ACI MATERIALS JOURNAL TECHNICAL PAPER

Title No. 117-M115

Predicting Concrete's Strength by Machine Learning: Balance between Accuracy and Complexity of Algorithms

by B. Ouyang, Y. Song, Y. Li, F. Wu, H. Yu, Y. Wang, G. Sant, and M. Bauchy

The properties of concretes are controlled by the rate of reaction of their precursors, the chemical composition of the binding phase(s), and their structure at different scales. However, the complex and multiscale structure of the cementitious hydrates and the dissimilar rates of numerous chemical reactions make it challenging to elucidate such linkages. In particular, reliable predictions of strength development in concretes remain unavailable. As an alternative route to physics- or chemistry-based models, machine learning (ML) offers a means to develop powerful predictive models for materials using existing data. Here, it is shown that ML models can be used to accurately predict concrete's compressive strength at 28 days. This approach relies on the analysis of a large data set (>10,000 observations) of measured compressive strengths for industrially produced concretes, based on knowledge of their mixture proportions. It is demonstrated that these models can readily predict the 28-day compressive strength of any concrete based merely on the knowledge of the mixture proportions with an accuracy of approximately ±4.4 MPa (as captured by the root-mean-square error). By comparing the performance of select ML algorithms, the balance between accuracy, simplicity, and interpretability in ML approaches is discussed.

Keywords: machine learning; modeling; strength.

INTRODUCTION

Thanks to standardized, popular, and straightforward means of measurement, the compressive strength of concrete at 28 days offers a convenient metric of engineering performance that forms a key input in structural design and quality control.[1] In addition, the development of other mechanical properties (for example, stiffness and flexural/tensile strength) is correlated with compressive strength.[2] Better predictability of concrete strength offers a means to reduce the extent of overdesign of field-produced concretes. This is a straightforward means of reducing its carbon footprint as it allows for more efficient use of cement, without sacrificing performance.[3] As such, new predictive models for concrete strength could accelerate the discovery of new concretes, simultaneously offering higher performance, longer service life, lower cost, and lower carbon impact.[4,5]

Although concrete's strength is governed largely by the water-cement ratio (*w*/*c*, mass basis), it is also affected by other features, such as chemical and mineral admixtures, cement type and quantity, aggregates types and quantity, and entrained air.[6] Altogether, the high number of features influencing concrete's strength and the fact that the effects of individual features may be nonlinear, competitive, and/or non-additive make reliable prediction of strength development in concrete extremely challenging.[7] Although the development of physics- and chemistry-based predictive models would be desirable,[8] despite decades of research, no robust, accurate models that can precisely, accurately, and reliably predict concrete's strength are currently available.[9]

As an alternative route to physics- and chemistry-based models, artificial intelligence and machine learning (ML) offer an attractive option to develop data-driven models by "learning from example" based on existing data sets.[10] In this article, by building on previous studies that seek to predict concrete strength by ML,[11,12] several classes of ML algorithms (for example, polynomial regression, artificial neural network, random forest, and boosted tree) are compared in predicting concrete's strength. On the basis of these results, the trade-offs between accuracy, complexity, and interpretability offered by each ML approach are discussed.

RESEARCH SIGNIFICANCE

This paper demonstrates that ML offers a robust approach to predicting the 28-day compressive strength of concrete. Specifically, it is shown that a well-optimized random forest model successfully predicts, with an accuracy of ±4.4 MPa, the 28-day compressive strength of concrete mixtures that are hidden from the model during its training. On the basis of these results, an analysis of the balance between accuracy, complexity, and interpretability in ML models applied to small and noisy data sets is presented. The outcomes offer a powerful new basis for predicting the strength of any concrete simply based on knowledge of its mixture proportions.

BACKGROUND AND METHODS

Feature selection

This study leverages a data set (comprising 10,264 observations) of measured compressive strength values obtained from actual jobsite mixtures and their corresponding mixture proportions.[11] The mixture proportions reported reflect the actual mixture proportions—that is, based on the batch weights of industrially produced concretes that were either truck- or central-plant mixed. Furthermore, all the strength measurements reported in the data sets used ASTM C150-compliant Type I/II ordinary portland cement (OPC).

ACI Materials Journal, V. 117, No. 6, November 2020.
MS No. M-2019-457.R1, doi: 10.14359/51728128, received November 27, 2019, and reviewed under Institute publication policies. Pertinent discussion including author's closure, if any, will be published ten months from this journal's date if the discussion is received within four months of the paper's print publication.

