A New Algorithm to Estimate the Parameters of Nonlinear Regression

Zaid Adil Abd alkreem and Bayda Atiya Kalaf

Department of Mathematics, College of Education (Ibn Al – Haitham), University of Baghdad, Iraq

E-mail: baydaa.a.k@ihcoedu.uobaghdad.edu.iq

Abstract. The procedures to estimate the parameters are important in many scientific fields that are required to develop mathematical models. Thus, this paper is proposed as a Gravitational Search algorithm for estimating the parameters of nonlinear regression models. Also, a simulation study is conducted to investigate the performance of the proposed methods in this paper. The results show that GSA approach provides accurate estimates and is satisfactory for the parameter estimation of the nonlinear regression models.

1. Introduction
Nonlinear Regression Analysis (NLRA) was one of the most widely used accurate statistical steps that explain the relationship between two variables or more [1]. The common form of a regression model is $y = f(x, \beta) + \varepsilon$. Which $y$ is the dependent variable, $x$ is an independent vector variable(s), $\beta$ is a vector of the parameter(s), and $\varepsilon$ is the error factor usually supposed to be uncorrelated with mean zero and constant difference. In the parameter estimation problem, the form of the nonlinear regression function is known but it contains unknown parameters $\beta_1, \ldots, \beta_p$.

There are a large number of articles on how to estimates the parameter of nonlinear regression models. Aşıkgil and Erar [2] examined the nonlinear parameter estimation efficiency under the issue of auto reconditioned errors. The most and commonly used algorithm Gauss-Newton method (also called the Newton-Raphson method) [3]. However, the nonlinearity model produces a hard estimation of parameters and creates a very difficult and challenging statistical analysis of parameter estimates. In addition, it is not considered an easy controlled by practitioners and need much more detailed information to work properly. These difficulties are arising because of an increased number of parameters and the multi-conditioned nature of the function of the objective. Nonlinear regression models. Michailidis[4]considered Jaya's optimization algorithm for estimating nonlinear metaheuristic algorithm named Jaya .then tested it on a set of benchmark regression problems. Tvrd’ik and K’riv’y [5] used some stochastic algorithms to solve the issue of global optimization of nonlinear regression models. These algorithms were applied to estimate the nonlinear regression model parameters. Tabatabai et al.[6] provided a robust alternative method to the normal Least Squares nonlinear regression method.

In recent decades, the researcher aims to resolve complicated problems by using metaheuristic to overcome drawbacks of classical procedures and have many benefits containing the simplicity of implementation, reliability, robustness, and effectiveness.[7] Adibifard et al.[8] used PSO algorithm to perform nonlinear regression in well test analysis. Root Mean Square Error over pressedurized and
pressurized derivative data is employed to determine the formula of cost function $f$ and the multi-objective issue is minimized to a single-objective one by containing the weight for each cost function related to over pressurized and pressurized derivative data. Özsoy, and Örkçü,[9] proposed Particle Swarm Optimization (PSO) algorithm in order to improve the accuracy of parameters estimation for nonlinear regression models. The PSO algorithm is examined on the famous 28 nonlinear regression tasks of various levels of difficulty.

On the other hand, Gravitational Search Algorithm (GSA) is a modern meta-heuristic and population-based search algorithm that depends on gravity Newton’s law and motion law. Additionally, GSA has many benefits as, adaptive learning rate, memory-less algorithm and, perfect and rapid convergence. Additionally, GSA has been successfully used in complex problems. Thus, in this paper, Gravitational Search Algorithm was used to estimate the parameters of nonlinear regression models.

The organized paper is, section two provides the Maximum likelihood Estimation of two nonlinear regression models; Section three describes the procedural Gravitational Search Algorithm; Section four consists of a simulation study; a conclusion is provided in section five.

