As computers are taught to think and solve more complex problems, the impact will be felt from the tools engineers use to the skills required.

Αρ. 127 Β – ΙΟΥΝΙΟΣ 2019

ΕΛΛΗΝΙΚΗ
ΕΠΙΣΤΗΜΟΝΙΚΗ
ΕΤΑΙΡΕΙΑ
ΕΔΑΦΟΜΗΧΑΝΙΚΗΣ
& ΓΕΩΤΕΧΝΙΚΗΣ
ΜΗΧΑΝΙΚΗΣ

Τα Νέα της Ε Ε Ε Ε Γ Μ

127
Β

How artificial intelligence will reshape civil engineering

The next industrial revolution is dawning – powered not by steam but artificial intelligence and big data. Arup's Tim Chapman looks at what this will mean for the industry.

The UK construction industry is being challenged to make huge improvements in its performance, in terms of the speed of project delivery, out-turn cost and contributing to reductions in national carbon emissions.

Fortunately, these challenges come at a time when technology is rapidly advancing, and the next industrial revolution is dawning. This second machine age is seeing machines that can think rather than just do.

Just as machine-brawn made vast earth-moving operations so much simpler from the middle of the 19th century, so thinking machines will make many intellectual tasks so much easier in the 21st century.

We are used to computers and their ability to do tasks for us – the speed of communications has vastly sped up over the past quarter century as email has replaced fax and telex. But the power of computers to change our industry is just starting. BIM is already occurring, implemented successfully on many projects, enabling electronic models of new schemes to be collaboratively shared and developed, saving time and improving deliverability. However films like Ex Machina shows us that

(συνέχεια στην σελίδα 3)

Π Ε Ρ Ι Ε Χ Ο Μ Ε Ν Α

(συνέχεια από την πρώτη σελίδα)

current technology can achieve vastly more when the full powers of Artificial Intelligence begin to be effectively harnessed.

Big data revolution

Artificial intelligence will be the next huge wave to engulf our industry – using the vast data banks built up on our projects, supplemented by terabytes of easily accessible data from providers like Apple and Google and a myriad of other data providers that will emerge. Once we start to detect patterns and learn from these experiences and processes, then we will enable computers to be vastly more helpful – and to make our industry vastly more efficient.

Big data style data crunching can reveal hugely insightful patterns that we humans may suspect but can't prove – so machines will hugely assist our engineering judgement. The sorts of revolution that have happened in retail and financial services will be visited on us, for good and for bad.

The good and the bad

The good will be excellent – with a huge number of routine project planning and design tasks made so much slicker, with efficiencies feeding directly into construction processes too. The bad things will be more insidious and will need our professional institutions to ponder hard on how they should influence the future.

Artificial Intelligence will render many of the simpler professional tasks redundant – potentially replacing entirely many of the tasks by which our younger engineers and other professional learn the details of our trade. Experienced engineers probably have less to fear, at least initially, but we need to decide how we may form and develop the experienced engineers of tomorrow, if the tasks for younger engineers of today have been computerised. Later, as expert systems replace human thinking and process improvement steps up several gears, we also will need to reconsider the ethics that underlie our profession, as we code computers to replace much of what we now call engineering judgement exercised by humans.

BIM, drones and autonomous vehicles are today's technologies for which we can foresee many opportunities tomorrow. Artificial Intelligence is tomorrow's technology that will shake every aspect of our profession – mainly for great good, but not always. We need to consciously shape how such changes are introduced.

(Tim Chapman, Director Infrastructure Design Group, Arup, 30 August 2016, https://www.ice.org.uk/news-and-insight/ice-thinks/infrastructure-transformation/how-artificial-intelligence-will-reshape-civil-eng)

ΑΡΘΡΑ

Artificial Intelligence in Geotechnical Engineering: Applications, Modeling Aspects, and Future Directions

Mohamed A. Shahin
Department of Civil Engineering, Curtin University, Perth, WA, Australia

(Chapter 8 of the book **Metaheuristics in Water, Geotechnical and Transport Engineering**, Edited by Xin-She Yang, Amir Hossein, Siamak Talatahari and Amir Hossein Alavi. 2013 Elsevier Inc. London, Waltham, MA, pp. 169-204, https://www.sciencedirect.com/science/article/pii/B9780123982964000088?via%3Dihub)

8.1 Introduction

Geotechnical engineering deals with materials (e.g., soil and rock) that, by their very nature, exhibit varied and uncertain behavior due to the imprecise physical processes associated with the formation of these materials. Modeling the behavior of such materials is complex and usually beyond the ability of most traditional forms of physically based engineering methods. Artificial intelligence (AI) is becoming more popular and particularly amenable to modeling the complex behavior of most geotechnical engineering materials because it has demonstrated superior predictive ability compared to traditional methods. Over the last decade, AI has been applied successfully to virtually every problem in geotechnical engineering. However, despite this success, AI techniques are still facing classical opposition due to some inherent reasons such as lack of transparency, knowledge extraction, and model uncertainty, which will be discussed in detail in this chapter.

Among the available AI techniques are artificial neural networks (ANNs), genetic programming (GP), evolutionary polynomial regression (EPR), support vector machines, M5 model trees, and K-nearest neighbors (Elshorbagy et al., 2010). In this chapter, the focus will be on three AI techniques, including ANNs, GP, and EPR. These three techniques are selected because they have been proved to be the most successful applied AI techniques in geotechnical engineering. Of these, ANN is by far the most commonly used one.

8.2 AI Applications in Geotechnical Engineering

In this section, the applications of the three selected AI techniques (i.e., ANNs, GP, and EPR) are briefly examined. Note that only post-2005 ANN applications are acknowledged, for brevity; interested readers are referred to Shahin et al. (2001), where the pre-2001 applications are reviewed in some detail, and Shahin et al. (2009), where the post-2001 papers are briefly examined.

The behavior of foundations (deep and shallow) in soils is complex, uncertain, and not yet entirely understood. This fact has encouraged many researchers to apply the AI techniques to the prediction of behavior of foundations. For example, ANNs have been used extensively for modeling the axial and lateral load capacities of pile foundations in compression and uplift, including driven piles (Ahmad et al., 2007; Ardalan et al., 2009; Das and Basudhar, 2006; Pal and Deswal, 2008; Shahin, 2010), drilled shafts (Goh et al., 2005; Shahin, 2010), and ground anchor piles (Shahin and Jaksa, 2005, 2006). Predictions of the settlement and load-settlement response of piles have also been modeled by ANNs (Alkroosh and Nikraz, 2011b; Ismail and Jeng, 2011; Pooya Nejad et al., 2009). On the other hand, the prediction of the behavior of shallow foundations has been investigated by ANNs, including settlement estimation (Chen et al., 2006; Shahin et al., 2005a) and bearing capacity (Kuo et al., 2009; Padmini et al., 2008). The GP applications in foundations include the bearing capacity of piles (Alkroosh and Nikraz, 2011a; Gandomi and Alavi, 2012), uplift capacity of suction caissons (Gandomi et al., 2011), and settlement of shallow foundations (Rezania and Javadi, 2007). The single EPR application in foundations is the uplift capacity of suction caissons (Rezania and Javadi, 2008).

Classical constitutive modeling based on elasticity and plasticity theories has only a limited capability to simulate the behavior of geomaterials properly. This is attributed to reasons associated with the formulation complexity, idealization of material behavior, and excessive empirical parameters (Adeli, 2001). In this regard, AI techniques have been proposed as a reliable and practical alternative to modeling the constitutive monotonic and hysteretic behavior of geomaterials, including ANNs (Banimahd et al., 2005; Chen et al., 2010; Fuetal., 2007; Garagaand Latha, 2010; Johari et al., 2011; Najjar and Huang, 2007; Obrzudetal., 2009; Peng et al., 2008; Shahin and Indraratna, 2006), GP (Alkroosh and Nikraz, 2012; Cabalar et al., 2009; Shahnazari et al., 2010), and EPR (Javadi and Rezania, 2009).

Liquefaction during earthquakes is one of the very dangerous ground failure phenomena that can cause a large amount of damage to most civil engineering structures. Although the liquefaction mechanism is well known, the prediction of liquefaction potential is very complex (Baziar and Ghorbani, 2005). This fact has attracted many researchers to investigate the applicability of AI techniques, including ANNs, for predicting liquefaction (Alavi and Gandomi, 2011a; Baziar and Ghorbani, 2005; Hanna et al., 2007a,b; Javadi et al., 2006; Khozaghi and Choobbasti, 2007; Samui and Sitharam, 2011; Shuh-Gi and Ching-Yinn, 2009; Young-Su and Byung-Tak, 2006), GP (Alavi and Gandomi, 2011b, 2012; Baziar et al., 2011; Gandomi and Alavi, 2011, 2012; Javadi et al., 2006; Kayadelen, 2011), and EPR (Rezania et al., 2010, 2011).

Geotechnical properties of soils are controlled by factors such as mineralogy, fabric, and porewater, and the interactions of these factors are difficult to establish solely by traditional statistical methods due to their interdependence (Yang and Rosenbaum, 2002). Based on the application of AI techniques, methodologies have been developed for estimating several soil properties including, for ANNs, preconsolidation pressure and soil compressibility (Celik and Tan, 2005; Jianping et al., 2011; Park and Lee, 2011), shear strength parameters and stress history (Baykasoglu et al., 2008; Byeon et al., 2006; Dincer, 2011; Gunaydin et al., 2010; Kaya, 2009; Kayadelen et al., 2009; Narendara et al., 2006; Tawadrous et al., 2009), soil swelling and swell pressure (Ashayeri and Yasrebi, 2009; Doostmohamadi et al., 2008; Erzin, 2007; Ikizleretal., 2009), lateral earth pressure (Dasand Basudhar, 2005; Uncuoglu et al., 2008), soil permeability (Erzinetal., 2009; Park, 2011), and properties of soil dynamics (Baziar and Ghorbani, 2005; Garcia et al., 2006; Kamatchi et al., 2010; Kogut, 2007; Shafiee and Ghate, 2008; Singhand Singh, 2005; Tsompanakis et al.,2009). For GP, properties include hydraulic conductivity and shear strength (Johari et al., 2006; Kayadelen et al., 2009; Mollahasani et al., 2011; Narendara et al., 2006; Parasuraman et al., 2007), and for EPR, they include soil permeability (Ahangar-Asretal., 2011).

Other applications of ANNs in geotechnical engineering include earth-retaining structures (Goh and Kulhawy, 2005; Kungetal., 2007; Yildiz et al., 2010), dams (Kim and Kim, 2008; Yuetal., 2007), blasting (Lu, 2005), mining (Singh and Singh, 2005), rock mechanics (Cevik et al., 2010; Garcia and Roma, 2009; Ma et al., 2006; Maji and Sitharam, 2008; Sarkaretal., 2010; Singhetal., 2005, 2007; Sitharam et al.,

2008), site characterization (Caglar and Arman, 2007), tunneling and underground openings (Alimoradi et al., 2008; Boubou et al., 2010; Chen et al., 2009; Hajihassani et al., 2011; Neaupane and Adhikari, 2006; Santosetal., 2008; Tsekouras et al., 2010; Yoo and Kim, 2007), slope stability and landslides (Cho, 2009; Das et al., 2011a; Ferentinou and Sakellariou, 2007; Kanungo et al., 2006; Lee et al., 2008; Sakellariou and Ferentinou, 2005; Samui and Kumar, 2006; Wang and Sassa, 2006), deep excavation (Soroush et al., 2006), soil composition and classification (Bhattacharya and Solomatine, 2006; Kurup and Griffin, 2006), soil stabilization (Das et al., 2011b; Liao et al., 2011; Park and Kim, 2011; Tekin and Akbas, 2011), scouring of soils (Firat and Gungor, 2008; Zounemat-Kermani et al., 2009), and soil compaction and permeability (Abdel-Rahman, 2008; Sinha and Wang, 2008; Sivrikaya and Soycan, 2011; Sulewska, 2010). Other applications of GP include dams (Alavi and Gandomi, 2011b), slope stability (Adarsh and Jangareddy, 2010; Alavi and Gandomi, 2011b), tunneling (Alavi and Gandomi, 2011b; Gandomi and Alavi, 2012), soil classification (Alavi et al., 2010), and rock modeling (Feng et al., 2006). Other applications of EPR include slope stability (Ahangar-Asr et al., 2010) and compaction characteristics (Ahangar-Asr et al., 2011).

8.3 Overview of AI

AI is a computational method that attempts to mimic, in a very simplistic way, human cognition capability (e.g., emulating the operation of the human brain at the neural level) to solve engineering problems that have defied solution using conventional computational techniques (Flood, 2008). The essence of AI techniques in solving any engineering problem is to learn by examples of data inputs and outputs presented to them so that the subtle functional relationships among the data are captured, even if the underlying relationships are unknown or the physical meaning is difficult to explain. Thus, AI models are data-driven models (DDMs) that rely on the data alone to determine the structure and parameters that govern a phenomenon (or system) and do not make any assumptions about the physical behavior of the system. This is in contrast to most physically based models that use the first principles (e.g., physical laws) to derive the underlying relationships of the system and usually justifiably simplified with many assumptions, and require prior knowledge about the nature of the relationships among the data. This is one of the main benefits of AI techniques when compared to most physically based empirical and statistical methods.

The AI modeling philosophy is similar to a number of conventional statistical models, in the sense that both are attempting to capture the relationship between a historical set of model inputs and corresponding outputs. For example, imagine a set of *x*-values and corresponding *y*-values in two-dimensional space, where *y=f(x)*. The objective is to find the unknown function *f*, which relates the input variable *x* to the output variable *y*. In a linear regression statistical model, the function *f* can be obtained by changing the slope *tanφ* and intercept *β* of the straight line in Figure 8.1A, so that the error between the actual outputs and the outputs of the straight line is minimized. The same principle is used in AI models. AI can form the simple linear regression model by having one input and one output (Figure 8.1B). AI uses available data to map between the system inputs and the corresponding outputs using machine learning by repeatedly presenting examples of the model inputs and outputs (training) in order to find the function *y=f(x)* that minimizes the error between the historical (actual)outputs and the outputs predicted by the AI model.

If the relationship between *x* and *y* is nonlinear, statistical regression analysis can be applied successfully only if prior knowledge of the nature of the nonlinearity exists. On the contrary, this prior knowledge of the nature of the non-linearity is not required for AI models. In the real world, it is likely that complex and highly nonlinear problems are encountered, and in such situations, traditional regression analyses are inadequate (Gardner and Dorling, 1998). In this section, a brief overview of three selected AI techniques (i.e., ANNs, GP, and EPR) is presented below.

8.3.1 Artificial Neural Networks

ANNs are a form of AI that attempt to mimic the function of the human brain and nervous system. Although the concept of ANNs was first introduced in 1943 (McCulloch and Pitts, 1943), research into applications of ANNs has blossomed since the introduction of the back-propagation training algorithm for feed-forward multilayer perceptrons in 1986 (Rumelhart e tal., 1986). Many authors have described the structure and operation of ANNs (Fausett, 1994; Zurada, 1992). Typically, the architecture of an ANN consists of a series of processing elements (PEs), or nodes, that are usually arranged in layers: an input layer, an output layer, and one or more hidden layers, as shown in Figure 8.2.

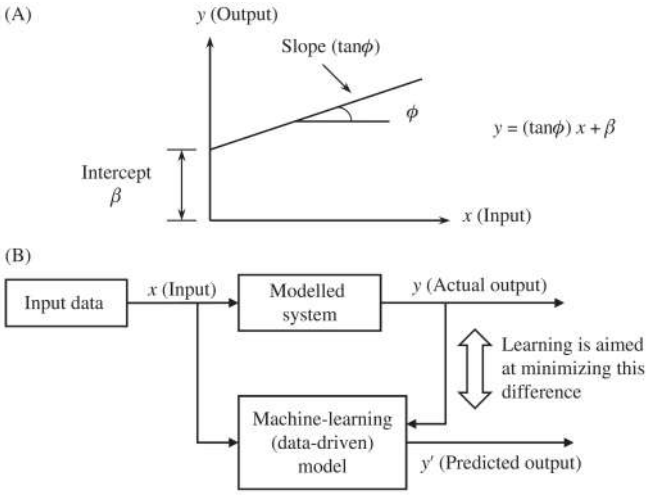
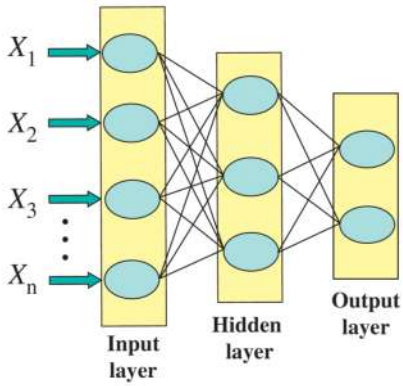
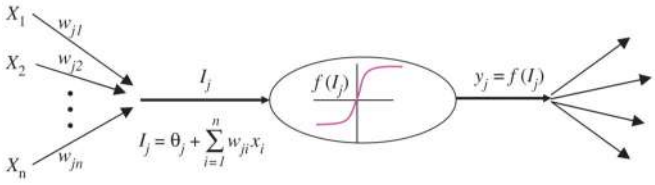


Figure 8.1 Linear regression versus AI modeling. (A) Linear regression modeling (Shahin et al., 2001); (B) AI data-driven modeling. Source: Adapted from Solomatine and Ostfeld (2008).



Artificial neural network



Processing element

Figure 8.2 Typical structure and operation of ANNs (Shahin et al., 2009).

The input from each PE in the previous layer x_i is multiplied by an adjustable connection weight w_{ji}. At each PE, the weighted input signals are summed and a threshold value θ_j is added. This combined input I_j is then passed through a nonlinear transfer function *f(.)* to produce the output of the PE y_j. The output of one PE provides the input to the Pes in the next layer. This process is summarized in Eqs. (8.1) and (8.2) and illustrated in Figure 8.2:

$$I_j = \sum w_{ji} x_i + \theta_j \quad \text{summation} \tag{8.1}$$

$$y_j = f(I_j) \quad \text{transfer} \tag{8.2}$$

The propagation of information in an ANN starts at the input layer, where the input data are presented. The network adjusts its weights on the presentation of a training data set and uses a learning rule to find a set of weights that produces the input and output mapping that has the smallest possible error. This process is called *learning* or *training*. Once the training phase of the model has been successfully accomplished, the performance of the trained model needs to be validated using an independent validation set. The main steps involved in the development of an ANN, as suggested by Maier and Dandy (2000a), are illustrated in Figure 8.3, and several of these steps are discussed in some detail in the following section.

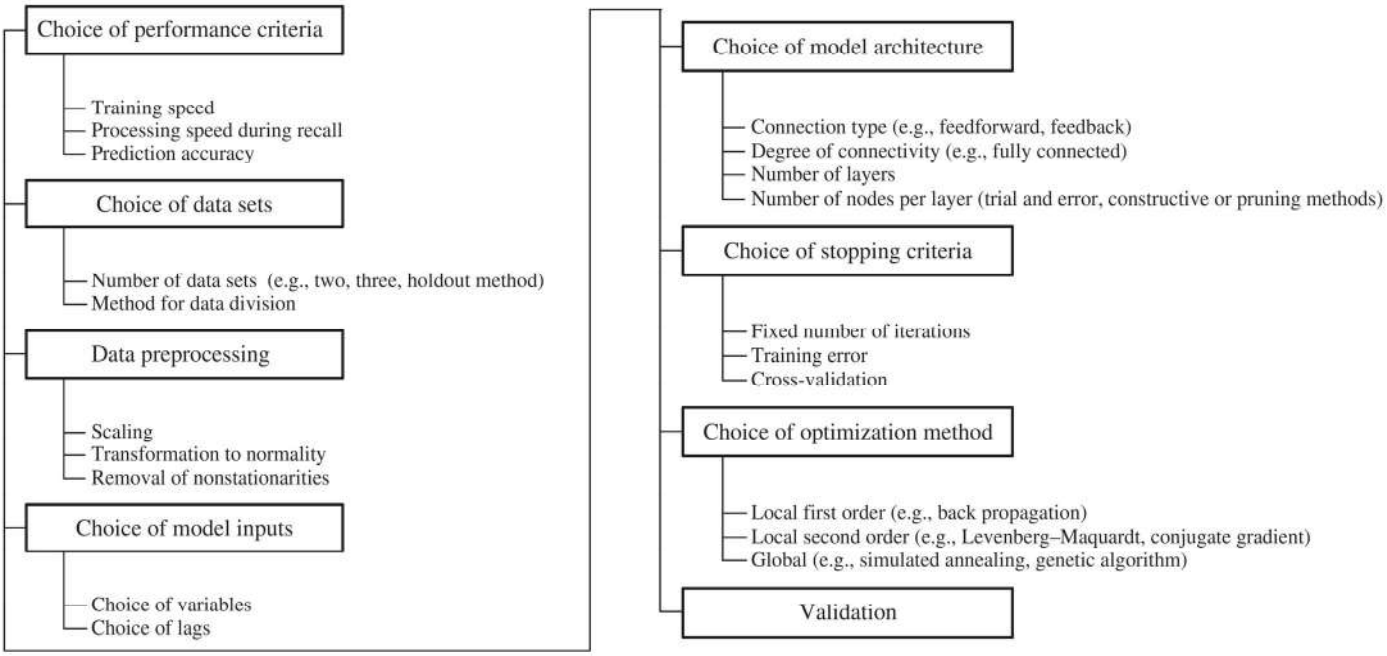


Figure 8.3 The main steps in ANN model development (Maier and Dandy, 2000a).

8.3.2 Genetic Programming

GP is an extension of genetic algorithms (GAs), which are evolutionary computing search (optimization) methods that are based on the principles of genetics and natural selection. In GA, some of the natural evolutionary mechanisms, such as reproduction, crossover, and mutation, are usually implemented to solve function identification problems. GA was first introduced by Holland (1975) and developed by Goldberg (1989), whereas GP was invented by Cramer (1985) and further developed by Koza (1992). The difference between GA and GP is that GA is generally used to evolve the best values for a given set of model parameters (i.e., parameter optimization), whereas GP generates a structured representation for a set of input variables and corresponding outputs (i.e., modeling or programming).

GP manipulates and optimizes a population of computer models (or programs) that have been proposed to solve a particular problem, so that the model that best fits the problem is obtained. A detailed description of GP can be found in many publications (e.g., Koza, 1992), and a brief overview is given herein. The modeling steps by GP start with the creation of an initial population of computer models (also called *individuals* or *chromosomes*) that are composed of two sets (i.e., a set of functions and a set of terminals) that are defined by the user to suit a certain problem. The functions and terminals are selected randomly and arranged in a tree-like structure to form a computer model that contains a root node, branches of functional nodes, and terminals, as shown by the typical example of GP tree representation in Figure 8.4. The functions can contain basic mathematical operators (e.g., +, -, x, /), Boolean logic functions (e.g., AND, OR, and NOT), trigonometric functions (e.g., sin and cos), or any other user-defined functions. The terminals, on the other hand, may consist of numerical constants, logical constants, or variables.

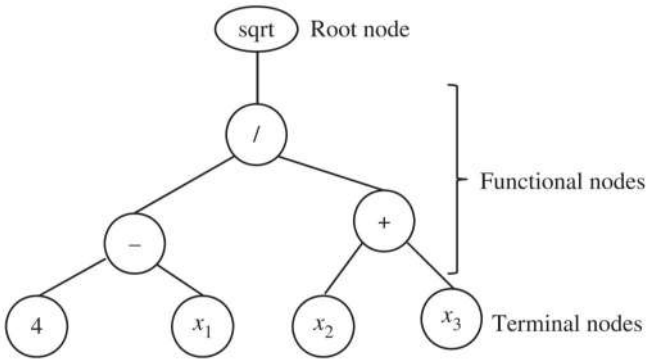


Figure 8.4 A typical example of GP tree representation for the function $[(4-x_1)/(x_2+x_3)]^2$.

Once a population of computer models has been created, each model is executed using available data for the problem at hand, and the model fitness is evaluated depending on how well it is able to solve the problem. For many problems, the model fitness is measured by the error between the output provided by the model and the desired actual output. A

generation of new population of computer models is then created to replace the existing population. The new population is created by applying the following three main operations: reproduction, crossover, and mutation. These three operations are applied on certain proportions of the computer models in the existing population, and the models are selected according to their fitness. Reproduction is copying a computer model from an existing population into the new population without alteration. Crossover is genetically recombining (swapping) randomly chosen parts of two computer models. Mutation is replacing a randomly selected functional or terminal node with another node from the same function or terminal set, provided that a functional node replaces a functional node and a terminal node replaces a terminal node. The evolutionary process of evaluating the fitness of an existing population and producing new population is continued until a termination criterion is met, which can be either a particular acceptable error or a certain maximum number of generations. The best computer model that appears in any generation identifies the result of the GP process. There are currently three variants of GP available in the literature, including linear genetic programming, gene expression programming (GEP), and multi expression programming (Alavi and Gandomi, 2011b). More recently, multi-stage genetic programming (Gandomi and Alavi, 2011) and multi-gene genetic programming (Gandomi and Alavi, 2012) are also introduced. However, GEP is the most commonly used GP method in geotechnical engineering and is thus described in some detail next.

GEP was developed by Ferreira (2001) and utilizes the evolution of mathematical equations that are encoded linearly in chromosomes of fixed length and expressed nonlinearly in the form of expression trees (ETs) of different sizes and shapes. The chromosomes are composed of multiple genes, each of which is encoded in a smaller subprogram or subexpression tree (Sub-ET). Every gene has a constant length and consists of a head and a tail. The head can contain functions and terminals (variables and constants) required to code any expression, whereas the tail solely contains terminals.

The genetic code represents a one-to-one relationship between the symbols of the chromosome and the function or terminal. The process of information decoding from chromosomes to ETs is called *translation*, which is based on sets of rules that determine the spatial organization of the functions and terminals in the ETs and the type of interaction (link) between the Sub-ETs (Ferreira,2001). The main strength of GEP is that the creation of genetic diversity is extremely simplified as the genetic operators work at the chromosome level. Another strength is regarding the unique multigenetic nature of GEP, which allows the evolution of more powerful models/programs composed of several subprograms (Ferreira, 2001).

