Parameter estimation of a truncated regression model based on improving numerical optimization algorithms

1) Marwan Saleh Jameel, 2) Ghalya Tawfeeq Basheer, 3) Abbas Y. Al-Bayati, 4) Zakariya Yahya Algamal

1) Department of Environmental Technology, University of Mosul, Mosul, Iraq. Email: marwanjameel3@gmail.com, ORCID: 0000-0003-3397-2749

2) Department of operations research and intelligent techniques, University of Mosul, Mosul, Iraq. E-mail: ghalia.tawfeek@uomosul.edu.iq.

3) Department of Mathematics, College of Basic Education, University of Telafer, Iraq. Email: profabbasalbayati@yahoo.com, ORCID: 0000-0002-2941-6945

4) Department of Statistics and Informatics, University of Mosul, Mosul, Iraq. E-mail: zakariya.algamal@uomosul.edu.iq, ORCID: 0000-0002-0229-7958

Abstract: Limited dependent variable models, including truncated regression models, have traditionally been estimated by the method of maximum likelihood. The conventional optimization algorithms; which is known as Quasi-Newton algorithm namely BFGS Quasi-Newton algorithm is used to reach the optimum values for estimated parameters. In this paper, the nature-inspired algorithm is employed to improve the numerical optimization algorithms to better estimation. Our Monte Carlo simulation results suggest that our proposed improving can bring significant improvement relative to others, in terms of mean squared error and prediction mean squared error.

Keywords: Truncated distributions; numerical optimization; Quasi-Newton; swarm intelligence.

1. Introduction

Maximum likelihood estimation (MLE) is a method of estimating the parameters of a probability distribution by maximizing a likelihood function, so that under the assumed statistical model the observed data is most probable. The point in the parameter space that maximizes the likelihood function is called the maximum likelihood estimate (Rossi 2018).

\[ L(\theta; x) = f_n(\hat{\theta}; x) \] (1)

To find the maxima(minima) of this function, we can take the derivative of this function w.r.t \( \theta \) and equate it to 0. "Accordingly, the question of estimating the coefficients converts into question of finding the best solution which means optimization problem."
Optimization generally can be categorized into two types which is constrained optimization and unconstrained optimization. In optimization, an objective function \( f(x) \) needs to be solved either to minimize or maximize the function:

\[
\min f(x) \quad x \in \mathbb{R}^n
\]  

(2)

Where \( f \) is twice continuously differentiable function from \( \mathbb{R}^n \) into \( \mathbb{R} \). If it is a constrained optimization, then (2) will be bounded to a series of constraint functions. However, consider problem (2) is unconstrained optimization.

By not considering the exact Hessian, Quasi-Newton method was developed using the Hessian approximation formula at every of iteration and we call it as update Hessian approximation formula at \( k+1 \)th iteration. There are several of update Hessian formulas in Quasi-Newton method such as Broyden-Fletcher-Goldfarb-Shanno (BFGS), Davidon-Fletcher-Powell (DFP), Symmetric Rank-One (SR1) and Broyden Family update.

The quasi-Newton method is an iterative method, whereby at \( k+1 \)th the iteration, \( x_{k+1} \) is given by:

\[
x_{k+1} = x_k + \alpha_k d_k
\]  

(3)

where \( \alpha_k \) is its step length, and \( d_k \) denotes the search direction and is calculated using

\[
d_k = -B_k^{-1} g_k
\]  

(4)

In (4), the quantity \( g_k = \nabla f(x_k) \) denotes the gradient of \( f \) at \( x_k \) while \( B_k \) is the Hessian approximation \( \nabla^2 f(x_k) \) that fulfills the Quasi-Newton equation:

\[
s_k B_k = y_k
\]  

(5)

The formula of update Hessian in quasi-Newton method is derived by

\[
B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k y_k} + \eta \left( s_k^T B_k s_k \right) v_k v_k^T + \frac{y_k y_k^T}{s_k^T y_k}
\]  

(6)

Where \( v_k = \left( \begin{array}{c} y_k \\ s_k/y_k \\ \frac{s_k^T B_k s_k}{s_k^T B_k y_k} \end{array} \right) \), \( \eta \) a scalar.

