An Efficient Hybrid Algorithm with Particle Swarm Optimization and Nelder-Mead Algorithm for Parameter Estimation of Nonlinear Regression Modeling

Aynur YONAR1,*, Harun YONAR1
1Selçuk University, Department of Statistics, Konya, TURKEY
2Selçuk University, Department of Biostatistics, Konya, TURKEY

Highlights
• An efficient hybrid algorithm is proposed for parameter estimation in nonlinear regression models.
• The initial values problem for the parameters in the NM algorithm is solved through PSO.
• The exploitation capability of the PSO is improved thanks to rapid convergence feature of NM.
• More accurate estimates are obtained via the proposed hybrid algorithm.

Article Info
Received: 19 Jan 2021
Accepted: 13 Jun 2021

Abstract
Nonlinear regression analysis is an important statistical method widely used in many fields of science to model the complex relationships between variables. Therefore, many studies have been conducted to estimate the parameters of nonlinear regression models using various iterative techniques. In this study, an efficient hybrid algorithm, namely PSONM, by combining the exploration capability of Particle Swarm Optimization (PSO) and the exploitation capability of the Nelder-Mead (NM) algorithm is proposed to obtain parameter estimates of nonlinear regression models. To show the performance of the proposed hybrid algorithm, 20 nonlinear regression tasks with various levels of difficulty, and real data sets in the agriculture field have been tested. The experimental results indicated that the suggested hybrid algorithm provides accurate estimates, and its performance is much superior to those of NM and PSO algorithms.

1. INTRODUCTION

Estimation of parameters in nonlinear regression has an important place in terms of statistical inferences. However, estimating the parameters for nonlinear regression models is very difficult because of the complexity of them and this significantly limits their application and development. Model parameters of nonlinear regression models are nonlinear and they cannot be estimated by the ordinary optimization techniques used in parameter estimation linear regression models. Therefore, various numerical algorithms such as the Levenberg-Marquardt, Newton-Raphson, Quasi-Newton, Gauss-Newton, Steepest Descent, Nelder-Mead methods [1], etc. have been developed for estimating the parameters of nonlinear regression modeling. These methods are only effective for some special cases and have several constraints such as derivability, being unimodal, continuity. Moreover, one of the most important problems encountered in these methods is the specification of the initial values for the parameters [2,3]. If a reasonable initial point of parameters is not assigned for these algorithms, especially the NM method, they can easily fall into local optimums or no convergence at all. In order to overcome these problems, metaheuristic methods are applied to examine points in different areas of a search space, thus avoiding local optimum and rapidly converging to the global optimal solution.

There are several studies about the estimation of parameters in nonlinear regression models via different metaheuristic methods such as Artificial Bee Colony (ABC) [4], Differential Evolution (DE) [5], Genetic Algorithm (GA) [6-9], Particle Swarm Optimization (PSO) [10-13], etc. in the literature. These methods have the advantage of being able to escape easily from local optimal to find better solutions but they may
not be able to search well in the local search. In other words, the exploration (finding the promising solutions in the all-region) performances of these methods are quite good, but their exploitation (finding the enhanced solutions in sub-regions) performances are not very well. However, exploration and exploitation are two important opposite criteria that must be well balanced in a metaheuristic method to avoid premature convergence and achieve better performance [14]. Hybridization may be thought of as a tool for generating an effective and efficient searching procedure to deal with the deficiencies of a pure single algorithm [15,16]. Furthermore, NM iterative method can offer robust exploitation capabilities. Therefore, we proposed an algorithm named PSONM by combining the global search/ exploration capability of PSO and the local search/ exploitation capability of the NM algorithm to improve the efficiencies of these algorithms in the estimation of the parameters of nonlinear regression models.

In this study, the PSO method, which is a well-known metaheuristic technique inspired by the foraging behaviour of birds and fishes, is considered since it allows for improved parameter estimation performance with less computational efforts when compared to other metaheuristic methods.

The originality of our study comes from the fact that we use parameter estimation values obtained with the PSO algorithm to define the reasonable initial values for the parameters in the NM algorithm. By combining the advantages of these two methods, we obtained a more efficient algorithm that can both avoid premature convergences and give a good performance to reach the global best solution. To show the performance of the proposed hybrid algorithm, the algorithm is tested on the well-known 20 databases with various levels of difficulty from The National Institute for Standard and Technology (NIST) [17] and real data sets in the agriculture field. In addition, the results of the proposed hybrid algorithm are compared to the results of PSO and NM algorithms.

The rest of the paper is designed as follows. In Section 2, nonlinear regression modeling and least square estimation are presented. In section 3, Nelder -Mead, PSO, and the proposed hybrid methods are introduced. In section 4, comparative results of the experimentation of three algorithms for various nonlinear regression models are given and three different real data sets in the agriculture field are tested. Finally, conclusions are presented in section 5.

