Engineering with Computers (2010) 26:111–118
DOI 10.1007/s00366-009-0140-7

ORIGINAL ARTICLE

Multi expression programming: a new approach to formulation of soil classification

**Amir Hossein Alavi · Amir Hossein Gandomi ·
Mohammad Ghasem Sahab · Mostafa Gandomi**

Received: 10 February 2009 / Accepted: 4 August 2009 / Published online: 1 September 2009

Abstract This paper presents an alternative approach to formulation of soil classification by means of a promising variant of genetic programming (GP), namely multi expression programming (MEP). Properties of soil, namely plastic limit, liquid limit, color of soil, percentages of gravel, sand, and fine-grained particles are used as input variables to predict the classification of soils. The models are developed using a reliable database obtained from the previously published literature. The results demonstrate that the MEP-based formulas are able to predict the target values to high degree of accuracy. The MEP-based formulation results are found to be more accurate compared with numerical and analytical results obtained by other researchers.

Keywords Multi expression programming · Soil classification · Formulation

A. H. Alavi
College of Civil Engineering,
Iran University of Science & Technology, Tehran, Iran
e-mail: ah_alavi@hotmail.com

A. H. Gandomi (✉)
The Highest Prestige Scientific and Professional National Foundation, National Elites Foundation, Tehran, Iran
e-mail: a.h.gandomi@gmail.com

M. G. Sahab
College of Civil Engineering, Tafresh University,
Tafresh, Iran
e-mail: sahab@aut.ac.ir

M. Gandomi
School of Civil Engineering, College of Engineering,
University of Tehran, Tehran, Iran
e-mail: mostafa.gandomi@gmail.com

1 Introduction

One of the main objectives of soil classification is to find the suitability of soil for construction of different structures like dams and embankments subgrade. A range of complex factors affect the naming of soils because soils are not usually available in nature separately as sand, gravel or any other single component but mostly are found as mixture with varying proportions of particles of different size [1]. For instance, sandy clay has most of the properties of clay but contains a significant amount of sand. The soil is given the name of the constituent that appears to have significant influence on its behavior. The behavior of soil mass under load deeply depends on various constituents existing in the mass, degree of density saturation, and environmental conditions. Accordingly, it is idealistic to develop predictive models to able to evaluate the classification of the soil and overcome the limitations of existing classification systems by considering all factors related to the soil formation.

Genetic programming (GP) [2, 3] is a developing subarea of evolutionary algorithms [4] inspired from Darwin's evolution theory. GP may generally be defined as a supervised machine learning technique that searches a program space instead of a data space [2]. Recently, a particular variant of GP that uses a linear representation of chromosomes, namely multi expression programming (MEP) [5] has been proposed. MEP has a special ability to encode multiple computer programs of a problem in a single chromosome. Based on numerical experiments, the MEP approach is able to significantly outperform similar techniques and can be utilized as an efficient alternative to the traditional tree-based GP [6]. Despite the significant advantages of MEP, there has been just some little scientific effort directed at applying it to civil engineering tasks [7–9].

The main purpose of this paper is to utilize the MEP technique to obtain formulas for the determination of soil classification. A comparison between the results of the proposed formulas and those of existing models in the literature was conducted. A reliable database including previously published soil classification test results was utilized to develop the models.

2 Review of previous studies

Artificial neural networks (ANNs) are a branch of computational intelligence techniques [10] that have successfully been applied to the soil classification problem [11–13]. Despite the successful performance of ANNs, they have some fundamental disadvantages that limit them to be used by several researchers. ANNs are black-box models that usually do not give a certain function to calculate the outcome using input values. Hence, a better understanding of the nature of the derived relationship between the different interrelated input and output data is not provided by them. ANN only has final synaptic weights to obtain outcome in parallel manner. ANN approach is appropriate to be used as a part of a computer program and is not suitable for practical calculations.