2. Maximum likelihood Estimation Nonlinear regression models
The Maximum likelihood method (MLE) was used to estimate the parameter for two models of nonlinear regression (Misra 1d, and MGH 09 Model) as follows:

2.1 Maximum likelihood method to solve Misra 1d Model

The Misra 1d model is

$$f(x_i; \beta) = \frac{\beta_1 \beta_2 x}{1 + \beta_2 x}$$

(1)

MLE method of estimation depends on maximizing the pdf estimation:

$$L = f(x_1, x_2, ..., x_n, \beta, \sigma^2)$$

$$L = (2\pi\sigma^2)^{-\frac{n}{2}} e^{-\frac{\sum_{i=1}^{n}(y_i - f(x_i; \beta))^2}{2\sigma^2}}$$

$$\ln L = -\frac{n}{2} \ln 2\pi - \frac{n}{2} \ln \sigma^2 - \frac{\sum_{i=1}^{n}(y_i - f(x_i; \beta))^2}{2\sigma^2}$$

Numerical procedures as Newton-Raphson was used to estimate the parameters since the equations are complicated to be solved. Therefore, the equation for this method for the first model is as follows

$$\begin{pmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\sigma}^2 \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ \beta_{20} \\ \sigma_0^2 \end{pmatrix} - \begin{pmatrix} \partial h_1 \\ \partial h_2 \\ \partial h_3 \end{pmatrix} \begin{pmatrix} \partial L \\ \partial \beta_1 \\ \partial \beta_2 \\ \partial \sigma^2 \end{pmatrix}^{-1} \begin{pmatrix} \partial L \\ \partial \beta_1 \\ \partial \beta_2 \\ \partial \sigma^2 \end{pmatrix}$$

(2)

$$(\begin{pmatrix} \beta_{10} \\ \beta_{20} \\ \sigma_0^2 \end{pmatrix})$$ Represents the vector of the initial parameters
\[ 
\begin{align*}
    h_1 &= \frac{\partial \ln L}{\partial \beta_1} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i \beta_2 x}{1 + \beta_2 x} \right) - \sum_{i=1}^{n} \left( \frac{\beta_1 \beta_2^2 x^2}{(1 + \beta_2 x)^2} \right), \\
    h_2 &= \frac{\partial \ln L}{\partial \beta_2} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x}{1 + \beta_2 x} \right) - \sum_{i=1}^{n} \left( \frac{2 \beta_1 \beta_2 x^2}{(1 + \beta_2 x)^3} \right), \\
    h_3 &= \frac{\partial \ln L}{\partial \sigma^2} = -\frac{n}{2 \sigma^2} + \sum_{i=1}^{n} \frac{1}{2 \sigma^4} (y_i - \beta_1 \beta_2 x)^2. 
\end{align*} 
\]

\[ 
\frac{\partial h_1}{\partial \beta_1} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{\beta_1 \beta_2^2 x^2}{(1 + \beta_2 x)^2} \right), \\
\frac{\partial h_1}{\partial \beta_2} = \frac{2 \beta_1 \beta_2 x^2}{\sigma^2 (1 + \beta_2 x)^3}, \\
\frac{\partial h_2}{\partial \beta_1} = \frac{2 \beta_1 \beta_2 x^2}{\sigma^2 (1 + \beta_2 x)^3}, \\
\frac{\partial h_2}{\partial \beta_2} = \frac{2 \beta_1 \beta_2 x^2}{\sigma^2 (1 + \beta_2 x)^3}, \\
\frac{\partial h_3}{\partial \beta_1} = -\frac{1}{\sigma^4} \sum_{i=1}^{n} \left( \frac{y_i \beta_1 x}{1 + \beta_2 x} \right) - \sum_{i=1}^{n} \left( \frac{\beta_1 \beta_2^2 x^2}{(1 + \beta_2 x)^3} \right), \\
\frac{\partial h_3}{\partial \beta_2} = -\frac{1}{\sigma^4} \sum_{i=1}^{n} \left( \frac{y_i \beta_1 x}{1 + \beta_2 x} \right) - \sum_{i=1}^{n} \left( \frac{\beta_1 \beta_2^2 x^2}{(1 + \beta_2 x)^3} \right), \\
\frac{\partial h_3}{\partial \sigma^2} = -\frac{1}{\sigma^4} \sum_{i=1}^{n} \frac{1}{\sigma^6} (y_i - \beta_1 \beta_2 x)^2. 
\]

2.2 Maximum likelihood method to solve MGH 09 Model

The MGH 09 Model is

\[ f(x_i, \beta) = \frac{\beta_1 (x^2 + x \beta_2)}{x^2 + x \beta_3 + \beta_4} \] (3)