2. NONLINEAR REGRESSION MODELING AND LEAST SQUARE ESTIMATION

Nonlinear regression analysis is a method that enables to explain the relationship between a set of variables by constructing a reasonable model containing nonlinear parameters. The general form of nonlinear regression models is given by

$$Y_i = f(X_i; \beta) + \varepsilon_i, \quad i = 1, 2, ..., n \quad (1)$$

where $Y_i$ is a response or dependent variable, $f(X_i; \beta)$ is a nonlinear model function according to at least one of the components of the parameter estimation vector; $X_i = (X_{i1}, X_{i2}, ..., X_{ip})$ is a vector of independent variables; $\beta = (\beta_1, \beta_2, ..., \beta_p)$, $\beta \in \Theta \subset \mathbb{R}^p$ is an unknown parameter vector; $\varepsilon_i$ is a random error term that has similar assumptions in the linear models [18].

The Least Square (LS) method is widely used to estimate the unknown parameter $\beta$ of nonlinear regression and approximate a model fit. The main idea behind of LS method is to minimize the sum of squared errors [19]. Thus, it involves an optimization problem given as follows:

$$\hat{\beta} = \arg \min_{\beta \in \Theta} S(\beta). \quad (2)$$

Here, $\hat{\beta}$ is LS parameters estimates and $S(\beta)$ is the LS function defined as follows:
\[ S(\beta) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - f(x_i, \beta))^2. \]  

(3)

Since \( S(\beta) \) is a nonlinear function, the solution of Equation (2) cannot be made analytically [20,21]. Therefore, iterative methods such as Levenberg-Marquardt, Newton-Raphson, Quasi-Newton, Gauss-Newton, Steepest Descent, Nelder Mead, etc. and/or heuristic algorithms such as GA, PSO, DE, ABC, etc. can be used to obtain parameter estimations to obtain exact parameter values. In this study, we resort to the NM and PSO methods.

3. PARAMETER ESTIMATION ALGORITHMS FOR NONLINEAR REGRESSION MODELING

This section describes the Nelder-Mead, Particle Swarm Optimization, and proposed hybrid PSONM method, briefly.

3.1. Nelder-Mead Algorithm

The Nelder-Mead method introduced by Nelder, Mead [22] is one of the most successful direct search techniques for finding the minimum of a function in multidimensional, unconstrained space. Unlike classical gradient methods, it is derivative-free.

Since NMs working principle is based on simplex, it is also known as the simplex search algorithm. A simplex is determined as a set of points defined as vertices in n dimension. The principle of the algorithm depends on the comparison of the function values by moving the simplex until a minimum is found. The function values at each vertex are evaluated iteratively and the vertex with the highest function value is displaced by another vertex following a new simplex. By the reflection, expansion, contraction, and shrink operators, the current simplex is updated [23].

The algorithm for the NM method can be summarized as following and more details can be obtained in the original paper [22].

i. Determine the NM coefficients such as reflection coefficient, expansion coefficient, contraction coefficient, and shrink coefficient.

ii. Randomly construct an initial n-simplex with \( n+1 \) vertices around the pre-defined starting value and determine their function values at the vertices.

iii. Until the convergence criterion is satisfied, update the current simplex by using the suitable one of reflection, expansion, contraction, or reduction operators. When the convergence is achieved, the point having the lowest value is an optimal minimum.

3.2. Particle Swarm Optimization

Particle Swarm Optimization (PSO) developed by Eberhart, Kennedy [24] is based on swarm behaviour, such as bird flocking and fish schooling to find a place with enough food in nature. PSO has been extensively thought in various applied studies due to its easy implementation, high exactness, and fast convergence [25,26].

PSO is a population-based metaheuristic algorithm that has a self-controlling mechanism. The population is formed a set of candidate solutions called particles. It randomly generates a population and assigns a random velocity to them. Each of the particles memorizes its own best position (\( p_{best} \)) and the best position (\( g_{best} \)) recorded by particles in the population so far. Until a global optimal position is obtained, all particles update their velocity (\( v_i \)) and position (\( x_i \)) at each iteration given as:

\[
v_i^{t+1} = w v_i^t + c_1 r_1 (p_{best}^t - x_i^t) + c_2 r_2 (g_{best}^t - x_i^t) \tag{4}
\]
and

\[ x^{t+1}_i = x^t_i + v^{t+1}_i, \]  

respectively. Here, \( v^t_i \) is the velocity of individual \( i \) at iteration \( t \), \( w \) is the inertia weight, \( c_1 \) and \( c_2 \) are two constant, called learning parameters or acceleration coefficient, \( r_1 \) and \( r_2 \) are the uniformly distributed random numbers with range \([0,1]\). \( x^t_i \) is the position of particle \( i \) at iteration \( t \), \( pbest^t_i \) is the best position of individual \( i \) until iteration \( t \), \( gbest^t \) is the best position of the population until iteration \( t \) [23,27].

