



A soft computing approach to tunnel face stability in a probabilistic framework

Enrico Soranzo¹ · Carlotta Guardiani¹ · Wei Wu¹

Received: 25 February 2021 / Accepted: 26 May 2021 / Published online: 2 August 2021
© The Author(s) 2021

Abstract

Tunnel face is important for shallow tunnels to avoid collapses. In this study, tunnel face stability is studied with soft computing techniques. A database is created based on the literature which is used to train some broadly adopted soft computing techniques, ranging from linear regression to the artificial neural network. The soil dry density, cohesion, friction angle, cover depth and the tunnel diameter are used as the input parameters. The soft computing techniques state whether the face support is stable and predict the face support pressure. It is found that the artificial neural network outperforms the other techniques. The face support pressure is predicted with the artificial neural network for statistically distributed samples, and the failure probability is obtained with Monte Carlo simulations. In this way, the stability of the tunnel face can be reliably assessed and the support pressure can be estimated fairly accurately.

Keywords Failure probability · Machine learning · Monte Carlo simulation · Soft computing · Tunnel face stability

Abbreviations

ANN	Artificial neural network
CMT	Centrifuge model test
DEM	Discrete element method
DTR	Decision tree regressor
FDM	Finite difference method
FEM	Finite element method
KNR	K-Nearest neighbours
LIR	Linear regression
LSL	Limit state line
LOR	Logistic regression
LRM	Load reduction method
MAE	Mean absolute error
ML	Machine learning
MD	Monitoring data
MC	Monte Carlo
NA	Numerical analysis
NATM	New Austrian tunnelling method
PSO	Particle swarm optimisation
RFR	Random forest regressor
RMSE	Root mean squared error

SVR	Support vector regressor
TBM	Tunnel boring machine
VR	Voting regressor
XGB	Extreme gradient boosting
1gMT	1g model test

List of symbols

C	Soil cover
C_f	Cost function
COV	Coefficient of variation
D	Euclidean distance, tunnel diameter
F_n	Number of false negatives
F_p	Number of false positives
I_d	Density index
L	Layer of the ANN
K	Covariance matrix
K_0	Coefficient of earth pressure at rest
N_c	Bearing factor
N_q	Bearing factor
N_γ	Bearing factor
P_f	Probability of failure of the reliability method
\widetilde{P}_f	Probability of failure of the MC method
R^2	Coefficient of determination
S_r	Degree of saturation
T_n	Number of true negatives
T_p	Number of true positives
X	Features

✉ Enrico Soranzo
enrico.soranzo@boku.ac.at

¹ Institute of Geotechnical Engineering, University of Natural Resources and Life Sciences, Feistmantelstraße 4, 1180 Vienna, Austria

X'	Scaled features
a	Activation function
b	Bias term
c'	Effective cohesion
e	Void ratio
e_{\max}	Maximum void ratio
e_{\min}	Minimum void ratio
n	Number of data points, porosity
q	Surcharge load acting at the ground level
w	Weight term
x	Vector of the random variables
y	Label
\bar{y}	Mean observed label
y_i	Observed label
\hat{y}_i	Predicted label
Φ	Cumulative distribution function of a standard normal variable
β_i	Coefficients of the parametric models
β_{HL}	Hasofer and Lind reliability index
φ	Friction angle
γ	Soil unit weight, kernel coefficient
γ'	Effective unit weight
γ_d	Dry unit weight
γ_s	Soil particle unit weight
γ_{sat}	Saturated unit weight
γ_w	Unit weight of water
μ	Vector of the mean values of the random variables
μ_c	Mean values of the cohesion
μ_φ	Mean value of the friction angle
$\rho_{c\varphi}$	Coefficient of correlation between cohesion and friction angle
ρ_s	Soil particle density
σ_c	Standard deviation of the cohesion
σ_φ	Standard deviation of the friction angle
σ_T	Support pressure at the tunnel axis
$\hat{\sigma}_{T,1}, \hat{\sigma}_{T,2}$	Normalised support pressure at the tunnel axis
$\hat{\sigma}_{T,3}$	Support pressure ratio at the tunnel axis
σ'_T	Effective support pressure at the tunnel axis
$\hat{\sigma}'_{T,3}$	Effective support pressure ratio at the tunnel axis
σ_w	Pore water pressure at the tunnel axis

1 Introduction

Most tunnels are excavated either with the New Austrian Tunnelling Method (NATM) or with the Tunnel Boring Machine (TBM). For deep tunnels, the essence of NATM is to allow some ground deformation to reduce the pressure on the tunnel lining. For shallow tunnels in soft ground,

however, the deformation should be minimised to maintain the inherent strength of the ground and to avoid damage to surface structures. Moreover, the stability at the tunnel face is important for shallow tunnels to avoid collapses. In principle, the face stability can be improved either by enhancing the strength of the surrounding ground or by providing support measures such as supporting cores, shotcrete sealing, horizontal anchors and forepoling. The strength enhancement at the tunnel face can be achieved by either grouting or by lowering the groundwater level. In mechanised tunnels, the face support is provided by the TBM, e.g. slurry shield or EPB shield. An estimate of the support pressures is required for safe and efficient construction.

The problem of face stability can be solved with analytical, numerical and experimental approaches. The analytical methods are mainly based on the limit state analysis. Alternatively, the problem can also be studied by 1 g model tests and centrifuge model tests. The face stability can also be studied by numerical analysis, e.g. the Finite Element Method (FEM), the Finite Difference Method (FDM) and the Discrete Element Method (DEM).

Recently, soft computing (SC) has emerged as a promising technique for predictive assessment in geotechnical engineering. Until now, little has been done to apply SC to the tunnel face stability and failure probability. Compared with the aforementioned deterministic approaches, SC is particularly appealing in view of the natural variability of the soil properties. This paper focuses on the application of SC methods based on four datasets: 1 g model tests, centrifuge tests, monitoring data and numerical analysis. The face stability is considered both as a classification and as a regression problem. While the classification deals with the question of whether a face support is necessary, the regression answers the question of how much support is needed. For the classification, the Logistic Regression (LOR) is considered. For the regression, the Linear Regression (LIR) is used as a benchmark and some Machine Learning (ML) techniques are critically assessed, such as the Decision Tree Regressor (DTR), the K-Nearest Regressor (KNR) and the Support Vector Regressor (SVR). The combinations of various ML methods—the so-called ensembles—such as the Random Forest Regressor (RFR), Voting Regressor (VR) and Extreme Gradient Boosting (XGB), are also considered. Finally, the Artificial Neural Network (ANN) is assessed.

Then, the most accurate SC technique is selected and its predictive capacity is augmented within a probabilistic framework. To estimate the failure probability, two statistical distributions of the shear parameters (i.e. normal and log-normal) and the correlation within the parameters are compared.

This paper is organised as follows. In the next section, the relevant literature is reviewed. In Sect. 3, the dataset, the SC techniques and the probabilistic simulations used in this study are presented. Section 4 presents the results which are discussed in Sect. 5. Finally, Sect. 6 concludes the paper.

2 Tunnel face stability

Different approaches to tunnel face stability can be found in the recent guidelines [14]. The effective support pressure can be written in the following form which resembles Terzaghi's bearing capacity of shallow foundations:

$$\sigma_T = \gamma DN_\gamma + qN_q - cN_c \quad (1)$$

where γ is the soil unit weight, D is the tunnel diameter, q is the surcharge load at the ground level and c is the cohesion (Fig. 1). N_c , N_q and N_γ are the bearing capacity factors. The contribution of water is equal to the pore water pressure σ_w acting at the level of the tunnel face [14]. Therefore, the two contributions, i.e. that of earth and of water pressure, are expressed as

$$\sigma_T = \sigma'_T + \sigma_w \quad (2)$$

where σ'_T is the effective support pressure.

Analytically, either limit equilibrium or limit analysis is used. In limit equilibrium approaches, a failure mechanism is defined and the equilibrium equations for the forces acting onto or within the failure volume are solved. The recent guidelines for engineering practice [14] are based on this approach. The limit equilibrium mechanisms for tunnels were first developed by [22]. These methods were applied to TBM tunnels by Anagnostou and Kovári [5, 6] and Jancsecz and Steiner [25]. Recent developments include the works of Anagnostou [4], Chen et al. [11] and Hu et al. [23].

Within limit analysis, various solutions have been formulated. In this framework, two types of solutions exist: upper and lower bounds [15, 32, 43, 56].

Other than analytically, face stability can be also assessed experimentally. Two classes of experiments exist in the literature: 1g and centrifuge model tests. 1g model tests are easier to carry out and allow a more sophisticated instrumentation. Ahmed and Iskander [2] evaluated face stability with transparent soils. Chen et al. [12] validated their FDM tests with 1g-tests. Kirsch [30] studied the development of the failure mechanism and the support force at the face in dry sand. Liu et al. [35] developed a model test device for shield excavation with ideal slurry film to validate 2D FDM tests. Lüe et al. [38] carried out 1g model tests under seepage conditions. Lüe et al. [37]

studied the failure of shield tunnel face in cross-anisotropic granular media with 1 g model tests and DEM. Sterpi and Cividini [55] performed 1g model tests and FEM to clarify the phenomenon of strain localisation.

Under their augmented gravity field, centrifuge model tests can mimic the level of stress of real world tunnels. Experimental results validate analytical formulations. For instance, Messerli et al. [41] validated the limit equilibrium method of Anagnostou and Kovári [5]; centrifuge model tests [10, 40] validated the model of Davis et al. [15] and Leca and Dormieux [32]. Lüe et al. [39] carried out centrifuge tests to validate their FEM tests, which also considered seepage forces. Centrifuge tests under unsaturated conditions were provided by Soranzo and Wu [53].

Finally, the problem of tunnel face stability can also be solved by numerical analysis. Mostly, the tunnel support pressure is obtained with the FEM [3, 18, 55, 57, 58, 68], but also the FDM is largely adopted [12, 34, 61]. Thanks to the ever increasing computational speed of modern computing, the DEM is recently gaining momentum [13, 37, 59, 60, 69].

2.1 Soft computing techniques

Soft Computing is a collective term for various disciplines of computer science that deal with approximate solution methods that are similar to natural information processing. Various methods explored in this study belong to a subsets of SC techniques, namely Machine Learning (ML). According to two state-of-the-art reviews of SC applications [17, 64], the adoption of SC techniques in geotechnical engineering is growing exponentially. ANN is the leading technique and represents about half of the studies in this field. Most geotechnical engineering applications deal with the soil and rock properties, slope stability and deep foundations.

A comprehensive review of the state-of-the-art application of SC techniques to underground excavations can be found in [66]. In the following, some of the studies related to tunnel excavations in soil are recalled. The studies focus on few different aspects such as prediction of the lining stability [33, 65], of the convergence of the cavity [1, 46], of the magnitude of ground settlements [31, 62, 70] and of the occurrence of the overbreak [26]. Studies on TBM performance [8, 45] and steering [63] are also found in the literature. These studies applied ANN [8, 31], RFR [67, 70] and SVR [45, 46, 68]. One study considers also LIR and KNR [66].

One formal attempt to address the tunnel face stability problem with SC techniques was made by [48]. The authors applied ML and defined a Face Vulnerability Index (FVI) to assess the stability conditions of tunnels based on a database of 36 case histories. This objective of the

defined index—which varies between 0 and 100—is to represent the instability potential of a tunnel [48]. However, the authors made no attempt to predict the support pressure. The present study is poised to fill this knowledge gap.