Class F fly ash compliant with ASTM C618 was used in select cases.

To select the features to be used as inputs for the ML models considered herein, a permutation importance analysis was conducted,[13] in which the feature importance is determined by independently randomly shuffling each feature and tracking the associated loss in accuracy (wherein important features result in more significant accuracy loss). Note that this analysis is conducted based on the artificial neural networks (ANN) model presented in the section "Machine learning algorithms." Figure 1 shows the outcome of this analysis. As expected, the water-cement ratio (*w*/*c*) features the highest importance. Then, the features used herein to train the ML model on the basis of their importance were selected, while nevertheless limiting the dimensionality of the feature space by disregarding features of low importance. On the basis of this analysis, the following six most influential features controlling concrete's strength were selected, namely (in order of decreasing importance), 1) *w*/*c* (mass basis); 2) fine aggregate mass fraction; 3) water-reducing admixture (WRA) dosage; 4) coarse aggregate mass fraction; 5) fly ash mass fraction; and 6) air-entraining admixture (AEA) dosage. In contrast, due to their low importance, the following features are disregarded in this study: concrete load size, ambient temperature, and plant origin (categorial variable). Note that, for normalization purposes, all the relevant features are converted into a weight fraction (in %). The cement mass fraction is not considered in this work, because it is redundant with other features (that is, the sum of all the weight-based features is 100%). In the following, the overall ML strategy as well as the different ML algorithms that are considered and compared herein are briefly described.

Training and testing of models

To avoid any risk of overfitting, a fraction of the data points (randomly chosen) are hidden from the models and are used as a "test set" to *a posteriori* assess the accuracy of each model; that is, to minimize variance and bias. The test set is formed by randomly selecting 30% of the data points within the data set. The rest (that is, 70%) are used as a "training set," that is, to train "by example" the ML models. Further, to obtain a proper setting for the hyperparameters of each model, a fraction of the remaining training set is kept as a "validation set." However, isolating a fixed validation set would further reduce the number of points used to train the models, which can be a serious issue in the case of a small data set as that used in this work. To overcome this limitation, the *k*-fold cross-validation (CV) technique was adopted.[14] The CV technique consists in splitting the training set into *k* distinct sets, wherein the model is trained on "*k* – 1" of the folds and validated on the basis of the remaining data (that is, the last remaining fold). The results are then averaged by iteratively using each of the *k* folds for validation. Here, $k = 5$.

Optimization of model complexity

ML models must present an optimal balance between accuracy and simplicity—models that are too simple (that is, biased) tend, in turn, to be poorly accurate (that is, "underfitted"), whereas models that are too complex (that is, high variance) tend to place too much weight on the noise of the training set (that is, "overfitted"). By keeping the memory of the noise of the training set, overfitted models often show poor transferability to unknown sets of data. To avoid under- and overfitting, one needs to identify the optimal degree of complexity (for example, number of hyperparameters) for each model. Here, the degree of complexity of each model is optimized by gradually increasing its complexity and tracking the accuracy of the model predictions for both the training and the test sets. As shown in the following, although the accuracy of the training set prediction typically monotonically increases with increasing model complexity, overfitted models usually manifest themselves by a decrease in the accuracy of the test set prediction.

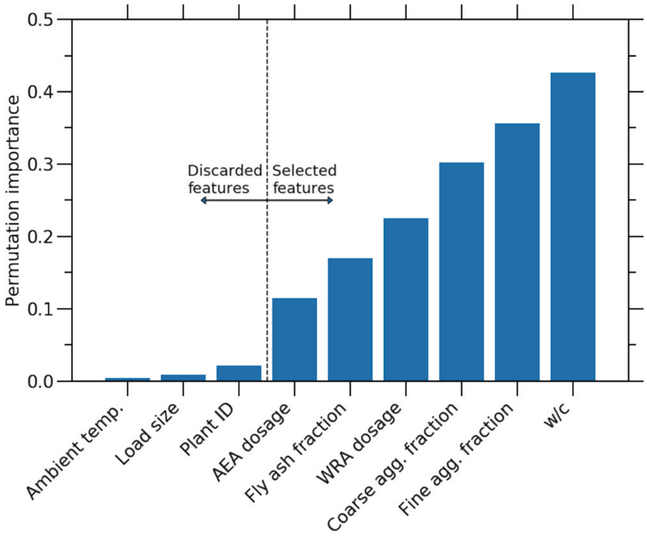

Fig. 1—Permutation importance of each feature considered herein—namely (in order of increasing importance) ambient temperature, concrete load size, plant origin (categorical variable), AEA dosage, fly ash mass fraction, WRA dosage, coarse aggregate mass fraction, fine aggregate mass fraction, and w/c *(mass basis).*