There are three parts for updating the velocity in Equation (4). The first part \( w v^t_i \) models the tendency of a particle to stay in the same direction and is referred to as "inertia", "habit" or "momentum". The second part \( c_1 r_1 (pbest^t_i - x^t_i) \) is the linear tendency of the particle towards its own best position \( (pbest^t_i) \) with an arbitrary weight \( (c_1 r_1) \). This part is called "memory", "nostalgia" or "self-knowledge". The third part \( c_2 r_2 (gbest^t - x^t_i) \) is a measure of the linear tendency of the particle towards to the best position of all particles \( (gbest^t) \) with an arbitrary weight \( (c_2 r_2) \). This term is called "collaboration", "shared knowledge" or "social knowledge" [28,29].

### 3.3. Proposed Hybrid Method

Considering the fact that the NM algorithm is for local optimization, it works best when the initial values that are pretty close to the global optimum in the search space are chosen. Hence, its performance in reaching the global solution depends heavily on the initial values, but there is a gap in the literature regarding how the best way to choose or determine the initial values. PSO algorithm can be used to determine the staring values of NM. PSO method resists easily trapped in the local best solution thanks to a good exploration ability of it, but its exploitation to search for improved solutions in local areas is low. To eliminate these drawbacks of the two algorithms, we proposed a new hybrid algorithm named PSONM integrating the two algorithms. The flowchart of the PSONM algorithm is given in Figure 1.

The advantages of using the proposed PSONM is that (i) the initial values of the parameters in NM are accurately determined via PSO so that the algorithm does not fall into local optimum and (ii) the exploitation capability of the PSO algorithm is improved thanks to the potential for rapid convergence of NM. As a result, it is possible to obtain an efficient algorithm that can both avoid premature convergences and give a good performance by integrating the advantages of the two algorithms.
4. EXPERIMENTAL RESULTS

This section consists of two parts. In the first part, the nonlinear regression tasks taken from NIST are used. In the second part, real data in the agriculture field is analyzed.

4.1. Results for NIST Nonlinear Regression Tasks

In this part, we have considered well-known 20 nonlinear regression tasks taken from NIST whose list is given in Table 1 to demonstrate the performance of the proposed PSONM algorithm for the nonlinear regression parameter estimation. These nonlinear regression tasks are categorized according to their level of difficulty into three levels: lower (task 1–7), average (task 8–12), and higher (task 13–20). Furthermore, each task has different characteristics in terms of its model structure or its data set. In this study, exponential or rational types of nonlinear regression models are considered. The number of their parameters varies between 2 and 9, and the number of observations ranges from 6 to 250.
### Table 1. List of the Nonlinear Regression Models