2.2 Probabilistic methods

For engineering structures, failure is defined as the condition for which load equals resistance. Semi-probabilistic approaches overcome uncertainties by multiplying load and resistance by partial safety factors. However, building standards acknowledge that this is a simplification and allow the probabilistic approach [9]. Probabilistic methods account for the uncertainties associated with the variables at play. Therefore, failure is not defined by a binary variable, such as stable/unstable, but rather by its probability of occurrence. In a deterministic framework, geotechnical parameters are fixed; within probabilistic methods, instead, they are defined by a statistical distribution. Typically, a statistical distribution is assumed for the shear parameters only, but other parameters can also be considered.

The second ingredient in every probabilistic method is the definition of a failure domain. The failure and safety domains are separated by the Limit State Line (LSL). The shape of the LSL can be defined by considering an analytical solution. However, if the solution is not explicit, the shape of the LSL is unknown a priori. This situation occurs in practice because of the complexity of the problem (number of variables involved, non-linear behaviour, etc.). In this case, the LSL can be found tentatively by numerical analysis [20]. Eventually, a closed-form LSL can be interpolated based on the numerical results.

Once the LSL is retained, failure probability can be computed according to the procedure outlined in Low and Tang [36] and depicted in Fig. 2. In Fig. 2, a normal distribution is assigned to the shear parameters. In doing so, the normal dispersion ellipse is obtained according to [16].

$$\beta_{HL} = \min_{x \in F} \sqrt{(x - \mu)^T K^{-1} (x - \mu)} \tag{3}$$

where β is the [21] reliability index, F is the failure region, x and μ are the random variables and their mean values and K is the covariance matrix. The failure probability is approximated with Eq. 4.

$$P_f = \Phi(-\beta_{HL}) \tag{4}$$

where Φ is the cumulative distribution function of a standard normal variable. Low and Tang [36] also outline the procedure to deal with log-normally distributed variables.

Alternatively, a Monte Carlo (MC) simulation can be performed. With MC, samples are generated with their chosen probability distribution functions. Then, the

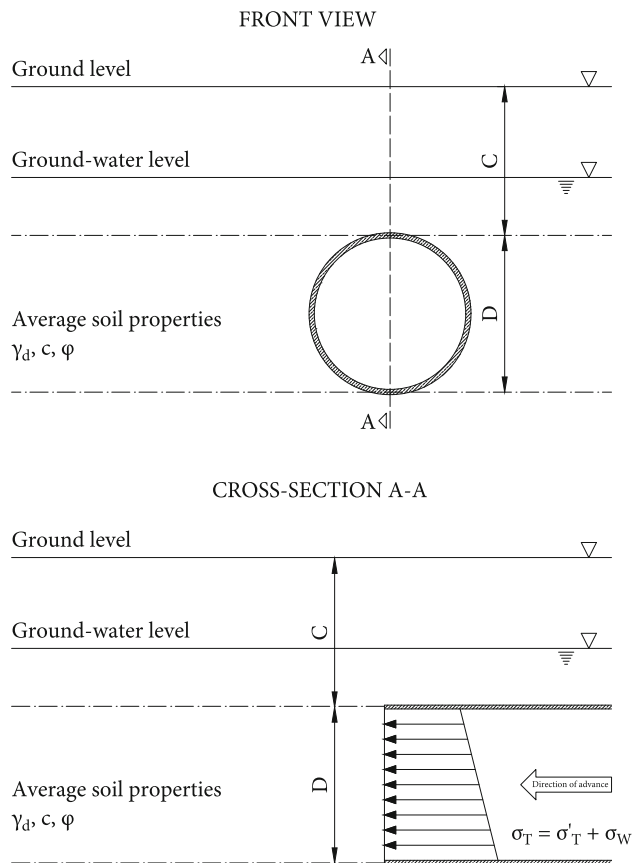


Fig. 1 Problem statement

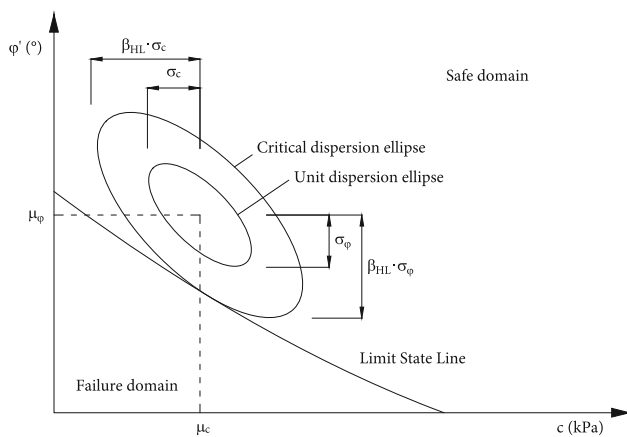


Fig. 2 Limit State Line and normal dispersion ellipse

structural response (failure/safety) is calculated for each point. The failure probability is calculated with the following equation:

$$\widetilde{P}_f = \frac{1}{n} \sum_{i=1}^n I(x_i) \tag{5}$$

where $I(x) = 1$ if the point is inside the failure domain. In Fig. 3, for example, a normally correlated sample with $n =$

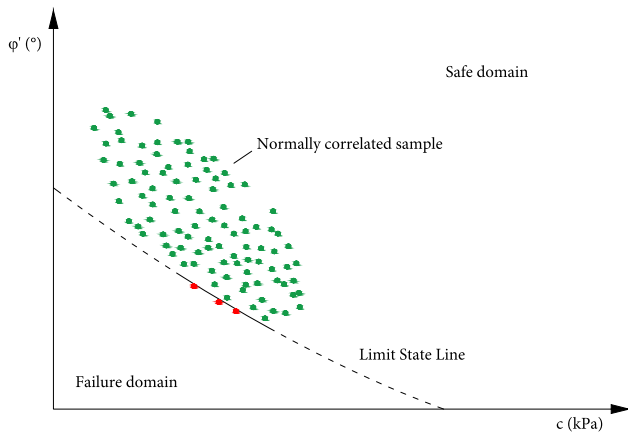


Fig. 3 Monte Carlo simulation

100 data points is generated. Three points fall within the failure zone. Therefore, the failure probability is 3% (Fig. 4).

Probabilistic methods for tunnel face stability [42, 44, 50] rely on limit analysis to determine the LSL. Pan and Dias [49] employed SC techniques to enhance computational efficiency in their probabilistic method, which is ultimately based on a complex analytical formulation of the LSL. Goh and Kulhawy [20] do not apply any aprioristic definition of the LSL, but rather employ SC techniques to determine the LSL based on numerical results.

In this paper, the LSL is neither calculated analytically, nor predicted via SC techniques based on numerical analysis. A third way is explored, in which the structural response is calculated for each point in the sample with SC techniques and the LSL is the ideal boundary separating the stable from the unstable points.

3 Methodology

In this section, the methodology is presented. It consists of the dataset preparation (Sect. 3.1), the classification and regression methods (Sects. 3.3 and 3.4) and the MC simulations (Sect. 3.5).

3.1 Dataset preparation

A dataset is constructed based on the available literature in Scopus. The keywords “Tunnel”, “Face” and “Stability” are used. The keywords “Rocks”, the analytical methods (“Limit Analysis” and “Limit Equilibrium”) and seepage are excluded. The results are limited to the English language and irrelevant disciplines are neglected (“Business”, “Medicine”, etc.). With this approach, the authors were able to collect 658 documents as to 01.10.2020. Within these documents, 21 have been selected, which present data on tunnel face stability. These papers are listed in Table 1. The data originate from 1g Model Tests (1gMT), Centrifuge Model Tests (CMT), Numerical Analysis (NA) and Monitoring Data (MD).

In data science parlance, independent and dependent variables are called “features” and “labels”, respectively. In this dataset, the features are the dry unit weight γ_d , the cohesion c , the friction angle ϕ , the soil cover C and the diameter D . The label is the effective support pressure σ'_T . Obviously, if the soil is dry, $\sigma_w = 0$ and $\sigma'_T = \sigma_T$. Data are not always in the wished form. Therefore, feature engineering (the practice of adapting the variables based on domain knowledge) is performed as follows. The values of the features and labels are taken at the tunnel axis, except for the data of Zhang et al. [68]. In Zhang et al. [68], two layers of soil are present at the tunnel face. Therefore, the

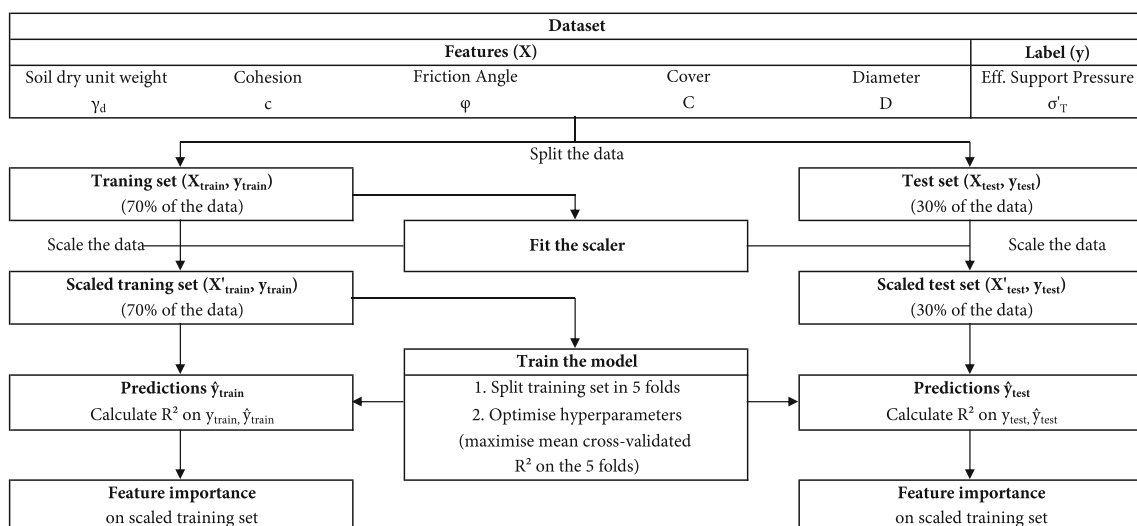


Fig. 4 Workflow of the proposed SC techniques

Table 1 Dataset references

Reference	Test type ^a	Soil type ^b
Ahmed and Iskander [2]	1gMT	Sa ^c
Alagha [3]	NA	N/A
Chambon and Corté [10]	CMT	Sa
Chen et al. [13]	NA	Sa
Chen et al. [12]	1gMT	Sa
Eshragi and Zare [18]	MD	Cl, Gr, Sa
Idinger et al. [24]	CMT	Sa
Kim and Tonon [28]	NA	Cl, Sa
Kirsch [30]	1gMT	Sa
Li et al. [34]	Na	N/A ^d
Liu et al. [35]	1gMT	Sa
Lüe et al. [38]	1gMT	Sa
Lüe et al. [37]	1gMT	Sa ^e
Lüe et al. [39]	CMT ^f	Sa
Soranzo et al. [54]	CMT	sa Si
Ukritchon et al. [57]	NA	Cl
Wang [59]	NA	Sa
Wang et al. [60]	NA	N/A
Zhang et al. [68]	MD ^g	Sa
Zhang et al. [61]	NA	Cl, Sa
Zhang et al. [69]	NA	Cl

^a1gMT = 1 g Model Tests, CMT = Centrifuge Model Tests, MD = Monitoring Data, NA = Numerical Analysis

