plot the profile of $F(\lambda, \tau_0(\lambda))$ for $\lambda > 0$. From Statements 1), 4) and the things mentioned above, we can see that $\lim_{\lambda \to \pm\infty} F(\lambda, \tau_0(\lambda)) = \infty$.

On the other hand, when we seek the profile of $F(\lambda, \tau)$ for $\lambda < 0$, we replace 1), 2) and 7) with 1′), 2′) and 7′), respectively:

1′) Set $\tau = \tau^*$ and $\lambda = -\varepsilon_0$.

2′) For a sufficient small preassigned constant $\lambda_{min}^- < 0$, if $\lambda < \lambda_{min}^-$, go to 8). Otherwise, set $\tau_{min} = -\lambda x_1$ and $\tau_{max} = \tau_U^-(\lambda)$.

7′) If $\left|\frac{\partial F}{\partial \tau}(\lambda, \tau)\right| < \varepsilon_2$, then record $(\lambda, \tau, F(\lambda, \tau))$, set $\lambda = \lambda - \triangle\lambda$ and go to 2′).

For simulation studies in Section 3, some of the constants will be given as follows:

$$\varepsilon_0 = 0.05, \quad \varepsilon_1 = 10^{-14}, \quad \varepsilon_2 = 0.01, \quad \lambda_{max}^+ = 6, \quad \lambda_{min}^- = -6.$$

Here, note that the interval in which a $\tau_0(\lambda)$ exists becomes rapidly narrower as $|\lambda|$ becomes larger. When $\lambda = 6$, for example, the width of $(-\lambda x_n, \tau_U^+(\lambda))$ is $\frac{6}{e^{36}-1}(\bar{x} - x_n)$, which means the value of λ_{max}^+ is large enough.

If a local ML estimate exists and we set $\triangle\lambda$ at a sufficiently small positive value, then, from the plot data we can immediately get the extreme point of $F(\lambda, \tau)$ with high accuracy.

3 Simulation studies and discussion

In this section we give numerical results for data simulated by using the function that was used in Vera and Díaz-García (2008).

Table 1 shows the results given by Vera and Díaz-García (2008). When they obtained the results, the combination of the SA algorithm and our parameterization was used. In their simulation, μ and σ were fixed at 0 and 1, respectively. In addition, by communicating with one of them, the present author has known that

1. the function 'randn' with a method 'state' in MATLAB was used,

2. the initial state changed each time to generate 1000 sets of pseudo-random data,

3. the existence of a non-degenerate solution was manually checked for each data set.

These mean that it is almost impossible to reproduce the same data sets and results. Thus, whereas we reconstruct a similar setting for simulation by using the same function, differently from their way we use a constant initial

Table 1: Successful rate in finding a local ML estimate and existence rate of a local ML estimate in Vera and Díaz-García (2008)

		λ							
		.25	.50	.75	1.0	1.25	1.5	1.75	2.0
n	10	.989	.966	.942	.880	.736	.651	.604	.623
		(.962)	(.951)	(.903)	(.791)	(.655)	(.375)	(.169)	(.077)
	15	1.00	.997	.995	.991	.974	.926	.822	.686
		(.990)	(.994)	(.990)	(.975)	(.944)	(.891)	(.697)	(.379)

Note: A value in parentheses indicates an existence rate.

state for the function and seek for existence rates automatically by utilizing $F(\lambda, \tau_0(\lambda))$.

Before Monte Carlo experiments, let us see two examples as a single set of data. They are given in Table 2, which were generated by setting $(\lambda, n) = (2, 10)$ or $(2, 20)$ and the initial state at 0 for the function 'randn'. For these data sets, Figure 1 shows profiles of F. They indicate cases in which a local ML estimate exists or not. In the figure, the solid curves denote $F(\lambda, \tau_0(\lambda))$, whereas the dash curves denote $F(\lambda, \tau_U^+(\lambda))$ or $F(\lambda, \tau_U^-(\lambda))$. As in the figure, because we may regard $F(\lambda, \tau_U^+(\lambda))$ or $F(\lambda, \tau_U^-(\lambda))$ as $F(\lambda, \tau_0(\lambda)))$ for large $|\lambda|$ (Komori and Hirose, 2004), we can see that the interval $[-6, 6]$ of λ is large enough for the global search. For other empirical data sets, see Komori and Hirose (2004).

In a Monte Carlo experiment for a pair of n and λ, 16 batches of sets are considered and 1000 sets of pseudo-random data are simulated for each batch, where n pseudo-random data are generated for each set. Table 3 indicates the mean and the standard deviation of the existence rates of a local ML estimate,

Table 2: Examples of data set

data set (a)					
4.9124	4.8934	0.4622	0.3889	0.2090	0.1424
−0.0363	−0.2895	−0.4495	−0.4821		
data set (b)					
38.8787	4.9124	4.8934	3.7223	1.6349	0.4622
0.3889	0.2090	0.1424	0.1280	0.0629	−0.0363
−0.0871	−0.1194	−0.1558	−0.2895	−0.3458	−0.4054
−0.4495	−0.4821				

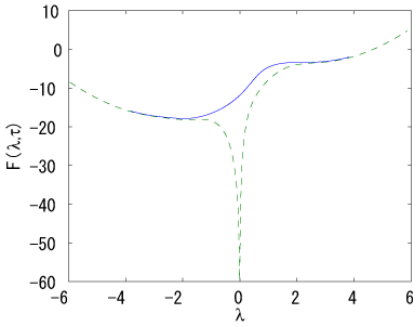
Table 3: Existence rate of a local ML estimate obtained by our algorithm