| Dataset name | Difficulty Levels | Model Form | Numb. of Param. | Numb. of Obs. |
|--------------|-------------------|------------|----------------|--------------|
| Misra1a      | Lower             | \( y = \beta_0 \left[ 1 - \exp(-\beta_1 x) \right] + \epsilon \) | 2 | 14 |
| Chwirut2     | Lower             | \( y = \frac{\exp(-\beta_1 x)}{\beta_1 + \beta_2 x} + \epsilon \) | 3 | 54 |
| Chwirut1     | Lower             | \( y = \frac{\exp(-\beta_1 x)}{\beta_1 + \beta_2 x} + \epsilon \) | 3 | 214 |
| Gauss1       | Lower             | \( y = \beta_0 \exp(-\beta_1 x) + \beta_2 \exp \left[ \frac{(x - \beta_3)^2}{\beta_4^2} \right] + \beta_3 \exp \left[ \frac{(x - \beta_5)^2}{\beta_6^2} \right] + \epsilon \) | 8 | 250 |
| Gauss2       | Lower             | \( y = \beta_0 \exp(-\beta_1 x) + \beta_2 \exp \left[ \frac{(x - \beta_3)^2}{\beta_4^2} \right] + \beta_3 \exp \left[ \frac{(x - \beta_5)^2}{\beta_6^2} \right] + \epsilon \) | 8 | 250 |
| DanWood      | Lower             | \( y = \beta_0 x^n + \epsilon \) | 6 | |
| Misra1b      | Lower             | \( y = \beta_0 \left[ 1 - \frac{1}{(1 + \beta_1 x / 2)^2} \right] + \epsilon \) | 2 | 14 |
| Kirby2       | Average           | \( y = \frac{\beta_0 + \beta_1 x + \beta_2 x^2}{1 + \beta_3 x + \beta_4 x^2} + \epsilon \) | 5 | 151 |
| Gauss3       | Average           | \( y = \beta_0 \exp(-\beta_1 x) + \beta_2 \exp \left[ \frac{(x - \beta_3)^2}{\beta_4^2} \right] + \beta_3 \exp \left[ \frac{(x - \beta_5)^2}{\beta_6^2} \right] + \epsilon \) | 8 | 250 |
| Misra1c      | Average           | \( y = \beta_0 \left[ 1 - \frac{1}{(1 + 2 \beta_1 x)^{1/2}} \right] + \epsilon \) | 2 | 14 |
| Misra1d      | Average           | \( y = \frac{\beta_0 \beta_2 x}{1 + \beta_1 x} + \epsilon \) | 2 | 14 |
| Roszman1     | Average           | \( y = \beta_0 + \beta_1 x - \frac{\arctan \left( \frac{\beta_2}{x - \beta_3} \right)}{\pi} + \epsilon \) | 4 | 25 |
| ENSO         | Average           | \( y = \beta_0 \sin(2\pi x / \beta_1) + \beta_1 \cos(2\pi x / \beta_1) \) | 9 | 168 |
| MGH09        | Higher            | \( y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \epsilon \) | 4 | 11 |
| Thumber      | Higher            | \( y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \epsilon \) | 7 | 37 |
| BoxBOD       | Higher            | \( y = \beta_0 \left[ 1 - \exp(-\beta_1 x) \right] + \epsilon \) | 2 | 6 |
| Rat42        | Higher            | \( y = \frac{\beta_0}{1 + \exp(\beta_1 - \beta_2 x)} + \epsilon \) | 3 | 9 |
| Eckerle4     | Higher            | \( y = \frac{\beta_0 \exp \left[ -\frac{(x - \beta_1)^2}{2\beta_2^2} \right]}{1 + \exp(\beta_1 - \beta_2 x)} + \epsilon \) | 3 | 35 |
| Rat43        | Higher            | \( y = \frac{\beta_0}{\left[ 1 + \exp(\beta_1 - \beta_2 x) \right]^\beta_3} + \epsilon \) | 4 | 15 |
| Bennett5     | Higher            | \( y = \frac{\beta_0}{(x + \beta_1)^\beta_3} + \epsilon \) | 3 | 154 |
We proposed the PSONM algorithm for estimating the parameters of the nonlinear regression models given in Table 1 in this study. The PSO algorithm parameters: inertia weight $w$, acceleration coefficients $c_1$ and $c_2$ are selected as 0.7, 2 and 2, respectively, which are the most frequently used values in the literature. The number of particles is considered to be 20 or 50 according to the structure of the nonlinear models and the maximum iteration number is taken to be 1000. Each experiment is replicated 100 times for examining the robustness of the estimation strategy. All computations in this section are conducted in R 4.0.3.

The parameter estimation values $\hat{\beta}$ and the estimated sum of squared error function values $s(\hat{\beta})$ of the NM, PSO, and HPSONM algorithms for nonlinear regression models are reported in Table 2. Parameter searching spaces used in PSO ve PSONM algorithms are also given in this table. It should be stated that we used randomly generated numbers in these search spaces for the parameter initial values of the NM algorithm in this study. Moreover, Table 2 includes certified parameter values $\beta$ and the sum of squared error function values $s(\beta)$, taken from NIST [17].

The closeness of the estimated parameter value to the certified parameter value and the estimated sum of the squared error function values to their certified value are utilized to compare the algorithms in this study. In Table 2, the best results are highlighted in bold.

### Table 2. The parameter estimation results of the NM, PSO, and PSONM algorithms for nonlinear regression models

| Dataset name | Searching Space (min-max) | Certified Parameter Values (\(\beta\) and \(s(\beta)\)) | Estimated Parameter values with NM (\(\hat{\beta}\) and \(s(\hat{\beta})\)) | Estimated Parameter values with PSO (\(\hat{\beta}\) and \(s(\hat{\beta})\)) | Estimated Parameter values with PSONM (\(\hat{\beta}\) and \(s(\hat{\beta})\)) |
|--------------|--------------------------|---------------------------------------------------------|-------------------------------------------------------------------------|------------------------------------------------------------------------|------------------------------------------------------------------------|
| Misra1a      | 0-1000 0-100 0-1000 0-100 | 238.9421 0.000550 0.124551 43.34071 30.07031 6761.788 | 238.9437 0.000550 0.124551 | 238.9421 0.000550 0.124551 | 238.9421 0.000550 0.124551 |
| Chwirut2     | 0-100 0-100 0-100 0-100 | 0.166576 0.005165 0.012150 -1.271261 -18.80249 43.34071 | 0.166574 0.005165 0.012150 | 0.166574 0.005165 0.012150 | 0.166574 0.005165 0.012150 |
| Chwirut1     | 0-100 0-100 0-100 0-100 | 0.190273 0.006131 0.010531 -1.26187 -13.59133 2384.477 | 0.190273 0.006131 0.010531 | 0.190273 0.006131 0.010531 | 0.190273 0.006131 0.010531 |
| Gauss1       | 0-1000 0-100 0-1000 0-1000 0-1000 0-1000 | 98.77821 0.010497 0.010531 10.3129 27.23047 2384.477 | 98.77821 0.010497 0.010531 | 98.77821 0.010497 0.010531 | 98.77821 0.010497 0.010531 |
| Gauss2       | 0-1000 0-100 0-1000 0-1000 0-1000 0-1000 | 99.01832 0.010531 0.010531 10.3129 27.23047 2384.477 | 99.01832 0.010531 0.010531 | 99.01832 0.010531 0.010531 | 99.01832 0.010531 0.010531 |