The formula for MLE is for MGH 09 Model:

\[ \ln L = -\frac{n}{2} \ln 2\pi - \frac{n}{2} \ln \sigma^2 - \frac{\sum_{i=1}^{n} (y_i - \beta_1 (x^2 + x \beta_2))^2}{2 \sigma^2} \]

Thus, the following equation matrixes are applied to estimate the parameters for the non-linear regression model by using the Newton-Raphson method for the second model.
\[
\begin{bmatrix}
\hat{\beta}_1 \\
\hat{\beta}_3 \\
\hat{\beta}_4 \\
\hat{\sigma}^2
\end{bmatrix}
= \begin{bmatrix}
\beta_{10} \\
\beta_{20} \\
\beta_{30} \\
\beta_{40} \\
\sigma_0^2
\end{bmatrix}
- \left(\begin{bmatrix}
\frac{\partial h_1}{\partial \beta_1} & \frac{\partial h_1}{\partial \beta_2} & \frac{\partial h_1}{\partial \beta_3} & \frac{\partial h_1}{\partial \beta_4} & \frac{\partial h_1}{\partial \sigma^2} \\
\frac{\partial h_2}{\partial \beta_1} & \frac{\partial h_2}{\partial \beta_2} & \frac{\partial h_2}{\partial \beta_3} & \frac{\partial h_2}{\partial \beta_4} & \frac{\partial h_2}{\partial \sigma^2} \\
\frac{\partial h_3}{\partial \beta_1} & \frac{\partial h_3}{\partial \beta_2} & \frac{\partial h_3}{\partial \beta_3} & \frac{\partial h_3}{\partial \beta_4} & \frac{\partial h_3}{\partial \sigma^2} \\
\frac{\partial h_4}{\partial \beta_1} & \frac{\partial h_4}{\partial \beta_2} & \frac{\partial h_4}{\partial \beta_3} & \frac{\partial h_4}{\partial \beta_4} & \frac{\partial h_4}{\partial \sigma^2} \\
\frac{\partial h_5}{\partial \beta_1} & \frac{\partial h_5}{\partial \beta_2} & \frac{\partial h_5}{\partial \beta_3} & \frac{\partial h_5}{\partial \beta_4} & \frac{\partial h_5}{\partial \sigma^2}
\end{bmatrix} \right)^{-1}
\left(\frac{\partial \ln L}{\partial \beta_1} \quad \frac{\partial \ln L}{\partial \beta_2} \quad \frac{\partial \ln L}{\partial \beta_3} \quad \frac{\partial \ln L}{\partial \beta_4} \quad \frac{\partial \ln L}{\partial \sigma^2}\right)
\]

\[
\begin{bmatrix}
\beta_{10} \\
\beta_{20} \\
\beta_{30} \\
\beta_{40} \\
\sigma_0^2
\end{bmatrix}
\]

Represents the vector of the initial parameters.

\[
h_1 = \frac{\partial \ln L}{\partial \beta_1} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i(x^2 + x \beta_2)}{x^2 + x \beta_3 + \beta_4} - \frac{\beta_1(x^2 + x \beta_2)^2}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
h_2 = \frac{\partial \ln L}{\partial \beta_2} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x}{x^2 + x \beta_3 + \beta_4} - \frac{\beta_1(x^2 + x \beta_2)^2}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
h_3 = \frac{\partial \ln L}{\partial \beta_3} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{\beta_1^2 x(x^2 + x \beta_2)^2}{(x^2 + x \beta_3 + \beta_4)^3} \right)
\]

\[
h_4 = \frac{\partial \ln L}{\partial \beta_4} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{\beta_1^2 x(x^2 + x \beta_2)^2}{(x^2 + x \beta_3 + \beta_4)^3} \right)
\]

\[
h_5 = \frac{\partial \ln L}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^{n} \left( \frac{y_i - \beta_1(x^2 + x \beta_2)^2}{x^2 + x \beta_3 + \beta_4} \right)^2
\]

\[
\frac{\partial h_1}{\partial \beta_1} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{2y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
\frac{\partial h_1}{\partial \beta_2} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x}{x^2 + x \beta_3 + \beta_4} - \frac{2\beta_1 x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
\frac{\partial h_1}{\partial \beta_3} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2\beta_1 x(x^2 + x \beta_2)^2(x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right)
\]

