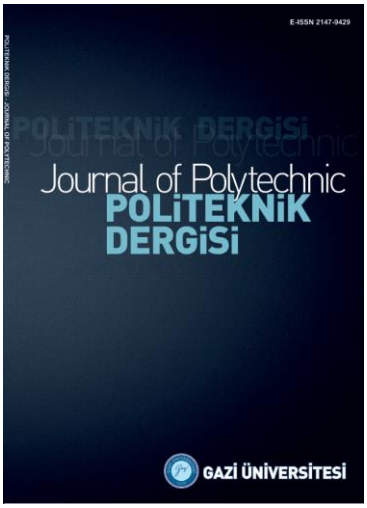


POLİTEKNİK DERGİSİ

JOURNAL of POLYTECHNIC

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)
URL: http://dergipark.org.tr/politeknik

CLARIVATE ANALYTICS
EMERGING SOURCES CITATION INDEX
INDEXED

Implementation of a chaotic particle swarm optimization algorithm to estimate the parameters of weibull distribution

Weibull dağılımının parametrelerini tahmin etmek için kaotik parçacık sürüsü optimizasyon algoritmasının uygulanması

***Yazar(lar) (Author(s))**: Emre KOÇAK[1], Büşra AKSEL[2], H. Hasan ÖRKCÜ[3]*

ORCID[1]: 0000-0001-6686-9671

ORCID[2]: 0000-0001-9170-4959

ORCID[3]: 0000-0002-2888-9580

ERKEN GÖRÜNÜM

To cite to this article: Koçak E., Aksel B., and Örkcü H. H., "Implementation of a Chaotic Particle Swarm Optimization Algorithm to Estimate the Parameters of Weibull Distribution", ***Journal of Polytechnic***, *(*): *, (*).

Bu makaleye şu şekilde atıfta bulunabilirsiniz: Koçak E., Aksel B. ve Örkcü H. H., "Implementation of a Chaotic Particle Swarm Optimization Algorithm to Estimate the Parameters of Weibull Distribution", ***Politeknik Dergisi***, *(*): *, (*).

Erişim linki (To link to this article): http://dergipark.org.tr/politeknik/archive

DOI: 10.2339/politeknik.1733077

Implementation of a Chaotic Particle Swarm Optimization Algorithm to Estimate the Parameters of Weibull Distribution

Highlights

- *Experiments show that the suggested chaotic map approach performs better than the traditional linear decreasing inertia weights.*
- *The findings of the simulation show that one type of chaotic map cannot outperform the others since different chaotic maps perform better in various contexts.*
- *Better results have been obtained by including chaotic maps into such algorithms in order to overcome the early convergence issue that prevents improved performance in challenging optimization problems.*

Graphical Abstract

The parameters of the three-parameter Weibull distribution, which are notoriously difficult to estimate, are estimated using the Particle Swarm Optimization algorithm.

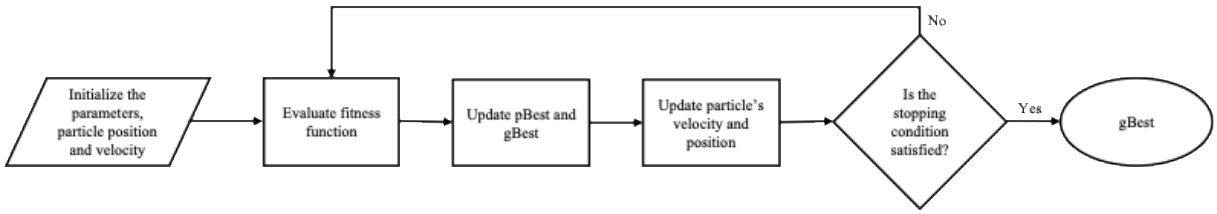


Figure. The flowchart of PSO

Aim

It was attempted to avoid local solutions more readily by employing chaotic maps as opposed to classical methods.

Design & Methodology

Chaotic maps were used to overcome the premature convergence problem, which is a problem in Particle Swarm Optimization, and the results were comparatively examined in the parameter estimation of the Weibull distribution.

Originality

A significant feature of this research is the detailed examination of chaotic maps for the inertia weight, which influences the convergence of the Particle Swarm Optimization, for the estimate of various parameter values of the three-parameter Weibull distribution.

Findings

Simulation findings demonstrate that for predicting the three-parameter Weibull distribution, the suggested chaotic map technique outperforms traditional linear decreasing inertia weights.

Conclusion

Chaotic maps should be integrated into such algorithms to avoid the premature convergence problem.

Declaration of Ethical Standards

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

Implementation of a Chaotic Particle Swarm Optimization Algorithm to Estimate the Parameters of Weibull Distribution

Research Article / Araştırma Makalesi

Emre KOÇAK[1*], Büşra AKSEL[1], H. Hasan ÖRKCÜ[1]

[1]Department of Statistics, Faculty of Science, Gazi University, Türkiye

(Received/Geliş : 03.07.2025 ; Accepted/Kabul : 07.10.2025 ; Early View/Erken Görünüm : 02.11.2025)

ABSTRACT

Numerous studies on the statistical inferences of the Weibull distribution's parameters have been performed because it is among the most well-known and widely applied distributions in several fields, including lifetime studies and reliability. Although maximum likelihood is a widely used method in the estimation of unknown parameters, estimating the parameters by maximizing the likelihood function is very challenging for some distributions, like the three-parameter Weibull distribution. The Particle Swarm Optimization (PSO) algorithm is examined in order to address this issue and achieve improved outcomes. However, different parameter values for the algorithm need to be adjusted to achieve good results and increase the performance of PSO. In this context, it is very important to determine the inertia weight, which significantly affects the search process. As a novelty in this paper, chaotic maps for the inertia weight, which is the factor affecting the convergence of the PSO, are examined in detail for the estimation of different parameter values of the three-parameter Weibull distribution. The effectiveness of the suggested method is investigated by a thorough Monte-Carlo simulation analysis. The simulation findings demonstrate that the proposed chaotic map approach outperforms the classic linear decreasing inertia weights.

Keywords: Chaotic map, particle swarm optimization, weibull distribution, maximum likelihood

Weibull Dağılımının Parametrelerini Tahmin Etmek için Kaotik Parçacık Sürüsü Optimizasyon Algoritmasının Uygulanması

ÖZ

Güvenilirlik ve yaşam süresi çalışmaları gibi çeşitli disiplinlerde en iyi bilinen ve yaygın olarak kullanılan dağılımlardan biri olduğu için Weibull dağılımının parametrelerinin istatistiksel çıkarımları üzerine çok sayıda çalışma yapılmıştır. Maksimum olabilirlik, bilinmeyen parametrelerin tahmin sürecinde yaygın olarak kullanılan bir yöntem olmasına rağmen, üç parametreli Weibull dağılımı gibi bazı dağılımlar için, olabilirlik fonksiyonunu maksimize ederek parametreleri tahmin etmek oldukça zordur. Bu sorunu ele almak ve daha iyi sonuçlar elde etmek amacıyla Parçacık Sürü Optimizasyonu (PSO) algoritması incelenmektedir. Ancak, iyi sonuçlar elde etmek ve PSO'nun performansını artırmak için algoritma için farklı parametre değerlerinin ayarlanması gerekmektedir. Bu bağlamda, arama sürecini önemli ölçüde etkileyen atalet ağırlığının belirlenmesi oldukça önemlidir. Bu makalede bir yenilik olarak, PSO'nun yakınsamasını etkileyen faktör olan eylemsizlik ağırlığı için kaotik haritalar ele alınmış ve üç parametreli Weibull dağılımının farklı parametre değerlerinin tahmini için ayrıntılı olarak incelenmiştir. Önerilen yöntemin etkinliği kapsamlı bir Monte-Carlo simülasyon analizi ile araştırılmıştır. Simülasyon bulguları, önerilen kaotik harita yaklaşımının klasik doğrusal azalan eylemsizlik ağırlıklarından daha iyi performansa sahip olduğunu göstermektedir.

Anahtar Kelimeler: Kaotik harita, parçacık sürüsü optimizasyonu, weibull dağılımı, maksimum olabilirlik

ERKEN GÖRÜNÜM

1. INTRODUCTION

The three-parameter Weibull distribution, which is a popular distribution due to application in numerous scientific areas, such as reliability, engineering, and renewable energy, is handled by researchers in different disciplines. It is commonly well-known that the three-parameter Weibull distribution can match a very wide range of empirical observations quite well [1]. For this reason, statistical inferences regarding the parameters of this distribution have gained importance over time and various studies have been conducted on this subject.