Note: The parameter estimation results are highlighted in bold for the best results.
|        | 0-100  | 0-100  | 0-100  | 0-100  | 0-100  |
|--------|--------|--------|--------|--------|--------|
|        | 107.0309 | 1275.298 | 103.1903 | 107.0310 |        |
|        | 23.57858 | 191.1002 | 14.27156 | 23.57860 |        |
|        | 72.04558 | 13.07832 | 79.94281 | 72.04555 |        |
|        | 153.2701 | 252.7907 | 128.4810 | 153.2701 |        |
|        | 19.52597  | 0.527467 | 54.89770 | 19.52598 |        |
|        | 1247.528 | 239946  | 13842.35 | 1247.528 |        |
| DanWood| 0-100  | 0.768862 | 0.7688598 | 0.7688623 | 0.768862 |
|        | 0-100  | 3.860405 | 3.860419 | 3.8604060 | 3.860406 |
|        |        | 0.004317 | 0.004317 | 0.004317 | 0.004317 |
| Misra1b| 0-1000| 337.9974 | 48.95245 | 337.9979 | 337.9975 |
|        | 0-100  | 0.000390 | -0.05206262 | 0.000390 | 0.000390 |
|        |        | 0.075464 | 4465.895 | 0.075464 | 0.075464 |
| Kirby2 | 0-100  | 1.674506 | 18.46072 | 10.22254 | 1.674519 |
|        | -100-100| -0.139273 | -1.260945 | -0.718547 | 0.139274 |
|        | 0-100  | 0.00259611 | 0.016312 | 0.002982 | 0.0025961 |
|        | -100-100| -0.00174181 | 0.0218917 | 0.009081 | -0.001724 |
|        | 0-1    | 2.1664E-05 | 7.212E-05 | 4.755E-05 | 2.1664E-05 |
|        |        | 3.9050739 | 354.2576 | 198.3985 | 3.905074 |
| Gauss3 | 0-1000 | 98.94036 | 11.427.00 | 101.9710 | 98.94038 |
|        | 0-100  | 0.010945 | 36.32227 | 0.012410 | 0.010945 |
|        | 0-1000 | 100.6955 | -2.58859 | 35.36661 | 100.6956 |
|        | 0-100  | 111.6361 | 687.1507 | 103.8336 | 111.6362 |
|        | 0-1000 | 23.3050 | 25.99509 | 13.71012 | 23.3052 |
|        | 0-1000 | 23.3050 | 25.99509 | 13.71012 | 23.3052 |
|        | 0-1000 | 73.70503 | 99.16669 | 100.0000 | 73.70504 |
|        | 0-1000 | 147.7161 | 95.58334 | 129.5960 | 147.7161 |
|        | 0-1000 | 19.66822 | 95.74653 | 36.56456 | 19.66820 |
|        |        | 334.485 | 165720 | 4893.655 | 1244.485 |
| Misra1c| 0-1000 | 636.4272 | 636.4273 | 636.4274 | 636.4273 |
|        | 0-100  | 0.000208 | 0.000208 | 0.000208 | 0.000208 |
|        |        | 0.040966 | 0.040966 | 0.040966 | 0.040966 |
| Misra1d| 0-1000 | 437.3697 | 437.3697 | 437.3706 | 437.3697 |
|        | 0-100  | 0.000302 | 0.000302 | 0.000302 | 0.000302 |
|        |        | 0.056419 | 0.056419 | 0.056419 | 0.056419 |
| Rosman1| 0-100  | 0.201968 | 0.312769 | 0.2015996 | 0.2019687 |
|        | -100-100| -6.19E-06 | -3.34E-05 | -6.14E-06 | -6.19E-06 |
|        | 0-104  | 1204.455 | 1364.074 | 1206.0310 | 1204.4560 |
|        | 100-1000| -181.342 | 614.7845 | -182.1643 | -181.3427 |
|        |        | 0.000494 | 0.0009866 | 0.000494 | 0.000494 |
| ENSO   | 0-100  | 10.51074 | 11.19584 | 10.51343 | 10.51075 |
|        | 0-100  | 3.076212 | 3.044512 | 3.089324 | 3.076205 |
|        | 0-100  | 0.532801 | 0.477053 | 0.542223 | 0.532815 |
|        | 0-100  | 44.31108 | 269.0700 | 44.41431 | 44.31119 |
|        | -100-100| -1.62314 | -118.287 | -1.61903 | -1.62313 |
|        | 0-100  | 0.525544 | -6.62352 | 0.577107 | 0.525580 |
|        | 0-100  | 26.88761 | 269.9465 | 26.90461 | 26.88762 |
|        | 0-100  | 0.212322 | 118.4061 | 0.233057 | 0.212324 |
It is obvious from Table 2 that PSO and PSONM outperform NM for almost all cases. This is because NM is an algorithm with an initial value problem for the parameters, and the random generation of parameter initial values of the NM in a wide search space has caused this algorithm to give unreliable results. Although PSO and PSONM generally give similar results, it is observed that PSONM also gives much better results than PSO in cases where the number of unknown parameters is high, such as Gauss2, Kirby2, Gauss 3, Thurber, and Rat43 data sets. This shows that PSO alone is not sufficient, especially for situations with a large number of unknown parameters. It is also clear that PSONM gives the closest results to certified parameter values for most cases. This result shows that PSONM is considerably superior to both NM and PSO. Therefore, we can strongly advise using the proposed PSONM algorithm to estimate the parameters of non-linear regression models.