The major steps in the GEP procedure are schematically represented in Figure 8.5. The process begins with choosing sets of functions *F* and terminals *T* to create randomly an initial population of chromosomes of mathematical equations. One could choose, for example, the four basic arithmetic operators to form the set of functions, i.e., *F* = {+, -, x, /}, and the set of terminals will obviously consist of the independent variables of a particular problem; for example, for a problem that has two independent variables, x_1 and x_2 would be *T* = *{x1, x2}*. Choosing the chromosomal architecture, i.e., the number and length of genes and linking functions (e.g., addition, subtraction, multiplication, and division), is also part of this step. The chromosomes are then given in the form of ETs of different sizes and shapes, and the performance of each individual chromosome is evaluated by comparing the predicted and actual values of presented data. One could measure the fitness f_i of an individual chromosome *i* using the following expression:

$$f_i = \sum_{j=1}^{C_t} (M - |C_{(i,j)} - T_j|), \tag{8.3}$$

where M is the range of selection, $C_{(ij)}$ is the value returned by the individual chromosome *i* for fitness case j (out of C_t fitness cases), and *Tj* is the target value for the fitness case j. There are, of course, other fitness functions available that can be appropriate for different problems. If the desired results (according to the measured errors) are satisfactory, the GEP process is stopped; otherwise, some chromosomes are selected and mutated to reproduce new chromosomes, and the process is repeated for a certain number of generations or until the desired fitness score is obtained.

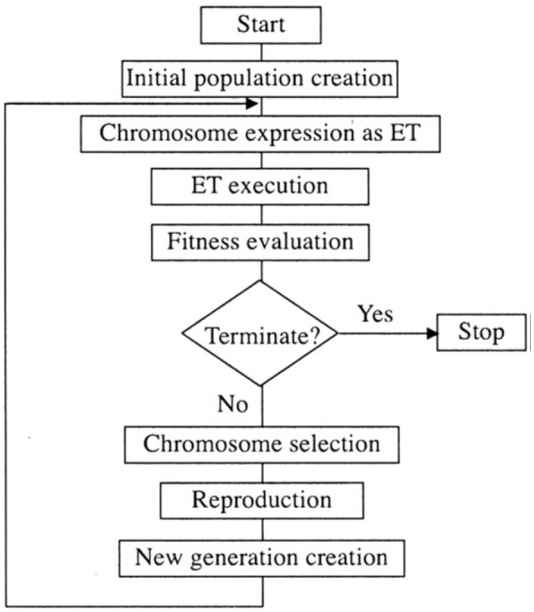


Figure 8.5 The algorithm of GEP (Teodorescu and Sherwood, 2008).

Figure 8.6 shows a typical example of a chromosome with one gene, and its ET and corresponding mathematical equation. It can be seen that, while the head of a gene contains arithmetic and trigonometric functions (e.g., +, -, x, /, √, sin, cos), the tail includes constants and independent variables (e.g.,1, *a, b, c*). The ET is codified reading the ET from left to right in the top line of the tree and from top to bottom.

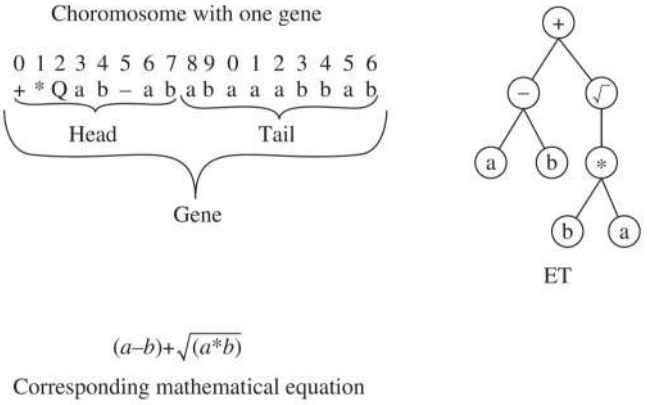


Figure 8.6 Schematic representation of a chromosome with one gene and its ET and corresponding mathematical equation (Kayadelen, 2011).

8.3.3 Evolutionary Polynomial Regression

EPR is a hybrid regression technique based on evolutionary computing that was developed by Giustolisi and Savic

(2006). It constructs symbolic models by integrating the soundest features of numerical regression, with GP and symbolic regression (Koza, 1992). This strategy provides the information in symbolic form, as usually defined in the mathematical literature. The following two steps roughly describe the underlying features of the EPR technique, which aimed to search for polynomial structures representing a system. In the first step, the selection of exponents for polynomial expressions is carried out, employing an evolutionary searching strategy by means of GAs (Goldberg, 1989). In the second step, numerical regression using the least squares method is conducted, aiming to compute the coefficients of the previously selected polynomial terms. The general form of expression in EPR can be presented as follows (Giustolisi and Savic, 2006):

$$y = \sum_{j=i}^{m} F(X, f(X), a_j) + a_0 \qquad (8.4)$$

where y is the estimated vector of output of the process, m is the number of terms of the target expression, F is a function constructed by the process, X is the matrix of input variables, f is a function defined by the user, and a_j is a constant. A typical example of EPR pseudo-polynomial expression that belongs to the class of Eq. (8.4) is as follows (Giustolisi and Savic, 2006):

$$\hat{Y} = a_0 + \sum_{j=i}^{m} a_j \cdot (X_1)^{ES(j,1)} \cdots (X_k)^{ES(j,k)} \cdot f[(X_1)^{ES(j,k+1)} \cdots (X_k)^{ES(j,2k)}] \qquad (8.5)$$

where $\hat{Y}$ is the vector of target values, m is the length of the expression, a_j is the value of the constants, X_i is the vector(s) of the k candidate inputs, ES is the matrix of exponents, and f is a function selected by the user.

EPR is suitable for modeling physical phenomena, based on two features (Savic et al., 2006): (1) the introduction of prior knowledge about the physical system/process, to be modeled at three different times, namely before, during, and after EPR modeling calibration; and (2) the production of symbolic formulas, enabling data mining to discover patterns that describe the desired parameters. In EPR feature (1), before the construction of the EPR model, the modeler selects the relevant inputs and arranges them in a suitable format according to their physical meaning. During the EPR model construction, model structures are determined by following user-defined settings such as general polynomial structure, user-defined function types (e.g., natural logarithms, exponentials, and tangential hyperbolics) and searching strategy parameters. The EPR starts from true polynomials and also allows for the development of nonpolynomial expressions containing user-defined functions (e.g., natural logarithms). After EPR model calibration, an optimum model can be selected from among the series of models returned. The optimum model is selected based on the modeler's judgment, in addition to statistical performance indicators such as the coefficient of determination. A typical flow diagram of the EPR procedure is shown in Figure 8.7, and a detailed description of the technique can be found in Giustolisi and Savic (2006).

8.3.4 Current Development and Future Directions in the Utilization of AI

Based on the author's experience, there are several factors in the use of AI techniques that need to be systematically investigated when developing AI models, so that model performance can be improved. These factors include the determination of adequate model inputs, data division, data preparation, model validation, model robustness, model transparency and knowledge extraction, model extrapolation, and model uncertainty. Some of these factors have received recent attention; others require further research. Each of these is discussed below.

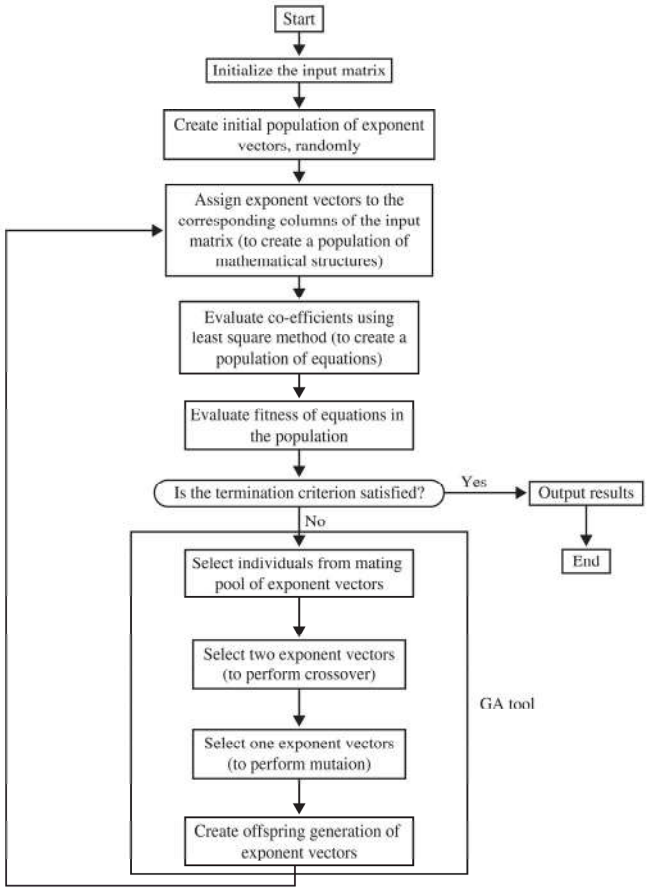


Figure 8.7 A typical flow diagram of the EPR procedure (Rezania et al., 2011).

Determination of Model Inputs

An important step in developing AI models is to select the model input variables that have the most significant impact on model performance. A good subset of input variables can substantially improve model performance. Presenting as large a number of input variables as possible to AI models usually increases the model size, resulting in a decrease in processing speed and model efficiency. A number of techniques have been suggested in the literature to assist with the selection of input variables. An approach that is usually utilized in the field of geotechnical engineering is that appropriate input variables can be selected in advance based on *a priori* knowledge. Another approach used by some researchers (Goh, 1994; Najjaretal., 1996; Ural and Saka, 1998) is to develop many models with different combinations of input variables and to select the model that has the best performance. A step wise technique described by Maier and Dandy (2000b) can also be used in which separate models are trained, each using only one of the available variables as model inputs, and the model that performs the best is then retained, combining the variable that results in the best performance with each of the remaining variables. This process should be repeated for an increasing number of input variables, until the addition of additional variables results in no further improvement in model performance. Another useful approach is to employ a GA to search for the best sets of input variables (NeuralWare, 1997). For each possible set of input variables chosen by the GA, a model is trained and used to rank different subsets of possible inputs. A set of input variables derives its fitness from the model error obtained based on those variables. The adaptive spline modeling of observation data algorithm proposed by Kavli (1993) is also a useful technique that can be used for developing parsimonious models by automatically selecting a combination of

model input variables that have the most significant impact on the outputs.

A potential shortcoming of these approaches is that they are model based. In other words, the determination as to whether a parameter input is significant or not is dependent on the error of a trained model, which is not only a function of the inputs but also model structure and calibration. This can potentially obscure the impact of different model inputs. In order to overcome this limitation, model-free approaches can be utilized, which use linear dependence measures, such as correlation, or nonlinear measures of dependence, such as mutual information, to obtain the significant model inputs prior to developing the AI models (Bowden et al., 2005, May et al., 2008).

Data Division

As described earlier, AI models are similar to conventional statistical models in the sense that model parameters are adjusted in the model calibration phase (training) so as to minimize the error between model outputs and the corresponding measured values for a particular data set (the training set). AI models perform best when they do not extrapolate beyond the range of the data used for calibration. Therefore, the purpose of AI models is to nonlinearly interpolate (generalize) in high-dimensional space between the data used for calibration. Unlike conventional statistical models, AI models generally have a large number of model parameters and can therefore overfit the training data, especially if the training data are noisy. In other words, if the number of degrees of freedom of the model is large compared with the number of data points used for calibration, the model might no longer fit the general trend, as desired, but might learn the idiosyncrasies of the particular data points used for calibration leading to memorization, rather than generalization. Consequently, a separate validation set is needed to ensure that the model can generalize within the range of the data used for calibration. It is a common practice to divide the available data into two subsets: a training set, to construct the model, and an independent validation set, to estimate the model performance in a deployed environment. Usually, two-thirds of the data are suggested for model training and one-third for validation (Hammerstrom, 1993). A modification of this data division method is cross-validation in ANNs (Stone, 1974), in which the data are divided into three sets: training, testing, and validation. The training set is used to adjust the model parameters, whereas the testing set is used to check the performance of the model at various stages of training and to determine when to stop training to avoid overfitting. The validation set is used to estimate the performance of the trained network in the deployed environment. In an attempt to find the optimal proportion of the data to use for training, testing, and validation in ANN models, Shahin et al. (2004) investigated the impact of the proportion of data used in various subsets on model performance for a case study of settlement prediction of shallow foundations and found that there is no clear relationship between the proportion of data for training, testing, and validation and model performance; however, they found that the best result was obtained when 20% of the data were used for validation and the remaining data were divided into two parts, 70% for training and 30% for testing.

In many situations, the available data are small enough to be solely devoted to model training, and collecting any more data for validation is difficult. In this situation, the leave-k-out method (Masters, 1993), which involves holding back a small fraction of the data for validation and using the rest of the data for training, can be used. After training, the performance of the trained network has to be estimated with the aid of the validation set. A different small subset of data is held back and the model is trained and tested again. This process is repeated many times with different subsets until an optimal model can be obtained from the use of all of the available data.

In the majority of AI applications in geotechnical engineering, the data were divided into their subsets on an arbitrary basis. However, some studies have found that the way the data are divided can have a significant impact on the results obtained (Tokar and Johnson, 1999). As AI models have difficulty extrapolating beyond the range of the data used for calibration, in order to develop the best AI models, given the available data, all of the patterns that are contained in the data need to be included in the calibration set. For example, if the available data contain extreme data points that were excluded from the calibration data set, the model cannot be expected to perform well because the validation data will test the model's extrapolation ability rather than its interpolation ability. If all of the patterns that are contained in the available data are contained in the calibration set, the toughest evaluation of the generalization ability of the model is if all the patterns (and not just a subset) are contained in the validation data. In addition, if cross- validation is used in ANN models, the results obtained using the testing set have to be representative of those obtained using the training set, as the testing set is used to decide when to stop training or, for example, which model architecture or learning rate is optimal. Consequently, the statistical properties (e.g., mean and standard deviation) of the various data subsets (e.g., training, testing, and validation) need to be similar to ensure that each subset represents the same statistical population (Masters, 1993). If this is not the case, it may be difficult to judge the validity of AI models.

This fact has been recognized for some time (ASCE, 2000; Maier and Dandy, 2000b; Masters, 1993), and several studies have used ad hoc methods to ensure that the data used for calibration and validation have the same statistical properties (Braddock et al., 1998; Campolo et al., 1999; Ray and Klindworth, 2000; Tokar and Johnson, 1999). Masters (1993) strongly confirms the above strategy of data division as he says, "if our training set is not representative of the data on which the network will be tested, we will be wasting our time." However, it was not until a few years ago that systematic approaches for data division have been proposed in the literature. Bowden et al. (2002) used a GA to minimize the difference between the means and standard deviations of the data in the training, testing, and validation sets. While this approach ensures that the statistical properties of the various data subsets are similar, there is still a need to choose which proportion of the data to use for training, testing, and validation. Kocjancic and Zupan (2000) and Bowden et al. (2002) used a self-organizing map (SOM) to cluster high-dimensional input and output data in two-dimensional space and divided the available data so that values from each cluster were represented in the various data subsets. This ensures that data in the different subsets were representative of each other and had the additional advantage that there was no need to decide what percentage of the data to use for training, testing, and validation. The major shortcoming of this approach is that there are no guidelines for determining the optimum size and shape of the SOM (Cai et al., 1994; Giraudel and Lek, 2001). This has the potential to have a significant impact on the results obtained, as the underlying assumption of the approach is that the data points in one cluster provide the same information in high-dimensional space. However, if the SOM is too small, there may be significant intracluster variation. Conversely, if the map is too large, too many clusters may contain single data points, making it difficult to choose representative subsets. To overcome the problem of determining the optimum size of clusters associated with using SOMs, Shahin et al. (2004) introduced a data division approach that utilizes a fuzzy clustering technique so that data division can be carried out in a systematic manner.

Data Preparation

Data preparation is the process of presenting the data in a suitable form before they are presented to the AI techniques. Once the available data have been divided into their subsets (e.g., training and validation), it is important to preprocess the data to ensure that all variables receive equal attention during training. Preprocessing of the data also usually speeds up the learning process, and it can be in the form of data scaling or transformation (Masters, 1993). Scaling of the data is not necessary but almost always recommended (Masters, 1993). Transformation of the data into normal distribution or some known forms (e.g., linear, log, and exponential) may be helpful to improve the performance of AI models. The influence of data transformation was undertaken in a study carried out by Bowden et al. (2003) using

Different transformation methods, including linear, logarithmic, and seasonal transformations, histogram equalization, and a transformation to normality. In this study, it was found that the model using the linear transformation resulted in the smallest error, whereas more complex transformations did not improve model performance. Moreover, empirical trials carried out by Faraway and Chatfield (1998) showed that the model fits were the same, regardless of whether raw or transformed data were used. The author's own experience in geotechnical engineering is that data scaling is useful, but data transformation does not improve model performance.

Model Validation

Once the training phase of the model has been successfully accomplished, the performance of the trained model should be validated. The purpose of the model validation phase is to ensure that the model has the ability to generalize within the limits set by the training data in a robust fashion, rather than simply having memorized the input-output relationships that are contained in the training data. The approach that is generally adopted in the literature to achieve this is to test the performance of trained AI models on an independent validation set that has not been used as part of the model building process. If such performance is adequate, the model is deemed to be able to generalize and is considered to be robust.

The choice of a suitable error function to investigate model validation is quite important, and the main measures that are often used in the literature to evaluate the performance of AI models include the coefficient of correlation, r; the root mean squared error, RMSE; and the mean absolute error, MAE. The formulas of these measures are as follows:

$$r = \frac{\sum_{i=1}^{N}(O_i - \overline{O})(P_i - \overline{P})}{\sqrt{\sum_{i=1}^{N}(O_i - \overline{O})^2 \sum_{i=1}^{N}(P_i - \overline{P})^2}} \tag{8.6}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(O_i - P_i)^2}{N}} \tag{8.7}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|O_i - P_i| \tag{8.8}$$

where N is the number of data points presented to the model; O_i and P_i are the observed and predicted outputs, respectively; and O and P are the mean of observed and predicted outputs, respectively.

The coefficient of correlation, r, is a measure that is used to determine the relative correlation and the goodness-of-fit between the predicted and the observed data. Smith (1986) suggested the following guide for values of $|r|$ between 0.0 and 1.0:

- $|r| \geq 0.8$—Strong correlation exists between two sets of variables,
- $0.2 < |r| < 0.8$—Correlation exists between the two sets of variables, and
- $|r| \leq 0.2$—Weak correlation exists between the two sets of variables.

However, Das and Sivakugan (2010) argued that the use of r could be misleading because some times higher values of r may not necessarily indicate better model performance due to the tendency of the model to deviate toward higher or lower values, particularly when the data range is very wide and most of the data are distributed about their mean. It was suggested that the coefficient of efficiency, E, proposed by Nash and Sutcliffe (1970), can give an unbiased estimate and would be a better measure for model performance. E is calculated as follows:

$$E = 1 - \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}(O_i - \overline{O})^2} \tag{8.9}$$

According to Eq. (8.9), E may range from -∞ to 1.0, where a value of 90% and above indicates very satisfactory performance and a value below 80% indicates unsatisfactory performance. However, Legates and McCabe (1999) raised the issue that E is oversensitive to extreme values (caused by squaring the difference terms), and introduced the modified coefficient of efficiency, E_1, which uses the absolute differences rather than their squares and can be computed as follows:

$$E_1 = 1 - \frac{\sum_{i=1}^{N}|(O_i - P_i)|}{\sum_{i=1}^{N}|(O_i - \overline{O})|} \tag{8.10}$$

The RMSE is the most popular error measure and has the advantage that large errors receive much greater attention than small errors (Hecht-Nielsen, 1990). However, as indicated by Cherkassky et al. (2006), there are situations when RMSE cannot guarantee that the model performance is optimal. Moreover, it was also argued by Das and Sivakugan (2010) that RMSE reflects only the short-term (overall) performance of the model information, showing the overall difference between the predicted and the measured values. Das and Sivakugan (2010) suggested that the use of the normalized mean biased error, NMBE, provides information with respect to overestimation or underestimation predictions and thus can give a better estimation in relation to the long-term model performance. In contrast with RMSE, MAE eliminates the emphasis given to large errors. Both RMSE and MAE are desirable when the evaluated output data are smooth or continuous (Twomey and Smith, 1997).

It is advised by Guven and Aytek (2009) that the combined use of RMSE, E, and E1 provides a sufficient assessment of AI model performance and allows comparison of the accuracy of different AI modeling approaches. On the other hand, Elshorbagy et al. (2010) suggested that four different error statistics including RMSE, mean absolute relative error (MARE), mean bias (MB), and coefficient of correlation (r), along with the visual comparison between the observed and the predicted output values, are sufficient to reveal any significant differences among the various modeling techniques with regard to their prediction accuracy. However, Elshorbagy et al. (2010) mentioned that sometimes conflicting results may arise due to the use of various measures and proposed a new error measure that combines the effects of the above –mentioned four error measures in one indicator. The new indicator is called the ideal pointer or (IPE), and it is calculated as follows (Elshorbagy et al. 2010):

$$IPE_{ij} = \left\{0.25\left[\left(\frac{RMSE_{ij}-0.0}{\max\ RMSE_{ij}}\right)^2 + \left(\frac{MARE_{ij}-0.0}{\max\ MARE_{ij}}\right)^2 + \left|\frac{MB_{ij}-0.0}{\max|MB_{ij}|}\right|^2 + \left(\frac{r_{ij}-1.0}{1/\max\ r_{ij}}\right)^2\right]\right\}^{1/2} \quad (8.11)$$

where *i* and *j* denote model (*i*) and technique (*j*), respectively; and MARE and MB are calculated as follows:

$$MARE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{O_i - P_i}{O_i}\right| \quad (8.12)$$

$$MB = \frac{1}{N}\sum_{i=1}^{N}(O_i - P_i) \quad (8.13)$$

The IPE relies on identifying the ideal point in the four-dimensional error (space) that a model aims to reach. The ideal point should have the following coordinates: RMSE = 0.0, MAR = 0.0, MB =0.0, and r = 1.0. The IPE measures how far a model performance is from the ideal point. All individual error measures are given equal relative weights and normalized using their maximum error, so the final IPE value ranges from 0.0 (for the best model performance) to 1.0 (for the worst model performance).

Model Robustness

Model robustness is the predictive ability of AI models to generalize over a range of data similar to that used for model training. With regard to ANNs, Kingston et al. (2005b) stated that if "ANNs are to become more widely accepted and reach their full potential…, they should not only provide a good fit to the calibration and validation data, but the predictions should also be plausible in terms of the relationship modeled and robust under a wide range of conditions," and that "while ANNs validated against error alone may produce accurate predictions for situations similar to those contained in the training data, they may not be robust under different conditions unless the relationship by which the data were generated has been adequately estimated. "This agrees with the investigation into the robustness of ANNs carried out by Shahin et al. (2005b) for a case study of predicting the settlement of shallow foundations on granular soils. Shahin et al. (2005b) found that good performance of ANN models on the data used for model calibration and validation does not guarantee that the models will perform in a robust fashion over a range of data similar to those used in the model calibration phase. For this reason, Shahin et al. (2005b) proposed a method to test the robustness of the predictive ability of ANN models by carrying out a parametric study to investigate the response of ANN model outputs to changes in its inputs. The robustness of the model can then be determined by examining how well model predictions are in agreement with the known underlying physical processes of the problem in hand over a range of inputs. Shahin et al. (2005b) presented two different ANN models, which have the performance given in Table 8.1. Both the models were developed using the same software, model parameters, and architecture (i.e., five inputs: footing width, applied pressure, average Standard Penetration Test (SPT) blow count, footing geometry, and embedment ratio, and one hidden layer with two nodes and a single output: a foundation settlement), except that the models were optimized with different sets of random starting weights. It can be seen from Table 8.1 that both the models perform very well when assessed against traditional measures such as the coefficient of correlation, r, RMSE, and MAE. In the absence of any further information, one would normally adopt either of the two models and use it for predictive purposes within the range of the input data used to train the models.

Table 8.1 Performance of the ANN Models Developed by Shahin et al. (2005b)

Model No.	Data Set	r	RMSE (mm)	MAE (mm)
1	Training	0.92	10.8	7.4
	Testing	0.94	8.4	5.8
	Validation	0.88	12.9	9.8
2	Training	0.94	9.1	6.3
	Testing	0.94	9.1	6.8
	Validation	0.89	11.8	9.6

Figure 8.8 shows the results of the parametric study performed to assess the gen- realization ability of both models. In order to carry out the parametric study, all input variables except one were fixed to the mean values used for training and a set of synthetic data (whose values lie between the minimum and the maximum values used for model training) were generated for the single input that was allowed to vary. The synthetic data were generated by increasing their values in increments equal to 5% of the total range between the minimum and the maximum values. These input values were then entered into both ANN models and the corresponding outputs were obtained. The robustness of the models was then determined by examining how well the predicted output (in this case, the footing settlement) agrees with the known underlying physical processes over the range of inputs examined. It can be seen that the results obtained for Model-1 agree with what one would expect based on the known physical behavior of the settlement of shallow foundations on granular soils. For example, in Figure 8.8A, B, and D, there is an increase in the predicted settlement, in a relatively consistent and smooth fashion, as the footing width, footing net applied pressure, and footing geometry, respectively, increase. On the other hand, in Figure 8.8C and E, the predicted settlement decreases, also in a consistent and smooth fashion, as the average SPT blow count and footing embedment ratio, respectively, increase. In contrast, it can be seen from Figure 8.8 that the results obtained for Model-2 have an unexpected shape that is difficult to justify from a physical understanding of footing settlement. For example, there are abrupt changes in the predicted settlement in some instances and no change in predicted settlement for a range of inputs in others.

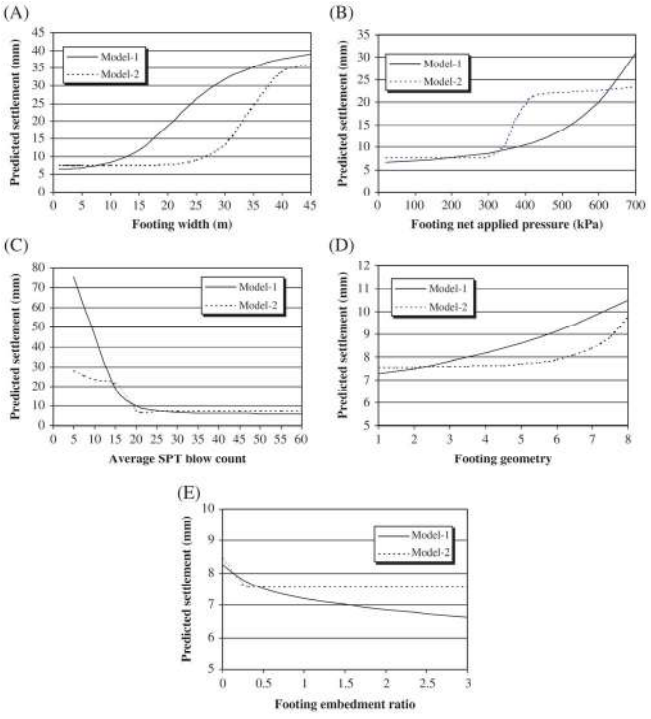


Figure 8.8 Results of the parametric study to test the robustness of the ANN models (Shahin et al., 2005b).

