Maximum Likelihood Parameter Estimation for the Exponentially-Modified Logistic Distribution Based on Particle Swarm Optimization

ADI OMAIA FAOURI
Department of Statistics
University of Ondokuz Mayis
Samsun, TURKEY
ORCID id: 0000-0003-4499-1240

PELIN KASAP
Department of Statistics
University of Ondokuz Mayis
Samsun, TURKEY
ORCID id: 0000-0002-1106-710X

Abstract: Exponentially-modified logistic distribution is a new flexible modified distribution. It is regarded as a strong competitor for widely used classical symmetrical and non-symmetrical distributions such as normal, logistic, lognormal, and log-logistic. In this study, the unknown parameters of the distribution have been estimated using the maximum likelihood method. Meta-heuristic algorithms have been used to solve the nonlinear equations of this method. The algorithms used in this study are the Sine Cosine and the Particle Swarm Optimization Algorithms. The efficiencies of maximum likelihood estimates for these algorithms are compared via a Monte-Carlo simulation study. It has been seen that the likelihood estimates for the location α and scale β parameters of the exponentially-modified logistic distribution developed with the Particle Swarm Optimization algorithm are more efficient than the Sine Cosine algorithm.

Keywords**:** *Maximum Likelihood, Exponentially-Modified Logistic Distribution, Particle Swarm Optimization, Sine Cosine, Monte Carlo Simulation.*

Received: July 19, 2023. Revised: February 25, 2024. Accepted: March 11, 2024. Published: May 8, 2024.

1. Introduction

The logistic distribution is recognized to be similar to the normal distribution. Both normal and logistic distribution are members of the location-scale family. However, the logistic distribution has heavier tails than the normal distribution [1]. The importance of logistic distribution is that it has the ability to be used in many scientific areas like physical science, finance, and many applications in reliability and survival analysis, besides the major utility of its distribution function in logistic regression, logit models, and neural networks [2]. For more properties and details about logistic distribution, see [3, 4]. On the other hand, the exponential distribution, which is concerned with the measurement of time needed until the occurrence of a specific event [2], was priviously the basis of reliability and life expectancy evaluation for many lifetime data distributions. For more details about the exponential distribution, see [5]. However, in further research in reliability theory, it was revealed that modeling by the exponential distribution is only useful for the first approximation and can't be enough for a lot of problems in many cases. In the last two decades, in order to give better-fitting solutions and increase fit effectiveness for model functions that have no closed-form and require a numerical method in lifetime data analysis, many generalizations and modified extensions of the exponential distribution have been suggested to become more flexible and capable for modeling real-world data, especially when the characteristics of classical distributions are limited

and, practically, they cannot provide a good fit in many situations [2,6,7,8,9]. Various exponentiated distributions have been generalized; for instance, in 1998, the exponentiated exponential distribution was introduced by Gupta and Kundu [10] and is considered the first extension of the exponential distribution family. The exponentiated Weibull distribution was extended in 2006 by Pal et al. [11], the exponentiated Gamma distribution was generalized in 2007 by Nadarajah and Gupta [12], another extension was proposed in 2011 by Nadarajah and Haghighi [13], the exponentiated log-logistic distribution with two parameters was extended in 2019 by Chaudhary [14], and many other well-known distributions have recently been extended and modified by the exponential distribution family. Reyes et al. (2018) have presented the two-parameter exponentially-modified logistic (EMLOG) distribution [15], which used the same methodology as Grushka, who constructed the exponentially-modified Gaussian (EMG) distribution by combining the normal and exponential distributions [16]. So, the only difference is that the logistic distribution is used in place of the normal distribution. One of the advantages of the combined new probability distributions obtained is that they generally have longer tails than the originals without combining distributions, thus giving rise to better fits for empirical frequency distributions [15, 16].

In general, there are many different statistical methodologies for estimating the parameters of any distribution, such as the Maximum Likelihood (ML) method, the Method of Moments (MOM), the Least Squares (LS) method, and so forth. ML is the most widely used methodology among all statistical methods because of its high performance and well-known asymptotic properties for parameter estimators such as bias, consistency, efficiency, and so forth in comparison with any other method [17]. The basic principle of the ML methodology is to find the estimator values for the parameters of concern that maximize the likelihood function of the model, but in most cases, an explicit solution is not available because of the presence of nonlinear functions. Therefore, iterative algorithms can be used to maximize the likelihood function [18].