|     | 0-100 | 0-10   | 0-10   | 0-10   |
|-----|-------|--------|--------|--------|
|     | 1.496687 | 4.886753 | 1.482518 | 1.496692 |
|     | 788.5398 | 1126.59 | 788.645 | 788.5398 |
| MGH09 |        |        |        |        |
| 0-10 | 0.192806 | 0.037518 | 0.1867338 | 0.193040 |
| 0-10 | 0.191282 | 51.15052 | 0.3975372 | 0.193964 |
| 0-10 | 0.123056 | 5.819366 | 0.1943043 | 0.123532 |
| 0-10 | 0.136062 | 4.000618 | 0.2221072 | 0.137675 |
|     | 0.000300 | 0.000916 | 0.0003388 | 0.000300 |
|     |        |        |        |        |
| Thurber |        |        |        |        |
| 0-10000 | 1288.139 | 1285.10 | 1321.34 | 1288.140 |
| 0-10000 | 1491.079 | 1937.134 | 1275.078 | 1491.079 |
| 0-100 | 583.2383 | 13055.86 | 565.4782 | 583.2385 |
| 0-100 | 75.41664 | 2385.759 | 90.28131 | 75.41667 |
| 0-100 | 0.966295 | 14.75594 | 0.830795 | 0.966295 |
| 0-100 | 0.397972 | 6.740311 | 0.502413 | 0.397972 |
| 0-100 | 0.049727 | 2.467610 | 0.000000 | 0.049727 |
|     | 5642.708 | 22287.44 | 41986.13 | 5642.708 |
|     |        |        |        |        |
| BoxBOD |        |        |        |        |
| 0-1000 | 213.8094 | 213.8094 | 213.8094 | 213.8094 |
| 0-100 | 0.547237 | 0.547237 | 0.547237 | 0.547237 |
|     | 1168.008 | 1168.009 | 1168.009 | 1168.009 |
|     |        |        |        |        |
| Rat42 |        |        |        |        |
| 0-100 | 72.46223 | 28.83778 | 72.46224 | 72.46224 |
| 0-100 | 2.618076 | -13725.1 | 2.618077 | 2.618077 |
| 0-100 | 0.067359 | 13092.13 | 0.067359 | 0.067359 |
|     | 8.056522 | 4648.063 | 8.056523 | 8.056523 |
|     |        |        |        |        |
| Eckerle4 |        |        |        |        |
| 0-100 | 1.554382 | 1.554383 | 1.554383 | 1.554383 |
| 0-100 | 4.088832 | 4.088832 | 4.088832 | 4.088832 |
| 0-1000 | 451.5412 | 451.5412 | 451.5412 | 451.5412 |
|     | 0.001463 | 0.001463 | 0.001463 | 0.001463 |
|     |        |        |        |        |
| Rat43 |        |        |        |        |
| 0-1000 | 699.8645 | 423.2954 | 716.3472 | 699.6415 |
| 0-100 | 5.277125 | 25.21918 | 12.71113 | 5.277124 |
| 0-100 | 0.759629 | 89.6269 | 1.439589 | 0.759629 |
| 0-100 | 1.294248 | 57.84976 | 4.098929 | 1.279249 |
|     | 8786.404 | 107646.2 | 17865.44 | 8786.405 |
|     |        |        |        |        |
| Bennett5 |        |        |        |        |
| -1014-101 | -2523.505 | -36.1220 | -2841.876 | -2790.995 |
| 0-100 | 46.73656 | 101.4423 | 47.95140 | 47.79961 |
| 0-100 | 0.932184 | 43.03051 | 0.912065 | 0.915187 |
|     | 0.000524 | 48.26545 | 0.0005614 | 0.000526 |
4.2. Results for Real Data Sets

The meanings of the parameters in the model are generally mixed, but in some models, they may be quite clear. Therefore, a person can define the starting values of the parameters based on their knowledge of the subject. Desta et al. [30] used various nonlinear regression models such as Monomolecular, Mitcherlich, Gompertz, Logistic, Chapman–Richards, Richard’s, and Weibull model to model the relationship between top height and age with the top height growth (THG) data, collected by the Bowmont Norway Spruce Thinning Experiment (1930–1974) from sample plot 3661. Mahanta and Borah [31] discussed nonlinear Weibull Growth models for mean diameter at breast height (MDBH) data and average height growth (AHG) data. Using the meanings of the parameters in these models, they determined the initial values for the parameters to be used in the iterative method and thus were able to obtain parameter estimates for the nonlinear regression models they deal with.