Shahin et al. (2005b) argued that since cross-validation (Stone, 1974) was adopted during the model development phase and an independent validation set was used to test the predictive ability of both models, the only plausible explanation for the different behaviors exhibited by both models was the connection weights included in each model. Shahin et al. (2005b) then advised that the connection weights should be examined as part of the interpretation of ANN model behavior, using, for example, the method suggested by Garson (1991). On the other hand, Kingston et al. (2005b) adopted the connection weight approach of Olden et al. (2004) for a case study in hydrological modeling in order to assess the relationship modeled by the ANNs. On the other hand, GP and EPR are claimed to provide better generalization ability than ANNs and therefore are worth further consideration in relation to achieving improved model robustness. However, it is also important to assess the relationship that has been modeled in the validation of AI models, rather than basing it on an error measure alone.

Model Transparency and Knowledge Extraction

Model transparency and knowledge extraction are the feasibility of interpreting AI models in a way that provides insights into how model inputs affect outputs. Figure 8.9 shows a representation of the classification of modeling techniques based on colors (Giustolisi et al., 2007) in which the higher the physical knowledge used during model development, the better the physical interpretation of the phenomenon that the model provides to the user. It can be seen that the color coding of mathematical modeling can be classified into white-, black-, and gray-box models, each of which can be explained as follows (Giustolisi et al., 2007). White-box models are systems that are based on first principles (e.g., physical laws) where model variables and parameters are known and have physical meaning by which the underlying physical relationships of the system can be explained. Black-box models are data-driven or regressive systems in which the functional form of relationships between model variables are unknown and need to be estimated. Black-box models rely on data to map the relationships between model inputs and corresponding outputs rather than to find a feasible structure of the model input-output relationships. Gray-box models are conceptual systems in which the mathematical structure of the model can be derived, allowing further information of the system behavior to be resolved.

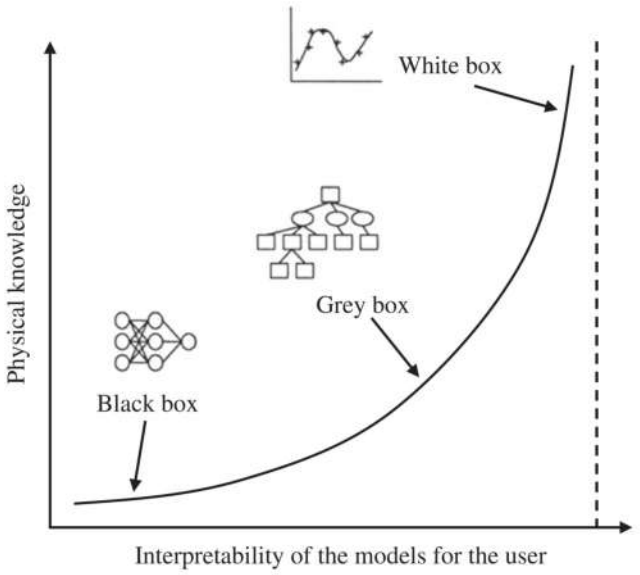


Figure 8.9 Graphical classification of modeling techniques. Source: Adapted from Giustolisi et al. (2007).

According to the above classification of modeling techniques based on color, whereby meaning is related to three levels of prior information required, ANNs belong to the class of black-box models due to their lack of transparency and the fact that they neither consider nor explain the underlying physical processes explicitly. This is because the knowledge extracted by ANNs is stored in a set of weights that are difficult to interpret properly; and due to the large complexity of the network structure, ANNs fail to give a transparent function that relates the inputs to the corresponding outputs. Consequently, it is difficult to understand the nature of the input-output relationships that are derived. This issue has been addressed by many researchers with respect to hydrological engineering. For example, Jain et al. (2004) examined whether the physical processes in a watershed were inherent in a trained ANN rainfall-runoff model. This was carried out by assessing the strengths of the relationships between the distributed components of the ANN model, in terms of the responses from the hidden nodes, and the deterministic components of the hydrological process, computed from a conceptual rainfall-runoff model, along with the observed input variables, using correlation coefficients and scatter plots. They concluded that the trained ANN, in fact, captured different components of the physical process and a careful examination of the distributed information contained in the trained ANN can be informative about the nature of the physical processes captured by various components of the ANN model. Sudheer (2005) performed perturbation analysis to assess the influence of each individual input variable on the output variable and found it to be an effective means of identifying the underlying physical process inherent in the trained ANN. Olden et al. (2004), Sudheer and Jain (2004), and Kingston et al. (2006) also addressed this issue of model transparency and knowledge extraction.

In the context of geotechnical engineering, Shahin et al. (2002) and Shahin and Jaksa (2005) expressed the results of the trained ANNs in the form of relatively straightforward equations. This was possible due to the relatively small number of input and output variables and hidden nodes. Neuro-fuzzy applications are another means of knowledge extraction that facilitate model transparency via extraction of rules. Neuro-fuzzy networks use the fuzzy logic system to store knowledge acquired from a set of input variables (x_1, x_2, ..., x_n) and the corresponding output variable (y) in a set of linguistic fuzzy rules that can be easily interpreted, such as IF (x_1 is high AND x_2 is low) THEN (y is high), c = 0.9, where (c = 0.9) is the rule confidence, which indicates the degree to which the above rule has contributed to the output. Examples of such applications in geotechnical engineering include Ni et al. (1996), Shahin et al. (2003), Gokceoglu et al. (2004), Provenzano et al. (2004), and Padmini et al. (2008). More recently, Cao and Qiao (2008) introduced the so-called neural network committee - based sensitivity analysis strategy to reveal the underlying relationships among the influential factors affecting a system through estimation of the relative contribution of each explicative (input) variable and dependent (output) variables. The strategy was applied to a case study of strata movement and provides employing a factor sensitivity analysis, instead of conventional single neural network analysis, to reveal the underlying mechanism of strata movement. This involves the following steps (Cao and Qiao, 2008): (i) the entire data set on strata movement is randomly split into two subsets, a training subset (4/5 of the samples) and a testing subset (1/5 of the samples); (ii) the model connection weights are adjusted using the training subset, and the model performance is tested using the testing subset; (iii) this process is repeatedly carried out many times so as to determine the best configuration of ANN, which captures the intrinsic mechanism of strata movement and transfers the observed data to implicit knowledge carried by the successfully trained neural network model.

Other researchers proposed the use of sensitivity analyses to explore the AI models by measuring the effects on the output of a given model when the inputs are varied through their

range of values. This approach allows a ranking of the inputs based on the amount of output changes produced due to disturbances in a given input, enabling the model to be explained. The quantification of this process is determined by holding all input variables at a fixed base line values (e.g., their average values), except one input attribute that is varied between its range ($x_a \in \{x_1, \ldots, x_n\}$), with ($j \in \{1, \ldots, L\}$) levels. The sensitivity response ($\hat{y}_{a,j}$) is determined for x_a to obtain the input relevance (R_a) using the sensitivity measure (S_a), as follows:

$$R_a = S_a / \sum_{i=1}^{n} S_i \times 100\ (\%) \tag{8.14}$$

For continuous regression tasks, the sensitivity measures (S_a) can take one of the following three measures, including the range (r_a), gradient (g_a), and variance (v_a), as follows:

$$S_a = r_a = \max(\hat{y}_a) - \min(\hat{y}_a) \tag{8.15}$$

$$S_a = g_a = \sum_{i=2}^{L} |\hat{y}_{a,j} - \hat{y}_{a,j-1}/(L-1)| \tag{8.16}$$

$$S_a = v_a = \sum_{i=2}^{L} (\hat{y}_{a,j} - \bar{\hat{y}}_a)^2/(L-1) \tag{8.17}$$

For more input influence details, Cortez and Embrechts (2011) proposed the global sensitivity analysis algorithm in combination with several visualization techniques, such as the variable effect characteristics (VEC) curve. For a given input variable, the VEC curve plots the *L* level values on the x-axis versus the sensitivity analysis responses on the y-axis, enabling increased interpretability of the AI models. Another sensitivity method introduced by Francone (2001) for the GP-based models and applied in geotechnical engineering by Alavi et al. (2010) allows the determination of the contribution of input variables to predict target outputs in the form of frequency values of input variables. The frequency value evaluates the importance of an input variable by determining how many times the variable appears in the contribution of the fitness of the GP-evolved programs (Alavi et al., 2010). A frequency value of 1.0 indicates that the input variable appears in 100% of the best GP-evolved programs, indicating that the predictive model is more sensitive to this input variable.

The GP and EPR, on the other hand, can be classified as gray-box techniques (conceptualization of physical phenomena); despite the fact that they are based on observed data, they return a mathematical structure that is symbolic and usually uncomplicated. The nature of obtained GP/EPR models permits the global exploration of expressions, which provides insights into the relationship between the model inputs and the corresponding outputs; i.e., it allows the user to gain additional knowledge of how the system performs. An additional advantage of GP/EPR over ANNs is that the structure and network parameters of ANNs (e.g., the number of hidden layers and their number of nodes, transfer functions, and the learning rate) should be identified a priori and are usually obtained using ad hoc, trial-and-error approaches. However, the number and combination of terms, as well as the values of GP/EPR modeling parameters, are all evolved automatically during model calibration. At the same time, the prior physical knowledge based on engineering judgment or other human knowledge can be used hypothesize about the elements of the objective functions and their structure, hence enabling refinement of final models. It should be noted that while white-box models provide maximum transparency, their construction may be difficult to obtain due to many geotechnical engineering problems where the underlying mechanism is not entirely understood.

Model Extrapolation

Model extrapolation is the model's ability to appropriately predict outside the range of the data used for model calibration. It is generally accepted that DDMs perform best when they do not extrapolate beyond the range of the data used for model calibration, which is considered to be an important limitation of AI models because it restricts their usefulness and applicability. Extreme value prediction is of particular concern in several areas of civil engineering, such as hydrological engineering, when floods are forecast, as well as in geotechnical engineering, when liquefaction potential and the stability of slopes are assessed. Sudheer et al. (2003) highlighted this issue and proposed a methodology, based on the Wilson-Hilferty transformation, for enabling ANN models to predict extreme values with respect to peak river flows. Their methodology yielded superior predictions compared to those obtained from an ANN model using untransformed data. More recently, Ismail and Jeng (2011) suggested the use of nonasymptotic PEs such as high-order neural networks (HONs) in modeling the load-settlement behavior of piles. AHON uses polynomial functions to map inputs into outputs and can be trained through error back- propagation algorithm. It uses high-order neurons instead of summation neurons (e.g., sigmoid) as Pes and the advantage of this is that the input and output para- meters do not have to be normalized within a certain range. This is because HON models are not asymptotic and do not have a limited dynamic range. To the author's knowledge, this type of neural networks has not been applied in geotechnical engineering and therefore is worth further consideration in relation to achieving improved model extrapolation.

Model Uncertainty

Finally, a further limitation of AI models is that the uncertainty in model predictions is seldom quantified. Failure to account for such uncertainty makes it impossible to assess the quality of AI model predictions, which may limit their efficacy. In addition, estimating uncertainty associated with predictions provided by DDMs is very important for decision making regardless of the generalization ability of the predictive model. This is because, from the point of view of a decision maker, the value of a prediction depends on the availability of additional information that helps to estimate the risk sassociated with decisions taken upon this prediction (Cherkassky et al. 2006).

In an effort to address the issue of model uncertainty, a few researchers have applied Bayesian techniques to ANN training (Buntine and Weigend, 1991; Kingston et al., 2005a, 2008; MacKay, 1992) in the context of hydrological engineering and Goh et al. (2005) did the same with respect to geotechnical engineering. In these studies, various Bayesian methods have been used to estimate the uncertainties in ANN parameters (weights) and Goh et al. (2005) observed that the integration of the Bayesian framework into the back-propagation algorithm enhanced neural network prediction capabilities and provided assessment of the confidence associated with network predictions. Research to date has demonstrated the value of Bayesian neural networks, although further work is needed in the area of geotechnical engineering. Shahin et al. (2005a) also incorporated uncertainty in the ANN process by developing a series of probabilistic design charts expressing the reliability of settlement predictions for shallow foundations on cohesionless soils. In the context of hydrological engineering, Shrestha and Solomatine (2006) introduced an approach to estimate model uncertainty using machine learning, and the method was tested in forecasting river flows. The idea is to build local models in which uncertainty is expressed in the form of the two quantiles (constituting the prediction interval) of the underlying distribution of prediction errors. Clustering and fuzzy logic are then used to model the propagation of integral uncertainty through the models.

8.4 Discussion and Conclusions

In the field of geotechnical engineering, it is possible to encounter some types of problems that are very complex and not well understood. In this regard, AI provides several advantages over more conventional computing techniques. For most traditional mathematical models, the lack of physical understanding is usually supplemented by either simplifying the problem or incorporating several assumptions into the models. Mathematical models also rely on assuming the structure of the model in advance, which may be less than optimal. Consequently, many mathematical models fail to simulate the complex behavior of most geotechnical engineering problems. In contrast, AI techniques are a data-driven approach in which the model can be trained on input-output data pairs to determine its structure and parameters. In this case, there is no need to either simplify the problem or incorporate any assumptions. Moreover, AI models can always be updated to obtain better results by presenting new training examples as new data become available. These factors combine to make AI techniques a powerful modeling tool in geotechnical engineering.

Despite the success of AI techniques in geotechnical engineering and other disciplines, they suffer from some shortcomings in relation to model transparency and knowledge extraction, ability of extrapolation, and model uncertainty, which need further attention in the future. For example, special attention should be paid to incorporating prior knowledge about the underlying physical process based on engineering judgment or human expertise into the learning formulation, checking of model robustness, and evaluation of model results. Furthermore, the standard RMSE error functions conventionally used in AI applications have to be updated and replaced with more representative error measures. Moreover, according to Flood (2008), ANNs in civil engineering (including geotechnical engineering) were used mostly as simple vector-mapping devices for function modeling of applications that require rarely more than a few tens of neurons without higher-order structuring. Together, improvements in these issues will greatly enhance the usefulness of ANN models and will provide the next generation of applied ANNs with the best way for advancing the field to the next level of sophistication and application.

The review of geotechnical engineering literature indicates that findings with regard to superiority of one AI technique over the other traditional methods are sometimes contradictory. Consequently, such findings should be treated as data specific and should not be generalized. The author suggests that for the time being, AI techniques might be treated as a complement to conventional computing techniques rather than as an alternative, or may be used as a quick check on solutions developed by more time-consuming and in-depth analyses.

References

Abdel-Rahman, A.H., 2008. Predicting compaction of cohesionless soils using ANN. Ground Improv. 161, 3-8.

Adarsh, S.A., Jangareddy, M., 2010. Slope stability modeling using genetic programming. Int. J. Earth Sci. Eng. 3, 1-8.

Adeli, H., 2001. Neural networks in civil engineering: 1989-2000. Comput. Aided Civ. Infrastruct. E. 16, 126-142.

Ahangar-Asr, A., Faramarzi, A., Javadi, A.A., 2010. A new approach for prediction of the stability of soil and roc slopes. Eng. Comput. Int. J. Comput. Aided Eng. Softw. 27, 878-893.

Ahangar-Asr, A., Faramarzi, A., Mottaghifard, N., Javadi, A.A., 2011. Modeling of perme ability and compaction characteristics of soils using evolutionary polynomial regression. Comput. Geosci. 37, 1860-1869.

Ahmad, I., El Naggar, H., Kahn, A.N., 2007. Artificial neura network application to estimate kinematic soil pile interaction response parameters. Soil Dynam. Earthquake Eng. 27, 892-905.

Alavi, A.H., Gandomi, A.H., 2011a. Prediction of principal ground motion parameters using a hybrid method coupling artificial neural networks and simulated annealing. Comput. Struct. 89, 2176-2194.

Alavi, A.H., Gandomi, A.H., 2011b. A robust data mining approach for formulation of geotechnical engineering systems. Eng. Comput. Int. J. Comput. Aided Eng. Softw. 28, 242-274.

Alavi, A.H., Gandomi, A.H., 2011c. A robust data mining approach for formulation of geotechnical engineering systems. Eng. Comput. 28, 242-274.

Alavi, A.H., Gandomi, A.H., 2012. Energy-based numerical models for assessment of soil liquefaction. Geosci. Frontiers. 3, 541-555.

Alavi, A.H., Gandomi, A.H., Sahab, M.G., Gandomi, M., 2010. Multi expression programming: a new approach to formulation of soil classification. Eng. Comput. 26, 111-118.

Alimoradi, A., Moradzadeh, A., Naderi, R., Salehi, M.Z., Etemadi, A., 2008. Prediction of geological hazardous zones in front of a tunnel face using TSP-203 and artificial neural networks. Tunn. Undergr. Space Technol. 23, 711-717.

Alkroosh, I., Nikraz, H., 2011a. Correlation of pile axial capacity and CPT data using gene expression programming. Geotech. Geol. Eng. 29, 725-748.

Alkroosh, I., Nikraz, H., 2011b. Simulating pile load-settlement behavior from CPT data using intelligent computing. Cent. Eur. J. Eng. 1, 295-305.

Alkroosh, I., Nikraz, H., 2012. Predicting axial capacity of driven piles in cohesive soils using intelligent computing. Eng. Appl. Artif. Intell. 25, 618-627.

Ardalan, H., Eslami, A., Nariman-Zadeh, N., 2009. Piles shaft capacity from CPT and CPTU data by polynomial neural networks and genetic algorithms. Comput. Geotech. 36, 616-625.

ASCE, 2000. Artificial neural networks in hydrology. I: Preliminary concepts. J. Hydrol. Eng. 5, 115-123.

Ashayeri, I., Yasrebi, S., 2009. Free-swell and swelling pressure of unsaturated compacted clays; experiments and neural networks modelling. Geotech. Geol. Eng. 27, 137-153.

Banimahd, M., Yasrobi, S.S., Woodward, P.K., 2005. Artificial neural network for stress-strain behavior of sandy soils: knowledge based verification. Comput. Geotech. 32, 377-386.

Baykasoglu, A., Gullu, H., Canakci, H., Ozbakir, L., 2008. Prediction of compressive and tensile strength of limestone via genetic programming. Expert Syst. Appl. 35, 111-123.

Baziar, M.H., Ghorbani, A., 2005. Evaluation of lateral spreading using artificial neural networks. Soil Dynam. Earthquake Eng. 25, 1-9.

Baziar, M.H., Jafarian, Y., Shahnazari, H., Movahed, V., Tutunchian, M.A., 2011. Prediction of strain energy-based liquefaction resistance of sand-silt mixtures: an evolutionary approach. Comput. Geotech. 37, 1883-1893.

Bhattacharya, B., Solomatine, D., 2006. Machine learning in soil classification. Neural Networks. 19, 186-195.

Boubou, R., Emeriault, F., Kastner, R., 2010. Artificial neural network application for the prediction of ground surface movements induced by shield tunnelling. Can. Geotech. J. 47, 1214-1233.

Bowden, G.J., Maier, H.R., Dandy, G.C., 2002. Optimal division of data for neural network models in water resources applications. Water Resour. Res. 38, 2.1-2.11.

Bowden, G.J., Dandy, G.C., Maier, H.R., 2003. Data transformation for neural network models in water resources applications. J. Hydroinform. 5, 245-258.

Bowden, G.J., Dandy, G.C., Maier, H.R., 2005. Input determination for neural network models in water resources applications: Part1—Background and methodology. J. Hydrol. 301, 75-92.

Braddock, R.D., Kremmer, M.L., Sanzogni, L., 1998. Feed-forward artificial neural network model for forecasting rainfall run-off. Environmetrics. 9, 419-432.

Buntine, W.L., Weigend, A.S., 1991. Bayesian back-propagation. Complex Systems. 5, 603-643.

Byeon, W.Y., Lee, S.R., Kim, Y.S., 2006. Application of flat DMT and ANN to Korean soft clay deposits for reliable estimation of undrained shear strength. Int. J. Offshore Polar Eng. 16, 73-80.

Cabalar, A.F., Cevik, A., Guzelbey, I.H., 2009. Constitutive modeling of Leighton Buzzard sands using genetic programming. Neural Comput. Appl. 19, 657-665.

Caglar, N., Arman, H., 2007. The applicability of neural networks in the determination of soil profiles. Bull. Eng. Geol. Environ. 66, 295-301.

Cai, S., Toral, H., Qiu, J., Archer, J.S., 1994. Neural network based objective flow regime identification in air-water two-phase flow. Can. J. Chem. Eng. 72, 440-445.

Campolo, M., Soldati, A., Andreuss, I.P., 1999. Forecasting-river flow rate during low-flow periods using neural networks. Water Resour. Res. 35, 3547-3552.

Cao, M., Qiao, P., 2008. Neural network committe-based sensitivity analysis strategy for geotechnical engineering problems. Neural Comput. Appl. 17, 509-519.

Celik, S., Tan, O., 2005. Determination of preconsolidation pressure with artificial neural network. Civ. Eng. Environ. Syst. 22, 217-231.

Cevik, A., Sezer, E.A., Cabalar, A.F., Gokceoglu, C., 2010. Modeling of the uniaxial compressive strength of some clay-bearing rocks using neural network. Appl. Soft. Comput. 11, 2587-2594.

Chen, Y., Azzam, R., Fernandez, T.M., Li, L., 2009. Studies on construction pre-control of a connection aisle between two neighbouring tunnels in Shanghai by means of 3D FEM, neural networks and fuzzy logic. Geotech. Geol. Eng. 27, 155-167.

Chen, Y., Azzam, R., Zhang, F., 2006. The displacement computation and construction pre-control of a foundation pit in Shanghai utilizing FEM and intelligent methods. Geotech. Geol. Eng. 24, 1781-1801.

Chen, Y., Hsieh, S., Liu, C., 2010. Simulation of stress-strain behavior of saturated sand in undrained triaxial tests based on genetic adaptive neural networks. Electron. J. Geotech. Eng. 15, 1814-1834.

Cherkassky, V., Krasnopolsky, V., Solomatine, D.P., Valdes, J., 2006. Computational intelligence in earth sciences and environmental applications: issues and challenges. Neural Networks. 19, 113-121.

Cho, S.E., 2009. Probabilistic stability analyses of slopes using the ANN-based response surface. Comput. Geotech. 36, 787-797.

Cortez, P., Embrechts, M.J., 2011. Opening black box data mining models using sensitivity analysis. In: Proceedings of the 2011 IEEE Symposium on Computational Intelligence and Data Mining, Paris, pp.341-348.

Cramer, N.L., 1985. A representation for the adaptive generation of simple sequential programs. In: Proceedings of the International Conference on Genetic Algorithms and Their Applications, Carnegie-Mellon University, Pittsburgh, PA, pp. 183-187.

Das, S.K., Basudhar, P.K., 2005. Prediction of coefficient of lateral earth pressure using artificial neural networks. Electron. J. Geotech. Eng. 10.

Das, S.K., Basudhar, P.K., 2006. Undrained lateral load capacity of piles in clay using artificial neural network. Comput. Geotech. 33, 454-459.

Das, S.K., Sivakugan, N., 2010. Discussion of: intelligent computing for modeling axial capacity of pile foundations. Can. Geotech. J. 47, 928-930.

Das, S.K., Biswal, R.K., Sivakugan, N., Das, B., 2011a. Classification of slopes and prediction of factor of safety using differential neural networks. Environ. Earth Sci. 64, 201-210.

Das, S.K., Samui, P., Sabat, A.K., 2011b. Application of artificial intelligence to maximum dry density and unconfined compressive strength of cement stabilized soil. Geotech. Geol. Eng. 29, 329-342.

Dincer, I., 2011. Models to predict the deformation modulus and the coefficient of subgrade reaction for earth filling structure. Adv. Eng. Softw. 42, 160-171.

Doostmohamadi, R., Moosavi, M., Araabi, B.N., 2008. Modelling time dependent swelling potential of mudrock using laboratory tests and artificial neural network. Min. Tech. 117, 32-41.

Elshorbagy, A., Corzo, G., Srinivasulu, S., Solomatine, D.P., 2010. Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology— part 1: concepts and methodology. Hydrol. Earth Syst. Sci. 14, 1931-1941.

Erzin, Y., 2007. Artificial neural networks approach for swell pressure versus soil suction behaviour. Can. Geotech. J. 44, 1215-1223.

Erzin, Y., Gumaste, S.D., Gupta, A.K., Singh, D.N., 2009. Artificial neural network (ANN) models for determining hydraulic conductivity of compacted fine-grained soils. Can. Geotech. J. 46, 955-968.

Faraway, J., Chatfield, C., 1998. Time series forecasting with neural networks: a comparative study using the airline data. Appl. Stat. 47, 231-250.

Fausett, L.V., 1994. Fundamentals Neural Networks: Architecture, Algorithms, and Applications. Prentice-Hall, Englewood Cliffs, NJ.

Feng, X.T., Chen, B., Yang, C., Zhou, H., Ding, X., 2006. Identification of visco-elastic models for rocks using genetic programming coupled with the modified particles warm optimization algorithm. Int. J. Rock Mech. Min. Sci. 43, 789-801.

Ferentinou, M.D., Sakellariou, M.G., 2007. Computational intelligence tools for the prediction of slope performance. Comput. Geotech. 34, 362-384.

Ferreira, C., 2001. Gene expression programming: a new adaptive algorithm for solving problems. Complex Systems. 13, 87-129.

Firat, M., Gungor, M., 2008. Generalized regression neural networks and feed forward neural networks for prediction of scur depth around bridge piers. Adv. Eng. Softw. 40, 731-737.

Flood, I., 2008. Towards the next generation of artificial neural networks for civil engineering. Adv. Eng. Inform. 22, 4-14.

Francone, F.D., 2001. Discipulus Software Owner's Manual. Register Machine Learning Technologies Inc., Littleton, CO.

