Development of a Genetic Algorithm (GA) Code in Python Language for Fracture Porosity Analysis

Munasinghe P.T. and *Giao P.H.

Geosystem Exploration and Petroleum Geoengineering Program
Asian Institute of Technology, Bangkok, Thailand

*Corresponding author - hgiao@ait.asia

Abstract
Machine Learning (ML) techniques are more and more applied in hydrocarbon exploration and production (E&P) in general, and in petrophysics in particular. In this research, a Genetic Algorithm (GA) code was developed in Python language to analyze the fracture porosity of a Fractured Granite Basement (FGB) reservoir, which is difficult to calculate due to the reservoir heterogeneity caused by fracture networks. The study well was in the Cuu long basin, Vietnam. The steps of GA code development include defining the GA and evaluation functions, calculating fracture porosity, training and generating new population as well as printing and plotting the results of the models. For main GA functions, the Multiple Linear Regression (MLR) and Root Mean Square Error (RMSE) formulas were used. The best model was evaluated based on the least total prediction error, cost and execution time. The fracture porosity was first calculated by a conventional method and further used to train the GA models, among which the GA model consisting of 1080-training data with 100 population showed the best performance.

Keywords: Cuu long basin, Fractured Granite Basement Reservoirs, Fracture Porosity, Genetic Algorithm, Python

1 Introduction

Porosity is one of three main petrophysical properties of a hydrocarbon reservoir, next to permeability and water saturation. While, for porous clastic reservoir, its calculation is quite straight forward, for a fractured reservoir is very complicated [1].

In this study, application of an optimization method, i.e., genetic algorithm (GA), in petrophysical analysis was attempted. The study location is in the Cuu Long Basin (Figure 1), where oil exploration and production have started since 1986[2].

Figure1: Cuu Long Basin [2].

2 Fracture Porosity Determination

In this study, Elkewidy and Tiab (1998)'s method [3] was used. First, porosity from density log ($PHID$) is by Eq. 1:

$$PHID = \frac{\rho_{ma} - RHOB}{\rho_{ma} - \rho_f} \qquad (1)$$

Where,
$RHOB$: Well log density (g/cc)
ρ_{ma}: Matrix density (g/cc), 2.71 g/cc
ρ_f: Fluid density (g/cc), 1.0 g/cc

Next, total porosity is calculated by Eq. 2:

$$\emptyset_t = \frac{NPHI + PHID}{2} \qquad (2)$$

Then, fracture porosity of a fractured reservoir can be estimated by Eq. 3 below:

$$\emptyset_f = \frac{\emptyset_t^{m+1} - \emptyset_t^m}{\emptyset_t^m - 1} \qquad (3)$$

Where,
$NPHI$: Neutron porosity (fraction)
$\emptyset_t$: Total porosity (fraction)
$\emptyset_f$: Fracture porosity (fraction)
M: Cementation exponent

3 Genetic Algorithm (GA)

The GA was introduced by Holland (1975) at the University of Michigan [4]. This method is based on the natural genetics and the natural selection.

In this algorithm, fitter approximate solutions are selected based on the randomly initiated population [5]. Population will be reproduced and developed over generations during the competition process. Chromosome is a set of genes or solution and the position of a gene on a chromosome is locus.

4 Python Programming Language

Currently, the machine-learning trend is supporting to develop this high-level language. As a matter of fact, Python became the world's fastest-growing language in 2018 [6]. The software is free to use for major operating systems. In this study, program codes were written and tested using Python 3.7 which was released on June 27, 2018 [7].

5 Preparation of GA Program for Analysis

The GA program is designed based on the flow chart in Figure 2.