This update is satisfied the quasi-Newton equation (5). \( \eta \) is the parameter has value in the interval \( [0,1] \). later it written in the following form by (Huang and Absil 2015)

\[
B_{k+1} = (1-\eta)B_k^\text{BFGS} + \eta B_k^\text{DFP}
\]  

(7)

Hence, if we set \( \eta = 0 \) or \( \eta = 1 \) in (7) getting the most important formula of Quasi-Newton update matrix Broyden-Fletcher-Goldfarb-Shanno(BFGS) and Davidon-Fletcher-Powell(DFP) respectively.

The symbol \( \alpha_k \) in (3) is the step length obtained by line search procedure and in this paper, we only considered Quasi-Newton algorithm using the exact line search as suggested by (Al-Bayati 1991) that is

\[
\min_{\alpha \geq 0} f(x_k + \alpha d_k)
\]  

(8)

The exact line search in (8) must satisfy both Wolfe’s conditions.
2. Truncated distributions

If the observations come only from a restricted part of the population distribution then the sample is called truncated. So that the truncated distribution is a part of untruncated distribution, where the truncated can occur from one or two sides: truncation from below at point \( a \) (left truncated \( (x > a) \)), and truncation from above at point \( b \) (right truncated \( (x < b) \)) and truncation from two sides at interval \([a, b]\). Thus, the probability density function of the truncated random variable is a conditional distribution as shown below (Green 2003, Heij and et al. 2004):

**Case (1):** Truncation from the left side \( (x > a) \)

\[
f_T (x \mid x > a) = \frac{f(x; \theta)}{\Pr(x > a)} \quad a \leq x \leq \infty
\]

**Case (2):** Truncation from the right side

\[
f_T (x \mid x < b) = \frac{f(x; \theta)}{\Pr(x < b)} \quad -\infty \leq x \leq b
\]

**Case (3):** Truncation from both sides

\[
f_T (x \mid a < x < b) = \frac{f(x; \theta)}{\Pr(a < x < b)} \quad a \leq x \leq b
\]

Where \( a \) and \( b \) represent the truncation points from the left and right respectively, \( f(x; \theta) \) represents the probability density function of distribution, while \( f_T(x; \cdot) \) represents probability density function of truncated distribution.

3. Truncated normal distribution

Let a continuous random variable follow normal distribution with mean \( \mu \) and variance \( \sigma^2 \) then the probability density function of normal distribution is (Green 2003, Demaris 2004):

\[
f(x; \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad -\infty \leq x \leq \infty
\]

Therefore, the probability density function of the truncated normal distribution is:

**Case (1):** Truncation from the left side \( (x > a) \)

\[
f_T (x \mid x > a) = \frac{f(x; \mu, \sigma)}{\Pr(x > a)} \quad a \leq x \leq \infty
\]
As known as
\[
\Pr(x > a) = 1 - \Pr(x \leq a) = 1 - \Phi\left(\frac{a - \mu}{\sigma}\right)
\]
(15)

So that the probability density function of the truncated normal distribution is a conditional distribution:
\[
f_T(x \mid x > a) = \frac{f(x; \mu, \sigma)}{\Pr(x > a)} = \frac{\frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{1 - \Phi\left(\frac{a - \mu}{\sigma}\right)} = \frac{\phi(z)}{\sigma(1 - \Phi(\alpha))}
\]
(16)

Where \(\phi(z)\) represents probability density function of standard normal distribution \(z = \frac{x - \mu}{\sigma}\) and \(\Phi(z)\) represents cumulative function of standard normal distribution \(\alpha = \frac{a - \mu}{\sigma}\).