The cumulative distribution function (cdf) and the corresponding probability density function (pdf) of the three-parameter Weibull distribution are given by

$$F_X(x) = 1 - e^{-\left(\frac{x-\gamma}{\eta}\right)^{\beta}} \quad (1)$$

$$f_X(x) = \frac{\beta}{\eta}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} e^{-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}} \quad (2)$$

where $x > \gamma > 0$, $\eta > 0$ and $\beta > 0$ are location, scale, and shape parameter, respectively [1]. The mean, variance, skewness ($\mathcal{H}_1$), and kurtosis ($\mathcal{H}_2$), which represent the four basic statistics of this distribution, are defined by

*Corresponding Author (Sorumlu Yazar)
e-posta : emrekocak@gazi.edu.tr

$$E(x) = \gamma + \eta\Gamma\left(1+\frac{1}{\beta}\right) \quad (3)$$

$$Var(x) = \eta^2\left(\Gamma\left(1+\frac{2}{\beta}\right) - \left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^2\right) \quad (4)$$

$$\mathcal{H}_1 = \frac{\Gamma\left(1+\frac{3}{\beta}\right)-3\Gamma\left(1+\frac{2}{\beta}\right)\Gamma\left(1+\frac{1}{\beta}\right)+2\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^3}{\left(\Gamma\left(1+\frac{2}{\beta}\right)-\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^2\right)^{3/2}} \quad (5)$$

$$\mathcal{H}_2 = \frac{\Gamma\left(1+\frac{4}{\beta}\right)-4\Gamma\left(1+\frac{3}{\beta}\right)\Gamma\left(1+\frac{1}{\beta}\right)+6\Gamma\left(1+\frac{2}{\beta}\right)\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^2-3\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^4}{\left(\Gamma\left(1+\frac{2}{\beta}\right)-\left[\Gamma\left(1+\frac{1}{\beta}\right)\right]^2\right)^2} \quad (6)$$

where Γ demonstrates the Gamma function [2, 3]. The skewness and kurtosis values of the three-parameter Weibull distribution are shown in Table 1 to further comprehend the form of the distribution. Additionally, depending on the various β values, the Weibull distribution has several different shapes. For a few example values of the shape parameter β, the pdf and cdf plots of the Weibull distribution are shown in Figure 1. The skewness and kurtosis values of the three-parameter Weibull distribution are extremely varied, as seen in Figure 1 and Table 1. Consequently, due to its flexibility, it can be utilized for modeling purposes in a variety of scientific areas.

Table 1. The skewness and kurtosis values of the three-parameter Weibull distribution for values of the shape parameter

β	1.5	2	2.5	3	4	6
$\mathcal{H}_1$	1.072	0.631	0.359	0.168	-0.087	-0.373
$\mathcal{H}_2$	4.390	3.245	2.857	2.730	2.748	3.036

Estimation theory is crucial for graphic methods and statistical studies, including the Maximum Likelihood (ML), Least Square (LS), weighted LS, and method of moments, which are a few popular techniques for estimating the Weibull distribution's unknown parameters [4, 5]. Various studies have been carried out on the estimation methods of Weibull parameters in the literature [4-17]. As a result of these studies, one of the best approaches for estimating distribution parameters with a variety of desired attributes is generally the ML approach, even if there are numerous different techniques described for estimating the parameters of a probability distribution using a specific sample in the literature. Since ML estimators are asymptotically unbiased with the least variance, it is also a widely used technique for parameter estimation. [18]. Because of both its usefulness in complex estimation issues and providing estimators with many desirable statistical properties, ML is the most often used estimating methodology.

The parameter vector of the distribution considered in this article is $\theta = (\gamma, \eta, \beta)$. Finding a vector that optimizes the likelihood function was the goal in this situation. We may also utilize its logarithm, say lnL, to maximize L. When used with a three-parameter Weibull distribution, maximizing the L function is a difficult solution.The following is the three-parameter Weibull distribution's L function:

$$L = \prod_{i=1}^{n} f_{X_i}(x_i;\gamma,\eta,\beta) = \prod_{i=1}^{n} \frac{\beta}{\eta}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1} e^{-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}} \quad (7)$$

Its logarithm is as follows:

$$lnL = nln\left(\frac{\beta}{\eta}\right) + \sum_{i=1}^{n}\left[-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta} + (\beta-1)ln\left(\frac{x_i-\gamma}{\eta}\right)\right] \quad (8)$$

The likelihood objective function or its logarithm may be maximized using a variety of numerical approaches, such as traditional gradient-based and direct search techniques. Because the search progresses from the initial parameter estimate to the maximum or minimum, these techniques might be considered as local search techniques [19]. It is important to note that conventional optimization methods do not ensure the finding of a sequence of parameter values that precisely maximizes the lnL [4, 7]. To overcome these difficulties, which are related to the multimodal nature of a large number of parameters and objective functions, it may be thought about using an effective heuristic optimization method. Particle Swarm Optimization (PSO), the most well-known and widely used in different fields, is a population-based computational intelligence method that can converge to an optimal solution in numerous situations while most analytical approaches are unable to do so. In addition, it is not so much impacted by the dimension and nonlinearity of the problem [20]. It offers a few advantages, including ease of use, reliability, and widespread acceptance as a powerful method for global optimization. For these reasons, the importance of this paper is further enhanced by the use of the PSO algorithm.

ERKEN GÖRÜNÜM

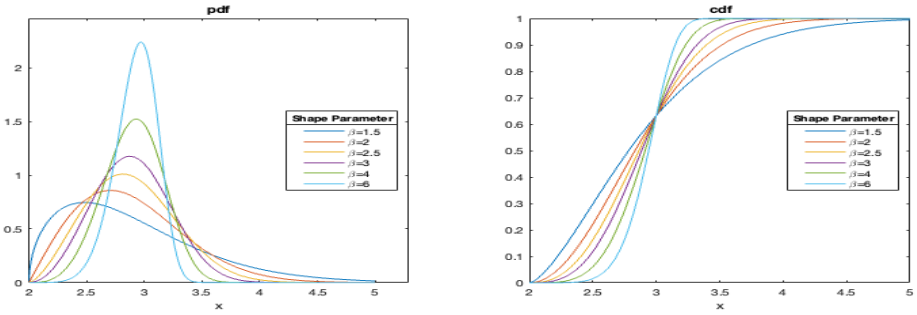


Figure 1. The pdf and cdf plots of the three-parameter Weibull distribution for values of the shape parameter $(Weibull(\gamma = 2, \eta = 1, \beta))$

Comparatively to two-parameter examples, estimating the parameters of the three-parameter Weibull distribution family is an extremely challenging task. As far as we know, there is not a lot of paper in the literature on using PSO to estimate the three-parameter Weibull parameters, which will also be discussed in this paper. Wen et al. [21] suggested using the PSO algorithm-based correlation coefficient optimization method to estimate the parameters of the three-parameter Weibull distribution. Örkcü et al. [4] presented a comprehensive Monte-Carlo simulation study on ML estimation of the parameters of the three-parameter Weibull distribution using a PSO approach. Acitas et al. [5] conducted a parameter estimation study for the three-parameter Weibull distribution by proposing a new adaptive search space based on confidence intervals in the PSO.

The inertia weight in the PSO algorithm controls the balance between global search and local search ability. Because the global search is simplified by a large inertia weight, whereas the local one is simplified by a small inertia weight. Therefore, it is important to modify the inertia weight appropriately. The originality of this paper is to consider different chaotic map types instead of inertial weight, which is the most important factor affecting the search process in PSO. In this study, it was tried to avoid local solutions more easily compared to classical algorithms. In this way, it was aimed to increase global convergence. In addition to make parameter estimation with a comprehensive Monte Carlo simulation study in different search spaces for different parameter values of the three-parameter Weibull distribution, which has a highly flexible structure.

The remainder of the paper is organized as follows five sections: Section 2 describes the PSO and Section 3 outlines the proposed methodology. Experimental results are presented in Section 4; Section 5 concludes the paper.

2. PARTICLE SWARM OPTIMIZATION (PSO)

PSO, first presented by Kennedy and Eberhart [22] and inspired by the collective behavior of bird flocks, is an optimization approach that uses a population-based stochastic process. PSO, also referred as an optimizer, is a population-based, self-adaptive search optimization technique [23]. In PSO, each named particle in the population, called swarm, modifies its trajectory to move toward both its own best position and the best position previously attained by any member of its topological neighborhood [24].

A population is a group of solutions or particles that make up PSO. Each solution is a point in a multidimensional space represented by a set of parameters. All particles in the swarm work individually under the rule of acceleration towards the best individual and overall position, constantly evaluating the value of their present position. Each particle is equipped with a memory that enables it to remember its prior best position, known as the personal best (pbest) and the global best (gbest). The personal best, or pbest, is the location that a particle individually found to have the highest fitness value. Based on the path it has taken, each particle has its own pbest. The particle compares the fitness value of its present location to that of pbest at each point along its path. Pbest is changed to the present location if the present one has a greater fitness value. It was also possible to determine the location of each particle's maximum concentration inside the swarm. The global best, or gbest, is the name given to this site for the best fitness ever observed. There is just one gbest to which every particle is pulled for the entire swarm. Each particle compares the fitness of its present location to that of gbest at each point along its path. Gbest is changed to reflect a particle's present location if it is at a location with a higher fitness value.