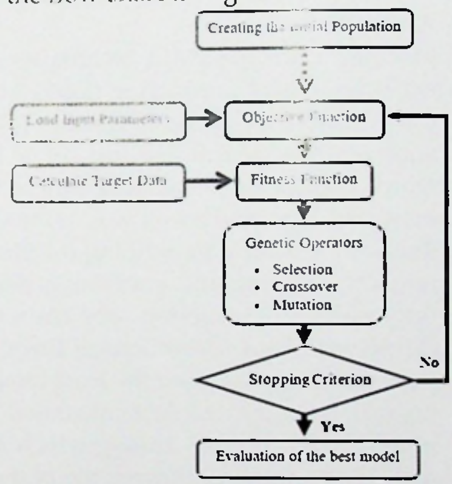

Figure 2: Flowchart of GA Code.

The well log data of the study well separated into few different sections as seen in Table 1. CAL, GR, RHOB, NPHI, LLS and LLD were used as input parameters.

Table 1: GA Models for Each Analysis Zone of the Study Site.

Zone	Depth (m)	Models		
	From To	From		To
CL_01	2515-2540	1B	-	3B
CL_02	2540-2655	4B	-	7B
CL_03	2790-3015	8B	-	11B
Total	2515-3015	12B	-	15B

The Fracture Porosity was first conventionally calculated. Dataset separated into training, validation and testing by 60:20:20 proportions [8]. GA models were then trained after creating the initial population. This population was represented using real-coded (RC) representation [9]. A random real number generated as a gene based on the calculated regression analysis results. The 500th generation was considered as the stopping criterion. Each functions and operators of the GA program is explained in the following sections.

5.1 Objective Function

The multiple linear regression (MLR) equation is used as the objective function as given by Eq. 4[10]. The function will take initial population and training data to return estimated GA fracture porosity which can be evaluated using fitness function.

$$y_i = \beta_0 + \beta_1 x_{1i} + \ldots + \beta_p x_{pi} \quad (4)$$

Where:

y: The predicted or expected value

$x_1, \ldots, x_p$: p number of input parameters

i: The index of observations

β_0: Intercept

$\beta_1, \ldots, \beta_p$: The regression coefficients

5.2 Fitness Function

Here both conventionally calculated and estimated porosity values accept to return the fitness value[11]. For the function, root mean square error (RMSE) is employed to estimate the gap between predicted (x_i) and the actual or calculated values (y_i). The average sum square error is estimated by summing of the squares of all residuals of values (n). Numerically, smaller fitness is considered the best solution based on Eq. 5.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(x_1 - y_1)^2}{n}} \quad (5)$$

5.3 Selection

This function accepts the population and fitness values then returns the selected parents. The tournament selection method was used based on results of Saleh &Tuama (2016)[12].

5.4 Genetic Operators

Crossover: Arithmetic crossover (AMXO) method [13] was used to linearly combine two parents (P_1, P_2) to produce two off spring (O_1, O_2) as follows:

$$O_1 = a \cdot P_1 + (1 - a)P_2 \quad (6)$$

$$O_2 = (1 - a)\, P_1 + a \cdot P_2 \quad (7)$$

Where a is random weighting factor, 0.7.

Mutation:a random value from the range of permissible values was assigned to a random gene based on the random resetting mutation to regulate the diversity of the population.

5.5 Evaluating the Model Performance

Each model was evaluated using following functions [14] to see the validity of before making any predictions and to minimize the mistakes.

Cost Function:

$$J(w) = \frac{1}{2m}\sum_{i=1}^{m}(h_w(x_i) - \emptyset_i)^2 \quad (8)$$

$$h_w(x) = w_0 + w_1 x_1 + \cdots + w_n x_n \quad (9)$$

The Total Prediction Error (E^2)

$$E^2 = \frac{1}{N}\sum_{i=1}^{N}(\emptyset_i - w_0 - w_1 x_1 - \ldots - w_n x_n)^2 \quad (10)$$

Where,

$\emptyset$= Actual Porosity

w = weighting coefficient

$h_w(x)$= Hypothesis

N = No. Testing data of Total

m = No. Training data of Total

$x_1, x_2, \ldots, x_n$= Input parameters

6 Scenarios of GA Analyses

The analyses conducted using four sizes of training samples for GA analysis, i.e., 120, 550, 1080 and 2400. Population sizes were raised with 10, 50, 100 and 500 as seen in Table 2.