Then the truncated mean and truncated variance of is:
\[
E[x \mid truncation] = \mu + \sigma \lambda(\alpha)
\]
(17)
\[
Var[x \mid truncation] = \sigma^2 \left[1 - \delta(\alpha)\right]
\]
(18)

Where
\[
\lambda(\alpha) = \frac{\phi(\alpha)}{1 - \Phi(\alpha)} \quad \text{if truncation is } x > a
\]
(19)
\[
\delta(\alpha) = \frac{\lambda(\alpha) - \alpha < 0 < 1}{\lambda(\alpha) - \alpha < 0 < 1} \quad \text{for all value of } \alpha
\]
(20)

Case (2): Truncation from the right side
\[
f_T(x \mid x < b) = \frac{f(x; \mu, \sigma)}{\Pr(x < b)} = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = \phi(z)
\]
(21)

As known as
\[
\Pr(x < b) = \Phi\left(\frac{b - \mu}{\sigma}\right)
\]
(22)

So that the probability density function of the truncated normal distribution is a conditional distribution:
Then the truncated mean and truncated variance of is:

\[ E \left[ x \mid \text{truncation} \right] = \mu + \sigma \hat{\lambda}(\gamma) \]  \hspace{1cm} (25)
\[ \text{Var} \left[ x \mid \text{truncation} \right] = \sigma^2 \left[ 1 - \delta(\gamma) \right] \]  \hspace{1cm} (26)

Where

\[ \hat{\lambda}(\gamma) = -\phi(\gamma)/\Phi(\gamma) \quad \text{if \ truncation is } x < b \]  \hspace{1cm} (27)
\[ \delta(\gamma) = \hat{\lambda}(\gamma) - \gamma \quad 0 < \delta(\gamma) < 1 \quad \text{for all value of } \gamma \]

Where \( \gamma = \left( \frac{b - \mu}{\sigma} \right) \) and \( \hat{\lambda}(\alpha), \hat{\lambda}(\gamma) \) called inverse mills ratio (IMR) or hazard function of standard normal distribution.

**Case (3):** Truncation from both sides

\[ f_T(x \mid a < x < b) = \frac{f(x; \mu, \sigma)}{\Pr(a < x < b)} \quad a \leq x \leq b \]  \hspace{1cm} (28)

As known as

\[ \Pr(a < x < b) = \Phi \left( \frac{b - \mu}{\sigma} \right) - \Phi \left( \frac{a - \mu}{\sigma} \right) \]  \hspace{1cm} (29)

So that the probability density function of the truncated normal distribution is a conditional distribution:

\[ f_T(x \mid a < x < b) = \frac{f(x; \mu, \sigma)}{\Pr(a < x < b)} \]
\[ = \frac{\left( 2\pi\sigma^2 \right)^{-\frac{1}{2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\Phi \left( \frac{b - \mu}{\sigma} \right) - \Phi \left( \frac{a - \mu}{\sigma} \right)} \]  \hspace{1cm} (30)

Then the truncated mean and truncated variance of is:

\[ \text{Then the truncated mean and truncated variance of is:} \]

\[ E \left[ x \mid \text{truncation} \right] = \mu + \sigma \hat{\lambda}(\gamma) \]  \hspace{1cm} (25)
\[ \text{Var} \left[ x \mid \text{truncation} \right] = \sigma^2 \left[ 1 - \delta(\gamma) \right] \]  \hspace{1cm} (26)

Where

\[ \hat{\lambda}(\gamma) = -\phi(\gamma)/\Phi(\gamma) \quad \text{if \ truncation is } x < b \]  \hspace{1cm} (27)
\[ \delta(\gamma) = \hat{\lambda}(\gamma) - \gamma \quad 0 < \delta(\gamma) < 1 \quad \text{for all value of } \gamma \]

Where \( \gamma = \left( \frac{b - \mu}{\sigma} \right) \) and \( \hat{\lambda}(\alpha), \hat{\lambda}(\gamma) \) called inverse mills ratio (IMR) or hazard function of standard normal distribution.