There has been only limited research with the specific objective of introducing explicit formulas for soil classification by means of ANNs. Rajasekaran and Amalraj [13] built an empirical model using a sequential learning approach (SLA) for single hidden radial basis function (RBF) neuron neural networks proposed by Zhang and Morris [14]. They developed a sequential learning neural network (SLNN) model for the prediction of soil classification. They introduced the following equations based on experimental results and using the values of the weights obtained from neural network training to predict the soil classification (SC):

$$\mathrm{SC_{SLNN}} = 2,653.92e^{-W_1} \tag{1}$$

and

$$\begin{aligned} W_1 = \frac{1}{0.80439^2}\Big[&(x_1 - 1.1843)^2 + (x_2 - 1.0463)^2 \\ &+ (x_3 - 0.8604)^2 + (x_4 - 1.0218)^2 \\ &+ (x_5 - 0.2890)^2 + (x_6 - 2.3476)^2\Big] \end{aligned} \tag{2}$$

where x_1 color of soil; x_2 percentage of gravel; x_3 percentage of sand; x_4 percentage of fine-grained particles; x_5 percentage of liquid limit; x_6 percentage of plastic limit x_1, …, x_6 are the six input parameters to the model. For inputs to the SLNN network the following rule was used for the color of the soil. 0.1: brown; 0.2: brownish gray; 0.3: grayish brown; 0.5: reddish yellow; 0.7: yellowish red.

The output of the network is the classification of soil which is given as: 0.1: clayey soil; 0.2: clay with medium compressibility; 0.3: clay of low compressibility; 0.6: silt with medium compressibility. It should be noted that the required data, used for the training and testing the SLNN models described above, were taken from Suresh [12]. The database has also been utilized in the present study to develop the models.

3 Genetic programming

GP is one of the branches of evolutionary methods that creates computer programs to solve a problem using the principle of Darwinian natural selection. GP was introduced by Koza [2] as an extension of the genetic algorithms, in which programs are represented as tree structures and expressed in the functional programming language LISP [2]. A comprehensive description of GP can be found in [2, 3]. GP has successfully been applied to some of civil engineering problems [15–19].

3.1 Multi expression programming

MEP is a subarea of GP that was developed by Oltean and Dumitrescu [5]. MEP uses linear chromosomes for solution encoding and has a special ability to encode multiple solutions (computer programs) of a problem in a single chromosome. According to the fitness values of the individuals, the best encoded solution is chosen to represent the chromosome. Comparing to the other GP variants that store a single solution in a chromosome, MEP does not increase the complexity of the decoding process except on the cases, where the set of training data is not a priori known [6]. The evolutionary steady-state MEP algorithm starts by the creation of a random population of individuals. In order to evolve the best expression from a data file of inputs and outputs along a specified number of generations, MEP uses the following steps until a termination condition is reached [20]:

- Selecting two parents by using a binary tournament procedure and recombining them with a fixed crossover probability.
- Obtaining two offspring by the recombination of two parents.
- Mutating the offspring and replacing the worst individual in the current population with the best of them (if the offspring is better than the worst individual in the current population).

MEP is represented similar to the way in which C and Pascal compilers translate mathematical expressions into machine code [21]. The number of MEP genes per chromosome is constant and specifies the length of the chromosome. A terminal (an element in the terminal set T) or a function symbol (an element in the function set *F*) is encoded by each gene. A gene that encodes a function includes pointers towards the function arguments. Function parameters always have indices of lower values than the position of that function itself in the chromosome. The first symbol in a chromosome must be a terminal symbol as stated by the proposed representation scheme.

An example of MEP chromosome can be seen below. It should be noted that numbers to the left stand for gene labels that do not belong to the chromosome. Using the set of arithmetic operators as $F = \{+, \times, /\}$ and the set of terminals as $T = \{x_1, x_2, x_3, x_4\}$, the example is given as follows:

0: x_1
1: x_2
2: × 0, 1
3: x_3
4: + 2, 3
5: x_4
6: /4, 5

The translation of MEP individuals into computer programs can be obtained by reading the chromosome top–down starting with the first position. A terminal symbol defines a simple expression and each of function symbols specifies a complex expression obtained by connecting the operands specified by the argument positions with the current function symbol [20]. In the present example, genes 0, 1, 3, and 5 encode simple expressions formed by a single terminal symbol. These expressions are

$$E_0 = x_1,$$
$$E_1 = x_2,$$
$$E_3 = x_3,$$
$$E_5 = x_4,$$

Gene 2 indicates the operation × on the operands located at positions 0 and 1 of the chromosome. Therefore, gene 2 encodes the expression

$$E_2 = x_1 \times x_2.$$

Gene 4 indicates the operation + on the operands located at positions 2 and 3. Therefore, gene 4 encodes the expression

$$E_4 = (x_1 \times x_2) + x_3.$$

Gene 6 indicates the operation / on the operands located at positions 4 and 5. Therefore, gene 6 encodes the expression

$$E_6 = ((x_1 \times x_2) + x_3)/x_4.$$

In order to choose one of these expressions (E_1, …, E_6) as the chromosome representer, multiple solutions in a single chromosome are encoded. Each of the MEP chromosomes encodes a number of expressions equal to the chromosome length (the number of genes). Because of its multi expression representation, each MEP chromosome may be viewed as a forest of trees rather than as a single tree, which is the case of GP. Figure 1 demonstrates the forest of expressions encoded by the previously presented MEP chromosome. Each of these expressions can be considered as a possible solution of a problem. The fitness of each expression encoded in a MEP chromosome is defined as the fitness of the best expression encoded by that chromosome. For solving symbolic regression problems, the fitness of a MEP chromosome (*f*) may be computed using the following equation [6]:

$$f = \min_{i=1,m}\left\{\sum_{j=1}^{n}\left|E_j - O_j^i\right|\right\} \tag{3}$$

where n is the number of fitness cases, E_j is the expected value for the fitness case j, O_j^i is the value returned for the jth fitness case by the ith expression encoded in the current chromosome and m is the number of chromosome genes.

4 Model development

The details of developing the MEP-based models including database description and comparison of performance of the models are presented in the following subsections.

4.1 Database

In the present study, the unified soil classification or IS classification system is considered to verify the obtained results. In practice, soil classification is determined on the basis of existing experimental results. The Bureau of Indian

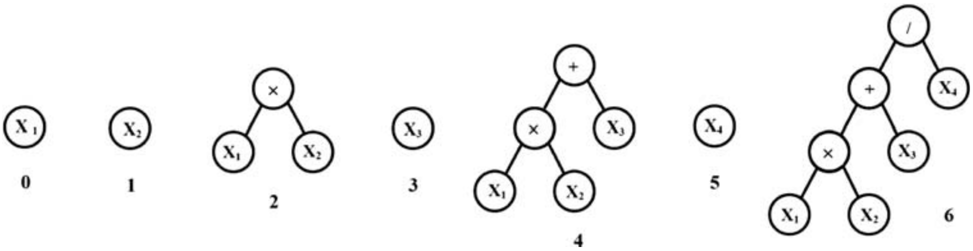

Fig. 1 Expressions encoded by a MEP chromosome represented as *trees*

Standards classifies soils based on color of soil (CS), percentages of gravel (%G), sand (%S), fine-grained particles (%F), liquid limit (LL) and plastic Limit (PL). These six important properties are utilized as the input parameters to the MEP models to predict the soil classification (SC). The following values were assigned to the color of soils:

0.1: Brown;
0.2: Brownish gray;
0.3: Grayish brown;
0.5: Reddish yellow;
0.7: Yellowish red.

Table 1 Database used for model development

Sample no.	CS	G (%)	S (%)	F (%)	LL (%)	PL (%)	SC
Training							
1	0.1	0	80	20	36	22	0.1
2	0.2	2	56	42	30	18	0.1
3	0.1	0	28	72	43	26	0.2
4	0.1	0	27	73	42	25	0.2
5	0.2	0	43	57	34	23	0.3
6	0.2	0	42	58	34	22	0.3
7	0.2	0	45	55	35	23	0.3
8	0.2	3	55	42	31	19	0.1
9	0.2	0	48	52	36	28	0.6
Testing							
10	0.2	4	56	40	30	18	0.1
11	0.1	0	28	72	42	25	0.2
12	0.2	0	44	56	34	22	0.3
13	0.2	0	42	58	35	23	0.3
14	0.1	0	76	24	37	23	0.1
15	0.2	4	54	42	31	18	0.1
16	0.1	0	78	22	36	23	0.1
17	0.1	0	28	72	43	25	0.2