\[
\frac{\partial h_1}{\partial \beta_4} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2\beta_1 x(x^2 + x \beta_2)^2(x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right)
\]

\[
\frac{\partial h_1}{\partial \sigma^2} = -\frac{1}{\sigma^4} \sum_{i=1}^{n} \left( \frac{y_i x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{\beta_1(x^2 + x \beta_2)^2}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
\frac{\partial h_2}{\partial \beta_1} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x}{x^2 + x \beta_3 + \beta_4} - \frac{2\beta_1 x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]

\[
\frac{\partial h_2}{\partial \beta_2} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i x}{x^2 + x \beta_3 + \beta_4} - \frac{2\beta_1 x(x^2 + x \beta_2)}{(x^2 + x \beta_3 + \beta_4)^2} \right)
\]
\[ \frac{\partial h_2}{\partial \beta_1} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i \beta_1 x^2}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2 \beta_1^2 x^2 (x^2 + x \beta_2)^2 (x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right) \]

\[ \frac{\partial h_2}{\partial \beta_2} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i \beta_1 x}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2 \beta_1^2 x (x^2 + x \beta_2)^2 (x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right) \]

\[ \frac{\partial h_2}{\partial \beta_3} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i \beta_1}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2 \beta_1^2 x (x^2 + x \beta_2)^2 (x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right) \]

\[ \frac{\partial h_2}{\partial \beta_4} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left( \frac{y_i \beta_1 x}{(x^2 + x \beta_3 + \beta_4)^2} - \frac{2 \beta_1^2 x^2 (x^2 + x \beta_2)^2 (x^2 + x \beta_3 + \beta_4)}{(x^2 + x \beta_3 + \beta_4)^4} \right) \]
3. Gravitational Search Algorithm Technique

In 2009, Rashedi et al. [10] introduced the Gravitational Search Algorithm (GSA) for solving optimization problems. The populace-established heuristic algorithm is founded on the mass interactions and gravity law. The solutions in the population of GSA are called agents, through the gravity force the agents interact with each other. In the population, measured the performance of each agent by its mass. The best solution is the solution with a heavier mass.

The objects masses are obeying of gravity low as following:

\[
F_{ij} = G \frac{M_{aj} \times M_{pj}}{R^2}
\]  \hspace{1cm} (4)

Where \( G \) is gravitational constant, \( M_{pj} \) is the mass of the second object, \( M_{aj} \) is the mass of the first object, \( F \) is a magnitude of the gravitational force, and \( R \) is the distance between the two objects \( M_{aj}, M_{pj} \).

\[
a_i = \frac{F_{ij}}{M_{ij}}
\]  \hspace{1cm} (5)

The following steps of the GSA can be summarized as below:

Step 1. initializes values of gravitational constant \( G_0 \), \( \alpha \), \( \epsilon \), and the iteration counter \( t \). 
Step 2. generated the initial population randomly and consists of N agents, the position defined for each agent as below:

\[
x_i(t) = \left( x_{i1}(t), x_{i2}(t), ..., x_{in}(t) \right) \hspace{0.5cm} i = 1, 2, ..., N
\]

Step 3. This step is replicated until satisfied with termination criteria:

A. assigned the best, worst agents and evaluated the population for all agents

B. the constant of gravitational is updated as Equation 4

C. calculates the force as follows:

\[
F_i^d(t) = G(t) \frac{M_{aj}(t) \times M_{pj}(t)}{R_{ij}(t) + \epsilon} \left( X_j^d(t) - X_i^d(t) \right)
\]  \hspace{1cm} (8)

Where \( M_{aj} \) is the active gravitational mass of agent \( j \), \( M_{pj} \) is the passive gravitational mass of agent \( i \), \( G(t) \) is gravitational constant at time \( t \). 

D. At iteration \( t \), calculate the total force acting on agent \( i \) as follows:

\[
F_i^d(t) = \sum_{j \in \text{Kbest}, j \neq i} \text{rand}_j F_{ij}^d(t)
\]  \hspace{1cm} (9)

Where \( \text{K best} \) is the set of first \( K \) agents with the best fitness value and biggest mass

E. Calculate the inertial mass as following
\[ m_i(t) = \frac{fit - \text{worst}(t)}{\text{best}(t) - \text{worst}(t)} \]  
\[ M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{n} m_j(t)} \]  

F. Calculated the agent of acceleration as following:

\[ a_i(t) = \frac{F_i(t)}{M_i(t)} \]  

G. The position and velocity of agent i are computed as shown in Equations 6, 7.

H. The iteration counter is increased until termination criteria satisfied.

Step 4. The best optimal solution is produced.