Table 2: Summary of GA Analyses in this Study.

No. of Population	No. of Data			
	200	920	1800	4002
	No. of Training Data			
	120	550	1080	2400
10	1B	4B	8B	12B
50	2B	5B	9B	13B
100	3B	6B	10B	14B
500		7B	11B	15B

7 Results and Discussion

Figures 3-6 show the fracture porosity (PHIF) analysis results of the study site in Cuu Long Basin as obtained by various GA model sand training datasets.

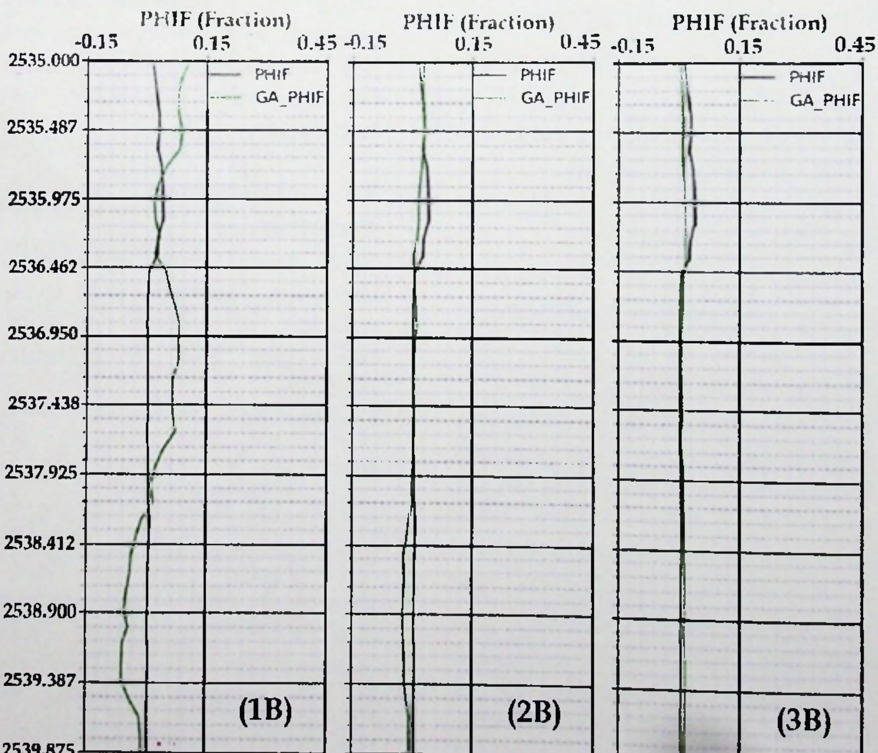

Figure 3: Analysis Results by GA Model 1B, 2B and 3B with 120-Training Dataset.