Similar to the SLNN network, the output of MEP is the classification of soil given as below:

0.1: Clayey soil (SL)
0.2: Clay with medium compressibility (CI)
0.3: Clay of low compressibility (CL)
0.6: Silt with medium compressibility (MI)

The database used for model development contains soil classification test results reported by Suresh [12]. The tests were conducted on 17 undisturbed soil samples obtained from different parts of India. The soil samples were collected from trail pits at 1.5–2.0 m depth below ground level. To determine the index properties, disturbed but representative soil samples were also collected from trail pits. The samples were collected using thin-walled samplers satisfying the requirements of IS: 2132 − 1986 [22]. Atterberg limits and grain size distribution characteristics were determined according to relevant IS codes of practice [23, 24]. The soil samples were classified in accord with IS classification system [25]. All these test results are summarized in Table 1. Before the learning process, the input and output parameters were normalized between 0.1 and 0.9. The statistics of different input and output parameters involved in the model development are given in Table 2.

4.2 Model development using MEP

The main goal is to obtain explicit formulas for soil classification (SC) as a function of variables given as follows:

$$SC = f(CS, G, S, F, LL, PL) \tag{4}$$

The six parameters are used for the MEP models as the input variables. Two MEP-based formulas (SC_1, SC_2) were obtained for soil classification considering two different function sets for the MEP runs. The first function set that consists of nearly all operators was used for developing SC_1. The latter includes just addition, subtraction, division,

Table 2 The variables used in model development

Parameters	Min.	Max.	Range	Mean	Median	SD	Kurtosis	N.V.
Inputs								
Color of soil (CS)	0.1	0.2	0.1	–	–	–	–	–
Gravel (%G)	0	4	4	0.76	0.00	1.48	1.11	18
Sand (%S)	27	80	53	48.78	44.94	17.08	−0.43	82
Fine-grained particles (%F)	20	73	53	50.39	54.94	17.46	−0.76	84
Liquid limit (%LL)	30	43	13	35.80	34.99	4.36	−0.77	59
Plastic limit (%PL)	18	28	18	22.52	22.98	2.92	−0.43	34
Output								
Soil classification (SC)	0.1	0.6	0.6	–	–	–	–	–

Min. minimum; *Max.* maximum; *SD* standard deviation; *N.V.* normalization value and multiplication in order to obtain short and very simple formulas (SC_2).

Various parameters involved in the MEP predictive algorithm such as population size, chromosome length, number of generations, tournament size and other parameters that are shown in Table 4. The parameter selection will affect the model generalization capability of MEP. They were selected based on some previously suggested values [8] and also after a trial and error approach. For the analysis, source code of MEP [26] in C++ was modified by the authors to be utilizable for the available problem. The parameter settings for the MEP algorithm are shown in Table 3.

For the analysis, the data sets were divided into training, and testing subsets. Out of the 17 data sets, the first nine values of were taken for training the MEP algorithm and the next eight values were used for testing the generalization capability of the models. In order to evaluate the capabilities of the proposed MEP models, correlation coefficient (R), mean squared error (MSE), and mean absolute error (MAE) were used as follows:

Table 3 Parameter settings for MEP

Parameter	Settings	
	SC_1	SC_2
Function set	+, -, ×, /, $\sqrt{}$, exp, sin, cos	+, −, ×, /
Population size	1,500–2,000	1,500–2,000
Chromosome length	50 genes	50 genes
Number of generations	250	250
Crossover probability	0.5, 0.9	0.5, 0.9
Crossover type	Uniform	Uniform
Mutation probability	0.01	0.01
Terminal set	Problem inputs	Problem inputs

$$R = \frac{\sum_{i=1}^{n}(h_i - \bar{h}_i)(t_i - \bar{t}_i)}{\sqrt{\sum_{i=1}^{n}(h_i - \bar{h}_i)^2 \sum_{i=1}^{n}(t_i - \bar{t}_i)^2}} \tag{5}$$

$$\mathrm{MSE} = \frac{\sum_{i=1}^{n}(h_i - \bar{h}_i)^2}{n} \tag{6}$$

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}\left|h_i - \bar{h}_i\right|}{n} \tag{7}$$

where h_i and t_i, are respectively, the actual and calculated output values for the ith output, $\bar{h}_i$ is the average of the actual outputs, and n is the number of sample.

