



# Theory-guided machine learning to predict density evolution of sand dynamically compacted under $K_0$ condition

Amir Tophel<sup>1</sup> · Jeffrey P. Walker<sup>2</sup> · Troyee Tanu Dutta<sup>1</sup> · Jayantha Kodikara<sup>1</sup>

Received: 1 August 2021 / Accepted: 28 November 2021

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

## Abstract

This paper introduces a theory-guided machine learning (TGML) framework, which combines a theoretical model (TM) and a machine learning (ML) algorithm to predict compaction density under cyclic loading. Several 1-D tests were conducted on uniformly graded fine sand compacted at varying moisture contents ( $w$ ), stress levels ( $\sigma_z$ ) and loading frequencies ( $f$ ), simulating the field compaction of materials using a vibratory roller. The laboratory compaction data were first analysed using a revised TM and an artificial neural network (ANN), and their performance was measured using mean absolute error (MAE). Next, the data were analysed using the TGML framework, which involves three different techniques. TGML1 increased the ML's ability to extrapolate (MAE improved from  $2.2 \times 10^{-3}$  to  $1.2 \times 10^{-3}$ ); TGML2 ensured ML and TM complemented each other to model observations better (MAE improved from  $2.3 \times 10^{-3}$  to  $7.9 \times 10^{-4}$ ); and TGML3 assisted in regularising the ML with an additional loss function which ensured the model followed the mechanistic understandings of the underlying physics (MAE improved from  $9.2 \times 10^{-3}$  to  $2.7 \times 10^{-3}$ ). Considering TGML3 during modelling is essential when dealing with noisy field datasets, and this is the highlight of this paper. TGML frameworks showed less error and lower model uncertainty, estimated using the novel Monte Carlo dropout technique. Furthermore, the developed TGML framework was used to demonstrate a termination criterion, i.e. the number of cycles of roller movement required to achieve the desired degree of compaction. Finally, an approach is proposed by which a simplified TM and ML model can estimate field compaction behaviour during roller movement.

**Keywords** Compaction · Density · Monte Carlo dropout · Termination criteria · Theory-guided machine learning · Uncertainty

## List of symbols

$F$	Activation function
$Y$	Actual output or observation
ANN	Artificial neural network
$\epsilon^p$	Axial or volumetric plastic strain

$b_i$	Bias matrix
$C_C$	Coefficient of curvature
$K_0$	Coefficient of lateral pressure
$C_u$	Coefficient of uniformity
DL	Deep learning
FS	Fine sand
$f$	Frequency of vibration
$\mathcal{N}$	Gaussian noise
$z_i$	Hidden layer output
$\lambda_{reg}$	Hyperparameter for L2 regularization
$N_{initial}$	Initial guess of no. of cycles in termination criteria
$e_0$	Initial void ratio
$X$	Input matrix
GPR	Gaussian process regression
IC	Intelligent compaction
ML	Machine learning
$Y_{ML}$	Machine learning prediction
MDD	Maximum dry density

✉ Jayantha Kodikara  
jayantha.kodikara@monash.edu

Amir Tophel  
amir.tophel@monash.edu

Jeffrey P. Walker  
jeff.walker@monash.edu

Troyee Tanu Dutta  
troyee.dutta@monash.edu

<sup>1</sup> ARC Industrial Transformation Research Hub (ITRH) – SPARC Hub, Department of Civil Engineering, Monash University, Clayton Campus, Clayton, VIC 3800, Australia

<sup>2</sup> Department of Civil Engineering, Monash University, Clayton Campus, Clayton, VIC 3800, Australia

$\sigma_z$	Maximum vertical stress
MAE	Mean absolute error
$\mu$	Mean of the noise
$D_{50}$	Median diameter
$C_1, C_2$ and $m$	Model parameters
$w$	Moisture content
MLP	Multi-layer perceptron
$Y_{Noisy}$	Noisy observations
$\frac{e_0 - e_{30}}{e_0}$	Normalized relative change in void ratio
NDG	Nuclear density gauge
$N$	Number of cycles
$\eta$	Optimization rate
$f_{opt}$	Optimum compaction frequency
$S_{ropt}$	Optimum degree of saturation
OMC	Optimum moisture content
$U$	Output after applying activation function
$Y$	Output matrix
$\hat{Y}$	Prediction of ANN
$e_{proctor}$	Proctor void ratio
QA	Quality assurance
QC	Quality control
$\sigma_x$	Radial stress
RF	Random forest
ReLU	Rectified linear unit
RNN	Recurrent neural network
$e_0 - e_{30}$	Relative change in void ratio
$G_s$	Specific gravity
$\sigma_s$	Static stress
SGD	Stochastic gradient descent
SVM	Support vector machine
SVR	Support vector regression
$N_{target}$	Target number of passes
$e_{target}$	Target void ratio
TM	Theoretical model
$Y_{TM}$	Theoretical model prediction
TGML	Theory-guided machine learning
$T_V$	Duration for vibratory load
$s^2$	Variance of the noise
$\sigma_v$	Vibratory stress
$e$	Void ratio
$e_1$	Void ratio after the first cycle
$e_N$	Void ratio at $N$ th cycle
$W_i$	Weight matrix

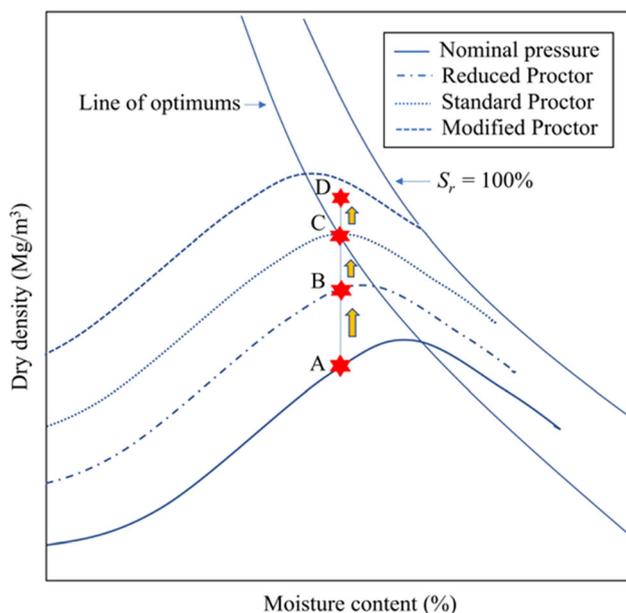
## 1 Introduction

Compaction of materials (soils and unbound granular materials) in the field is required to ensure satisfactory performance under external factors such as repeated traffic loads and environmental effects. Density is an indicator of the degree of compaction, and is commonly characterised with respect to the maximum dry density (MDD) determined at the optimum moisture content (OMC) using Proctor compaction in the laboratory. Achievement of the designated dry density (DD) and OMC in the field is crucial, as under-compacted or over-compacted materials lead to premature failure and/or undesirable permanent deformation or rutting. Various laboratory studies have shown that in general, greater material density results in better resistance to rutting, thus enhancing the service life of pavements [30, 31, 2]. Current density measurement techniques (nuclear density gauge (NDG), sand cone tests and gravimetric tests based on field sampling) can only measure density levels after compaction is complete; moreover, these measurements are confined to a discrete location and are time-consuming. It has been reported that compliance with field specifications following these density measurements either unacceptably delays construction or becomes practically difficult to accomplish [29, 34]. Conversely, there is a significant push to adopt continuous compaction control (CCC) as in intelligent compaction (IC), but it is limited by the inability to estimate geomaterial density proximally. Hence, a genuine need exists to develop methodologies to predict density during compaction using appropriate numerical modelling.

The compaction of material in the field is an example of cyclic loading and unloading because of the movement of road compactors. Modelling the response of unsaturated materials to complex cyclic loading at laboratory scale has been attempted by various researchers using sophisticated analytical and finite element formulations [9, 37, 41–43, 69]. These models may capture complex behaviour well, but require sophisticated and time-consuming tests to determine the model parameters. Generally, the behaviour of unsaturated material is more complicated to model than that of its saturated counterpart because of the interaction among the three medium phases, i.e. soil, water, and air. This drawback limits the use of numerical models for field applications, especially for the real-time prediction of material behaviour. In the field, compaction using a roller

involves large deformation with complex nonlinear elastic–plastic behaviour, limiting the development of an adequate theoretical solution [71]. A close examination of the compaction process in the field allows several assumptions to be made, simplifying the modelling process. First, the compaction of the material is commonly performed at a designated moisture content, which is known a priori. The fact that the material is normally compacted at a moisture content usually at  $\pm 2\%$  of OMC simplifies model development [24]. In addition, the 3-D response of the material under field conditions may be approximated using a 1-D model discussed in detail in the Discussion section.

The material compaction response is crucial to calculate the number of passes of the roller required to achieve a target density. The problem of estimating the number of passes has been well-identified, and this estimation is commonly undertaken by trial and error. Figure 1 depicts the compaction process in the field or laboratory at a target moisture content, where the density increases (A, B, C, D) as the energy input to the soil increases, either by increased number of roller passes or blows of the Proctor hammer. It can be considered that the material is initially at a low-density state (point A) corresponding to a nominal pressure. As the energy input increases with the number of cycles ( $N$ ) of compaction, the material state moves from point A to B to C, where the material reaches the line of optimums (LOO) corresponding to the optimum degree of saturation ( $S_{r,opt}$ ). On the basis of experimental evidence, Tatsuoka and Gomes Correia [60] highlighted that LOO is mostly unique for a certain soil, regardless of the mode of



**Fig. 1** Compaction process illustrated with a family of Proctor curves. The applied energy decreases in the following order: modified Proctor, standard Proctor, reduced Proctor, and nominal pressure

compaction, whether by Proctor hammer or by field rollers. Kodikara [23] and Kodikara et al. [25] highlighted the significance of  $S_{r,opt}$ , when the air phase is trapped in a relatively continuous water phase. Hence, attempts to compact beyond this density (i.e. D) cause the material state to go to the wet side of the LOO, which can have undesirable effects of “over-compaction”, such as heaving of the material with multiple shear planes and loosening of the already compacted material due to chaotic motions of the roller [4, 33]. Furthermore, experimental evidence indicates that material compacted to the wet side of LOO generally undergoes undesirable plastic deformation under repetitive loading similar to that expected from traffic loading [6, 30, 38]. It therefore follows that knowledge of the evolution of the density, and hence the degree of saturation, is beneficial for the development of an effective termination criterion during field compaction. This aspect is also addressed in the present paper.

Based on the assumptions noted above, this paper attempts to simulate the compaction process using a simplified theoretical model (TM) based on the literature. The compaction behaviour of the material with respect to the degree of saturation is also highlighted using the simplified model. The lack of an appropriate constitutive model and a generalised TM necessitates the development of a data-driven machine learning (ML) model, as explored in this study. Further, the fusion of TM and ML models is also investigated. This paper shows that the fusion of TM and ML is a better model for noisy data which may come from data collected from the field because of uncertainties involved with testing, measurement, limitations of the equipment used, and human error.

## 2 Machine learning approaches

### 2.1 Artificial neural networks (ANNs)

ANNs are an ML tool for modelling and solving nonlinear relationships between input and output data [5]. They are considered to be data-driven models with an unrestricted number of model parameters, and are very useful when there is a large amount of data. There are numerous applications of ANNs, and some examples include image classification [40], regression [49], forecasting [26] and real-time optimisation [63].

A neural network structure consists of three distinct layers: an input layer, hidden layer, and an output layer. This multi-layer system is also known as a multi-layer perceptron (MLP). During ANN model training, the input layer, which can have one or many nodes, passes the information to the nodes of the hidden layers. The

information from a node is multiplied by a weight matrix, denoted as  $W$ , and added to a value called a bias matrix, denoted as  $b$ . The output is then passed to an activation function  $F$ . This process continues between each node until the information finally reaches the output layer. Function  $F$  is the activation function, which in this study, incorporates a rectified linear unit (ReLU) for the hidden layers, and a linear function at the output layer [14]. The network architecture and equations for a fully connected network can be found in any standard textbook, for example [14].