^bFor NA, the soil type is implicit in the soil parameters. For the purpose of this table, the soil type is reported as stated in the corresponding reference, unless no soil type is explicitly given, for which this feature is not applicable (“N/A”)

^cThe authors use a transparent soil with “similar stress-strain characteristics to sand.” [2]

^dThe authors do not state what type of soil they simulate. However, they consider a purely cohesionless soil

^eThe authors use rice grains to simulate Toyura sand [37]

^fIn this paper, centrifuge test results are back-analysed with numerical analysis. Only the centrifuge test data points are retained in this dataset

^gIn this paper, monitoring results are back-analysed with numerical analysis. Only the monitoring data points are retained in this dataset

weighted average values are taken for the features. In most sources listed in Table 1, the dry soil unit weight γ_d is given directly. In some others [28, 68], the soil is saturated and γ_d is back-calculated with Eq. 6.

$$\gamma_{sat} = \gamma_d + n \cdot \gamma_w \quad (6)$$

The porosity is obtained with Eq. 7.

$$n = 1 - \frac{\gamma_d}{\gamma_s} \quad (7)$$

where $\gamma_s = \rho_s \cdot g$ is the soil particle unit weight and ρ_s is

the soil particle density which is assumed equal to 2.65 kg/m³ for sand and silt. In Kirsch [30], the density index I_d and the minimum and maximum void ratios e_{min} , e_{max} are given. The void ratio e is calculated with Eq. 8.

$$e = e_{max} - I_d \cdot (e_{max} - e_{min}) \quad (8)$$

In this case, the dry unit weight γ_d is obtained with Eq. 9 which is derived from Eq. 7.

$$\gamma_d = \frac{\gamma_s}{1 + e} \quad (9)$$

The same procedure is followed to obtain the dry density from the data of Lüe et al. [37] and Zhang et al. [69]. One reference [57] is neglected in further calculations, given the impossibility of obtaining the soil dry unit weight. The support pressure is sometimes given as a ratio. The normalised support pressure is defined with Eq. 10a in Ahmed and Iskander [2] and with Eq. 10b in Kirsch [30] and Liu et al. [35].

$$\hat{\sigma}_{T,1} = \frac{\sigma_T}{\gamma_d \cdot (C + \frac{D}{2})} \quad (10a)$$

$$\hat{\sigma}_{T,2} = \frac{\sigma_T}{\gamma_d \cdot D} \quad (10b)$$

The support pressure ratio [61, 69] is given in Eq. 11.

$$\hat{\sigma}_{T,3} = \frac{\sigma_T}{K_0 \cdot \gamma_d \cdot (C + \frac{D}{2})} \quad (11)$$

where K_0 and γ_d are the coefficient of earth pressure at rest and the dry unit weight, respectively (Tables 2, 3). The effective support pressure ratio [28] is given in Eq. 12.

$$\hat{\sigma}'_{T,3} = \frac{\sigma'_T}{K_0 \cdot \gamma' \cdot (C + \frac{D}{2})} \quad (12)$$

The descriptive statistics of the dataset are shown in Table 4; 1829 data points are retrieved from the dataset. Table 4 shows the mean value, standard deviation, minimum and maximum values, and percentile of the features and the label.

Their frequency distributions are shown along the diagonal of the pairplot in Fig. 5. The non-diagonal elements of Fig. 5 depict the correlation between the variables.

In quantitative terms, correlation can be efficiently resumed by the correlation matrix in Fig. 6. As expected, the support pressure correlates negatively with the cohesion and friction angle. It correlates positively with soil cover and tunnel diameter. Therefore, it appears that the soil cover, diameter and cohesion have the major impact on the results. Also, the cohesion and friction angle in the dataset correlate negatively with each other.

Table 2 Hyperparameters tuning

Regressor	Hyperparameters	Subset	Tuning procedure ^a
Individual SC techniques			
DTR	max_depth	1–10	Grid search
	max_features	1–5	
KNR	n_neighbours	0–50	Grid search
SVR	C	0–1000	Particle Swarm optimisation
	γ	0.001–1	
Ensemble learning			
RFR	max_depth	1–10	Grid search
	max_features	1–5	
	n_estimators	1–10	
VR	w_1	0–3	Particle Swarm optimisation
	w_2	0–3	
	w_3	0–3	
	with $w_1 + w_2 + w_3 = 3$		
XGB	max_depth	1–10	Grid search
	max_features	1–5	
	n_estimators	100–1000	

^aLOR, LIR and ANN are parametric techniques. Therefore, no hyperparameter tuning is performed

Table 3 Parameters of the Monte Carlo simulations

	Dry soil Unit weight γ_d (kN/m ³)	Cohesion c (kPa)	Friction angle φ (°)	Soil cover C (m)	Diameter D (m)
Deterministic/random variable	Deterministic	Random	Random	Deterministic	Deterministic
Mean value	20	5	30	10	10
Standard deviation	–	1	3	–	–
COV (%)	–	20	10	–	–

Table 4 Descriptive statistics of the dataset

	Dry soil unit weight γ_d (kN/m ³)	Cohesion c (kPa)	Friction angle φ (°)	Soil cover C (m)	Diameter D (m)	Effective support pressure σ'_T (kPa)
No. of values	1829	1829	1829	1829	1829	1829
Mean value	13.323	36.513	28.768	20.248	8.830	21.491
Standard deviation	0.893	33.082	6.508	14.467	3.525	28.488
Minimum value	7.000	0.000	8.770	0.038	0.025	0.000
25% percentile	13.160	5.000	25.000	10.000	5.000	0.000
50% percentile	13.160	30.000	30.000	14.000	7.000	10.878
75% percentile	13.160	60.000	35.000	28.000	10.000	32.633
Maximum value	21.300	100.000	45.000	56.000	15.430	168.805

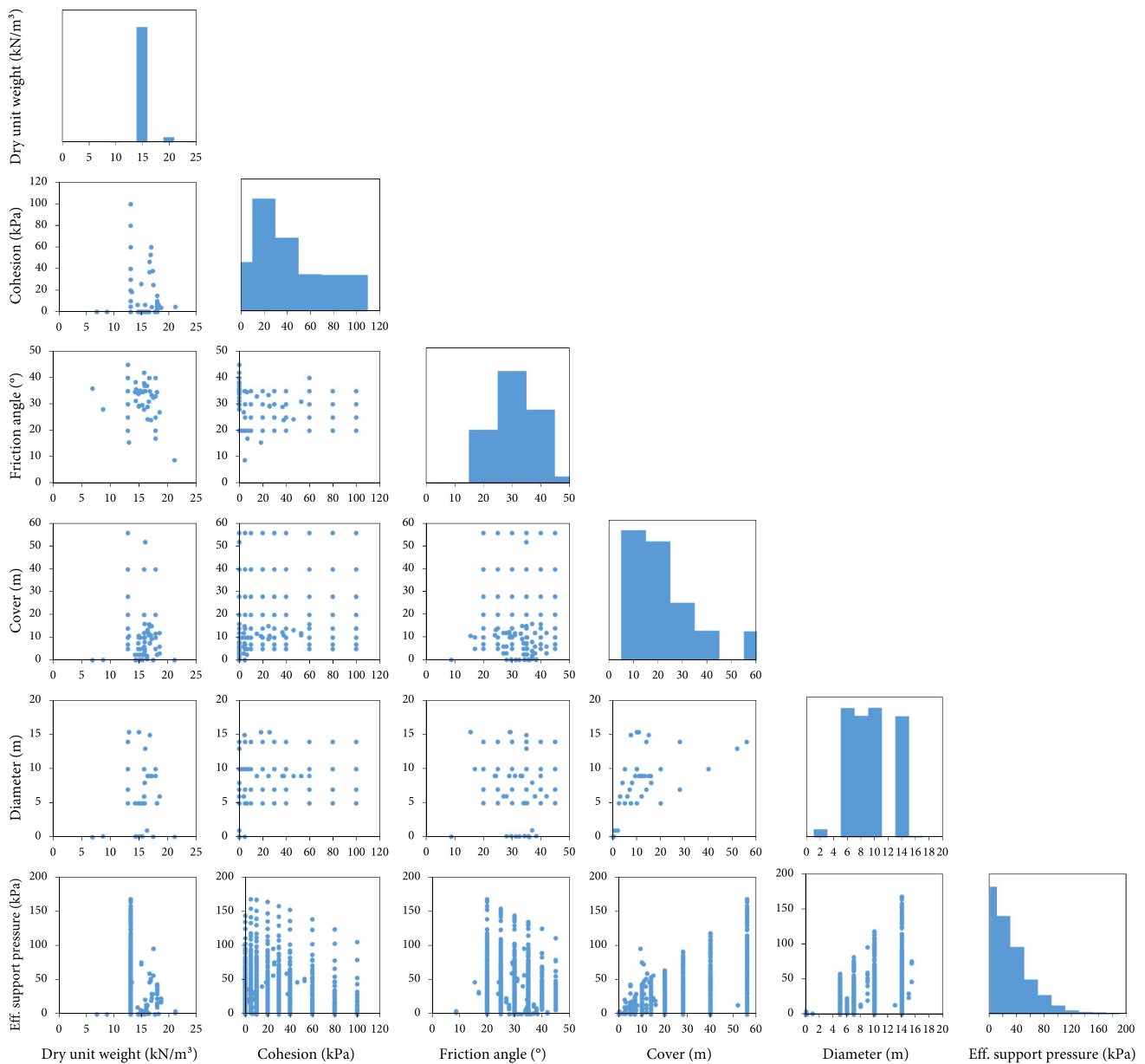


Fig. 5 Pairplot of the features and the label. Histograms of the statistical distribution and dot plots of the correlation between the variables

c (kPa)	0.16				
φ (°)	0.0046	-0.21			
C	-0.14	0.051	-0.022		
D (m)	-0.082	0.055	-0.032	0.57	
σ'_T (kPa)	-0.012	-0.4	-0.066	0.68	0.43
γ_d (kN/m ³)					
	c (kPa)	φ (°)	C (m)	D (m)	

Fig. 6 Correlation matrix

3.2 Workflow

The workflow depicted in Fig. 4 is carried out with the general-purpose programming language Python. First, the

columns of the dataset are assigned to either the features X or to the label y . Then, the dataset is split into the training and the test sets, as explained in Sect. 3.2.1. The data scaler is fitted on the training data only, but both the training and the test sets are scaled (Sect. 3.2.2). The hyperparameters are optimised (“tuned”) by maximising the mean cross-validated R^2 (Sect. 3.2.3) calculated on fivefold of the training data, as explained in Sect. 3.2.4. Once the model is trained, predictions are made for both the training and the test set. Finally, feature importance is calculated, as shown in Sect. 3.2.5. The same random state is selected for all models to compare the results.

3.2.1 Train–test split

70% of the data (1280 data points) is used to train the models; 30% of the data (549 data points) is used to evaluate the model (testing). The 70%-30% split is a common choice among data scientists. The splitting task is performed by the function `Train_Test_Split` from the Scikit-learn library [51].

3.2.2 Scaling

Scale differences across features jeopardise model performance. Large feature values determine large weight values, leading to numerical instability, poor learning performance during training and sensitivity to feature values. Not all models are equally affected by the scale difference. Models that use a weighted sum of input variables (LIR, LOR and ANN) or distance between predictors (KNR and SVR) are affected; DTR, RFR and XGB are unaffected. In this study, data are scaled using the `StandardScaler` preprocessing transformer from the Scikit-learn library, according to

$$X' = \frac{X - \mu}{\sigma} \quad (13)$$

where μ and σ are the mean value and the standard deviation of the feature X .