4.3 Explicit formulation of soil classification and analysis using MEP

Formulations of soil classification for the best test R values by the MEP algorithm are as given below:

$$\begin{aligned}\mathrm{SC}_1 = {} & \frac{\mathrm{PL}^2}{1{,}156}\left(\left(\frac{\mathrm{PL}}{34}\left(\sin\left(\frac{\mathrm{F}\times\mathrm{PL}}{2{,}856}\left(118\frac{\mathrm{CS}\times\mathrm{PL}/84-\mathrm{G}/18}{\mathrm{LL}}\right)\right.\right.\right.\right.\\ & \left.\left.\left.\left.-\frac{\mathrm{G}}{18}\right)^2\right)+2\mathrm{CS}\right)+\mathrm{CS}\right)\end{aligned} \tag{8}$$

$$\mathrm{SC}_2 = \mathrm{CS}\left(\frac{59\mathrm{PL}^3}{9{,}826\mathrm{LL}} - \frac{\mathrm{S}}{82}\right) \tag{9}$$

where SC_1 and SC_2 are the soil classification predictive equations obtained using, respectively, the first and second function sets shown in Table 3. CS, G, S, F, LL, and PL, respectively, denote the color of soil and percentages of gravel, sand, fine-grained particles, liquid limit, and plastic limit. The comparison of MEP predicted and actual soil classification for Eq. 8 is shown in Fig. 2. It can be seen from this figure that Eq. 8 yielded high R values equal to 0.9931 and 0.9823 for the training and testing data, respectively. Figure 3 shows the relevant results obtained by Eq. 9. It can be observed from this figure that Eq. 9