The predicted output ( $Y_{ML}$ ) is compared to the actual output ( $Y$ ) by computing the loss function ( $L$ ) according to the mean absolute error (MAE), for this study and the regularisation loss function  $L_{reg}$ , as

$$\begin{aligned} \text{Loss function (MAE)} &= L(Y, Y_{ML}) \\ &= \frac{1}{\text{len}(Y)} \sum_{i=1}^{\text{len}(Y)} |Y - Y_{ML}|, \end{aligned} \quad (1)$$

$$L_{reg} = \lambda_{reg} \|W\|, \quad (2)$$

where,  $\text{len}(Y)$  represents the length of matrix  $Y$  and  $\lambda_{reg}$  represents the regularisation hyperparameter [14]. The total loss  $L + L_{reg}$  is minimised using the back-propagation algorithm by adjusting the values of  $W$  and  $b$ . Once training of the model is achieved, the trained model is used to predict new sets of data.

The performance of the models was also measured using root mean-squared error (RMSE), which is defined as

$$\text{RMSE}(Y, Y_{\text{pred}}) = \sqrt{\frac{1}{\text{len}(Y)} \sum_{i=1}^{\text{len}(Y)} (Y - Y_{\text{pred}})^2}, \quad (3)$$

where  $Y_{\text{pred}}$  in general is the predicted output from any models.

## 2.2 Machine learning approaches to geotechnical prediction

Recent developments in data science, such as ML and especially deep learning (DL) models including ANNs, support vector machines (SVMs), and Gaussian process regression (GPR), have helped research scientists and engineers use the data available from measurements in various geotechnical applications [16, 19, 32, 35, 44, 73, 75]. For instance, data-driven models for capturing the complex behaviour of soil compaction in estimating the material properties for quality assurance (QA) and quality control (QC) purposes have recently been considered and integrated with IC [17]. In particular, the response of the drum reaction (i.e. the acceleration history) of the roller's vibratory drum is used to estimate the in situ states of the compacted material (e.g. modulus and roller-related

stiffness) [7, 10]. However, the use of the purely data-driven ML models has the following disadvantages:

- (1) *Non-adherence to underlying physics* Data-driven ML models are generally not able to follow the underlying physics, since the models are normally trained on a limited set of data representing limited conditions. In other words, they have been found to produce results which deviate from known mechanistic behaviour or scientific principles [20].
- (2) *Prone to overfitting* Data-driven models may learn the training data rather than the underlying patterns and perform poorly on unseen test data [51]. Various regularisation techniques address this issue and are discussed later in this article.
- (3) *Less interpolation ability* ML models tend to have greater errors in a sparse dataset even when predicting within the range of training datasets [45].
- (4) *Less extrapolation ability* Similar to all empirically based models, ML models have found it challenging to make correct predictions beyond the range of training data [45].

## 2.3 Comparison of TM and ML

The development of a TM for a complex process, such as field compaction, requires an understanding of the complex interactions between the material and the roller compactor, and is limited by parameter availability and incomplete technical embodiment. Calibration of a TM's parameters can also be a challenge because of the combinatorial nature of the search space, which may result in over-complex models [21]. Conversely, ML algorithms have been considered a "black box" because of the models' hidden complexity and the fact that they may produce outputs which lack physical meaning, and this limits their use [27]. Hence, it is prudent to consider combining the TM and ML algorithms with a view to utilising their respective strengths and reducing their respective drawbacks [18, 45–48]. In other words, ML model complexity is reduced by incorporating theoretical and scientific knowledge through the TM [45]. This approach is known as theory-guided machine learning (TGML), which uses theoretical knowledge or frameworks to guide the construction and training of ML and DL.

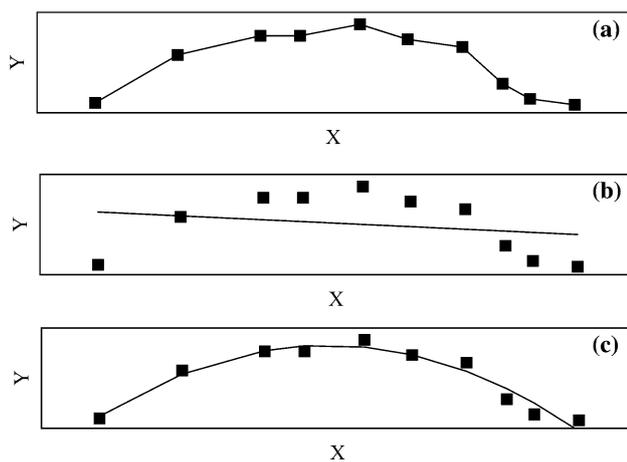
The TGML framework can be applied to any domain using theoretical and physics-based knowledge and available data. TGML techniques have been applied to solve differential equations [46] and cyber-physical systems [45], but there has been very limited use in the Civil or Geotechnical engineering domains focused on liquefaction assessment and groundwater flow modelling [11, 74]. Accordingly, this paper presents the TGML framework,

combining the use of TM and ML in three other ways to predict material density during compaction. The first technique (TGML1) addresses the ability to interpolate and extrapolate ML models by generating additional observations using already developed TM models. The second technique (TGML2) considers TM prediction as an additional input to ML training, and TGML3 presents a novel way of training ML models based on the underlying physics.

## 2.4 Bias variance trade-off

It is crucial to understand the bias variance trade-off to understand the workings of models, TM, ML and TGML. Models are developed on the training data, whereas, test data are used to evaluate the performance of developed models. Bias is the difference between a prediction of the model and the corresponding observed value, while variance is the variability of the model prediction for a given dataset, and reflects a measure of the prediction spread. A high-bias model is an oversimplified model which pays little attention to the data and thus fits the data poorly. Similarly, a high-variance model is a complex model which sometimes models the noise or error in the data, potentially leading to high error when unseen inputs are given to the model.

Figure 2 shows three models, A, B and C, representing a low bias-high variance model, a high bias high variance model and a low bias low variance model, fitted to a random dataset. The  $X$  and  $Y$  parameters in Fig. 2 represent the input and output data, respectively, for illustration purposes. Although Model A performs better on the training



**Fig. 2** **a** Model A: Low bias high variance model, **b** Model B: High bias high variance model, **c** Model C: Low bias low variance model ( $X$  and  $Y$  represent input and output data for illustration purposes)

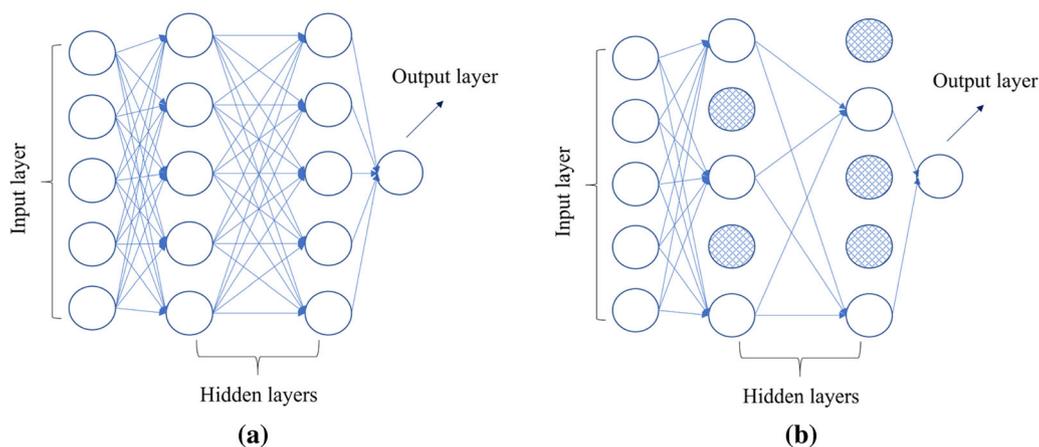
dataset, it is very complex, has more parameters, and may show poor prediction in the test dataset. Additional parameters imply a greater capacity for memorisation capacity and therefore perfect mapping with the expected and predicted values. Conversely, Model B may have a high bias representing significantly fewer parameters, producing poor predictive capability and high errors on both training and test data. Model C, which has low variance and low bias, is expected to yield better accuracy in both the training and test datasets.

Both TM and ML can be any of the above models; usually, ML and highly complex TM fall into the Model A category. Preferably, a model should fall into the category of Model C; this concept is further explained and used later in this paper.

## 2.5 Uncertainty in ML models

Trained neural network (NN) models produce only a single set of predictions when fed with test data or new data. Sometimes it is essential to know the level of confidence of the model outputs; hence, careful uncertainty quantification is crucial for practical applications.

Despite being able to handle complex processes with significant accuracy, NNs are poor at quantifying predictive uncertainty and often produce over-confident predictions [13, 28]. For example, suppose a NN trained on one dataset is evaluated on a completely different dataset. In that case, the network outputs high predictive uncertainty together with the prediction. The uncertainties associated with the model may also be because of uncertainties with the estimation of appropriate model weights and biases or be due to limited data availability. A measure of uncertainty provides users' confidence in the results obtained. Bayesian neural networks (BNNs), part of Bayesian approaches, are used to tackle uncertainties in NNs [22]. However, it has been found that BNNs can be very computationally expensive and require substantial customisation to training procedures [28, 53]. Recently, Gal and Ghahramani [12] introduced the Monte Carlo (MC) dropout to estimate model uncertainty. MC dropout is a variant of dropout used to prevent over-fitting [55]. Dropout refers to the temporary removal of nodes in a NN; as shown in Fig. 3, the dropped nodes are represented by shaded nodes. The selection of units to drop is random, with a dropout rate varying between 0 and 0.5, and the dropout rate refers to the fraction of nodes to be temporarily deactivated [55]. Uncertainty estimation using MC dropout requires making multiple predictions with different sets of nodes being dropped out. After the multiple predictions are completed, the mean and variance of the prediction are calculated.

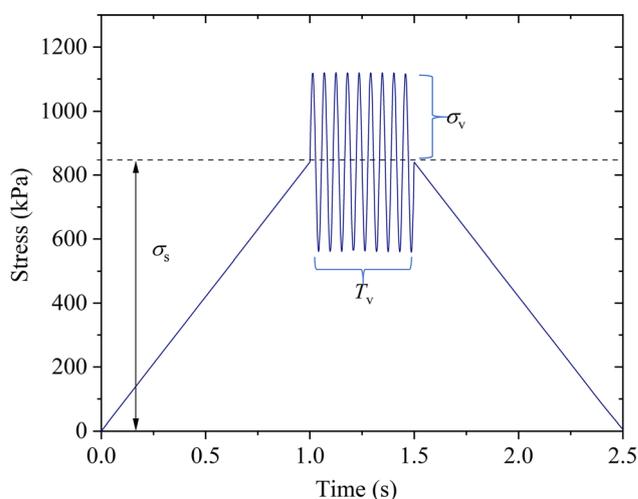


**Fig. 3** Comparison between **a** A standard NN and **b** NN after dropout (shaded nodes have been dropped) (modified after Srivastava et al. 2014)

### 3 Materials and testing method

The simulation of field compaction in a laboratory is a challenge, and very limited data are available in the literature, as it involves loading the sample up to 2 MPa of stress at around 18–30 Hz of vibratory loading [39, 50]. Hence, experiments were carried out in the laboratory with loading conditions which had some essential features of field compaction by roller. Table 1 shows the physical properties of the uniformly graded fine sand (FS) used for this study. One load cycle of repetitive loading simulating compaction is shown in Fig. 4. The samples were compacted at different initial void ratios ( $e_0$ ) and moisture contents ( $w$ ) in a modified Proctor mould (diameter 151.5 mm and height 132.2 mm) and were subjected to varying loading conditions, as given in Table 2.