		λ							
		.25	.50	.75	1.0	1.25	1.5	1.75	2.0
n	10	.974	.955	.914	.831	.702	.539	.378	.248
		[.006]	[.008]	[.008]	[.012]	[.014]	[.015]	[.016]	[.012]
	15	1.00	.999	.995	.982	.947	.865	.717	.521
		[.001]	[.001]	[.002]	[.004]	(.008)	[.012]	[.019]	[.020]

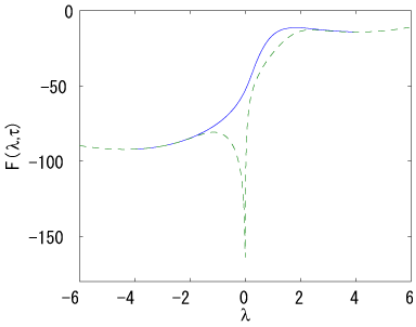
Note: A value in brackets indicates standard deviation.

which were obtained for 16 batches by our algorithm automatically. Because the standard deviation is small in all cases, we can see that the size of one batch (1000 sets) is large enough to obtain the existence rates appropriately.

In Table 1 the values of successful rates are not for 1000 sets, but for the limited sets in which the existence of a local ML estimate was checked manually (Vera and Díaz-García, 2008). In the case of $n = 10$ and $\lambda = 2.0$, for example, the rate 0.623 is for 77 sets. In order to see the genuine rates



(a) Case of no ML estimate



(b) Case of existence a local ML estimate

Figure 1: Profiles of $F(\lambda, \tau)$

Table 4: Rate in finding a local ML estimate for 1000 sets in Vera and Díaz-García (2008)

		λ							
		.25	.50	.75	1.0	1.25	1.5	1.75	2.0
n	10	.951	.919	.851	.696	.482	.244	.102	.048
	15	.990	.991	.985	.966	.920	.825	.573	.260

for 1000 sets, we need values of successful rate multiplied by existence rate in Table 1. These are shown in Table 4. Here, note that the values express how often their algorithm could find a local ML estimate automatically

By comparing Tables 3 and 4, we can see how much the combination of the SA algorithm and our parameterization worsens performance in finding estimates by itself, especially when $n = 10$ and $\lambda \geq 1$. In general, it becomes more difficult to find a local ML estimate as n becomes smaller and/or λ becomes larger. This fact, thus, shows that the combination of the SA algo-

rithm and our parameterization is not useful for such difficult situations.

4 Concluding remarks

Through the simulation studies, we have shown that the combination of our algorithm and our parameterization works better in finding a local ML estimate than that of the SA algorithm and our parameterization, especially when n is small and/or λ is large. Therefore, we strongly recommend using our algorithm for our parameterization. The following are also remarkable.

- Our algorithm always successfully finds a local ML estimate if it exists because the algorithm is based on the bisection method.
- In Vera and Díaz-García (2008) the combinations of the SA algorithm and the other parameterizations indicated similar performance to that of the SA algorithm and our parameterization.

MATLAB R2007b codes for the simulation and examples are obtainable from the following web page:

http://galois.ces.kyutech.ac.jp/~komori/software.html

The codes will serve as a useful tool when readers want to estimate the parameters in the distribution.

Acknowledgments

The author would like to thank the referee for his/her valuable comments which helped him to improve the earlier version of this paper.

References

[1] Cheng, R.C.H., Iles, T.C. (1990). Embedded models in three-parameter distributions and their estimation. *J. Royal Statist. Soc.* B 52:135–149.

[2] Eastham, J.F., LaRiccia, V.N., Schuenemeyer, J.H. (1987). Small sample properties of the maximum likelihood estimators for an alternative parameterization of the three-parameter lognormal distribution. *Comm. Statist.—Simulation Comput.* 16:871–884.

[3] Hill, B.M. (1963). The three-parameter lognormal distribution and Bayesian analysis of a point-source epidemic. *J. Amer. Statist. Assoc.* 58:72–84.

[4] Hirose, H. (1997). Maximun likelihood parameter estimation in the three-parameter log-normal distribution using the continuation method. *Comput. Statist. Data Anal.* 24:139–152.

[5] Johnson, N.L., Kotz, S., Balakrishnan, N. (1994). *Continuous Univariate Distributions*. New York: John Wiley & Sons.

[6] Komori, Y., Hirose, H. (2001). Easy estimation of by a new parameterization for the three-parameter lognormal distribution. Technical Report CSSE-16, Kyushu Institute of Technology. Preprint of Komori and Hirose (2004).

[7] Komori, Y., Hirose, H. (2004). Easy estimation of by a new parameterization for the three-parameter lognormal distribution. *J. Statist. Comput. Simulation* 74:63–74.

[8] Munro, A.H., Wixley, R.A.J. (1970). Estimation on order statistics of small samples from a three-parameter lognormal distribution. *J. Amer. Statist. Assoc.* 65:212–225.

[9] Vera, J.F., Díaz-Garcí, J.A. (2008). A global simulated annealing heuristic for the three-parameter lognormal maximum likelihood estimation. *Comput. Statist. Data Anal.* 52:5055–5065.

[10] Wingo, D.R. (1984). Fitting three-parameter lognormal models by numerical global optimization - an improved algorithm. *Comput. Statist. Data Anal.* 2:13–25.