The log-likelihood function was used as a fitness function of the GSA.

4. Graph Simulation study

In order to verify the performance of the different estimator’s methods, Simulation was used based on Mean Squares Error (MSE) to estimate the parameters of two models for nonlinear regression (Misra 1d and MGH 09 Model). Calculate the values of response variable \( y_i \) depend on \( x_i \), were generated according to the exponential distribution \( \text{exp}(2) \), while the random variable \( e_i \) is generated according to \( N(0, \sigma^2) \). In addition, each simulation condition was generated by 1000 replications. Various sample sizes are tested: 20, 40, 80,160, and 200. The simulation program was written using Matlab 2013. Generate the initial value of parameters for each model’s different set parameters utilized for each model as: \( (\beta_1, \beta_2) = (1000, 200) \), and \( (500, 500) \) for the first model, while for the second model \( (\beta_1, \beta_2, \beta_3, \beta_4) = (0.3, 0.2, 0.1, 0.1) \) and \( (0.3, 0.2, 0.3, 0.4) \), respectively.

Tables (1- 4) illustrated the results of the estimate parameters and the MSE for each parameter for two models of NLRA. The GSA algorithm provides a better result than MLE method.

**Table 1**: comparative results of GSA and MLE based on the first model when \( \beta_1 = 500 \) \( \beta_2 = 500 \)

| Model       | Methods | \( n \) | \( \beta_1 \) | \( \beta_2 \) | MSE       |
|-------------|---------|---------|---------------|---------------|-----------|
| **Misra 1d**|         |         |               |               |           |
|             | MLE     | 20      | Estimated     | 499.5956      | 295.229   | 5.670051  |
|             |         |         | MSE           | 11.09005      | 439283.6  |           |
|             | GSA     |         | Estimated     | 500.1654      | 220.6128  | 2.50637   |
|             |         |         | MSE           | 20.79941      | 174409.2  |           |
|             | MLE     | 40      | Estimated     | 498.7145      | 129.2191  | 17.83184  |
|             |         |         | MSE           | 19.13281      | 689491.3  |           |
|             | GSA     |         | Estimated     | 502.1351      | -64800.4  | 4.408084  |
|             |         |         | MSE           | 77.98416      | 4.24E + 10|           |
|             | MLE     | 80      | Estimated     | \( \frac{549.2827}{39.4545} \) | \( \frac{537.708}{1.34} \) | 9.4350790  |
|             |         |         | MSE           | 33.3333       | 3333.33   |           |
|             | GSA     |         | Estimated     | 502.3401      | 310.6598  | 1.366088  |
|             |         |         | MSE           | 69.37687      | 187373.4  |           |
|             | MLE     | 160     | Estimated     | 499.8794      | 236.9898  | 8.501998  |
|             |         |         | MSE           | 5.362929      | 321257.5  |           |
|             | GSA     |         | Estimated     | 500.4134      | 417.3795  | 1.514707  |
|             |         |         | MSE           | 5.3301        | 20407.72  |           |
|             | MLE     | 400     | Estimated     | 9765.851      | 1107440   | 349.4619  |
|             |         |         | MSE           | 9765.851      | 1107440   |           |
Table 2: comparative results of GSA and MLE based on the first model when $\beta_1 = 1000\ \beta_2 = 200$