Fu, Q., Hashash, Y.M.A., Hung, S., Ghaboussi, J., 2007. Integration of laboratory testing and constitutive modelling of soils. Comput. Geotech. 34, 330-345.

Gandomi, A.H., Alavi, A.H., 2011. Multi-stage genetic programming: a new strategy to non-linear system modeling. Inf. Sci. 181, 5227-5239.

Gandomi, A.H., Alavi, A.H., 2012. A new multi-gene genetic programming approach to non-linear system modeling, PartII: Geotechnical and earthquake engineering problems. Neural Comput. Appl. 21, 189-201.

Gandomi, A.H., Alavi, A.H., Gun, J.Y., 2011. Formulation of uplift capacity of suction cassions using multi expression programming. KSCE J. Civ. Eng. 15, 363-373.

Garaga, A., Latha, G.M., 2010. Intelligent prediction of the stress-strain response of intact and jointed rocks. Comput. Geotech. 37, 629-637.

Garcia, S.R., Roma, M.P., 2009. Rockfill strength evaluation using cascade correlation networks. Geotech. Geol. Eng. 27, 289-304.

Garcia, S.R., Romo, M.P., Figueroa-Nazuno, J., 2006. Soil dynamic properties determination: a neurofuzzy system approach. Contr. Intell. Syst. 34, 1-11.

Gardner, M.W., Dorling, S.R., 1998. Artificial neural networks (the multi layer perceptron)— a review of applications in the atmospheric sciences. Atmos. Environ. 32, 2627-2636.

Garson, G.D., 1991. Interpreting neural-network connection weights. AI Expert. 6, 47-51.

Giraudel, J.L., Lek, S., 2001. A comparison of self-organizing map algorithm and some conventional statistical methods for ecological community ordination. Ecol. Modell. 146, 329-339.

Giustolisi, O., Savic, D.A., 2006. A symbolic data-driven technique based on evolutionary polynomial regression. J. Hydroinform. 8, 207-222.

Giustolisi, O., Doglioni, A., Savic, D.A., Webb, B.W., 2007. A multi-model approach to analysis of environmental phenomena. Environ. Model. Softw. 22, 674-682.

Goh, A.T., Kulhawy, F.H., Chua, C.G., 2005. Bayesian neural network analysis of undrained side resistance of drilled shafts. J. Geotech. Geoenviron. Eng. 131, 84-93.

Goh, A.T.C., 1994. Seismic liquefaction potential assessed by neural network. J. Geotech. Geoenviron. Eng. 120, 1467-1480.

Goh, A.T.C., Kulhawy, F.H., 2005. Reliability assessment of serviceability performance of braced retaining walls using a neural network approach. Int. J. Numer. Anal. Methods Geomech. 29, 627-642.

Gokceoglu, C., Yesilnacar, E., Sonmez, H., Kayabasi, A., 2004. A neuro-fuzzy model for modulus of deformation of jointed rock masses. Comput. Geotech. 31, 375-383.

Goldberg, D.E., 1989. Genetic Algorithms in Search Optimization and Machine Learning. Addison-Wesley, Reading, MA.

Gunaydin, O., Gokoglu, A., Fener, M., 2010. Prediction of artificial soil's unconfined compression strength test using statistical analyses and artificial neural networks. Adv. Eng. Softw. 41, 1115-1123.

Guven, A., Aytek, A., 2009. New approach for stage-discharge relationship: gene-expression programming. J. Hydrol. Eng. 14, 812-820.

Hajihassani, M., Marto, A., Nazmi, E., Abad, S., Shahrbabaki, M., 2011. Prediction of surface settlements induced by NATM tunneling based on artificial neural networks. Electro. J. Geotech. Eng. 16, 1471-1480.

Hammerstrom, D., 1993. Working with neural networks. IEEE Spectr. 30, 46-53.

Hanna, A.M., Ural, D., Saygili, G., 2007a. Evaluation of liquefaction potential of soil deposits using artificial neural networks. Int. J. Comput. Aided Eng. Softw. 24, 5-16.

Hanna, A.M., Ural, D., Saygili, G., 2007b. Neural network model for liquefaction potential in soil deposits using Turkey and Taiwan earthquake data. Soil Dynam. Earthquake Eng. 27, 521-540.

Hecht-Nielsen, R., 1990. Neurocomputing. Addison-Wesley, Reading, MA.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. Ann Arbor, Mich., University of Michigan Press.

Ikizler, S.B., Aytekin, M., Vekli, M., Kocabas, F., 2009. Prediction of swelling pressures of expansive soils using artificial neural networks. Adv. Eng. Softw. 41, 647-655.

Ismail, A.M., Jeng, D.S., 2011. Modelling load-settlement behavior of piles using high-order neural network (HON-PIL model). Eng. Appl. Artif. Intell. 24, 813-821.

Jain, A., Sudheer, K.P., Srinivasulu, S., 2004. Identification of physical processes inherent in artificial neural network rainfall runoff models. Hydrolog. Process. 18, 571-581.

Javadi, A.A., Rezania, M., 2009. Intelligent finite element method: an evolutionary approach to constitutive modelling. Adv. Eng. Inform. 23, 442-451.

Javadi, A., Rezania, M., Mousavi, N.M., 2006. Evaluation of liquefaction induced lateral displacements using genetic programming. Comput. Geotech. 33, 222-233.

Jianping, J., Changhong, Y., Guangyun, G., 2011. Prediction of compressibility coefficient of soil based on RBF neural network. J. Jiangsu Univ. Natural Sci. Ed. 32, 232-235.

Johari, A., Habibagahi, G., Ghahramani, A., 2006. Prediction of soil-water characteristic curve using genetic programming. J. Geotech. Geoenviron. Eng. 132, 661-665.

Johari, A., Javadi, A., Habibagahi, G., 2011. Modelling the mechanical behavior of unsaturated soils using a genetic algorithm-based neural network. Comput. Geotech. 38, 2-13.

Kamatchi, P., Rajasankar, J., Ramana, G.V., Nagpal, A.K., 2010. A neural network based methodology to predict site-specific spectral acceleration values. Earthquake Eng. Vib. 9, 459-472.

Kanungo, D.P., Arora, M.K., Sarkar, S., Gupta, R.P., 2006. A comparative study of conventional ANN black box, fuzzy and combined neural and fuzzy weighting procedures for landslide susceptibility zonation in Darjeeling Himalayas. Eng. Geol. 85, 347-366.

Kavli, T., 1993. ASMOD—An algorithm for adaptive spline modelling of observation data. Int. J. Control. 58, 947-967.

Kaya, A., 2009. Residual and fully softened strength evaluation of soils using artificial neural networks. Geotech. Geol. Eng. 27, 281-288.

Kayadelen, C., 2011. Soil liquefaction modeling by genetic expression programming and neuro-fuzzy. Expert Syst. Appl. 38, 4080-4087.

Kayadelen, C., Gunaydin, O., Fener, M., Demir, A., Ozvan, A., 2009. Modeling of the angle of shearing resistance of soils using soft computing systems. Expert Syst. Appl. 36, 11814-11826.

Khozaghi, S.S.H., Choobbasti, A.J.A., 2007. Predicting of liquefaction potential in soils using artificial neural networks. Electron. J. Geotech. Eng. 12.

Kim, Y., Kim, B., 2008. Prediction of relative crest settlement of concrete-faced rockfill dams analyzed using an artificial neural network model. Comput. Geotech. 35, 313-322.

Kingston, G.B., Lambert, M.F., Maier, H.R., 2005a. Bayesian parameter estimation applied to artificial neural networks used for hydrological modelling. Water Resour. Res. 41. doi:10.1029/2005WR004152.

Kingston, G.B., Maier, H.R., Lambert, M.F., 2005b. Calibration and validation of neural networks to ensure physically plausible hydrological modelling. J. Hydrol. 314, 158-176.

Kingston, G.B., Maier, H.R., Lambert, M.F., 2006. A probabilistic method to assist knowledge extraction from artificial neural networks used for hydrological prediction. Math. Comput. Model. 44, 499-512.

Kingston, G.B., Maier, H.R., Lambert, M.F., 2008. Bayesian model selection applied to artificial neural networks used for water resources modelling. Water Resour. Res. 44. doi: 10.1029/2007WR006155.

Kocjancic, R., Zupan, J., 2000. Modelling of the river flow rate: the influence of the training set selection. Chemomet. Intell. Lab. Syst. 54, 21-34.

Kogut, J., 2007. Dynamic soil profile determination with the use of a neural network. Comput. Assist. Mech. Eng. Sci. 14, 209-217.

Koza, J.R., 1992. Genetic Programming: on the Programming of Computers by Natural Selection. MIT Press, Cambridge, MA.

Kung, G.T., Hsiao, E.C., Schuster, M., Juang, C., 2007. A neural network approach to estimating deflection of diaphragm walls caused by excavation in clays. Comput. Geotech. 34, 385-396.

Kuo, Y.L., Jaksa, M.B., Lyamin, A.V., Kaggwa, W.S., 2009. ANN-based model for predicting the bearing capacity of strip footing on multi-layered cohesive soil. Comput. Geotech. 36, 503-516.

Kurup, P.U., Griffin, E., 2006. Prediction of soil composition from CPT data using general regression neural network. J. Comput. Civil. Eng. 20, 281-289.

Lee, T.L., Lin, H.M., Jeng, D.S., Lu, Y.P., 2008. Back-propagation neural network for assessment of highway slope failure in Taiwan. Geotech. Eng. 39, 121-128.

Legates, D.R., Mccabe, G.J., 1999. Evaluating the use of goodness-of-fit measures in hydrologic and hydroclimatic model validation. Water Resour. Res. 35, 233-241.

Liao, K., Fan, J., Huang, C., 2011. An artificial neural network for groutability prediction of permeation grouting with microfine cement grouts. Comput. Geotech. 38, 978-986.

Lu, Y., 2005. Underground blast induced ground shock and its modelling using artificial neural network. Comput. Geotech. 32, 164-178.

Ma, S., Cao, L., Li, H., 2006. The improved neural network and its application for valuing rock mass mechanical parameter. J. Coal Sci. Eng. 12, 21-24.

Mackay, G., 1992. A practical Bayesian framework for backpropagation networks. Neural. Comput. 4, 448-472.

Maier, H.R., Dandy, G.C., 2000a. Applications of artificial neural networks to forecasting of surface water quality variables: issues, applications and challenges. In: Govindaraju, R.S., Rao, A.R. (Eds.), Artificial Neural Networks in Hydrology. Kluwer, Dordrecht, The Netherlands.

Maier, H.R., Dandy, G.C., 2000b. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environ. Modell. Softw. 15, 101-124.

Maji, V.B., Sitharam, T.G., 2008. Prediction of elastic modulus of jointed rockmass using artificial neural networks. Geotech. Geol. Eng. 26, 443-452.

Masters, T., 1993. Practical Neural Network Recipes in C++. Academic Press, San Diego, CA.

May, R.J., Maier, H.R., Dandy, G.C., Fernando, T.M., 2008. Non-linear variable selection for artificial neural networks using partial mutual information. Environ. Model. Softw. 23, 1312-1326.

Mcculloch, W.S., Pitts, W., 1943. A logical calculus of ideas imminent in nervous activity. Bull. Math. Biophys. 5, 115-133.

Mollahasani, A., Alavi, A.H., Gandomi, A.H., 2011. Empirical modeling of plate load test moduli of soil via gene expression programming. Comput. Geotech. 38, 281-286.

Najjar, Y.M., Huang, C., 2007. Simulating the stress-strain behavior of Georgia Kaolin via recurrent neuro net approach. Comput. Geotech. 34, 346-362.

Najjar, Y.M., Basheer, I.A., Naouss, W.A., 1996. On the identification of compaction characteristics by neuronets. Comput. Geotech. 18, 167-187.

Narendara, B.S., Sivapullaiah, P.V., Suresh, S., Omkar, S.N., 2006. Prediction of unconfined compressive strength of soft grounds using computational intelligence techniques: a comparative study. Comput. Geotech. 33, 196-208.

Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models. J. Hydrol. 10, 282-290.

Neaupane, K.M., Adhikari, N.R., 2006. Prediction of tunnel-

ling-induced ground movement with the multi-layer perceptron. Tunn. Undergr. Space Technol. 21, 151-159.

NeuralWare, 1997. NeuralWorks Predict Release 2.1. NeuralWare Inc., Pittsburgh.

Ni, S.H., Lu, P.C., Juang, C.H., 1996. A fuzzy neural network approach to evaluation of slope failure potential. J. Microcomput. Civ. Eng. 11, 59-66.

Obrzud, R.F., Vulliet, L., Truty, A., 2009. A combined neural network/gradient-based approach for the identification of constitutive model parameters using self-boring pressuremeter tests. Int. J. Numer. Anal. Methods Geomech. 33, 817-849.

Olden, J.D., Joy, M.K., Death, R.G., 2004. An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. Ecol. Model. 178, 389-397.

Padmini, D., Ilamparuthi, K., Sudheer, K.P., 2008. Ultimate bearing capacity prediction of shallow foundations on cohesionless soils using neurofuzzy models. Comput. Geotech. 35, 33-46.

Pal, M., Deswal, S., 2008. Modeling pile capacity using support vector machines and generalized regression neural network. J. Geotech. Geoenviron. Eng. 134, 1021-1024.

Parasuraman, K., Elshorbagy, A., Bing, C.S., 2007. Estimating saturated conductivity using genetic programming. Soil Sci. Soc. Am. J. 71, 1676-1684.

Park, H., Lee, S.R., 2011. Evaluation of the compression index of soils using an artificial neural network. Comput. Geotech. 38, 472-481.

Park, H.I., 2011. Development of neural network model to estimate the permeability coefficient of soils. Mar. Georesour. Geotechnol. 29, 267-278.

Park, H.I., Kim, Y.T., 2011. Prediction of strength of reinforced lightweight soil using an artificial neural network. Eng. Comput. 28, 600-615.

Peng, X., Wang, Z., Luo, T., Yu, M., Luo, Y., 2008. An elastoplastic constitutive model of moderate sandy clay based on BC-RBFNN. J. Cent. South Univ. Technol. 15, 47-50.

Pooya Nejad, F., Jaksa, M.B., Kakhi, M., Mccabe, B.A., 2009. Prediction of pile settlement using artificial neural networks based on standard penetration test data. Comput. Geotech.

Provenzano, P., Ferlisi, S., Musso, A., 2004. Interpretation of a model footing response through an adaptive neural fuzzy inference system. Comput. Geotech. 31, 251-266.

Ray, C., Klindworth, K.K., 2000. Neural networks for agrichemical vulnerability assessment of rural private wells. J. Hydrol. Eng. 5, 162-171.

Rezania, M., Faramarzi, A., Javadi, A., 2011. An evolutionary based approach for assessment of earthquake-induced soil liquefaction and lateral displacement. Eng. Appl. Artif. Intell. 24, 142-153.

Rezania, M., Javadi, A., 2007. A new genetic programming model for predicting settlement of shallow foundations. Can. Geotech. J. 44, 1462-1472.

Rezania, M., Javadi, A., Giustolisi, O., 2010. Evaluation of liquefaction potential based on CPT results using evolutionary polynomial regression. Comput. Geotech. 37, 82-92.

Rezania, M., Javadi, A.A., 2008. An evolutionary-based data mining technique for assessment of civil engineering systems. Eng. Comput. Int. J. Comput. Aided Eng. Softw. 25, 500-517.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representation by error propagation. In: Rumelhart, D.E., McClelland, J.L. (Eds.), Parallel Distributed Processing. MIT Press, Cambridge.

Sakellariou, M.G., Ferentinou, M.D., 2005. A study of slope stability prediction using neural networks. Geotech. Geol. Eng. 23, 419-445.

Samui, P., Kumar, B., 2006. Artificial neural network prediction of stability numbers for two-layered slopes with associated flowrule. Electron. J. Geotech. Eng. 11, www.ejge.com.

Samui, P., Sitharam, T.G., 2011. Determination of liquefaction susceptibility of soil based on field test and artificial intelligence. Int. J. Earth Sci. Eng. 4, 216-222.

Santos, J., Ovidio, J., Celestino, T.B., 2008. Artificial neural networks analysis of Sao Paulo subway tunnel settlement data. Tunnell. Undegr. Space Technol. 23, 481-491.

Sarkar, K., Tiwary, A., Singh, T.N., 2010. Estimation of strength parameters of rock using artificial neural networks. Bull. Eng. Geol. Environ. 69, 599-606.

Savic, D.A., Giutolisi, O., Berardi, L., Shepherd, W., Djordjevic, S., Saul, A., 2006. Modelling sewer failure by evolutionary computing. Proc. Inst. Eng. Water Manage. 159, 111-118.

Shafiee, A., Ghate, R., 2008. Shear modulus and damping ratio in aggregate-clay mixtures: an experimental study versus ANNs prediction. J. Appl. Sci. 8, 3068-3082.

Shahin, M.A., 2010. Intelligent computing for modelling axial capacity of pile foundations. Can. Geotech. J. 47, 230-243.

Shahin, M.A., Indraratna, B., 2006. Modelling the mechanical behavior of railway ballast using artificial neural networks. Can. Geotech. J. 43, 1144-1152.

Shahin, M.A., Jaksa, M.B., 2005. Neural network prediction of pullout capacity of marquee ground anchors. Comput. Geotech. 32, 153-163.

Shahin, M.A., Jaksa, M.B., 2006. Pullout capacity of small ground anchors by direct cone penetration test methods and neural networks. Can. Geotech. J. 43, 626-637.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2001. Artificial neural network applications in geotechnical engineering. Aust. Geomech. 36, 49-62.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2002. Artificial neural network-based settlement prediction formula for shallow foundations on granular soils. Aust. Geomech. 37, 45-52.

Shahin, M.A., Maier, H.R., Jaksa, M.B., 2003. Settlement prediction of shallow foundations on granular soils using B-spline neurofuzzy models. Comput. Geotech. 30, 637-647.

Shahin, M.A., Maier, H.R., Jaksa, M.B., 2004. Data division for developing neural networks applied to geotechnical engineering. J. Comput. Civ. Eng. 18, 105-114.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2005a. Neural network based stochastic design charts for settlement prediction. Can. Geotech. J. 42, 110-120.

Shahin, M.A., Maier, H.R., Jaksa, M.B., 2005b. Investigation into the robustness of artificial neural network models for a

case study in civil engineering. In: Proceedings of the International Congress on Modelling and Simulation, MODSIM 2005, Melbourne, Australia, pp.79-83.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2009. Recent advances and future challenges for artificial neural systems in geotechnical engineering applications. J. Adv. Artif. Neural Syst. doi:10.1155/2009/308239.

Shahnazari, H., Dehnavi, Y., Alavi, A.H., 2010. Numerical modeling of stress-strain behavior of sand under cyclic loading. Eng. Geol. 116, 53-72.

Shrestha, D.L., Solomatine, D.P., 2006. Machine learning approaches for estimation of prediction interval for the model output. Neural Networks. 19.

Shuh-Gi, C., Ching-Yinn, L., 2009. CPT-based simplified liquefaction assessment by using fuzzy-neural network. J. Mar. Sci. Technol. 17, 326-331.

Singh, T.N., Singh, V., 2005. An intelligent approach to prediction and control ground vibration in mines. Geotech. Geol. Eng. (23), 249-262.

Singh, T.N., Verma, A.K., Singh, V., Sahu, A., 2005. Slake durability of shaly rock and its predictions. Environ. Geol. 47, 246-253.

Singh, T.N., Verma, A.K., Sharma, P.K., 2007. A neuro-genetic approach for prediction of time dependent deformational characteristic of rock and its sensitivity analysis. Geotech. Geol. Eng. 25, 395-407.

Sinha, S.K., Wang, M.C., 2008. Artificial neural network prediction models for soil compaction and permeability. Geotech. Eng. J. 26, 47-64.

Sitharam, T.G., Samui, P., Anbazhagan, P., 2008. Spatial variability of rock depth in Bangalore using geostatistical, neural network and support vector machine models. Geotech. Geol. Eng. 26, 503-517.

Sivrikaya, O., Soycan, T.Y., 2011. Estimation of compaction parameters of fine-grained soils in terms of compaction energy using artificial neural networks. Int. J. Numer. Anal. Methods Geomech. 35, 1830-1841.

Smith, G.N., 1986. Probability and Statistics in Civil Engineering: An Introduction. Collins, London.

Solomatine, D.P., Ostfeld, A., 2008. Data-driven modelling: some past experience and new approaches. J. Hydroinform. 10, 3-22.

Soroush, A., Foroozan, R., Asadollahi, P., 2006. Simulation of 3D effect of excavation face advancement using an ANN trained by numerical models. Electron. J. Geotech. Eng. 11, www.ejge.com.

Stone, M., 1974. Cross-validatory choice and assessment of statistical predictions. J. Roy. Stat. Soc. B. 36, 111-147.

Sudheer, K.P., 2005. Knowledge extraction from trained neural network river flow models. J. Hydrol. Eng. 10, 264-269.

Sudheer, K.P., Jain, A., 2004. Explaining the internal behavior of artificial neural network river flow models. Hydrol. Processes. 18, 833-844.

Sudheer, K.P., Nayak, P.C., Ramasastri, K.S., 2003. Improving peak flow estimates in artificial neural network river flow models. Hydrol. Processes. 17, 677-686.

Sulewska, M.J., 2010. Neural modelling of compactability characteristics of cohessionless soil. Comput. Assist. Mech. Eng. Sci. 17, 27-40.

Tawadrous, A.S., Degagne, D., Pierce, M., Masivars, D., 2009. Prediction of uniaxial compression PFC3D model micro-properties using artificial neural networks. Int. J. Numer. Anal. Methods Geomech. 33, 1953-1962.

Tekin, E., Akbas, S.O., 2011. Artificial neural networks approach for estimating the groutability of granular soils with cement-based grouts. Bull. Eng. Geol. Environ. 70, 153-161.

Teodorescu, L., Sherwood, D., 2008. High energy physics event selection with gene expression programming. Comupt. Phys. Commun. 178, 409-419.

Tokar, S.A., Johnson, P.A., 1999. Rainfall-runoff modeling using artificial neural networks. J. Hydrol. Eng. 4, 232-239.

Tsekouras, G.J., Koukoulis, J., Mastorakis, N.E., 2010. An optimized neural network for predicting settlements during tunneling excavation. WSEAS Trans. Syst. 9, 1153-1167.

Tsompanakis, Y., Lagaros, N.D., Psarropoulos, P.N., Georgopoulos, E.C., 2009. Simulating the seismic response of embankments via artificial neural networks. Adv. Eng. Softw. 40, 640-651.

Twomey, J.M., Smith, A.E., 1997. Validation and verification. In: Kartam, N., Flood, I., Garrett, J.H. (Eds.), Artificial Neural Networks for Civil Engineers: Fundamentals and Applications. ASCE, New York.

Uncuoglu, E., Laman, M., Saglamer, A., Kara, H.B., 2008. Prediction of lateral effective stresses in sand using artificial neural network. Soils Found. 48, 141-153.

Ural, D.N., Saka, H., 1998. Liquefaction assessment by neural networks. Electron. J. Geotech. Eng. 3, www.ejge.com.

Wang, H.B., Sassa, K., 2006. Rainfall-induced landslide hazard assessment using artificial neural networks. Earth Surf. Processes Landforms. 31, 235-247.

Yang, Y., Rosenbaum, M.S., 2002. The artificial neural network as a tool for assessing geotechnical properties. Geotech. Eng. J. 20, 149-168.

Yildiz, E., Ozyazicioglu, M.H., Ozkan, M.Y., 2010. Lateral pressures on rigid retaining walls: a neural network approach. Gazi Univ. J. Sci. 23, 201-210.

Yoo, C., Kim, J., 2007. Tunneling performance prediction using an integrated GIS and neural network. Comput. Geotech. 34, 19-30.

Young-Su, K., Byung-Tak, K., 2006. Use of artificial neural networks in the prediction of liquefaction resistance of sands. J. Geotech. Geoenviron. Eng. 132, 1502-1504.

Yu, Y., Zhang, B., Yuan, H., 2007. An intelligent displacement back-analysis method for earth-rockfill dams. Comput. Geotech. 34, 423-434.

Zounemat-Kermani, M., Beheshti, A.A., Ataie-Ashtiani, B., Asabbagh-Yazdi, S.R., 2009. Estimation of current-induced scour depth around pile groups using neural network and adaptive neuro-fuzzy inference system. Appl. Soft Comput. 9, 746-755.

Zurada, J.M., 1992. Introduction to Artificial Neural Systems. West Publishing Company, St. Paul.

State-of-the-art review of some artificial intelligence applications in pile foundations

Mohamed A. Shahin

Abstract

Geotechnical engineering deals with materials (e.g. soil and rock) that, by their very nature, exhibit varied and uncertain behavior due to the imprecise physical processes associated with the formation of these materials. Modeling the behavior of such materials in geotechnical engineering applications is complex and sometimes beyond the ability of most traditional forms of physically-based engineering methods. Artificial intelligence (AI) is becoming more popular and particularly amenable to modeling the complex behavior of most geotechnical engineering applications because it has demonstrated superior predictive ability compared to traditional methods. This paper provides state-of-the-art review of some selected AI techniques and their applications in pile foundations, and presents the salient features associated with the modeling development of these AI techniques. The paper also discusses the strength and limitations of the selected AI techniques compared to other available modeling approaches.

1. Introduction

Over the last decade, artificial intelligence (AI) has been applied successfully to virtually every problem in geotechnical engineering. Examples of the available AI techniques are artificial neural networks (ANNs), genetic programming (GP), evolutionary polynomial regression (EPR), support vector machines (SVM), M5 model trees, and k-nearest neighbors (Elshorbagy et al., 2010). Of these, ANNs are by far the most commonly used AI technique in geotechnical engineering. More recently, GP and EPR have been frequently used in geotechnical engineering and have proved to be successful. The main focus of the current paper is on the use of ANNs, GP, and EPR in pile foundations.

The behavior of pile foundations in soils is complex, uncertain, and not yet entirely understood. This fact has encouraged many researchers to apply the AI techniques for prediction and modelling of the behavior of pile foundations, including the ultimate bearing capacity, settlement estimation, and load-settlement response. The objective of this paper is to provide an overview of the salient features relevant to the process and operation of ANNs, GP, and EPR, and to present a review of their applications to date in pile foundations. The paper also discusses most of the current challenges as well as future directions in relation to the use of AI techniques in geotechnical engineering prediction and modelling.