Machine learning algorithms

Polynomial regression—The first focus is on the polynomial regression (PR) method. PR is a special case of multiple linear regression that includes higher-degree polynomial terms and treats these higher-degree polynomes as other independent variables.[15] In the case of a single input, the *N*-th degree PR method can be described as

$$Y = \beta_0 + \sum_{i=1}^{N} \beta_i X^i \tag{1}$$

where *Y* is the output (here, the 28-day compressive strength); *X* is the input of the models; *N* is the maximum polynomial degree; and β_i are the parameters of the model.

The least-square method is then used to identify the coefficients β_i that minimize the sum of the squared difference between the true strength values and those predicted by the PR method (that is, *Y*) during the training phase. Equation (1) is then extended to account for additional inputs (such as *w/c* and fraction of fly ash) in the case of multivariate PR. The complexity of PR models depends on the choice of the number *N* of polynomial degrees considered during training.

Artificial neural networks—ANNs aim to mimic the learning process of human brains.[16] ANN models consist of an input layer that is connected to an output layer through some "hidden" layers of neurons. Each neuron takes as inputs the signals from the previous layer and produces a new output (to be used as input by the neurons from the next layer). The output y_i of a neuron *i* in one of the hidden layers is calculated as[17]

$$y_i = s\left(\sum_1^M w_i x_i + T_i^{hid}\right) \tag{2}$$

where *s*( ) is an activation function; *M* is the number of input neurons in the previous layer; x_i are the input values; w_i is the weight associated with each edge of the network; and T_i^{hid} is the threshold term (bias) of hidden neurons.

To capture the nonlinear relationship between concrete mixture proportions and compressive strength values, a sigmoid function as activation function was adopted herein

$$s(u) = \frac{1}{1+e^{-u}} \tag{3}$$

The resilient backpropagation (BP) algorithm is adopted to train the neural network model, which allows the network to iteratively learn from its errors.[18] The BP algorithm is an efficient method that tunes the weight, w_i, of the ANN model by calculating the gradient of the loss function E_{loss} (refer to Eq. (4)). To measure the error between the predicted and real strength outputs after a training sample has propagated through the network, the squared Euclidean distance was used to calculate the loss function E_{loss} over *n* training outputs as

$$E_{loss} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - Y_i'\right)^2 \tag{4}$$

where Y_i and Y_i' are the predicted and real strength outputs, respectively. Then, after a given number *k* of iterations, each weight w_i is modified by applying an increment

$$w_i^{(k+1)} = w_i^{(k)} + \Delta^{(k)} w_i \tag{5}$$

where $w_i^{(k+1)}$ is the updated weight for the step *k* + 1; $w_i^{(k)}$ is the initial weight before update; and $\Delta^{(k)} w_i$ is the increment, which is calculated by steepest descent gradient in the error function as

$$\Delta^{(k)} w_i = -\gamma_i^{(k)} \text{sign}(\nabla_i E^{(k)}) \tag{6}$$

where sign( ) is the "sign function"; $\nabla_i E^k$ is the partial derivative (that is, gradient) of the error function $E^{(k)}$ with respect to the weights *w* at the *k*-th iteration; and $\gamma_i^{(k)}$ is the learning rate data used for training.

The model is iteratively refined until the relative change in the loss function becomes less than 10^{-4}. The degree of complexity of ANN models is characterized by the number of nonzero weights, which depends on the number of layers and neurons per layer.

Random forest—The random forest (RF) method relies on building of a "forest"—that is, an ensemble of decision trees.[19] The RF approach creates a tree by randomly choosing *n* samples from the training set (bootstrap method). Then, at each node, it uses a randomly selected subset of variables to choose the best split to construct trees. RF runs input data on all n_t trees and yields a prediction that is the average of all the values returned by each tree

$$Y = \frac{1}{n_t}\sum_{i=1}^{n_t} Y_i(X) \tag{7}$$

where $Y_i(X)$ is the individual value predicted by one of the trees for an input vector *X*; and *Y* is the overall output of the RF model with n_t trees.[20] The degree of complexity of RF models is characterized by the number of variables in the random subset and trees in the forest.

