Geomechanics and Engineering, *Vol. 31, No. 3 (2022) 265-279*
https://doi.org/10.12989/gae.2022.31.3.265

Assessment of wall convergence for tunnels using machine learning techniques

Arsalan Mahmoodzadeh[1a], Hamid Reza Nejati*[1], Mokhtar Mohammadi[2b], Hawkar Hashim Ibrahim[3c], Adil Hussein Mohammed[4d] and Shima Rashidi[5e]

[1]*Rock Mechanics Division, School of Engineering, Tarbiat Modares University, Tehran, Iran*
[2]*Department of Information Technology, College of Engineering and Computer Science, Lebanese French University, Kurdistan Region, Iraq*
[3]*Department of Civil Engineering, College of Engineering, Salahaddin University-Erbil, 44002 Erbil, Kurdistan Region, Iraq*
[4]*Department of Communication and Computer Engineering, Faculty of Engineering, Cihan University-Erbil, Kurdistan Region, Iraq*
[5]*Department of Computer Science, College of Science and Technology, University of Human Development, Sulaymaniyah, Kurdistan Region, Iraq*

(*Received November 20, 2021, Revised September 18, 2022, Accepted October 12, 2022*)

Abstract. Tunnel convergence prediction is essential for the safe construction and design of tunnels. This study proposes five machine learning models of deep neural network (DNN), K-nearest neighbors (KNN), Gaussian process regression (GPR), support vector regression (SVR), and decision trees (DT) to predict the convergence phenomenon during or shortly after the excavation of tunnels. In this respect, a database including 650 datasets (440 for training, 110 for validation, and 100 for test) was gathered from the previously constructed tunnels. In the database, 12 effective parameters on the tunnel convergence and a target of tunnel wall convergence were considered. Both 5-fold and hold-out cross validation methods were used to analyze the predicted outcomes in the ML models. Finally, the DNN method was proposed as the most robust model. Also, to assess each parameter's contribution to the prediction problem, the backward selection method was used. The results showed that the highest and lowest impact parameters for tunnel convergence are tunnel depth and tunnel width, respectively.

Keywords: feature selection; machine learning applications; tunnel construction; tunnel convergence

1. Introduction

As the population grows, the demand for tunnels increases more and more, both in travel and transportation (Mahmoodzadeh and Zare 2016). For this reason, the attention of engineers to the construction of tunnels to minimize risks and uncertainties is increasing worldwide (Mahmoodzadeh *et al.* 2021a). Uncertainties in tunnels include the tunnel route's geological conditions, the designs made, the maintenance systems considered, and the construction expenses and time. Improper designs of a tunnel by engineers based on pre-construction studies can increase time consumption, cost overruns, and even the death of working personnel (Mahmoodzadeh *et al.* 2021b).

One of the most critical problems in tunneling projects that requires special attention is the phenomenon of tunnel convergence. Accurate prediction of this phenomenon can be very important in establishing a proper and timely maintenance system for tunnels and reducing risks. To evaluate the convergence of tunnels, the researchers have recommended many computational, mathematical, numeric and machine learning (ML) approaches.

Analytical methods (Fahimifar *et al.* 2010, Nomikos *et al.* 2011, Sterpi and Gioda 2009) and numerical methods (Debernardi and Barla 2009, Nadimi *et al.* 2011, Sharifzadeh *et al.* 2013) aim to model and solve the rheological and mechanical equations, which are monitoring the convergence of tunnels during times. However, despite their considerable benefits, analytical and numerical use has trouble implementing in the field because they depend on precise parameters. Finding the proper parameters are also difficult because of using small-scale in the laboratory compared to the field (Sterpi and Gioda 2009) and/or the time and expense of the tests needed to classify materials' rheological behavior (Guan *et al.* 2009).

The empirical models are an alternative method (Kontogianni *et al.* 2006, Sakurai 1978) that could replicate patterns of convergence seen in actual tunnels using simplistic models that need only a few parameters. Based on these models, a software tool could be constructed to characterize and predict tunnel convergence and evaluate the convergence of real tunnels through experimental data and curve fitting methods (Asadollahpour *et al.* 2014, Vu *et al.* 2013). It is challenging to use such models for

*Corresponding author, Ph.D.
E-mail: h.nejati@modares.ac.ir
[a]M. Sc.
E-mail: m.arsalan@modares.ac.ir
[b]Ph.D.
E-mail: mukhtar@lfu.edu.krd
[c]M. Sc.
E-mail: hawkar.ibrahim@su.edu.krd
[d]M. Sc.
E-mail: adil.mohammed@cihanuniversity.edu.iq
[e]M. Sc.
E-mail: shima.rashid@uhd.edu.iq

ISSN: 2005-307X (Print), 2092-6219 (Online)

measuring tunnel convergence because these methodological convergence models are not yet usable, and choosing suitable parameters for utilizing these models is not simple. It needs more compressive study to decide it (González el Á lamo and Jiménez 2011).

ML methods are fast and inexpensive, and researchers' potential ability has been proved by researchers in the engineering field (Madenci and Gülcü 2020, Trinh and Jun 2021, Mahmoodzadeh *et al.* 2022, Kostinakis and Morfidis 2020, Aktas and Ozerdem 2020, Liu *et al.* 2020). These methods also have shown good accuracy in the prediction of tunnel convergence. For this purpose, some ML techniques have been utilized by the researchers, including adaptive neuro-fuzzy inference system (ANFIS) (Adoko and Wu 2012), artificial neural networks (ANN) (Adoko *et al.* 2013, Mahdevari *et al.* 2013, Mahdevari and Torabi 2012, Rafiai and Moosavi 2012), support vector machines (SVM) (Mahdevari *et al.* 2013, Mahdevari *et al.* 2012), multivariate adaptive regression spline (MARS) (Adoko *et al.*, 2013), gene expression programming (GEP) (Hajihassani *et al.* 2019), multilayer perceptron (MLP)-ANN, radial basis function (RBF)-ANN, multivariate linear regression (MLR), multivariate non-linear regression (MNR), support vector regression (SVR), Gaussian process regression (GPR), regression trees (RT), besides ensembles of trees (ET) (Torabi-Kaveh and Sarshari 2020).

Since there are many algorithms in ML, it is crucial to examine the ability of the other algorithms for tunnel convergence prediction. For this propose, in this article, five ML approaches, including deep neural networks (DNN), K-nearest neighbors (KNN), GPR, SVR, and decision tree (DT) are applied. A database consisting of 650 datasets has been used in the models, and 5-fold and hold-out cross-validation (CV) techniques are applied to categorize the database into three groups of training, validation, and test. Finally, to specify the best prediction model of tunnel convergence, all the tunnel convergence results predicted by the ML models are compared with each other and with the measured (actual) values. One of the most important differences between this work and the previous works is the tunnel type parameter's effect on the models' inputs. All the previous works have been done separately on tunnels excavated in the methods of mechanized and traditional. But, in the database applied in this study, all the tunnels excavated using tunnel boring machine (TBM), earth pressure balance (EPB)-TBM, new Austrian tunneling method (NATM), and drilling and blasting (DB) are considered.

Another contribution of this work is determining the most influential parameters considered in this study on tunnel convergence using the backward selection method.

The remaining of the study is arranged as follows: The literature review of the tunnel convergence prediction is explained in the next section. Section 3 introduces some descriptions of the ML methods applied in this study in computer science. The database of this study is described in section 4. All of the ML model prediction results are analyzed in section 5. A comparison is made between the ML predictions in section 6 to specify the most accurate model. Discussion and conclusions are presented respectively in sections 7 and 8.

2. Literature review

In their research down on the tunnels excavated using the new Austrian tunneling method (NATM), Adoko and Wu (2012) applied an ANFIS method to predict the convergence phenomenon. They applied a database including 1057 data sets in their research, and the ANFIS results showed an excellent prediction accuracy. They suggested their model to estimate the convergence phenomenon of the future NATM tunnels. Mahdevari *et al.* (2012) applied the ANN approach for the prediction of tunnel convergence. Among 12 effective input parameters considered in their study, they concluded that four parameters of cohesion (C), angle of internal friction (φ), elasticity modulus (E), and uniaxial compressive strength (UCS), and one parameter of the uniaxial tensile strength (σ_t) are the most effective and least effective parameters on the tunnel convergence, respectively. Mahdevari *et al.* (2013) applied two ML methods of ANN and SVM to predict the convergence of a tunnel excavated by tunnel boring machine (TBM). Both of their models produced good prediction results, but the SVM performs better than the ANN. Rafiai and Moosavi (2012) an ANN-based solution method to predict convergence of lined circular tunnels. They discussed the limitations associated with the solution. Adoko *et al.* (2013) applied two methods of MARS and ANN to foresee the convergence of a tunnel with a high-speed railway. Finally, the study revealed high precise tunnel convergence model predictability with MARS, which showed slightly less accuracy. Mahdevari *et al.* (2013) presented a dynamic prediction model based on SVM algorithm to predict tunnel convergence during construction. Their model could be updated during the construction tunnel, and new data from the excavated sections were utilized as the training data sets in the model. Each time, the updating procedure resulted in more accurate predictions. Feng *et al.* (2019) proposed a Bayesian approach to improve the convergence phenomenon's prediction results by updating the model with new data monitored during the tunnel construction. Hajihassani *et al.* (2019) suggested a model based on GEP approach, and they showed the GEP model's potential ability in tunnel convergence prediction. Torabi-Kaveh and Sarshari (2020) used eight ML methods of MLP-ANN, RBF-ANN, MLR, MNR, SVR, GPR, RT, and ET in this research down on the tunnel convergence rate prediction. Among the eight models used, the MLP-ANN was the most accurate, and the RT and GPR were the least accurate models in the prediction of tunnel convergence.