2. Overview of artificial intelligence

Artificial intelligence (AI) is a computational method that attempts to mimic, in a very simplistic way, the human cognition capability so as to solve engineering problems that have defied solution using conventional computational techniques (Flood, 2008). The essence of AI techniques in solving any engineering problem is to learn by examples of data inputs and outputs presented to them so that the subtle functional relationships among the data are captured, even if the underlying relationships are unknown or the physical meaning is difficult to explain. Thus, AI models are data-driven models that rely on the data alone to determine the structure and parameters that govern a phenomenon (or system), with less assumptions about the physical behavior of the system. This is in contrast to most physically-based models that use the first principles (e.g., physical laws) to derive the underlying relationships of the system, which usually justifiably simplified with many assumptions and require prior knowledge about the nature of the relationships among the data. This is one of the main benefits of AI techniques when compared to most physically-based empirical and statistical methods.

The AI modeling philosophy in attempting to capture the relationship between a historical set of model inputs and the corresponding outputs is similar to a number of conventional statistical models. For example, imagine a set of x-values and corresponding y-values in two-dimensional space, where $y = f(x)$. The objective is to find the unknown function f that relates the input variable x to the output variable y. In a linear regression statistical model, the function f can be obtained by changing the slope $\tan\varphi$ and intercept β of the straight line in Fig. 1a, so that the error between the actual outputs and the outputs of the straight line is minimized. The same principle is used in AI models. Artificial intelligence can form the simple linear regression model by having one input and one output (Fig. 1b). Artificial intelligence uses available data to map between the system inputs and the corresponding outputs using machine learning by repeatedly presenting examples of the model inputs and outputs (training) in order to find the function $y = f(x)$ that minimizes the error between the historical (actual) outputs and the outputs predicted by the AI model.

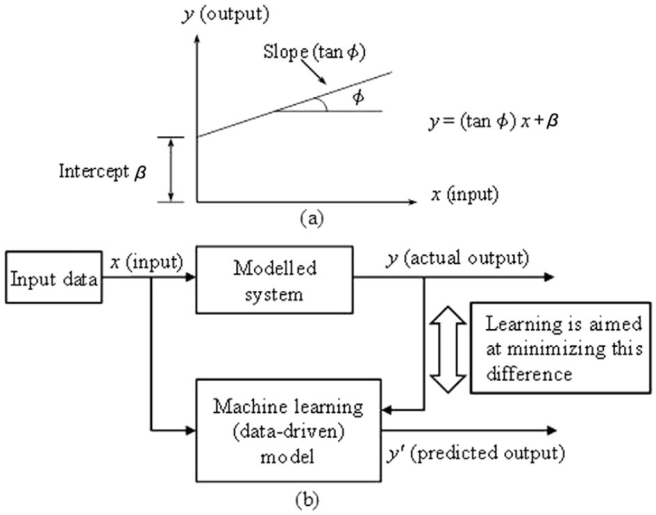


Figure 1. Linear regression versus artificial intelligence (AI) modeling: (a) linear regression modeling (after Shahin et al., 2001); (b) AI data-driven modeling (adapted from Solomatine and Ostfeld, 2008).

If the relationship between x and y is non-linear, statistical regression analysis can be applied successfully only if prior knowledge of the nature of the non-linearity exists. On the contrary, this prior knowledge of the nature of the non-linearity is not required for AI models. In the real world, it is likely that complex and highly non-linear problems are encountered, and in such situations, traditional regression analyses are inadequate (Gardner and Dorling, 1998). In this section, a brief overview of three selected AI techniques (i.e., ANNs, GP, and EPR) is presented below.

2.1. Artificial neural networks

Artificial neural networks (ANNs) are a form of AI that attempt to mimic the function of the human brain and nervous system. Although the concept of ANNs was first introduced in 1943 (McCulloch and Pitts, 1943), research into applications of ANNs has blossomed since the introduction of the back-propagation training algorithm for feed-forward multi-layer perceptrons (MLPs) in 1986 (Rumelhart et al., 1986). Many authors have described the structure and operation of ANNs (e.g., Zurada, 1992; Fausett, 1994). Typically, the architecture of ANNs consists of a series of processing elements (PEs), or nodes, that are usually arranged in layers: an input layer, an output layer, and one or more hidden layers, as shown in Fig. 2.

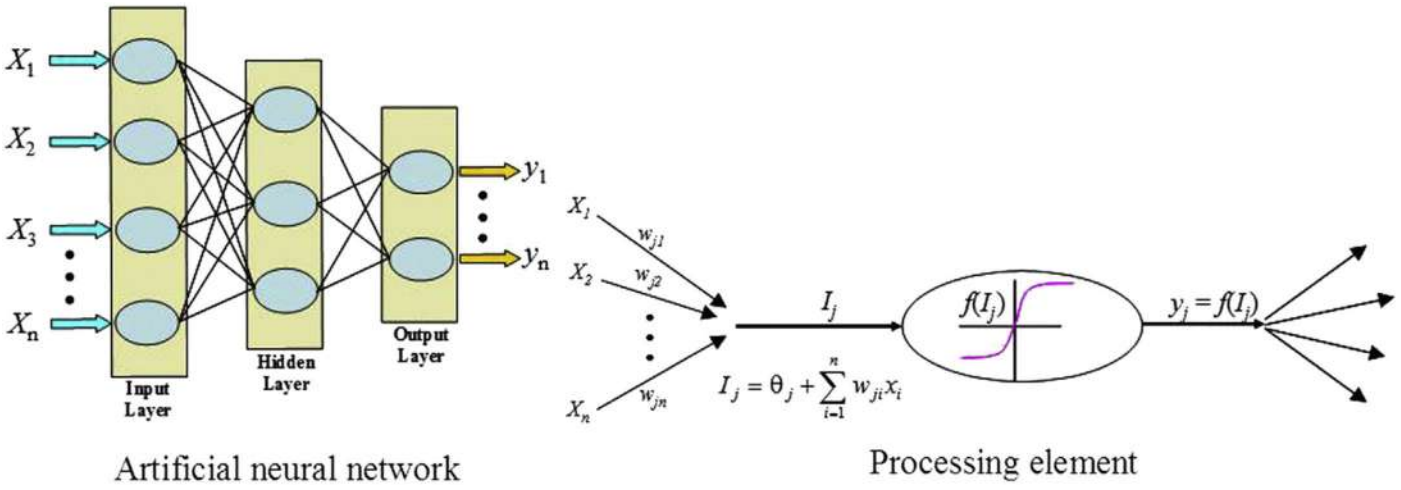


Figure 2. Typical structure and operation of artificial neural networks (ANNs) (after Shahin et al., 2009).

The input from each PE in the previous layer x_i is multiplied by an adjustable connection weight w_{ji}. At each PE, the weighted input signals are summed and a threshold value θ_j is added. This combined input I_j is then passed through a non-linear transfer function $f(.)$ to produce the output of the PE y_j. The output of one PE provides the input to the PEs in the next layer. This process is summarized in Eqs. (1) and (2), and illustrated in Fig. 2.

$$I_j = \sum w_{ji} x_i + \theta_j \quad \text{summation} \tag{1}$$

$$y_j = f\left(I_j\right) \quad \text{transfer} \tag{2}$$

The propagation of information in an ANN starts at the input layer, where the input data are presented. The network adjusts its weights on the presentation of a training data set and uses a learning rule to find a set of weights that produces the input/output mapping that has the smallest possible error. This process is called learning or training. Once the training of the model has successfully accomplished, the performance of the trained model needs to be validated using an independent validation set. The main steps involved in the development of an ANN, as suggested by Maier and Dandy (2000), are illustrated in Fig. 3 and discussed in some depth in Shahin (2013).

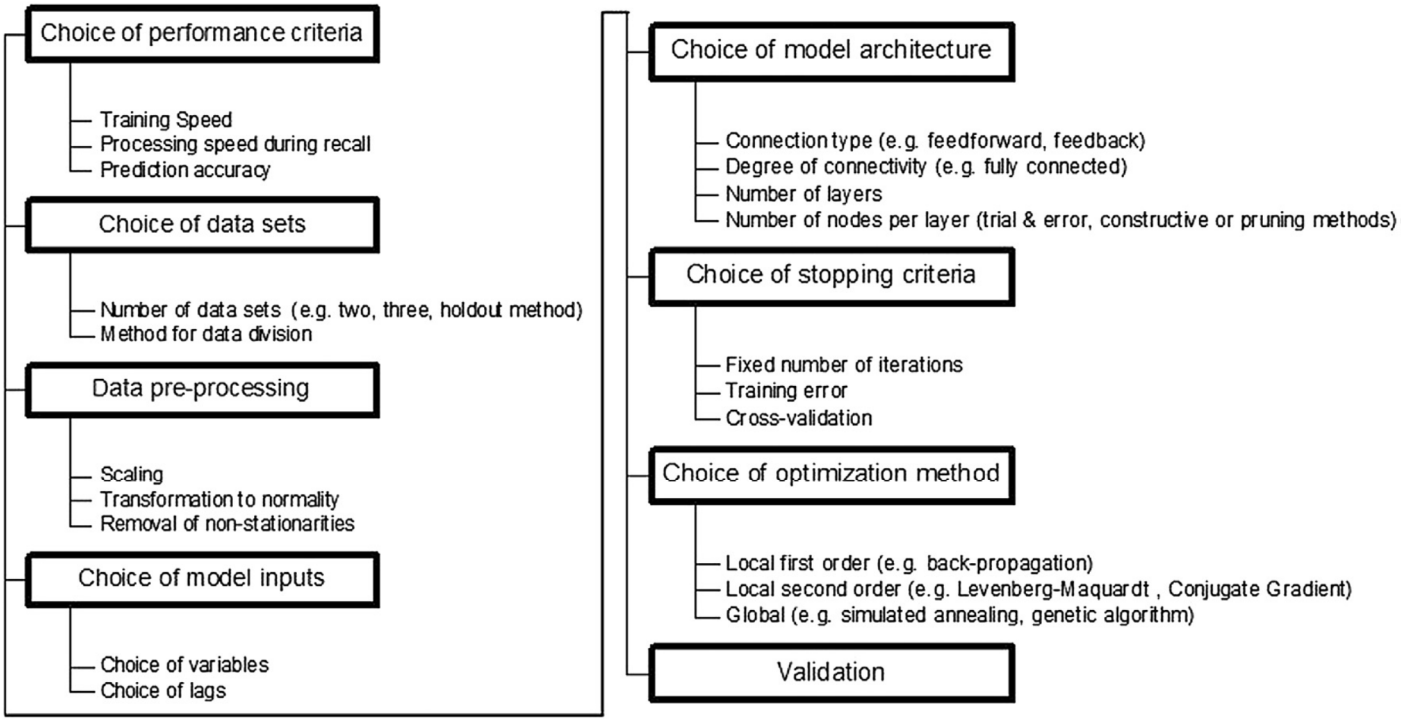


Figure 3. Main steps in artificial neural network (ANN) model development (Maier and Dandy, 2000).

2.2. Genetic programming

Genetic programming (GP) is an extension of genetic algorithms (GA), which are evolutionary computing search (optimization) methods that are based on the principles of genetics and natural selection. In GA, some of the natural evolutionary mechanisms, such as reproduction, cross-over, and mutation, are usually implemented to solve function identifi-

cation problems. GA was first introduced by Holland (1975) and developed by Goldberg (1989), whereas GP was invented by Cramer (1985) and further developed by Koza (1992). The difference between GA and GP is that GA is generally used to evolve the best values for a given set of model parameters (i.e., parameters optimization), whereas GP generates a structured representation for a set of input variables and corresponding outputs (i.e., modeling or programming).

Genetic programming manipulates and optimizes a population of computer models (or programs) proposed to solve a particular problem, so that the model that best fits the problem is obtained. A detailed description of GP can be found in many publications (e.g., Koza, 1992), and a brief overview is given herein. The modelling steps by GP start with the creation of an initial population of computer models (also called *chromosomes*) that are composed of two sets (i.e., a set of functions and a set of terminals) that are defined by the user to suit a certain problem. The functions and terminals are selected randomly and arranged in a treelike structure to form a computer model that contains a root node, branches of functional nodes, and terminals, as shown by the typical example of GP tree representation in Fig. 4. The functions can contain basic mathematical operators (e.g., +, -, x, /), Boolean logic functions (e.g., AND, OR, NOT), trigonometric functions (e.g., sin, cos), or any other user-defined functions. The terminals, on the other hand, may consist of numerical constants, logical constants, or variables.

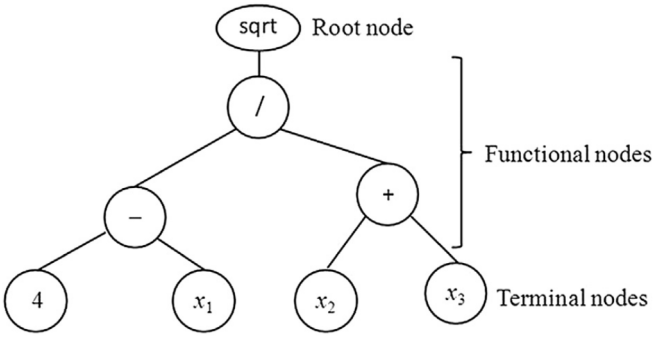


Figure 4. Typical example of genetic programming (GP) tree representation for the function $[(4-x_1)/(x_2+x_3)]^2$.

Once a population of computer models has been created, each model is executed using available data for the problemat hand, and the model fitness is evaluated depending on how well it is able to solve the problem. For many problems, the model fitness is measured by the error between the output provided by the model and the desired actual output. One could measure the fitness f_i of an individual chromosome i using the following expression:

$$f_i = \sum_{j=1}^{C_t} \left(M - \left| C_{(i,j)} - T_j \right| \right) \quad (3)$$

where M is the range of selection, $C_{(i,j)}$ is the value returned by the individual chromosome i for fitness case j (out of C_t fitness cases), and T_j is the target value for the fitness case j. There are, of course, other fitness functions available that can be appropriate for different problems. If the desired results (according to the measured errors) are satisfactory, the GP process is stopped, otherwise, a generation of new population of computer models is then created to replace the existing population, and the process is repeated for a certain number of generation or until the desired fitness score is obtained. The new population is created by applying the following three main operations: reproduction, cross-over, and mutation. These three operations are applied on certain proportions of the computer models in the existing population, and the models are selected according to their fitness. Reproduction is copying a computer model from an existing population into the new population without alteration. Cross-over is genetically recombining (swapping) randomly chosen parts of two computer models. Mutation is replacing a randomly selected functional or terminal node with another node from the same function or terminal set, provided that a functional node replaces a functional node and a terminal node replaces a terminal node. The evolutionary process of evaluating the fitness of an existing population and producing new population is continued until a termination criterion is met, which can be either a particular acceptable error or a certain maximum number of generations. The best computer model that appears in any generation designates the result of the GP process.

2.3. Evolutionary polynomial regression

Evolutionary polynomial regression (EPR) is a hybrid regression technique based on evolutionary computing that was developed by Giustolisi and Savic (2006). It constructs symbolic models by integrating the soundest features of numerical regression, with genetic programming and symbolic regression (Koza, 1992). The following two steps roughly describe the underlying features of the EPR technique, aimed to search for polynomial structures representing a system. In the first step, the selection of exponents for polynomial expressions is carried out, employing an evolutionary searching strategy by means of GA (Goldberg, 1989). In the second step, numerical regression using the least square method is conducted, aiming to compute the coefficients of the previously selected polynomial terms. The general form of expression in EPR can be presented as follows (Giustolisi and Savic, 2006):

$$y = \sum_{j=i}^{m} F(X, f(X), a_j) + a_o \quad (4)$$

where y is the estimated vector of output of the process, m is the number of terms of the target expression, F is a function constructed by the process, X is the matrix of input variables, f is a function defined by the user, and a_j is a constant. A typical example of EPR pseudo-polynomial expression that belongs to the class of Eq. (4) is as follows (Giustolisi and Savic, 2006):

$$\hat{Y} = a_o + \sum_{j=i}^{m} a_j \cdot (X_1)^{ES(j,1)} \ldots (X_k)^{ES(j,k)} \cdot f\left[(X_1)^{ES(j,k+1)} \ldots (X_k)^{ES(j,2k)} \right] \quad (5)$$

where $\hat{Y}$ is the vector of target values, m is the length of the expression, a_j is the value of the constants, X^i is the vector(s) of the k candidate inputs, ES is the matrix of exponents, and f is a function selected by the user.

Evolutionary polynomial regression is suitable for modeling physical phenomena, based on two features (Savic et al., 2006): (i) the introduction of prior knowledge about the physical system/process, to be modeled at three different times, namely before, during, and after EPR modelling calibration; and (ii) the production of symbolic formulas, enabling data mining to discover patterns that describe the desired parameters. In the first EPR feature (i) above, before the construction of the EPR model, the modeler selects the relevant inputs and arranges them in a suitable format according to their physical meaning. During the EPR model construction, model structures are determined by following userdefined settings such as general polynomial structure, user defined function types (e.g., natural logarithms, exponentials, tangential hyperbolics), and searching strategy parameters. The EPR starts from true polynomials and also allows for the development of non-polynomial expressions containing user-defined functions (e.g., natural logarithms). After EPR model calibration, an optimum model can be selected from among the series of models returned. The optimum

model is selected based on the modeller's judgement, in addition to statistical performance indicators such as the coefficient of determination. A typical flow diagram of the EPR procedure is shown in Fig. 5, and a detailed description of the technique can be found in Giustolisi and Savic (2006).

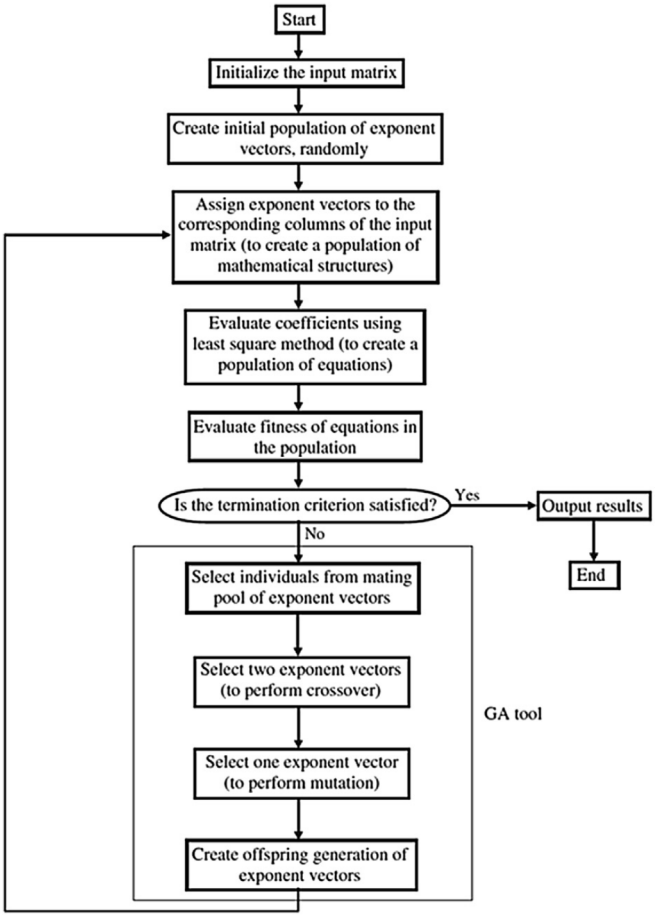


Figure 5. Typical flow diagram of the evolutionary polynomial regression (EPR) procedure (after Rezania et al., 2011).

3. Artificial intelligence applications in pile foundations

This section provides an overview of the applications of three selected AI techniques, including ANNs, GP, and EPR, that have appeared to date in relation to examining the relative success or otherwise of AI in pile foundations. It should be noted that it is not intended in the current paper to cover every single application or scientific paper of the three selected AI techniques in pile foundations that can be found in the literature but rather the intention is to provide a general overview of some of the more relevant applications in engineering problem of pile foundations. Some works are selected to be described in some detail, while others are acknowledged for reference purposes. On the other hand, the applications of the three selected AI techniques in geotechnical engineering are beyond the scope of the current paper and can be found elsewhere. Interested readers are referred to Shahin et al. (2001), where the pre-2001 ANN applications in geotechnical engineering are reviewed in some detail, and Shahin et al. (2009) and Shahin (2013), where the post-2001 papers of ANN applications in geotechnical engineering are briefly examined. Interested readers are also referred to Shahin (2013), where applications of GP and EPR in geotechnical engineering are presented.

Based on the author's experience, there are several factors in the use of AI techniques that need to be systematically investigated when developing AI models for geotechnical engineering problems, including pile foundations, so that model performance can be improved. These factors include the determination of adequate model inputs, data division, data preparation, model validation, model robustness, model transparency, knowledge extraction, and model uncertainty. Some of these factors have received recent attention, whereas others require further research. Discussion of these factors are beyond the scope of this paper but can be found in Shahin (2013). Some of these factors are briefly discussed in the applications presented below.

3.1. Bearing capacity prediction

The design of foundations is generally controlled by two major criteria, i.e., bearing capacity and settlement. For pile foundations, prediction of the load carrying capacity is often being the governing factor; hence, has been examined by several AI researchers especially using ANNs. For example, Goh (1994, 1995b) presented a neural network model to predict the friction capacity of piles in clays and the model was trained with field data of actual case records. The considered model inputs were the pile length, pile diameter, mean effective stress, and undrained shear strength. The skin friction resistance was the only model output. The results obtained from the neural network model were compared with those calculated using the method proposed by Semple and Rigden (1986) as well as the β method developed by Burland (1973), as shown in Table 1. The performance measures used were the coefficient of correlation, r, and error rate between the predicted versus measured bearing capacities. It is evident from Table 1 that the ANN model outperforms the conventional methods. Goh (1995a, 1996), soon after, developed another neural network model to estimate the ultimate load capacity of driven piles in cohesionless soils. In this study, the data used were derived from the results of load testing carried out on piles made of timber, precast concrete, and steel, driven into sandy soils. The inputs to the ANN model that found to be more significant were the hammer weight, drop and type, and pile length, weight, cross sectional area, set and modulus of elasticity. The model output was the pile load capacity. When the model was examined using a testing set, it was observed that the model successfully predicted the pile load capacity. By examining the connection weights, it was observed that the more important input factors are the pile set as well as the hammer weight and type. The study compared the results of the ANN model with the following common formulae: Engineering News formula (Wellington, 1892), Hiley method (Hiley, 1922), and Janbu method (Janbu, 1953). Table 2 summarises the results, which indicate that ANN predictions of the load carrying capacity of driven piles are significantly better than those obtained from the traditional methods. More recently, Goh et al. (2005) used a Bayesian neural network algorithm to model the relationship between the soil undrained shear strength, effective overburden pressure, and undrained side resistance alpha factor for drilled shafts (bored piles). The advantage of using the Bayesian ANN approach is that instead of just giving a single prediction as in conventional backpropagation ANN, it produces a probability distribution over the predicted value. The benefit of this distribution is that it provides information on the characteristic error of the prediction that arises from the uncertainty associated with interpolating noisy data. It also allows assessment of the confidence associated with any prediction. The model was trained using a database that contained 127 field load tests on drilled shafts in a variety of cohesive soil profiles. Comparison was made between the ANN predictions and those obtained from the method proposed by Chen and Kulhawy (1994). The comparison indicated that the ANN model was reasonably accurate in its predictions and achieved an improvement over those calculated using the method of Chen and Kulhawy (1994), especially in the training set.

Among the available methods for predicting the axial capacity of pile foundations that have been shown to give better predictions in many situations, are the cone penetration test

(CPT)-based models. This can be attributed to the fact that CPT-based methods have been developed in accordance with the CPT results, which have been found to yield reliable soil properties; hence, more accurate axial pile capacity predictions. In an attempt to develop more well established CPT-based pile capacity prediction models that provide more accurate axial capacity predictions, Shahin (2010) developed ANN models for driven piles and drilled shafts using a series of in-situ load tests, as well as CTP results. The data were collected form the literature and comprised 80 driven pile and 94 drilled-shaft load tests. The predictive ability of the ANN models was examined by comparing their predictions with those obtained from the most commonly used CPT-based pile capacity prediction methods. For driven piles, the ANN model was compared with the European method (de Ruiter and Beringen, 1979), Laboratoire Central des Ponts et Chaussees (LCPC) method (Bustamante and Gianeselli, 1982), and the method by Eslami and Fellenius (1997). For drilled shafts, the ANN model was compared with the Schmertmann method (Schmertmann, 1978), LCPC method (Bustamante and Gianeselli, 1982), and Alsamman method (Alsamman, 1995). The comparison was carried out analytically using the rank index, RI, proposed by Abu-Farsakh and Titi (2004), which comprises of four combined statistical performance criteria. Sensitivity analyses were also carried out on the ANN models to explore their generalization ability (robustness). The results indicated that the ANN models were capable of accurately predicting the ultimate capacity of pile foundations with high level of performance. The RI results yielded the following overall rank: ANN model (Shahin, 2010), Eslami and Fellenius (1997), LCPC method (Bustamante and Gianeselli, 1982), and European method (de Ruiter and Beringen, 1979). On the other hand, for drilled shafts, the results of RI showed an equal overall rank for the ANN model (Shahin, 2010) and the method proposed by Alsamman (1995), followed by the Schmertmann method (Schmertmann, 1978) and the LCPC method (Bustamante and Gianeselli, 1982). The sensitivity analyses indicated that predictions from the ANN models compare well with what one would expect based on available geotechnical knowledge and underlying physical meaning, as well as experimental results.

Table 1
Performance of artificial neural network (ANN) model and traditional methods for predicting friction capacity of piles in clays (Goh, 1995b).