All the samples were subjected to a total of 30 cycles ( $N$ ). Static stress ( $\sigma_s$ ), vibratory stress ( $\sigma_v$ ), duration for vibratory load ( $T_v$ ), and frequency of vibration ( $f$ ) were chosen such that they imitated actual field compaction and were varied to determine their effects on compaction. The static stress ( $\sigma_s$ ) reflects the stress due to the static weight



**Fig. 4** First cycle loading profile applied to samples showing static stress ( $\sigma_s$ ); vibratory stress ( $\sigma_v$ ); duration of vibratory load ( $T_v$ )

of the roller, whereas  $\sigma_v$  corresponds to the vibratory load of the roller. The experimental testing program is given in Table 2.

**Table 1** Geotechnical properties of material FS used in this study

Geotechnical property	Value	Standard
Specific gravity ( $G_s$ )	2.61	AS 1289.3.5.2 [56]
Median diameter ( $D_{50}$ ) mm	0.35	AS 1289.3.6.1 [57]
MDD standard $Mg/m^3$	1.69	AS 1289.5.1.1 [58]
OMC standard (%)	11.74	AS 1289.5.1.1 [58]
Optimum degree of saturation ( $S_{ropt}$ ) (%)	57	AS 1289.5.1.1 [58]
Coefficient of uniformity ( $C_u$ )	2.27	AS 1289.3.6.1 [57]
Coefficient of curvature ( $C_c$ )	0.97	AS 1289.3.6.1 [57]
Unified soil classification system (USCS) classification	SC	AS 1289.3.6.1 [57]

**Table 2** Experimental program undertaken for this study with varying initial conditions and stress levels

Sample ID	$w(\%)$	$e_0$	$f(\text{Hz})$	$\sigma_s(\text{kPa})$	$\sigma_v(\text{kPa})$
FS_1	0	0.632	18	840	280
FS_2	5	0.789	18	840	280
FS_3	10	0.772	18	840	280
FS_4	5	0.780	18	840	280
FS_5	10	0.747	18	840	280
FS_6	7	0.904	18	840	280
FS_7	7	0.934	18	840	280
FS_8	0	0.645	18	1400	840
FS_9	15	0.871	18	840	280
FS_10	15	0.823	25	1400	840
FS_11	15	0.790	18	1400	840
FS_12	15	0.829	30	1400	840
FS_13	17	0.767	18	1400	840
FS_14	0	0.636	18	1400	840
FS_15	0	0.629	25	1400	840
FS_16	0	0.634	30	1400	840

## 4 Theoretical model development

The theoretical model used in this study is an extension of the semi-empirical plastic strain accumulation model proposed by Sawicki and Swidzinski [52], modified for 1-D zero lateral strain. These researchers developed the relationship for axial or volumetric plastic strain ( $e^p$ ) accumulation with  $N$  when the sample was subjected to maximum vertical stress ( $\sigma_z = \sigma_s + \sigma_v$ ). Accordingly, for dry granular materials with different  $e_0$  and  $\sigma_z$ ,

$$e^p = C_1 \ln(1 + C_2(1 - K_0)^m N \sigma_z^m). \quad (4)$$

The stress-invariant (the stress tensor responsible for compaction) for Eq. (4) is considered to be  $(\sigma_z - \sigma_x)^m$ , which equals  $(1 - K_0)^m \sigma_z^m$ , where  $\sigma_x$  is the radial pressure and  $(K_0 = \frac{\sigma_x}{\sigma_z})$  denotes the coefficient of lateral pressure, while  $C_1$ ,  $C_2$ , and  $m$  are the material parameters. In the case of a 1-D oedometric test, since there are no lateral strains, the external work done on the sample by  $\sigma_x$  becomes zero, so the only work done is because of  $\sigma_z$ . Therefore, in the case of a 1-D condition, the  $\sigma_x$  term can be removed, and Eq. (4) can be approximated by Eq. (5). The stress term is normalised with  $1 \text{ kPa}$ :

$$e^p = C_1 \ln\left(1 + C_2 N \left(\frac{\sigma_z}{1 \text{ kPa}}\right)^m\right). \quad (5)$$

The evolution of  $e^p$  is then written in terms of the evolution of the void ratio ( $e$ ) with  $N$ ,

$$e_N = e_0 - (1 + e_0) C_1 \ln\left(1 + C_2 N \left(\frac{\sigma_z}{1 \text{ kPa}}\right)^m\right), \quad (6)$$

where  $e_N$  is the void ratio at the  $N$ th cycle,  $C_1$ ,  $C_2$  and  $m$  as functions of  $w$ , and other loading variables are presented later.

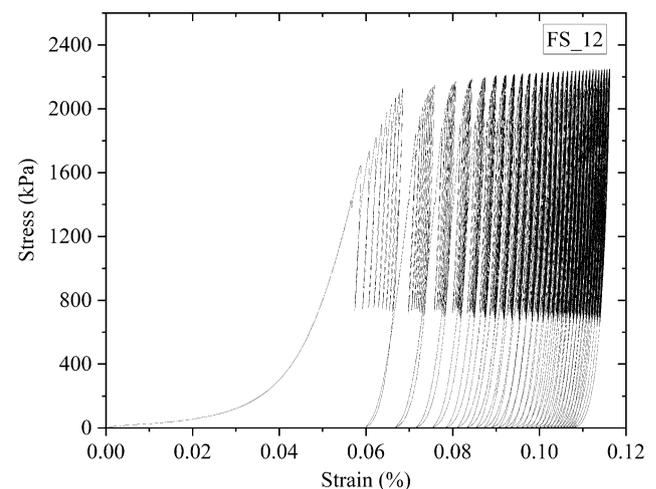
## 5 Experimental results

The stress–strain curve for sample ID FS\_12 is shown in Fig. 5, showing that most of the compression occurs in the first cycle ( $N = 1$ ) (in this sample 50% of total compression), and the compression rate then decreases with increasing  $N$ .

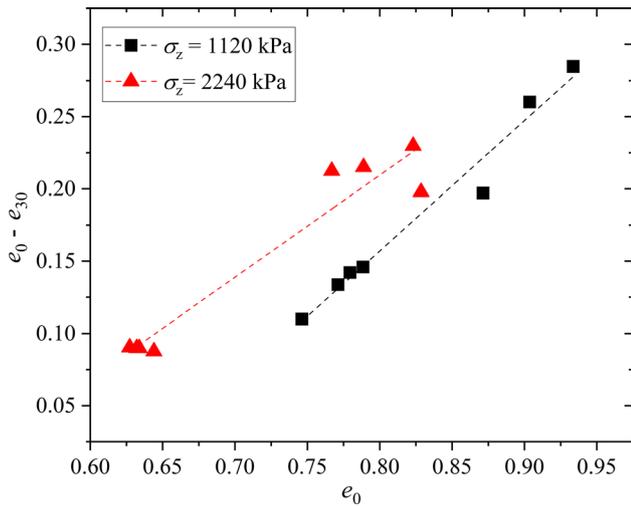
The relative change in the void ratio ( $e_0 - e_{30}$ ) is an indicator of the maximum compression the sample undergoes during loading for  $N = 30$ , and is plotted against  $e_0$  in Fig. 6. This shows that the degree of compaction, or in other words, the accumulation of  $e^p$ , decreases with an increase in the packing density (decrease in  $e_0$ ). In addition, the change in void ratio per cycle  $\frac{de}{dN}$  decreases with  $N$  for all the samples; some examples are shown in Fig. 7. Figure 7 also shows the effect of stress level and the higher the stress level the faster the stabilisation of the void ratio.

### 5.1 Influence of frequency on compaction

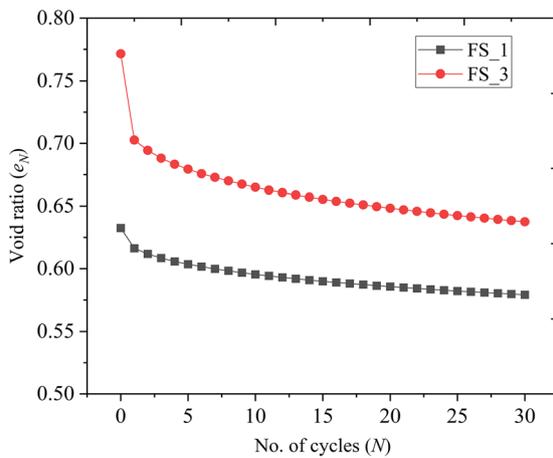
Figure 8 shows the normalised relative change in the void ratio  $\left(\frac{e_0 - e_{30}}{e_0}\right)$  plotted against the vibration frequency, illustrating that  $\frac{e_0 - e_{30}}{e_0}$  displays a peak at what is referred to as the optimum compaction frequency ( $f_{\text{opt}}$ ), which gives the maximum reduction in void ratio or increase in density (23 Hz for  $w = 15\%$  and 25 Hz for  $w = 0\%$ ). A similar observation has been made for field compaction so, ideally,  $f_{\text{opt}}$  should be estimated before soil compaction for optimal



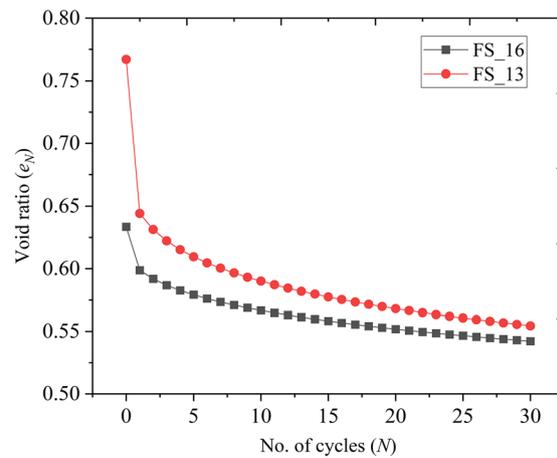
**Fig. 5** Stress–strain curve for sample ID FS\_12 subjected to vibratory loading



**Fig. 6** Variation of  $e_0 - e_{30}$  with  $e_0$  for different stress levels ( $\sigma_z = 1120$  kPa and  $\sigma_z = 2240$  kPa)

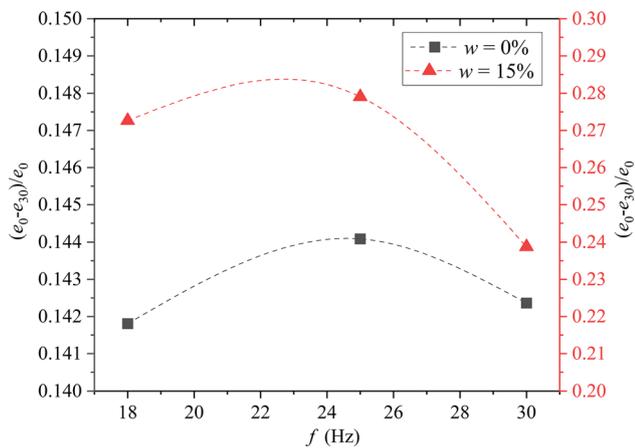


**(a)**

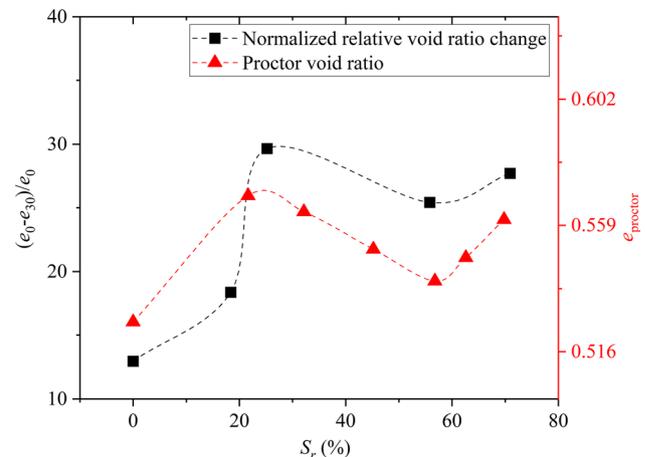


**(b)**

**Fig. 7** Evolution of void ratio with number of cycles for different samples **a** at  $\sigma_z = 1120$  kPa and **b**  $\sigma_z = 2240$  kPa



**Fig. 8**  $\frac{e_0 - e_{30}}{e_0}$  is plotted against  $f$  for  $w = 0\%$  (y-axis left) and  $15\%$  (y-axis right)



**Fig. 9** Comparison with  $\frac{e_0 - e_{30}}{e_0}$  and Proctor void ratio ( $e_{\text{proctor}}$ ) with  $S_r$

operation of the rollers, thus saving energy costs and minimising potential damage to the rollers [67].