The aim of this study is to obtain the maximum likelihood estimation of the location α and scale β parameters of the EMLOG distribution. However, for the two-parameter EMLOG distribution, explicit solutions to the likelihood equations do not exist, and this is the main problem highlighted by this study. In this study, to solve this problem, we have used the Sine Cosine (SC) algorithm and the Particle Swarm Optimization (PSO) algorithm. To the best of our knowledge, this is the first study to obtain the ML estimators in the context of parameter estimation for the EMLOG based on PSO and SC. However, it should be noted that this problem can also be tackled by using other meta-heuristic algorithms such as grey wolf optimization, whale optimization, etc. The SC and PSO algorithms have been compared by conducting a Monte Carlo simulation study.

The rest of the article is organized as follows: In Section (2), the two-parameter EMLOG distribution has been introduced. In Section (3), the SCA and PSO algorithms that were applied for the ML estimation method in this study are introduced. In Section (4), the efficiencies of the parameter estimators are compared via a Monte-Carlo simulation study. In the last section, the conclusions are given.

2. Two-Parameter Exponentially-Modified Logistic Distribution

The generalization of this distribution is made by the combination of a logistic distribution with parameters for location α and scale β and the same scale parameter for the exponential distribution. As a result, the two-parameter exponentially modified logistic distribution is produced, with the left tail influenced by exponential distribution and the right tail distributed by logistic distribution. For the sake of simplicity, in the remaining portion of the study, this distribution will be denoted by EMLOG distribution.

If X is a random variable with a parameterized location α and scale β that follows an EMLOG distribution, $X\sim EMLOG\ (\alpha,\ \beta)$, then the probability density function (pdf) of X is:

$$f(x;\alpha,\beta)=\frac{1}{\beta\left(e^{\left(\frac{x-\alpha}{\beta}\right)}+1\right)}\left[\left(e^{-\left(\frac{x-\alpha}{\beta}\right)}+1\right)\log\left(e^{\left(\frac{x-\alpha}{\beta}\right)}+1\right)-1\right],\ x\in R,\quad \alpha\in R,\ \beta>0 \qquad (1)$$

The ML estimation for the parameters of interest are the values in the parameter space that maximize the likelihood function; for calculation simplicity, the likelihood function's logarithm is used. In this study, the log-likelihood ($ln\ L$) function is given below for estimating the unknown parameters α and β for the EMLOG distribution.

$$\ln L(\alpha,\beta)=-n\log(\beta)-\sum_{i=1}^{n}\log(e^{z_i}+1)+\sum_{i=1}^{n}\log[(e^{-z_i}+1)\log(e^{z_i}+1)-1] \qquad (2)$$

where $z_i=(x_i-\alpha)/\beta$. In order to estimate the likelihood parameters for the $ln\ L$ function for the EMLOG distribution, the partial derivatives with respect to the parameters of interest are taken and equated to zero. The likelihood equations are given as follows:

$$\frac{\partial \ln L(\alpha,\beta)}{\partial\alpha}=\sum_{i=1}^{n}\frac{e^{z_i}}{e^{z_i}+1}+\sum_{i=1}^{n}\frac{e^{-z_i}\log(e^{z_i}+1)-1}{(e^{-z_i}+1)\log(e^{z_i}+1)-1}=0 \qquad (3)$$

and

$$\frac{\partial \ln L(\alpha,\beta)}{\partial\beta}=-n\beta+\sum_{i=1}^{n}\frac{(x_i-\alpha)e^{z_i}}{e^{z_i}+1}+\sum_{i=1}^{n}\frac{(x_i-\alpha)(e^{z_i}\log(e^{z_i}+1)-1)}{(e^{-z_i}+1)\log(e^{z_i}+1)-1}=0 \qquad (4)$$

As we can see from equations (3) and (4), they have nonlinear functions, and an explicit solution for the likelihood equations cannot be obtained. Therefore, iterative algorithms are needed to solve these equations and obtain ML estimates for the location and scale. In this study, SC, and PSO are some effective and powerful algorithms considered as numerical techniques for estimating the likelihood estimators for the EMLOG distribution, and they are briefly introduced in the next subsections.

2.1 Sine Cosine Algorithm

The SC algorithm is a population-based meta-heuristic technique proposed by Mirjalili in 2016, which is motivated by the mathematical trigonometric sine and cosine functions [19]. It's been utilized to overcome a wide range of optimization issues in several areas by initializing, within the search space, a collection of a population of solutions that are iteratively assessed in relation to the objective function under the control of a set of developed optimization parameters. After that, the algorithm keeps the better solution and continuously updates it until convergence is satisfied by reaching the maximum number of iterations. This updated best position represents the best solution [20, 21].

