Hacettepe Journal of
Mathematics & Statistics

Hacet. J. Math. Stat.
Volume 53 (2) (2024), 495 – 505
DOI : 10.15672/hujms.1335481

Research Article

A comparison of methods for estimating parameters of the stochastic Lomax process: through simulation study

Ahmed Nafidi, Ilyasse Makroz*

Department of Mathematics and Informatics, Hassan First University of Settat, National School of Applied Sciences, Berrchid, Morocco

Abstract

In this paper, we study the problem of parameter estimation of the stochastic Lomax diffusion process, this process was introduced in [A. Nafidi, I. Makroz, and R. Gutiérrez Sánchez, A stochastic lomax diffusion process: Statistical inference and application, Mathematics, 2021][14], and the authors suggested the method of simulated annealing to find the maximum likelihood estimators of this process. In this work, we propose alternative methods for finding the maximum likelihood estimators, namely Genetic algorithm and Nelder-Mead, we also investigate the use of Markov Chain Monte Carlo method to determine the model parameters. Finally, an example of application through the simulation of paths for the process is suggested. Then, a comparison is made between the application of three algorithms based on their accuracy and time of execution.

Mathematics Subject Classification (2020). 62F99, 60G99, 65Y99

Keywords. Lomax diffusion process, trend analysis, metaheuristic optimization algorithms, Markov Chain Monte Carlo, simulation, likelihood estimation

1. Introduction

The difficulty of using statistical inference to estimate the parameters of stochastic diffusion processes has attracted widespread attention in the last few years. In most cases, statistical inference is based on the maximum likelihood function. This latter is the product of the probability transition densities functions (ptdfs) of the process. There are few cases when ptdf of the process are known. Hence the exact likelihood estimation is usually infeasible. Consequently, the inference is based on approximating the likelihood function with continuous sampling. An extensive bibliography with more details on this subject can be found, for example, in [1,3,8,11]. Other works focused on other alternative methodologies, such as a nonparametric method, for example, in [2,4,9] and the references therein. Without overlooking the works based on the Bayesian methods by [6] and [19] and others.

In several situations, even though the ptdf of the process is known, it is not possible to obtain the explicit forms of the maximum likelihood estimators. Therefore, we must

*Corresponding Author.

Email addresses: ahmed.nafidi@uhp.ac.ma (A. Nafidi), i.makroz@uhp.ac.ma (I. Makroz)

Received: 31.07.2023; Accepted: 02.11.2023

solve nonlinear equations. Then, the issue boils down to a nonlinear optimization problem, whose solution cannot be obtained by classical numerical procedures. Hence various metaheuristic algorithms appear to overcome this problem. For a particular diffusion process, the algorithm often used is simulated annealing (SA). Various researches emerge in this topic, including those by [7] for the three-parameter stochastic lognormal diffusion model, [12] for Weibull diffusion process, and [13] for LundqvistKorf and [17] for the Gompertz diffusion process. Another algorithm is the variable neighborhood search for the stochastic Richards diffusion process [18].

For a Lomax diffusion process, we used the SA algorithm in the case of [14]. In this work, we suggest three different algorithms to those mentioned before, to get the likelihood estimators for the parameters of this process, and who are Genetic algorithm (GA) [20] and Nelder-Mead (NM) [10], and Markov Chain Monte Carlo (MCMC) [15].

We organized this paper as follows. In Section 2, we gave a brief overview of the stochastic Lomax diffusion process. Then we formulated the problem of the likelihood estimation of the parameters in this process using the maximum likelihood method. In Section 3, we proposed a computational methodology that represents the necessary tools to solve the problem of parameter estimation. In Section 4, we gave a simulation study, where we treat the problem. Then, we illustrated the results obtained. Finally, we gave a conclusion.

2. Problem formulation

2.1. A brief overview of the Lomax stochastic diffusion process

The Lomax stochastic diffusion process is the one-dimensional inhomogeneous diffusion process $\{x(t) : t \in [t_1, T], t_1 \geq 0\}$ taking values on $(0, \infty)$ and satisfying the following stochastic differential equation (SDE):

$$dx(t) = -\frac{\alpha}{t+\beta} x(t)dt + \sigma x(t)dw(t) \tag{2.1}$$

where $\sigma > 0$, $\beta > 0$ and α are real parameters. $w(t)$ is the one-dimensional standard Wiener process and x_{t_1} is fixed in $\mathbb{R}_+^*$.