In a n-dimensional search space, the position and velocity of particle i are represented as the vectors $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$ and $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$. Each particle in the swarm is given a score using a function to determine its fitness value, which indicates how well it solves the problem. Each particle keeps track of its own pbest and gbest. Then, updated rules for new positions are applied to all particles in the n-dimensional search space until the global optimal position is discovered. The modified velocity and position of each particle can be calculated as in Eq. (9) and Eq. (10), respectively [20]:

$$V_i^{k+1} = \omega V_i^k + c_1 R_{rand_1}\left(pbest_i^k - X_i^k\right) + c_2 R_{rand_2}\left(gbest^k - X_i^k\right) \quad (9)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (10)$$

where V_i^k the velocity of particle i at iteration k, ω inertia weight, c_1, c_2 acceleration coefficients, R_{rand_1}, R_{rand_2} are random numbers uniformly distributed $(0,1)$, X_i^k position of particle i at iteration k, pbest_i^k best position of particle i until iteration k, and gbest^k best position of the group until iteration k.

According to Eq. (9), a particle's velocity changes depending on three components. The first component is its previous velocity, V_i^k, scaled by an inertia, ω. This component is frequently referred to as habitual behavior. The second component is a linear attraction toward its previous best position, pbest_i^k, scaled by the product of acceleration coefficients, c_1, and a random number. It should be noted that each dimension is assigned a different random number. This component is frequently referred to as memory or self-knowledge. The third component is a linear attraction toward the global best position, gbest^k scaled by the product of an acceleration constant, c_2, and a random number. This component is frequently referred to as teamwork or social knowledge [4].

The weights of stochastic acceleration terms that direct a particle towards pbest and gbest, respectively, are indicated by personal and social learning factors with acceleration coefficients c_1 and c_2. Small values enable a particle to explore far away from the target locations, while large ones cause the particles to move suddenly

ERKEN GÖRÜNÜM

towards the target locations. Acceleration coefficient values between 0 and 4 are usually taken as equal.

The control of global exploration and local exploitation is a key in the issue PSO algorithm. Shi and Eberhart [25] presented the notion of inertia weight to the first version of PSO to equilibrate local and global searches throughout the evolution process. Appropriate inertial weight selection requires fewer iterations on average to find an optimal solution, as it ensures a balance between global and local exploration. In this context, the classic linearly decreasing inertia weight concept applied by Shi and Eberhart [25] to PSO is as given in Eq. (11).

$$\omega = \omega_{max} - \frac{\omega_{max}-\omega_{min}}{Iter_{max}} \text{ x } Iter \quad (11)$$

where initial and final inertia weights are ω_{max} and ω_{min}, respectively. $Iter$ is the current iteration number, whereas $Iter_{max}$ is the maximum iteration number [25]. In literature studies, the values of ω_{max} and ω_{min} parameters are usually set as 0.9 and 0.4, respectively [4, 21].

3. THE PROPOSED METHOLOGY

The PSO search process will be significantly impacted by the inertia weight ω, which is the primary factor driving convergence. The global search is facilitated by a large inertia weight, whereas the local search is facilitated by a small inertia weight. Therefore, it is important to set the inertia weight value correctly for the algorithm to work effectively. The PSO process frequently experiences particle entrapment in a local optimum, which results in the above-mentioned premature convergence. For this reason, chaotic maps are employed to avoid early convergence.

3.1. Chaotic Maps

Chaos, which is generally a dynamic, nonconvergent and deterministic method, is a phenomenon where even a small modification in the starting circumstances can result in a nonlinear change in behavior that follows [26]. Although it appears random and unpredictable, it also has a certain element of regularity [27]. The nature-inspired optimization techniques include some parameters that are chosen at random from a Gaussian or uniform distribution. However, chaos behaves similarly to chance but has better statistical and dynamical properties [28]. The chaotic variables must have these properties to cycle through all possible states within a certain range without repeating any of them. Consequently, chaos search, as opposed to traditional stochastic search, can avoid a local optimal solution, and increase the convergence rate.

In this paper, five different one-dimensional chaotic maps are adopted to obtain the chaotic sets. These chaotic maps are widely used in many related works [29-33]. The mathematical definitions of the five applied chaotic maps are listed in Table 2.

Table 2 shows these maps, where w_t indicates the t-th number in the chaotic sequence and t is defined as the index of the chaotic sequence w. The other parameters including c, d, and μ are the control parameters. These parameters are used to determine the chaotic behavior of the dynamic system. For all chaotic maps, we set the initial value w_0 to 0.7, which is according to [31, 32, 34]. Figure 2 demonstrates the visualization of five applied chaotic maps over 500 iterations. As can be seen, each chaotic map represents a different kind of chaotic behavior.

Table 2. Chaotic maps

Chaotic map	Mathematical formula	Parameter values	Value range
Circle	$w_{t+1} = mod\left(w_t + d - \left(\frac{c}{2\pi}\right) sin(2\pi w_t), 1\right)$	$c = 0.5, d = 0.2$	(0,1)
Logistic	$w_{t+1} = cw_t(1 - w_t)$	$c = 4$	(0,1)
Piecewise	$w_{t+1} = \begin{cases} \frac{w_t}{l}, & 0 \le w_t < l \\ \frac{w_t - l}{0.5 - l}, & l \le w_t < 0.5 \\ \frac{1 - l - w_t}{0.5 - l}, & 0.5 \le w_t < 1 - l \\ \frac{1 - w_t}{l}, & 1 - l \le w_t < 1 \end{cases}$	$l = 0.4$	(0,1)
Singer	$w_{t+1} = \mu(7.86w_t - 23.31w_t^2 + 28.75w_t^3 - 13.302875w_t^4)$	$\mu = 1.07$	(0,1)
Tent	$w_{t+1} = \begin{cases} \frac{w_t}{0.7}, & w_t < 0.7 \\ \frac{10}{3}(1 - w_t), & w_t \ge 0.7 \end{cases}$		(0,1)

ERKEN GÖRÜNÜM

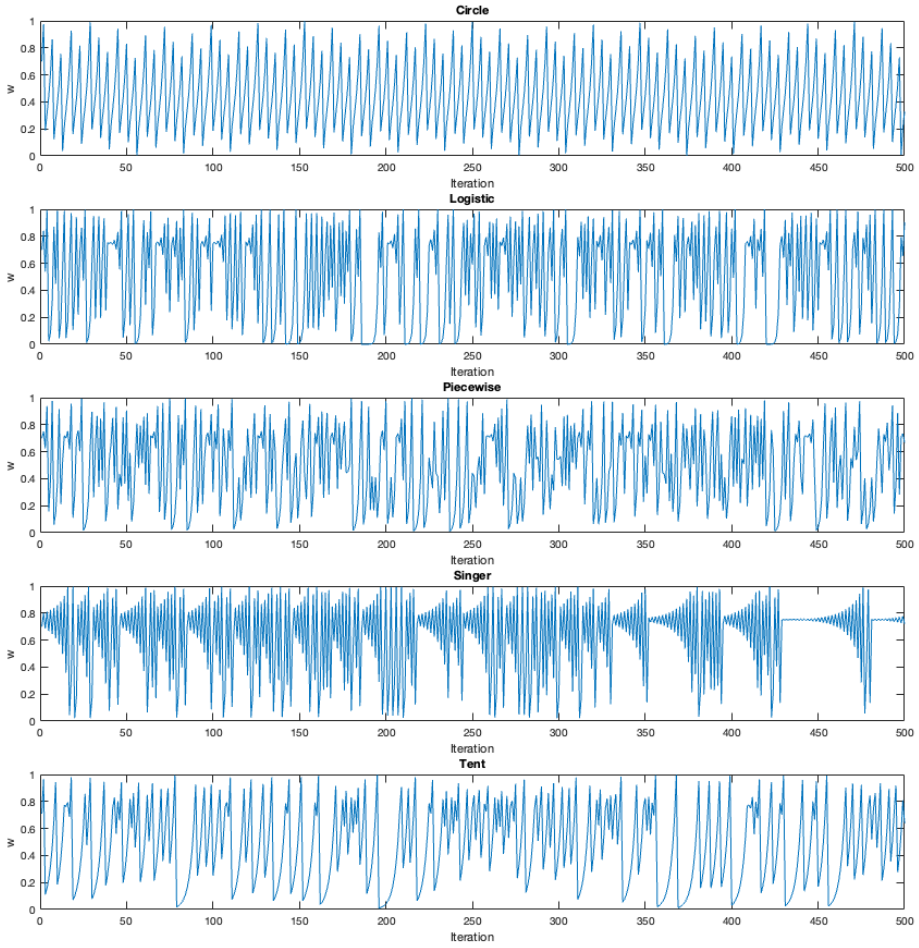


Figure 2. Chaotic maps

4. EXPERIMENTAL RESULTS

For an accurate and sensitive estimation in ML, it is necessary to be aware of the type of distribution from which the samples are drawn to estimate the parameters of any distribution type, not just the three-parameter Weibull distribution. Several real values of the three-parameter Weibull distribution's parameters, as described in vector $\theta = (\gamma, \eta, \beta)$, are used, along with various sample sizes n, to evaluate the performance of the PSO method.