3.2.3 Hyperparameter tuning

SC techniques can be parametric or nonparametric. Techniques that select a form for the predictor are parametric. These techniques learn the function coefficients from the training data. LOR and LIR belong to this class. Other techniques, such as the DTR, KNR, RFR, SVR and XGB, do not assume the form of the function and are called “nonparametric”. ANN are parametric in nature, but, given their adaptive training procedure, they operate as nonparametric. The term “nonparametric” does not imply that models are without parameters. In fact, their architecture is defined by so-called “hyperparameters” whose values are selected with the procedure called “hyperparameter tuning”. This procedure aims at minimising an objective function. In this study, the objective function is $1 - R^2$, where R^2 is the mean value of the coefficient of determination of the cross-validated folds, as explained in Sect. 3.2.4. Depending on the SC technique chosen, two tuning strategies are used, namely Grid Search (Sect. 3.2.3) and Particle Swarm Optimisation (Sect. 3.2.3). The tuning procedures for each SC technique are listed in Table 2. Grid Search is used when the hyperparameters have discrete values (DTR, RFR, KNR and XGB), Particle Swarm Optimisation for continuous values (SVR and VR). Since

both find the global minima, the consistency of algorithms comparison is ensured.

Grid Search Grid Search fits the model with all the possible combinations of the given subset of hyperparameters. By fitting every combination one by one, this procedure is computationally expensive. However, it is effective for discrete values of the hyperparameters.

Particle Swarm Optimisation Common optimisation algorithms such as Gradient Descent perform best for convex functions in a low-dimensional space. Since these conditions are seldom verified, Particle Swarm Optimisation (PSO) is often adopted in SC. PSO is based on the social behaviour exhibited by birds or fish when striving to reach a destination. In optimisation terms, the destination is the global minimum. In practice, random particles are generated, which search the minimum of the objective function (i.e. the function to be minimised) in their vicinity. At the end of each iteration, the particles communicate the value of the objective function and their locations to the swarm. Then, the particles move towards the best individual position and the procedure is repeated until the termination criterion is met. In this study, PSO is used to optimise the hyperparameters of the SVR and VR which are continuous variables. The Python library `pyswarms` [47] is used. 1000 and 100 particles are chosen for the SVR and VR, respectively. As explained in Sect. 3.4.2, VR is built on three regressors. Since these regressors are already optimised, a lower quantity of particles than for SVR is sufficient for the tuning procedure.

3.2.4 Cross-validation

In order to train SC algorithms with the maximum amount of available data, validation sets are often neglected in data science. In this study, no validation set is present, but just the training and the test set. Instead, cross-validation is performed on the training set. In cross-validation, the training set is split into several subsets of the same size, which are called folds. In this study, fivefold cross-validation is chosen. For a given set of hyperparameters, the model is trained on four folds and tested on the remaining one. At each iteration, the mean cross-validated score R^2 is computed.

3.2.5 Feature importance

Feature importance calculates the contribution rate of each feature to the results. Different methods yield different feature importances. However, their values are generally comparable. In Python, DTR, RFR and XGB have their built-in feature importance algorithms which are based on Gini importance, i.e. the probability of misclassifying a

data point if it were randomly labelled. For the remaining models, permutation importance is used which randomly shuffles each feature and computes the corresponding variation in the performance of the model.

3.2.6 Performance metrics

Four performance metrics are calculated for classification, namely Precision, Recall, F_1 -Score and Accuracy. They are defined according to Eqs. 14, 15, 16, 17.

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (14)$$

$$\text{Recall} = \frac{T_p}{T_p + F_n} \quad (15)$$

$$F_1\text{-Score} = \frac{2(\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}} \quad (16)$$

$$\text{Accuracy} = \frac{T_p + T_n}{n} \quad (17)$$

T_p is the number of true positives (the number of correctly classified positive labels), T_n is the number of true negatives, F_p is the number of false positives, F_n is the number of false negatives and n is the number of data points.

Three performance metrics are calculated for regression, namely the coefficient of determination (R^2), the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE). They are computed as shown in Eqs. 18, 20, 20.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (18)$$

$$\text{RMSE} = \sqrt{\sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n}} \quad (19)$$

$$\text{MAE} = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (20)$$

where y_i and \hat{y}_i are the observed and predicted labels, respectively, and \bar{y} is the mean observed label.

3.3 Classifier

At first, tunnel face stability is framed as a classification problem. As mentioned before, this problem corresponds to that of NATM tunnelling: the objective is to determine whether the tunnel face is stable. If the support pressure is equal to or lower than zero, the tunnel face is stable. For the classification, the values of the support pressure in the dataset greater than zero are changed to 1. In doing so, the support pressure becomes a binary value equal to:

- 0 when no support is required (the tunnel face is stable)

- 1 when a support is required (the tunnel face is unstable)

If the tunnel face is unstable, some type of support must be provided.

Classification is performed with Logistic Regression. In this study, LOR models the probability of failure which ranges between 0 (stability) and 1 (failure). For any given value of the features, a prediction can be made according to Eq. 21.

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \sum_{i=1}^n \beta_i X_i \quad (21)$$

The model is fitted with the method of maximum likelihood, i.e. estimates for the model parameters β_i are searched such that the predicted probability $p(X_i)$ of failure for each data point corresponds as closely as possible to the observed status.

3.4 Regressors

Regression is used to predict the support pressure. Eight methods, four individual SC techniques (Sect. 3.4.1) and four ensemble methods (3.4.2) are considered.

3.4.1 Individual SC techniques

Linear Regression The objective of LIR is to find a hyperplane that models the data points the best according to Eq. 22.

$$\hat{y} = \beta_0 + \sum_{i=1}^n \beta_i X_i \quad (22)$$

The coefficients in Eq. 22 are found by minimising the error between the predicted and observed values.

Decision Tree Regression Decision Tree Regression segments the features domain. In each segment, predictions are made based on their mean value. The name Decision Tree comes from the set of splitting rules used to segment the features domain which resembles a tree. In DTR, there are three types of nodes: root, interior and leaf nodes. The root node is the initial node (the entire dataset), the interior nodes represent the features, and the leaf nodes represent the label. The branches represent the decision rules. In this study, two hyperparameters are tuned, namely `max_depth` (the maximum depth of the tree) and `max_features` (the number of features to consider when looking for the best split). DTR is a straightforward method. However, it typically underperforms most methods.

K-Nearest Regression The objective of KNR is to find a certain number k of data points close to another point and to predict its label based on its neighbours. The number of nearest neighbours k is a hyperparameter. The nearest

neighbours are defined based on their distance from the new point. In this study, the Euclidean distance is considered, as defined in Eq. 23.

$$D(X, X_j) = \sqrt{\sum_{i=1}^n (X_i - X_j)^2} \quad (23)$$

X_j is the new point and X_i are the neighbours. Despite its simplicity, KNR is a moderately accurate method.

Support Vector Regression SVR is a supervised learning method that is effective in high dimensional spaces. It uses a subset of training points (“support vectors”) in the decision function (“kernel function”). The underlying idea of SVR is to retain the error within a given margin (“ ε -tube”).

In this study, the hyperparameter ε is set equal to 0.1 and the tolerance is equal to 10^{-3} . The Radial Basis Function kernel (RBF) is considered. The penalty parameter C and the kernel coefficient γ are the hyperparameters to be optimised.

Artificial Neural Network The underlying idea behind ANN is to artificially mimic biological intelligence [52]. Within the ANN framework, a function $f(X)$ of the data X is approximated by a neural network. Neurons represent the input features X and the output label y . The features and the labels are indirectly connected by one or more hidden layers. The function $f(X)$ is expressed as in Eq. 24.

$$\hat{y} = f(X) = \sum_{i=1}^n x_i \cdot w_i + b_i \quad (24)$$

where w_i are the weights and b_i the bias terms. In a single neuron, the function $f(x)$ can be constrained by using an activation function. Let $z = x \cdot w + b$, the REctified Linear Unit (RELU) activation function is used in this study according to Eq. 25.

$$a = f(z) = \max(0, z) \quad (25)$$

The weights and biases of the network are updated by minimising the cost function of Eq. 26. This process is called backpropagation.

$$C_f = \frac{1}{2n} \sum_x \|y(x) - a^L(x)\|^2 \quad (26)$$

L represents the model layers. The cost function is minimised using Adam, a method for stochastic optimisation [29] with the step size (“learning rate”) of 0.001. In this study, the input and the output layers consist of five and one neuron, respectively. The optimal model architecture (number of hidden layers and their neurons) are found by trial and error in order to maximise the performance on the test data and minimise the model complexity. The number of epochs (the process of passing the entire dataset forward and backward through the ANN) and the batch size (the

chunk of data fed to the ANN at each substep) are selected in the same way.

3.4.2 Ensemble techniques

Ensemble techniques combine predictions from multiple models to improve performance. In this study, three classes of ensembles are considered: averaging, bagging and boosting. In averaging, predictions from different models are averaged. A weighted average can also be considered. VR is an example of averaging (Sect. 3.4.2).

Bagging combines the predictions of multiple models. These models are trained on subsets of the dataset (with replacement). This segmentation of the training dataset is called bootstrapping. A base model (weak model) is created on each of these subsets. The models run in parallel and are independent of each other. The final predictions are determined by averaging the predictions of the models.

With boosting, a sequence of models is considered, in which each model corrects the predictions of the previous one. This ensemble technique works as follows. First, a subset of the training data is considered and equal weights are assigned to the data points. Then, a base model is created on this subset, which is used to make predictions on the whole dataset. The error is calculated using the actual values and predicted values, and the observations with higher errors are given higher weights. At this point, another model is created which tries to correct the errors from the previous one. In doing so, several models are created, the final model (“strong learner”) being the weighted mean of all the models (“weak learners”).

RFR (Sect. 3.4.2) and XGB (Sect. 3.4.2) are examples of bagging and boosting, respectively.

Random Forest Regressor RFR is a bagging method that applies several decision trees to the training data. Every decision tree is trained with a different data subset. In doing so, the prediction of every tree is different. The final output is the average prediction.

In this study, the maximum depth of the three (max_depth), the number of features to consider when looking for the best split (max_features) and the number of trees in the forest (n_estimators) are optimised with Grid Search.

Voting Regressor VR can combine different algorithms by averaging their predictions. This ensemble enhances the strength and downsizes the weakness of the individual components. Hereto, we combine DTR, KNR and SVR. The average prediction can be the ordinary arithmetic mean of the predictions or their weighted mean. In this study, the second approach is considered and the weights are optimised via PSO.

Extreme Gradient Boosting XGB was formulated by Friedman [19]. Within the boosting framework, decision

trees are added one at a time to the XGB ensemble and fit to correct the prediction errors made by the prior models by optimising the loss function with a gradient descent algorithm (hence the name “gradient boosting”). Given its computational efficiency and high performance, XGB enjoys a well-established reputation, especially for structured datasets.

In this study, the learning rate of the XGB algorithm is set to 0.01. The least squared error is chosen as the loss function. The maximum depth of the tree (max_depth), the number of features to consider when looking for the best split (max_features) and the number of trees in the forest (n_estimators) are optimised with Grid Search.

3.5 Description of the Monte Carlo simulations

In order to show that the SC techniques can be efficiently employed to infer the failure probability, an example application is described in this section. In fact, by coupling the SC techniques and MC simulations, the failure probability is calculated with a number of data points up to 10^6 . Applications relying on the combination of numerical analysis and MC simulations with comparable sample size would be computationally unfeasible. Instead, the proposed method calculates the failure probability within few minutes even for the largest sample sizes. For this reason, crude MC simulations are carried out instead of more advanced methods such as Latin hypercube sampling or importance sampling.