The analytical expression of the process is given by (see [14]) :

$$x(t) = x_s \left(\frac{s+\beta}{t+\beta}\right)^{\alpha} \exp\left[-\frac{\sigma^2}{2}(t-s) + \sigma\left(w(t) - w(s)\right)\right] \tag{2.2}$$

The transition density of the process is given by (see [14]) :

$$f(x, t \mid x_s, s) = \frac{1}{x\sqrt{2\pi(t-s)\sigma^2}} \exp\left(-\frac{\left[log\left(\frac{x}{x_s}\right) + \alpha log\left(\frac{t+\beta}{s+\beta}\right) + \frac{\sigma^2}{2}(t-s)\right]^2}{2\sigma^2(t-s)}\right) \tag{2.3}$$

The conditional trend function of the process is :

$$E(x(t) \mid x(s) = x_s) = x_s \left(\frac{s+\beta}{t+\beta}\right)^{\alpha} \tag{2.4}$$

In addition, taking into account the initial condition $P(x(t_1) = x_{t_1}) = 1$, the trend function of the process is :

$$E(x(t)) = x_{t_1} \left(\frac{t_1+\beta}{t+\beta}\right)^{\alpha} \tag{2.5}$$

2.2. Problem of parameter estimation

In the present study, with discrete sampling, we estimate the parameters α, β and σ of the process by applying maximum likelihood estimation methodology. Let us consider a discrete sampling of the process $x_1, x_2, \quad x_n$ for time $t_1, t_2, \ldots, n$, we denote $x(t_i) = x_i$ moreover we set $t_i - t_{i-1} = h$, and assume the initial condition $P(x(t_1) = x_1) = 1$, then the associated likelihood function can be obtained by the following expression:

$$L(x_1, x_2, \ldots, x_n) = \prod_{i=2}^{n} f(x_i, t_i \mid x_{i-1}, t_{i-1})$$

By using the expression of the transition density function in the Eq. 2.3, the log-likelihood function leads us:

$$\begin{aligned} log(L(x_1, x_2, \ldots, x_n)) &= \sum_{i=2}^{n} -log(x_i) - \frac{1}{2}log(2\pi h) - \frac{1}{2}log(\sigma^2) \\ &\qquad - \frac{1}{2\sigma^2 h}\left[log\left(\frac{x_i}{x_{i-1}}\right) + \alpha log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right) + \frac{\sigma^2}{2}h\right]^2 \\ &= -\frac{n-1}{2}log(\sigma^2) - \frac{n-1}{2}log(2\pi h) \\ &\qquad - \sum_{i=2}^{n}\left[log(x_i) + \frac{1}{2\sigma^2 h}\left(C_{i,\alpha,\beta} + \frac{\sigma^2}{2}h\right)^2\right] \quad , \end{aligned}$$

where $C_{i,\alpha,\beta} = log\left(\frac{x_i}{x_{i-1}}\right) + \alpha log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right)$.

By applying the MLE principal that is taking the derivatives of the log-likelihood function with respect to the parameters α, β and σ and equaling to zero those derivatives , we obtain the following likelihood equations

$$\begin{aligned} \frac{\partial log(L(x_1, x_2, \ldots, x_n))}{\partial\alpha} &= -\sum_{i=2}^{n}\frac{1}{\sigma^2 h}\left(C_{i,\alpha,\beta} + \frac{\sigma^2}{2}h\right)log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right) = 0 \\ \frac{\partial log(L(x_1, x_2, \ldots, x_n))}{\partial\beta} &= -\sum_{i=2}^{n}\frac{1}{\sigma^2 h}\left(C_{i,\alpha,\beta} + \frac{\sigma^2}{2}h\right)\left(\frac{\alpha}{t_i+\beta} - \frac{\alpha}{t_{i-1}+\beta}\right) = 0 \\ \frac{\partial log(L(x_1, x_2, \ldots, x_n))}{\partial\sigma^2} &= -\frac{n-1}{2\sigma^2} - \frac{(n-1)h}{8} - \sum_{i=2}^{n}\frac{C_{i,\alpha,\beta}^2}{2\sigma^4 h} = 0 \end{aligned}$$

After various operations which are not shown the maximum likelihood estimator of σ^2 is given by:

$$\hat{\sigma}^2 = \frac{2}{h}\left(\left(1 + \frac{1}{n-1}\sum_{i=2}^{n} C_{i,\alpha,\beta}^2\right)^{1/2} - 1\right) \tag{2.6}$$