| Model | $n$ | Methods   | $\beta_1$ | $\beta_2$ | $\text{MSE}$ |
|-------|-----|-----------|-----------|-----------|-------------|
| Misra 1d | 20  | MLE Estimated $\hat{\beta}_1$ | 986.7453 | $-124.206$ | 746.2497 |
|       |     | MLE $\text{MSE}$ | 270.5365 | 574591.1  |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 990.7536 | 136.1845  | 0.014534   |
|       |     | GSA $\text{MSE}$ | 301.8324 | 202523.2  |             |
|       | 40  | MLE Estimated $\hat{\beta}_1$ | 994.8666 | 295.9005  | 323.5123   |
|       |     | MLE $\text{MSE}$ | 46.18677 | 177214.2  |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 993.8156 | 264.0664  | 0.009915   |
|       |     | GSA $\text{MSE}$ | 123.4615 | 139393.2  |             |
|       | 80  | MLE Estimated $\hat{\beta}_1$ | 988.8398 | $-15.6546$ | 364.7559   |
|       |     | MLE $\text{MSE}$ | 243.1879 | 493297.7  |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 994.3077 | 289.225   | 0.011026   |
|       |     | GSA $\text{MSE}$ | 136.9521 | 107220.2  |             |
|       | 160 | MLE Estimated $\hat{\beta}_1$ | 994.6233 | $-61.9938$ | 0.21395    |
|       |     | MLE $\text{MSE}$ | 66.24217 | 569137.5  |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 988.4391 | 116.1104  | 0.009643   |
|       |     | GSA $\text{MSE}$ | 546.578  | 107172.1  |             |
|       | 200 | MLE Estimated $\hat{\beta}_1$ | 986.8886 | $-391.065$ | 1262.264   |
|       |     | MLE $\text{MSE}$ | 279.8969 | 1538285   |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 989.9985 | 292.5989  | 171.3449   |
|       |     | GSA $\text{MSE}$ | 243.2437 | 171441.3  |             |

Table 3: comparative results of GSA and MLE based on the second model when $\beta_1 = 0.3\ \beta_2 = 0.2\beta_3 = 0.1\ \beta_4 = 0.1$

| Model | $n$ | Methods   | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\text{MSE}$ |
|-------|-----|-----------|-----------|-----------|-----------|-----------|-------------|
| MGH 09 | 20  | MLE Estimated $\hat{\beta}_1$ | 1.954471 | $-0.43388$ | 0.491298 | 1.235867 | 41.61445   |
|       |     | MLE $\text{MSE}$ | 25.54746 | 4.15548   | 27.19137 | 7.717002 |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | 0.113475 | $-8994006$ | $-1.6E + 07$ | 16957031 | 0.2866     |
|       |     | GSA $\text{MSE}$ | 0.049797 | 8.09 + 14 | 2.59E + 15 | 2.88E + 15 |             |
|       | 40  | MLE Estimated $\hat{\beta}_1$ | 0.039963 | 0.19082   | $-1.04359$ | 0.751308 | 0.15719    |
|       |     | MLE $\text{MSE}$ | 0.888632 | 1.093795  | 2.205161 | 2.738818 |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | $-3.1008$ | 12.36802  | $-26.5377$ | 26.52696 | 0.005814   |
|       |     | GSA $\text{MSE}$ | 116.6475 | 1743.968  | 8644.996 | 6842.597 |             |
|       | 80  | MLE Estimated $\hat{\beta}_1$ | 0.119452 | 0.580837  | $-1.31832$ | 0.831274 | 183.59     |
|       |     | MLE $\text{MSE}$ | 0.132054 | 1.198069  | 2.818023 | 3.0717 |             |
|       |     | GSA Estimated $\hat{\beta}_1$ | $-2.20153$ | $-180.734$ | $-165.358$ | 150092.7 | 4.938464   |
|       |     | GSA $\text{MSE}$ | 49.18432 | 3.26 + 11 | 2.73E + 11 | 2.25E + 11 |             |
|       | 160 | MLE Estimated $\hat{\beta}_1$ | 0.219287 | 0.084688  | $-1.4635$ | 0.782096 | 1.025123   |
|       |     | MLE $\text{MSE}$ | 0.026464 | 1.068443  | 2.902361 | 2.835297 |             |
### Table 4: comparative results of GSA and MLE based on the second model when $\beta_1 = 0.3$ $\beta_2 = 0.2\beta_3 = 0.3$ $\beta_4 = 0.4$