MC simulations are carried out for a tunnel with parameters according to Table 3. In Table 3, the shear parameters are random variables with given mean value and standard deviation. Their coefficients of variation $COV_c = \sigma_c/\mu_c$ and $COV_\varphi = \sigma_\varphi/\mu_\varphi$ are commonly used values [44]. Both correlated and uncorrelated shear parameters are considered. The coefficient of correlation is considered as $\rho_{c\varphi} = -0.5$. The covariance matrix is calculated with Eq. 27.

$$K = \begin{bmatrix} \sigma_c^2 & \rho_{c\varphi} \cdot \sigma_c \cdot \sigma_\varphi \\ \rho_{c\varphi} \cdot \sigma_c \cdot \sigma_\varphi & \sigma_\varphi^2 \end{bmatrix} \quad (27)$$

Both normally and log-normally distributed samples are considered with variable sample size up to 10^6 . The support pressure is predicted by using the SC technique with the best performance on the test dataset. It is shown in Sect. 4 that the best SC technique is ANN. Based on the mean values of the shear parameters, ANN predicts the support pressure of 32 kPa. The support pressure in the MC simulation is set to an arbitrary higher value of 36 kPa and the corresponding failure probability \widetilde{P}_f is computed. The failure probability is estimated as follows:

1. The support pressure is calculated for every data point in the sample
2. If the calculated support pressure of the data points is higher or equal to 36 kPa, the points are labelled as stable
3. If the calculated support pressure of the data points is lower than 36 kPa, the points are labelled as unstable

The failure probability is calculated as the number of unstable data points divided by the total number of points (Eq. 5). This value varies with the assumed distribution (normal/log-normal) and correlation among the features. Also, in order to achieve a stable failure probability, the sample size must be adequate. Therefore, a sensitivity analysis is performed by varying the sample size from 1 to 10^6 .

4 Results

In the following sections, the results are presented for Soft Computing (Sect. 4.1), Feature Importance (Sect. 4.2) and for the MC simulations (Sect. 4.3).

4.1 Soft computing techniques

In this section, the results for various SC techniques are graphically depicted (Figs. 7, 8, 9, 10, 11, 12, 13, 14, 15). The diagrams are shown in terms of predicted versus measured support pressure. The bisector indicates a perfect match between data and prediction. The closest the points are to the bisector, the better the predictions. The predictions are shown both for the training and for the test data for each SC technique. The coefficient of determination is shown on each graph.

The optimal architecture for ANN is shown in Fig. 14. It consists of two hidden layers, one with three and one with four neurons. The chosen number of epochs is 500 and the batch size is 16.

All performance metrics are resumed in Tables 5 and 6 for classification and regression, respectively. The individual techniques are sorted based on their performance on the test data. LOR performs perfectly for the classification problem for all metrics considered. For this reason, no other technique is assessed for the classification. For regression, the best performance is achieved with ANN, the worst with LIR.

The optimised (hyper)parameters are listed in Table 7.

4.2 Feature importance

Feature importance is shown in Fig. 16 for various individual techniques. In accordance with the exploratory data

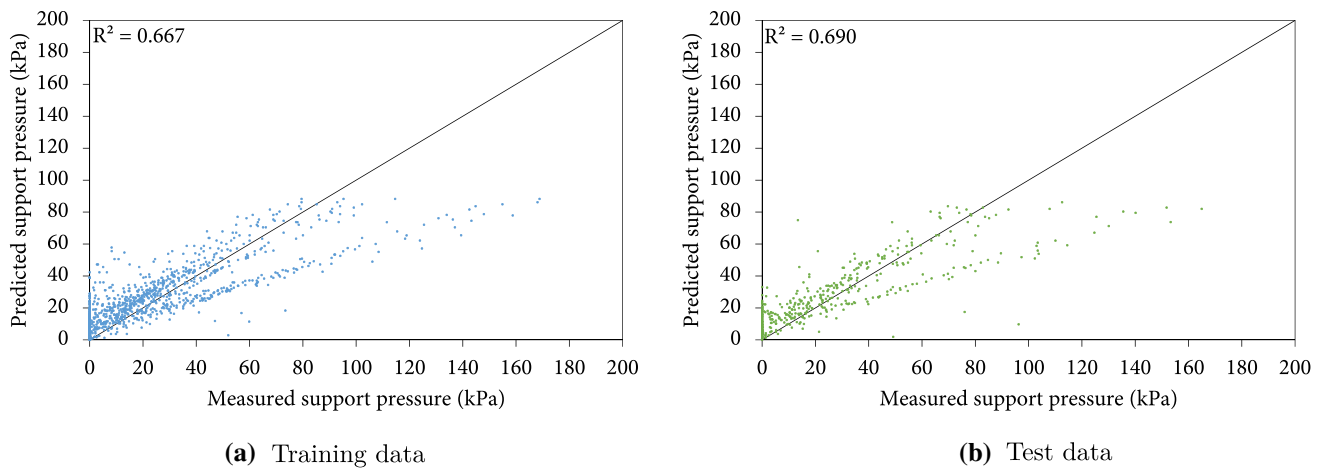


Fig. 7 Linear regressor

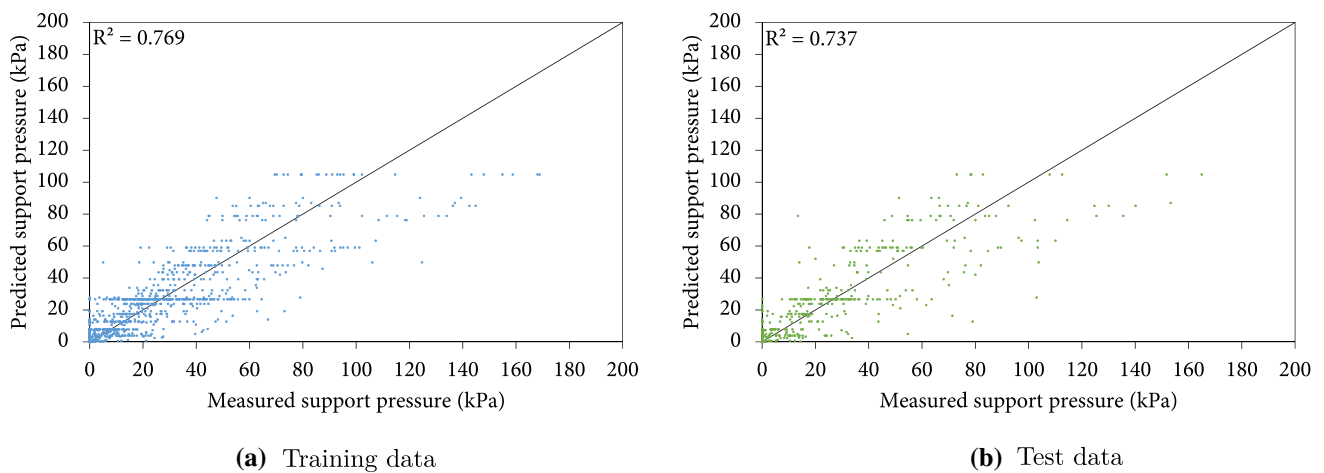


Fig. 8 Decision tree regressor

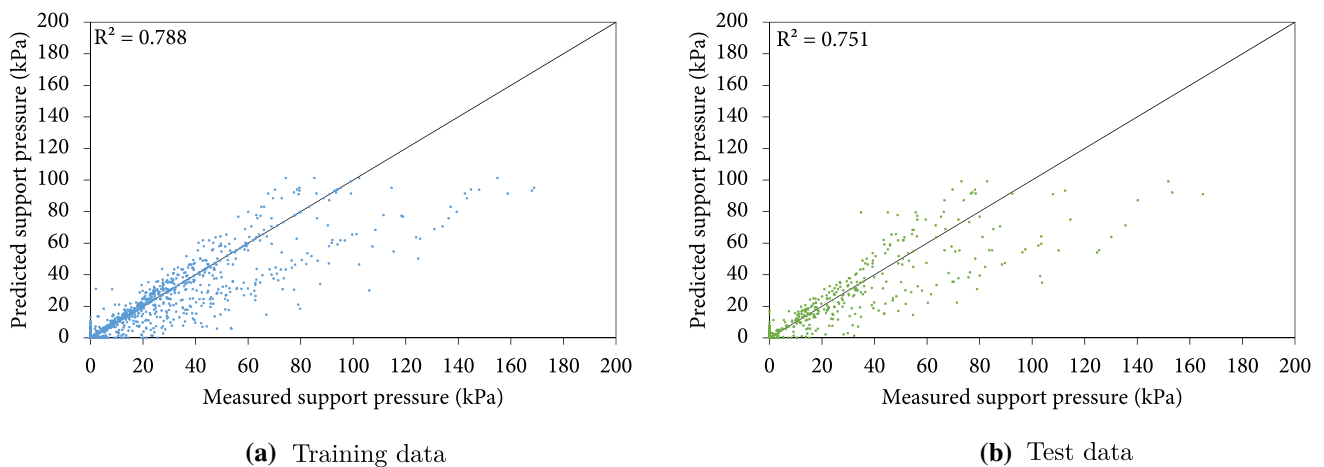


Fig. 9 Support vector regressor

analysis (Sect. 3.1), soil cover, cohesion and diameter have the major impact on the results. For soil cover and diameter, there are some differences in feature importance

among the individual techniques. For instance, the depth of the soil cover and tunnel diameter are more important for LIR and DTR, respectively, than for the other techniques.