Therefore, the maximum likelihood estimators of α and β are the solutions to the following non-linear equations

$$\sum_{i=2}^{n}\left(C_{i,\alpha,\beta} + \left(1 + \frac{1}{n-1}\sum_{i=2}^{n} C_{i,\alpha,\beta}^2\right)^{1/2} - 1\right)log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right) = 0 \tag{2.7}$$

$$\sum_{i=2}^{n}\frac{C_{i,\alpha,\beta} + \left(1 + \frac{1}{n-1}\sum_{i=2}^{n} C_{i,\alpha,\beta}^2\right)^{1/2} - 1}{(t_i+\beta)(t_{i-1}+\beta)} = 0 \tag{2.8}$$

Obviously, this is a set of non-linear equations whose solutions may be difficult to find. To address this problem we use numerical resolution methods. In [14], the simulated annealing algorithm is used to solve this non-linear likelihood function. It is worth noting here that a closed-form expression of $\hat{\alpha}$ can be obtained, in fact, let :

$$a = \sum_{i=2}^{n} log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right) \quad , \quad b = \sum_{i=2}^{n} log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right) log\left(\frac{x_i}{x_{i-1}}\right)$$

$$c = \sum_{i=2}^{n}\left[log\left(\frac{x_i}{x_{i-1}}\right)\right]^2 \quad , \quad d = \sum_{i=2}^{n}\left[log\left(\frac{t_i+\beta}{t_{i-1}+\beta}\right)\right]^2$$

Then, we can show that $\hat{\alpha}$ is given by:

$$\hat{\alpha} = \frac{-\left[2\frac{bd}{a^2} - 2\frac{d}{a} - \frac{b}{n-1}\right] - \sqrt{\left(\frac{2d}{a} - \frac{2b}{n-1}\right)^2 + \frac{4d^2c}{a^2(n-1)}\left(1 - \frac{a^2}{(n-1)d}\right)}}{2\left[\frac{d^2}{a^2} - \frac{d}{n-1}\right]} \tag{2.9}$$

3. Computational methodology

Obviously, this is a set of non-linear equations whose solutions may be difficult to find. As we have already mentioned, to address this problem we suggest three numerical algorithms, that can be presented as follows

3.1. Nelder-Mead algorithm (downhill simplex method)

Nelder-Mead algorithm is a simplex algorithm which uses a simplex of $n+1$ points for n-dimensional vectors x. The algorithm makes a simplex around the initial guess x_0. At every iteration, it proceeds to move this simplex, one vertex at a time, towards an optimal solution. During each step, it calculates the function value of one or more predetermined combinations of the points in the current simplex, and chooses one that shifts it towards a better region of the domain. After few iterations the simplex starts to shrink inwardly, thus yielding an optimal solution [10].

- let us denote the current simplex with $\{x_j\}_{j=1}^{n+1}$.
- the algorithm orders the points in the simplex according to their function value in increasing order. At each iteration it discards the worst point x_{n+1} and accept another point into the simplex.
- calculate the function value of the reflected point given by:

$$r = \frac{2}{n}\sum_{i=1}^{n} x_i - x_{n+1}$$

- if $f(x_1) \leq f(r) \leq f(x_n)$ accept r in the simplex.
- if $f(r) < f(x_1)$, calculate the function value of the expansion point whose expression is given by :

$$c = \frac{3}{n}\sum_{i=1}^{n} x_i - 2 \times x_{n+1}$$

- if $f(c) < f(r)$ accept c in the simplex. Otherwise accept r.
- if $f(r) \geq f(x_n)$.

3.2. Genetic algorithm

Genetic algorithm is a search heuristic that is inspired by Charles Darwin's theory of natural selection, where the fittest individuals are selected for reproduction in order to produce offspring of the next generation [5].

The algorithm begins by creating a random initial population, it then creates a sequence of new populations. At each step the algorithm uses the current generation to create the next population, using mutation , which is achieved by making some random change to a single parent, or crossover , which is achieved by combining the vector entries of a pair of parents. In addition the next generation contains the members of the current generation that have lowest fitness values, those members are called elite. The algorithm converges after few generations.