Literature reviews on the tunnel convergence prediction using ML approach with the input parameters considered within them are presented in Table 1.

3. ML methods

Regardless of the advantages of different machine learning methodologies, the No-Free-Lunch (NFL) theorem declares that no machine learning model can solve all engineering challenges. As a result, academics have

Table 1 Literature review on the tunnel convergence prediction using ML approaches

Reference	Method	Inputs	Output
Adoko and Wu (2012)	ANFIS	GEC, T, SRM, γ, H, D	TC
Mahdevari and Torabi (2012)	ANN	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	TC
Mahdevari *et al.* (2012)	ANN	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	TC
	SVM	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	TC
Rafiai and Moosavi (2012)	ANN	C, E, ψ, σ_1, ν_1, t_l, ν, σ_3, φ, E_l, R	TC
Adoko *et al.* (2013)	MARS	γ, SRM, T, φ, C, X, E, H	TC
	ANN	γ, SRM, T, φ, C, X, E, H	TC
Mahdevari *et al.* (2013)	SVM	C, W_c, ρ, E, φ, K_s	TC
Hajihassani *et al*. (2019)	GEP	UW, φ, H, C, ν, E	TC
Torabi-Kaveh and Sarshari (2020)	MLP-ANN	NB, σ_c, RMR, C, H, φ	TC
	RBF-ANN	NB, σ_c, RMR, C, H, φ	TC
	MLR	NB, σ_c, RMR, C, H, φ	TC
	MNR	NB, σ_c, RMR, C, H, φ	TC
	SVR	NB, σ_c, RMR, C, H, φ	TC
	GPR	NB, σ_c, RMR, C, H, φ	TC
	RT	NB, σ_c, RMR, C, H, φ	TC
	ET	NB, σ_c, RMR, C, H, φ	TC

Note SRM: Surrounding rock mass rating index, GEC: Ground engineering conditions rating index, H: Tunnel overburden, γ: Rock density, X: Distance between monitoring station and working face, T: Elapsed time, TC: Tunnel convergence, R^2: Correlation coefficient, RMSE: Root mean square error, VAF: Variance account for, C: Cohesion, φ: Friction angle, E: Elasticity modulus, UCS: Uniaxial compressive strength, GSI: Geological strength index, σ_t: Uniaxial tensile strength, RQD: Rock quality designation, H: Tunnel depth, ν: Poisson's ratio, γ_{dry}: Dry unit weight, γ_{sat}: Saturation unit weight, σ_c: Unconfined compressive strength, R: Tunnel radius, (σ_1, σ_3): In situ stresses, ψ: Dilation angle, t_l: Lining thickness, ν_1: Poisson's ratio of lining, E_l: Elasticity modulus of lining, SRM: Class rating index of the surrounding rock mass, W_c: Water content, ρ: Density, K_s: Modulus of subgrade reaction, GEP: Gene expression programming, GL: Groundwater level, EA: Excavation area, UW: Unit weight, RMR: Rock mass rating, NB: Number of installed rock bolts.

attempted to assess the efficacy of various ML techniques for solving various optimization problems. DNN, KNN, GPR, SVR, and DT are five distinct machine learning (ML) and deep learning (DL) models with distinct features and capabilities that we employ as NFL theory. However, the following are the important aspects of the mentioned models that inspire us to apply them:
models that inspire us to apply them:

- **Regression Analysis**

Regression analysis is a statistical tool for predicting a continuous target variable using one or more independent variables. Regression analysis is frequently used for variables that are naturally occurring rather than variables that have been manipulated through experimentation. Variety forms of regression are available, so how do we select the one to use once we have chosen to use regression analysis?

- **We chose GPR because:**

- In GPR, the model uncertainty can directly be captured. In regression, for instance, instead of providing a single value as the prediction, GPR provide a distribution value. This uncertainty is not explicitly captured in neural network.
- By choosing different kernel functions, While utilizing GPR, we can include prior knowledge and requirements regarding the model's shape. For example, we may choose different priors based on the responses to the following questions. Is the model slick? Is there a lot of it? Should it be able to undergo significant change? Should it be distinguishable? This capability provides researchers with adaptable models that may be fitted to a variety of datasets.

- **We chose SVR because:**

The utilization of kernels, sparse solutions, and Vapnik-Chervonenkis (VC) regulation of the margin and number of support vectors distinguish SVR. Its computational complexity, which is one of the prime advantage of SVR, is independent of the dimensions of the input space. When compared to other regression algorithms, it requires less computing. It also offers excellent prediction precision, good generalization, and is resistant to outliers.

- **We chose DT because**:

DTs are a type of supervision learning method that splits the sample numerous times based on specified sample queries. For prediction problems, these can play a significant role. They are relatively simple to comprehend and very operative. DTs reflect a series of decisions with varying chances of occurring. This technique can be used to identify the most significant factors as well as the relationship between two or more variables. Because we have several variables in our problem that are related to one another, we choose DT as one of the comparing models. In other words, a decision tree has the advantage of forcing the analysis of all possible outcomes and tracing each direction to a conclusion. It performs a thorough analysis of the implications of each branch and defines the required decision nodes.

Key advantages:

- There is no need to preprocess the data.
- Data distribution assumptions are not required.
- It can effectively handle collinearity.
- The prediction in a comprehensible manner can be obtained in DT.

- **We chose KNN because**:

In this paper, KNN were used as a non-parametric approach for prediction. It is one of the simplest ML techniques that has lately been used. It is a slow learning model with local approximation. We employ this model while keeping the following terms in mind.

Key Advantages:

- It is a straightforward machine learning model and easy to use.
- It can tune only a few hyperparameters.

Hyperparameters:

- There are two main hyperparameters in KNN: K value and distance function.
- K value: the number of neighbors to include in the KNN algorithm. The value of k should be adjusted according to the validation error.
- Distance function: the most frequently used similarity function is Euclidean distance. The Manhattan distance, Hamming distance, and Minkowski distance are three options.

Assumptions:

- A thorough knowledge of the input domain is necessary.
- A sample size should feasibly be moderate.
- Prior to training, outliers and collinearity should be tackled.

Comparison with other models:

- The main difference between KNN and other models is that KNN needs a lot of real-time computation.

- **We chose DNN because**:

The substantial advantage of DNN over its predecessor is that it automatically identifies essential features without the need for human intervention. In this case, no knowledge about the reliance of feature is available, therefore, DNN as one of the comparators is utilized the dimensionality of datasets can be reduced and the effects of the automatic dimensionality reduction in the results can be visualized.

3.1 DNN

In the layered neural network of DNN, a deep feed-forward neural network is available. The hidden units between its outputs and inputs are more than one layer. The layers contain input layers, which are followed by mid-layers, hidden layers, and finally, the output layer. Consequently, the input, hidden, and output layers are all connected to the network's neighboring layers. Furthermore, DNN is especially well-suited to analyzing raw input data because it can recognize patterns and learn useful features from it without the need for rigorous feature engineering, hand-crafted guidelines, or data pre-processing. Moreover, with the rise in training data, its efficiency further improves. DNNs have a much wide range of applications, from simple text creation to machine vision tasks, and the early uses of DNN are in automatic translation (Schulz and Behnke 2012).

3.2 KNN

KNN is a simple and well-managed machine learning algorithm that can be applied to classification and regression problems. It is important to apply and comprehend, but it is very difficult to do as the amount of data in use expands considerably. KNN functions through the determination of the ranges between a query and all the examples in the results, choosing the specified number of examples for the query (K), then voting for the most frequent label (in the situation of classifying) or averaging the labels (In the situation of a regression). During the regression and classification process, selecting the correct K for the data is achieved by trying multiple Ks and finding the one that fits better (Altman 1992).

3.3 GPR

GP could be described as a nonparametric machine learning algorithm based on supervised learning, divided into two major groups: regression and classification. Whereas the regression target values are continuous quantity forecasts, the classifying results are discrete labels. However, the GP is defined as a collection of arbitrary variables in which any finite subset of the variables has a joint multivariate Gaussian distribution. The covariance functions and mean functions specify a GP. The process of $f\ (x)$ with the covariance function $k(x, x')$ and the mean function $m(x)$ could be defined as (Rasmussen and Williams 2016, Pedregosa *et al*. 2011)

$$\mu(x) = E\ [f(x)] \tag{1}$$

$$k(x, x') = E[\big(f(x) - \mu(x)\big)\big(f(x') - \mu(x')\big)] \tag{2}$$

Hence, the GP is defined as

$$f(X) \sim GP\big(\mu(X), k(X, X')\big) \tag{3}$$

For the notational simplicity, the $\mu(x)$ function (i.e., mean function) is assumed to be zero. In addition, the $k(x, x')$ function (i.e., covariance function) will be calculated by a function known as kernel function (Rasmussen and Williams 2016, Pedregosa *et al*. 2011).