Boosted tree—In boosted tree (BT),[21] trees are grown sequentially, each successive tree grown so as to correct the errors contained in the previous trees. At every iteration, a decision tree is fitted to the negative gradient of the loss function and added to the output value obtained from the previous iteration. In details, in the first iteration, a base decision tree learner *F*(*X*) is fitted to the training set (*X*,*Y*′), where X denotes the input variables, and *Y*′ denotes the true concrete strength. Then, for the next iteration, *i*, the pseudo-residuals r_i are calculated as

$$r_i = -\left[\frac{\partial L\left(Y, F_i(X)\right)}{\partial F_i(X)}\right] \tag{8}$$

where *L*( ) is the loss function. Then, a new tree learner $h_i(X)$ is fitted to the pseudo-residuals using the training set (*X*, r_i). The multiplier coefficient, m_i, is then computed by solving the following optimization problem

$$m_i = \text{argmin}[L(Y, F_{i-1}(X) + mh_i(X)] \tag{9}$$

where *m* is a constant (taken as 1 here). The decision tree learner is then updated as

$$F_i(X) = F_{i-1}(X) + \lambda m_i h_i(X) \tag{10}$$

where λ is the learning rate (taken as 0.01 herein), which is applied at each step to help prevent the model from becoming overfitted. The model is refined until convergence.

Determination of model accuracy

Finally, the accuracy of each model (with different degrees of complexity) is assessed by computing the root-mean-square error (RMSE), the mean absolute percentage error (MAPE), and the coefficient of determination (R^2) metrics. The RMSE factor measures the average Euclidian distance between the predicted and real output values as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - Y_i'\right)^2} \tag{11}$$

where Y_i and Y_i' are the predicted and real output values, respectively; and n is the number of samples in the data set. The MAPE factor measures the average percent error between the predicted and real output values as

$$\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \left(Y_i - Y_i' \right) / Y_i' \right| \tag{12}$$

The RMSE has the property of having the same units as that of the output values and, hence, can be used to estimate the accuracy of the compressive strength values predicted by each model (namely, lower RMSE values indicate higher accuracy). Here, the RMSE of the training and test sets is used to determine the optimal degree of complexity for each ML model.

To complement the RMSE and MAPE metrics, the R^2 determination factor was computed, which captures the response variable variation. This factor can be used to quantify how close the output data are to the fitted line. $R^2 = 1$ indicates a perfect prediction, while smaller values indicate less accurate predictions. Here, we use the R^2 factor to compare the performances of each different ML algorithms (once the degree of complexity has been optimized based on the RMSE).

Finally, the error distribution was computed—that is, the distribution of the difference between measured and predicted strength. Then determined were: 1) the mean value of the error distribution (that is, average error), which offers a measure of the systematic bias of a given model, that is, its propensity to under- or overestimate strength; and 2) the 90% (95%) confidence interval, wherein the strength that is predicted for an unknown concrete shows a 90% (95%) probability of falling within this error interval.

RESULTS

Polynomial regression

First assessed was the ability of the different ML algorithms considered herein to predict the relationship between concrete mixture proportions (inputs) and compressive strength (output). The first focus was on the performance of PR. Figure 2(a) shows the RMSE offered by PR for the training and test sets as a function of the maximum polynomial degree considered in the model. As expected, it was observed that the RMSE of the training set decreases upon increasing polynomial degree (that is, increasing model complexity). This indicates that as the model becomes more complex, it becomes able to better interpolate the training set. In contrast, an increase in the RMSE was observed when the polynomial degree is equal to 1 or 2—which signifies that, in this domain, the model is underfitted (so that the model is too simple to properly interpolate the training set). This confirms, as shown extensively in the literature,[22,23] that linear models based on additive relationships are intrinsically unable to properly describe the linkages between concrete mixture proportions and compressive strength.

On the other hand, it was observed that the RMSE of the test set initially decreases with increasing polynomial degree, shows a minimum for degree 3, and eventually increases with increasing degree. This demonstrates that the models incorporating some polynomial terms that are strictly larger than 3 are overfitted. This arises from the fact that, in the case of high degrees, the model starts to fit the noise of the training set rather than the "true" overall trend. These results exemplify 1) how the accuracy of the training set interpolation allows identification of the minimum level of model complexity that is required to avoid underfitting; and 2) how the accuracy of the test set prediction allows tracking of the maximum level of model complexity before overfitting. Overall, the optimal polynomial degree (here found to be 3) manifests itself by a minimum in the RMSE of the test set.