- scores each member of the current population by computing its fitness value. These values are called the raw fitness scores.
- scales the raw fitness scores to convert them into a more usable range of values. These scaled values are called expectation values.
- select members, called parents, based based on their expectation.
- some of the individuals in the current population that have lower fitness are chosen as elite. These elite individuals are passed to the next population.
- produces children from parents. Children are produced either by making random changes to a single parent (mutation) or by combining the vector entries of a pair parents (crossover).
- replaces the current population with children to form the next generation.

3.3. Markov Chain Monte Carlo algorithm

Direct calculation of a desired quantity from a model of interest, such as expected probability, density, or other aspects of the probability distribution, is difficult for most probabilistic models. Instead, different techniques must be used to estimate the expected probability or density.

Drawing independent samples from the probability distribution and repeating this process multiple times to estimate the target quantity is a common solution. This is referred to as Monte Carlo sampling, which is a technique for randomly sampling a probability distribution and approximating a desired quantity. Monte Carlo sampling has the drawback of not working well in high-dimensions, on the other hands this technique assumes that each random sample drawn from the target distribution is independent and can be independently drawn, which is not always the case. One popular solution to sampling probability distributions in high-dimensions is to use Markov Chain MonteCarlo (MCMC), which is a method for obtaining information about distributions, especially for estimating posterior distributions in Bayesian inference. MCMC rely on the idea of using the information provided by the observed data to update prior beliefs about a set of parameters, that is ,by sampling from the a posterior distribution, we can estimate any statistic of this distribution. In order to obtain samples from the posterior distribution one popular MCMC technique is Gibbs sampling. Suppose we want to estimate some random variables denoted by X_1, X_2, X_3 , we denote by x_i^j the value of the variable x_i at iteration j, we start by setting those variables to their initial values $x_1^{(0)}, x_2^{(0)}, x_3^{(0)}$ and then at each iteration the values of x_i are updated as follows until convergence:

- Sample $x_1^{(j)} \sim P(X_1 = x_1 \mid X_2 = x_2^{(j-1)}, X_3 = x_3^{(j-1)})$
- Sample $x_2^{(j)} \sim P(X_2 = x_2 \mid X_1 = x_1^{(j)}, X_3 = x_3^{(j-1)})$
- Sample $x_3^{(j)} \sim P(X_3 = x_3 \mid X_1 = x_1^{(j)}, X_2 = x_2^{(j)})$

Where $P(X \mid Y)$ denotes the conditional probability of X given Y.

In this work we use the publicly available package in R, rjags (Just Another Gibbs Sampler) [16] which contains several tools for assessing the convergence of the MCMC method.

3.4. Estimated trend functions

The trend analysis in this study is based on the trend functions of the model, this requires analysis to obtain fits and forecasts to real data by means of the estimated trend function (ETF) and the estimated conditional trend function (ECTF). The latter functions are obtained.

$$\hat{E}(x(t) \mid x(s) = x_s) = x_s \left(\frac{s + \hat{\beta}}{t + \hat{\beta}} \right)^{\hat{\alpha}} \tag{3.1}$$

With the initial condition $P(x(t_1) = x_{t_1}) = 1$, the estimated trend function (ETF) of the process leads us:

$$\hat{E}(x(t)) = x_{t_1} \left(\frac{t_1 + \hat{\beta}}{t + \hat{\beta}} \right)^{\hat{\alpha}} \tag{3.2}$$

4. Simulation study

The trajectory of the model can be obtained by simulating the exact solution of SDE Eq.2.1 given in Eq.2.2. From this explicit solution, the simulated trajectories of the process are obtained from the following equidistant discretisation of the interval $[s, T]$ with $t_i = t_{i-1} + (i-1)h$ for $i = 2, ..., N$, and taking into account that the random variable in the latter expression $\sigma\,(w(t) - w(t_1))$ is distributed as a one-dimensional normal distribution $\mathcal{N}\left(0, \sigma^2\,(t - t_1)\right)$. Let $(t_1 = s)$ and assume a discretisation step $h = \frac{T-s}{N}$ where N denotes the size of the sample. The process was simulated, with s = 0, T = 1000 , $x_s = 100$ and $N = 50, 100, 200, 1000$ respectively. We reserve the last five values observed for comparison with the corresponding prediction by each model.
Tables 1, 2, 3 summarize the results obtained, by estimating the parameters using NM, GA and MCMC respectively. $\hat{\alpha}$, $\hat{\beta}$ and $\hat{\sigma}$ denotes the estimated values for α, β and σ respectively, $std(\alpha)$, $std(\beta)$ and $std(\sigma)$ denotes the standard deviation of the estimated value for each parameter, and time is the time needed by the algorithm to estimate the parameters.