Training data for real process f is denoted as

$$D = \left\{\left(X^{(i)}, y^{(y)}\right) \mid i = 1,2, \dots, n\right\} \tag{4}$$

where X is an input vector of dimension d, y is the corresponding real-valued noisy observation such that $y = f(X) + \in$. Let D be the collection of d x n input matrix, X, and target vectors, y (Rasmussen and Williams 2016, Pedregosa *et al.* 2011).

$$D = \{X, y\} \tag{5}$$

The challenge now would be to calculate the required values f^* for unknown input X^*. As per the Gaussian process prior, the f and f^* joint distribution is (Rasmussen and Williams 2016, Pedregosa *et al*. 2011)

$$\begin{bmatrix} f \\ f^* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K+\sigma_n^2 I & K^{*^T} \\ K^* & K^{**} \end{bmatrix}\right) \quad (6)$$

where σ_n^2 is the noise variance, I is the Identity matrix, $K^*$is a matrix denoting the kernel function assessed for each member of X^* x X and similarly $K$is defined over X x X and K^{**}over X^* x X^*. Hence, the conditional distribution of f^* given X^*, X and f can be found (Rasmussen and Williams 2016, Pedregosa *et al*. 2011)

$$f^*|X^*, X, f \sim N\left(K^*(K+\sigma_n^2 I)^{-1} f, K^{**} - K^*(K+\sigma_n^2 I)^{-1} K^{*^T}\right) \quad (7)$$

To reduce the mean squared error (MSE), the best estimator of f^* will be the mean of the above normal distribution (Rasmussen and Williams 2016, Pedregosa *et al.* 2011)

$$\hat{f}^* = K^*(K+\sigma_n^2 I)^{-1} f \quad (8)$$

The GP technique could be represented as the linear combination of n kernel functions (Rasmussen and Williams 2016, Pedregosa *et al.* 2011).

$$\hat{f}^* = \sum_{i=1}^{n} \alpha^{(i)} k\left(x^*, x^{(i)}\right) \quad (9)$$

where α is the coefficient vector, and the optimal MSE coefficient vector will be $\alpha = (K+\sigma_n^2 I)^{-1} f$. Thus, the most crucial aspect of the GPR is the covariance (Kernel) function.

3.4 SVR

Vapnik (2000) modified his first version model (ε-support vector regression, SVR) by changing the ε-insensitive loss function. This modification permits the SVR model to use the margin idea in the regression process. Margin in the modified model can be described by the summation of the distances of hyperplane from the closest points of two classes. Minimizing errors between the actual training data and the hyperplane are the main target of the SVR. Kernel function idea has introduced by Vapnik (2013)for non-linear SVR. Readers are directed to Vapnik (2000) understand more about SVR.

3.5 DT

The DT is a classification and regression approach that is based on the technique of non-parametric survived learning. It also includes a set of if-then-else decision rules. The model's finest perdition occurs when the DT digs deeper and deeper to find the best fit with the actual data. The DT has a number of advantages. First, there is no need to make assumptions about the distribution of explanatory variables. Second, substantial relationships between independent factors have no effect on DT outcomes. Third, DT can encompass a wide range of dependent variables, including survived data, category data, and numerical data. Fourth, this strategy includes the powerful factors while excluding the least powerful variables that characterize the dependent variable. The DT can predict both small and large datasets well, despite the fact that it was originally designed to predict only large datasets well (Quinlan 1987).

The DT algorithm is explained as follows (Kamiński *et al.* 2017):

1. First, the calculated, targeted variance is computed.
2. The database is divided into multiple portions based on the various features, and the variance of each sectioned component is subtracted from the variance previous to the division. This is known as variance reduction. Node N can be defined by the variance reduction as

$$I_V(N) = \frac{1}{|S|^2}\sum_{i\in S}\sum_{j\in S}\frac{1}{2}(x_i - x_j)^2 - \left(\frac{1}{|S_t|^2}\sum_{i\in S_t}\sum_{j\in S_t}\frac{1}{2}(x_i - x_j)^2 + \frac{1}{|S_f|^2}\sum_{i\in S_f}\sum_{j\in S_f}\frac{1}{2}(x_i - x_j)^2\right) \quad (10)$$

S is a group of samples that is not separated yet, S_t is a group of separated samples with true result and S_f is a group of separated samples with a false result. Without referring to the mean, each of the summands presented above is indeed variance estimates though written in a form. In each summation term in Eq. (10), variance estimation is required in such a way the mean is not referred to directly.

4. Database

A database consisting of 650 datasets is applied in this study, obtained during the construction of several tunnels in different locations in Iran (see Fig. 1). The datasets are divided into three groups of training (440), validation (110), and test (100). In the database, based on the experience gained from the different types of tunnel construction, available information on the considered tunnels, and based on the previous studies in this field, 12 effective input parameters on the tunnel convergence are considered. The tunnel convergence is considered as the output parameter. The correlations between inputs and output parameters are provided in Fig. 1.

As in Fig. 2, the input parameters are categorized within three groups of geotechnical conditions, tunnel specifications, and tunnel constructions.

The point that needs to mention here is that the convergences used in this paper are measured in the R1-L1 direction, as shown in Fig. 3. In other words, the convergence of tunnel walls is considered. Of course, convergence also can be considered in other directions, as shown in Fig. 3.

5. Results analysis

For GPR, SVR, DT, and LR modeling, MATLAB 2018

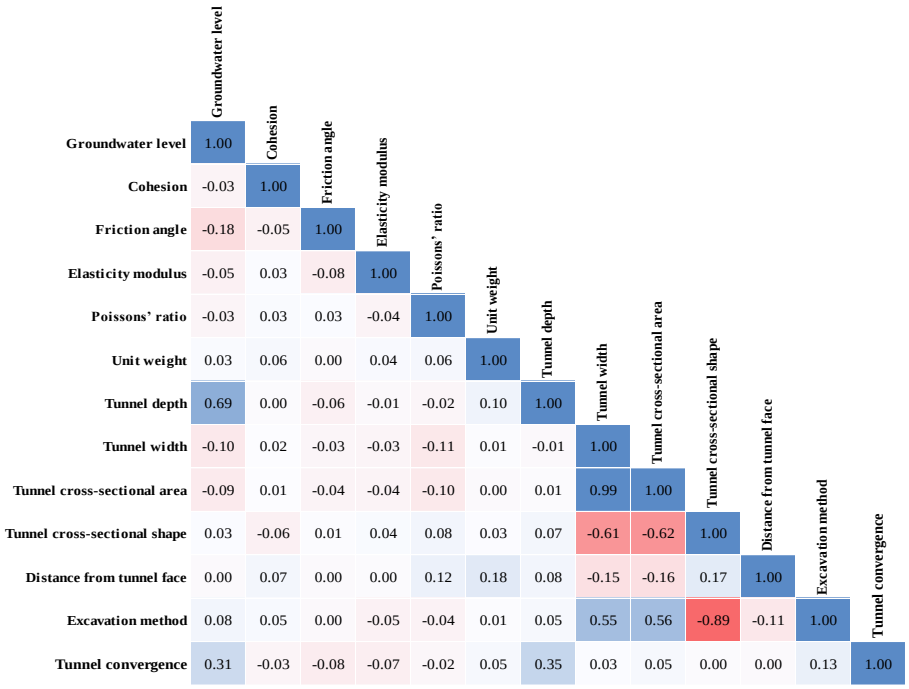

Fig. 1 The ccorrelations between inputs and output parameters

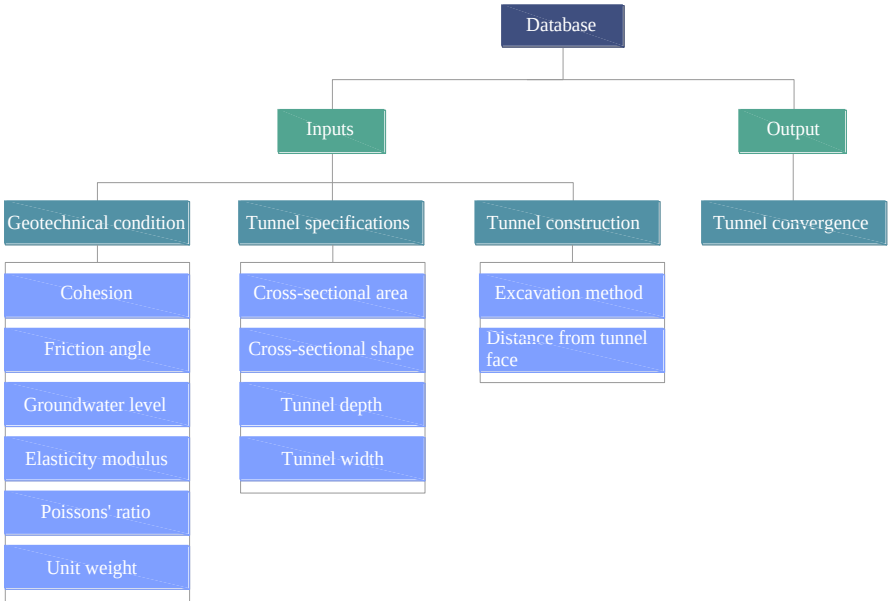

Fig. 2 An overview of the database

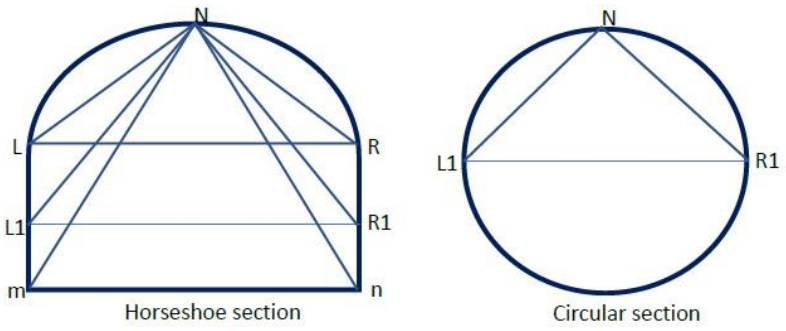

Fig. 3 Directions for measurement of tunnel convergence. The L1-R1 direction is considered in this study