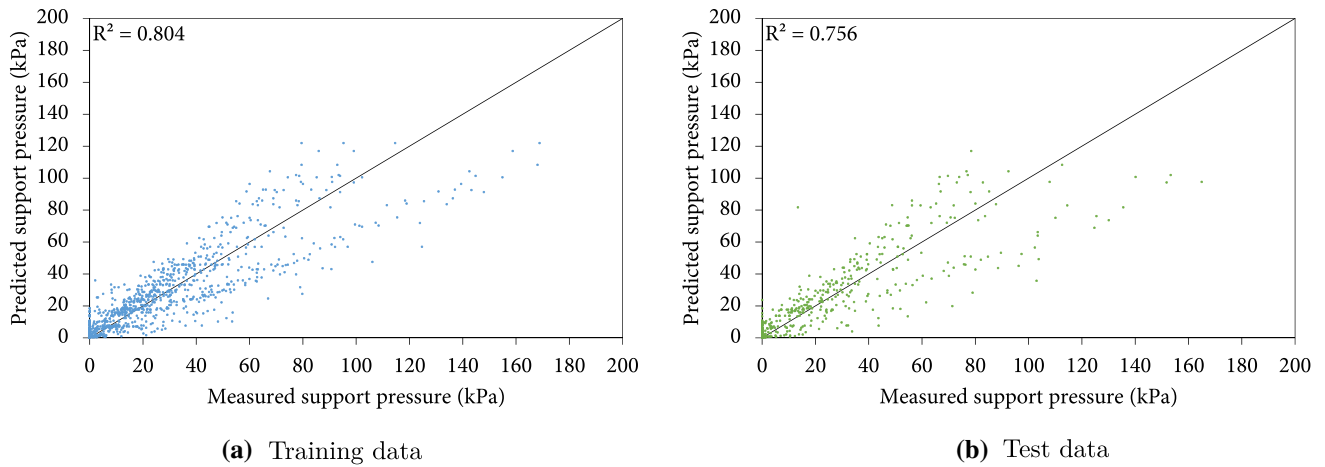


Fig. 10 Random forest regressor

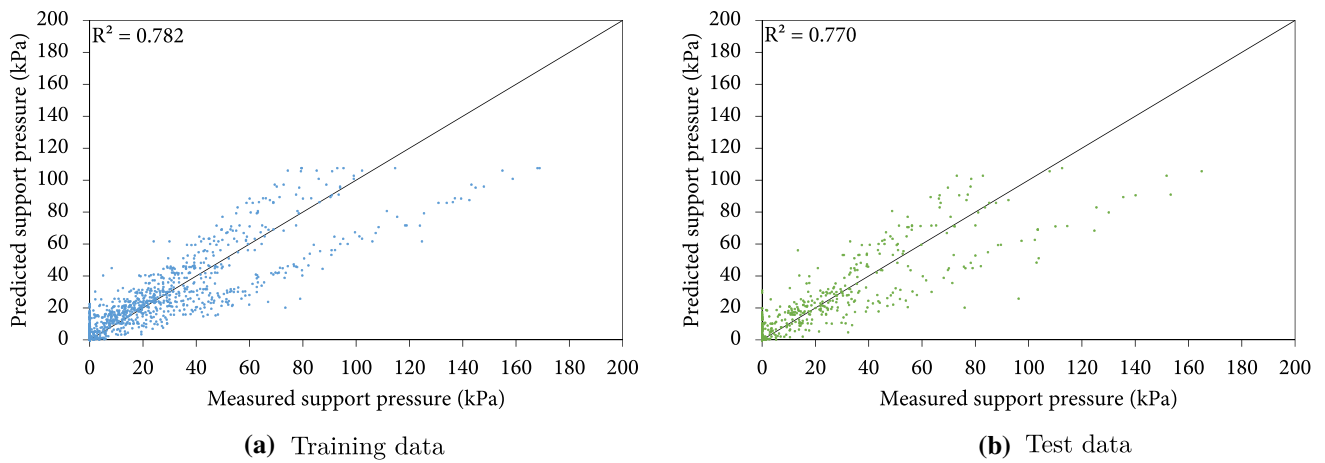


Fig. 11 K-Nearest regressor

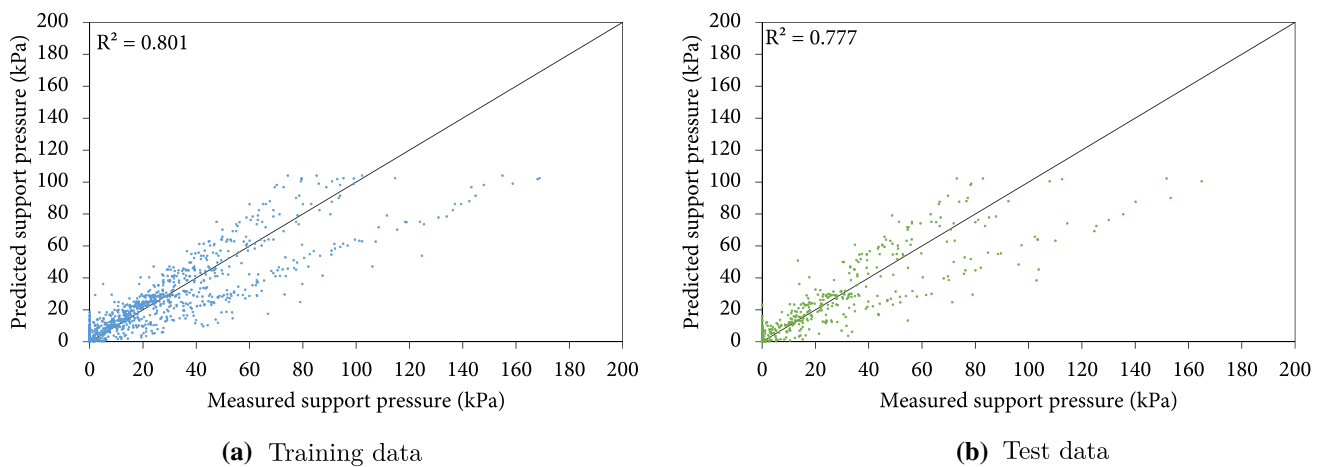


Fig. 12 Voting regressor

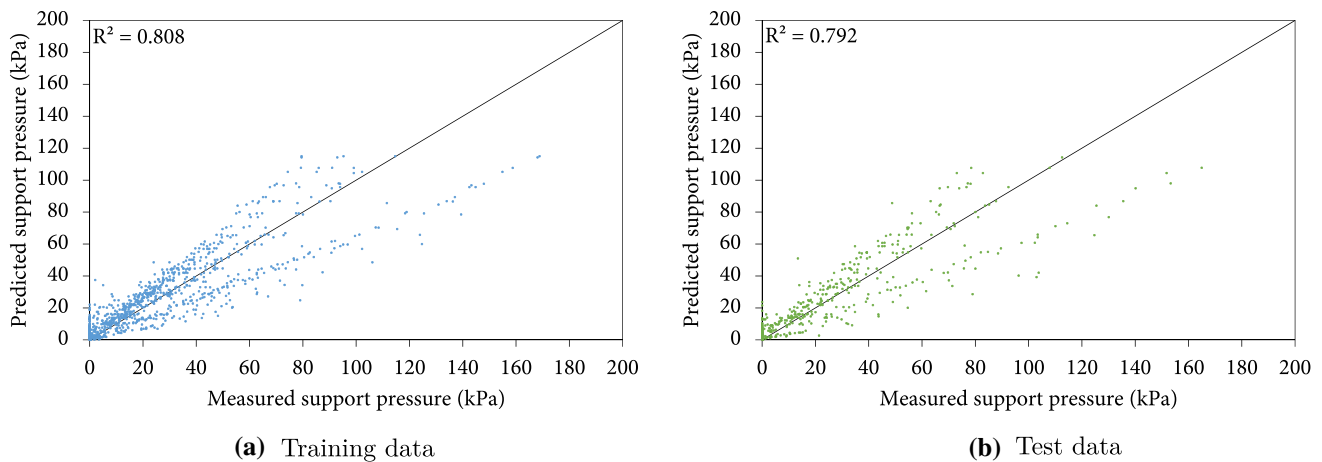


Fig. 13 Extreme gradient boosting regressor

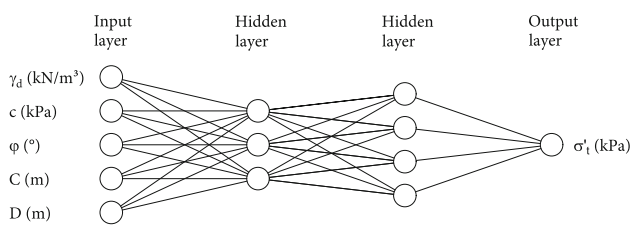


Fig. 14 Chosen architecture for the artificial neural network

In general, however, the feature importance is generally consistent among different techniques.

4.3 Results of the Monte Carlo simulations

Figures 17 and 18 show the results of the MC simulations performed with 10^4 data points for the correlated and uncorrelated samples. The stable and unstable points are shown in green and red, respectively. The Limit State Line is implicitly obtained as the ideal boundary separating the stable and unstable points. The probability of failure is calculated based on the number of unstable points divided by the total. An accurate estimation of the failure probability calculation requires that the sample be large enough. Therefore, the sensitivity analysis of Fig. 19 is carried out.

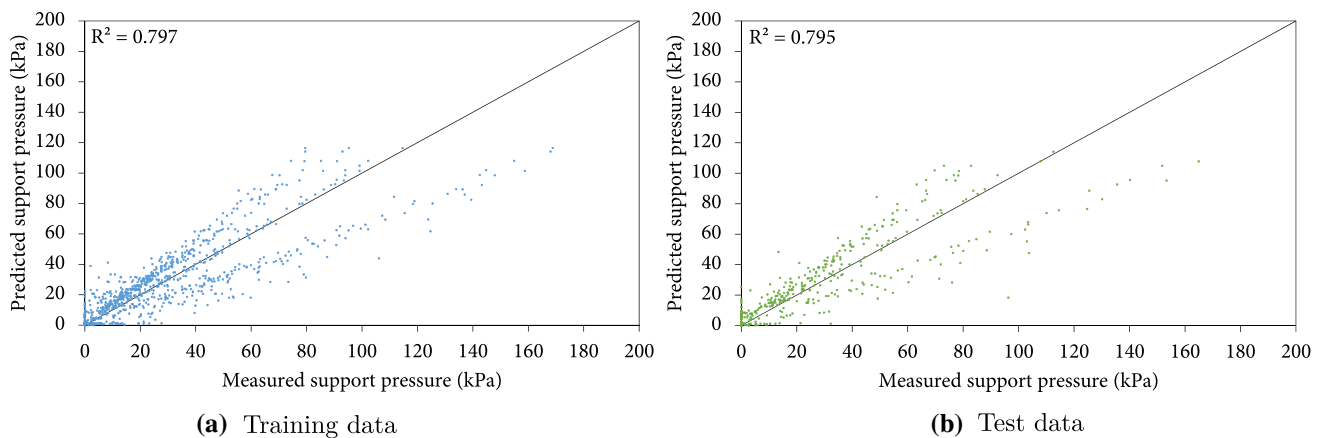


Fig. 15 Artificial neural network, results

Table 5 Overview of the results for classification

Classifier	Training dataset				Test dataset			
	Precision	Recall	F ₁ -Score	Accuracy	Precision	Recall	F ₁ -Score	Accuracy
LOR	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Performance metrics

Table 6 Overview of the results for regression

Regressor	Training dataset			Test dataset		
	R ²	RMSE	MAE	R ²	RMSE	MAE
LIR	0.667	16.142	11.659	0.689	16.540	11.391
DTR	0.769	13.429	8.400	0.736	15.221	9.609
SVR	0.788	12.862	6.088	0.751	14.781	7.845
RFR	0.804	12.393	7.683	0.756	14.653	8.913
KNR	0.782	13.060	8.141	0.770	14.212	8.813
VR	0.801	12.542	7.427	0.777	13.915	8.380
XGB	0.808	12.243	7.730	0.792	13.528	8.342
ANN	0.797	12.608	7.695	0.795	13.425	8.027

Performance metrics

Table 7 Overview of the results

Classifier	Best parameters
LOR	$\beta_0 = 1.589, \beta_1 = -2.271, \beta_2 = -1.164, \beta_3 = -1.881, \beta_4 = -0.840, \beta_5 = -1.310$
Regressor	Best parameters
LIR	$\beta_0 = 21.110, \beta_1 = 0.0093, \beta_2 = -13.738, \beta_3 = -4.252, \beta_4 = 18.002, \beta_5 = 2.572$
Regressor	Best hyperparameters
DTR	max_depth = 6, max_features = 3
KNR	n_neighbours = 17
SVR	C = 293.349, $\gamma = 0.567$
RFR	max_depth = 5, max_features = 4, n_estimators = 9
VR	$w_1 = 0.658, w_2 = 1.629, w_3 = 0.713$
XGB	learning_rate = 0.01, n_estimators = 900, max_depth = 3, max_features = 1

Chosen parameters and hyperparameters of all the SC techniques

The four cases of Figs. 17 and 18 with sample size ranging from 1 to 10⁶ are considered. The probability of failure converges to a stable solution with sample sizes of 2 × 10⁴ and 5 × 10³ for the normally and log-normally distributed samples, respectively. With a sample size of 10⁶, the probabilities of failure for the normally distributed sample are 0.69% and 0.82% for the correlated and uncorrelated cases, respectively. For the log-normally distributed samples, these values drop to 0.26% and 0.34%, respectively.