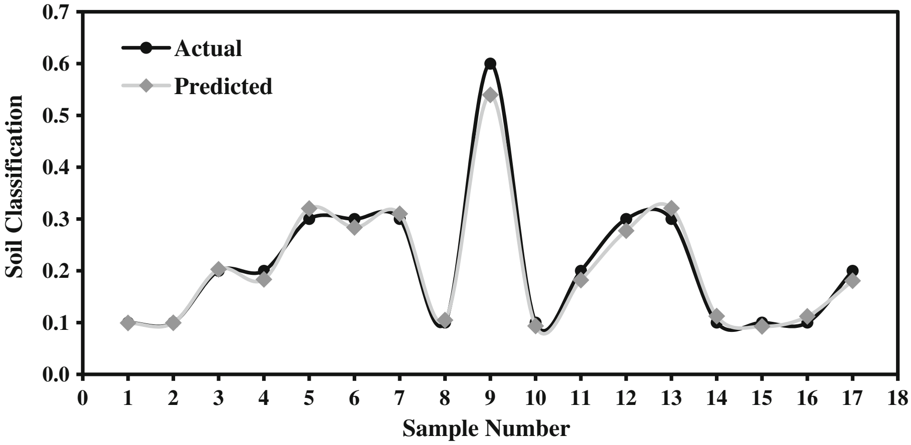

Fig. 2 Results of MEP predicted and actual soil classification obtained by Eq. 8

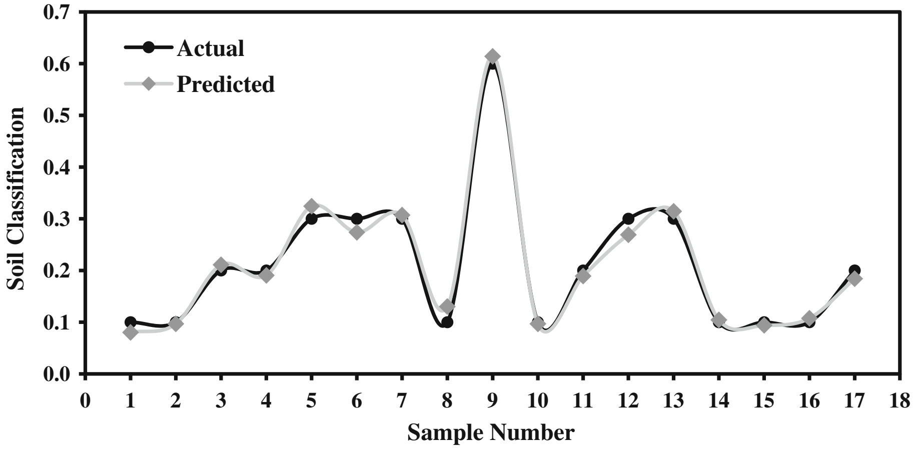

Fig. 3 Results of MEP predicted and actual soil classification obtained by Eq. 9

yielded R values equal to 0.9932 and 0.9871 for the training and testing data, respectively.

In order to evaluate how many times each input appears in a way that contributes to the fitness of the MEP programs that contain them (importance of input parameters), frequency values of input parameters of the predictive models were obtained and presented in Fig. 4. A value of 1.00 in this figure indicates that this input variable appeared in 100% of the best 30 programs evolved by MEP. The frequency values are achieved for the best test R values of MEP runs. From Fig. 4, it can be found that in both of the proposed models, soil classification is more sensitive to CS and PL in comparison with the other inputs.

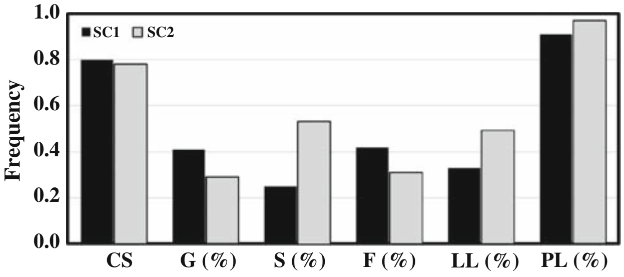

Fig. 4 Frequency values of input parameters of soil classification predictive models

5 Discussion of results

In the present study, two MEP-based formulas were obtained for the classification of soil in terms of CS, %G, %S, %F, LL, and PL. As mentioned previously, R, MSE, and MAE were considered as the target statistical parameters to evaluate the performance of the models. Figure 5 represents a comparison of the ratio between actual and predicted soil classification for different models. Statistical performance of the MEP-based formulations, as well as the SLNN model, are summarized in Table 4. Comparing the performance of the MEP-based equations, it can be observed that the best performance is achieved by Eq. 9 on the training, testing, and all element tests data. On the other hand, Eq. 8, which was developed using the first function set, has taken into account the effects of more parameters compared with Eq. 8.

Comparing the results of MEP and SLNN, it can be seen that both of the formulae obtained by the MEP approach perform superior than the SLNN model on the testing and whole of data. Table 5 shows a comparison between the

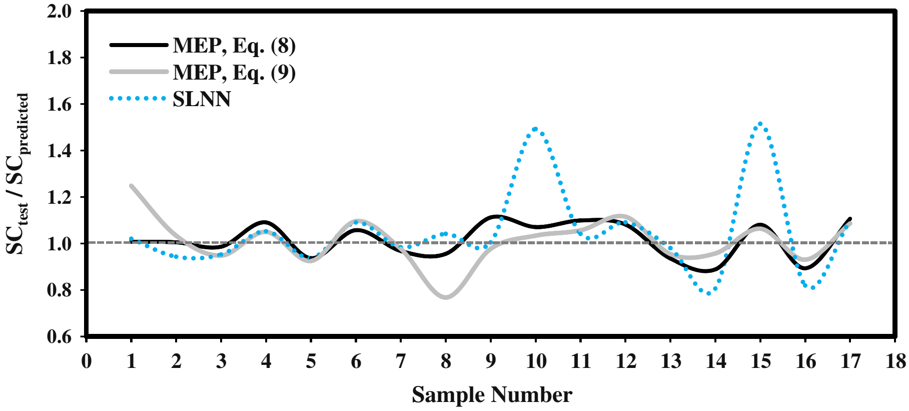

Fig. 5 A comparison of the ratio between actual and predicted soil classification for different models