Method	Coefficient of correlation (r)		Error rate (kPa)	
	Training	Testing	Training	Testing
ANN (Goh, 1995b)	0.985	0.956	1.016	1.194
Semple and Rigden (1986)	0.976	0.885	1.318	1.894
β method (Burland, 1973)	0.731	0.704	4.824	3.096

In an attempt to facilitate the use of the obtained ANN models and to make them more accessible, Shahin (2010) translated the connection weights and biases of the developed neural network models into tractable and relatively simple formula suitable for hand calculations. The derived formula can be used to calculate the ultimate bearing capacity of driven piles, Q_u (kN), as follows (Shahin, 2010):

$$Q^{ANN}_{u(driven.piles)} = 290 + \left[\frac{4210}{1+e^{(-1.699-4.193\tanh H_1+2.242\tanh H_2)}}\right] \quad (6)$$

H1 and *H2* are two parameters obtained for steel piles, as follows:

$$H_1 = -5.1 + 10^{-3}\left(3.59D_{eq} + 45.51L_p + 112.23\bar{q}_{c-tip} - 21.39\bar{q}_{c-shaft} + 6.86\bar{f}_s\right) \quad (7)$$

$$H_2 = 1.164 - 10^{-3}\left(2.47D_{eq} + 33.96L_p - 8.37\bar{q}_{c-tip} + 1.58\bar{q}_{c-shaft} - 0.24\bar{f}_s\right) \quad (8)$$

where D_{eq} (mm) is the equivalent pile diameter, L_p (m) is the pile embedment length, $\bar{q}_{c\text{-}tip}$ (MPa) is the weighted average cone point resistance over pile tip failure zone; $\bar{q}_{c\text{-}shaft}$ (MPa) is the weighted average cone point resistance along pile embedment length, and f_s (kPa) is the weighted average sleeve friction along pile embedment length.

Alternatively, for concrete piles:

$$H_1 = -5.158 + 10^{-3}\left(3.59D_{eq} + 45.51L_p + 112.23\bar{q}_{c-tip} - 21.39\bar{q}_{c-shaft} + 6.86\bar{f}_s\right) \quad (9)$$

$$H_2 = 0.816 - 10^{-3}\left(2.47D_{eq} + 33.96L_p - 8.37\bar{q}_{c-tip} + 1.58\bar{q}_{c-shaft} - 0.24\bar{f}_s\right) \quad (10)$$

On the other hand, the ultimate drilled shafts capacity, Q_u (kN), can be calculated as follows:

$$Q^{ANN}_{u(drilled.shafts)} = 355.8 + \left[\frac{9296.3}{1+e^{(-1.673-3.364\tanh H_1+4.223\tanh H_2+3.336\tanh H_3)}}\right] \quad (11)$$

$$H_1 = -6.509 + 10^{-3}\left(1.069D_{stem} + 2.351D_{base} - 41.152L_s - 2.174\bar{q}_{c-base} + 11.271\bar{q}_{c-shaft}\right) \quad (12)$$

$$H_2 = 0.528 + 10^{-3}\left(0.553D_{stem} + D_{base} + 38.75L_s + 1.59\bar{q}_{c-base} + 5.344\bar{q}_{c-shaft}\right) \quad (13)$$

$$H_3 = 3.777 + 10^{-3}\left(0.772D_{stem} - 0.537D_{base} + 83.37L_s + 23.31\bar{q}_{c\_base} + 56.23\bar{q}_{c\_shaft}\right) \quad (14)$$

where D_{stem} (mm) is the shaft stem diameter, D_{base} (mm) is the shaft base diameter, L (m) is the shaft embedment length, $\bar{q}_{c\_base}$ (MPa) is the weighted average cone point resistance over shaft base failure zone, and $\bar{q}_{c\_shaft}$ (MPa) is the weighted average cone tip resistance along shaft embedment length.

Shahin and Jaksa (2005, 2006) assessed the applicability of ANNs for predicting the pull-out capacity of marquee ground anchors (these are, in effect, micro-piles) using multi-layer perceptrons (MLPs) and B-spline neurofuzzy networks. Neurofuzzy networks are a type of ANN modeling technique that combines the explicit linguistic knowledge representation of fuzzy systems with the learning power of neural networks (Brown and Harris, 1995). Neurofuzzy networks can be trained by processing data samples to perform input/output mappings, similar to the way traditional neural networks do, with the additional benefit of being able to provide a set of production *If-then* linguistic fuzzy rules that describe the model input/output relationships in a transparent way, such as:

$$\text{IF}(x_1 \textit{ is high AND } x_2 \text{ is low) THEN } (y \text{ is high}), c = 0.9 \quad (15)$$

where x_1 and x_2 are input variables, y is the corresponding output variable, and ($c = 0.9$) is the rule confidence which indicates the degree to which the above rule has contributed to the output. Both the MLP and B-spline neurofuzzy models were trained using five inputs including the anchor diameter, anchor embedment length, average cone tip resistance from the cone penetration test along the anchor embedment length, average cone sleeve friction along the embedment length, and installation technique. The single model output was the ultimate anchor pull-out capacity. The results obtained were also compared with those obtained from three of the most commonly used traditional methods, namely, the LCPC method proposed by Bustamante and Gianeselli (1982), and the methods proposed by Das (1995) and Bowles (1997). The results indicated that the MLP and B-spline models were able to predict well the pull-out capacity of marquee ground anchors and significantly outperform the traditional methods. Over the full range of pull-out capacity prediction, the coefficients of correlation, r, using the MLP and B-spline models were 0.83 and 0.84, respectively. In contrast, these measures ranged from 0.46 to 0.69 when the other methods were used.

To predict the pile capacity from dynamic testing data, Chan et al. (1995) developed a neural network model as an alternative to the commonly used pile driving formula approach. The neural network model was trained with the same input parameters listed in the simplified Hiley formula (Broms and Lim, 1988), including the elastic compression of pile and soil, pile set, and driving energy delivered to the pile. The model output considered was, again, the pile capacity. The root mean squared percentage error of the neural network model was found to be 13.5 and 12.0% for the training and testing sets, respectively, compared with 15.7% in both the training and testing sets for the simplified Hiley formula.

Lee and Lee (1996) utilized neural networks to predict the ultimate bearing capacity of piles using data obtained from a calibration chamber model pile load tests as well as results of in-situ pile load tests. For the simulation using the model pile load test data, the neural network model inputs were the penetration depth ratio (i.e., penetration depth of pile/pile diameter), mean normal stress of the calibration chamber, and number of blows. The ultimate bearing capacity was the model output. The prediction of the neural network model showed a maximum error not greater than 20% and an average summed square error of less than 15%. For simulation using the in-situ pile load test data, five input variables were used representing the penetration depth ratio, average standard penetration number along the pile shaft, average standard penetration number near the pile tip, pile set, and hammer energy. The data were arbitrarily partitioned into two parts, odd and even numbered sets and two neural network models were developed. The results of these models were compared with Meyerhof's equation (Meyerhof, 1976), based on the average standard penetration value. The results of the estimated versus measured pile bearing capacities obtained from the neural network models and Meyerhof's equation showed that the predicted values from the neural networks matched the measured values much better than those obtained from Meyerhof's equation.

Abu-Kiefa (1998) introduced three ANN models (referred to in his paper as GRNNM1, GRNNM2, and GRNNM3) to predict the capacity of driven piles in cohesionless soils. The first model was developed to estimate the total pile capacity, whereas the second and third models were employed to estimate the pile tip and shaft capacities. In the first model, five variables were selected to be the model inputs including the angle of shear resistance for the soil surrounding the pile shaft, angle of shear resistance of soil at the tip of the pile, effective overburden pressure at the tip of the pile, pile length, and equivalent pile cross-sectional area. The model, again, had one output representing the total pile capacity. In the model used to evaluate the pile tip capacity, the above variables were also used. The number of input variables used to predict the pile shaft capacity was four, representing the average standard penetration number around the pile shaft, angle of shear resistance around the pile shaft, pile length, and pile diameter. The results of the neural network models obtained in this study were compared with four other empirical techniques including those proposed by Meyerhof (1976), Coyle and Castello (1981), American Petroleum Institute (1984), and Randolph (1985). The results of the total pile capacity prediction demonstrated high coefficients of determination ($R^2 = 0.95$) for all data records obtained from the neural network models, while those for the other methods ranged between 0.52 and 0.63.

Teh et al. (1997) proposed a neural network model for estimating the static pile capacity determined from dynamic stresswave data for precast reinforced concrete piles with a square section. The neural network model was trained to associate the input stress-wave data with capacities derived from the CAPWAP technique (Rausche et al., 1972). The study was concerned with predicting the 'CAPWAP predicted capacity' rather than the true bearing capacity of piles. The neural network model learned the training data set almost perfectly for predicting the static total pile capacity with a root mean square error of less than 0.0003. The trained neural network model was assessed for its ability to generalize by means of a testing data set. Good prediction was obtained for seven out of ten piles. Another application of ANNs includes the prediction of axial and lateral load capacity of steel H-piles, steel piles and pre-stressed and reinforced concrete piles by Nawari et al. (1999). In this application, ANNs were found to be an accurate technique for the design of pile foundations. Prediction of the undrained lateral load pile capacity of piles in clay was modelled using ANNs by Das and Basudhar (2006), and a model equation based on the produced neural network parameters was developed.

Other ANN applications in pile foundations include predicting the total pile capacity by generalized regression neural networks developed using stress-wave data (Pal and Deswal, 2010), modelling pile shaft capacity from CPT and CPTU data by polynomial neural networks (Ardalan et al., 2009), predicting the total resistance of driven piles as well as the resistance at the tip and along the shaft using dynamic load tests (Park and Cho, 2010), predicting pile setup for three pile types (pipe, concrete, and H-pile) using dynamic load tests (Tarawneh, 2013; Tarawneh and Imam, 2014), and analysing mechanism of time effect and soil consolidation on vertical ultimate bearing capacity of preformed concrete piles (Tian et al., 2010).

The application of GP technique in estimating the capacity of pile foundations is relatively recent. Alkroosh and Nikraz (2011a, 2012) developed GP correlation models for predicting the relationship between pile axial capacity and CPT data. The GP models were developed for bored piles as well as driven piles (a model for each of concrete and steel piles). The performance of the GP models was evaluated by comparing their results with experimental data as well as the results of a number of currently used CPT-based methods. The results indicated the potential ability of GP models in predicting the bearing capacity of pile foundations and outperformance of the developed models over existing methods. More recently, Alkroosh and Nikraz (2014) developed a GP model that correlates the pile capacity with the dynamic input and SPT data. The performance of the GP model was assessed by comparing its predictions with those calculated using two commonly used traditional methods and an ANN model. It was found that the GP model performed well with coefficients of determination of 0.94 and 0.96 in the training and testing sets, respectively. The results of comparison with other available methods showed that the GP model predicted the pile capacity more accurately than the existing traditional methods and ANN model. Another successful application of Genetic programming in pile capacity prediction was carried out

by Gandomi and Alavi (2012) for the assessment of the undrained lateral load capacity of driven piles and undrained side resistance alpha factor of drilled shafts.

The application of EPR in predicting the capacity of pile foundations is very recent. Using the same database of Shahin (2010) and similar model inputs and outputs, Shahin (2014c) developed successful EPR models for predicting the axial capacity, Q_u (kN), of driven piles and drilled shafts. The formulations of the developed EPR models are as follows (Shahin, 2014c):

For driven (steel) piles:

$$Q^{EPR}_{u(steel.driven)} = -2.77\frac{D\bar{q}_{c-tip}}{\sqrt{\bar{q}_{c-shaft}\bar{f}_{s-shaft}}} + 0.096DL + 1.714 \times 10^{-4}D^2\bar{q}_{c-tip}\sqrt{L} - 6.279 \times 10^{-9}D^2L^2\sqrt{\bar{q}_{c-tip}\bar{f}_{s-tip}} + 243.39 \quad (16)$$

Alternatively, for driven (concrete) piles:

$$Q^{EPR}_{u(concrete.driven)} = -2.777\frac{D\bar{q}_{c-tip}}{\sqrt{\bar{q}_{c-shaft}\bar{f}_{s-shaft}}} + 0.096DL + 1.714 \times 10^{-4}D^2\bar{q}_{c-tip}\sqrt{L} - 6.279 \times 10^{-9}D^2L^2\sqrt{\bar{q}_{c-tip}\bar{f}_{s-tip}} + 486.78 \quad (17)$$

For drilled shafts:

$$Q^{EPR}_{u(drilled.shafts)} = 0.6878L^2\sqrt{\bar{f}_{s-shaft}} + 1.581 \times 10^{-4}B^2\sqrt{\bar{f}_{s-shaft}} + 1.294 \times 10^{-4}L^2\bar{q}^2_{c-tip}\sqrt{D} + 7.8 \times 10^{-5}D\bar{q}_{c-shaft}\bar{f}_{s-shaft}\sqrt{\bar{f}_{s-tip}} \quad (18)$$

where D (mm) is the pile perimeter/π (for driven piles) or pile stem diameter (for drilled shafts), L (m) is the pile embedment length, B (mm) is the drilled shaft base diameter, $\bar{q}_{c\_tip}$ (MPa) is the weighted average cone point resistance over pile tip failure zone, $\bar{f}_{s\_tip}$ (kPa) is the weighted average cone sleeve friction over pile tip failure zone, q_{c_shaft} (MPa) is the weighted average cone point resistance over pile embedment length, and $\bar{f}_{s\_shaft}$ (kPa) is the weighted average cone sleeve friction over pile embedment length. The above EPR models represented by equations we compared with the traditional methods and were found to outperform most available methods.

Finally, using the same database of Shahin and Jaksa (2005, 2006) and similar model inputs and outputs, Shahin (2014c) developed successful EPR models for predicting the ultimate pullout capacity of marquee anchors, Q_u (kN), that yielded the following two formula, for static and dynamics installation, respectively:

$$Q^{EPR}_{u(static)} = -0.376\sqrt{\bar{q}_c} - 6.727 \times 10^{-9}L\bar{q}^2_c\bar{f}^2_s + 5.357 \times 10^{-5}L\sqrt{D\bar{q}_c\bar{f}_s} + 0.75 \quad (19)$$

$$Q^{EPR}_{u(dynamic)} = -0.376\sqrt{2\bar{q}_c} - 6.727 \times 10^{-9}L\bar{q}^2_c\bar{f}^2_s + 5.357 \times 10^{-5}L\sqrt{D\bar{q}_c\bar{f}_s} + 0.75 \quad (20)$$

where D (mm) is the equivalent anchor parameter (= anchor perimeter/π), L (m) is the anchor embedment length, $\bar{q}_{c\_shaft}$ (MPa) is the arithmetic average cone tip resistance along the embedment length, and $\bar{f}_s$ (kPa) is the arithmetic average sleeve friction along the embedment length. The performance of the EPR models in the training and validation sets is given in Table 3, and the comparison of model performance in the validation set with the other available methods in given in Table 4. The methods used for comparison include the ANN model developed by Shahin and Jaksa (2005), LCPC method (Bustamante and Gianeselli, 1982), Das method (Das, 1995) and Bowles method (Bowles, 1997). The performance of the EPR models and comparison with other methods were evaluated using five different analytical standard measures including the coefficient of correlation, r, the coefficient of determination, R^2, root mean squared error, RMSE, mean absolute error, MAE, and ratio of average measured to predicted outputs, ì. It can be seen in Table 3 that the EPR models perform well in the training and validation sets, and that the EPR models outperform the other available methods including the ANN model.

Table 3
Analytical performance of EPR model for pull-out capacity of ground anchors (Shahin, 2014a).

Performance measure	Training set	Validation set
r	0.789	0.872
R^2	0.619	0.753
RMSE (kN)	0.46	0.43
MAE (kN)	0.34	0.37
μ	1.02	0.99

Table 4
Comparison of EPR model and other methods in the validation set for pull-out capacity of ground anchors (Shahin, 2014a).

Performance measure	Method				
	EPR (Shahin, 2014a)	ANNs (Shahin and Jaksa, 2005)	LCPC (1982)	Das (1995)	Bowles (1997)
r	0.872	0.845	0.489	0.857	0.550
R^2	0.753	0.705	−0.455	−1.844	−0.102
RMSE (kN)	0.43	0.47	1.03	1.45	0.90
MAE (kN)	0.37	0.37	0.88	0.98	0.61
μ	0.99	0.95	1.84	0.72	1.86

3.2. Settlement estimation

Settlement is one of the two criteria that govern the design of pile foundations as settlement needs to be checked to ensure that it does not to exceed certain limits. However, settlement of pile foundations is less significant compared to bearing capacity and thus received less attention from the AI researchers. The number of AI publications for settlement prediction is significantly less than those of bearing capacity and solely related to the use of artificial neural networks (no applications are currently available for the use of either GP or EPR in settlement prediction of pile foundations). For example, Goh (1994) developed a neural network for the prediction of settlement of a vertically loaded pile foundation in a homogeneous soil stratum. The input variables for the neural network consisted of the ratio of the elastic modulus of the pile to the shear modulus of the soil, pile length, pile load, shear modulus of the soil, Poisson's ratio of the soil, and radius of the pile. The output variable was the pile settlement. The desired output that was used for the neural network model training was obtained by means of finite element and integral equation analyses developed by Randolph and Wroth (1978). A comparison of the theoretical and predicted settlements for the training and testing sets is given in Fig. 6. The results show that the neural network was able to model successfully the settlement of pile foundations.

Nawari et al. (1999) developed neural network models to predict the deflection of drilled shafts based on the standard

penetration test (SPT) data and the shaft geometry. The developed models involved back-propagation as well as generalized regression neural networks. Prediction results from the developed neural network models were compared with the classical technique, namely the *p-y* method, after Reese et al. (2006). The deviation of prediction of deflection with depth at a specific load level from the measured deflections, in case of the back-propagation neural network model, was found to be between 9 and 15%. On the other hand, the generalized neural network model gave prediction of good approximation and the deflection with depth was found to correlate very well with the predicted values with variation within 10%. The results also indicated that the neural network models correlate closer to the measured values than the *p-y* solution.

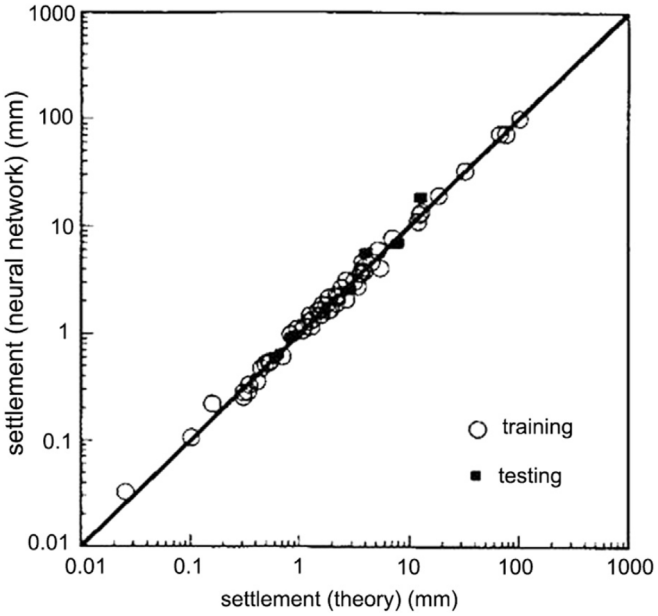


Figure 6. Comparison between theoretical settlements and artificial neural network (ANN) predictions for pile foundations (after Goh, 1994).

More recently, Nejad et al. (2009) developed neural network a model for predicting pile settlement also based on SPT data. Approximately 1000 data sets, obtained from the published literature, were used for model development. Model predictions were also compared with those obtained from a number of traditional methods; namely those of Vesic (1977), Poulos and Davis (1980), Das (1995), and the non-linear t-z method of Reese et al. (2006). The results indicated that the neural network model has the ability to predict the settlement of pile with an acceptable degree of accuracy of correlation coefficient $r = 0.972$ for settlement up to 185 mm sensitivity analyses carried out on the developed model indicated that the applied load, embedded length of pile, and soil properties, in this case the SPT-N values, have the most significant effect on the predicted settlement. It was also demonstrated that the neural network model outperforms the traditional methods and provides more accurate pile settlement predictions.

3.3. Load-settlement response modeling

As mentioned earlier, the design of pile foundations requires good estimation of the pile load-carrying capacity and settlement. Design for bearing capacity and design for settlement have been traditionally carried out separately. However, soil resistance and settlement are influenced by each other, and the design of pile foundations should thus consider the bearing capacity and settlement inseparably. This requires the full load-settlement response of piles to be well predicted. However, it is well known that the actual load-settlement response of pile foundations can be obtained only by load tests carried out in situ, which are expensive and time consuming. Consequently, some AI researchers have made attempts to develop AI prediction models that can resemble the full load-settlement response of piles. However, all attempts have used ANNs and no attempts are currently available that use either GP or EPR.

Shahin (2014a,b) used recurrent neural networks (RNN) to develop prediction models for the full load-settlement response of drilled shafts and steel driven piles, subjected to axial loading. The developed RNN models were calibrated and validated using several in-situ full-scale pile load tests, as well as cone penetration test (CPT) data. The tests were conducted on sites of different soil types and geotechnical conditions, ranging from cohesive clays to cohesionless sands including layered soils. Six factors affecting the capacity of piles were considered as potential model input variables. These factors include the pile diameter, pile embedment length, weighted average cone point resistance over pile tip failure zone, weighted average friction ratio over pile tip failure zone, weighted average cone point resistance over pile embedment length, and weighted average friction ratio over pile embedment length. Three other input variables are also considered to represent the current state of stress/strain including the normalized axial settlement (= pile settlement/pile diameter), increment of axial settlement, and pile load. The single model output variable is the pile load at the next state of loading. The models yielded high level of correlation between the measured and predicted data, and the graphical performance of the models in the training and validations sets are shown in Figs. 7 and 8. It can be seen that excellent agreement between the actual pile load tests and the RNN models' predictions are obtained for both the drilled shafts and driven piles. The nonlinear relationships of the load-settlement response are well predicted, and the results demonstrate that the RNN models have a strong capability to simulate the behaviour of pile foundations quite well.

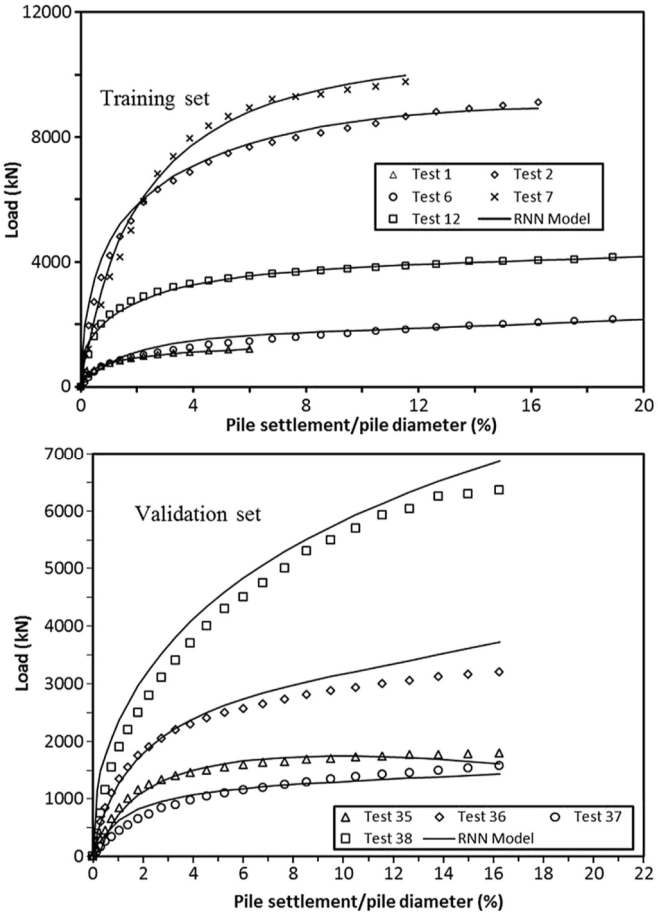


Figure 7. Some simulation results of the recurrent neural network (RNN) model in the training and validation sets for drilled shafts (after Shahin, 2014a).

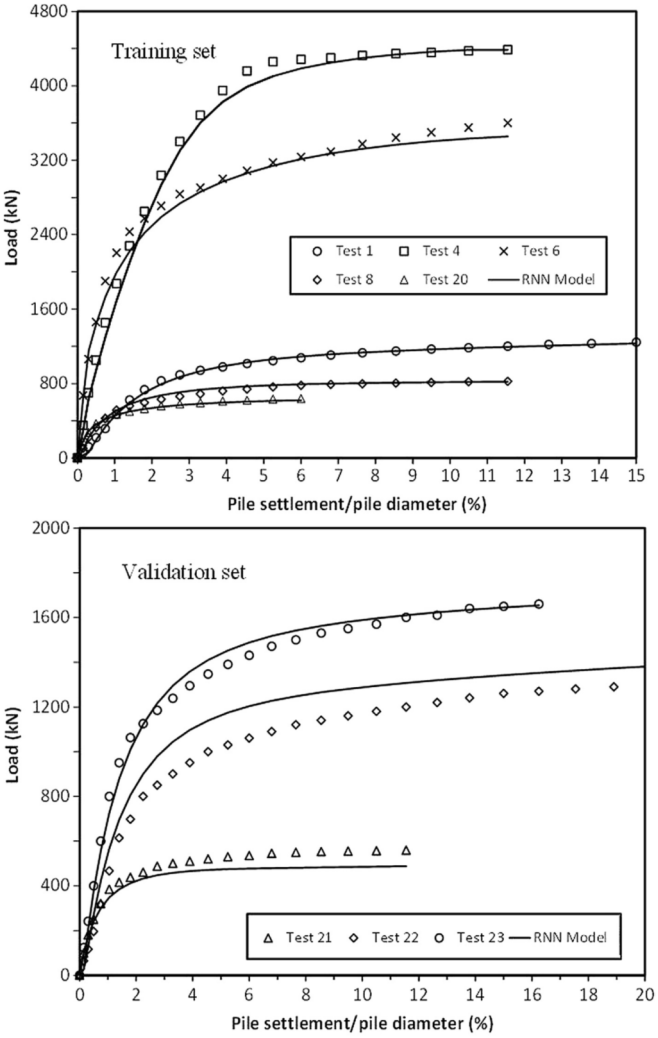


Figure 8. Some simulation results of the recurrent neural network (RNN) model in the training and validation sets for steel driven piles (after Shahin, 2014b).

Ismail and Jeng (2011) developed a high-order neural (HON) network to simulate the pile load-settlement curves using properties of the pile and SPT data along the depth of pile embedment as inputs. HON networks use polynomial functions to map inputs into output and can be trained through error back-propagation (BP) algorithm. As discussed by Ismail and Jeng (2011), the main advantage of HON networks over traditional BPN networks is that BPN networks use the sigmoid transfer function which is biasymptotic and becomes insensitive to the variation of inputs as it approaches either 1 or 0. This may limit the ability of BPN networks to make reasonable extrapolations outside the extreme values of inputs and outputs used in model training. On the contrary, HON networks use non-asymptotic processing elements (i.e., high-order neurons) to overcome such a problem. The input data used for the HON network consisted of the average value of SPT along the pile shaft, the SPT value at the pile base, the pile stiffness, the shaft and base area, and the pile load. Other parameters used include soil type and installation method. Based on the coefficient of determination and root mean squared error, as well as the quality of load-settlement curves, a significant improvement was observed from the comparison of HON model results with BPN, elastic and hyperbolic models. Also, the HON model was found to respond reasonably well to various input parameters in a manner consistent with the anticipated behaviour of axially loaded piles.