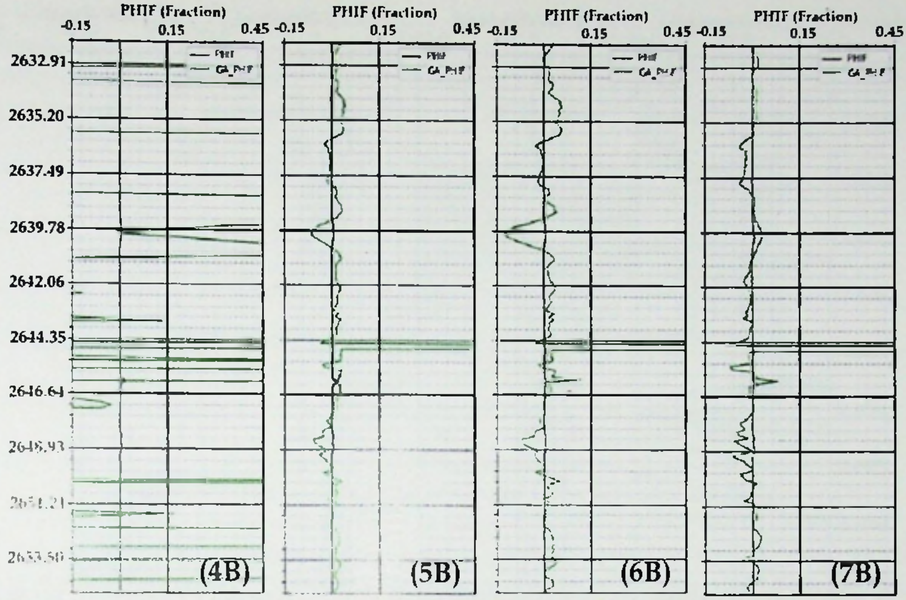

Figure 4: Analysis Results by GA Model 4B, 5B, 6B and 7B with 550-Training Dataset.

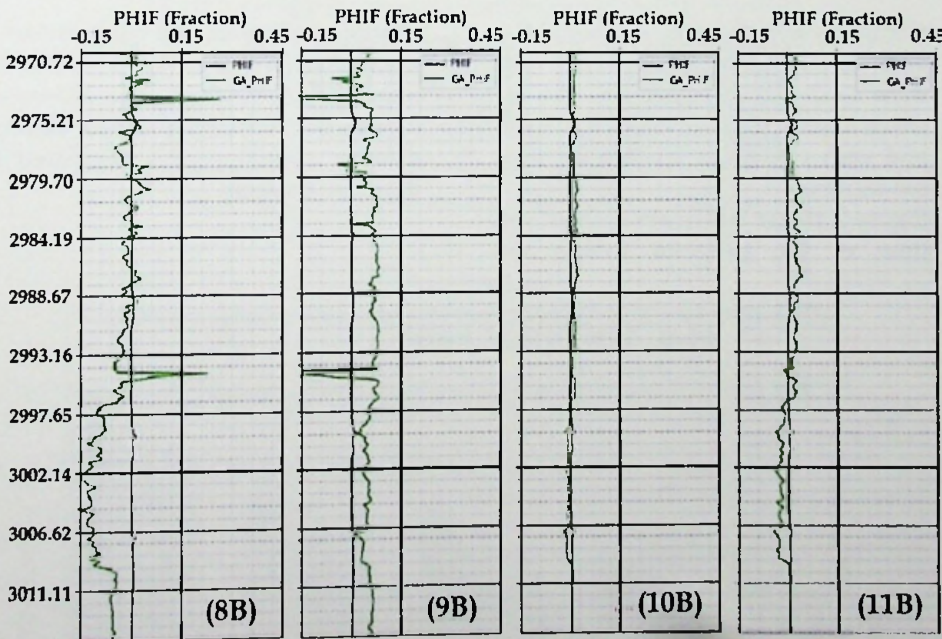

Figure 5: Analysis Results by Model 8B, 9B, 10B and 11B with 1080-Training Dataset.