5 Discussion

Abundant data for the assessment of face stability of tunnels are available in the literature. Most features vary in a range close to the one encountered in nature, although the dry soil unit weight is concentrated around its mean value (Table 4). This is due to the fact that the numerical study of [28] keeps the value of this feature constant while varying the others. However, based on the calculated feature importances (Sect. 4.2), the dry soil unit weight is the least significant feature (Fig. 16). The effective support pressure shows a smooth probability distribution in the typical range applied to tunnels excavated in soil (Fig. 5). The dataset shows a negative correlation between the cohesion and friction angle of $\rho_{c\phi} = -0.21$ (Fig. 6), which complies with the values found in the literature [27]. The positive and negative correlations—shown by the soil cover and tunnel diameter on the one hand, and by dry soil unit weight, cohesion and friction angle on the other hand—with the effective support pressure are also supported by previous findings [7, 58].

Classification, performed with simple Logistic Regression, delivers statistically impeccable results (Table 5).

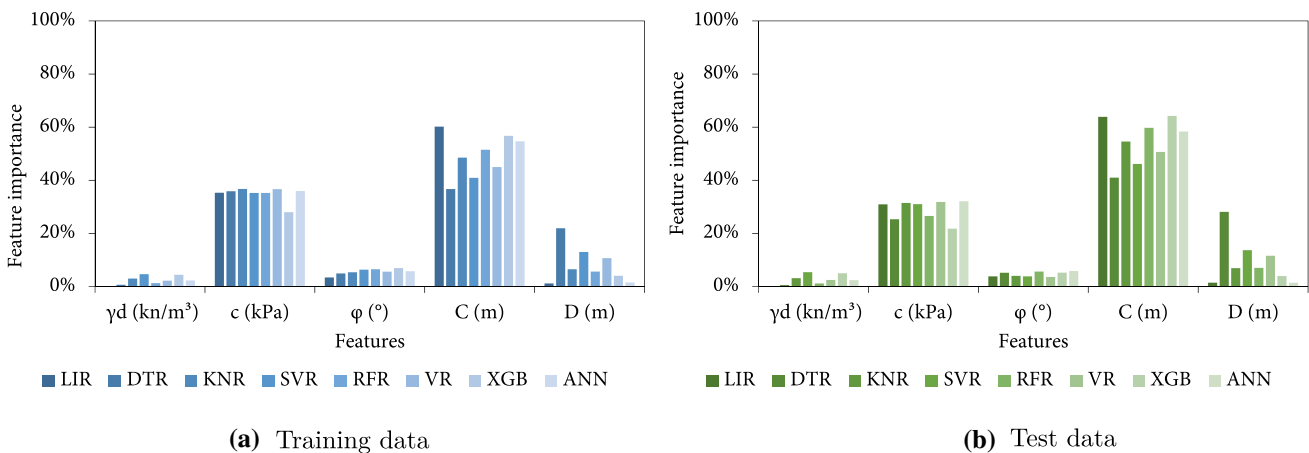


Fig. 16 Feature importance

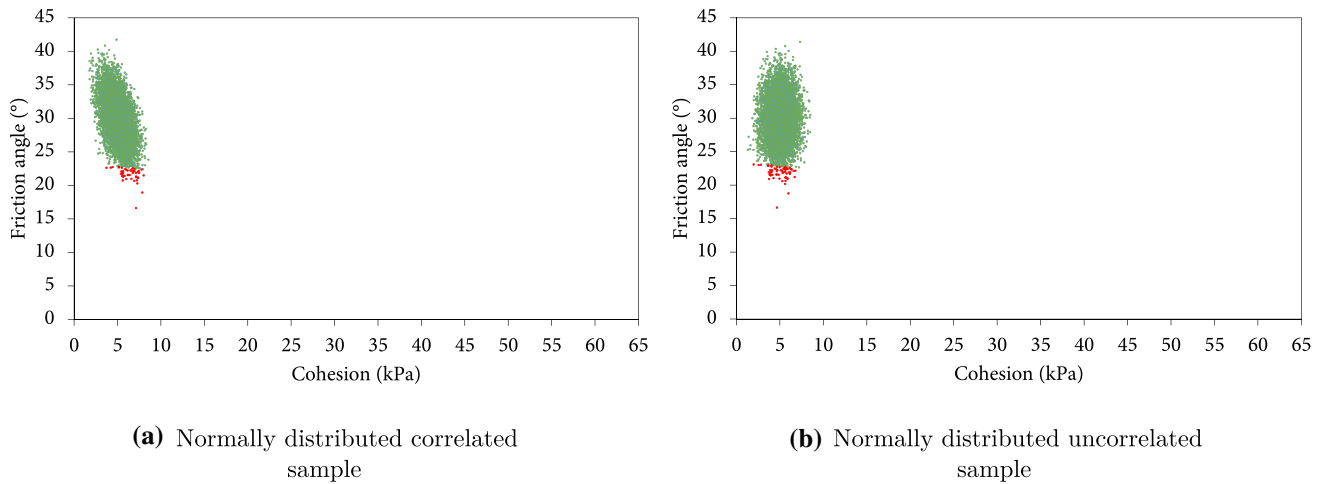


Fig. 17 Stable and unstable experimental points for a normally distributed sample

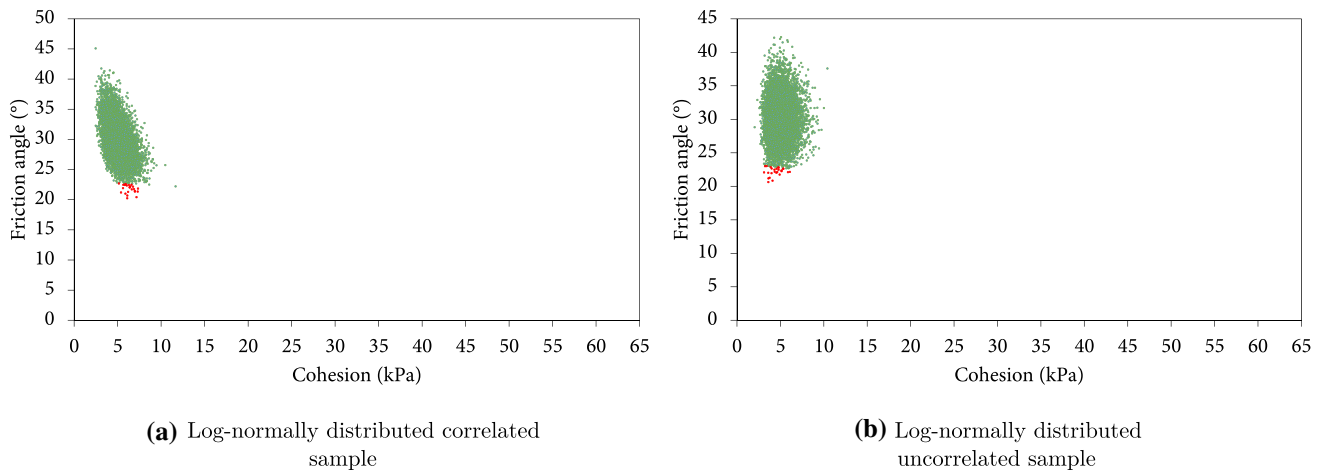


Fig. 18 Stable and unstable experimental points for a log-normally distributed sample

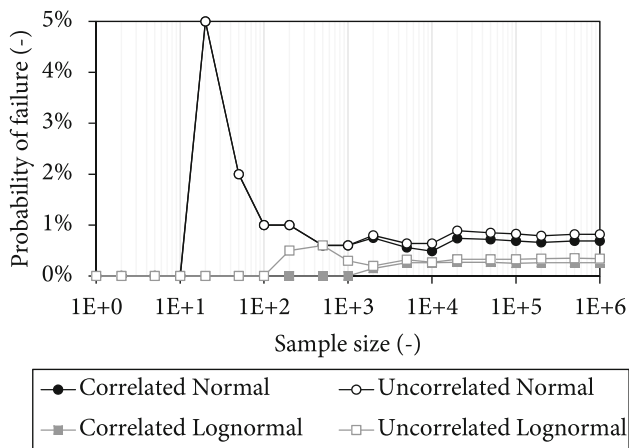


Fig. 19 Failure probability versus sample size

Given its outstanding results, the authors need not apply any more sophisticated method.

As for the regression, the model performance generally increases alongside complexity: LIR and ANN show the lowest and the highest performance, respectively (Table 6). As expected, LIR underperforms the nonparametric methods. The ensemble methods (RFR, VR and XGB) outperform individual SC techniques, the exception being KNR which outperforms RFR on the test dataset. Finally, the ANN outperforms all the other methods. All in all, given the limited number of features considered and the fact that all the available data were retained, the achieved performance in the order of magnitude of $R^2 = 0.80$ is more than satisfactory.

Monte Carlo simulations coupled with SC techniques are a convenient way to obtain the LSL implicitly, as shown in Fig. 17 and 18. The failure probability depends on the sample distribution adopted, the correlation between the features and the sample size. The normally distributed samples yield a higher failure probability than the log-normally distributed ones. Also, the uncorrelated samples

yield a higher failure probability than the negative correlated ones. These results confirm the findings of [44]. The sample size plays a pivotal role. As shown in Fig. 19 for the normally distributed sample, the probability of failure skyrockets from zero to a considerable value with the sample size increasing from 1 to 20. Then, it decreases and finally converges to a much lower value. For the log-normally distributed sample, this sudden increase occurs with the sample size equal to 200 and is much less sharp. Therefore, care must be taken in the choice of the sample size for these simulations.

6 Conclusions

Traditional approaches to tunnel face stability include analytical methods, numerical simulations and physical modelling. Based on the results from this body of knowledge, this study takes a different route, i.e. a soft computing approach. The prerequisite of this approach is data abundance: 658 peer-reviewed publications were found in the literature at the time of this study. The authors narrowed down this body of knowledge to 21 papers from which 1829 data points were retrieved. These studies are multifaceted, covering numerical analysis, physical modelling (1 g and in the centrifuge) and monitoring data.

The problem of face stability is framed both as classification (is the tunnel face stable?) and regression (what support pressure is needed?). The classification problem is efficiently solved with the Logistic Regression. The regression problem is more intricate. The nonparametric regression techniques are found to outperform parametric ones; ensemble learning outperforms the individual SC techniques. The artificial neural network shows the best performance with $R^2 = 0.795$ on the test data.

A viable and quick method to determine the probability of failure is established with Monte Carlo simulations and the effect of sample distribution, size and features correlation is shown.

With this study, the authors are confident that the SC techniques have demonstrated their applicability to the tunnel face stability problem and to the calculation of the probability of failure.

Acknowledgements Financial support for this research is provided by the Otto Pregl Foundation for Geotechnical Fundamental Research.