Table 4 Statistical performance of soil classification prediction models

Models	Training			Testing			All elements		
	R	MSE	MAE	*R*	MSE	MAE	*R*	MSE	MAE
MEP, Eq. 8	0.9931	0.00052	0.01480	0.9823	0.00026	0.01500	0.9907	0.00040	0.01490
MEP, Eq. 9	0.9932	0.00021	0.01150	0.9871	0.00034	0.01610	0.9919	0.00027	0.01390
SLNN	0.9967	0.0013	0.0871	0.9657	0.0086	0.2337	0.9904	0.0103	1.906

Table 5 Comparative analysis of the proposed MEP models with experimental and SLNN results

Sample no.	SC_{test}	SC_1, Eq. 8	SC_{test}/SC_1	SC_2, Eq. 9	SC_{test}/SC_2	SC_{SLNN}	SC_{test}/SC_{SLNN}
Training							
1	0.1	0.099	1.01	0.080	1.25	0.098	1.02
2	0.1	0.099	1.01	0.097	1.03	0.106	0.94
3	0.2	0.203	0.99	0.211	0.95	0.210	0.95
4	0.2	0.183	1.09	0.190	1.05	0.190	1.05
5	0.3	0.320	0.94	0.324	0.93	0.320	0.94
6	0.3	0.284	1.06	0.274	1.10	0.275	1.09
7	0.3	0.310	0.97	0.307	0.98	0.305	0.98
8	0.1	0.105	0.96	0.130	0.77	0.096	1.04
9	0.6	0.540	1.11	0.614	0.98	0.597	1.01
Testing							
10	0.1	0.093	1.07	0.097	1.03	0.067	1.49
11	0.2	0.182	1.10	0.189	1.06	0.192	1.04
12	0.3	0.277	1.08	0.269	1.12	0.275	1.09
13	0.3	0.321	0.94	0.314	0.95	0.306	0.98
14	0.1	0.113	0.89	0.104	0.96	0.124	0.81
15	0.1	0.093	1.08	0.094	1.06	0.066	1.52
16	0.1	0.112	0.89	0.107	0.93	0.122	0.82
17	0.2	0.181	1.11	0.184	1.09	0.182	1.10

results of the proposed MEP formulations, SLNN model, and actual experimental values.

6 Conclusions

In this paper, an application of a particular subset of GP, namely MEP to the soil classification prediction is presented along with its performance comparisons. Two formulas for the classification of soil were obtained by means of MEP and considering two different function sets. A reliable database of the previously published soil classification test results was used for training and testing the prediction models. The MEP-based formulation results were compared with the experimental results and an existing model in the literature namely, SLNN (RBF). The values of performance measures for the models indicate that the proposed MEP models are able to predict the target values to high degree of accuracy. The results also demonstrate that the formulas evolved by MEP outperform those of the SLNN model. In addition to the considerable accuracy of MEP-based prediction equations, they are quite short and very simple and seem to be more practical for use compared with the equations produced by SLNN. However, this investigation revealed that MEP is a very promising approach for its future applications to the formulation of many civil engineering tasks.