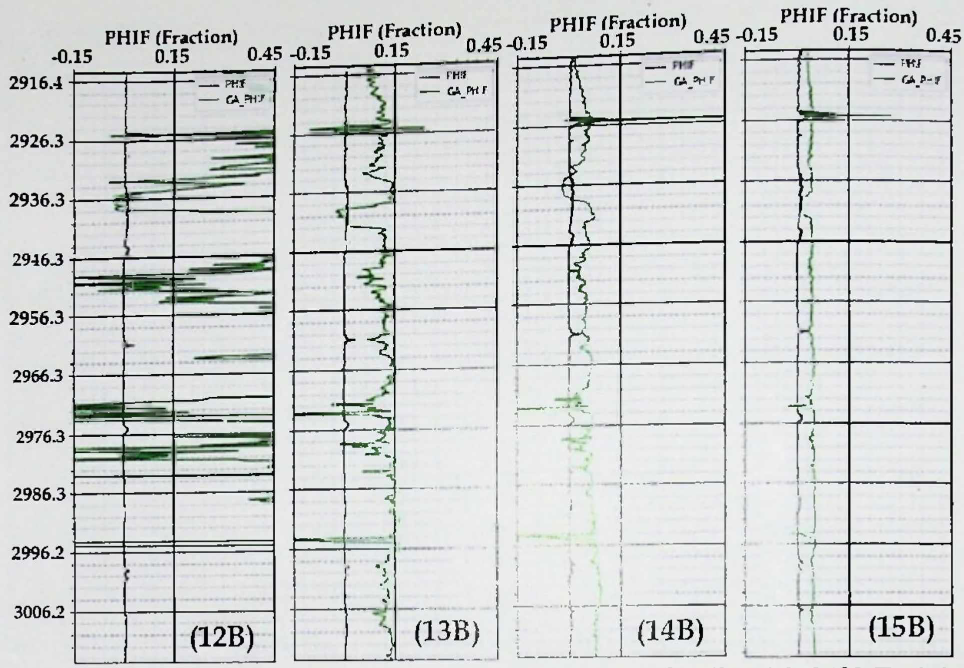

Figure 6: Analysis Results by Model 12B, 13B, 14B and 15B with 2400-Training Dataset.

Comparison of the performance of GA analyses conducted in this study was done as shown in Figure 7-9. The bar charts represent results of GA analysis and each column is labelled by their model name.

Figure 7 shows information about run time of the program during the execution. Overall, the chart shows a gradual upward trend in time with the increase in data sample size. However, models at 120 training sample elapsed a lower execution time compared to the other datasets.

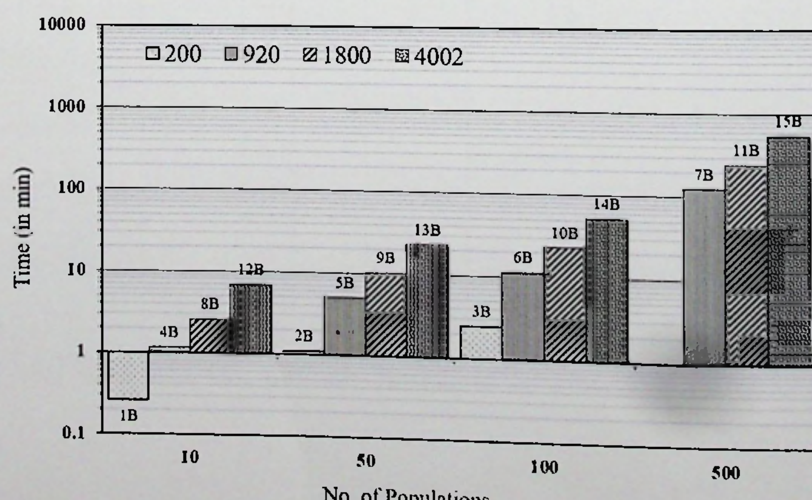

Figure 7: Runtime of Group B vs. No. of Population.

In Figure 8, the 120-training dataset has shown best fitness with increasing populations.

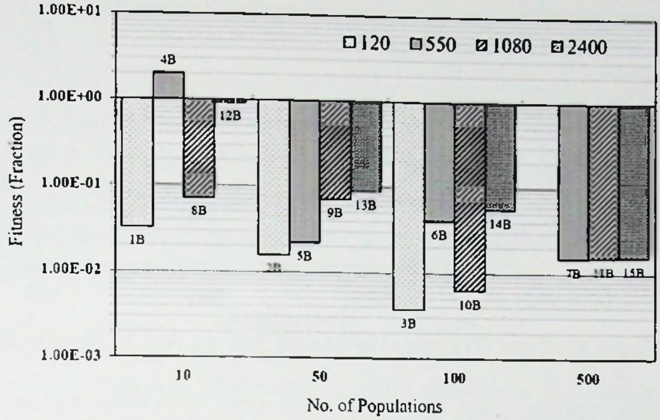

Figure 8: Fitness (Fraction) of Group B vs. No. of Population.