SC algorithm steps: [22,23]

The main two phases of the SC algorithm are 1) exploration (diversification), considered a global lookup search, and 2) exploitation (intensification), considered a local lookup search. The steps of these phases are summarized as follows:

1. Initialize the position of N numbers of the population solutions randomly within the search space for the first iteration, as well as the random parameters r_1, r_2, r_3, and r_4 of this algorithm, which are incorporated to strike a balance between exploration and exploitation capabilities and thus to avoid settling for local optimums. The parameter r_1 helps in determining whether an updated solution position or the movement direction of the next position is towards the best solution in the search space ($r_1 < 1$) or outwards from it ($r_1 > 1$). The r_1 parameter falls linearly from a constant (a) to 0, as seen in the equation:

$$r_1=a-t*\left(\frac{a}{T_{max}}\right) \quad (5)$$

   The parameter r_2 is set within the interval, which helps in determining how large the extended movement of the solution towards or away from the intended target will be. The r_3 parameter is a random weight score to emphasize ($r_3 > 1$) or underemphasize ($r_3 < 1$) the significant effect of the intended target on distance calculation. The final random parameter, which is a random value defined in [0, 1], can be considered a switch to choose between the trigonometric functions of sine and cosine elements.
2. Evaluate the fitness value of each solution using the fitness effect represented by the objective function in this study. Each fitness value refers to the position of each solution. The best (highest) value in the population is found and saved.
3. Update the main parameters, which are r_1 by using equation (5), and r_2, r_3, and r_4 randomly.
4. Update the positions of all solution agents by utilizing the given equation:

$$X_i^{t+1}=\begin{cases} X_i^t+r_1\times\sin(r_2)\times\left|r_3P_i^t-X_i^t\right|, & r_4<0.5 \\ X_i^t+r_1\times\cos(r_2)\times\left|r_3P_i^t-X_i^t\right|, & r_4\geq 0.5 \end{cases} \quad (6)$$

   where X_i^t denotes the position of the current solution in the i^{th} dimension at the t^{th} iteration and P_i^t denotes the position of the target destination point in the dimension.
5. Loop back to step 2 to continue iterating until the maximum number of iterations is reached. The solution values are called the SC parameter estimates.

2.2 Particle Swarm Optimization

The PSO is considered one of the best-known population-based meta-heuristic algorithms dependent on swarm intelligence. It was proposed by Kenedy and Eberhart [24]. PSO is a simulation of the continuous movements of particles in a swarm in a specific search area that mimics the movement behavior of bird flocks in nature using certain formulas until finally reaching the optimal solution [25]. It can be used to solve various constrained or unconstrained optimization problems, multi-objective optimization, non-linear programming, probabilistic programming, and combinatorial optimization issues [26].

PSO algorithm steps: [27, 28]

1. Initialize randomly the position and velocity of ***N*** number of population solutions (particles) for the first iteration as well as the algorithm parameters, which are c_1, c_2 representing acceleration coefficients, r_1, r_2 representing random numbers uniformly distributed among 0 and 1, and ω indicating the inertia weight parameter.
2. Evaluate the fitness value of each solution (particle) by the fitness function *ln L* in this study. Each fitness value refers to the position of each solution. The best (highest) value of each particle in the population is found, compared with its previous historical movement, and then saved as a personal best solution (pbest) value. At the same time, the best fitness value for each particle is found, compared with the previous historical global best, and saved as a (gbest) value.
3. Update each solution's position and velocity using the following equations:

$$V_i^{t+1}=\omega V_i^t+c_1r_1\left(pbest_i^t-X_i^t\right)+c_2r_2\left(gbest^t-X_i^t\right) \quad (7)$$

$$x_i^{t+1}=x_i^t+V_i^{t+1} \quad (8)$$

where V_i^t represents particle *i*'s velocity at iteration *t*, X_i^t is the location of particle i during iteration *t*, $pbest_i^t$ is the best position of a particle at iteration, and $gbes^t$ is the most optimal (best) location of the group at iteration *t*.

4. Loop back to step 2 again until the convergence is satisfied. The solution values are called the PSO parameter estimates.