## 5.2 Effect of degree of saturation ( $S_r$ ) on compaction

The effect of  $S_r$  is studied using Fig. 9, showing the variation of  $\frac{e_0 - e_{30}}{e_0}$  with  $S_r$ . Figure 9 shows that  $\frac{e_0 - e_{30}}{e_0}$  follows a similar trend to that of the compaction curve with the minima at around a  $S_r$  of 57%. This observation is consistent with that of past researchers, who reported that compaction curves with different compaction energies follow a similar trend (peak density or minimum  $e$  occurring at the same  $S_r$  as also seen in Fig. 1) irrespective of where those curves are developed, i.e., in the laboratory or field [24, 59–61].

### 5.3 Evaluation of $C_1, C_2$ and $m$

The model parameters ( $C_1, C_2$ , and  $m$ ) were found using the least-squares fitting procedure. Exploratory analyses using Eq. (6) highlighted that this equation is over-parameterised, and removing  $C_2$  helped reduce the standard error [3] in the estimation of  $m$  [72]. This was because the stress in this study was kept constant and the constant stress resulted in having a non-unique solution of  $C_2$  and  $m$ . However, for study where the stress is not constant,  $C_2$  would be required. The removal of  $C_2$  reduced model complexity, this however, increased the model's bias, but decreased the model's variance, which is more advantageous for usage with the test data (not shown). In other words, Model A is changed to Model C (as discussed in Fig. 2). Hence, after removing  $C_2$ , Eq. (6) is reduced to

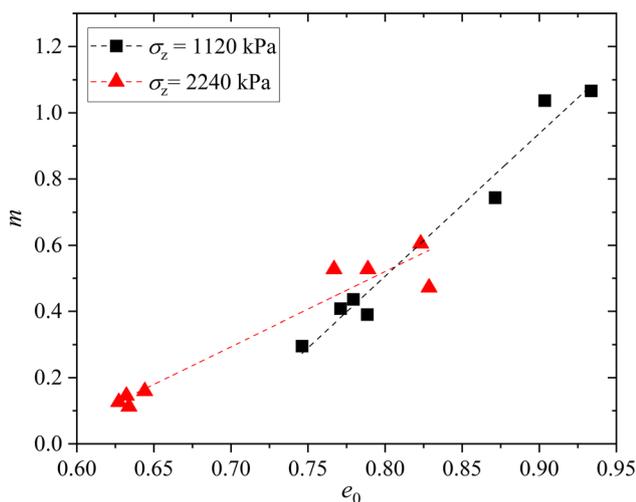
$$e_N = e_0 - (1 + e_0)C_1 \ln\left(1 + N\left(\frac{\sigma_z}{1 \text{ kPa}}\right)^m\right). \quad (7)$$

The value of the exponent  $m$  increases with  $e_0$ , as shown in Fig. 10, with the rate of change differing for different stress conditions. This indicates that stress dependency increases with an increase in the  $e_0$ , implying that loose samples are more dependent on stress. However, the value of  $C_1$  was found to be constant with values varying slightly within  $0.014 \pm 0.003$ .

## 6 Machine learning models

### 6.1 Hyperparameter tuning for ANNs

For this study, ANNs were used to model the dataset which comprised void ratios with the number of cycles from all



**Fig. 10** Variation of model parameter  $m$  with  $e_0$  for different stress levels ( $\sigma_z = 1120$  kPa and  $\sigma_z = 2240$  kPa)

the specimens with back-propagation algorithms. The void ratio at the  $N$  th cycle ( $e_N$ ) was the target or output, whereas  $N, e_0, w, f, \sigma_z$  were the various input parameters considered with appropriate hyperparameters. The hyperparameters of any neural networks are the variables which govern the training process, speed, and accuracy of any ML model. There are two different types of hyperparameters: (a) model hyperparameters, which include the number and width of hidden layers; and (b) algorithm hyperparameters, which encompass the learning rate for optimisers such as stochastic gradient descent (SGD) or the Adam optimiser for training the model [14]. Since these variables remain constant over the training process and directly impact the ML program's performance, they should be selected before training any model. Usually, the hyperparameters are selected by trial-and-error for optimal performance, and the procedure is generally referred to as hyperparameter tuning or hyper-tuning. The hyperparameters of the ANN and the multi-output ANN for this study were selected/tuned using the Keras tuner and the optimum values obtained are listed in Table 3.

### 6.2 Implementation of the ML models

All ML models were implemented in Python software and additional packages, including Keras, TensorFlow, Pandas, Numpy, and Seaborn [1, 8, 15, 36, 62, 64]. The Adam optimisation algorithm was used for performing back-propagation to evaluate the NNs model parameters with a maximum number of epochs equal to 10,000.

Various regularisation techniques were used to avoid over-fitting; first, by dividing the total dataset into training and test datasets randomly. For this study, 80% of the total dataset was used for training and 20% of the data for testing. An early stopping procedure was employed using a further 20% of the training data for validation to avoid overfitting. The value of patience for early stopping was kept equal to 500. L2 regularisation, also called Ridge regularisation, was also applied to force the weights to take small values, making weights more regular [14]. The complete dataset was normalised to zero mean and unity standard deviation to bring all parameters to the same scale. This normalisation was essential, because the model

**Table 3** Hyperparameter details of the ANN used for this study

Hyperparameter	Value
No. of hidden layers ( $H$ )	1
No. of nodes in the hidden layer	4
Optimiser	Adam
Learning rate	0.1

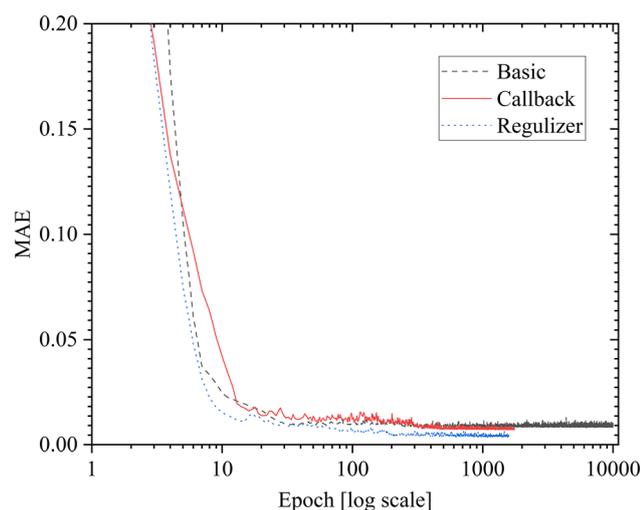
inputs used different scales and ranges. Since these inputs are multiplied by the model weights, the scales of the outputs and gradients were affected by the inputs' scales. Although a model might converge without feature normalisation, this normalisation makes training much more stable.

The fully connected NN architecture comprised three layers: one input layer, one hidden layer and one output layer with 5, 4, and 1 nodes, respectively. The values of the hyper-parameters were kept the same in all models, demonstrating that no unique tuning of hyper-parameters was performed for a specific problem.

Figure 11 shows the effect of the different regularisation techniques used for this study (basic, callback and regulariser). The basic model had 10,000 epochs without any specific regularisation techniques. For the basic model, MAE fluctuated with epochs because of the constant learning rate; therefore, in the callback model, the learning rate was set to hyperbolically decrease with epochs for better convergence of MAE. The callback model also included the early stopping technique by monitoring the change in MAE; this also helped to save computational time in comparison with the basic model. The regulariser model included callback and L2 regularisation, and was better than the callback function in terms of time and cost, as shown in Fig. 11, where the model training stopped early at lower epochs. Henceforth, all the analysis was conducted based on the regulariser model's hyperparameters.

### 6.3 Extrapolation ability of TM and ML

The extrapolation ability of TM and ML was studied by splitting the experimental data into two parts: 1–20 cycles



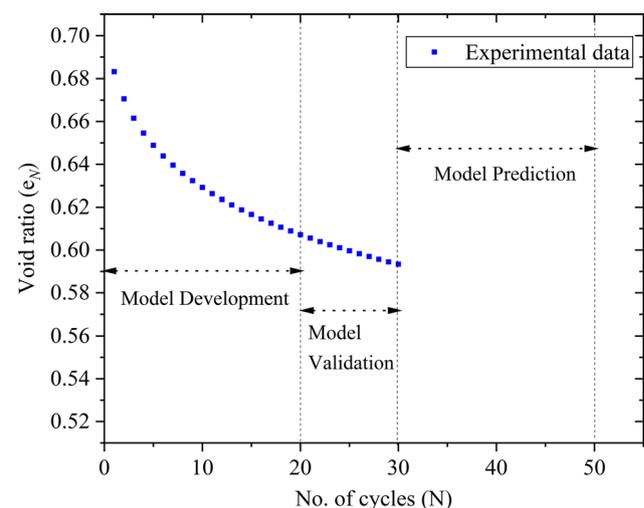
**Fig. 11** MAE comparison of different regularisation models and variation with number of epochs (basic model, callback model and regulariser model)

and 21–30 cycles. As shown in Fig. 12, both models were developed with data from 1 to 20 cycles, and their prediction accuracy was validated against the data from 21 to 30.

Figure 13 shows that TM and ML predicted unseen data and extrapolated well, as they have a low MAE and RMSE, although TM was slightly better than ML. As the model's prediction capability has been validated, in the subsequent sections of this article, the model is developed for the complete dataset (i.e. cycles 0–30), and the prediction is presented for cycles 31–50 beyond the measured data to demonstrate the model's extrapolation capability.

### 6.4 Uncertainty estimation of the ANN used in this study

For the present study, the data for sample FS\_10 (randomly selected) were used to demonstrate the MC dropout. A different dropout ratio was considered to determine the effect of the dropout ratio on prediction accuracy and the 95% confidence band. The results of this analysis are shown in Fig. 14. The results indicate that the dropout ratios of 0.1 and 0.2 have lower MAE and a narrow confidence band. For the dropout ratios of 0.3 and 0.4, MAE increases together with the confidence band. Compared with no dropout ( $MAE = 2.7 \times 10^{-3}$ ), the dropout ratios of 0.1 and 0.2 perform better in terms of error, whereas 0.3 and 0.4 perform worse. Figure 14 also reveals that when predicting the void ratio for cycles 31–50, where training data were not available, the confidence band is broader than that for cycles 1–30, showing the model's higher uncertainty in extrapolation.