In this study, the parameters $\theta = (2,1,2.5)$, $\theta = (2,1,4)$, and $\theta = (2,1,6)$ are taken into account. Therefore, a sample space known to originate from the Weibull distribution obtained with the help of the formula given below must be found initially. Meanwhile, this formula is the inverse of the cdf function given in Eq. (1).

$$X: \gamma, \eta, \beta \sim \gamma + \eta\left(-\log\left(rand(0,1)\right)\right)^{\frac{1}{\beta}} \quad (12)$$

Where $rand(0,1)$ is a uniform random number in the range $[0,1]$.

To demonstrate the development in the ML estimate for PSO algorithm, three different circumstances for $\theta = (\gamma, \eta, \beta)$ and various sample sizes of 30, 50, 100, and 500 for each example are used. The basic estimating theory makes it clear that a larger sample size will result in better estimation. However, in this case, the likelihood function to maximize will get increasingly challenging as the sample size increases.

In PSO, particle size is crucial; robustness and algorithm complexity are both impacted by it. Large populations will require more computational work and may cause slow convergence, whereas small populations may produce local convergence. Particle sizes of 50 and 250 are used in this study depending on the three-parameter Weibull likelihood function's structure, the search space, and the number of estimated parameters, like Örkcü et al. [4]. The personal and social learning factors, or acceleration coefficients c_1 and c_2, are often chosen with c_1 equal to c_2 and a range of [0, 4]. Coefficients c_1 and c_2 are both set to 2 in this study [4, 25].

In addition to all these parameters, it is an important issue to determine the inertia weight ω, which is examined in detail in this study because of the significance of the convergence behavior of the PSO algorithm. The record of prior velocities impacts the present velocity, and this effect is controlled by the inertia weight. The parameter thus controls the swarm's ability to trade off its ability to explore both locally and globally. The swarm's ability to explore both locally and globally is balanced by an

appropriate value for the inertia weight, which leads to a better solution. A large inertia weight helps with global searches whereas a small inertia weight helps with local searches. For the algorithm to work effectively, it is crucial to set the inertia weight value appropriately. Premature convergence is caused by particle entrapment in a local optimum, which occurs often in the PSO process. Because of this, five different chaotic maps are used to prevent early convergence in this study.

How it optimizes the search space depends on the values of these parameters. The [0, 10] and [0, 1000] search ranges are chosen for each of the three parameters in this study. The search space was altered in accordance with x samples in each experiment because of $x > \gamma > 0$. In this paper, where the stopping criterion is used as the repetition limit, the algorithm is stopped after 100 iterations, and for the deficiencies (DEFs) to be used in the evaluation of the predictions, 1000 different trials are performed. The overall efficiency of the ML estimators of the model parameters is measured by the DEF value, which is defined as Eq. (13) [35]. This scale is essential for evaluating how well parameter estimate methods perform [36]. Furthermore, the parameters used to calculate the Def criterion have mean squared error (MSE) values that are provided in Eq. (14-16). When all cases are considered, a total of $3\ x\ 4\ x\ 2\ x\ 2\ x\ 6\ x\ 1000 = 288000$ independent experiments are carried out.

The following conclusions can be drawn from Table 3-8, which shows the estimation results for different parameters. In addition, detailed graphs of the parameter results are given in the Appendix. As an example, the case where parameter values are $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$, the sample size is 30; particle size is 50; and search space is [0, 10] and [0, 1000] are given in Figure 3 and Figure 4, respectively. In terms of the Def criterion, it is obviously seen that all chaotic maps give better results than the classic linearly decreasing inertia weight concept applied by Shi and Eberhart [37] for all cases. Furthermore, the Tent has never been the type of chaotic map that gives the best results.

The Def value has a lower score even though the particle size is high for all cases in the classic method and all chaotic map types. For example, for $\theta = (2,1,2.5)$, $n = 30$, and [0, 10] search space, Def values are 3.4003 and 0.4307 in particle size 50 and 250, respectively. Similarly, the Def value decreased as the sample size increased while the particle size and search space were the same for the classic method and all chaotic map types. For example, consider the case where the particle size and search space for $\theta = (2,1,2.5)$ are 50 and [0, 10], respectively. For this case, the Def values are obtained as 3.4003, 3.2932, 3.2509, and 2.8359 in sample sizes 30, 50, 100, and 500, respectively.

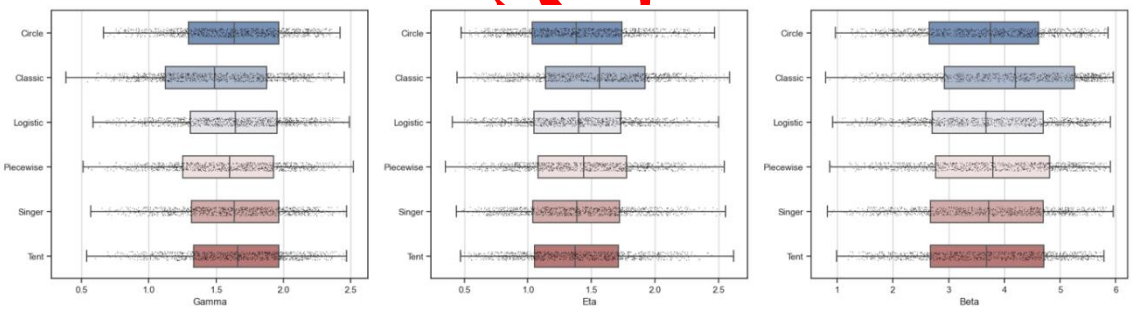


Figure 3. The boxplot graphs of 1000 independent experiments for the $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$, $n = 30$, particle size 50, and [0, 10] search space

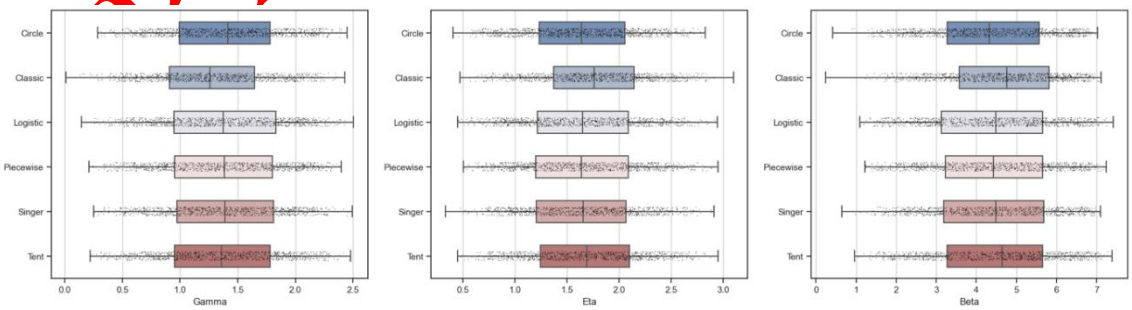


Figure 4. The boxplot graphs of 1000 independent experiments for the $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$, $n = 30$, particle size 50, and [0, 1000] search space

$$DEF(\hat{\gamma}, \hat{\eta}, \hat{\beta}) = MSE(\hat{\gamma}) + MSE(\hat{\eta}) + MSE(\hat{\beta}) \quad (13)$$