Ismail et al. (2013), soon after, developed a new load-deformation model for axially loaded piles by coupling the particle swarm optimisation (PSO) (Eberhart and Kennedy, 1995) and backpropagation (BP) algorithms for model training. The results showed that the proposed PSO-BP hybrid model simulates the load-deformation curves of axially loaded piles more accurately than previous HON model. The PSO-BP model also turned out to be more accurate than traditional hyperbolic and t-z models.

Alkroosh and Nikraz (2011b) also developed artificial neural network (ANN) models for simulating the load-settlement behavior of pile foundations embedded in sand or mixed soils, subjected to axial loads. Three ANN models were developed, a model for bored piles and two models for driven piles (a model for each of concrete and steel piles). The data used for development of the ANN models comprised a series of in-situ pile load tests as well as cone penetration test (CPT) results. Predictions from the ANN models were comrade with the results of experimental data and with predictions of number of currently adopted load-transfer methods. The results indicated that the ANN models perform well and able to predict the pile-settlement behavior accurately.

4. Discussion and conclusions

In geotechnical engineering, it is most likely to encounter problems that are very complex and not well understood. In this regard, artificial intelligence (AI) provides several advantages over more traditional computing techniques. For most traditional mathematical models, the lack of physical understanding is usually supplemented by either simplifying the problem or incorporating several assumptions into the models. Mathematical models also rely on assuming the structure of the model in advance, which may be sub-optimal. Consequently, many mathematical models fail to simulate the complex behavior of most geotechnical engineering problems. In contrast, AI techniques are data-driven approaches in which the model development is based on training of input-output data pairs to determine the structure and parameters of the model. In this case, there is less need to either simplify the problem or incorporate assumptions. Moreover, AI models can always be updated to obtain better results by presenting new training examples as new data become available. These factors combine to make AI a powerful modelling tool in geotechnical engineering.

It was evident from the review presented in this paper that AI techniques have been applied successfully to behavior of pile foundations including bearing capacity prediction, settlement estimation, and modeling of load-settlement response. However, most available applications focused on bearing capacity prediction and settlement estimation received less attention, which can be attributed to the fact that settlement of pile foundations is less significant than bearing capacity. In most reviewed AI applications in pile foundations, it was possible to provide simple formulations suitable for hand calculations for the relationships between the model inputs and the corresponding outputs. This helps to facilitate the use of the developed AI models and to make them accessible to the users. Based on the results of the reviewed applications, it can be concluded that AI techniques perform better than, or at least as good as, the most traditional methods.

Despite the success of AI techniques, they are still facing classical opposition due to some inherent shortcomings that need further attention in the future including the lack of transparency, knowledge extraction, and model uncertainty. Detailed discussion of such shortcomings is beyond the scope of this paper but have been presented in detail by Shahin (2013). For example, special attention should be paid to incorporating prior knowledge about the underlying physical process based on engineering judgment or human expertise into the learning formulation. Improvements in such issues will greatly enhance the usefulness of AI techniques and will provide the next generation of applied AI models with the best way for advancing the field to the next level of sophisti-

cation and application. The author suggests that AI techniques for the time being might be treated as a complement to conventional computing techniques rather than as an alternative, or may be used as a quick check on solutions developed by more time-consuming and in-depth analyses.

References

Abu-Farsakh, M.Y., Titi, H.H., 2004. Assessment of direct cone penetration test methods for predicting the ultimate capacity of friction driven piles. Journal of Geotechnical and Geoenvironmental Engineering 130 (9), 935-944.

Abu-Kiefa, M.A., 1998. General regression neural networks for driven piles in cohesionless soils. Journal of Geotechnical & Geoenvironmental Engineering 124 (12), 1177-1185.

Alkroosh, I., Nikraz, H., 2011a. Correlation of pile axial capacity and CPT data using gene expression programming. Geotechnical and Geological Engineering 29, 725-748.

Alkroosh, I., Nikraz, H., 2011b. Simulating pile load-settlement behavior from CPT data using intelligent computing. Central European Journal of Engineering 1 (3), 295-305.

Alkroosh, I., Nikraz, H., 2012. Predicting axial capacity of driven piles in cohesive soils using intelligent computing. Engineering Applications of Artificial Intelligence 25 (3), 618-627.

Alkroosh, I., Nikraz, H., 2014. Predicting pile dynamic capacity via application of an evolutionary algorithm. Soils and Foundations 54 (2), 233-242.

Alsamman, O.M., 1995. The Use of CPT for Calculating Axial Capacity of Drilled Shafts. PhD Thesis. University of Illinois-Champaign, Urbana, Illinois.

American Petroleum Institute, 1984. RP2A: Recommended Practice for Planning, Designing and Constructing Fixed Offshore Platforms (Washington, DC).

Ardalan, H., Eslami, A., Nariman-Zadeh, N., 2009. Piles shaft capacity from CPT and CPTU data by polynomial neural networks and genetic algorithms. Computers and Geotechnics 36 (4), 616-625.

Bowles, J.E., 1997. Foundation Analysis and Design. McGraw-Hill, New York.

Broms, B.B., Lim, P.C., 1988. A simple pile driving formula based on stress-wave measurements. In: Proc., Proceedings of the 3rd International Conference on the Application of Stress-wave Theory to Piles. BiTech Publishers, Vancouver, pp. 591-600.

Brown, M., Harris, C.J., 1995. A perspective and critique of adaptive neurofuzzy systems used for modelling and control applications. International Journal of Neural Systems 6 (2), 197-220.

Burland, J.B., 1973. Shaft friction of piles in clay. Ground Engineering 6 (3), 1-15.

Bustamante, M., Gianeselli, L., 1982. Pile bearing capacity prediction by means of static penetrometer CPT. In: Proc., Proceedings of the 2nd European Symposium on Penetration Testing, pp. 493-500. Amsterdam.

Chan, W.T., Chow, Y.K., Liu, L.F., 1995. Neural network: an alternative to pile driving formulas. Computers and Geotechnics 17, 135-156.

Chen, Y.J., Kulhawy, F.H., 1994. Case History Evaluation of the Behavior of Drilled Shafts under Axial and Lateral Loading. Report No. TR-104601. Electric Power Research Institute, Palo Alto, Calif.

Coyle, H.M., Castello, R.R., 1981. New design correlations for piles in sand. Journal of Geotechnical Engineering 107 (7), 965-986.

Cramer, N.L., 1985. A representation for the adaptive generation of simple sequential programs. In: Proc., Proceedings of the International Conference on Genetic Algorithms and Their Applications. Carnegie-Mellon University, Pittsburgh, PA, pp. 183-187.

Das, B.M., 1995. Principles of Foundation Engineering, third ed. PWS Publishing Company, Boston, MA.

Das, S.K., Basudhar, P.K., 2006. Undrained lateral load capacity of piles in clay using artificial neural network. Computers and Geotechnics 33 (8), 454-459.

Eberhart, R.C., Kennedy, J., 1995. A new optimizer using particle swarm theory. In: Proc., Proceedings of the Sixth International Symposium on Micro-machine and Human Science. IEEE, Nagoya, pp. 39-43.

Elshorbagy, A., Corzo, G., Srinivasulu, S., Solomatine, D.P., 2010. Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology-part 1: concepts and methodology. Hydrology and Earth System Science 14, 1931-1941.

Eslami, A., Fellenius, B.H., 1997. Pile capacity by direct CPT and CPTu methods applied to 102 case histories. Canadian Geotechnical Journal 34 (6), 886-904.

Fausett, L.V., 1994. Fundamentals Neural Networks: Architecture, Algorithms, and Applications. Prentice-Hall, Englewood Cliffs, New Jersey.

Flood, I., 2008. Towards the next generation of artificial neural networks for civil engineering. Advanced Engineering Informatics 22 (1), 4-14.

Gandomi, A.H., Alavi, A.H., 2012. A new multi-gene genetic programming approach to non-linear system modeling. Part II: geotechnical and earthquake engineering problems. Neural Computing Applications 21, 189-201.

Gardner, M.W., Dorling, S.R., 1998. Artificial neural networks (the multilayer perceptron) e a review of applications in the atmospheric sciences. Atmospheric Environment 32 (14/15), 2627-2636.

Giustolisi, O., Savic, D.A., 2006. A symbolic data-driven technique based on evolutionary polynomial regression. Journal of Hydroinformatics 8 (3), 207-222.

Goh, A.T., Kulhawy, F.H., Chua, C.G., 2005. Bayesian neural network analysis of undrained side resistance of drilled shafts. Journal of Geotechnical and Geoenvironmental Engineering 131 (1), 84-93.

Goh, A.T.C., 1994. Nonlinear modelling in geotechnical engineering using neural networks. Australian Civil Engineering Transactions CE36 (4), 293-297.

Goh, A.T.C., 1995a. Back-propagation neural networks for modeling complex systems. Artificial Intelligence in Engineering 9, 143-151.

Goh, A.T.C., 1995b. Empirical design in geotechnics using neural networks. Geotechnique 45 (4), 709-714.

Goh, A.T.C., 1996. Pile driving records reanalyzed using neural networks. Journal of Geotechnical Engineering 122 (6), 492-495.

Goldberg, D.E., 1989. Genetic Algorithms in Search Optimization and Machine Learning. Addison e Wesley, Mass.

Hiley, A., 1922. The efficiency of the hammer blow, and its effects with reference to piling. Engineering 2, 673.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan.

Ismail, A., Jeng, D.-S., 2011. Modelling load-settlement behaviour of piles using high-order neural network (HON-PILE model). Engineering Applications of Artificial Intelligence 24 (5), 813-821.

Ismail, A., Jeng, D.-S., Zhang, L.L., 2013. An optimised product-unit neural network with a novel PSO-BP hybrid training algorithm: applications to load deformation analysis of axially loaded piles. Engineering Applications of Artificial Intelligence 26 (10), 2305-2314.

Janbu, N., 1953. Une analyse energetique du battage des pieux a l'aide de parametres sans dimension. In: Proc., Norwegian Geotechnical Institute, pp. 63-64. Oslo, Norway.

Koza, J.R., 1992. Genetic Programming: on the Programming of Computers by Natural Selection. MIT Press, Cambridge (MA).

Lee, I.M., Lee, J.H., 1996. Prediction of pile bearing capacity using artificial neural networks. Computers and Geotechnics 18 (3), 189-200.

Maier, H.R., Dandy, G.C., 2000. Applications of artificial neural networks to forecasting of surface water quality variables: issues, applications and challenges. In: Govindaraju, R.S., Rao, A.R. (Eds.), Artificial Neural Networks in Hydrology. Kluwer, Dordrecht, The Netherlands, pp. 287-309.

McCulloch, W.S., Pitts, W., 1943. A logical calculus of ideas imminent in nervous activity. Bulletin and Mathematical Biophysics 5, 115-133.

Meyerhof, G.G., 1976. Bearing capacity and settlement of pile foundations. Journal of Geotechnical Engineering 102 (3), 196-228.

Nawari, N.O., Liang, R., Nusairat, J., 1999. Artificial intelligence techniques for the design and analysis of deep foundations. Electronic Journal of Geotechnical Engineering 4. http://geotech.civeng.okstate.edu/ejge/ppr9909/index.html.

Nejad, F.P., Jaksa, M.B., Kakhi, M., McCabe, B.A., 2009. Prediction of pile settlement using artificial neural networks based on standard penetration test data. Computers and Geotechnics 36 (7), 1125-1133.

Pal, M., Deswal, S., 2010. Modelling pile capacity using Gaussian process regression. Computers and Geotechnics 37 (7-8), 942-947.

Park, H.I., Cho, C.W., 2010. Neural network model for predicting the resistance of driven piles. Marine Georesources and Geotechnology 28 (4), 324-344.

Poulos, H.G., Davis, E.H., 1980. Pile Foundation Analysis and Design. Wiley, New York.

Randolph, M.F., 1985. Capacity of Piles Driven into Dense Sand. Rep. Soils TR 171. Engineering Department of Cambridge University, Cambridge.

Randolph, M.F., Wroth, C.P., 1978. Analysis of deformation of vertically loaded piles. Journal of Geotechnical Engineering 104 (12), 1465-1488.

Rausche, F., Moses, F., Goble, G.G., 1972. Soil resistance predictions from pile dynamics. Journal of Soil Mechanics and Foundation Engineering Division 98, 917-937.

Reese, L.C., Isenhower, W.M., Wang, S.T., 2006. Analysis and Design of Shallow and Deep Foundations. John Wiley & Sons, New York.

Rezania, M., Faramarzi, A., Javadi, A., 2011. An evolutionary based approach for assessment of earthquake-induced soil liquefaction and lateral displacement. Engineering Applications of Artificial Intelligence 24 (1), 142-153.

de Ruiter, J., Beringen, F.L., 1979. Pile foundation for large North Sea structures. Marine Geotechnology 3 (3), 267-314.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representation by error propagation. In: Rumelhart, D.E., McClelland, J.L. (Eds.), Parallel Distributed Processing. MIT Press, Cambridge, pp. 318-362.

Savic, D.A., Giutolisi, O., Berardi, L., Shepherd, W., Djordjevic, S., Saul, A., 2006. Modelling sewer failure by evolutionary computing. Proceedings of the Institution of Engineers, Water Management 159 (WM2), 111-118.

Schmertmann, J.H., 1978. Guidelines for Cone Penetration Test, Performance and Design. FHWA-TS-78e209. U. S. Department of Transportation, Washington, D. C., p. 145

Semple, R.M., Rigden, W.J., 1986. Shaft capacity of driven pipe piles in clay. Ground Engineering 19 (1), 11-17.

Shahin, M.A., 2010. Intelligent computing for modelling axial capacity of pile foundations. Canadian Geotechnical Journal 47 (2), 230-243.

Shahin, M.A., 2013. Artificial intelligence in geotechnical engineering: applications, modeling aspects, and future directions. In: Yang, X., Gandomi, A.H., Talatahari, S., Alavi, A.H. (Eds.), Metaheuristics in Water, Geotechnical and Transport Engineering. Elsevier Inc., London, pp. 169-204.

Shahin, M.A., 2014a. Load-settlement modeling of axially loaded drilled shafts using CPT-based recurrent neural networks. International Journal of Geomechanics 14 (6). http://dx.doi.org/10.1061/(ASCE)GM.1943-5622.0000370.

Shahin, M.A., 2014b. Load-settlement modeling of axially loaded steel driven piles using CPT-based recurrent neural networks. Soils and Foundations 54 (3), 515-522.

Shahin, M.A., 2014c. Use of evolutionary computing for modelling some complex problems in geotechnical engineering. Geomechanics and Geoengineering: An International Journal. http://dx.doi.org/10.1080/17486025.2014.921333.

Shahin, M.A., Jaksa, M.B., 2005. Neural network prediction of pullout capacity of marquee ground anchors. Computers and Geotechnics 32 (3), 153-163.

Shahin, M.A., Jaksa, M.B., 2006. Pullout capacity of small ground anchors by direct cone penetration test methods and neural networks. Canadian Geotechnical Journal 43 (6), 626-637.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2001. Artificial neural network applications in geotechnical engineering. Australian Geomechanics 36 (1), 49-62.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2009. Recent advances and future challenges for artificial neural systems in geotechnical engineering applications. Journal of Advances in Artificial Neural Systems 2009. Article ID: 308239. http://dx.doi.org/10.1155/2009/308239.

Solomatine, D.P., Ostfeld, A., 2008. Data-driven modelling: some past experience and new approaches. Journal of Hydroinformatics 10 (1), 3-22.

Tarawneh, B., 2013. Pipe pile setup: database and prediction model using artificial neural network. Soils and Foundations 53 (4), 607-615.

Tarawneh, B., Imam, R., 2014. Regression versus artificial neural networks: predicting pile setup from empirical data. KSCE Journal of Civil Engineering 18 (4), 1018-1027.

Teh, C.I., Wong, K.S., Goh, A.T.C., Jaritngam, S., 1997. Prediction of pile capacity using neural networks. Journal of Computing in Civil Engineering 11 (2), 129-138.

Tian, X., Wei, W., Xiaoni, W., 2010. Artificial neural network model for time-dependent vertical bearing capacity of preformed concrete pile. Advanced Materials Research 29-32 (Part 1), 226-230.

Vesic, A.S., 1977. Design of Pile Foundations. National Cooperative Highway Research, Synthetic of Practice No. 42. Transportation Research Board, Washington DC.

Wellington, A.M., 1892. The iron wharf at Fort Monroe, Va. ASCE Transactions 27, 129-137.

Zurada, J.M., 1992. Introduction to Artificial Neural Systems. West Publishing Company, St. Paul.

Geoscience Frontiers, 7 (2016) 33-44, https://www.sciencedirect.com/science/article/pii/S1674987114001327).

Artificial Neural Network Model for Prediction of Liquefaction Potential in Soil Deposits

F. Farrokhzad, A. J. Choobbasti, A. Barari

ABSTRACT

With the increase in population, the evaluation of liquefaction is becoming more important for land use planning and development. In soil deposits under undrained condition, earthquakes induce cyclic shear stresses, may lead to soil liquefaction. Artificial neural network (ANN) is one of the, artificial intelligence (AI) approaches that can be classified as machine learning. Simplified methods have been practiced by researchers to assess nonlinear liquefaction potential of soil. In order to address the collective knowledge built up in conventional liquefaction engineering, an alternative general regression neural network model is proposed in this paper.

To meet this objective, a total of 30 boreholes are introduced into the model. The data includes the results of field test from (Babol, Mazandaran, Iran).

The results produced by the proposed Artificial Neural Network model compared well with the determined liquefaction decision obtained by simplified methods. It provides a viable liquefaction potential assessment tool that assist geotechnical engineers in making an accurate and realistic predictions. Furthermore, this study integrates knowledge learned from field test and seismic parameters to the ongoing development of liquefaction analysis.

The results show that there is liquefaction potential in western part of Babol, and in southern part of Babol no liquefaction potential were seen. In middle part and eastern part low liquefaction potential were predicted by ANNs. This study shows that neural networks are a powerful computational tool which can analyze the complex relationship between soil liquefaction potential and effective parameters in liquefaction.

INTRODUCTION

When saturated sand deposits are subjected to earthquake induced shaking, pore water pressures are built-up leading to liquefaction or loss of soil strength. Major earthquakes that have occurred during past years, such as the 1964 Alaska, 1964 Niigata, 1989 Loma-prieta and the 1995 Hyogoken-Nambu have demonstrated the damaging effects of soil liquefaction. Therefore, it is necessary to obtain a proper understanding of effective parameters such as soil properties and nature of earthquake on severity of soil liquefaction (Seed HB, Idriss IM, Makdisi F, Banerjee N).

Liquefaction is a phenomenon in which the strength and stiffness of a soil is reduced by earthquake shaking or other rapid loading. During the liquefaction, pore water pressure exerts a pressure on the soil particles that influences how tightly the particles themselves are pressed together. Prior to an earthquake, the water pressure is relatively low (Ishihara K, Yasuda S). However, earthquake shaking can cause the water pressure to increase to the point where the soil particles can readily move with respect to each other. Earthquake shaking often triggers this increase in water pressure, but construction related activities such as blasting can also cause an increase in water pressure.

When liquefaction occurs, the strength of the soil decreases and, the ability of a soil deposit to support foundations for buildings and bridges is reduced (Seed HB, Idriss IM).

In the 1960, Gonzalo Castro, a student of Casagrande, performed an important series of undrained, stress-controlled triaxial tests. Castro observed three different types of stress-strain behavior depending upon the soil state. Dense specimens initially contracted but then dilated with increasing effective confining pressure and shear stress. Very loose samples collapsed at a small shear strain level and failed rapidly with large strains. Castro called this behavior liquefaction; it is also commonly referred to as flow liquefaction. Medium dense soils initially showed the same behavior as the loose samples but, after initially exhibiting contractive behavior, the soil transformed and began exhibiting dilative behavior. Castro referred to this type of behavior as limited liquefaction (Whitman RV).

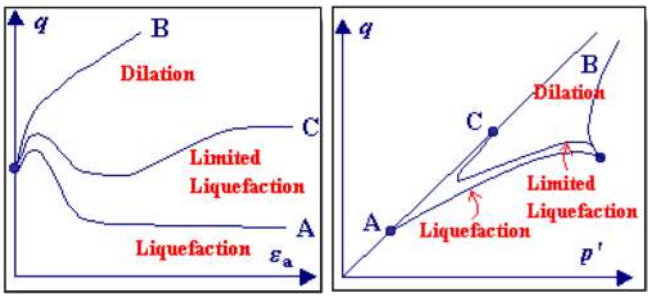


Fig. 1.Static triaxial test stress paths for two specimens of different densities.

Ground response analyses based on the finite element method provide a better assessment of liquefaction of a soil deposit by taking into account the nature of the earthquake and the pore pressure dissipation; they are often costly and time consuming. In addition, constitutive models used in those programs need large number of parameters to determine the pore pressure generation in soil due to earthquake loading. Therefore, simplified methods in assessing soil liquefaction are popular among practicing engineers. These procedures are very useful at the preliminary design stages to assess the liquefaction risk. If the liquefaction risk is high, then a detailed finite element analysis can be carried out to obtain the pore pressure distribution and ground displacement along the depth of the soil deposit, which is necessary in subsequent design of deep foundations. In more details improving the reliability of liquefaction risk, may lead to cost reduction and helps to operation planning (NCEER).

An artificial neural network is a mathematical model or computational model based on Biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

Artificial neural networks mimic human brains to learn the relationships between certain inputs and outputs from experience. They are considered as information processing systems that have the abilities to learn, recall and generalize from training data. An ANN consists of several layers of highly interconnected computational units called neurons. Figure 2 shows the general structure of a three layer feedforward ANN. The neural network contains one input layer, one or two hidden layers, and one output layer The number of nodes in the input layer equals the number of parameters in the process. The output layer represents the quality responses of the product (Agrawal, G., Weeraratne, S., and Khilnani, K). The hidden layer represents the interactions between the input and output layers. Normally the number of nodes in the hidden layer is set to be half of the total number of input nodes and output nodes. If the relationships between the operation parameters and quality responses are difficult to identify, two hidden layers may be used. Such neural networks are capable of capturing complex nonlinear relationships inherent in a process (Hornik K).

The ANN uses a set of examples in a training database as input, a learning algorithm to adjust the weights and an ac-

tivation function to derive an output. If the connection weight between the neurons is changed, the relationship of the network's output to its input will be altered. The process of adjusting the connection weights by repeatedly exposing the network to known input-output data is called training. The error back-propagation learning method is the most popular and successful training technique. A trained ANN can take inputs and produce outputs very quickly, which is an advantage in doing optimization in the proposed approach (Agrawal, G., Chameau, J. A., and Bourdeau, P. L).

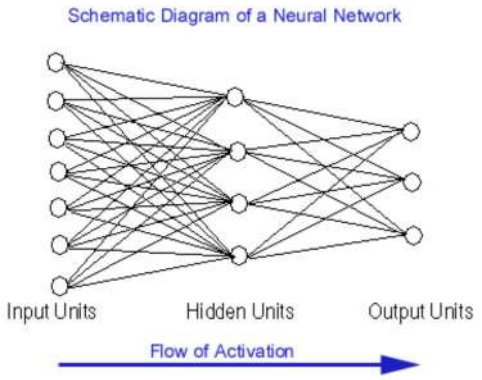


Fig. 2. A three-layer feed-forward neural network structure.

ANNs have been proved to be an universal estimator, hence they are promising techniques in solving pattern recognition and classification, optimization and function approximation problems. Recently, ANNs are used to model complex manufacturing processes and to identify the optimal process setting. In this research, the ANN is used to establish the nonlinear multivariate relationships between liquefaction potential and parameters, which can be used to predict the liquefaction potential in soil.

Recently, extensive studies have been done on application of ANN to Geotechnical engineering problems. Chan et al. (1995) developed a neural network as an alternative to pile driving formulae. The network was trained with the same input parameters listed in the simplified Hiley formula (Broms and Lim 1988), including the elastic compression of the pile and soil, the pile set and the driving energy delivered to the pile (Abu-Kiefa, M. A).

Lee (1996) utilized neural networks to predict the ultimate bearing capacity of piles. The problem was simulated using data obtained from model pile load tests using a calibration chamber and results of insitu pile load tests. Teh et al. (1997) proposed a neural network for estimating the static pile capacity determined from dynamic stress-wave data for precast reinforced concrete piles with a square section.

Sivakugan et al. (1998) explored the possibility of using neural networks to predict the settlement of shallow foundations on granular soils. A neural network was trained with five inputs representing the net applied pressure, average blow count from the standard penetration test, width of foundation, shape of foundation and depth of foundation. The output was the settlement of the foundation (Riedmiller, M. and Braun, H).

Most recently, Shahin et al. (2000) carried out similar work for predicting the settlement of shallow foundations on cohesionless soils. In this work, 272 data records were used for modelling. The input variables considered to have the most significant impact on settlement prediction were the footing width, the footing length, the applied pressure of the footing and the soil compressibility (). The results of the ANN were compared with three of the most commonly used traditional methods. These methods were Meyerhof (1965), Schultze and Sherif (1973) and Schmertmann et al. (1978). The results of the study confirmed those found by Sivakugan et al. (1998), in the sense that ANNs were able to predict the settlement well and outperform the traditional methods (Cal, Y).

Ellis et al. (1995) developed an ANN model for sands based on grain size distribution and stress history. Sidarta and Ghaboussi (1998) employed an ANN model within a finite element analysis to extract the geometerial constitutive behaviour from non-uniform material tests. Penumadu and Jean-Lou (1997) used neural networks for representing the behaviour of sand and clay soils. Ghaboussi and Sidarta (1998) used neural networks to model both the drained and undrained behaviour of sandy soil subjected to triaxial compression-type testing. Penumadu and Zhao (1999) also used ANNs to model the stress-strain and volume change behaviour of sand and gravel under drained triaxial compression test conditions. Zhu et al. (1998a; 1998b) used neural networks for modelling the shearing behaviour of a fine-grained residual soil, dune sand and Hawaiian volcanic soil (Malvić, T., Velić, J. And Cvetković).