**Fig. 12** Data splitting for model development, prediction validation, and model prediction

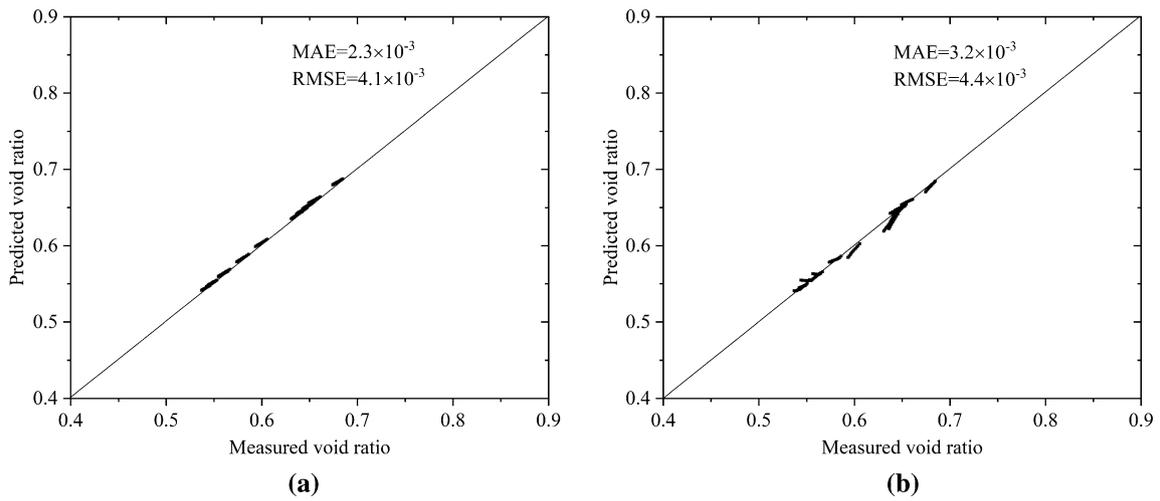


Fig. 13 Comparison of predicted and measured void ratio of **a** TM and **b** ML for the data of cycles 21–30

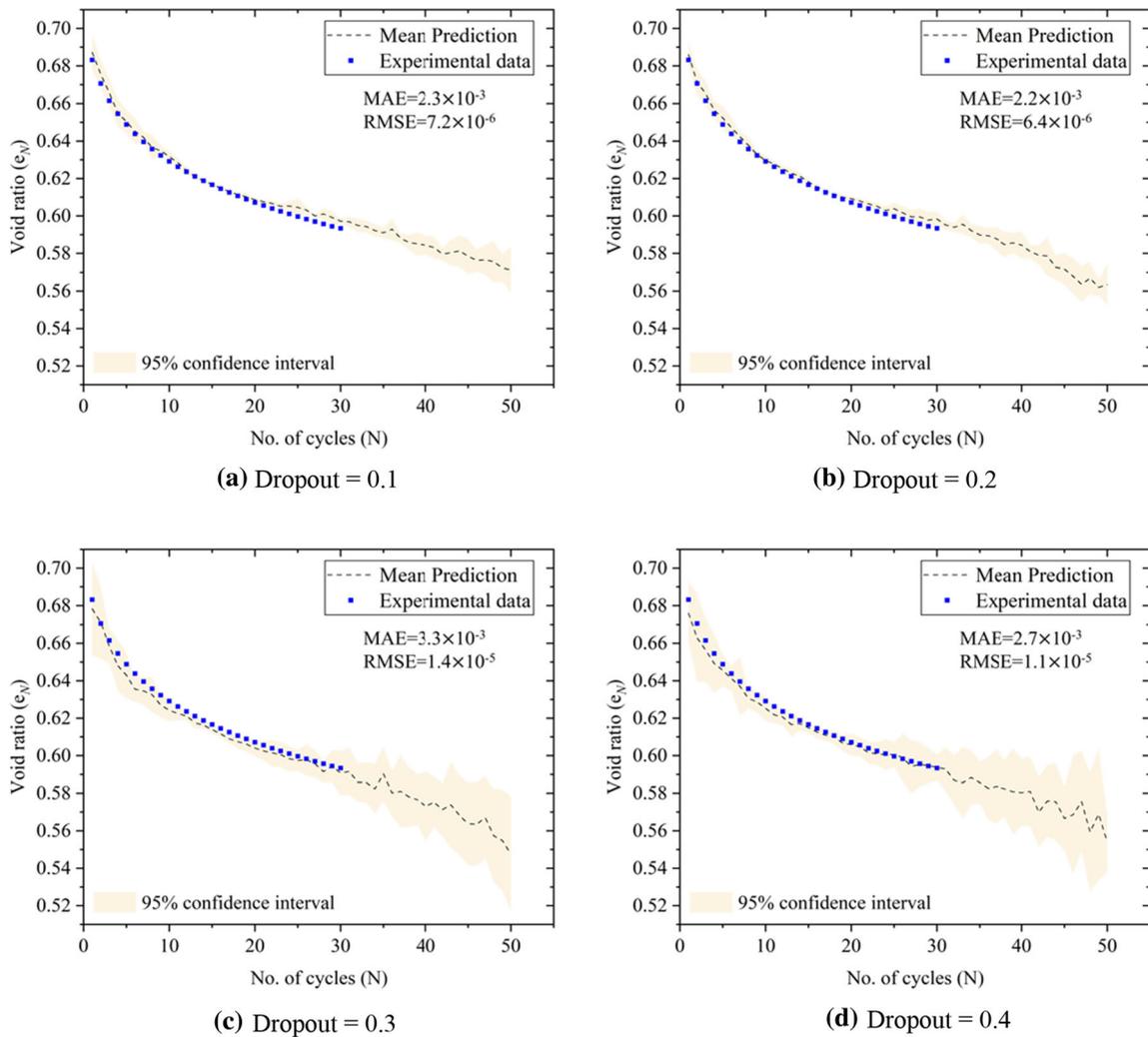


Fig. 14 Comparison of different dropout ratios **a** dropout ratio of 10%, **b** dropout ratio of 20%, **c** dropout ratio of 30%; and **d** dropout ratio of 40% on MAE, RMSE (for data up to 30 cycles), and confidence band

## 7 Theory-guided machine learning (TGML)

This section discusses the three techniques of TGML developed in this research to better predict the evolution of void ratio and eliminate the limitations associated with a theoretical model and machine learning-based models (in this study ANN). Schematic illustrations of the three TGML models are provided in Fig. 15, and their performance is measured and compared in subsequent sections.

### 7.1 Data augmentation (TGML1)

ML is a complex model which often attempts to learn the data used for training; it fits very well in the input domain and gives more substantial error when predicting outside the domain. Therefore, a complex ML model generally has a problem with extrapolation. This problem can be overcome by data augmentation from TM. Data augmentation is used to create or generate new observations to train an ML algorithm. Generally, prediction accuracy increases as the size of the input dataset increases. For example, in image classification problems, data scientists have managed to augment the data by adding a new image which is a rotated version of the original image [54]. Similarly, in the present research, the data augmentation technique using Eq. (7) was used to produce physics-based new data, as shown in Fig. 15c, with the model labelled TGML1. To use TM for data augmentation beyond the input's domain, the ability to extrapolate should be superior and was validated earlier in Fig. 13.

TGML1 saves time and resources in developing additional experimental data, preventing ML from overfitting, and improving prediction capability. To compare the efficiency of TGML1, first, ML was trained with experimental data of up to 30 cycles, as shown in Fig. 16a. Another network was trained with data of up to 50 cycles with observations from 31 to 50 cycles generated using Eq. (7), as shown in Fig. 16b. A comparison of the results reveals that the MAE calculated for cycles up to 30 of the TGML1 network was less than that of the network without augmentation, and the confidence band was narrow for the TGML1 network. In this case, because the model was trained with additional data, the accuracy of the prediction increased. It should be noted that model improvement using data augmentation depends on the extrapolation ability of TM, and in this case, the accuracy improved because the TM used in this study could extrapolate well.

### 7.2 Ingesting the output from TM as an additional input parameter to ML (TGML2)

In TGML2, the data were first pre-processed with the TM. The output of the TM was then used as an additional parameter for ML, as shown in Fig. 15d. In TM, the input was mapped to output  $[X] \rightarrow [Y]$  by calibrating the model parameters using the experimental or observed data. Generally, in TM, assumptions are simplified, and simpler models are built; thus, TM predicts  $[Y_{TM}]$  as shown in Fig. 15a, which is not precisely equal to  $[Y]$ . Similarly, any ML model, when mapped from  $[X] \rightarrow [Y]$  over a set of training data, predicts  $[Y_{ML}]$  and is not exactly equal to  $[Y]$ , as given in Fig. 15b. By adding the  $[Y_{TM}]$  to ML's input parameters, ML complements TM and captures the remaining complexity of the system. If the TM is highly accurate, then the TGML2 ensures  $[Y_{TGML2}] = [Y_{TM}]$ .

The prediction accuracy of TGML2 is compared in Fig. 17, indicating that the model which considers the output of the TM as an additional input (Fig. 17b) is an improvement over the model which does not consider the output of the TM as an additional input (Fig. 17a) in terms of MAE and RMSE.

### 7.3 Theory-guided regularisation (TGML3)

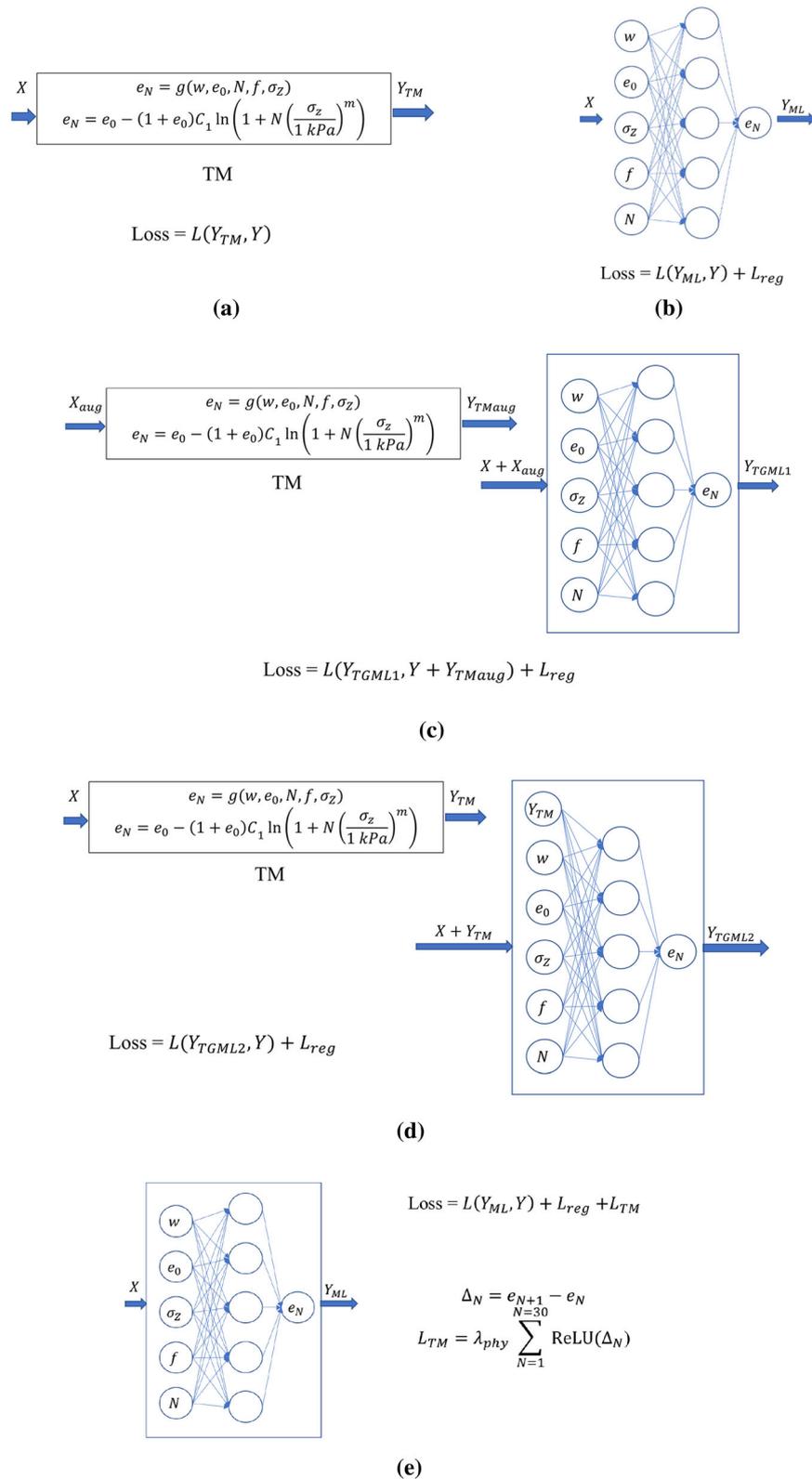
Theory-guided regularisation involves embedding the TM details in the loss function of ML. This way of embedding the TM in the loss function ensures that ML is constrained to comply with the theoretical model. The new architecture ensures that it penalises TM constraint violations by introducing additional regular loss function goals, as shown in Fig. 15e. The restriction is that with an increase in  $N$ , the  $e_N$  always decreases, which can be written as  $e_N - e_{N+1} > 0$ . The regular loss function was therefore modified by adding the TM-based loss function  $L_{TM}$ . The difference of the predicted void ratio as a pair was calculated as