$$MSE(\hat{\gamma}) = Var(\hat{\gamma}) + Bias^2(\hat{\gamma}) \quad (14)$$

$$MSE(\hat{\eta}) = Var(\hat{\eta}) + Bias^2(\hat{\eta}) \quad (15)$$

$$MSE(\hat{\beta}) = Var(\hat{\beta}) + Bias^2(\hat{\beta}) \quad (16)$$

Table 3. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$ for [0, 10] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.6303	2.0814	1.6077	2.0532	1.5817	2.0302	1.6055	2.0075
	$\hat{\eta}$	1.3936	0.9025	1.4157	0.9357	1.4444	0.9648	1.4214	0.9912
	$\hat{\beta}$	3.6630	2.2800	3.7055	2.3572	3.7374	2.4209	3.6691	2.4817
	DEF	3.4003	0.4307	3.2932	0.2244	3.2509	0.0841	2.8359	0.0128
Logistic	$\hat{\varphi}$	1.6240	2.0905	1.6468	2.0574	1.6141	2.0280	1.5895	2.0067
	$\hat{\eta}$	1.3990	0.8899	1.3748	0.9306	1.4097	0.9666	1.4377	0.9920
	$\hat{\beta}$	3.6700	2.2350	3.5974	2.3373	3.6547	2.4247	3.7187	2.4812
	DEF	3.5115	0.4342	2.9266	0.2395	2.9031	0.0813	2.8619	0.0124
Piecewise	$\hat{\varphi}$	1.5904	2.0859	1.6399	2.0532	1.6079	2.0299	1.5844	2.0076
	$\hat{\eta}$	1.4326	0.8970	1.3851	0.9373	1.4166	0.9657	1.4434	0.9923
	$\hat{\beta}$	3.7584	2.2552	3.6223	2.3489	3.6765	2.4064	3.7334	2.4825
	DEF	3.8129	0.4712	3.0354	0.1972	2.8273	0.0803	2.7275	0.0123
Singer	$\hat{\varphi}$	1.6229	2.0895	1.6063	2.0535	1.5917	2.0335	1.6273	2.0060
	$\hat{\eta}$	1.3941	0.8897	1.4177	0.9357	1.4335	0.9613	1.3976	0.9929
	$\hat{\beta}$	3.6681	2.2357	3.7348	2.3494	3.7308	2.4078	3.6083	2.4852
	DEF	3.4929	0.4646	3.4148	0.2056	3.2685	0.0836	2.5628	0.0137
Tent	$\hat{\varphi}$	1.6369	2.0992	1.6054	2.0621	1.5945	2.0292	1.5954	2.0052
	$\hat{\eta}$	1.3872	0.8841	1.4222	0.9275	1.4334	0.9669	1.4319	0.9954
	$\hat{\beta}$	3.6555	2.2259	3.7185	2.3277	3.7305	2.4247	3.7015	2.4887
	DEF	3.3924	0.4651	3.2883	0.2476	3.2044	0.0911	2.8916	0.0131
Classic	$\hat{\varphi}$	1.5012	2.0072	1.4345	2.0131	1.4485	2.0224	1.4631	2.0056
	$\hat{\eta}$	1.5284	0.9758	1.5930	0.9780	1.5841	0.9730	1.5680	0.9934
	$\hat{\beta}$	4.0223	2.5454	4.2362	2.4927	4.1390	2.4460	4.0714	2.4888
	DEF	6.3548	1.1379	6.1634	0.3552	5.3215	0.2059	4.6549	0.0169

Table 4. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$ for [0, 1000] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.3929	2.0080	1.3929	2.0034	1.4103	1.9984	1.4266	1.9953
	$\hat{\eta}$	1.6391	0.9811	1.6406	0.9913	1.6196	0.9981	1.6054	1.0047
	$\hat{\beta}$	4.3583	2.4872	4.3198	2.5066	4.2496	2.5181	4.1658	2.5158
	DEF	6.8991	1.0178	6.7133	0.5247	5.9215	0.2543	5.4192	0.0576
Logistic	$\hat{\varphi}$	1.3812	1.9956	1.4099	2.0083	1.3831	2.0102	1.4807	1.9916
	$\hat{\eta}$	1.6499	0.9914	1.6224	0.9849	1.6525	0.9872	1.5487	1.0088
	$\hat{\beta}$	4.4291	2.5337	4.2930	2.4819	4.3064	2.4743	4.0030	2.5271
	DEF	7.4088	0.9969	6.3718	0.5361	6.2110	0.2352	4.6963	0.0639
Piecewise	$\hat{\varphi}$	1.3820	1.9879	1.3815	2.0119	1.4010	1.9981	1.4218	2.0012
	$\hat{\eta}$	1.6526	0.9994	1.6510	0.9827	1.6316	0.9980	1.6107	0.9980
	$\hat{\beta}$	4.4105	2.5686	4.3718	2.4913	4.2658	2.5197	4.1743	2.4945
	DEF	7.2749	1.0571	6.7829	0.4998	5.8385	0.2718	5.3911	0.0496
Singer	$\hat{\varphi}$	1.3818	2.0087	1.4010	2.0121	1.4003	2.0104	1.5214	1.9997
	$\hat{\eta}$	1.6510	0.9735	1.6297	0.9808	1.6317	0.9867	1.5063	0.9995
	$\hat{\beta}$	4.4440	2.4872	4.3185	2.4656	4.2489	2.4834	3.8952	2.5056
	DEF	7.3386	0.9307	6.5529	0.4931	5.9711	0.2094	4.1633	0.0482
Tent	$\hat{\varphi}$	1.3575	2.0030	1.3692	2.0180	1.4359	2.0084	1.4521	2.0049
	$\hat{\eta}$	1.6754	0.9880	1.6652	0.9678	1.5936	0.9886	1.5775	0.9937
	$\hat{\beta}$	4.4975	2.5070	4.3716	2.4457	4.1653	2.4925	4.0952	2.4825
	DEF	7.6653	0.9758	6.7532	0.4966	5.8754	0.2349	5.1867	0.0541
Classic	$\hat{\varphi}$	1.2812	1.7292	1.1727	1.9414	1.2416	2.0022	1.0797	1.7584
	$\hat{\eta}$	1.7531	1.2799	1.8697	1.0596	1.7947	0.9972	1.9628	1.2554
	$\hat{\beta}$	4.6493	3.3789	4.9294	2.7117	4.6893	2.5182	5.1028	3.2034
	DEF	9.7747	3.4803	9.6940	0.7676	8.0893	0.3376	7.9515	0.3066

ERKEN GÖRÜNÜM

Table 5. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,4)$ for [0, 10] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.6718	1.9548	1.6549	1.9390	1.6240	1.8797	1.6044	1.8728
	$\hat{\eta}$	1.3343	1.0378	1.3523	1.0586	1.3832	1.1226	1.4052	1.1304
	$\hat{\beta}$	5.5708	4.2720	5.5954	4.3353	5.7138	4.5840	5.7798	4.5843
	DEF	5.8411	3.0425	5.2064	2.1485	5.1125	1.8088	4.6395	1.1389
Logistic	$\hat{\varphi}$	1.6586	1.9560	1.6848	1.9203	1.6566	1.9053	1.6008	1.8753
	$\hat{\eta}$	1.3476	1.0361	1.3212	1.0791	1.3509	1.0946	1.4093	1.1279
	$\hat{\beta}$	5.6548	4.2599	5.5012	4.4447	5.5776	4.4388	5.7994	4.5673
	DEF	5.6962	2.8310	4.8868	2.2533	4.8205	1.8414	4.5199	1.0765
Piecewise	$\hat{\varphi}$	1.6929	1.9830	1.6194	1.9418	1.6189	1.8889	1.5954	1.8897
	$\hat{\eta}$	1.3099	1.0112	1.3877	1.0545	1.3902	1.1130	1.4146	1.1130
	$\hat{\beta}$	5.5305	4.1843	5.7890	4.3101	5.7401	4.5250	5.8231	4.5022
	DEF	5.7727	2.6630	5.7593	2.1280	5.6268	1.4505	4.6060	1.0521
Singer	$\hat{\varphi}$	1.6641	1.9616	1.6565	1.9620	1.6312	1.9109	1.5985	1.8768
	$\hat{\eta}$	1.3409	1.0315	1.3521	1.0334	1.3763	1.0895	1.4114	1.1265
	$\hat{\beta}$	5.6229	4.2771	5.6254	4.2126	5.6879	4.4333	5.8058	4.5544
	DEF	5.7948	2.7439	5.5091	2.2441	5.4202	1.8676	4.6622	1.1570
Tent	$\hat{\varphi}$	1.6626	1.9656	1.6378	1.9193	1.6296	1.8844	1.5919	1.8658
	$\hat{\eta}$	1.3429	1.0279	1.3690	1.0799	1.3791	1.1148	1.4181	1.1380
	$\hat{\beta}$	5.6015	4.2358	5.6676	4.4159	5.6866	4.5399	5.8433	4.6100
	DEF	5.8708	2.7665	5.6194	2.3020	5.3824	1.8033	4.7056	1.1050
Classic	$\hat{\varphi}$	1.6826	1.9570	1.6778	1.3798	1.6426	1.8920	1.6312	1.8648
	$\hat{\eta}$	1.3238	1.0364	1.3295	1.6303	1.3669	1.1087	1.3777	1.1389
	$\hat{\beta}$	5.5280	4.3033	6.5199	6.8317	5.6322	4.4955	5.6535	4.6133
	DEF	7.2083	3.0901	7.0491	2.3775	6.0118	1.9579	4.8231	1.2078