In this part, we utilized the data sets of the THG, MDBH and AHG given in Tables 3-5 respectively to show the performance of the proposed PSONM model to estimate the parameters of the nonlinear regression model. Estimated parameter values $\hat{\beta}$ and estimated the sum of square error function values $s(\hat{\beta})$ by Desta et al. [30] or Mahanta and Borah [31], and proposed PSONM algorithm are given in Table 6.

| Data set name | Model name | Model form | Estimated parameter values by Desta et al. [30] or Mahanta and Borah [31] $(\hat{\beta})$ and $s(\hat{\beta})$ |
|---------------|------------|------------|--------------------------------------------------------------------------------------------------|
| THG           | Monomolecular | $y = \beta_0 \left[ 1 - \beta_1 \exp(-\beta_2 x) \right] + \epsilon$ | 34.48863, 1.05406148, 0.01438658; 34.48862, 1.054062, 0.0143866 |
|               | Mitcherlich | $y = \left[ \beta_0 - \beta_1 \beta_2^x \right] + \epsilon$ | 34.48863073, 36.35313733, 0.98571641; 34.48861, 36.35312, 0.9857164 |
|               | Gompertz    | $y = \beta_0 \exp(-\beta_1 \exp(-\beta_2 x)) + \epsilon$ | 25.67476848, 0.03626337; 25.48479, 0.03677709 |
It is clear from Table 6 that the two methods have same $\hat{\beta}$ and $s(\hat{\beta})$ values for THG data set except for Gompertz and Richard’s models. PSONM provides better estimates with lower $s(\hat{\beta})$ values for these models. Furthermore, PSONM gives much better parameter estimates with lower $s(\hat{\beta})$ values for MDBH and AHG data sets. Consequently, since it is not always possible to know the meanings of the parameters in the model, initial values may not be determined in this way used by Desta et al. [30] or Mahanta and
Borah [31] for most problems. Thus, it can be said that the proposed PSONM is more useful than the methods of Desta et al. [30] or Mahanta and Borah [31] because PSONM can be used in all situations.

5. CONCLUSION

In this study, the problem of estimating the parameters of nonlinear regression models is examined. The ordinary least squares method is the most popular method used to estimate the unknown parameters of the nonlinear regression function. This method is based on minimizing the sum of squares of error terms and hence involves an optimization problem. However, this problem cannot be solved by classical methods since the objective function is nonlinear, and therefore different iterative and/or metaheuristic algorithms have been used. In this study, we proposed a hybrid PSONM algorithm using PSO, which is a well-known metaheuristic technique, and NM iterative algorithms to estimate the parameters of the nonlinear regression model. An efficient algorithm is procured as expected by combining the exploration capability of Particle Swarm Optimization (PSO) and the exploitation capability of the Nelder-Mead (NM) algorithm. At the same time, the parameter initial value problem of the NM algorithm is solved. In order to demonstrate the performance of the proposed hybrid algorithm, 20 nonlinear regression tasks with various levels of difficulty and various real data in the agriculture field have been analyzed. The experimental results showed that the proposed hybrid PSONM algorithm provides accurate estimates and its performance is higher than those of NM and PSO algorithms.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Nash, J.C. and Walker-Smith, M., Nonlinear parameter estimation: An integrated system on BASIC. Marcel Dekker, New York, (1987).

[2] Křivý, I. and Tvrdík, J., Krpec, R., “Stochastic algorithms in nonlinear regression”, Computational Statistics Data Analysis, 33(3): 277–290, (2000).

[3] Yonar, A., Yarıcı Pehlivan, N., “A novel differential evolution algorithm approach for estimating the parameters of Gamma distribution: An application to the failure stresses of single carbon fibres”, Hacettepe Journal of Mathematics and Statistics, 49(4): 1493–1514, (2020).

[4] De los Cobos Silva, S.G., Andrade, M.A.G., García, E.A.R., Velázquez, P.L., Cornejo, M.A., “Estimación de parámetros de regresión no lineal mediante colonia de abejas artificiales”, Revista de Matemática: Teoría y Aplicaciones, 20(1): 49–60, (2013).