$$\Delta_N = e_{N+1} - e_N. \quad (8)$$

A positive value of  $\Delta_N$  can be viewed as a violation of physics.  $L_{TM}$  was therefore calculated as a non zero occurrence of  $\text{ReLU}(\Delta_N)$  summed over all the cycles, which is then multiplied by a suitable hyperparameter  $\lambda_{\text{phy}}$ .  $\lambda_{\text{phy}}$  is evaluated by trial-and-error like other hyperparameters in this study. The final equation for the  $L_{TM}$ ,

$$L_{TM} = \lambda_{\text{phy}} \sum_{N=1}^{N=30} \text{ReLU}(\Delta_N). \quad (9)$$

The statistical evolution of TGML3 was done with a noisy dataset and is discussed in the next section.



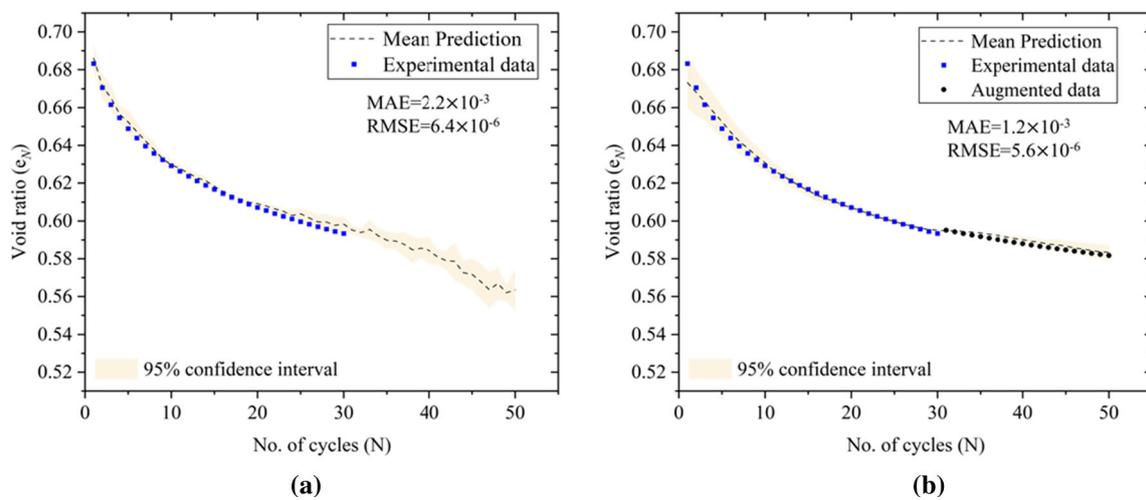
**Fig. 15** Schematic illustrations of various techniques to estimate the evolution of void ratio with number of cycles **a** Model based on theory-based equation (TM). **b** Data-driven ML. **c** Data augmentation using TM (TGML1). **d** Ingesting the output from TM as an additional input parameter to ML (TGML2) and **e** Theory-guided regularisation (TGML3)

## 7.4 Robustness to noisy input data

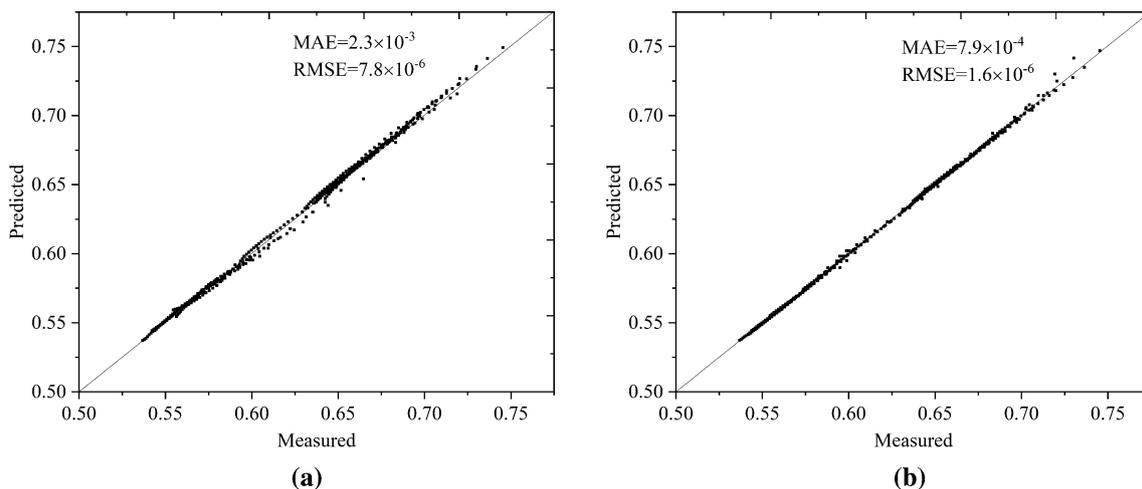
In the real world, data are never perfect; they always contain some noise. Whether the data source is an electrical signal or collected from the laboratory or field environment, it is bound to be noisy because of uncertainties involved with testing, measurement, limitations of the equipment used, and human error. To evaluate the robustness of the model against noisy data, Gaussian noise ( $\mathcal{N}$ ) was added to the experimental data.  $\mathcal{N}$  is defined with the mean ( $\mu$ ) and variance ( $s^2$ ) as  $\mathcal{N} \sim (\mu, s^2)$ . For this analysis, the  $\mu$  of the noise was set to zero, whereas  $s^2$  was set at 1% of  $\mu$  of the data collected. A noisy observation ( $Y_{\text{Noisy}}$ ) is written as

$$Y_{\text{Noisy}} = Y + \mathcal{N} \sim (\mu, s^2). \quad (10)$$

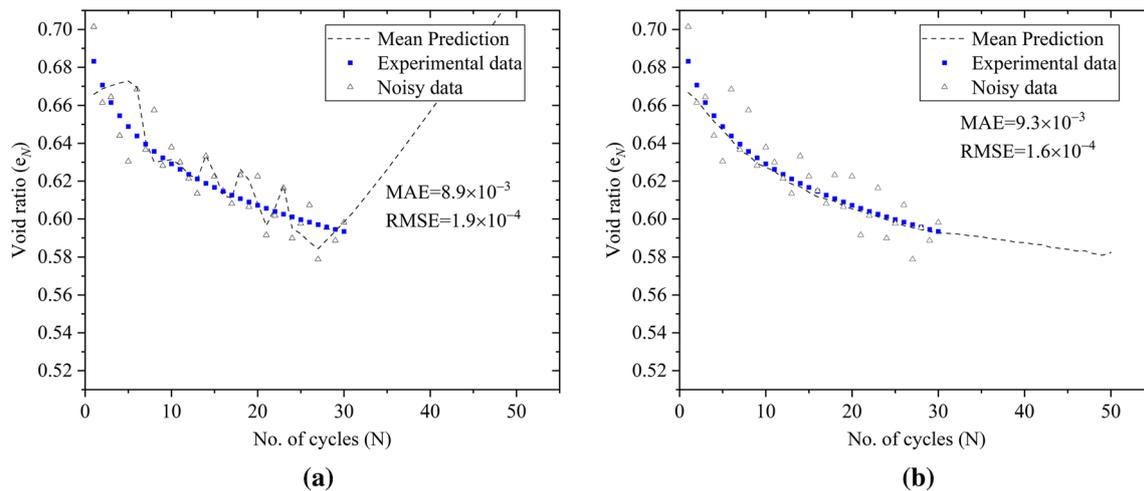
Figure 18 compares the prediction of models with and without TGML 3. When calculating the MAE with prediction and noisy dataset, the MAE in the model with TGML3 shows a higher value. However, when the model's prediction is compared with the actual experimental dataset, the model with TGML3 provides MAE of  $2.7 \times 10^{-3}$  and RMSE of  $1.5 \times 10^{-5}$ , whereas the model without TGML3 provides MAE of  $9.2 \times 10^{-3}$  and RMSE of  $1.1 \times 10^{-4}$ . In addition, the model with TGML3 follows theoretical knowledge, and the void ratio decreases with the number of cycles. TGML3 performs better with noisy



**Fig. 16** Void ratio evolution with the number of cycles of dataset **a** without augmentation, **b** with augmentation and their prediction accuracy in terms of MAE and RMSE (for data up to 30 cycles)



**Fig. 17** Measured and predicted void ratio comparison of all samples **a** without considering TM output, **b** with considering TM output (TGML2) and their prediction accuracy in terms of MAE and RMSE (for data up to 30 cycles)



**Fig. 18** Void ratio evolution of noisy dataset and prediction comparison of model **a** without TGML3, **b** with TGML3 and their prediction accuracy in terms of MAE and RMSE (for data up to 30 cycles)

data because combining TM and ML gives more complex models with high variance and low bias (Model A is changed to Model C as in Fig. 2). Therefore, when noise was added, the complex models tried to fit the noise and gave an error for the noise-free test data.

## 8 Termination criteria

The termination criteria, which are used to find the required number of cycles ( $N_{\text{target}}$ ) to reach a target void ratio ( $e_{\text{target}}$ ) or dry density for a given loading condition, can be measured using TGML in two different ways. First, the network needs to be trained again, but this time the output is  $N$ , where  $e_N$  is part of the input parameters. Retraining requires a substantial amount of time if the dataset is large. The second method of finding the  $N_{\text{target}}$  is by using the already trained model. This method is faster than the first method. The trained or learned TGML can be used to optimise the parameters for compaction using the back-propagation technique. The optimisation of one or more input parameters is the reverse of training a model. In training a model, the network parameters (weights and biases) are trained with fixed inputs, whereas optimising one or more input parameters is undertaken with fixed hyperparameters, which are known once the model is trained [70]. For the termination criteria,  $N_{\text{target}}$  to reach  $e_{\text{target}}$  is needed while the other input parameters and network parameters are fixed. This is a case of back-propagation optimisation where error is minimized to find the  $N_{\text{target}}$ . The differential equation solution to the optimisation problem is given by

$$N' = N_{\text{initial}} - \frac{\eta \partial E}{\partial N}, \quad (11)$$

where  $N_{\text{initial}}$  is the initial guess,  $E$  is the error (MAE) after the initial guess,  $\eta$  is the optimisation rate, and  $N'$  is the next prediction. The procedure continues until  $E$  is reduced to a minimum value. The optimisation algorithm was also executed in Python-based TensorFlow1. A simple illustration of this algorithm with parameters is shown in Table 4.