Table 6. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,4)$ for [0, 1000] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.3236	1.6643	1.3291	1.6061	1.3692	1.6496	1.4077	1.6023
	$\hat{\eta}$	1.6846	1.3363	1.6825	1.4004	1.6417	1.3567	1.6027	1.4069
	$\hat{\beta}$	7.1117	5.5100	6.9950	5.7254	6.7967	5.5431	6.5881	5.7461
	DEF	17.2529	7.2176	15.9181	7.0975	13.6507	6.1004	12.1385	6.0618
Logistic	$\hat{\varphi}$	1.2820	1.6057	1.3469	1.6765	1.3852	1.6303	1.4076	1.6144
	$\hat{\eta}$	1.7274	1.3977	1.6637	1.3298	1.6261	1.3764	1.6033	1.3943
	$\hat{\beta}$	7.2528	5.8264	6.9557	5.4701	6.7149	5.6194	6.5924	5.6890
	DEF	18.7422	8.8173	15.5795	6.4925	13.8643	6.3788	12.0732	5.7528
Piecewise	$\hat{\varphi}$	1.3062	1.6227	1.3302	1.5856	1.3419	1.6433	1.3807	1.6068
	$\hat{\eta}$	1.7058	1.3799	1.6796	1.4206	1.6682	1.3635	1.6309	1.4020
	$\hat{\beta}$	7.1731	5.7225	6.9388	5.8645	6.8858	5.5960	6.7171	5.7347
	DEF	17.7478	8.4142	15.5719	8.1365	15.0310	6.0652	12.9787	5.9513
Singer	$\hat{\varphi}$	1.3747	1.6428	1.3724	1.6459	1.3670	1.6260	1.4328	1.5881
	$\hat{\eta}$	1.6339	1.3577	1.6360	1.3579	1.6428	1.3818	1.5765	1.4212
	$\hat{\beta}$	6.8831	5.6654	6.8066	5.5750	6.7841	5.6358	6.4717	5.8051
	DEF	15.0319	8.4110	14.1942	6.9636	12.9216	6.2800	11.5281	5.3717
Tent	$\hat{\varphi}$	1.3127	1.6368	1.3569	1.6148	1.3781	1.6446	1.4105	1.6126
	$\hat{\eta}$	1.6953	1.3655	1.6522	1.3903	1.6310	1.3626	1.6003	1.3965
	$\hat{\beta}$	7.1634	5.6177	6.8757	5.7197	6.7246	5.5773	6.5884	5.7009
	DEF	18.1195	7.9785	15.1178	7.4795	13.3776	6.0868	11.8466	5.7479
Classic	$\hat{\varphi}$	1.0897	1.5212	1.2803	1.5461	1.1076	1.5262	1.1345	1.5270
	$\hat{\eta}$	1.9230	1.4835	1.7298	1.4613	1.9055	1.4809	1.8787	1.4816
	$\hat{\beta}$	8.3357	6.2819	7.2104	6.0648	7.8952	6.0769	7.7511	6.0684
	DEF	30.3416	12.7171	23.6384	10.1424	23.4816	9.2965	22.0899	8.7277

ERKEN GÖRÜNÜM

Table 7. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,6)$ for [0, 10] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.8910	1.9485	1.8934	1.9490	1.8558	1.9509	1.8178	1.9250
	$\hat{\eta}$	1.1065	1.0518	1.1066	1.0500	1.1448	1.0491	1.1839	1.0756
	$\hat{\beta}$	6.8479	6.5033	6.7717	6.4215	7.0000	6.3598	7.1790	6.4996
	DEF	4.1993	2.5970	3.9275	2.1138	3.8700	1.4851	3.8066	1.1218
Logistic	$\hat{\varphi}$	1.9178	1.9523	1.8833	1.9464	1.8568	1.9527	1.8254	1.9385
	$\hat{\eta}$	1.0804	1.0468	1.1155	1.0527	1.1434	1.0470	1.1764	1.0622
	$\hat{\beta}$	6.7002	6.5136	6.8475	6.4490	6.9643	6.3847	7.1375	6.4039
	DEF	3.8269	2.7223	3.7993	1.9879	3.7734	1.7423	3.6292	1.0100
Piecewise	$\hat{\varphi}$	1.8962	1.9474	1.8804	1.9579	1.8738	1.9489	1.8381	1.9167
	$\hat{\eta}$	1.1034	1.0499	1.1185	1.0409	1.1267	1.0511	1.1635	1.0847
	$\hat{\beta}$	6.8041	6.4888	6.8256	6.3926	6.8476	6.3696	7.0501	6.5500
	DEF	4.0041	2.6037	3.7552	2.0850	3.2757	1.4203	3.0970	1.0386
Singer	$\hat{\varphi}$	1.8705	1.9617	1.8923	1.9609	1.8816	1.9356	1.8329	1.9309
	$\hat{\eta}$	1.1282	1.0360	1.1073	1.0377	1.1193	1.0639	1.1686	1.0698
	$\hat{\beta}$	6.9705	6.4288	6.7624	6.3719	6.8611	6.4629	7.0739	6.4666
	DEF	4.2487	2.4523	3.8283	2.1599	3.5486	1.7462	3.5206	1.1252
Tent	$\hat{\varphi}$	1.9057	1.9535	1.8983	1.9590	1.8649	1.9580	1.8332	1.9273
	$\hat{\eta}$	1.0934	1.0441	1.1014	1.0413	1.1354	1.0415	1.1687	1.0739
	$\hat{\beta}$	6.7593	6.4950	6.7411	6.3656	6.9159	6.3229	7.0765	6.4873
	DEF	4.1972	2.5972	3.9975	2.1277	3.5376	1.6861	3.5086	1.0186
Classic	$\hat{\varphi}$	1.9069	1.9639	1.8934	1.9439	1.8561	1.9546	1.8139	1.9288
	$\hat{\eta}$	1.0920	1.0332	1.1062	1.0567	1.1455	1.0453	1.1879	1.0723
	$\hat{\beta}$	6.9420	6.5088	6.7782	6.4530	6.9740	6.3504	7.2168	6.4699
	DEF	4.8763	3.0517	4.1704	2.3072	3.9843	1.9092	3.9304	1.1920

Table 8. The estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,6)$ for [0, 1000] search space

Inertia weight	Sample size	30		50		100		500	
	Particle size	50	250	50	250	50	250	50	250
Circle	$\hat{\varphi}$	1.3417	1.5354	1.3835	1.5075	1.4108	1.5328	1.3997	1.4831
	$\hat{\eta}$	1.6616	1.4626	1.6179	1.4953	1.5917	1.4698	1.6036	1.5212
	$\hat{\beta}$	10.3448	9.0099	10.0112	9.1571	9.8036	8.9605	9.7870	9.2860
	DEF	34.7145	21.2947	31.5294	20.6933	27.8480	18.9634	26.7104	18.0927
Logistic	$\hat{\varphi}$	1.3428	1.5124	1.3679	1.5165	1.3865	1.4646	1.4056	1.4681
	$\hat{\eta}$	1.6614	1.4878	1.6356	1.4861	1.6166	1.5399	1.5980	1.5362
	$\hat{\beta}$	10.4185	9.1473	10.1438	9.1231	9.9323	9.3982	9.7465	9.3869
	DEF	36.9047	20.4832	31.2717	20.1010	29.1463	19.9832	26.9265	18.8476
Piecewise	$\hat{\varphi}$	1.3465	1.5422	1.3546	1.5169	1.3556	1.4904	1.4027	1.4603
	$\hat{\eta}$	1.6542	1.4583	1.6487	1.4853	1.6479	1.5137	1.6010	1.5441
	$\hat{\beta}$	10.2454	8.9879	10.1666	9.1068	10.0514	9.2426	9.7394	9.4265
	DEF	34.3044	20.0812	32.7664	19.4605	28.7810	19.2940	26.3441	18.8198
Singer	$\hat{\varphi}$	1.3853	1.5106	1.3761	1.5161	1.3410	1.5050	1.5078	1.4920
	$\hat{\eta}$	1.6172	1.4889	1.6264	1.4853	1.6629	1.4989	1.4950	1.5129
	$\hat{\beta}$	10.0517	9.1867	9.9898	9.1026	10.1704	9.1449	9.1260	9.2244
	DEF	32.7567	22.3122	30.4004	19.6672	26.1514	19.0346	20.8142	16.6408
Tent	$\hat{\varphi}$	1.3034	1.5219	1.3330	1.5232	1.4184	1.4760	1.4021	1.4707
	$\hat{\eta}$	1.7131	1.4785	1.6794	1.4782	1.5856	1.5284	1.6019	1.5334
	$\hat{\beta}$	10.6015	9.1233	10.3518	9.0703	9.7537	9.3243	9.7607	9.3585
	DEF	37.7319	21.5028	34.1970	19.8756	28.7571	19.7833	26.1082	19.3158
Classic	$\hat{\varphi}$	1.2515	1.2452	1.2677	1.5039	1.1896	1.2936	1.2637	1.4569
	$\hat{\eta}$	1.7512	1.7612	1.7376	1.4974	1.8141	1.7115	1.7409	1.5462
	$\hat{\beta}$	11.0998	11.4183	11.7184	9.2398	11.1848	10.6498	10.5967	9.4442
	DEF	44.7289	32.2701	40.3034	27.8455	38.0518	27.4710	35.8424	27.0349