Table 1. NM estimates and time of execution.

N	Time (s)	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}$	$std(\alpha)$	$std(\beta)$	$std(\sigma)$
50	0.150	1.521339	78.605367	0.010663	0.771321	0.020752	0.633648
100	0.160	1.539675	81.807833	0.010183	0.149761	0.099701	0.044959
200	0.209	1.512123	84.010827	0.010090	0.019806	0.076053	0.016911
1000	0.316	1.507265	90.702883	0.010050	0.004417	0.034268	0.047670

Table 2. GA estimates and time of execution.

N	Time (s)	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}$	$std(\alpha)$	$std(\beta)$	$std(\sigma)$
50	2.368	1.436254	76.999654	0.0106290	0.612526	0.721755	0.291876
100	2.980	1.459240	83.586855	0.010212	0.458887	0.357288	0.271272
200	4.198	1.474833	86.356472	0.010034	0.028434	0.074668	0.134827
1000	14.292	1.499980	89.747662	0.010005	0.044183	0.043401	0.061777

Table 3. MCMC estimates and time of execution.

N	Time (s)	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\sigma}$	$std(\alpha)$	$std(\beta)$	$std(\sigma)$
50	1.7003	1.375296	64.982110	0.000655	0.513138	0.650397	0.601039
100	3.5665	1.39675	77.903970	0.000905	0.402612	0.260824	0.454324
200	8.5425	1.432963	92.247471	0.010167	0.063847	0.018092	0.060140
1000	39.19377	1.494277	90.767567	0.010261	0.011398	0.082868	0.004690

Figures 1, 2, 3, 4 illustrate the ETFs obtained using NM, GA and MCMC respectively for each N along with simulated data.

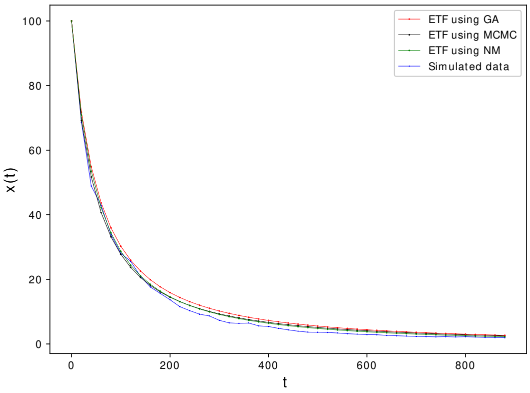

Figure 1. Simulated data vs trend functions for N = 50

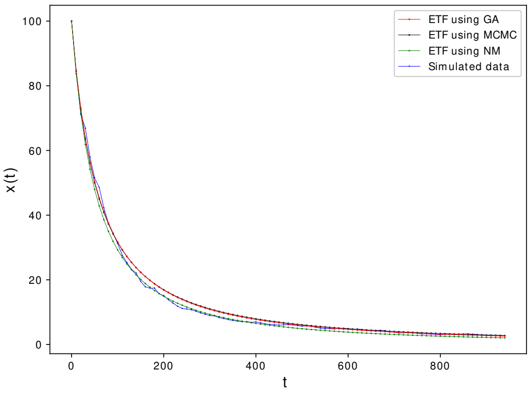

Figure 2. Simulated data vs trend functions for N = 100

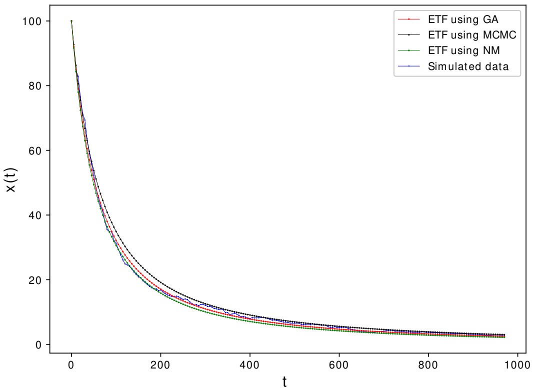

Figure 3. Simulated data vs trend functions for N = 200

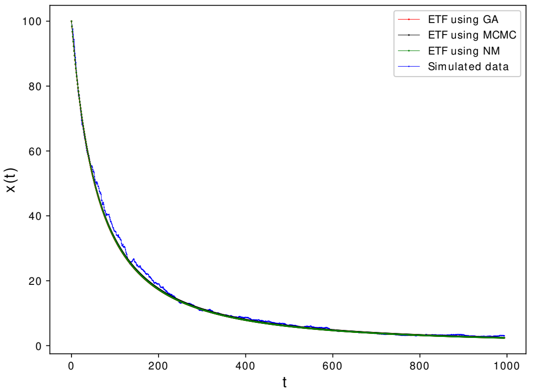

Figure 4. Simulated data vs trend functions for N = 1000

Figures 5 and 6 are given as an example of the ECTFs obtained using NM, GA and MCMC respectively for $N = 50$ and $N = 1000$ along with simulated data.