References

1. Mostashari M, Muazardalan M, Karimian N, Hosseini HM, Rezai H (2008) Phosphorus fractions of selected calcareous soils of Qazvin province and their relationships with soil characteristics. Am Eurasian J Agric Environ Sci 3(4):547–553
2. Koza JR (1992) Genetic programming: on the programming of computers by means of natural selection. MIT Press, Cambridge
3. Banzhaf W, Nordin P, Keller R, Francone F (1998) Genetic programming–an introduction: on the automatic evolution of computer programs and its application. Morgan Kaufmann, Heidelberg
4. Bäck T (1996) Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford University Press, USA

5. Oltean M, Dumitrescu D (2002) Multi expression programming. Technical report, UBB-01-2002. Babeş-Bolyai University, Cluj-Napoca, Romania
6. Oltean M, Grosşan C (2003) A comparison of several linear genetic programming techniques. Complex Syst 14(4):1–29
7. Baykasoglu A, Gullub H, Canakcı H, Ozbakır L (2008) Prediction of compressive and tensile strength of limestone via genetic programming. Expert Syst Appl 35(1–2):111–123
8. Alavi AH, Gandomi AH, Gandomi M, Sahab MG (2008) A data mining approach to compressive strength of CFRP confined concrete cylinders (submitted)
9. Alavi AH, Gandomi AH (2009) Energy-based numerical correlations for soil liquefaction assessment. Comput Geotech (in press)
10. Schwefel HP, Wegener I, Weinert K (2002) Advances in computational intelligence–theory and practice. Springer, Berlin
11. Levine ER, Kimes DS, Sillito VG (1996) Classifying soil structure using neural networks. Ecol Modell 92(1):101–108
12. Suresh D (2000) Application of neural networks to civil engineering problems. M.E. Thesis, Bharathiar University, Coimbatore, India
13. Rajasekaran S, Amalraj R (2002) Predictions of design parameters in civil engineering problems using SLNN with a single hidden RBF neuron. Comput Struct 80(31):2495–2505
14. Zhang J, Morris AJ (1988) A sequential learning approach for single hidden layer neural networks. Neural Netw 11(1):65–80
15. Alavi AH, Heshmati AAR, Gandomi AH, Askarinejad A, Mirjalili M (2008) Utilisation of computational intelligence techniques for stabilised soil. In: Papadrakakis M, Topping BHV (eds) Proceedings of the sixth international conference on engineering computational technology, Civil-Comp Press, Scotland, paper no. 175
16. Alavi AH, Heshmati AAR, Salehzadeh H, Gandomi AH, Askarinejad A (2008) Soft computing based approaches for high performance concrete. In: Papadrakakis M, Topping BHV (eds) Proceedings of the sixth international conference on engineering computational technology, Civil-Comp Press, Scotland, paper no. 86
17. Gandomi AH, Sahab MG, Alavi AH, Heshmati AAR, Gandomi M, Arjmandi P (2008) Application of a coupled simulated annealing-genetic programming algorithm to the prediction of bolted joints behavior. Am Eurasian J Sci Res 3(2):153–162
18. Gandomi AH, Alavi AH, Kazemi S, Alinia MM (2009) Behavior appraisal of steel semi-rigid joints using linear genetic programming. J Constr Steel Res 65:1738–1750
19. Cevik A, Cabalar AF (2008) Modelling damping ratio and shear modulus of sand–mica mixtures using genetic programming. Expert Syst Appl 36(4):7749–7757
20. Oltean M, Grosşan C (2003) Evolving evolutionary algorithms using multi expression programming. In: Banzhaf W et al (eds) The seventh European conference on artificial life, vol 2801, 14–17 September, LNAI, Springer-Verlag, Dortmund, pp 651–658
21. Aho A, Sethi R, Ullman J (1986) Compilers: principles, techniques and tools. Addison-Wesley, Reading
22. IS: 2132 (1986) (Reaffirmed 1997) Indian standard methods of test for soils, code of practice for thin walled tube sampling of soils. Bureau of Indian Standards, New Delhi
23. IS: 2720 (1985) (Reaffirmed 1995) Indian standard methods of test for soils. Part 5. Determination of liquid and plastic limit. Bureau of Indian Standards, New Delhi
24. IS: 2720 (1985) (Reaffirmed 1995) Indian standard methods of test for soils. Part 4. Grain size analysis. Bureau of Indian Standards, New Delhi
25. IS: 1498 (1970) (Reaffirmed 2002) Indian standard methods of test for soils. Bureau of Indian Standards, New Delhi
26. Oltean M (2004) Multi expression programming source code. http://www.mep.cs.ubbcluj.ro