ERKEN GÖRÜNÜM

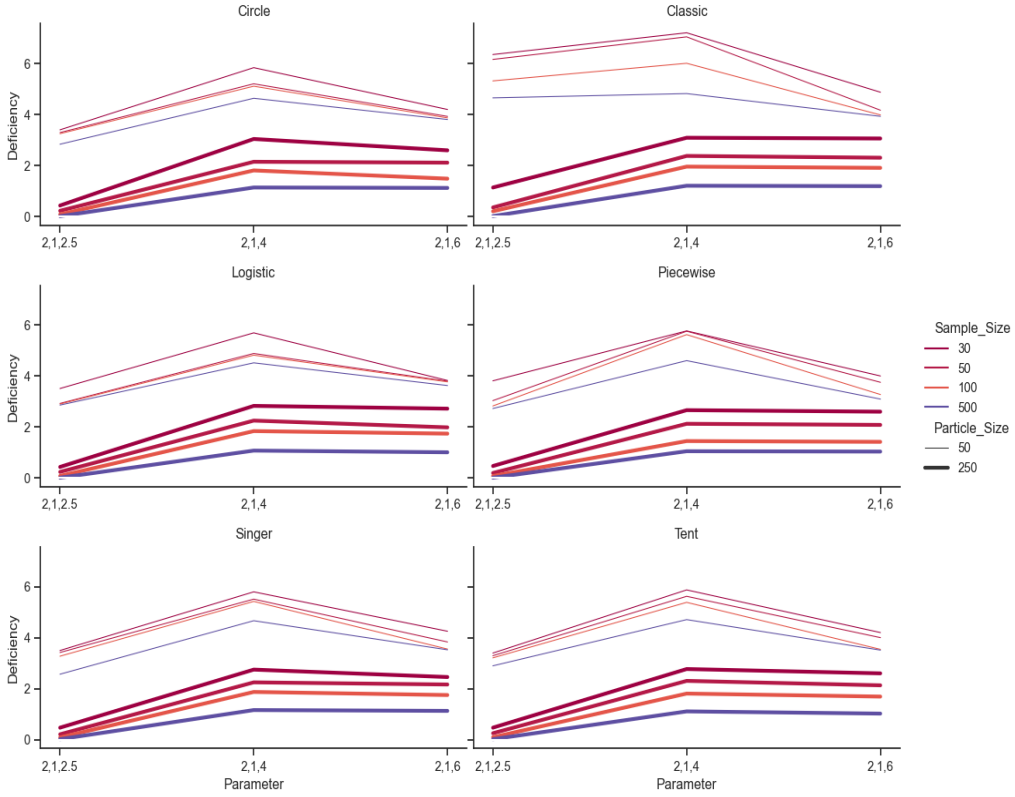


Figure 5. Changing Def values for different parameter values for [0, 10] search space

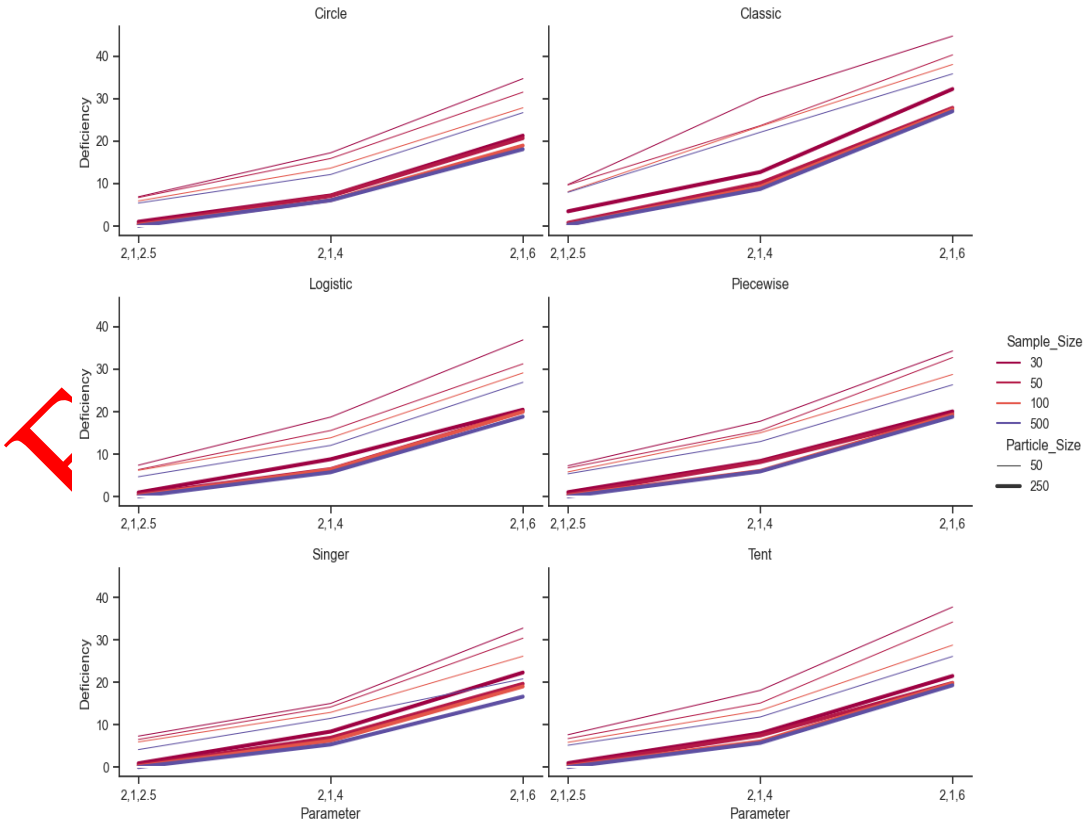


Figure 6. Changing Def values for different parameter values for [0, 1000] search space

In cases where the search space is [0, 10] and the distribution is skewed, i.e. $\beta = 2.5$ and $\beta = 6$, all chaotic maps including the classic method have better Def scores than the symmetric case, i.e. $\beta = 4$. As seen in Figure 5, when the distribution is skewed from symmetry to the left, the Def scores decline more quickly in all instances where the particle size is 50, but not when the particle size is 250. Furthermore, it is observed that the Def score increases as the β value increases for all cases where the search space is [0, 1000], as seen in Figure 6.

When the estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,2.5)$ related to the three-parameter Weibull distribution given in Table 3 and Table 4 are examined in detail, the Singer gives the best Def score in all cases where the particle size is 250 and the search space is [0, 1000]. Except for $n = 30$, the Piecewise also provides the best Def score in all situations where there are 250 particles and [0, 10] search space. In cases where the particle size is 50 and the search spaces are [0, 10] and [0, 1000], the chaotic maps that give the best result of the Def score show similarity. For example, the Circle when the sample size is 30 or the Logistic when it is 50.

For the parameters $\theta = (\gamma, \eta, \beta) = (2,1,4)$ shown in Table 5 and Table 6, when the estimates are examined, the Logistic gives the best Def score when the search space is [0, 10], while the Singer gives it when the search space is [0, 1000] in all cases where the particle size is 50. On the other hand, the Piecewise gives it when the search space is [0, 10], while different chaotic maps give it when the search space is [0, 1000] in all cases where the particle size is 250. Examining the estimates of the parameters $\theta = (\gamma, \eta, \beta) = (2,1,6)$ status presented in Table 7 and Table 8, the Singer provides the best Def score whenever there are 50 particles and [0, 1000] search space. Except for $n = 30$, the Piecewise also provides the best Def score in all situations where there are 50 particles and [0, 10] search space. For cases where the number of particles is 250, different chaotic maps give the best Def results when the search space is both [0, 10] and [0, 1000].

5. CONCLUSIONS AND DISCUSSIONS

The ML method for the parameters of the three-parameter Weibull distribution, encountered in many real-life problems, is discussed in this paper. Conventional optimization methods for finding the solution of likelihood functions do not guarantee to find the parameter values of interest with precision because these functions can have a very complex structure as in the three-parameter Weibull distribution. Therefore, one of the powerful heuristic optimization methods, PSO, is used to find ML estimates of the parameters of the three-parameter Weibull distribution and overcome such computational difficulties.

The originality of the paper is that the performance of the PSO algorithm in terms of inertia weights, particle size and search space size is examined on the three-parameter Weibull parameter estimation, unlike the existing studies in the literature. In addition to all these, five different types of chaotic maps are examined in detail for the inertial weight, which is the factor affecting the convergence of the algorithm.