Funding Open access funding provided by University of Natural Resources and Life Sciences Vienna (BOKU).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate

if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Adoko AC, Jiao YY, Wu L, Wang H, Wang ZH (2013) Predicting tunnel convergence using multivariate adaptive regression spline and artificial neural network. *Tunn Undergr Sp Tech* 38:368–376
- Ahmed M, Iskander M (2012) Evaluation of tunnel face stability by transparent soil models. *Tunn Undergr Sp Tech* 27(1):101–110
- Alagha ASN, Chapman DN (2019) Numerical modelling of tunnel face stability in homogeneous and layered soft ground. *Tunn Undergr Sp Tech* 94:1–14
- Anagnostou G (2012) The contribution of horizontal arching to tunnel face stability. *Geotechnik* 35(1):34–44
- Anagnostou G, Kovári K (1994) The face stability of slurry-shield-driven tunnels. *Tunn Undergr Sp Tech* 9(2):165–174
- Anagnostou G, Kovári K (1996) Face stability conditions with earth-pressure-balanced shields. *Tunn Undergr Sp Tech* 11(2):165–173
- Augarde CE, Lyamin AV, Sloan SW (2003) Stability of an undrained plane strain heading revisited. *Comput Geotech* 30(5):419–430
- Benardos AG, Kaliampakos DC (2004) Modelling TBM performance with artificial neural networks. *Tunn Undergr Sp Tech* 19(6):597–605
- CEN (European Committee for Standardization) (2002) Eurocode—basis of structural design. European Committee for Standardization, Brussels
- Chambon P, Corté JF (1994) Shallow tunnels in cohesionless soil: stability of tunnel face. *J Geotech Eng-ASCE* 120(7):1148–1165
- Chen R, Tang L, Yin XS, Chen YM, Bian XC (2015) An improved 3D wedge-prism model for the face stability analysis of the shield tunnel in cohesionless soils. *Acta Geotech* 10:683–692
- Chen R, Li J, Kong L, Tang L (2013) Experimental study on face instability of shield tunnel in sand. *Tunn Undergr Sp Tech* 33:12–21
- Chen R, Tang LJ, Ling D, Chen Y (2011) Face stability analysis of shallow shield tunnels in dry sandy ground using the discrete element method. *Comput Geotech* 38:187–195
- DAUB (Deutscher Ausschuss für unterirdisches Bauen) (2016) Recommendations for face support pressure calculations for shield tunnelling in soft ground. Deutscher Ausschuss für unterirdisches Bauen, Köln
- Davis EH, Gunn MJ, Mair RJ, Seneviratne HN (1980) The stability of shallow tunnels and underground openings in cohesive material. *Géotechnique* 30(4):397–416
- Ditlevsen O (1981) Uncertainty modelling: with applications to multidimensional civil engineering systems. McGraw-Hill, New York
- Ebid AM (2020) 35 years of (AI) in geotechnical engineering: state of the art. *Geotech Geol Eng*
- Eshraghi A, Zare S (2015) Face stability evaluation of a TBM-driven tunnel in heterogeneous soil using a probabilistic approach. *Int J Geomech* 15(6):1–10

19. Friedman JH (2001) Greedy function approximation: a gradient boosting machine. *Ann Stat* 29(5):1189–1232
20. Goh ATC, Kulhawy FH (2003) Neural network approach to model the limit state surface for reliability analysis. *Can Geotech J* 40(6):1235–1244
21. Hasofer AM, Lind NC (1974) Exact and invariant second moment code format. *J Eng Mech Div-ASCE* 100(1):111–121
22. Horn N (1961) Horizontal ground pressure on vertical faces of tunnel tubes. In: *Landeskonferenz der Ungarischen Tiefbauindustrie*, Budapest
23. Hu X, Zhang Z, Kieffer S (2012) A real-life stability model for a large shield-driven tunnel in heterogeneous soft soils. *Front Struct Civ Eng* 6(2):176–187
24. Idinger G, Aklik P, Wu W, Borja RI (2011) Centrifuge model test on the face stability of shallow tunnel. *Acta Geotech* 6:105–117
25. Jancsecz S, Steiner W (1994) Face support for a large mix-shield in heterogeneous ground conditions. In: *Tunnelling'94*. Springer, Boston
26. Jang H, Topal E (2013) Optimizing overbreak prediction based on geological parameters comparing multiple regression analysis and artificial neural network. *Tunn Undergr Sp Tech* 38:161–169
27. Javankhoshdel S, Bathrust RJ (2015) Influence of cross correlation between soil parameters on probability of failure of simple cohesive and $c-\phi$ slopes. *Can Geotech J* 53(5):839–853
28. Kim SH, Tonon F (2010) Face stability and required support pressure for TBM driven tunnels with ideal face membrane-Drained case. *Tunn Undergr Sp Tech* 25(5):526–542
29. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: *Proceedings of the international conference on learning representations*, San Diego, USA
30. Kirsch A (2010) Experimental investigation of the face stability of shallow tunnels in sand. *Acta Geotech* 5:43–62
31. Lai J, Qiu J, Feng Z, Chen J, Fan H (2016) Prediction of soil deformation in tunnelling using artificial neural networks. *Comput Intel Neurosc* 1–16
32. Leca E, Dormieux L (1990) Upper and lower bound solutions for the face stability of shallow circular tunnels in frictional material. *Géotechnique* 40(4):581–606
33. Leu S, Chen C, Chang S (2001) Data mining for tunnel support stability: neural network approach. *Automat Constr* 10(4):429–441
34. Li F, Liu W, Tang X, Gan P (2014) Numerical research on the influence of stress release on tunnel face stability. In: *Proceedings of the international conference on pipelines and trenchless technology*, Xiamen, China, pp 799–807
35. Liu H, Zhang Y, Liu H (2020) Failure mechanism of face for slurry shield-driven tunnel in sand. *KSCE J Civ Eng* 24(10):3105–3118
36. Low B, Tang WH (1997) Efficient reliability evaluation using spreadsheet. *J Eng Mech-ASCE* 123:749–752
37. Lu X, Zeng S, Zhao Y, Huang M, Ma S, Zhang Z (2020) Physical model tests and discrete element simulation of shield tunnel face stability in anisotropic granular media. *Acta Geotech* 15:3017–3026
38. Lü X, Zhou Y, Huang M, Zeng S (2018) Experimental study of the face stability of shield tunnel in sands under seepage condition. *Tunn Undergr Sp Tech* 74:195–205
39. Lü X, Su Z, Huang M, Zhou Y (2017) Strength reduction finite element analysis of a stability of large cross-river shield tunnel face with seepage. *Eur J Environ Civ Eng* 24(3):336–353
40. Mair R (1979) Centrifugal modelling of tunnel construction in soft clay. PhD Thesis, University of Cambridge, Cambridge
41. Messerli J, Pimentel E, Anagnostou G (2010) Experimental study into tunnel face collapse in sand. In: *Proceedings of the 7th international conference on physical modelling in geotechnics*, Zürich, Switzerland
42. Mollon G, Dias D, Soubra A (2011) Probabilistic analysis of pressurized tunnels against face stability using collocation-based stochastic response surface method. *J Geotech Geoenviron* 137(4):385–397
43. Mollon G, Dias D, Soubra A (2011) Rotational failure mechanisms for the face stability analysis of tunnels driven by a pressurized shield. *Int J Numer Anal Met* 35(12):1363–1388
44. Mollon G, Dias D, Soubra A (2009) Probabilistic analysis and design of circular tunnels against face stability. *Int J Geomech* 9(6):237–249
45. Mahdevari S, Shahriar K, Yagiz S, Shirazi MA (2014) A support vector regression model for predicting tunnel boring machine penetration rates. *Int J Rock Mech Min* 72:214–229
46. Mahdevari S, Haghghat HS, Torabi SR (2013) A dynamically approach based on SVM algorithm for prediction of tunnel convergence during excavation. *Tunn Undergr Sp Tech* 38:59–68
47. Miranda LJV (2017) PySwarms: a research toolkit for Particle Swarm Optimization in Python
48. Naghadehi MZ, Thewes M, Alimardani LA (2019) Face stability analysis of mechanized shield tunneling: an objective systems approach to the problem. *Eng Geol* 262:1–15
49. Pan Q, Dias D (2017) An efficient reliability method combining adaptive support vector machine and Monte Carlo simulation. *Struct Saf* 67:85–95
50. Pan Q, Dias D (2017) Probabilistic evaluation of tunnel face stability in spatially random soils using sparse polynomial chaos expansion with global sensitivity analysis. *Acta Geotech* 12:1415–1429
51. Pedregosa F, Gramfort Varoquaux GA, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in python. *J Mach Learn Res* 12(85):2825–2830
52. Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev* 65(6):386–408
53. Soranzo E, Wu W (2013) Centrifuge test of face stability of shallow tunnels in unsaturated soil. In: *Proceedings of the 5th biot conference on poromechanics*, Vienna, Austria
54. Soranzo E, Tamagnini R, Wu W (2015) Face stability of shallow tunnels in partially saturated soil: centrifuge testing and numerical analysis. *Géotechnique* 65(6):454–467
55. Sterpi D, Cividini A (2004) A physical and numerical investigation on the stability of shallow tunnels in strain softening media. *Rock Mech Rock Eng* 37(4):277–298
56. Tang X, Liu W, Albers B, Savidis S (2014) Upper bound analysis of tunnel face stability in layered soils. *Acta Geotech* 9:661–671
57. Ukritchon B, Keawsawasvong S, Yingchaloenkithajorn K (2017) Undrained face stability of tunnels in Bangkok subsoils. *Int J Geotech Eng* 11(3):1–16
58. Vermeer PA, Ruse N, Marcher T (2002) Tunnel heading stability in drained ground. *Felsbau* 20(6):1–17
59. Wang G (2019) Study on Face stability analysis of the aeolian sand tunnel. In: *Proceedings of the international conference on civil, architecture and disaster prevention*, Hefei, China
60. Wang J, He C, Xu G (2019) Face stability analysis of EPB shield tunnels in dry granular soils considering dynamic excavation process. *J Geotech Geoenviron* 145(11):1–10
61. Zhang C, Han K, Zhang D (2015) Face stability analysis of shallow circular tunnels in cohesive-frictional soils. *Tunn Undergr Sp Tech* 50:345–357
62. Zhang L, Wu X, Wenying J, AbouRizk SM (2016) Intelligent approach to estimation of tunnel-induced ground settlement using wavelet packet and support vector machines. *J Comput Civil Eng* 31(2):1–14

63. Zhang P, Chen R, Wu H (2019) Real-time analysis and regulation of EPB shield steering using random forest. *Automat Constr* 106:1–13
64. Zhang W, Li H, Li Y, Liu H, Chen Y, Ding X (2021) Application of deep learning algorithms in geotechnical engineering: a short critical review. *Artif Intell Rev* (**in press**)
65. Zhang W, Li Y, Wu C, Li H, Goh ATC, Liu H (2021) Prediction of lining response for twin tunnels constructed in anisotropic clay using machine learning techniques. *Underground Space* (**in press**)
66. Zhang W, Zhang R, Wu C, Goh ATC, Lacasse S, Liu Z, Liu H (2020) State-of-the-art review of soft computing applications in underground excavations. *Geosci Front* 11(4):1095–1106
67. Zhang W, Zhang R, Wu C, Li YQ, Liu Z, Liu H (2019) Soft computing approach for prediction of surface settlement induced by earth pressure balance shield tunneling. *Underground Space* (**in press**)
68. Zhang Z, Liu C, Huang X (2017) Numerical analysis of volume loss caused by tunnel face instability in soft soils. *Environ Earth Sci* 76:1–19
69. Zhang Z, Hu X, Scott KD (2011) A discrete numerical approach for modeling face stability in slurry shield tunnelling in soft soils. *Comput Geotech* 38(1):94–104
70. Zhou J, Xiuzhi S, Du K, Qiu X, Li X, Mitri HS (2017) Feasibility of random-forest approach for prediction of ground settlements induced by the construction of a shield-driven tunnel. *Int J Geomech* 17(6):1–12

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.