Table 2 The calculation formulas of the statistical indices used in this study

Indices	Formula	Indices	Formula
R^2	$R^2 = \left(\frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)(y_i' - \bar{y}_i')}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2 \sum_{i=1}^{n}(y_i' - \bar{y}_i')^2}}\right)^2$	RMSE	$RMSE = \sqrt{\left(\frac{1}{n}\right)\sum_{i=1}^{n}(y_i - y_i')^2}$
VAF	$VAF = 1 - \left[\frac{var(y_i - y_i')}{var(y_i)}\right] \times 100\%$	MAPE	$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left\|\frac{y_i - y_i'}{y_i}\right\| \times 100\%$
MAE	$MAE = \left(\frac{1}{n}\right)\sum_{i=1}^{n}\|y_i - y_i'\|$		

Note: y_i is the actual value, y_i' is the predicted value, $\bar{y}_i$ and $\bar{y}_i'$ are the means of actual and predicted values, and n is the number of samples

Table 3 The optimized parameters of the DNN model

No. of layers	Model type	No. of Neurons	Activation function	Optimizer	batch_size	Epochs	Kernel initializer
1 input layer 4 hidden layers 1 output layer	Sequential()	Input layer: 12 neurons 1st hidden layer: 160 2nd hidden layer: 80 3rd hidden layer: 40 4th hidden layer: 20 Output layer: 1	"ReLU"	"Nadam"	10	100	"Lecun_uniform"

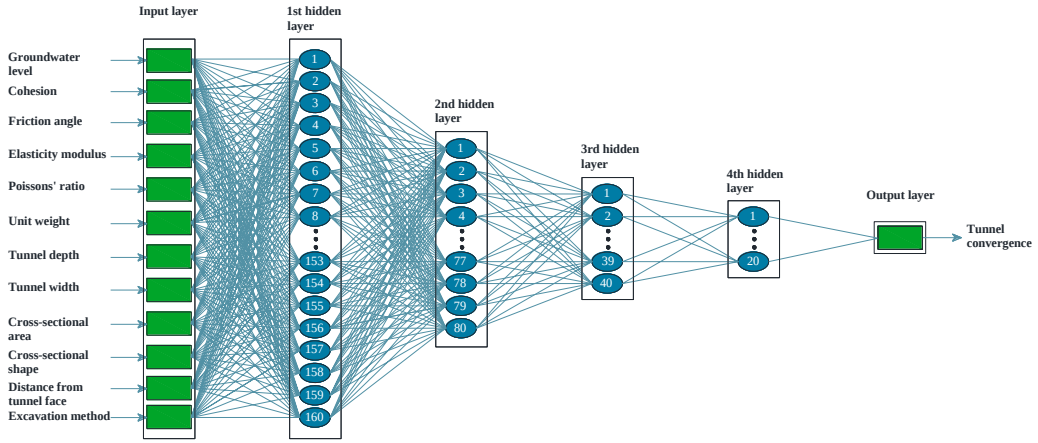

Fig. 4 The optimized DNN network

software is used in this study. The implementation environment for DNN, and KNN algorithms is Anaconda version 3.6, a metapackage containing all Python packages. To assess the prediction results, the K-fold CV is considered in this study (K=5). The sample is subdivided into K subsamples of the same size by random means in the K-fold CV. The validation data for model testing is maintained as one subsample of the K-samples, and the remainder (K−1) is used as training data. The benefit of this approach relative to repetitive random subsamples is that all experiment data are utilized for each training and validation. Additionally, any experiment is utilized for validation only once. The validated model using the K-fold CV approach should also surpass that verified with a validation set because it requires using K-fold CV validation subsets for validating model results. For the K-fold CV in this study, both training and validation datasets (550 datasets) are used in the modeling.

In order to evaluate the predictive accuracy of any model, several statistical indices such as coefficient of determination (R^2), variance account for (VAF), mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE) are taken into account. The formulas for calculating these indices are presented in Table 2.

In the following sub-sections, the tunnel convergence results predicted by the ML models are presented and analyzed in detail.

5.1 DNN

The structure network applied for the DNN model is shown in Fig. 4. In Table 3, an overview of the DNN model applied for this study is presented.

As shown in Fig. 5, the DNN prediction results agree with the measured results. Also, in Fig. 6, the R2 value of 0.9863 for the DNN outcomes vs. the actual mode proves

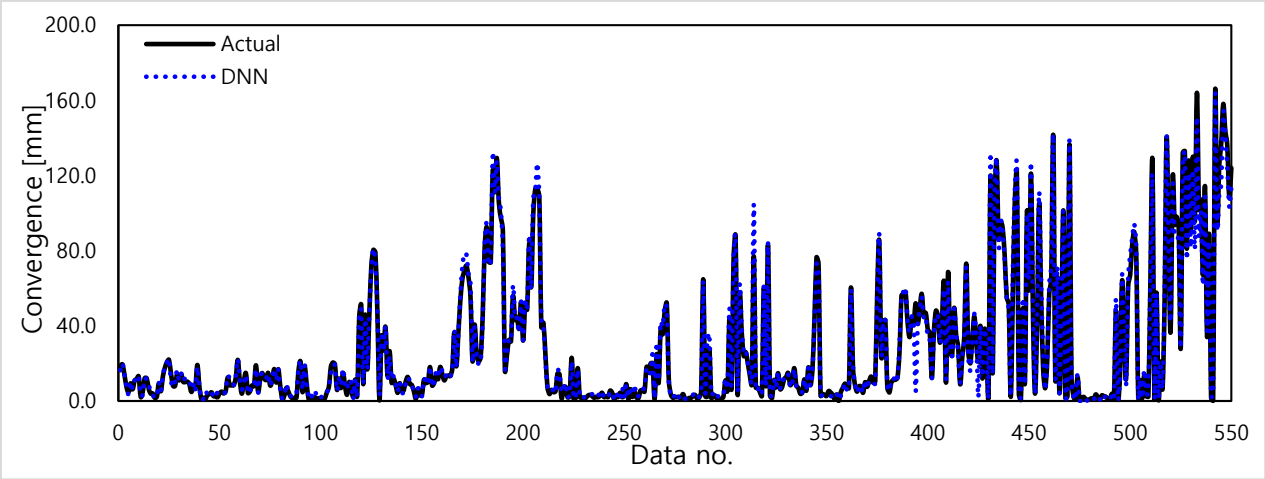

Fig. 5 The DNN model results in comparison with the actual mode

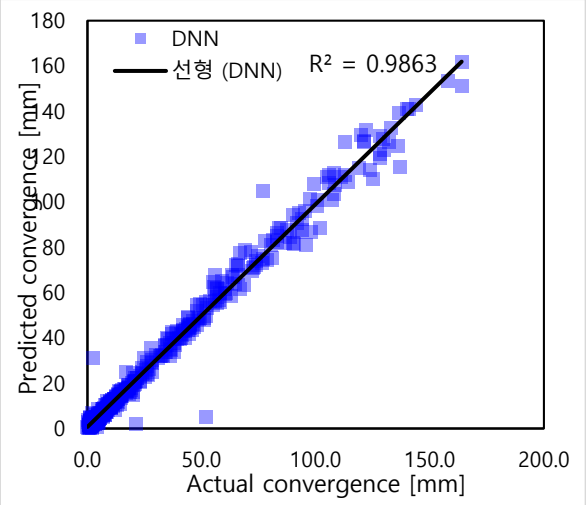

Fig. 6 Forecasting performance of the DNN model

Table 4 Statistical indices results of the DNN model

R^2	MAE	RMSE	MAPE	VAF
0.9863	2.02E+00	4.125061	20.69224	98.95788

Table 5 Statistical indices results of the KNN model

R^2	MAE	RMSE	MAPE	VAF
0.9519	4.88	7.905916	27.46223	96.87909

the DNN model's potential ability in this study in the tunnel convergence prediction. To more support the DNN model applicability in the tunnel convergence prediction, the other statistical indices are evaluated as in Table 4. Looking at Table 4, considering the database utilized in this study, the statistical indices results confirm the potential ability of the DNN model in the tunnel convergence prediction.