**Case (3):** Truncation from both sides

\[ f_T(x \mid a < x < b) = \frac{f(x; \mu, \sigma)}{\Pr(a < x < b)} \quad a \leq x \leq b \]  \hspace{1cm} (28)

As known as

\[ \Pr(a < x < b) = \Phi \left( \frac{b - \mu}{\sigma} \right) - \Phi \left( \frac{a - \mu}{\sigma} \right) \]  \hspace{1cm} (29)

So that the probability density function of the truncated normal distribution is a conditional distribution:

\[ f_T(x \mid a < x < b) = \frac{f(x; \mu, \sigma)}{\Pr(a < x < b)} \]
\[ = \frac{\left( 2\pi\sigma^2 \right)^{-\frac{1}{2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\Phi \left( \frac{b - \mu}{\sigma} \right) - \Phi \left( \frac{a - \mu}{\sigma} \right)} \]  \hspace{1cm} (30)

Then the truncated mean and truncated variance of is:
4. Truncated Regression Model

Consider the following linear regression model (Newey 2001, Green 2003, Heij and et al. 2004, Karlsson 2006):

\[ y_i = x_i^T \beta + \zeta_i, \quad i = 1, 2, \ldots, n \]

where

\[ x_i^T \beta = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_n x_{in} \quad (33) \]

\[ y_i \sim N \left[ x_i^T \beta, \sigma^2 \right] \]

In a truncated regression model the observations of \((y_i, x_i)\) are obtained only for part of the population so the probability distribution of error term \(\zeta_i\) become truncated distribution

**Case (1):** Truncation from the left side \((y_i > a)\). “in this case the conditional mean and variance of the truncated regression model can be written as:

\[ E \left[ y_i \mid y_i > a \right] = E \left( x_i^T \beta + \zeta_i \mid \zeta_i > a - x_i^T \beta \right) \]

\[ E \left[ y_i \mid y_i > a \right] = x_i^T \beta + \sigma \lambda (\alpha_i) \quad (34) \]

\[ Var \left[ y_i \mid y_i > a \right] = Var \left( \zeta_i \mid \zeta_i > a - x_i^T \beta \right) \]

\[ Var \left[ y_i \mid y_i > a \right] = \sigma^2 \left( 1 - \delta (\alpha_i) \right) \quad (35) \]

where

\[ \lambda (\alpha_i) = \phi (\alpha_i) \left[ 1 - \Phi (\alpha_i) \right] \quad \text{if truncation is } y_i > a \]

\[ \delta (\alpha_i) = \lambda (\alpha_i) \left[ \frac{\lambda (\alpha_i)}{\lambda (\alpha_i) - \alpha_i} \right] \]

\[ 0 < \delta (\alpha_i) < 1 \quad \text{for all values of } \alpha_i \]

\[ \alpha_i = \frac{a - x_i^T \beta}{\sigma} \]

**Case (2):** Truncation from the right side \((y_i < b)\). in this case the conditional mean and variance of the truncated regression model can be written as:
\[ E \left[ y_i \mid y_i < b \right] = x_i^T \beta + \sigma \lambda (\gamma_i) \]  
(36)

\[ \text{Var} \left[ y_i \mid y_i < b \right] = \sigma^2 \left( 1 - \delta (\gamma_i) \right) \]  
(37)

where
\[ \lambda (\gamma_i) = -\phi (\gamma_i) / \Phi (\gamma_i) \text{ if truncation } y_i < b \]
\[ \delta (\gamma_i) = \lambda (\gamma_i) \left( \lambda (\gamma_i) - \gamma_i \right) \]
\[ 0 < \delta (\gamma_i) < 1 \text{ for all values of } \gamma_i \]

where \[ \gamma_i = \frac{b - x_i^T \beta}{\sigma} \]