Next, the focus is on assessing the accuracy of the predictions offered by the best PR model identified herein (that is, with a maximum polynomial degree of 3). Figure 2(b) shows a comparison between the true compressive strength values of the test set (that is, which are unknown to the model) and those predicted by the ML model. The R^2 factor of the test set was found to be 0.596. This indicates that, even in the case of a simple algorithm like PR, ML is able to identify the underlying patterns represented in the data set. The error distribution (that is, the distribution of the difference between the predicted and true strength values of the test set) of the best PR model (that is, with a maximum polynomial degree of 3) is shown in Fig. 2(c). It was observed that the distribution does not exhibit any notable bias (which would manifest by a nonzero average error—here the mean error is 0.007 MPa) and presents a standard deviation of 4.52 MPa. The 90% and 95% confidence intervals extend from –7.43 to 7.43 MPa and –8.86 to 8.86 MPa, respectively. This indicates that there is a 90% probability that the predicted strength (for an unknown concrete) is within ±7.43 MPa of the true (measured) strength.

Artificial neural network

Next, focus is on the outcomes of the ANN algorithm. Figure 3(a) shows the RMSE offered by ANN for the training and test sets as a function of the number of neurons (that is, which characterizes the complexity of the model). Note that, here, this work is restricted to ANN architectures comprising a single hidden layer. In contrast to the outcomes of PR, it was observed that ANN does not yield any noticeable signature of overfitting at high model complexity, which would manifest itself by an increase in the RMSE of the test set. Here, in contrast, the RMSE of the test set only eventually shows a plateau with an increasing number of neurons. The optimal degree of complexity was selected (herein, seven neurons) as the one for which the RMSE of the test set becomes less than one standard deviation away from the minimum RMSE (that is, in the plateau regime), wherein the standard deviation is calculated on the basis of the RMSE of each fold during CV.

As for the accuracy of the optimal ANN model (that is, with seven neurons), Fig. 3(b) shows a comparison between the true compressive strength of the test set and that predicted by the ML model. The R^2 factor of the test set was 0.591. This suggests that, in the present case, the ability of the ANN algorithm to offer accurate predictions of the compressive strength values of unknown concretes (that is, test set) is slightly lower than in the case of PR. Figure 3(c) shows the error distribution. Again, no notable bias was observed

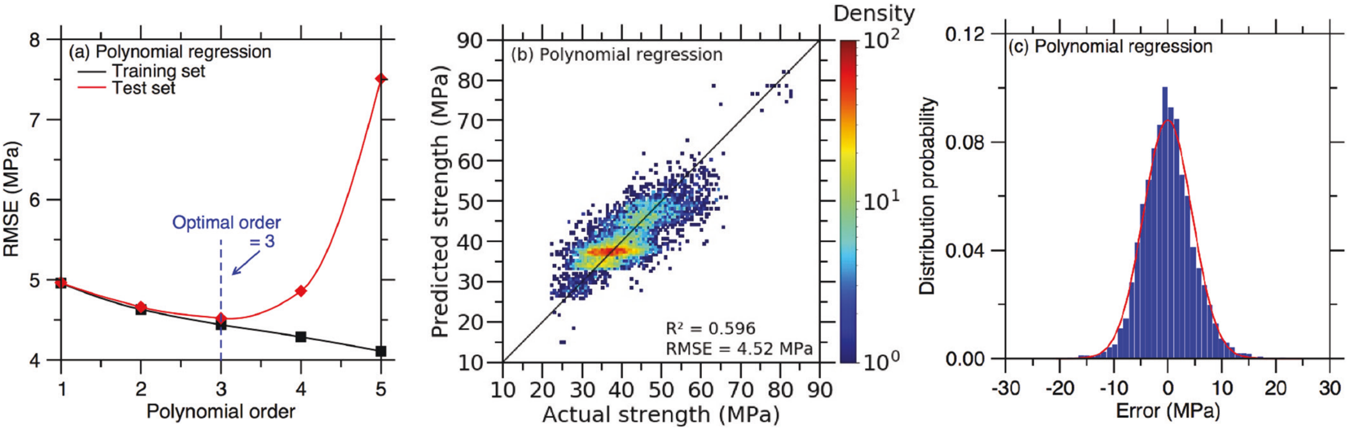

Fig. 2—(a) Accuracy (as captured by RMSE value) of PR models as function of maximum polynomial degree considered in each model, as obtained for training and test sets, respectively. Optimal polynomial order is chosen as that for which RMSE of test set is minimum. Line is to guide eye. (b) Comparison between true compressive strength values of test set and those predicted by optimal PR model (with degree of 3). and (c) Error distribution of predicted strength values offered by optimal PR model. Data are fitted by Gaussian distribution.