[5] Tvrdík, J., “Adaptation in differential evolution: A numerical comparison”, Applied Soft Computing, 9(3): 1449–1455, (2009).

[6] Kapanoğlu, M., Ozan Koc I., and Erdogmus, S., “Genetic algorithms in parameter estimation for nonlinear regression models: an experimental approach”, Journal of Statistical Computation Simulation, 77(10): 851–867, (2007).

[7] Chen, J., “A New Hybrid Genetic Algorithm for Parameter Estimation of Nonlinear Regression Modeling”, Proceedings of the 15th International Conference on Man–Machine–Environment System Engineering, 261–266, (2015).

[8] Altunkaynak, B. and Alptekin, E., “The genetic algorithm method for parameter estimation in nonlinear regression”, Gazi University Journal of Science, 17(2): 43–51, (2004).
[9] Karr, C.L., Weck, B., Massart, D.-L., and Vankeerberghen, P., “Least median squares curve fitting using a genetic algorithm”, Engineering Applications of Artificial Intelligence 8(2): 177–189, (1995).

[10] De-los-Cobos-Silva, S., Terceno-Gomez, A., Gutiérrez-Andrade, M., Rincon-Garcia, E., Lara-Velázquez, P., Aguilar-Cornejo, M., “Particle Swarm Optimization: An Alternative for Parameter Estimation in Regression”, Fuzzy Economic Review, 18(2): (2013).

[11] Cheng, S., Zhao, C., Wu, J., Shi, Y., “Particle swarm optimization in regression analysis: a case study”, International Conference in Swarm Intelligence, 55–63, (2013).

[12] Schwaab, M., Biscaia Jr, E.C., Monteiro, J.L., and Pinto, J.C., “Nonlinear parameter estimation through particle swarm optimization”, Chemical Engineering Science, 63(6): 1542–1552, (2008).

[13] Özsoy, V.S., Örkçü, H.H., “Estimating the Parameters of Nonlinear Regression Model Through Particle Swarm Optimization”, Gazi University Journal of Science, 29(1): (2016).

[14] Yonar, A., Yapıçı Pehlivan, N., “Artificial bee colony with levy flights for parameter estimation of 3-p Weibull distribution”, Iranian Journal of Science and Technology, Transactions A: Science, 44: 851–864, (2020).

[15] Wang, L., Intelligent optimization algorithms with applications. Tsinghua University Springer Press, Beijing, (2001).

[16] Li, L.-l., Wang, L., Liu, L.-h., “An effective hybrid PSOSA strategy for optimization and its application to parameter estimation”, Applied Mathematics Computation, 179(1): 135–146 (2006).

[17] NIST: The National Institute for Standard and Technology. https://www.itl.nist.gov/div898/strd/nls/nls_main.shtml. Access date: 07.10.2020.

[18] Seber, G.A., Wild, C.J., Nonlinear Regression. New Jersey: John Wiley Sons, 62–63, (2003).

[19] Neter, J., Kutner, M.H., Nachtsheim, C.I., Wasserman, W., Applied linear statistical models, WCB McGraw-Hill, (1996).

[20] Bates, D.M., Watts, D.O., Nonlinear regression analysis and its applications, Wiley New York, (1988).

[21] Gallant, A.R., “Nonlinear regression”, The American Statistician, 29(2): 73–81, (1975).

[22] Nelder, J.A., Mead, R., “A simplex method for function minimization”, The Computer Journal, 7(4): 308–313, (1965).

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

[24] Eberhart, R., Kennedy, J., “Particle swarm optimization”, Proceedings of the IEEE international conference on neural networks, 1942–1948, (1995).

[25] Acitas, S., Aladag, C.H., 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”, Reliability Engineering System Safety, 183: 116–127, (2019).
[26] Örkcü, H.H., Özsoy, V.S., Aksoy, E., Dogan, M.I., “Estimating the parameters of 3-p Weibull distribution using particle swarm optimization: A comprehensive experimental comparison”, Applied Mathematics Computational Statistics Data Analysis, 268: 201–226, (2015).

[27] Talbi, E.-G., Metaheuristics: from design to implementation, vol. 74. John Wiley and Sons, (2009).

[28] Rezaee Jordehi, A., Jasni, J., “Parameter selection in particle swarm optimization: a survey”, Journal of Experimental Theoretical Artificial Intelligence, 25(4): 527–542, (2013).

[29] Yonar, A., “Metaheuristic approaches for estimating parameters of univariate and multivariate distributions”, Phd. Thesis, Selçuk University Institute of Science, Konya, 48–52, (2020).

[30] Desta, F., Mac Siurtain, M., and Colbert, J., “Parameter estimation of nonlinear growth models in forestry”, Silva Fennica, 33(4): 327–336, (1999).

[31] Mahanta, D.J., Borah, M., “Parameter Estimation of Weibull Growth Models in Forestry”, International Journal of Mathematics Trends and Technology, 8(3): 157–163, (2014).