3. Monte Carlo Simulation Study

In this section, a Monte-Carlo simulation study is carried out to compare the efficiencies of ML estimators of the model parameters for varying sample sizes, utilizing meta-heuristic algorithms such as the SC, and the PSO algorithm. All computations for the simulation study are made by Matlab R2021a software. Each Monte Carlo simulation run is replicated 1,000 times. The location α and scale β parameters are considered to be (α =1) and (β= 1, 2), respectively, for different values of sample size (n), which is taken as n = 30, 50, 100, 150, and 200. The search space (SS) for both α and β parameters is selected to be [-20, 20]. The resulting estimates for location and scale parameters in the simulations are denoted by $\hat{\alpha}$ and $\hat{\beta}$, respectively. To analyze and evaluate the estimators' performance, the simulated mean, bias, variance, mean square error (MSE), and deficiency (Def) values given by the equations (9–13) below are used.

$$\text{Mean}\left(\hat{\theta}\right)=\frac{\sum_i^n \hat{\theta}_i}{n} \quad (9)$$

$$\text{Bias}\left(\hat{\theta}\right)=E\left(\hat{\theta}\right)-\theta \quad (10)$$

$$\text{Var}\left(\hat{\theta}\right)=\frac{1}{n-1}\sum_{i=1}^n\left(\hat{\theta}_i-\text{Mean }\hat{\theta}\right)^2 \quad (11)$$

$$\text{MSE}\left(\hat{\theta}\right)=\text{Var}\left(\hat{\theta}\right)+\left(\text{Bias}\left(\hat{\theta}\right)\right)^2 \quad (12)$$

$$\text{Def}\left(\hat{\alpha},\hat{\beta}\right)=\text{MSE}(\hat{\alpha})+\text{MSE}\left(\hat{\beta}\right) \quad (13)$$

where, $\theta=(\alpha,\beta)\in R\times R^+$. It can be seen from Table 1 that, according to the bias, MSE, and deficiency criteria, using the PSO algorithm for ML estimation of the EMLOG distribution is more efficient than the SC algorithm. In other words, it's clear that for all *n* values, the simulated values show that the PSO provides noticeably better results than the SC algorithm.

Table 1 is here

4. Conclusion

The EMLOG distribution is a new distribution obtained by combining a logistic distribution with an exponential distribution. Because of its flexibility, this distribution can outperform many prominent distributions, such as gamma, weibull, logistic, and others, and be used as an alternative to them with a superior fit in many application cases. Thus, it has a wide range of use in many areas, such as engineering, medical science, technology, energy, marketing, biology, and psychology. In this study, ML estimates of the EMLOG distribution's location and scale parameters are investigated, which cannot be obtained explicitly because of the complication of finding a solution for their nonlinear likelihood equations. Using meta-heuristic algorithms such as SCA and PSO can be considered a better alternative for finding ML estimators than other numerical algorithms. It is seen that, according to the simulation results, the ML estimates of the PSO algorithm show the best performance in comparison with the SCA algorithm. In our future study, this study will be expanded, and the ML estimates of the EMLOG distribution will be obtained with other efficient meta-heuristic algorithms like Grey Wolf Optimization (GWO) and Whale Optimization Algorithm (WOA) in addition to SCA and PSO algorithms. Then, the performance of the estimators will be compared and evaluated to determine the most efficient algorithm that can be used for estimating the parameters of the EMLOG distribution. Furthermore, some applications will be considered to show the flexibility of this distribution.

References

[1] Hinde, J. (2011). Logistic Normal Distribution, in International Encyclopedia of Statistical Science (pp. 754-755). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-04898-2_342

[2] Kissell, R.L. and Poserina, J. (2017). Advanced Math and Statistics. In Optimal Sports Math, Statistics, and Fantasy, 1st edn. Elsevier, pp 103-135. http://dx.doi.org/10.1016/B978-0-12-805163-4.00004-9

[3] Gupta, S.S. and Balakrishnan, N. (1992). Logistic order statistics and their

properties. Handbook of the Logistic Distribution, 123.

[4] Gupta, S.S. and Gnanadesikan, M. (1966). Estimation of the parameters of the logistic distribution. Biometrika, *53*(3-4): 565-570.https://doi.org/10.2307/2333660

[5] Gupta, A.K., Zeng, W.B., Yanhong, Wu (2010). Exponential Distribution, in Probability and Statistical Models. Birkhäuser, Boston, MA pp 23-43.https://doi.org/10.1007/978-0-8176-4987-6_2

[6] Rasekhi, M, Alizadeh, M, Altun, E, Hamedani, G.G., Afify, A.Z., Ahmad, M. (2017). The Modified Exponential Distribution With Applications. Pakistan Journal of Statistics, 33(5): 383-398.