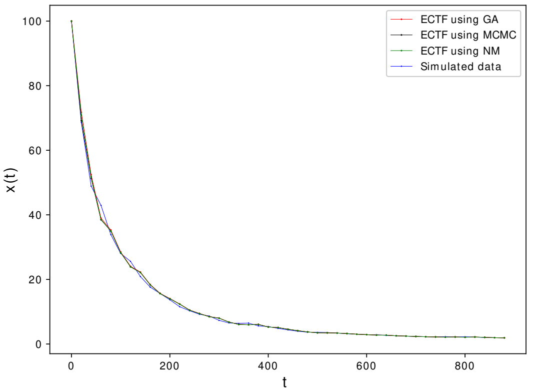

Figure 5. Simulated data vs conditional trend functions for N = 50

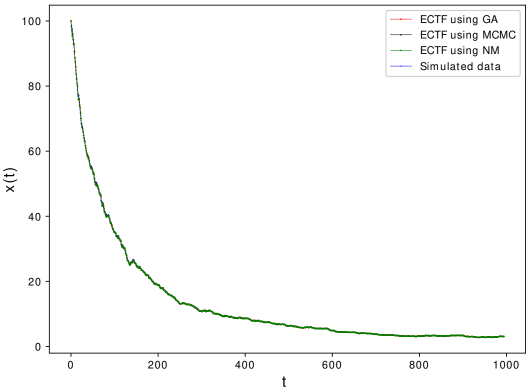

Figure 6. Simulated data vs conditional trend functions for N = 1000

In order to assess the accuracy of the model we compared the results obtained by each model with the actual last five values of the simulated data. The results for sample sizes $N = 50$ and $N = 1000$ are summarized in Tables 4 and 5 respectively.

Table 4. Forecasts for N = 50 .

Simulated data	ETF_{MCMC}	ETF_{GA}	ETF_{NM}	$ECTF_{MCMC}$	$ECTF_{GA}$	$ECTF_{NM}$
2.805834	2.446418	2.601532	2.157235	2.880933	2.878333	2.873422
2.665839	2.378362	2.526908	2.091850	2.727780	2.725349	2.720790
2.437297	2.313512	2.455843	2.029685	2.593150	2.590867	2.586617
2.451054	2.251656	2.388102	1.970523	2.372132	2.370069	2.366254
2.204117	2.136172	2.261754	1.860430	2.325344	2.321375	2.314114

Table 5. Forecasts for N = 1000.

Simulated data	ETF_{MCMC}	ETF_{GA}	ETF_{NM}	$ECTF_{MCMC}$	$ECTF_{GA}$	$ECTF_{NM}$
1.435982	2.451653	2.379925	2.371553	1.431848	1.431839	1.431831
1.439926	2.448283	2.376638	2.368264	1.434008	1.433999	1.433991
1.435080	2.444920	2.373358	2.364983	1.437948	1.437939	1.437931
1.434452	2.441566	2.370086	2.361710	1.433111	1.433102	1.433094
1.416012	2.434879	2.363565	2.355186	1.430523	1.430505	1.430489

Furthermore we calculated the mean absolute percentage error (MAPE) and the symmetric mean absolute percentage error (SMAPE) for the model obtained by each method and for different N. The results are shown in Tables 6 and 7.

Table 6. MAPE values for different values of N .

	N = 50	N = 100	N = 200	N = 1000
MCMC	7.977993	16.961279	10.866985	70.657774
GA	3.687408	9.312534	8.516675	65.662444
NM	19.313886	13.533037	19.849990	65.077682

Table 7. SMAPE values for different values of N .