Case (3): Truncation from both sides \((a \leq y_i \leq b)\), in this case the conditional mean and variance of the truncated regression model can be written as:

\[ E \left[ y_i \mid a < y_i < b \right] = x_i^T \beta - \sigma \left( \frac{\varphi(\gamma) - \varphi(\alpha)}{\Phi(\gamma) - \Phi(\alpha)} \right) \]  
(38)

\[ \text{Var} \left[ y_i \mid a < y_i < b \right] = \sigma^2 \left[ \frac{1 + \frac{\varphi(\alpha)}{\Phi(\gamma) - \Phi(\alpha)} \left( \frac{\varphi(\gamma)}{\Phi(\gamma) - \Phi(\alpha)} \right)}{\frac{\varphi(\gamma)}{\Phi(\gamma) - \Phi(\alpha)} \left( \frac{\varphi(\gamma)}{\Phi(\gamma) - \Phi(\alpha)} + 1 \right)} \right] \]  
(39)

The basic goal of regression analysis is to estimate the parameters and thus study the relationship between the response variable and the explanatory variables. In truncated regression model, the estimators resulting of the ordinary least squares are biased and inconsistent because \( E \left[ \zeta \mid x \right] \) is a function of \( x \) and not equal to zero. Therefore, the maximum likelihood estimator is used to estimate the parameters of this model.

5. Flower pollination algorithm (FPA)

The flower pollination algorithm is a bio-inspired algorithm that mimic the pollination process of flowers in plant. The first who suggested this algorithm is Yang in (2012) (Yang 2012) when it was used to solve single objective optimization problems. Then in (2014) Yang et al. (Yang and et al. 2014) extended this algorithm to solve multi objective optimization problems.

In the flower pollination algorithm, the following four basic rules are used (Yang 2012, Abdel-Basset and et al. 2017, Bozorg-Haddad 2018):

1. The global pollination includes biotic and cross-pollination, the pollinators move in a way which follows a lévy flight distribution.
2. The local pollination includes abiotic and self-pollination.
3. Flower constancy can be considered as the reproduction probability that is proportional to the similarity of two flowers involved.
4. We use a switch probability \( p \in [0,1] \) to switch between global pollination and local pollination.

Rules 1 and 3 can be explained mathematically by the following equation:
\[ x_{i}^{t+1} = x_{i}^{t} + \gamma L(\lambda)(x_{i}^{t} - g^{*}) \]  

(40)

where \( x_{i}^{t} \) is the solution vector or the pollen \( i \) at iteration \( t \), \( g^{*} \) is the current best solution that is found at the current iteration, \( \gamma \) is a scaling factor to control the step size, \( L(\lambda) \) is the step size in the lévy flights which is represented the strength of the pollination.

Rules 2 and 3 can be explained mathematically by the following equation:

\[ x_{i}^{t+1} = x_{i}^{t} + k \left( x_{j}^{t} - x_{k}^{t} \right) \]  

(41)

where \( x_{j} \) and \( x_{k} \) are the pollens (solution vectors) from different flowers of the same plant. \( k \) is the parameter draw from uniform distribution in \([0,1]\). To switch between common global pollination to intensive local pollination we used rule 4, Yang(2012) suggested that switch probability or proximity probability \( p = 0.8 \) for most applications (Yang 2015). The flower pollination algorithm can be described in Algorithm 1(Abdel-Basset and et al. 2018)

---

**Flower Pollination Algorithm**

**Begin**

- Define the objective function max or min \( f(x) \), and switch probability
- Initialize the population of \( n \) random flowers
- Evaluate each flowers in the population
- Find the best solution
- while (stopping criterion)
  - for \( i = 1:n \)
    - if \( (r<p) \), where \( r \in [0,1] \) is a random number
      - New solution = global pollination Eq. (40)
    - else
      - New solution = local pollination Eq. (41)
    - end if
  - Evaluate new solution
  - If new solution is better, update the population
  - end for
- Find current best solution
**End**

---

**Algorithm 1 Flower pollination algorithm**

6. A simulation study

In this section, we conducted a simulation study to analyze the performance of the proposed algorithms to estimate the parameters of the truncated regression model versus the traditional methods used to estimate these parameters. We have simulated number of variables (\( P=4, 12 \)) with samples of sizes (\( n=30, 50, 150, 200 \)) and the truncation point, \( \alpha \), was set to zero.