Figure 9 shows information on the total prediction error for testing data. At 100-population, the 120 and 2400-train datasets performed better than the other models.

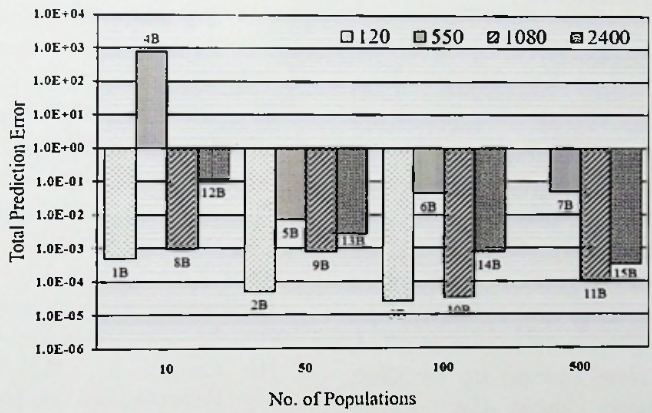

Figure 9: Total Prediction Error of Group B vs. No. of Population.

Furthermore, the models 3B and 10B evaluate using cost function as follows:

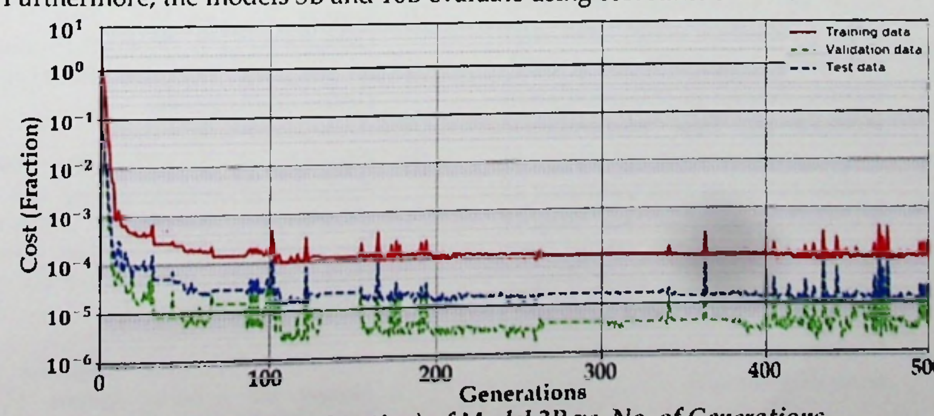

Figure 10: Cost (Fraction) of Model 3B vs. No. of Generations.

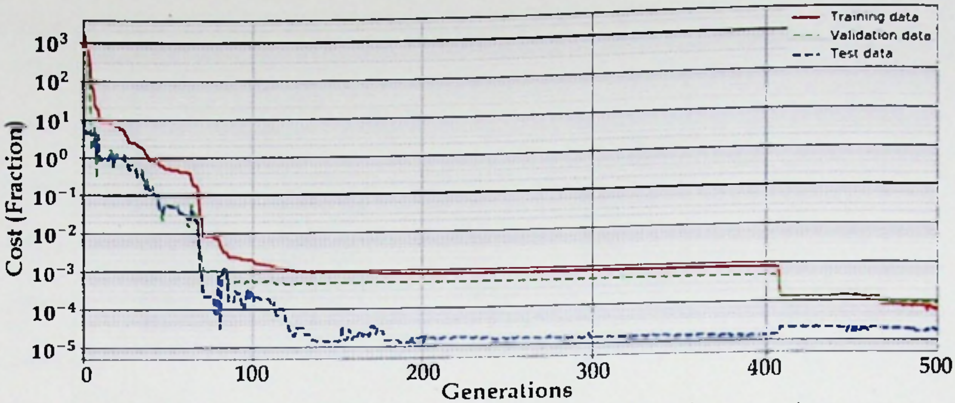

Figure 11: Cost (Fraction) of Model 10B vs. No. of Generations.