	N = 50	N = 100	N = 200	N = 1000
MCMC	8.381328	15.569057	10.253368	52.210942
GA	3.769672	8.830660	8.945266	49.432045
NM	21.432644	14.579248	22.083891	49.099897

5. Summary and conclusions

In this work we provided three methods to address the problem of parameter estimation of the stochastic Lomax diffusion process. In terms of speed, we conclude that NM algorithm is by far superior to GA and MCMC, however NM method require a lucky guess of the initial value, as the convergence is highly dependent on that initial value. In terms of complexity, we can see clearly that MCMC scales badly with large values of N, In particular each time we double the sample size N execution time doubles. In addition the values obtained for the MAPE and SMAPE suggests that forecasted values for the model obtained using GA method gives the best accuracy when the sample size is less than $N = 200$, and that the NM is slightly better for $N = 1000$, however further investigations are needed to determine if this holds for larger values of N.

References

[1] Y. Aït-Sahalia, *Maximum likelihood estimation of discretely sampled diffusions: a closed-form approximation approach*, Econometrica **70** (1), 223-262, 2002.

[2] F.M. Bandi and P.C. Phillips, *Fully nonparametric estimation of scalar diffusion models*, Econometrica **71** (1), 241-283, 2003.

[3] A. Beskos, O. Papaspiliopoulos, G.O. Roberts and P. Fearnhead, *Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discussion)*, J R Stat Soc Series B Stat Methodol **68** (3), 33-382, 2006.

[4] J. Fan, *A selective overview of nonparametric methods in financial econometrics*, Stat Sci **20** (4), 317-337, 2005.

[5] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1989.

[6] A. Golightly and D.J. Wilkinson, *Bayesian inference for nonlinear multivariate diffusion models observed with error*, Comput Stat Data Anal **52** (3), 1674-1693, 2008.

[7] R. Gutiérrez, R. Gutiérrez-Sánchez, A. Nafidi and E. Ramos, *Three-parameter stochastic lognormal diffusion model: statistical computation and simulating annealingapplication to real case*, J. Stat. Comput. Simul. **79** (1), 25-38, 2009.

[8] A.S. Hurn and K. Lindsay, *Estimating the parameters of stochastic differential equations*, Math Comput Simul **48** (4-6), 373-384, 1999.

[9] G.J. Jiang and J.L. Knight, *A nonparametric approach to the estimation of diffusion processes, with an application to a short-term interest rate model*, Econ Theory **13** (5), 615-645, 1997.

[10] J.C. Lagarias, J.A. Reeds, M.H. Wright and P.E. Wright, *Convergence properties of the neldermead simplex method in low dimensions*, SIAM J. Optim. **9** (1), 112-147, 1998.

[11] C. Li, *Maximum-likelihood estimation for diffusion processes via closed-form density expansions*, Ann. Stat. **41** (3), 1350-1380, 2013.

[12] A. Nafidi, M. Bahij, B. Achchab and R. Gutiérrez-Sánchez, *The stochastic Weibull diffusion process: Computational aspects and simulation*, Appl. Math. Comput. **348**, 575-587, 2019.

[13] A. Nafidi and A. El Azri, *A stochastic diffusion process based on the lundqvistkorf growth: Computational aspects and simulation*, Math Comput Simul **182**, 25-38, 2021.

[14] A. Nafidi, I. Makroz and R. Gutiérrez Sánchez, *A stochastic lomax diffusion process: Statistical inference and application*, Mathematics **9** (1), 100, 2021.

[15] M. Plummer, N. Best, K. Cowles and K. Vines,*Coda: convergence diagnosis and output analysis for mcmc*, R News **6** (1), 711, 2006.

[16] M. Plummer, A. Stukalov, M. Denwood and M.M. Plummer, R Package "rjags", Vienna, Austria, 2016.

[17] P. Román-Román, D. Romero, M. Rubio and F. Torres-Ruiz, *Estimating the parameters of a gompertz-type diffusion process by means of simulated annealing*, Appl. Math. Comput. **218** (9), 5121-5131, 2012.

[18] P. Román-Román and F. Torres-Ruiz, *A stochastic model related to the richards-type growth curve. estimation by means of simulated annealing and variable neighborhood search*, Appl. Math. Comput. **266**, 579-598, 2015.

[19] O. Stramer and M. Bognar, *Bayesian inference for irreducible diffusion processes using the pseudo-marginal approach*, Bayesian Anal. **6** (2), 231-258, 2011.

[20] D. Whitley, *A genetic algorithm tutorial*, Stat Comput **4** (2), 65-85, 1994.