[7] Aldahlan, M.A. and Afify, A.Z. (2020). The odd exponentiated half-logistic exponential distribution: estimation methods and application to engineering data. Mathematics, 8(10):1684. https://doi.org/10.3390/math8101684

[8] Gómez, Y.M, Bolfarine, H, Gómez, H.W. (2014). A new extension of the exponential distribution. Revista Colombiana de Estadística, 37(1):25-34. https://doi.org/10.15446/rce.v37n1.44355

[9] Hussein, M, Elsayed, H, Cordeiro, G.M. (2022). A New Family of Continuous Distributions: Properties and Estimation. Symmetry, 14(2):276. https://doi.org/10.3390/sym14020276

[10] Gupta, R.D, Kundu, D. (2001). Exponentiated exponential family: an alternative to gamma and Weibull distributions. Biometrical Journal: Journal of Mathematical Methods in Biosciences, 43(1):117-130.https://doi.org/10.1002/1521-4036(200102)43:1<117::AID-BIMJ117>3.0.CO;2-R

[11] Pal, M, Ali, M.M., Woo, J. (2006). Exponentiated weibull distribution. Statistica, 66(2):139-147. https://doi.org/10.6092/issn.1973-2201/493

[12] Nadarajah, S and Gupta, A.K. (2007). The exponentiated gamma distribution with application to drought data. Calcutta Statistical Association Bulletin, 59(1-2):29-54. https://doi.org/10.1177/0008068320070103

[13] Nadarajah, S and Haghighi, F. (2011). An extension of the exponential distribution. Statistics, 45(6):543-558. http://dx.doi.org/10.1080/02331881003678678

[14] Chaudhary, A.K.(2019). Frequentist Parameter Estimation of Two-Parameter Exponentiated Log-logistic Distribution BB. NCC Journal, 4(1):1-8. https://doi.org/10.3126/nccj.v4i1.24727

[15] Reyes, J., Venegas, O., Gómez, H.W. (2018). Exponentially-modified logistic distribution with application to mining and nutrition data. Appl. Math. Inf. Sci. 12(6):1109-1116. *http://dx.doi.org/10.18576/amis/120605*

[16] Grushka, E. (1972). Characterization of Exponentially Modified Gaussian Peaks in Chromatography. Anal. Chem., 44, 1733–1738.

[17] Yuan, K H. and Schuster, C. , 18 Overview of Statistical Estimation Methods. The Oxford handbook of quantitative methods, 361.http://dx.doi.org/10.1093/oxfordhb/9780199934874.013.0018

[18] Bartolucci, F. and Scrucca, L. (2010). Point Estimation Methods with Applications to Item Response Theory Models. https://doi.org/10.1016/B978-0-08-044894-7.01376-2

[19] Mirjalili, S. (2016). SCA: a sine cosine algorithm for solving optimization problems. Knowledge-based systems, 96: 120-133. http://dx.doi.org/10.1016/j.knosys.2015.12.022

[20] Abualigah, L and Diabat, A. (2021). Advances in sine cosine algorithm: a comprehensive survey. 54(4):2567-2608. https://doi.org/10.1007/s10462-020-09909-3

[21] Suid, M.H., Ahmad, M.A., Ismail, M.R.T.R, Ghazali, M.R, Irawan, A, Tumari, M.Z., (2018) An improved sine cosine algorithm for solving optimization problems. In 2018 IEEE Conference on Systems, Process and Control (ICSPC): 209-213.IEEE. https://doi.org/10.1109/SPC.2018.8703982

[22] Rizk-Allah, R.M, Mageed, H.A., El-Sehiemy, R.A., Aleem, S.H.E.A, El Shahat, A. (2017). A new sine cosine optimization algorithm for solving combined non-convex economic and emission power dispatch problems. Int J. Energy Convers, 5(6):180-192. https://doi.org/10.15866/irecon.v5i6.14291

[23] Gabis, A.B., Meraihi, Y., Mirjalili, S., Ramdane-Cherif, A. (2021). A comprehensive survey of sine cosine algorithm: variants and applications. Artificial Intelligence Review, 54(7):5469-5540. https://doi.org/10.1007/s10462-021-10026-y

[24] Kennedy, J. and Eberhart, R. (1995) Particle swarm optimization. in Proceedings of ICNN'95-international conference on neural networks. 4:1942-1948. IEEE. https://doi.org/10.1109/ICNN.1995.488968

[25] Júnior, S.F.A.X, Xavier, É.F.M, da Silva Jale, J., de Oliveira, T.A., Sabino,. A.L.C. (2020).