In model 10B, training data and validation data cost data met at around 400th generation. Cost of test data is lower than training data in both models.

8 Conclusions and Recommendations

8.1 Conclusions

1. A Genetic Algorithm (GA) program was successfully developed in Python language to analyze the fracture porosity of FGB reservoir. The study site is at CL basin, where there is a deep fractured granite basement reservoir from 2515 to 3015m.
2. Fifteen GA models were studied, which were trained up to 500th generation. Each GA model employed CAL, GR, RHOB, NPHI, LLS and LLD as input data. By considering the least total prediction error, cost and run time, model 10B using 1080-trainng dataset of 100-population showed the best performance. The fracture porosity of fractured granite reservoir was estimated as 0 to 0.03 by conventional method (Elkewidy and Tiab, 1998). The analysis depth was from 2,790m to 3,015m. The GA-estimated fracture porosity was in the range from 0.015 to 0.03.

8.2 Recommendations

1. The dataset in the GA analysis of this study are still limited in size. Future studies should deal with bigger datasets.
2. An integrated GA and ANN analysis would be worth trying to enhance the effectiveness of machine learning-based fracture porosity analysis.

References

[1] Crain, E. R., 2015. Crain's Petrophysical Handbook. [Online]. Available: https://www.spec2000.net [Accessed 25 February 2018].

[2] Bratton T., Can D.V., Duc N.V., Gillespie P., Li, B. (2006). The Nature of Naturally Fractured Reservoirs. *Oilfield Review Journal.*

[3] Elkewidy, T. I. &Tiab, D., 1998. *Application of Conventional Well Logs to Characterize Naturally Fractured Reservoirs with their Hydraulic (Flow) Units.* Alberta, Canada, Society of Petroleum Engineers.

[4] Holland, J. H., 1992. *Adaptation in Natural and Artificial Systems.* MA, USA: MIT Press Cambridge.

[5] Goldberg, D. E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. 2nd ed. Michigan: Addison-Wesley Publishing Company, Inc..

[6] Robinson, D., 2017. The Incredible Growth of Python. [Online]. Available: https://stackoverflow.blog/2017/09/06/incredible-growth-python/ [Accessed 06 September 2017].

[7] PSF, 2019. Python Software Foundation. [Online] Available: https://www.python.org [Accessed 25 April 2019].

[8] Giao, P. H. & Karunaratne, K. S., 2017. *Application of Deep Learning in Predicting Fracture Porosity*. Petro-vietnam, 10(4), pp. 14-22.

[9] Herrera, F., Lozano, M. &Verdegay, J. L., 1998. Tackling Real-Coded Genetic Algorithms, Artificial Intelligence Review, 12, pp.265-319.

[10] Cheusheva, S., 2019. Ablebits.com. [Online]. Available: https://www.ablebits.com/office-addins-blog/2018/08/01/linear-regression-analysis-excel/ [Accessed 23 July 2019].

[11] Saleh, M. H. &Tuama, H. M., 2017. Comparison of Different Types of Fitness Functions to Choose the Appropriate Attributes for Porosity Prediction. *Al-Nahrain Journal for Engineering Sciences*, pp. 737-743.

[12] Saleh, M. H. &Tuama, H. M., 2016. Novel Multi-Gen Multi Parameter Genetic Algorithm Representation for Attributes Selection and Porosity Prediction. *International Journal of Computer Applications*, 141(4), pp. 0975-8887.

[13] Maaita, A. A., Zraqou, J., Hamad, F., & Al-Sewadi, H. A., 2015. *A Generic Adaptive Multi-Gene-Set Genetic Algorithm (AMGA)*, International Journal of Advanced Computer Science and Applications", Vol. 6, No. 5, pp.12-18.

[14] Gudikandula, P., 2018. Toward Data Science. [Online]. Available: https://towardsdatascience.com/coding-deep-learning-for-beginners-linear-regression-part-2-cost-function-49545303d29f