The simulation results demonstrate that the proposed chaotic map approach gives better results than the classic linear decreasing inertia weights suggested by Shi and Eberhart [25] and is frequently used in PSO studies. On the other hand, a chaotic map type cannot be the dominant one since different chaotic maps give better results in different situations. Although Tent, one of the chaotic maps discussed, gives better results than the classic method, it could not be the best chaotic map in any case.

Better estimates are achieved as the dataset's dimension rises in terms of the Def criterion. Namely, the Def value decreased as the sample size increased while the particle size and search space were the same for the classic method and all chaotic map types.

Considering the results about the skewness of the three-parameter distribution, all chaotic maps including the classic method have a better Def score than the symmetric case when the distribution is skewed and the search space is [0, 10], as seen in Figure 5. Furthermore, for all circumstances where the search space is [0, 1000], it is seen that the Def score rises as the β value rises, as seen in Figure 6.

Future studies could establish a statistical experimental design to assess the PSO algorithm in the three-parameter Weibull parameter estimation, particularly with recommendations for various acceleration coefficients and particle sizes. Furthermore, the chaotic maps that highlight this study can be integrated into different heuristic algorithms, and algorithm performances can be examined in studies related to the three-parameter Weibull or different distributions.

ACKNOWLEDGEMENT

The authors would like to thank Gazi University Academic Writing Application and Research Center for proofreading the article.

DECLARATION OF ETHICAL STANDARDS

The authors of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

AUTHORS' CONTRIBUTIONS

Emre KOÇAK: Determining the methods used in the study, applying solution methods, reporting the results and writing the article.

Büşra AKSEL: Determining the methods used in the study, applying solution methods and reporting the results and writing the article.

ERKEN GÖRÜNÜM

H. Hasan ÖRKCÜ: Determining the methods used in the study, reporting the results and writing the article.

CONFLICT OF INTEREST

There is no conflict of interest in this study.

REFERENCES

[1] Nosal M., Legge A. H. and Krupa S. V., "Application of a stochastic, Weibull probability generator for replacing missing data on ambient concentrations of gaseous pollutants", ***Environmental Pollution***, (2000).

[2] Rousu D. N., "Weibull skewness and kurtosis as a function of the shape parameter", ***Technometrics***, 15(4), (1973).

[3] Cousineau D., "Fitting the three-parameter weibull distribution: Review and evaluation of existing and new methods", ***IEEE Transactions on Dielectrics and Electrical Insulation***, 16(1): 281-288, (2009).

[4] Örkcü H. H., Özsoy V. S., Aksoy E. and Dogan M. I., "Estimating the parameters of 3-p Weibull distribution using particle swarm optimization: A comprehensive experimental comparison", ***Appl Math Comput***, 268, (2015).

[5] Acitas S., Aladag C. H. and Senoglu B., "A new approach for estimating the parameters of Weibull distribution via particle swarm optimization: An application to the strengths of glass fibre data", ***Reliab Eng Syst Saf***, 183, (2019).

[6] Luus R. and Jammer M., "Estimation of parameters in 3-parameter Weibull probability distribution functions", ***Hungarian journal of industrial chemistry***, 33, (2005).

[7] Abbasi B., Eshragh Jahromi A. H., Arkat J. and Hosseinkouchack M., "Estimating the parameters of Weibull distribution using simulated annealing algorithm", ***Appl Math Comput***, 183(1), (2006).

[8] Örkcü H. H., Aksoy E. and Dogan M. I., "Estimating the parameters of 3-p Weibull distribution through differential evolution", ***Appl Math Comput***, 251, (2015).

[9] Kantar Y. M. and Şenoğlu B., "A comparative study for the location and scale parameters of the Weibull distribution with given shape parameter", ***Comput Geosci***, 34(12), (2008).

[10] Akram M. and Hayat A., "Comparison of Estimators of the Weibull Distribution", ***J Stat Theory Pract***, 8(2): 238-259, (2014).

[11] Teimouri M., Hoseini S. M. and Nadarajah S., "Comparison of estimation methods for the Weibull distribution", ***Statistics (Ber)***, 47(1): 93-109, (2013).

[12] Cohen A. C., Whitten B. J. and Ding Y., "Modified moment estimation for the three-parameter Weibull distribution", ***Journal of Quality Technology***, 16(3), (1984).

[13] Chen D. and Chen Z., "Statistical inference about the location parameter of the three-parameter Weibull distribution", ***J Stat Comput Simul***, 79(3), (2009).

[14] Garrido A., Caro-Carretero R., Jimenez-Octavio J. R., Carnicero A. and Such M., "A new approach to fitting the three-parameter Weibull distribution: An application to glass ceramics", ***Commun Stat Theory Methods***, 50(14), (2021).

[15] Yang X., Xie L., Song J., Zhao B. and Li Y., "On interval estimation methods for the location parameter of the Weibull distribution: An application to alloy material fatigue failure data", ***Commun Stat Theory Methods***, 53(17), (2024).

[16] Örgen F. K. D., Altıntaş A., Yaşar S., Öztürk M., Çiftçi E. and Tuncer A. D., "Software-Based Wind Energy Potential Assessment: A Case Study from Western Turkey", ***Politeknik Dergisi***, 26(1), (2023).

[17] Nassir N., Acar B. and Fahed A. K. A., "Vertical Axis Wind Turbine Installation Based on Wind Data Collected in Gharyan City, Libya", ***Politeknik Dergisi***, 25(3), (2022).

[18] Özsoy V. S., Örkcü H. H. and Bal H., "Particle Swarm Optimization Applied to Parameter Estimation of the Four-Parameter Burr III Distribution", ***Iran J Sci Technol Trans A Sci***, 42(2), (2018).

[19] Bard Y., "Comparison of Gradient Methods for the Solution of Nonlinear Parameter Estimation Problems", ***SIAM J Numer Anal***, 7(1), (1970).

[20] Eberhart R. C. and Shi Y., "Particle swarm optimization: Developments, applications and resources", ***Proceedings of the 2001 Congress on Evolutionary Computation***, Seoul, Korea (South), 81-8, (2001).

[21] Wen X., Yu-xia X., Gui-xiang S. and Ying-zhi Z., "Parameter estimation of three-parameter weibull distribution via particle swarm optimization algorithm", ***Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)***, Changchun, China, 336–338, (2011).

[22] Kennedy J. and Eberhart R., "Particle swarm optimization", ***Proceedings of ICNN'95 - IEEE International Conference on Neural Networks***, Perth, WA, Australia, 1942-1948 (1995).

[23] Fukuyama Y., "Fundamentals of Particle Swarm Optimization Techniques", in Modern Heuristic Optimization Techniques: Theory and Applications to Power Systems, ***IEEE***, 71-87, (2008).

[24] Kennedy J., "Particle Swarm Optimization", in Encyclopedia of Machine Learning, ***Boston, MA: Springer US***, (2010).

[25] Shi Y. and Eberhart R., "A modified particle swarm optimizer", ***1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence***, Anchorage, AK, USA, 69-73, (1998).

[26] Tharwat A. and Hassanien A. E., "Chaotic antlion algorithm for parameter optimization of support vector machine", ***Applied Intelligence***, 48(3), (2018).

[27] Alatas B., Akin E. and Ozer A. B., "Chaos embedded particle swarm optimization algorithms", ***Chaos Solitons Fractals***, 40(4), (2009).

[28] Ren B. and Zhong W., "Multi-objective optimization using chaos based PSO", ***Information Technology Journal***, 10(10), (2011).

[29] Wang G. G., Deb S., Gandomi A. H., Zhang Z. and Alavi A. H., "Chaotic cuckoo search", ***Soft comput***, 20(9), (2016).

[30] Arora S. and Anand P., "Chaotic grasshopper optimization algorithm for global optimization", ***Neural Comput Appl***, 31(8), (2019).

ERKEN GÖRÜNÜM

[31] Sayed G. I., Tharwat A. and Hassanien A. E., "Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection", ***Applied Intelligence***, 49(1), (2019).

[32] Sayed G. I., Khoriba G. and Haggag M. H., "A novel chaotic salp swarm algorithm for global optimization and feature selection", ***Applied Intelligence***, 48(10), (2018).

[33] Kohli M. and Arora S., "Chaotic grey wolf optimization algorithm for constrained optimization problems", ***J Comput Des Eng***, 5(4), (2018).

[34] Saremi S., Mirjalili S. and Lewis A., "Biogeography-based optimisation with chaos", ***Neural Comput Appl***, 25(5), (2014).

[35] Tiku M. L. and Akkaya A. D., "Robust estimation and hypothesis testing", ***New Delhi: New Age International***, (2004).

[36] Yang X. S., "Engineering optimization: an introduction with metaheuristic applications", ***John Wiley and Sons***, (2010).

[37] Shi Y. and Eberhart R. C., "Parameter selection in particle swarm optimization", ***Evolutionary Programming VII. Lecture Notes in Computer Science***, Berlin, 591-600, (1998).

ERKEN GÖRÜNÜM