An application of Particle Swarm Optimization (PSO) algorithm with daily precipitation data in Campina Grande, Paraíba, Brazil. *Research, Society and Development*, *9*(8): e444985841-e444985841. http://dx.doi.org/10.33448/rsd-v9i8.5841

[26] El-Shorbagy, M.A. and Hassanien, A.E. (2018). Particle swarm optimization from theory to applications. *International Journal of Rough Sets and Data Analysis (IJRSDA)*, *5*(2):1-24. http://dx.doi.org/10.4018/IJRSDA.2018040101

[27] Ren, C., An, N., Wang, J., Li, L., Hu, B., Shang, D. (2014). Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting. Knowledge-based systems, 56:226-239. http://dx.doi.org/10.1016/j.knosys.2013.11.015

[28] Ab Talib, M.H., Mat Darus, I.Z. (2017). Intelligent fuzzy logic with firefly algorithm and particle swarm optimization for semi-active suspension system using magneto-rheological damper. Journal of Vibration and Control, 23(3): 501-514. http://dx.doi.org/10.1177/1077546315580693

Table 1. Simulated Mean, Bias, Variance, MSE, and Def values for the ML estimators $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\beta}}$.

n	Method	$\hat{\alpha}$				$\hat{\beta}$				
$\alpha = 1, \beta = 1$		Mean	Variance	Bias	MSE	Mean	Variance	Bias	MSE	Def
30	SC	2.0237	17.8320	1.0237	18.8800	0.9270	0.0652	-0.0730	0.0705	18.9505
	PSO	0.9325	0.3457	-0.0675	**0.3503**	1.0387	0.1069	0.0387	**0.1084**	**0.4587**
50	SC	2.2017	21.4220	1.2017	22.8661	0.9408	0.0688	-0.0592	0.0723	22.9384
	PSO	0.9635	0.1723	-0.0365	**0.1736**	1.0285	0.0419	0.0285	**0.0427**	**0.2163**
100	SC	2.0745	19.1190	1.0745	20.2736	0.9394	0.0546	-0.0606	**0.0583**	20.3318
	PSO	0.9286	0.2145	-0.0714	**0.2196**	1.0408	0.0784	0.0408	0.0801	**0.2997**
150	SC	1.7817	14.2310	0.7818	14.8422	0.9598	0.0400	-0.0402	**0.0416**	14.8838
	PSO	0.9331	0.2009	-0.0669	**0.2054**	1.0434	0.0665	0.0434	0.0684	**0.2738**
200	SC	1.8953	16.2030	0.8953	17.0046	0.9519	0.0445	-0.0481	**0.0468**	17.0514
	PSO	0.9356	0.1518	-0.0644	**0.1559**	1.0407	0.0550	0.0407	0.0567	**0.2126**
$\alpha = 1, \beta = 2$										
30	SC	1.7568	13.7140	0.7569	14.2869	1.8889	0.2220	-0.1111	0.2343	14.5212
	PSO	1.0185	0.6051	0.0185	**0.6054**	1.9751	0.1007	-0.0249	**0.1013**	**0.7068**
50	SC	1.4822	9.1786	0.4822	9.4111	1.9197	0.1437	-0.0803	0.1501	9.5613
	PSO	0.9932	0.3727	-0.0068	**0.3727**	1.9882	0.0671	-0.0118	**0.0672**	**0.4400**
100	SC	1.3018	5.8544	0.3018	5.9455	1.9517	0.0862	-0.0483	0.0885	6.0340
	PSO	0.9628	0.2391	-0.0372	**0.2405**	2.0028	0.0449	0.0028	**0.0449**	**0.2854**
150	SC	1.1800	3.3322	0.1800	3.3646	1.9667	0.0506	-0.0333	0.0517	3.4163
	PSO	0.9913	0.1655	-0.0087	**0.1656**	2.0034	0.0369	0.0034	**0.0369**	**0.2025**
200	SC	1.2482	4.3658	0.2483	4.4275	1.9562	0.0565	-0.0438	0.0584	4.4859
	PSO	0.9955	0.1422	-0.0045	**0.1422**	1.9972	0.0294	-0.0028	**0.0294**	**0.1716**