To suggest a new algorithm for estimating the parameters of the truncated regression model, the traditional optimization algorithms BFGS has been linked to the flower pollination algorithm. “In the proposed algorithms, BFGS-FPA, the flower pollination algorithm was used to find the optimal step size in each iteration as the algorithm was provided with parameter values before truncating them as initial values of the algorithm, and this algorithm was programmed using Matlab (R2018a) and Tables 1 and 2 illustrate the results of the implementation of this algorithms.

In general, it can be observed from Tables 1 and 2 that the performance of the BFGS-FPA exhibits better than BFGS method providing the smallest Mean square error (MSE). For instance, when \( n=150, \)
P=4, the MSE of the BFGS-FPA was about 82.54% lower than that of BFGS. Moreover, it is clearly seen from Tables 1 and 2 that the efficiency of BFGS-FPA method is comparable with BFGS method in all cases in terms of iteration”. In other words, BFGS-FPA has less computational time than BFGS.

Table 1: Results obtained from estimating parameters of a truncated regression model when P=4

| N   | parameters | BFGS | BFGS-FPA |
|-----|------------|------|----------|
| iteration |          |      |          |
|      | 30        | 8    | 3        |
|      | $\sigma^2$ | 0.8000812 | 0.8982323 |
|      | $b_0$     | 1.229474 | 1.1106680 |
|      | $b_1$     | 12.37803 | 12.270910 |
|      | $b_2$     | 19.92319 | 20.157226 |
|      | $b_3$     | 14.32882 | 14.240665 |
|      | $b_4$     | 15.1841  | 15.032034 |
| MSE  | 0.063908  | 0.029944 |
| iteration | 14        | 3    |
|      | $\sigma^2$ | 1.0918473 | 1.0996110 |
|      | $b_0$     | 1.416172 | 1.20195  |
|      | $b_1$     | 11.94911 | 12.048791 |
|      | $b_2$     | 19.73807 | 20.064736 |
|      | $b_3$     | 13.97211 | 14.177230 |
|      | $b_4$     | 15.10521 | 15.033186 |
| MSE  | 0.044113  | 0.014965 |
| iteration | 10        | 4    |
|      | $\sigma^2$ | 0.766203834 | 0.882091796 |
|      | $b_0$     | 1.258671 | 1.083950818 |
|      | $b_1$     | 11.99812 | 12.02026416 |
|      | $b_2$     | 20.07073 | 20.08311958 |
|      | $b_3$     | 13.80249 | 14.00251194 |
|      | $b_4$     | 14.79469 | 14.91065744 |
| MSE  | 0.034623  | 0.006043 |
| iteration | 11        | 4    |
|      | $\sigma^2$ | 1.012669879 | 0.909455467 |
|      | $b_0$     | 0.9850523 | 0.962734616 |
|      | $b_1$     | 12.05163 | 12.02696115 |
|      | $b_2$     | 19.89314 | 20.00015797 |
|      | $b_3$     | 14.09797 | 14.10871836 |
|      | $b_4$     | 15.01501 | 15.01153281 |
| MSE  | 0.004049  | 0.003711 |
Table 2: Results obtained from estimating parameters of a truncated regression model when P=12