5.2 KNN

To tunnel convergence prediction using the KNN approach, several Ks were considered, and the KNN model was finally created based on K=3. The tunnel convergence predictions of the KNN model are shown in Fig. 7, and a comparison is made with the measured convergence values. Through looking at Fig. 7 can see the closeness of the predicted results with the actual model. The R^2 value of 0.9519 obtained for the KNN results, as shown in Fig. 8, also is a sign of good accuracy of the KNN model. The other statistical results evaluated as in Table 5 indicate the KNN model's good ability in the tunnel convergence prediction.

5.3 GPR

In MATLAB 2018, the optimization mode was considered. In this way, the best parameter values or types are considered by the MATLAB program to obtain the best and accurate predictions. The GPR approach has four model forms in the MATLAB software regression learner app: rational quadratic, squared exponential, Matern 5/2, and exponential. The model types that produced the most reliable predictions are known as GPR predictions. In this case, the Matern 5/2 model style was the most accurate. The

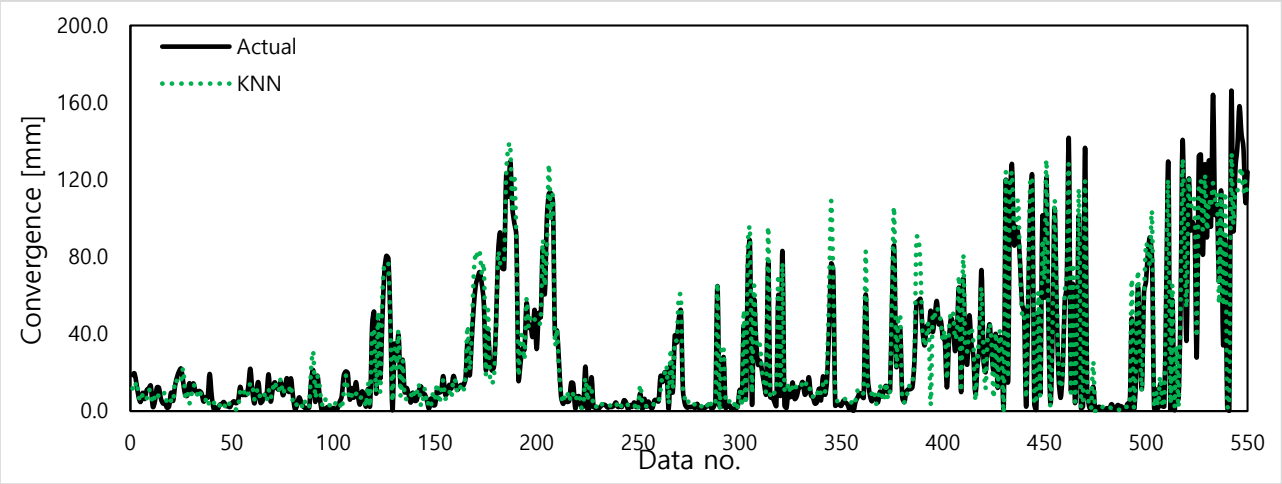

Fig. 7 The KNN model prediction results in comparison with the actual mode

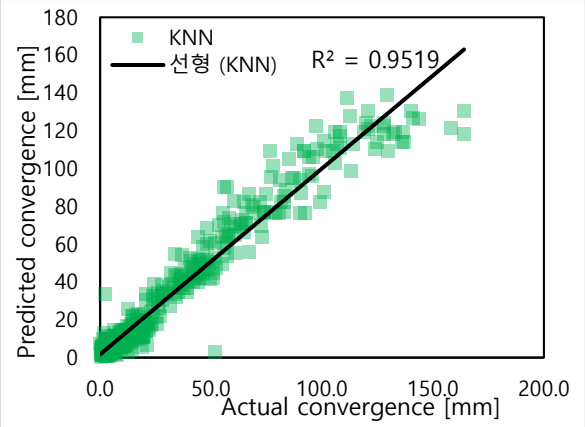

Fig. 8 Forecasting performance of the KNN model

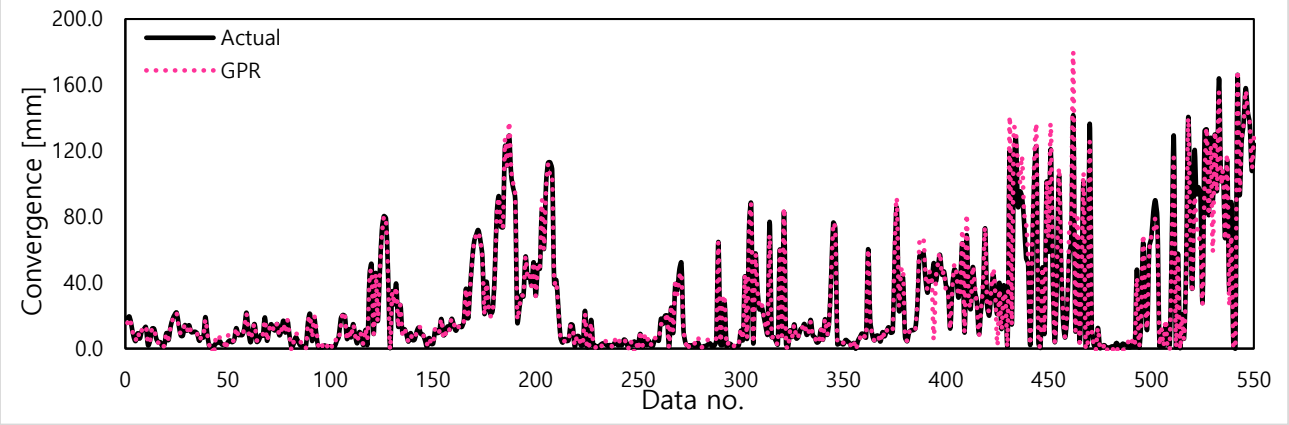

Fig. 9 Prediction results of the GPR model in comparison with the measured results

Table 6 The optimized parameters of the GPR model

Parameter	Value or type
"Sigma"	1.5115
"BasisFunction"	'Constant'
"KernelFunction"	'Matern 5/2'
"Beta"	162.5270
"FitMethod"	Exact GPR

other parameters considered in the GPR model through the optimization mode in the MATLAB software are presented in Table 6.

Fig. 9 displays the GPR model's tunnel convergence performance. From Fig. 9, very little difference can be seen between the actual method and predicted outcomes. The R^2 value of 0.9749 in Fig. 10 shows the good prediction performance of the GPR model. All the other statistical

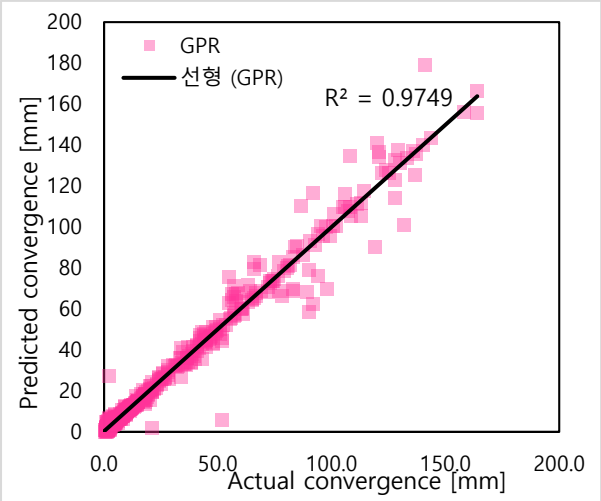

Fig. 10 Forecasting performance of the GPR model

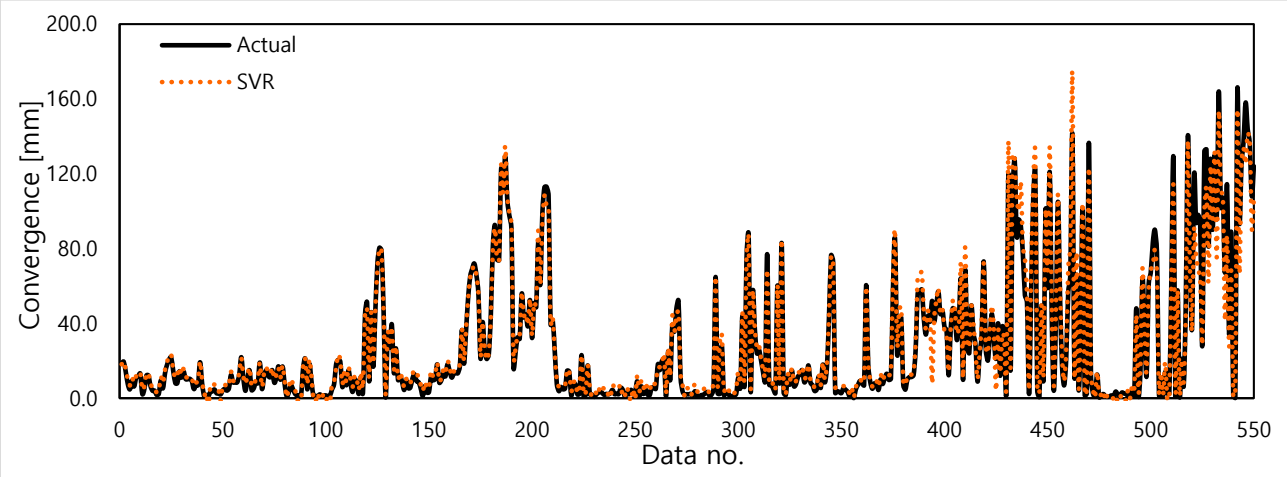

Fig. 11 Tunnel convergence results of the SVR model in comparison with the actual measured values

Table 7 Statistical indices results of the GPR model

R^2	MAE	RMSE	MAPE	VAF
0.9749	2.43	5.641771	29.60609	97.90998

indices in Table 7, proof the applicability of the GPR model in the tunnel convergence prediction considering the database applied in this study.