Using the data from Table 4 and TGML1 as the trained model, 41 cycles were required to reach the  $e_{\text{target}}$  of 0.62. The utilisation of TGML to calculate the termination criteria at the site has substantial practical applications. Currently, the required number of roller passes at the site to achieve the desired degree of compaction is obtained by doing some in situ density measurement, such as NDG testing, sampling, and other destructive testing methods. If the required number of passes can be estimated using TGML, the number of tests required can be substantially reduced, thus reducing the time and cost of the project. Moreover, as discussed earlier, it eliminates the disadvantages associated with over-compaction.

**Table 4** Parameters and target values used for the study of termination criteria

Parameter	Value
$e_0$	0.823
$w(\%)$	15
$f(\text{Hz})$	30
$\sigma_z(\text{kPa})$	2240
$e_{\text{target}}$	0.62

## 9 Discussion

The compaction due to a roller in the field is not exactly 1-D compression. Figure 19a demonstrates the initial condition when the material is placed loosely, while Fig. 19b displays the instantaneous deformed shape of the material when a roller is in operation, and Fig. 19c illustrates the state of the material after compaction. While the deformation is not exactly 1-D, when the compaction is completed the overall deformation behaviour of the section can be approximated as 1-D. Wersäll et al. [66] and Wersäll and Larsson [65] concluded that the settlement behaviour from rotating mass oscillators, which are similar to vibratory rollers, is predominately vertical, and horizontal displacement is negligible. Recent results and the discussion illustrated in Fig. 19 support that deformation is predominately 1D; therefore, complex 3-D compaction can be approximated using 1-D equations.

A similar idea of approximating 3-D behaviour with 1-D equations has been proposed by Raissi and Karniadakis [46], who showed that any high fidelity data or model can be simulated using a low fidelity model with corrections. Likewise, data from field compaction (high fidelity data) can be well-represented by a low fidelity (1-D model) and corrections made using appropriate ML techniques. Therefore, it is believed that void ratio evolution data from the field can be modelled using a combination of the 1-D TM developed in the present study and a suitable ML model.

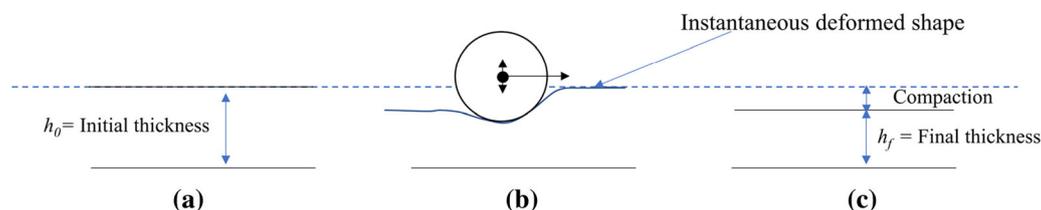
Laboratory test data on change in the void ratio of a fine sand during compaction were analysed and modelled using theoretical and ML approaches. Analysis of the experimental observations using TM showed that the model parameters and degree of compaction depend on different initial conditions, such as the initial void ratio ( $e_0$ ), the degree of saturation ( $S_r$ ), the load level ( $\sigma_z$ ), and the frequency of vibration ( $f$ ), as shown in Figs. 6, 7, 8, 9 and 10. The highlight of the analysis is shown in Fig. 9, which shows that the maximum relative compaction occurs when the sample is prepared close to the optimum degree of saturation ( $S_{ropt}$ ) ( $S_{ropt}$  obtained from Proctor compaction testing). This phenomenological feature emphasises the importance of  $S_r$  during compaction, which has also been

the focus of other recent studies [24, 59–61]. The importance of the vibration frequency for compaction is also highlighted in the analysis, i.e. there exists an optimum frequency at which the compaction is maximum. This observation is consistent with findings made during the field compaction of material, which also show an optimum frequency for maximum compaction [67, 68]. Ideally, this frequency should be evaluated before compaction starts for greater efficiency.

As discussed above, since model parameters depend on different initial conditions, TM cannot be generalised. Furthermore, with a large dataset, analysis using a TM would be challenging and computationally expensive. These issues were addressed with the use of 3-layer ANN models, which can generalise any loading condition. However, since ML models are prone to overfitting, different regularisation techniques, including callback and L2 regularisation, were used to reduce the error and decrease the computational time required to train the network.

It was also highlighted that greater accuracy could be achieved by combining data-driven ML models and TM encompassing essential physics, termed theory-guided machine learning (TGML). TGML increases the ability to interpolate and extrapolate, which is an essential aspect of the geotechnical engineering field [75]. TGML also reduces error and increases prediction confidence by reducing the 95% confidence band estimated using MC dropout. Noise in the dataset is unavoidable, especially measurements under field conditions. TGML provides much improved prediction in the case of noisy datasets compared with marginal improvement when dealing with clean laboratory datasets.

The termination criterion was discussed, which involves estimating the number of passes required to achieve a target density or void ratio. This analysis utilised the already-trained TGML1 model to predict the void ratio at any particular cycle, rather than re-training the model by interchanging the parameters, which could result in high computational cost.



**Fig. 19** Deformation pattern because of roller compaction; **a** initial loose soil, **b** instantaneous deformation pattern during compaction, **c** final deformed shape

## 10 Conclusions

A series of 1-D tests was conducted on uniformly graded fine sand with different initial densities and moisture contents. The loading condition simulated actual roller movement in terms of load level and frequency. The replication of roller loading conditions is challenging in the laboratory, as it involves applying loads as high as 2 MPa at a vibration frequency of 30 Hz, and there is an obvious lack of such data in the literature.

These data were analysed using a theoretical model based on the literature and an ANN model; both were found to be equally efficient in predicting the behaviour observed in the laboratory testing. Since the TM model parameters were dependent on initial conditions, TM could not be generalised. However, ML is more general, as it can consider all the parameters as an input, but its prediction lacked physical significance as it also tried to fit the noise of the data.

The fusion of TM and ML algorithms, termed TGML in this paper, addresses the disadvantages of TM and ML alone. The following conclusions on TGML were drawn:

1. TGML1, a data augmentation technique to improve ability to extrapolate, is an advantageous technique which addresses issues with the conduct of complex and expensive experiments. This technique involves the conduct of limited experiments for TM and using the developed TM to create new observations for any ML model. However, the extrapolation ability of TM should be
2. TGML2 considers the prediction of the TM as an additional input to the ML model. This method ensures that TM and ML complement each other. ML increases the complexity of a simpler 1-D TM, whereas TM restricts the prediction of ML to follow the physics involved in the compaction process.
3. TGML3 involves modification of the loss function of the ML model to include an additional loss function based on physical knowledge of the system. This involves rewriting the training steps to accommodate the additional loss term, but is very important, as shown in this paper. For this work, the constraint on void ratio was applied. This idea can be extended to other applications using different controls. For example, if the behaviour of material changes after  $S_{\text{ropt}}$ , the constraint could be provided in the model itself, rather than relying on the model to deduce it, which may not always be possible.
4. The highlight of this paper is the modelling of a noisy dataset. The paper shows that when modelling a clean dataset obtained from laboratory testing, TGML is marginally better than ML and TM; however, when

dealing with noisy datasets, the prediction of TGML is far superior to that of ML.

The TGML techniques discussed above could also be combined if an individual TGML model cannot capture a complex behaviour. For example, if the dataset is noisy and the model's ability to extrapolate needs to be improved, TGML1 and TGML3 can be combined. In this paper, the application of the TGML is shown for ANNs only; however, the same concept can be used for other ML algorithms such as support vector regression (SVR) and random forest (RF). The ML and TM should be carefully selected for developing the TGML. Accordingly, the ML should be relatively simple and flexible for merging with the TM, which should be reasonably accurate. It should be noted that a very poor-quality TM can cause the hybrid model to have unsatisfactory performance compared to the ML model alone.

**Acknowledgements** The first author received a Monash University Graduate Scholarship (MGS) to undertake this research project. This research work is also part of a research project (Project No IH18.03.3) sponsored by the Smart Pavements Australia Research Collaboration (SPARC) Hub (<https://sparchub.org.au>) at the Department of Civil Engineering, Monash University, funded by the Australian Research Council (ARC) Industrial Transformation Research Hub (ITRH) Scheme (Project ID: IH180100010). The authors gratefully acknowledge the financial and in-kind support of Monash University, SPARC Hub, Construction, Infrastructure, Mining and Concessions (CIMIC), and Engineering, Innovation and Capability (EIC) activities.

**Funding** This study was supported by the Australian Research Council (ARC) Industrial Transformation Research Hub (ITRH) Scheme (Grant No. IH180100010).

## Declaration

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1. Abadi M, Barham P, Chen J, et al (2016) Tensorflow: a system for large-scale machine learning. In: 12th USENIX symposium on operating systems design and implementation (OSDI 16), pp 265–283
2. Allen JJ, Thompson MR (1974) Resilient response of granular materials subjected to time dependent lateral stresses. *Transp Res Rec* 1–13
3. Alper JS, Gelb RI (1990) Standard errors and confidence intervals in nonlinear regression: comparison of Monte Carlo and parametric statistics. *J Phys Chem* 94:4747–4751. <https://doi.org/10.1021/j100374a068>
4. Anderegg R, Kaufmann K (2004) Intelligent compaction with vibratory rollers: feedback control systems in automatic compaction and compaction control. *Transp Res Rec J Transp Res Board* 1868:124–134. <https://doi.org/10.3141/1868-13>