It is known, that the engineering properties of soil varied from point to point and uncertain behaviour due to the complex and partially predictable physical processes associated with the forming of these deposits. This is in contrast to most other civil engineering materials, such as steel, concrete and timber, which exhibit far greater homogeneity and isotropy. In order to cope with the complexity of geotechnical behaviour, and the spatial variability of soil deposits, traditional forms of engineering design models are justifiably simplified. It is also known, that assessing liquefaction potential of soil plays an important role in geotechnical evaluation for construction of major structures (Cvetković).

Several methods for liquefaction assessment have been developed. One method of analyses (Seed and Idriss) proposes using the estimated shear stress level and cycle number likely to be developed in the field, due to a design earthquake. Comparison of these stresses with those causing liquefactionof soil samples obtained from laboratory tests helps identifying the liquefiable zones of a deposit. Another method (Seed et al.) considers field observations of performance of sites during previous earthquakes. By combining the data on earthquake characteristics and insitu properties of soil deposits, an empirical relationship is established.

The purpose of this research is to investigate the effect of the soil and seismic parameters, with an artificial intelligence computational tool, and its success in assessing liquefaction potential (National Research Council).

Data collection in explored soils is important for assessing of liquefaction as well as estimation of strata thickness, soil type, groundwater table etc. It is also time consuming and often expensive process, which includes many field and laboratory experiments. Therefore reliable prediction of liquefaction asks for carefully planning of sampling, testing and exploration methods. Data had been collected from the boreholes (maximum depth: 30 m) over a 6 square kilometres area of Babol municipal region. Artificial neural networks are trained with 60% and validated with 10% of borehole data for prediction of liquefaction. The whole system is eventually tested for efficiency, using 30% of borehole data left for test of the network, distributed randomly over the study area. Based on the obtained results and considering that the test data were not presented to the network in the training and validation process, it can be stated that the trained neural networks are capable of predicting variations in the liquefaction potential of soil with an acceptable level of confidence (Malvić, T. And Prskalo, S).

Successful prediction of liquefaction in soil deposit using the existing data leads to improve the reliability of data which will

be used for construction in future. Such approach is presented in the following text that generally comprises presentation of the study area, then description and selection of the neural model, its training, improving, and developing of final model used for prediction of liquefaction by specific ANN (Agrawal, G., Weeraratne, S., and Khilnani, K).

MATERIALS AND METHODS

Babol, a city of Mazandaran province in the northern part of Iran, is considered as the study area. As shown in Figure 3 the city is located approximately 20 kilometres south of Caspian sea on the west bank of the river Babolrood and receives abundant annual rainfall. Babolrood has 2 groups of river terraces, namely H1 and H2. H1 is referred to as river terraces with down surface level of height one to 2.5 (m) and width of 0 to150 m. It is as boundary of active (yearly) flood plain in parts of river and it is as alternative flood plain in many sections. It consists of fine-grained and unconsolidated alluvial sediments. H2 is referred to as river terraces; with high surface level of 4-6 (m). Vegetation on surface of terrace is compact. It consists of materials of Aeolian deposits (i.e. loess). Most major earthquakes occur around the boundaries of the tectonic plates such as those that exist in north of Iran.

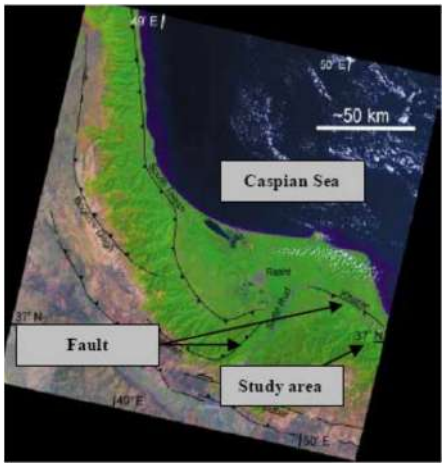


Fig. 3. Map of study area (top) and the zone of the Babolrood river (bottom).

Very often in geotechnical engineering, it is possible to encounter some types of problems that are very complex and can not be completely understood. Mathematical models that attempt to solve such problems can not included entire physics of process and necessarily need to simplify the model or incorporating some assumptions. Mathematical models also assumed the knowing of model structure in advance, which does not need to be optimal. Consequently, many mathematical models fail to simulate the complex behavior of most geotechnical engineering problems. In contrast, ANNs are based mostly on the input data structure, assuming that such structure and interaction among data can describe the prediction model. In this case, there is no need to neither simplify the problem nor incorporate any assumptions (expect user selection of data that are in some meaningful connection). Moreover, one obtained neural models can always be again trained with more extensive and newer dataset from the same area with goal to reach better results.

The data used in presented research, includes borehole logs (data collected from digging boreholes) bored in the study area (Figure 4) and is collected by different institutions for different research purposes. The database includes more than 40 borehole logs in an area of more than 6 square kilometres from Babol zone.

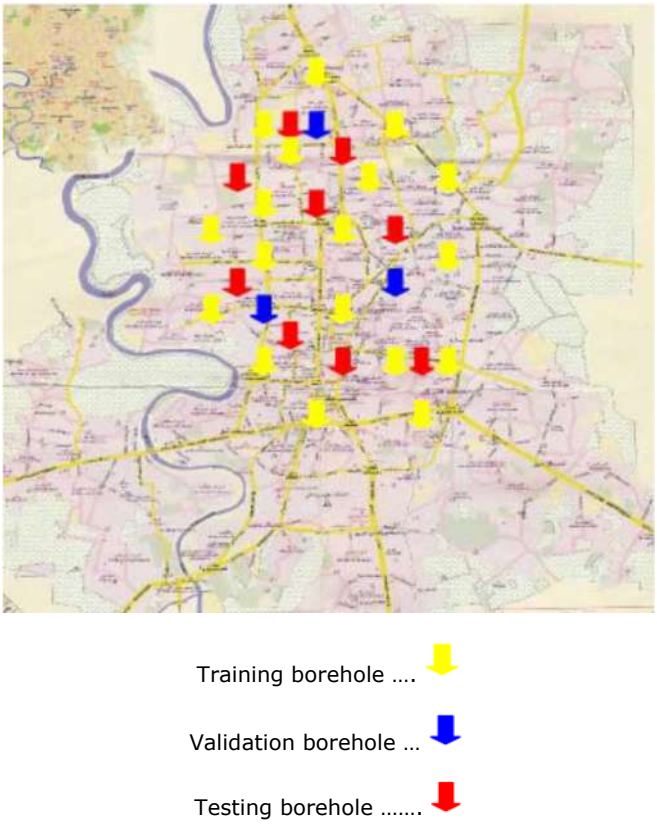


Fig. 4. The 6 zones in Babol area.

From the total of 40 raw borehole data, only 30 logs with a depth range of 10 to 30 meters were acceptable for using in ANN. The regular tests were performed on the samples.

The available data set is divided into three sets, namely training, validation, and test sets, based on random selection. This way we can examine the validity of the model in a more comprehensive manner (Choobbasti AJ, Farrokhzad F, Barari A). In ANN forecasting models, 60% of the records are selected as training, 30% are taken for test for final evaluation, and the remaining 10% used for validation or monitoring the performance of the model during the training phase (Table 1).

Table 1. Performance of different sets of data used in ANN.

	Training set	Validation set	Testing set
Number of boreholes	18	3	9
Number of data (I/O data pairs)	1500	250	750

In problems dealing with different variables and with different ranges and dimensions, the application of several networks may be a good choice. Neural networks are efficient tools when used as pattern classifiers, it is important to properly select the input variables for training (learning) process of ANNs, as the way how to determine relationships between input and output variables. A set of known input and output values is named as input-output pair. All such pairs are usually divided into three sets. The first and second are described as training and validation sets which are used to determine the connection weights or weighting coefficients (like in interpolation methods), usually marked as $\underline{w}_{ij}^{1}$. Also the training and validation sets are used during the training process and the test set is used for obtaining the estimates. All ANN models was trained using the automated regularization algorithm to improve generalization. The validation set served as a constraint on training, in order to minimize over fitting.

The usefulness of the neural network approach for populating the similarity model is presented In this case study. The inputs to the network were data on a set of soil formative environmental factors; the output from the network was a set of similarity values to a set of prescribed soil classes divided by grain size, thickness of each layer and groundwater table. A set of 2500 samplings are performed in study area from 30 boreholes. Data are collected using geotechnical investigation. Each sample is carefully checked, because to ensure the accurate prediction of an ANN model we need to build a reliable training, validating and testing sets.

In this analysis, regarding the available data and their quality, a neural network program written in back propagation algorithm, is used. Eight soil and seismic parameters are selected as input in different models, and these parameters are divided into data groups. Each data group is introduced to the network individually, and performance of the network on the assessment of liquefaction potential is investigated. The network predictions are compared with the conventional liquefaction determination method proposed by Seed *et al.*

Back propagation is selected as the training algorithm of neural network (Table 2). It is the best known training algorithm for multilayer perceptrons neural networks, and still one of the most useful and later improved in some advanced forms like RProp. Back propagation algorithm means that network training includes determination of the difference between true and wanted network response, i.e. means calculation of error that is backed in the neural network for obtaining optimal training. It has lower memory requirements than most algorithms, and usually reaches an acceptable estimation error quite quickly (in relative low number of iterations or epochs).

Table 2. Results of research in order to Learning / training algorithm selection.

Supervised Learning/ training algorithms	Back propagation	Conjugate Gradient Descent	Levenberg-Marquardt	Quick Propagation	Delta -bar- Delta
RMSE (%)	6.3 ✓(min.error)	12.1	8.7	10	9.2

The ANN model for this study was developed, trained, validated and tested within STATISTICA computational environment utilizing the neural network toolbox. And the accuracy of the ANN model was evaluated using RMSE between measured and predicted values and pressed as:

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n}(z_s - z_0)^2}{n}} \quad (1)$$

$$^{1}\text{-}\ y_i^k = f(y_i^{'}k) = f(\sum_{j=1}^{n_{k-1}} w_{ij}^k y_i^{k-1})$$

Where z_s is observed value, z_0 is predicted value, n is number of samples. The RMSE of the different neurons in hidden layer is plotted in Figure 5. The ANN architecture for prediction of soil classification and layers thickness in the study area was a feed forward, supervised, multilayer perceptron (MLP) network with one hidden layer and an output layer. The best fitting training data set was obtained with six neurons in the hidden layer for prediction of liquefaction.

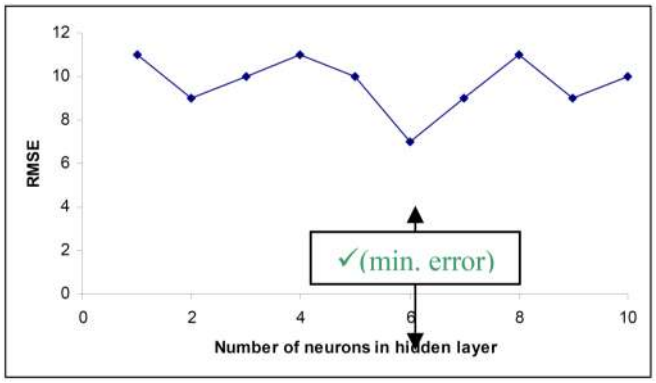


Fig. 5. The RMSE of the different neurons in hidden layer for prediction of soil liquefaction potential.

In the selection of learning / training algorithm number of neurons in different layers (input, hidden, output), number of epochs, learning rate and the momentum have been applied instant.

In each epoch, the entire training set is fed through the network, and used to adjust the network weights. Numbers of epochs are specified at the start, but also alternative stopping criterion may also be specified, and if over-trained network occurs the best network discovered during training can be retrieved. In this analysis, the number of epochs varied between 100 and 400.

A batch mode feed-forward multilayer perceptron (MLP) with back-propagation learning rules was used to create the desired ANN model using STATISTICA software. Also, an adaptive learning rate was employed to keep the learning step size as large as possible while the training is stable. According to a universal approximation theorem, demonstrated concurrently by several researchers for traditional MLP, a single hidden layer network is sufficient to uniformly approximate any continuous and nonlinear function. The model architecture was built with one hidden layer, a learning rate of 0.1 updated with a coefficient of 1.1 after each epoch and a momentum term of 0.9 updated with a coefficient of 0.95 after each epoch. The input vector is fully connected to the hidden neurons by a tan-sigmoid transfer function and the neurons of hidden layer are fully connected to the output layer via a linear function. Experimental studies were started with one hidden neurons to reach the optimum number of hidden neurons and desired precision. Input vector contains soil initial

parameters and output (the target vector) is liquefaction potential. In order to obtain a more efficient training process, the input and target were standardized to have zero mean and unity standard deviation. Cross-validation or employing another set of data for more testing can be used to increase the generality of the models for future predictions. In this study, 10% of borehole data were used as validation set. In fact, several ANN models using element tests data were constituted for generating the models. Among them, the model with better performance (greater coefficient of determination and smaller RMSE) for validation data set was selected. In other words, the ANN models were developed with the best performance concurrently for training, testing and validation data sets. Three different ANN models were developed using different combinations of input parameters in Table 3.

It can be seen from Table 3 that, except for model #1, performances of the models are generally improved when input parameters are increased.

Table . 3. Different combinations of input parameters.

Model #	1	2	3
Input	M, $\frac{a}{g}$, $\sigma, \sigma', R_d, D_r$	M, $\frac{\tau}{\sigma'}$, V_s, R_d, σ	$V_s, \frac{\tau}{\sigma'}$, R_d, σ
RMSE	13%	17%	16%

RESULTS AND DISCUSSION

In the previous section, the learning or training dataset is used to determine the weights. Then a second validation set is used to monitor the performance of the model during the training phase and to minimize over fitting and finally the test sets to evaluate the trained neural network. It is evident from test data sets that the experimental ANN can be applied successfully to predict liquefaction potential.

The samples are divided in to 3 groups (training, validation and testing). In Figure 6 samples of testing group are correlated in terms of sample number and the accuracy (comparison between prediction and real data) of each sample is shown. In these figures, terms of the ratio of actual data per predicted value (in Y-axis) versus Case number (in X-axis) for different soil samples are presented. It is clear that if the predicted and the true values were the same, such point lie on line y=1. Scattering pattern indicates on differences. It is clear that the average correlation of the ANN model and true data in all case is over 90%. So it can be concluded, that the prediction of liquefaction potential agrees with calculated value collected from boreholes.

CONCLUSION

In this research, the data used to train the model were taken from area of 6 km^2 of Babol region in the northern Iran. The dataset encompasses 2500 sampling points (samples) from 30 boreholes. The average accuracy between the ANN prediction and real data in all cases is over 90%. The liquefaction potential of a soil mass during an earthquake is dependent on both seismic and soil parameters. The impact of these soil and seismic variables on the liquefaction potential of soil is investigated through computational and knowledge based tools called neural networks. A back-propagation neural network model is utilized. The back propagation learning algorithm is a developing computational technique that assists in the evaluation of experimental and field data. The artificial neural network is trained using actual field soil records. The performance of the network models is investigated by changing the soil and seismic variables including earthquake magnitude, initial confining pressure, seismic coefficient, relative density, shear modulus, friction angle, shear wave velocity and electrical characteristics of the soil. The most efficient

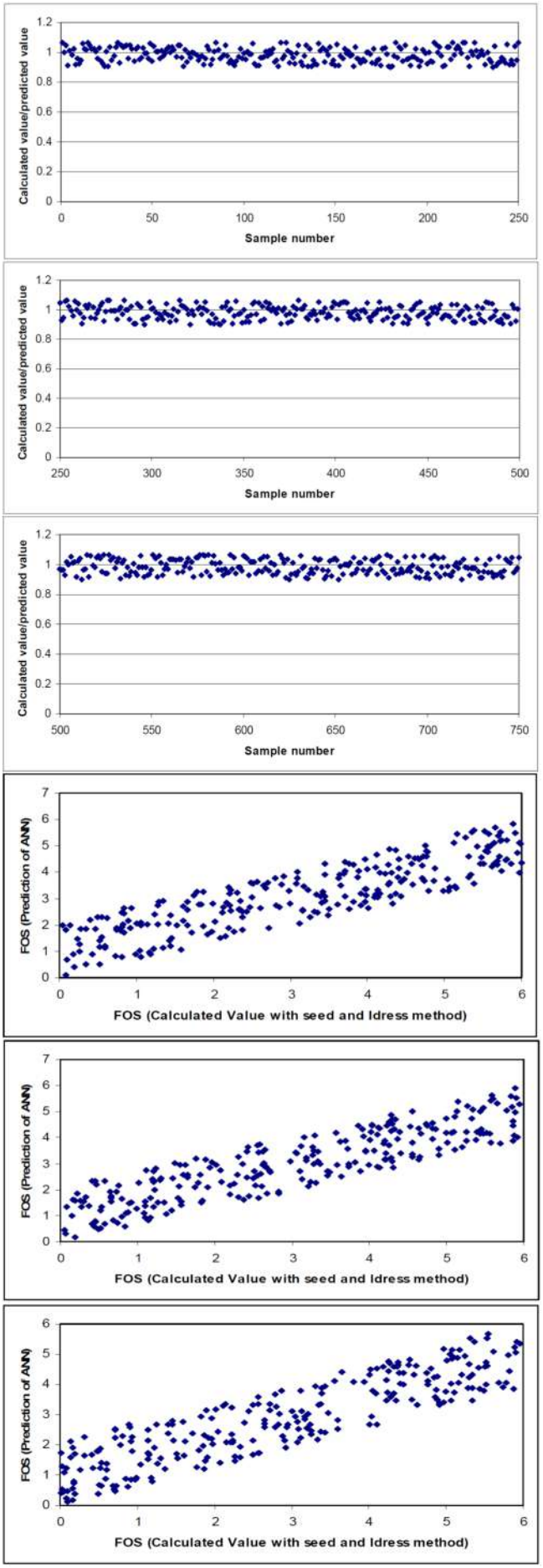


Fig. 6. Errors involved ANN for prediction of liquefaction potential.

and global model for assessing liquefaction potential and the most significant input parameters affecting liquefaction are summarized. A forecast study is performed for the city of Babol, Iran.

Based on the obtained results, it can be stated that the trained neural networks are capable of predicting liquefaction potential with an acceptable level of confidence. It is believed that, the prediction of liquefaction potential is a complex area of research requiring detailed investigation also with other methods, fieldwork and laboratory experiments. Further work on presented topic would be very useful to modify the procedure for better adapting artificial neural network with concept of prediction of liquefaction potential.

REFERENCES

Seed HB, Idriss IM. (1971) "Simplified procedure for evaluating soil liquefaction potential". J Soil Mech Found Div., ASCE; 97(SM8):1249–74.

Hornik K. (1991). "Approximation capability of multilayer feedforward networks as universal approximators". Neural Networks; 4: 251–7.

Whitman RV. (1971). "Resistance of soil to liquefaction and settlement". Soils Found 1971; 11(4):59–68.

Cvetković, M. (2007): "Petroleum Geology Use of Neural Networks of the Example of Reservoir in Kloštar Field." Unpublished Diploma thesis (in Croatian), Faculty of Mining, Geology and Petroleum Engineering, University of Zagreb, date of presentation 15.6.2007. (mentor Prof. Dr. Josipa Velić, comentor: Dr. Tomislav Malvić), 49 p. – generally about prediction of sandstone/marls and saturation with oil/water in Neogene sandstone.

Seed HB, Idriss IM, Makdisi F, Banerjee N. (1975). "Representation of irregular stress time histories by equivalent uniform stress series in liquefaction analyses". Report No. UCB/EERC-75/29, Earthquake Engineering Research Centre, U.C. Berkeley.

Ishihara K, Yasuda S. (1975). "Sand liquefaction in hollow cylinder torsion under irregular excitation". Soils and Found 1975; 15(1):45–59.

Choobbasti AJ, Farrokhzad F, Barari A. (2009) "Prediction of slop stability using artificial neural network". Arab J Geosci; 10.1007/s12517-009-0035-3.

National Research Council. (1985). "Liquefaction of soils during earthquakes". National research Council Report CETSEE-001, National Academy Press, Washington, DC.

NCEER. (1997). In: Proceedings of the NCEER workshop on evaluation of liquefaction resistance of soils. National Center for Earthquake Engineering Research. Technical Report No. NCEER970022, State University of New York at Buffalo, New York, December 31.

Cal, Y. (1995). "Soil classification by neural-network." Advances in Engineering Software, 22(2), 9597.

Abu-Kiefa, M. A. (1998), "General regression neural networks for driven piles in cohesionless soils." J. Geotech. & Geoenv. Engrg., ASCE, 124(12), 1177-1185.

Agrawal, G., Chameau, J. A., and Bourdeau, P. L. (1997). "Assessing the liquefaction susceptibility at a site based on information from penetration testing." In: Artificial neural networks for civil engineers: fundamentals and applications, N. Kartam, I. Flood, and J. H. Garrett, eds., New York, 185-214.

Malvić, T., Velić, J. And Cvetković, M. (2008): "Review of neural network analyses performed in Croatian part of Pannonian basin (petroleum geology data)". XII. Congress of Hungarian geomathematics and the first congress of Croatian and Hungarian geomathematics, Morahalom, Hungary, 29-31.05.2008.

Malvić, T. and Prskalo, S. (2007): "Some benefits of the neural approach in porosity prediction (Case study from Beničanci field)". Nafta, 58, 9, 455-467. – generally about porosity mapping in Neogene breccia.

Chan, W. T., Chow, y. K., and Liu, L. F. (1995). "Neural network: An alternative to pile driving formulas." J. Computers and Geotechnics, 17, 135-156.

Agrawal, G., Weeraratne, S., and Khilnani, K. (1994). "Estimating clay liner and cover permeability using computational neural networks." Proc., First Congress on Computing in Civil Engrg., Washington, June 20-22.

Riedmiller, M. and Braun, H. (1993): "A direct adaptive method for faster backpropagation learning": The Rprop algorithm. Proc. Of the IEEE Int. Conf. On Neural Networks, San Francisco, p. 586-591.

(International Conferences on Recent Advances in Geotechnical Earthquake Engineering and Soil Dynamics. 4. https://scholarsmine.mst.edu/icrageesd/05icrageesd/session04/4)

ΕΚΤΕΛΕΣΤΙΚΗ ΕΠΙΤΡΟΠΗ ΕΕΕΕΓΜ (2015 – 2019)

Πρόεδρος	:	Γεώργιος ΓΚΑΖΕΤΑΣ, Δρ. Πολιτικός Μηχανικός, Καθηγητής Ε.Μ.Π. gazetas@central.ntua.gr, gazetas50@gmail.com
Α' Αντιπρόεδρος	:	Παναγιώτης ΒΕΤΤΑΣ, Πολιτικός Μηχανικός, ΟΜΙΛΟΣ ΤΕΧΝΙΚΩΝ ΜΕΛΕΤΩΝ Α.Ε. otmate@otenet.gr
Β' Αντιπρόεδρος	:	Μιχάλης ΠΑΧΑΚΗΣ, Πολιτικός Μηχανικός mpax46@otenet.gr
Γενικός Γραμματέας	:	Μιχάλης ΜΠΑΡΔΑΝΗΣ, Πολιτικός Μηχανικός, ΕΔΑΦΟΣ ΣΥΜΒΟΥΛΟΙ ΜΗΧΑΝΙΚΟΙ Α.Ε. mbardanis@edafos.gr, lab@edafos.gr
Ταμίας	:	Γιώργος ΝΤΟΥΛΗΣ, Πολιτικός Μηχανικός, ΕΔΑΦΟΜΗΧΑΝΙΚΗ Α.Ε.- ΓΕΩΤΕΧΝΙΚΕΣ ΜΕΛΕΤΕΣ Α.Ε. gdoulis@edafomichaniki.gr
Έφορος	:	Γιώργος ΜΠΕΛΟΚΑΣ, Δρ. Πολιτικός Μηχανικός, Επίκουρος Καθηγητής ΤΕΙ Αθήνας gbelokas@teiath.gr, gbelokas@gmail.com
Μέλη	:	Ανδρέας ΑΝΑΓΝΩΣΤΟΠΟΥΛΟΣ, Δρ. Πολιτικός Μηχανικός, Ομότιμος Καθηγητής ΕΜΠ aanagn@central.ntua.gr
		Βάλια ΞΕΝΑΚΗ, Δρ. Πολιτικός Μηχανικός, ΕΔΑΦΟΜΗΧΑΝΙΚΗ Α.Ε. vxenaki@edafomichaniki.gr
		Μαρίνα ΠΑΝΤΑΖΙΔΟΥ, Δρ. Πολιτικός Μηχανικός, Αναπληρώτρια Καθηγήτρια Ε.Μ.Π. mpanta@central.ntua.gr
Αναπληρωματικό Μέλος	:	Κωνσταντίνος ΙΩΑΝΝΙΔΗΣ, Πολιτικός Μηχανικός, ΕΔΑΦΟΜΗΧΑΝΙΚΗ Α.Ε. kioannidis@edafomichaniki.gr
Εκδότης	:	Χρήστος ΤΣΑΤΣΑΝΙΦΟΣ, Δρ. Πολιτικός Μηχανικός, ΠΑΝΓΑΙΑ ΣΥΜΒΟΥΛΟΙ ΜΗΧΑΝΙΚΟΙ Ε.Π.Ε. editor@hssmge.gr, ctsatsanifos@pangaea.gr

ΕΕΕΕΓΜ
Τομέας Γεωτεχνικής
ΣΧΟΛΗ ΠΟΛΙΤΙΚΩΝ ΜΗΧΑΝΙΚΩΝ
ΕΘΝΙΚΟΥ ΜΕΤΣΟΒΙΟΥ ΠΟΛΥΤΕΧΝΕΙΟΥ
Πολυτεχνειούπολη Ζωγράφου
15780 ΖΩΓΡΑΦΟΥ

Τηλ. 210.7723434
Τοτ. 210.7723428
Ηλ-Δι. secretariat@hssmge.gr ,
geotech@central.ntua.gr
Ιστοσελίδα www.hssmge.org (υπό κατασκευή)

«ΤΑ ΝΕΑ ΤΗΣ ΕΕΕΕΓΜ» Εκδότης: Χρήστος Τσατσανίφος, τηλ. 210.6929484, τοτ. 210.6928137, ηλ-δι. ctsatsanifos@pangaea.gr, editor@hssmge.gr, info@pangaea.gr

«ΤΑ ΝΕΑ ΤΗΣ ΕΕΕΕΓΜ» «αναρτώνται» και στην ιστοσελίδα www.hssmge.gr