5.4 SVR

As the GPR model, the SVR model parameters described in Table 8 are considered by the MATLAB software through the optimization mode. Between six different models, which are linear, quadratic, cubic, fine gaussian, medium gaussian, and coarse gaussian that provided in the MATLAB software, the linear model is selected in this study as the best one. The comparison between the actual mode and predicted results of the tunnel convergence by the SVR model are illustrated in Fig. 11.

The comparison shows that the SVR model predicted tunnel convergence with a high degree of accuracy. The R^2 value of 0.9687 in Fig. 12 and the other statistical indices findings in Table 9 illustrate the SVR model's ability to predict tunnel convergence using the database used in this analysis.

Table 8 The optimized parameters of the SVR model

Parameter	Value or type
"Kernel Function"	'Linear'
"Solver"	'SMO'
"Epsilon"	2.7947
‘"Bias"	28.1748

Table 9 Statistical indices results of the SVR model

R^2	MAE	RMSE	MAPE	VAF
0.9687	3.46	6.292437	32.24720	97.77306

5.5 DT

The DT parameters are determined using the MATLAB program in the same way as the GPR and SVR models.

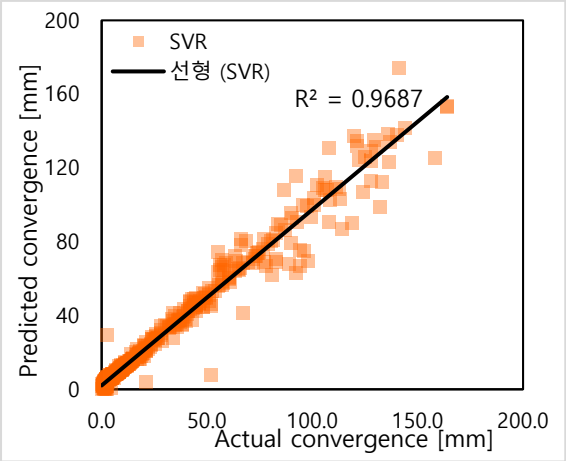

Fig. 12 Forecasting performance of the SVR model

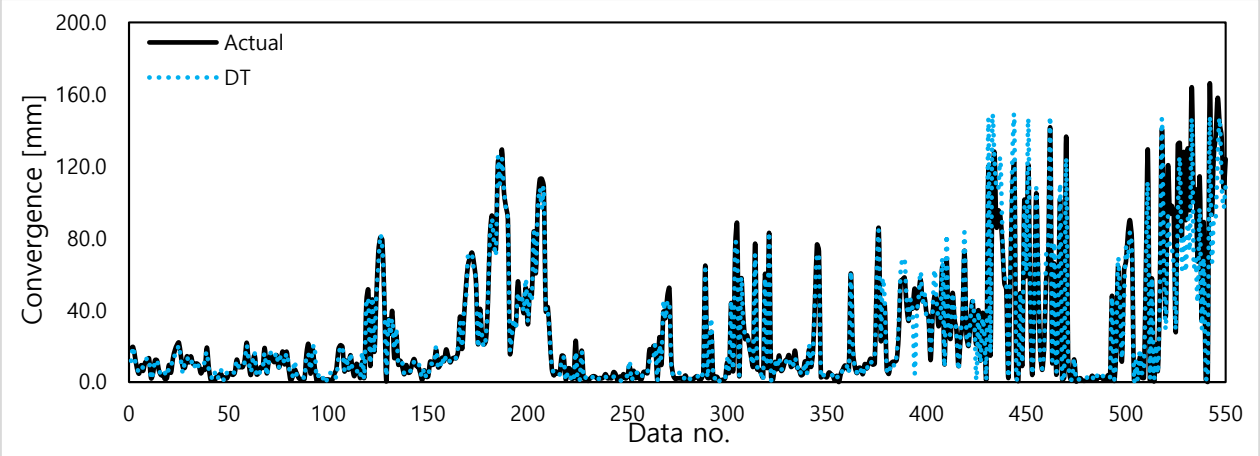

Fig. 13 Tunnel convergence prediction results in compare with the actual mode

Table 11 Statistical assessment indices outcomes of the DT model

R^2	MAE	RMSE	MAPE	VAF
0.9493	3.82	8.004067	26.99743	96.00853

Three model types are included in the MATLAB 2018 package: fine tree, medium tree, and coarse tree. The fine tree is the best and most reliable model for the DT model defined by the MATLAB tool. In addition, Table 10 also lists the other DT model parameters that were chosen by the MATLAB tool through the optimization mode.

The tunnel convergence outcomes predicted by the DT model are presented in Fig. 13, and compared with the measured convergences. The DT model results are in good agreement with the actual mode, as shown in Fig. 13. The R^2 value in Fig. 14 illustrates this agreement. In addition, given the database used in this analysis, the other statistical indices findings in Table 11 support the DT method's potential ability to predict tunnel convergence.

6. Models comparison

In the previous section, all the models' results were analyzed, and the applicability of them in the tunnel convergence prediction, given the database used in this study, was proved. Although all the models show very good accuracy, a comparison of their results to specify the best and most accurate model can be very significant.

Table 12 Comparison of the statistical indices values of the ML models applied in this study

Statistical indices	DNN	KNN	GPR	SVR	DT
R^2	0.9863	0.9519	0.9749	0.9687	0.9493
MAE	2.02	4.88	2.43	3.46	3.82
RMSE	4.12	7.90	5.64	6.29	8.00
MAPE	20.6	27.4	29.6	32.2	26.9
VAF	98.9	96.8	97.9	97.7	96.0

Based on Table 12 which displays the statistical indices' effects, the most accurate belongs to the DNN model. GPR and SVR are the third and fourth accurate models. The KNN and DT models' results are close to each other, and they produce the least accuracy.

Consequently, given the database used in this analysis, the DNN model produced the most accurate convergences compared to the other four ML models considered in this

Table 13 Comparing the outcomes of the ML models applied in this study with the other ML models in the literatu

Reference	Method	Inputs	Train	Test	R^2	RMSE	VAF
Adoko and Wu (2012)	ANFIS	GEC, T, SRM, γ, H, D	845	212	0.98	0.243	90.8
Mahdevari and Torabi (2012)	ANN	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	36	20	0.9358	---	---
Mahdevari *et al.* (2012)	ANN	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	42	18	0.87	---	---
	SVM	E, C, GSI, UCS, ν, σ_t, φ, RQD, σ_c, γ_{dry}, H, γ_{sat}	42	18	0.97	---	---
Rafiai and Moosavi (2012)	ANN	C, E, ψ, σ_1, ν_1, t_l, ν, σ_3, φ, E_l, R	2000	500	0.999	---	---
Adoko *et al.* (2013)	MARS	γ, SRM, T, φ, C, X, E, H	390	96	0.95	0.42	94.26
	ANN	γ, SRM, T, φ, C, X, E, H	390	96	0.97	0.29	95.81
Mahdevari *et al.* (2013)	SVM	C, W_c, ρ, E, φ, K_s	60	15	0.976	---	---
Hajihassani *et al.* (2019)	GEP	UW, φ, H, C, ν, E	79	39	0.986	0.45	---
Torabi-Kaveh and Sarshari (2020)	MLP-ANN	NB, σ_c, RMR, C, H, φ	70%	30%	0.925	0.169	---
	RBF-ANN	NB, σ_c, RMR, C, H, φ	70%	30%	0.81	0.269	---
	MLR	NB, σ_c, RMR, C, H, φ	70%	30%	0.61	0.386	---
	MNR	NB, σ_c, RMR, C, H, φ	70%	30%	0.648	0.366	---
	SVR	NB, σ_c, RMR, C, H, φ	70%	30%	0.66	0.46	---
	GPR	NB, σ_c, RMR, C, H, φ	70%	30%	0.53	0.423	---
	RT	NB, σ_c, RMR, C, H, φ	70%	30%	0.53	0.423	---
	ET	NB, σ_c, RMR, C, H, φ	70%	30%	0.63	0.374	---
This study	DNN	GL, CS, C, CA, φ, EM, H, W, ν, E, X, UW	440	110	0.9863	4.125	98.95
	KNN	GL, CS, C, CA, φ, EM, H, W, ν, E, X, UW	440	110	0.9519	7.905	96.87
	GPR	GL, CS, C, CA, φ, EM, H, W, ν, E, X, UW	440	110	0.9749	5.641	97.90
	SVR	GL, CS, C, CA, φ, EM, H, W, ν, E, X, UW	440	110	0.9687	6.292	97.77
	DT	GL, CS, C, CA, φ, EM, H, W, ν, E, X, UW	440	110	0.9493	8.004	96.00

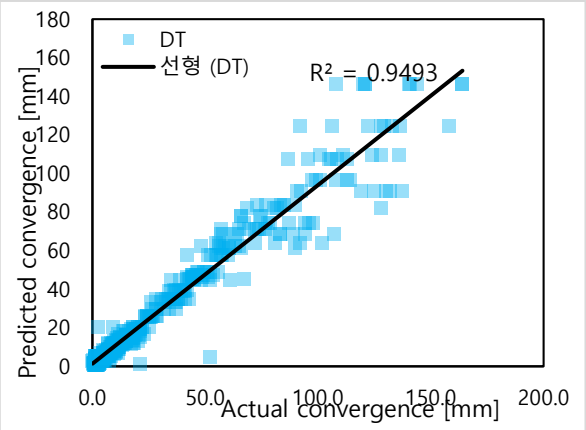

Fig. 14 Forecasting performance of the DT model

study. In any case, the potential of the other models cannot be denied, but they also provided excellent accuracies.