| N | iteration | BFGS | BFGS-FPA |
|---|-----------|------|----------|
| 30 | 22        | 0.34359691 | 0.598511775 |
|    |           | 0.606166   | 0.774785118 |
|    |           | 7.77767    | 7.813245387 |
|    |           | 10.08371   | 10.01997445 |
|    |           | 15.24883   | 15.09385523 |
|    |           | 11.69229   | 11.69129266 |
|    |           | 17.2453    | 17.29733151 |
|    |           | 5.458414   | 5.128901618 |
|    |           | 18.44117   | 18.51237757 |
|    |           | 3.492324   | 3.231966906 |
|    |           | 10.7247    | 10.70775177 |
|    |           | 18.56987   | 18.41085749 |
|    |           | 14.53999   | 14.28295034 |
|    |           | 7.177194   | 6.952258981 |
| MSE | 0.167607  | 0.077461   |
| iteration | 22        | 4 |
| 50 | 19        | 0.267098881 | 0.894512857 |
|    |           | 1.059456   | 0.848034026 |
|    |           | 8.323558   | 8.092465863 |
|    |           | 10.34875   | 9.955623786 |
|    |           | 15.10143   | 15.15775206 |
|    |           | 11.59126   | 11.79026339 |
|    |           | 17.4451    | 16.99650747 |
|    |           | 5.142911   | 5.195753981 |
|    |           | 18.63297   | 18.93724591 |
|    |           | 3.130889   | 3.003928563 |
|    |           | 11.11545   | 11.09494103 |
|    |           | 17.73667   | 17.82597894 |
|    |           | 13.66314   | 13.6617613 |
|    |           | 7.210901   | 7.306608153 |
| MSE | 0.111059  | 0.028829   |
| iteration | 19        | 2 |
| 150| 15        | 0.814012296 | 0.856347002 |
|    |           | 0.8294981  | 0.953292269 |
|    |           | 7.972413   | 7.973469027 |
|    |           | 10.0784    | 10.06715035 |
### 7. Conclusions

Truncated regression model is considered one of most used application in economics. In this paper, flower pollination algorithm has been proposed to improve the BFGS method in parameter estimation of the truncated regression model. Based on simulation study, the results have demonstrated that the performance of our proposed BFGS-FPA compared with the BFGS leads to a better performance in terms of MSE.

### REFERENCES

1. Abdel-Basset, M., D. El-Shahat and I. El-Henawy (2018). "Solving 0–1 knapsack problem by binary flower pollination algorithm." Neural Computing and Applications 31(9): 5477-5495.
2. Abdel-Basset, M., D. El-Shahat, I. El-Henawy and A. K. Sangaiah (2017). "A modified flower pollination algorithm for the multidimensional knapsack problem: human-centric decision making." Soft Computing 22(13): 4221-4239.
3. Al-Bayati, A. Y. (1991). "A new family of self-scaling variable metric algorithms for unconstrained optimization." *Journal of Education and Science, Mosul University* **12**: 25-54.
4. Bozorg-Haddad, O. (2018). "Advanced Optimization by Nature-Inspired Algorithms." *Springer Nature Singapore Pte Ltd.*
5. Demaris, A. (2004). "Regression with social data modeling continuous and limited response variables." A John Wiley & Sons, Inc., Publication.
6. Green, W. (2003). "Econometric analysis." Printed in the USA Fifth Edition.
7. Heij, C., P. D. Boer, P. Franses, T. Klock and H. V. Dijk (2004). "Econometric methods with applications in business and economics." *Oxford university press.*
8. Huang, W., Kyle A. Gallivan and P. A. Absil (2015). "A Broyden class of quasi-Newton methods for Riemannian optimization." *SIAM Journal on Optimization:* 1660-1685.
9. Karlsson, M. (2006). "Estimators of regression parameters for truncated and censored data." *Metrika* **63**: 329-341.
10. Newey, W. K. (2001). "Conditional moment restrictions in censored and truncated regression models." *Econometric Theory* **17**: 863-888.
11. Rossi, R. J. (2018). "Mathematical statistics: an introduction to likelihood based inference." John Wiley & Sons.
12. Yang, X.-S. (2012). "Flower pollination algorithm for global optimization." *Unconventional Computation and Natural Computation* **7445**: 240-249.
13. Yang, X.-S. (2015). "Recent Advances in Swarm Intelligence and Evolutionary Computation." *Springer International Publishing Switzerland.*
14. Yang, X.-s., M. Karamoglu and H. Xingshi (2014). "Flower pollination algorithm: A novel approach for multiobjective optimization." *Engineering Optimization.*