5. Braspenning PJ, Thuijssman F, Weijters AJMM (1995) Artificial neural networks: an introduction to ANN theory and practice. Springer, Berlin
6. Brown SF, Hyde AFL (1975) Significance of cyclic confining stress in repeated-load triaxial testing of granular material. *Transp Res Rec*. [https://doi.org/10.1016/0148-9062\(76\)90013-9](https://doi.org/10.1016/0148-9062(76)90013-9)
7. Cao L, Zhou J, Li T et al (2021) Influence of roller-related factors on compaction meter value and its prediction utilizing artificial neural network. *Constr Build Mater* 268:121078. <https://doi.org/10.1016/j.conbuildmat.2020.121078>
8. Chollet F, others (2015) Keras
9. Chong SH, Santamarina JC (2016) Sands subjected to repetitive vertical loading under zero lateral strain: accumulation models, terminal densities, and settlement. *Can Geotech J* 53:2039–2046. <https://doi.org/10.1139/cgj-2016-0032>
10. Commuri S, Mai AT, Zaman M (2011) Neural network-based intelligent compaction analyzer for estimating compaction quality of hot asphalt mixes. *J Constr Eng Manag* 137:634–644. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0000343](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000343)
11. Depina I, Jain S, Mar Valsson S, Gotovac H (2021) Application of physics-informed neural networks to inverse problems in unsaturated groundwater flow. *Georisk Assess Manag Risk Eng Syst Geohazards*. <https://doi.org/10.1080/17499518.2021.1971251>
12. Gal Y, Ghahramani Z (2016) Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: The 33rd international conference on machine learning ICML 2016, vol 3, pp 1651–1660
13. Ghahramani Z (2015) Probabilistic machine learning and artificial intelligence. *Nature* 521:452–459. <https://doi.org/10.1038/nature14541>
14. Géron A (2017) Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Newton
15. Harris CR, Millman KJ, van der Walt SJ et al (2020) Array programming with NumPy. *Nature* 585:357–362. <https://doi.org/10.1038/s41586-020-2649-2>
16. He X, Xu H, Sabetamal H, Sheng D (2020) Machine learning aided stochastic reliability analysis of spatially variable slopes. *Comput Geotech* 126:103711. <https://doi.org/10.1016/j.compgeo.2020.103711>
17. Imran SA, Barman M, Commuri S et al (2018) Artificial neural network-based intelligent compaction analyzer for real-time estimation of subgrade quality. *Int J Geomech* 18:1–14. [https://doi.org/10.1061/\(ASCE\)JGM.1943-5622.0001089](https://doi.org/10.1061/(ASCE)JGM.1943-5622.0001089)
18. Jia X, Karpatne A, Willard J et al (2018) Physics guided recurrent neural networks for modeling dynamical systems: application to monitoring water temperature and quality in lakes. In: 8th international workshop on climate informatics
19. Kang J, Zhaofeng C, Zhaoyu L, Wang SY (2021) Characterization of particle orientation of kaolinite samples using the deep learning-based technique. *Acta Geotech*. <https://doi.org/10.1007/s11440-021-01266-x>
20. Karpatne A, Atluri G, Faghmous JH et al (2017) Theory-guided data science: a new paradigm for scientific discovery from data. *IEEE Trans Knowl Data Eng* 29:2318–2331. <https://doi.org/10.1109/TKDE.2017.2720168>
21. Karpatne A, Watkins W, Read J, Kumar V (2017) Physics-guided neural networks (PGNN): an application in lake temperature modeling. *arXiv:1710.11431*
22. Khalaj S, BahooToroodi F, Mahdi Abaei M et al (2020) A methodology for uncertainty analysis of landslides triggered by an earthquake. *Comput Geotech* 117:103262. <https://doi.org/10.1016/j.compgeo.2019.103262>
23. Kodikara J (2012) New framework for volumetric constitutive behaviour of compacted unsaturated soils. *Can Geotech J* 49:1227–1243. <https://doi.org/10.1139/t2012-084>
24. Kodikara J, Islam T, Sounthararajah A (2018) Review of soil compaction: history and recent developments. *Transp Geotech* 17:24–34. <https://doi.org/10.1016/j.trgeo.2018.09.006>
25. Kodikara J, Jayasundara C, Zhou AN (2020) A generalised constitutive model for unsaturated compacted soils considering wetting/drying cycles and environmentally-stabilised line. *Comput Geotech* 118:103332. <https://doi.org/10.1016/j.compgeo.2019.103332>
26. Kolarik T, Rudorfer G (1994) Time series forecasting using neural networks. *ACM Sigapl Apl Quote Quad* 25:86–94
27. Kumar D, Wong A, Taylor GW (2017) Explaining the unexplained: a class-enhanced attentive response (CLEAR). *Arxiv* 36–44
28. Lakshminarayanan B, Pritzel A, Blundell C (2017) Simple and scalable predictive uncertainty estimation using deep ensembles. *Adv Neural Inf Process Syst* 30:6403–6414
29. Lee J, Lacey D, Look B (2017) P60: Best practice in compaction QA for pavement and subgrade materials (year 1–2016/2017). [www.nacoe.com.au](http://www.nacoe.com.au). Accessed 25 May 2021
30. Lekarp F, Dawson A (1998) Modelling permanent deformation behaviour of unbound granular materials. *Constr Build Mater*. [https://doi.org/10.1016/S0950-0618\(97\)00078-0](https://doi.org/10.1016/S0950-0618(97)00078-0)
31. Li N, Wang X, Qiao R et al (2020) A prediction model of permanent strain of unbound gravel materials based on performance of single-size gravels under repeated loads. *Constr Build Mater* 246:118492. <https://doi.org/10.1016/j.conbuildmat.2020.118492>
32. Liu Z, Shao J, Xu W, Wu Q (2015) Indirect estimation of unconfined compressive strength of carbonate rocks using extreme learning machine. *Acta Geotech* 10:651–663. <https://doi.org/10.1007/s11440-014-0316-1>
33. Liu D, Wang Y, Chen J, Zhang Y (2019) Intelligent compaction practice and development: a bibliometric analysis. *Eng Constr Archit Manag* 27:1213–1232. <https://doi.org/10.1108/ECAM-05-2019-0252>
34. Look BG (2020) Overcoming the current density testing impediment to alternative quality testing in earthworks. *Aust Geomech J* 55:55–74
35. Makasis N, Narsilio GA, Bidarmaghz A (2018) A machine learning approach to energy pile design. *Comput Geotech* 97:189–203. <https://doi.org/10.1016/j.compgeo.2018.01.011>
36. McKinney W et al (2010) Data structures for statistical computing in python. In: Proceedings of the 9th python in science conference, pp 51–56
37. Modoni G, Koseki J, Anh Dan LQ (2011) Cyclic stress-strain response of compacted gravel. *Geotechnique* 61:473–485. <https://doi.org/10.1680/geot.7.00150>
38. Monismith CL, Ogawa N, Freeme CR (1975) Permanent deformation characteristics of subgrade soils due to repeated loading. *Transp Res Rec*, vol 537
39. Mooney MA, Rinehart RV (2007) Field monitoring of roller vibration during compaction of subgrade soil. *J Geotech Geoenviron Eng* 133:257–265. [https://doi.org/10.1061/\(asce\)1090-0241\(2007\)133:3\(257\)](https://doi.org/10.1061/(asce)1090-0241(2007)133:3(257))
40. Park SB, Lee JW, Kim SK (2004) Content-based image classification using a neural network. *Pattern Recognit Lett* 25:287–300
41. Park J, Santamarina JC (2019) Sand response to a large number of loading cycles under zero-lateral-strain conditions: evolution of void ratio and small-strain stiffness. *Geotechnique* 69:501–513. <https://doi.org/10.1680/jgeot.17.P.124>
42. Pasten C, Shin H, Carlos Santamarina J (2014) Long-term foundation response to repetitive loading. *J Geotech Geoenviron*

- Eng 140:1–11. [https://doi.org/10.1061/\(ASCE\)GT.1943-5606.0001052](https://doi.org/10.1061/(ASCE)GT.1943-5606.0001052)
43. Pestana JM, Whittle AJ, Salvati LA (2002) Evaluation of a constitutive model for clays and sands: part I—sand behaviour. *Int J Numer Anal Methods Geomech* 26:1097–1121. <https://doi.org/10.1002/nag.237>
  44. Pooya Nejad F, Jaksa MB (2017) Load-settlement behavior modeling of single piles using artificial neural networks and CPT data. *Comput Geotech* 89:9–21. <https://doi.org/10.1016/j.compgeo.2017.04.003>
  45. Rai R, Sahu CK (2020) Driven by data or derived through physics? A review of hybrid physics guided machine learning techniques with cyber-physical system (CPS) focus. *IEEE Access* 8:71050–71073. <https://doi.org/10.1109/ACCESS.2020.2987324>
  46. Raissi M, Karniadakis GE (2018) Hidden physics models: machine learning of nonlinear partial differential equations. *J Comput Phys* 357:125–141. <https://doi.org/10.1016/j.jcp.2017.11.039>
  47. Raissi M, Perdikaris P, Karniadakis GE (2017) Machine learning of linear differential equations using Gaussian processes. *J Comput Phys* 348:683–693. <https://doi.org/10.1016/j.jcp.2017.07.050>
  48. Raissi M, Perdikaris P, Karniadakis GE (2017) Inferring solutions of differential equations using noisy multi-fidelity data. *J Comput Phys* 335:736–746. <https://doi.org/10.1016/j.jcp.2017.01.060>
  49. Rezaie-Balf M, Kisi O (2018) New formulation for forecasting streamflow: evolutionary polynomial regression vs. extreme learning machine. *Hydrol Res* 49:939–953. <https://doi.org/10.2166/nh.2017.283>
  50. Rinehart RV, Mooney MA (2009) Measurement of roller compactor induced triaxial soil stresses and strains. *Geotech Test J* 32:347–357. <https://doi.org/10.1520/gtj101889>
  51. Roelofs R, Fridovich-Keil S, Miller J et al (2019) A meta-analysis of overfitting in machine learning. *Adv Neural Inf Process Syst* 32:9179–9189
  52. Sawicki A, Swidzinski W (1995) Cyclic compaction of soils, grains and powders. *Powder Technol* 85:97–104. [https://doi.org/10.1016/0032-5910\(95\)03013-Y](https://doi.org/10.1016/0032-5910(95)03013-Y)
  53. Seoh R (2020) Qualitative analysis of Monte Carlo dropout. *arXiv* 1–13
  54. Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. *J Big Data*. <https://doi.org/10.1186/s40537-019-0197-0>
  55. Srivastava N, Hinton G, Krizhevsky A et al (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15:1929–1958
  56. Standards Australia (2002) Soil classification tests—determination of the soil particle density of combined soil fractions—vacuum pycnometer method. In: AS 1289.3.5.2
  57. Standards Australia (2009) Soil classification tests—determination of the particle size distribution of a soil—standard method of analysis by sieving. In: AS 1289.3.6.1
  58. Standards Australia (2017) Soil compaction and density tests—determination of the dry density/moisture content relation of a soil using standard compactive effort. In: AS 1298.5.1.1
  59. Tatsuoka F (2015) Compaction characteristics and physical properties of compacted soil controlled by the degree of saturation. *Geotech Synergy Buenos Aires* 2015:40–78
  60. Tatsuoka F, Gomes Correia A (2018) Importance of controlling the degree of saturation in soil compaction linked to soil structure design. *Transp Geotech* 17:3–23. <https://doi.org/10.1016/j.trgeo.2018.06.004>
  61. Tatsuoka F, Correia AG (2016) Importance of controlling the degree of saturation in soil compaction. In: *Procedia Engineering*
  62. Van Rossum G, Drake Jr FL (1995) Python reference manual. Centrum voor Wiskunde en Informatica Amsterdam
  63. Wang Y, Salehi S (2015) Application of real-time field data to optimize drilling hydraulics using neural network approach. *J Energy Resour Technol*. <https://doi.org/10.1115/1.4030847>
  64. Waskom ML (2021) Seaborn: statistical data visualization. *J Open Source Softw* 6:3021. <https://doi.org/10.21105/joss.03021>
  65. Wersäll C, Larsson S (2013) Small-scale testing of frequency-dependent compaction of sand using a vertically vibrating plate. *Geotech Test J* 36:394–403. <https://doi.org/10.1520/GTJ20120183>
  66. Wersäll C, Larsson S, Rydén N, Nordfelt I (2015) Frequency variable surface compaction of sand using rotating mass oscillators. *Geotech Test J* 38:198–207. <https://doi.org/10.1520/GTJ20130193>
  67. Wersäll C, Nordfelt I, Larsson S (2017) Soil compaction by vibratory roller with variable frequency. *Geotechnique* 67:272–278. <https://doi.org/10.1680/jgeot.16.P.051>
  68. Wersäll C, Nordfelt I, Larsson S (2018) Resonant roller compaction of gravel in full-scale tests. *Transp Geotech* 14:93–97. <https://doi.org/10.1016/j.trgeo.2017.11.004>
  69. Wichtmann T (2005) Explicit accumulation model for non-cohesive soils under cyclic loading. PhD. Thesis, Inst für Grundbau und Bodenmechanik Phd:274
  70. Wu G, Say B, Sanner S (2017) Scalable planning with tensorflow for hybrid nonlinear domains. *arXiv* 1–11
  71. Xu G, Chang GK (2020) Continuous compaction control—mathematical models and parameter identification. In: *Springer series in geomechanics and geoengineering*, pp 563–584
  72. Yu J, Liu L, Collins RL et al (2014) Analytical problems and suggestions in the analysis of behavioral economic demand curves. *Multivariate Behav Res* 49:178–192. <https://doi.org/10.1080/00273171.2013.862491>
  73. Zhang P, Jin ZYY (2021) Modelling the mechanical behaviour of soils using machine learning algorithms with explicit formulations. *Acta Geotech*. <https://doi.org/10.1007/s11440-021-01170-4>
  74. Zhang Y, Wang R, Zhang J-M, Zhang J (2020) A constrained neural network model for soil liquefaction assessment with global applicability. *Front Struct Civ Eng* 14:1066–1082. <https://doi.org/10.1007/s11709-020-0651-2>
  75. Zhang P, Yin Z-Y, Jin Y-F (2021) State-of-the-art review of machine learning applications in constitutive modeling of soils. *Arch Comput Methods Eng*. <https://doi.org/10.1007/s11831-020-09524-z>

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