In addition to comparing the ML prediction models used in this paper, comparing their results with the other ML prediction methods used in the literature could be very important. Table 13 provides a comparison with the other previous studies in the field. It can be found that the highest prediction accuracies belong to the ML models used in this study. Of course, it should be noted that the prediction results do not depend only on the type of prediction model, but also the type and number of parameters considered in the database play a significant role. However, the results gained in this study, compared to the former studies, show the prediction models' high accuracy and the correct choice of parameters affecting the tunnel convergence.

7. Discussion

In this work, the proposed DNN methodology is addressed in terms of its generalization. Generalization is a principle utilized to describe the capability of a model to interact and adapt to new information. Hence, after working with data not utilized during training, a model can assimilate new input and properly forecast. The ability to generalize is the foundation for a model's success and practical performance. No generalization can be obtained for a model which was well trained in training data. Although the model can successfully predict the training data, it is considered worthless when new data are provided. A model begins to 'memorize' rather than 'learn' the training data, this is considered as overfitting. Overfitting can be avoided by using feature selection, which minimizes the amount of features and hence the computational cost of the model. In this case, we utilize the stepwise method to select

Table 14 First step of feature selection

	Estimate	SE	tStat	p-value	Significance code
(Intercept)	3.3699	1.7519	1.9236	0.05493	.
Groundwater level	0.35087	0.0021362	164.25	5.39E-49	***
Cohesion	-0.075653	0.0090093	-8.3973	4.09E-16	***
Friction angle	-0.054353	0.0051849	-10.483	1.61E-23	***
Elasticity modulus	-0.059692	0.0037519	-15.91	5.62E-47	***
Poissons' ratio	0.63357	0.74571	0.84961	0.39592	
Unit weight	-0.11865	0.014351	-8.2683	1.07E-15	***
Tunnel depth	0.098191	0.0009	109.1	3.82E-56	***
Tunnel width	-0.73978	0.37735	-1.9604	0.050461	*
Tunnel cross-sectional area	0.14197	0.025405	5.5882	3.65E-08	***
Tunnel cross-sectional shape	0.073467	0.027686	2.6536	0.0081994	**
Distance from tunnel face	-0.071312	0.0056188	-12.692	1.83E-32	***
Excavation method	0.0015469	0.00028255	5.475	6.73E-08	***

Table 15 The second step of feature selection

	Estimate	SE	tStat	p-value	Significance code
(Intercept)	3.6667	1.7162	2.1365	0.033093	*
Groundwater level	0.35089	0.0021355	164.31	6.21E-49	***
Cohesion	-0.075406	0.0090023	-8.3764	4.77E-16	***
Friction angle	-0.054213	0.005181	-10.464	1.89E-23	***
Elasticity modulus	-0.059802	0.0037488	-15.953	3.41E-47	***
Unit weight	-0.11794	0.014322	-8.2346	1.37E-15	***
Tunnel depth	0.098159	0.00089895	109.19	4.84E-56	***
Tunnel width	-0.76497	0.37609	-2.034	0.042439	*
Tunnel cross-sectional area	0.14349	0.025335	5.6635	2.42E-08	***
Tunnel cross-sectional shape	0.072743	0.027666	2.6294	0.0087986	**
Distance from tunnel face	-0.070864	0.0055925	-12.671	2.20E-32	***
Excavation method	0.0015418	0.00028241	5.4595	7.30E-08	***

the most important set of characteristics from the total available features in the supplied dataset.

Stepwise regression has three strategies:

1. **Forward selection** begins with no predictors in the model and iteratively adds the most useful predictors until the improvement is no longer statistically significant.
2. **Backward selection** which begins with all predictors in the model, iteratively removes the least significant predictors, and finishes when all predictors are statistically significant.
3. **Stepwise selection** is a combination of forward and reverse selections. We begin with no forecasters and gradually add the most useful predictors (like forward selection). Remove any variables that no longer improve model fit after adding each new variable (like backward selection).

In this work, we used stepAIC() [MASS package], which uses AIC to select the best model. It has a direction option that can take the following values: I "both" (for stepwise regression, both forward and backward selection), "backward" (for backward selection), and "forward" (for forward selection) (for forward selection). It returns the best possible final model. StepAIC is one of the most extensively used feature selection search methods in R. To arrive at the final collection of features, we strive to minimize the stepAIC value as much as possible. Three stars (or asterisks) represent a very significant p-value in the results shown in the tables below. As a result, a small p-value for the intercept and slope suggests that we may reject the null hypothesis, allowing us to conclude that the predictor and the target variables have a strong association. A p-value of 5% (.05) or less is considered a good cut-off point.

The model was fitted with all predictors and targets in the first stage. Table 14 shows the results of the first step of feature selection. As seen in Table 14, 11 features have a considerable impact on tunnel convergence. In this situation, we do not remove Poissons' ratio feature and go to the next stage. Based on Table 14, the most effective parameter on slope stability with respect to the p-values is tunnel depth. Also, the Poissons' ratio parameter has the least impact on tunnel convergence.

The model now has 11 predictors. Therefore, as in Table 15, the smallest range of features are {GL, C, φ, E, UW, H, W, CA, CS, X, EM}, that the highest and lowest impact parameters for tunnel convergence are tunnel depth (H) and tunnel width (W), respectively.

Table 16 Comparison of the statistical index values of the ML models for the testing datasets

Statistical indices	DNN	KNN	GPR	SVR	DT
R^2	0.9851	0.9507	0.9716	0.9626	0.9440
MAE	2.26E+00	5.10 E+00	2.88E+00	3.72E+00	4.28E+00
RMSE	4.461241	8.549255	6.3925656	6.8462053	8.3194564
MAPE	23.63027	38.74760	31.064945	34.027592	35.826291
VAF	97.69251	95.72052	97.105392	96.825329	95.903543

8. Conclusions

In this study, tunnel convergence was predicted using five ML approaches of DNN, KNN, GPR, SVR, and DT. A database consisting of 650 data points (440 for training, 110 for validation, and 100 for test) gathered from several metros and road tunneling projects in Iran excavated using the NATM, DB, TBM, and EPB-TBM methods were applied in the models. Finally, considering the database applied in this study, it was concluded that between the ML methods applied in this study and the literature, the DNN model is the most accurate prediction method for the tunnel convergence prediction. The GPR, and SVR were the second, third, and fourth methods, respectively, in prediction accuracy. KNN and DT models with the closeness results to each other produced the least accurate results. Furthermore, the very good accuracies presented by the ML models can mention the right selection of the input parameters effective on the tunnel convergence.

As a suggestion to researchers, since there are many other significant problems in tunneling, they can investigate the prediction ability of these models for other problems.

Conflict of interest

There is no conflict of interest.

References

Adoko, A.C., Jiao, Y.Y., Wu, L., Wang, H. and Wang, Z.H. (2013), "Predicting tunnel convergence using multivariate adaptive regression spline and artificial neural network", *Tunn. Undergr. Sp. Tech.*, **38**, 368-376. https://doi.org/10.1016/j.tust.2013.07.023.

Aktas, G. and Ozerdem, M.S. (2020), "Displacement prediction of precast concrete under vibration using artificial neural networks", *Struct. Eng. Mech.*, **74**(4), 559-565. https://doi.org/10.12989/sem.2020.74.4.559.

Adoko, A.C. and Wu, L. (2012), "Estimation of convergence of a high-speed railway tunnel in weak rocks using an adaptive neuro-fuzzy inference system (ANFIS) approach", *J. Rock Mech. Geotech. Eng.*, **4**(1), 11-18. https://doi.org/https://doi.org/10.3724/SP.J.1235.2012.00011 .

Asadollahpour, E., Rahmannejad, R., Asghari, A. and Abdollahipour, A. (2014), "Back analysis of closure parameters of Panet equation and Burger's model of Babolak water tunnel conveyance", *Int. J. Rock Mech. Min. Sci.*, **68**, 159-166. https://doi.org/https://doi.org/10.1016/j.ijrmms.2014.02.017.