| Model | n  | Methods | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $MSE$ |
|-------|----|---------|-----------|-----------|-----------|-----------|-------|
| MGH09 | 20 | MLE     | Estimated | 0.268851  | 0.43925   | 0.604946  | 0.020863|
|       |    |         | $MSE$ | 0.027177  | 0.095005  | 0.166329  | 0.067891|
|       |    | GSA     | Estimated | 0.322879  | 0.323069  | 0.583845  | 0.486207|
|       |    |         | $MSE$ | 0.003817  | 0.107965  | 0.214893  | 0.137402|
|       | 40 | MLE     | Estimated | 0.386908  | 0.460786  | 0.495018  | 0.674059|
|       |    |         | $MSE$ | 0.022945  | 0.125823  | 0.114295  | 0.105835|
|       |    | GSA     | Estimated | 0.289872  | 0.578694  | 0.354008  | 0.668938|
|       |    |         | $MSE$ | 0.004592  | 0.24232   | 0.149802  | 0.145513|
|       | 80 | MLE     | Estimated | 0.403243  | 0.365894  | 0.638887  | 0.624496|
|       |    |         | $MSE$ | 0.04209   | 0.120838  | 0.158708  | 0.094339|
|       |    | GSA     | Estimated | 0.289862  | 0.387528  | 0.301556  | 0.465746|
|       |    |         | $MSE$ | 0.001168  | 0.10986   | 0.073137  | 0.050988|
|       | 160| MLE     | Estimated | 0.370653  | 0.480084  | 0.532659  | 0.567015|
|       |    |         | $MSE$ | 0.025058  | 0.203756  | 0.134628  | 0.108836|
|       |    | GSA     | Estimated | 0.313968  | 0.368803  | 0.596611  | 0.469617|
|       |    |         | $MSE$ | 0.000549  | 0.073742  | 0.119012  | 0.053978|
|       | 200| MLE     | Estimated | 0.333854  | 0.464869  | 0.474432  | 0.625284|
|       |    |         | $MSE$ | 0.030065  | 0.110386  | 0.104334  | 0.158318|
|       |    | GSA     | Estimated | 0.308305  | 0.274282  | 0.394676  | 0.495746|
|       |    |         | $MSE$ | 0.000833  | 0.032696  | 0.083541  | 0.035074|
Conclusion
In this study, a metaheuristic algorithm (Gravitational Search algorithm) was used as an alternative method to estimate the parameters of two models of nonlinear regression (Misra 1d and Myer 7). To improve the validation of the algorithm, a simulation study was used. The result showed that the Gravitational Search algorithm provides good results than the classical MLE estimator.

References
[1] PAN, Zhengjun, and et al. 1995 Parameter estimation by genetic algorithms for nonlinear regression. *High Technology* 946–953
[2] Aşıkgil, B and Erar, A 2013 Polynomial tapered two stage least squares method in nonlinear regression *Applied Mathematics and Computation* 219 (18) 9743-9754
[3] Nwobi FN and Ugomma CA 2014 A comparison of methods for the estimation of Weibull distribution Parameter *MetodoloskiZvezki* 11 (1) 65-78
[4] Michailidis, P. D. 2018 A Preliminary Performance Study on Nonlinear Regression Models using the Jaya Optimization Algorithm *International Journal of Applied Mathematics* 48(4) 1-5
[5] Křivý I, Tvrdík J and Krpec R 2000 Stochastic algorithms in nonlinear regression *Computational Statistics & Data Analysis* 33 (3) 277-
[6] Tabatabai M A, Kengwoung-Keumo J J, Eby W M, Bae S, Manne U, Fouad M, andamp; Singh K P 2014 A new robust method for nonlinear regression. *Journal of biometrics &amp; biostatistics* 5 (5) 211
[7] Atiya B, Bakheet A J K, Abbas I T, Bakar M R A, Soon L, and Monsi M B 2016 Application of simulated annealing to solve multi-objectives for aggregate production planning *AIP Conf. Pro.s* 1739(1) 020086) AIP Publishing LLC
[8] Adibifard M, Bashiri G, Roayaei E, and Emad M A 2016 Using particle swarm optimization (PSO) algorithm in nonlinear regression well test analysis and its comparison with levenberg-marquardt algorithm *International Journal of Applied Metaheuristic Computing (IJAMC)* 7 (3) 1-23
[9] Özsoy V S, Ünsal M G, and Örkcü H H 2020 Use of the heuristic optimization in the parameter estimation of generalized gamma distribution: comparison of GA, DE, PSO and SA methods. *Computational Statistics* 1-31
[10] Rashedi H, Nezamabadi-pour, and S Saryazdi 2009 GSA: A Gravitational Search Algorithm *Information Sciences* 179 (13) 2232–2248