Altman, N.S. (1992), "An introduction to kernel and nearest-neighbor nonparametric regression", The American Statistician, **46**(3), 175-185. doi:10.1080/00031305.1992.10475879.

Debernardi, D. and Barla, G. (2009), "New viscoplastic model for design analysis of tunnels in squeezing conditions", *Rock Mech. Rock Eng.*, **42**(2), 259. https://doi.org/10.1007/s00603-009-0174-6.

Fahimifar, A., Tehrani, F.M., Hedayat, A. and Vakilzadeh, A. (2010), "Analytical solution for the excavation of circular tunnels in a visco-elastic Burger's material under hydrostatic stress field", *Tunn. Undergr. Sp. Tech.*, **25**(4), 297-304. https://doi.org/https://doi.org/10.1016/j.tust.2010.01.002.

Feng, X., Jimenez, R., Zeng, P. and Senent, S. (2019), "Prediction of time-dependent tunnel convergences using a Bayesian updating approach", *Tunn. Undergr. Sp. Tech.*, **94**, 103118. https://doi.org/https://doi.org/10.1016/j.tust.2019.103118.

González el Álamo, J.A. and Jiménez, R. (2011), "Prediction of convergences in rock tunnels excavated by conventional methods", *Proceedings of the* 12th ISRM Congress, Beijing, China. https://doi.org/10.1201/b11646-319.

Guan, Z., Jiang, Y. and Tanabashi, Y. (2009), "Rheological parameter estimation for the prediction of long-term deformations in conventional tunnelling", *Tunn. Undergr. Sp. Tech.*, **24**(3), 250-259. https://doi.org/https://doi.org/10.1016/j.tust.2008.08.001.

Hajihassani, M., Abdullah, S.S., Asteris, P.G. and Armaghani, D.J. J.A.S. (2019), "A gene expression programming model for predicting tunnel convergence", *Appl. Sci.*, **9**(21), 4650. https://doi.org/10.3390/app9214650.

Kamiński, B., Jakubczyk, M. and Szufel, P. (2017), "A framework for sensitivity analysis of decision trees", *Central Eur. J. Operations Res.*, **26**(1),135-159. https://doi.org/10.1007/s10100-017-0479-6. PMC 5767274.

Kontogianni, V., Psimoulis, P. and Stiros, S. (2006), "What is the contribution of time-dependent deformation in tunnel convergence?", *Eng. Geol.*, **82**(4), 264-267. https://doi.org/https://doi.org/10.1016/j.enggeo.2005.11.001.

Kostinakis, K.G. and Morfidis, K.E. (2020), "Optimization of the seismic performance of masonry infilled R/C buildings at the stage of design using artificial neural networks", *Struct. Eng. Mech.*, **75**(3), 295-309. https://doi.org/10.12989/sem.2020.75.3.295.

Liu, X., Liu, Y., Lu, Y. and Kou, M. (2020), "Experimental and numerical study on pre-peak cyclic shear mechanism of artificial rock joints", *Struct. Eng. Mech.*, **74**(3), 407-423. https://doi.org/10.12989/sem.2020.74.3.407.

Mahdevari, S., Shirzad Haghighat, H. and Torabi, S.R. (2013), "A dynamically approach based on SVM algorithm for prediction of tunnel convergence during excavation", *Tunn. Undergr. Sp. Tech.*, **38**, 59-68. https://doi.org/https://doi.org/10.1016/j.tust.2013.05.002.

Mahdevari, S. and Torabi, S.R. (2012), "Prediction of tunnel convergence using Artificial Neural Networks", *Tunn. Undergr. Sp. Tech.*, **28**, 218-228. https://doi.org/https://doi.org/10.1016/j.tust.2011.11.002.

Madenci, E. and Gülcü, Ş. (2020), "Optimization of flexure stiffness of FGM beams via artificial neural networks by mixed FEM", *Struct. Eng. Mech.*, **75**(5), 633-642.

https://doi.org/10.12989/sem.2020.75.5.633.
Mahdevari, S., Torabi, S.R. and Monjezi, M. (2012), "Application of artificial intelligence algorithms in predicting tunnel convergence to avoid TBM jamming phenomenon", *Int. J. Rock Mech. Min. Sci*., **55**, 33-44. https://doi.org/https://doi.org/10.1016/j.ijrmms.2012.06.005
Mahmoodzadeh, A., Rashidi, S., Mohammed, A., Hama Ali, H. and Ibrahim, H. (2022). "Machine learning approaches to enable resource forecasting process of road tunnels construction", Communication Engineering and Computer Science, North America, mar. 2022. Available at: <https://conferences.cihanuniversity.edu.iq/index.php/COCOS/22/paper/view/718>. Date accessed: 21 Sep. 2022. http://doi.org/10.24086/cocos2022/paper.718.
Mahmoodzadeh, A. and Zare, S. (2016), "Probabilistic prediction of expected ground condition and construction time and costs in road tunnels", *J. Rock Mech. Geotech. Eng*., **8**(5), 734-745. https://doi.org/https://doi.org/10.1016/j.jrmge.2016.07.001.
Mahmoodzadeh, A., Mohammadi, M., Abdulhamid, S.N., Nejati, H.R., Noori, K.M.G., Ibrahim, H.H. and Hama Ali, H.F. (2021a), "Predicting construction time and cost of tunnels using Markov chain model considering opinions of experts", *Tunn. Undergr. Sp. Tech*., **116**, 104109. https://doi.org/10.1016/j.tust.2021.104109.
Mahmoodzadeh, A., Mohammadi, M., Abdulhamid, S.N., Ibrahim, H.H., Hama Ali, H.F. and Salim, S.G. (2021b), "Dynamic reduction of time and cost uncertainties in tunneling projects", *Tunn. Undergr. Sp. Tech*., **109**, 103774. https://doi.org/10.1016/j.tust.2020.103774.
Nadimi, S., Shahriar, K., Sharifzadeh, M. and Moarefvand, P. (2011), "Triaxial creep tests and back analysis of time-dependent behavior of Siah Bisheh cavern by 3-Dimensional Distinct Element Method", *Tunn. Undergr. Sp. Tech*., **26**(1), 155-162. https://doi.org/https://doi.org/10.1016/j.tust.2010.09.002.
Nomikos, P., Rahmannejad, R. and Sofianos, A. (2011), "Supported axisymmetric tunnels within linear viscoelastic burgers rocks", *Rock Mech. Rock Eng*., **44**(5), 553-564. https://doi.org/10.1007/s00603-011-0159-0.
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., *et. al*. (2011), "Scikit-learn: Machine learning in python (2011)", *J. Machine Learn.g Res*., **12**, 2825-2830.
Quinlan, J.R. (1987), "Simplifying decision trees", *Int. J. Man-Machine Studies*, **27**(3), 221-234. CiteSeerX 10.1.1.18.4267. doi:10.1016/S0020-7373(87)80053-6.
Rafiai, H. and Moosavi, M. (2012), "An approximate ANN-based solution for convergence of lined circular tunnels in elasto-plastic rock masses with anisotropic stresses", *Tunn. Undergr. Sp. Tech*., **27**(1), 52-59. https://doi.org/https://doi.org/10.1016/j.tust.2011.06.008
Rasmussen, C.E. and Williams, C.K.I. (2016), "Gaussian processes for machine learning", The MIT Press.
Sakurai, S. (1978), "Approximate time-dependent analysis of tunnel support structure considering progress of tunnel face", *Int. J. Numer. Anal. Method. Geomech*., **2**(2), 159-175. https://doi.org/10.1002/nag.1610020205.
Sharifzadeh, M., Tarifard, A. and Moridi, M.A. (2013), "Time-dependent behavior of tunnel lining in weak rock mass based on displacement back analysis method", Tunnelling and Underground Space Technology, **38**, 348-356. https://doi.org/https://doi.org/10.1016/j.tust.2013.07.014
Sterpi, D., and Gioda, G. (2009). "Visco-Plastic Behaviour around Advancing Tunnels in Squeezing Rock", *Rock Mech. Rock Eng*., **42**(2), 319-339. https://doi.org/10.1007/s00603-007-0137-8.
Schulz, H. and Behnke, S. (2012), "Deep Learning", Künstl Intell **26**, 357-363. https://doi.org/10.1007/s13218-012-0198-z
Torabi-Kaveh, M. and Sarshari, B. (2020), "Predicting Convergence rate of Namaklan twin tunnels using machine learning methods", Arabian Journal for Science and Engineering, **45**(5), 3761-3780. https://doi.org/10.1007/s13369-019-04239-1.
Trinh, M.C. and Jun, H. (2021), "Stochastic vibration analysis of functionally graded beams using artificial neural networks", *Struct. Eng. Mech*., **78**(5), 529-543. https://doi.org/10.12989/sem.2021.78.5.529.
Vapnik, V.N. (2000), "The nature of statistical learning theory. New York: Springer", Edition Number: 2, ISBN 978-1-4757-3264-1. DOI: 10.1007/978-1-4757-3264-1
Vu, T.M., Sulem, J., Subrin, D., Monin, N. and Lascols, J. (2013), "Anisotropic closure in squeezing rocks: The example of saint-martin-la-porte access gallery", *Rock Mech. Rock Eng*., **46**(2), 231-246. https://doi.org/10.1007/s00603-012-0320-4.

CC