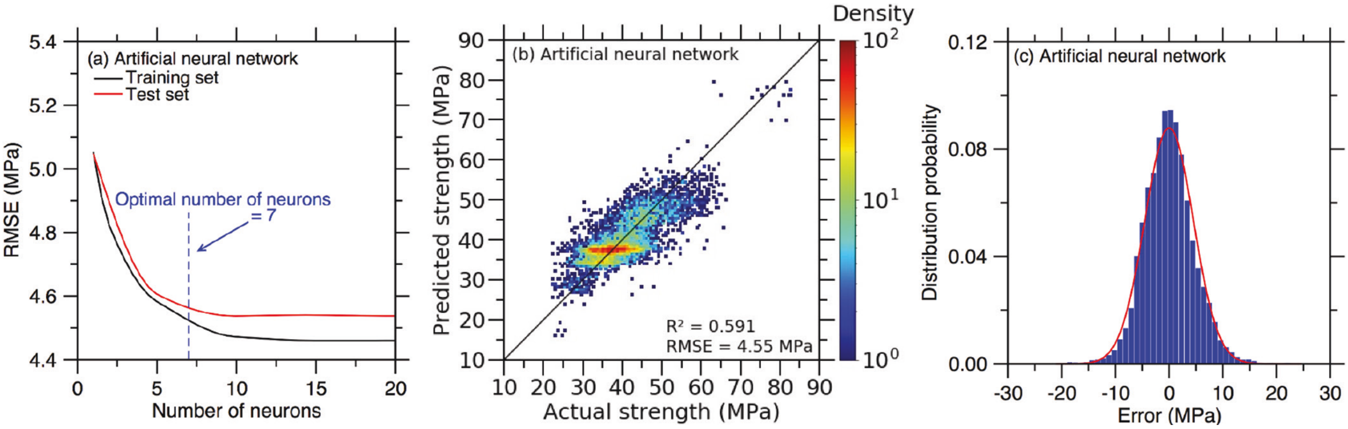

Fig. 3—(a) Accuracy (as captured by RMSE value) of ANN models as function of number of neurons considered in each model, as obtained for training and test sets, respectively. Optimal number of neurons is determined as that one for which RMSE of test set is one standard deviation away from minimum RMSE (that is, in plateau regime). (b) Comparison between true compressive strength values of test set and those predicted by optimal ANN model (with seven neurons). (c) Error distribution of predicted strength values offered by optimal ANN model. Data are fitted by Gaussian distribution.

(the mean error is 0.009 MPa), and the standard deviation of the distribution is 4.53 MPa. The 90% and 95% confidence intervals extend from –7.45 to 7.45 MPa and –8.88 to 8.88 MPa, respectively.

Boosted tree

Next, focus is on the outcomes of the BT algorithm. Figure 4(a) shows the RMSE offered by BT for the training and test sets as a function of the number of trees (that is, which characterizes the complexity of the model). Overall, as previously observed in the cases of ANN, BT does not yield any noticeable overfitting at high model complexity. Here, 700 was selected as being the optimal number of trees (by adopting the same criterion as that used in the case of ANN).

To characterize the accuracy of the optimal BT model (that is, with 700 trees), Fig. 4(b) shows a comparison between the true compressive strength of the test set and those predicted by the ML model. The R^2 factor of the test set was found to be 0.619. This indicates that, here, BT offers an increased ability to predict the compressive strength of unknown concretes as compared with the PR and ANN models. Figure 4(c) shows the corresponding error distribution. Note that, in the present case, BT presents a notable bias because the mean error is –0.322 MPa. This indicates that this method yields a nonnegligible systematic error and, overall, tends to underestimate the compressive strength. Such bias may arise from the high complexity of the present BT model and its propensity to overfit the data set.

The standard deviation of the error distribution is 4.38 MPa, and the 90% and 95% confidence intervals extend from –7.20 to 7.20 MPa and –8.88 to 8.28 MPa, respectively. The asymmetry of the 95% confidence interval is another signature of the bias of the BT model.

Random forest

Finally, the performance of the RF algorithm is investigated. Figure 5(a) shows the RMSE offered by RF for the

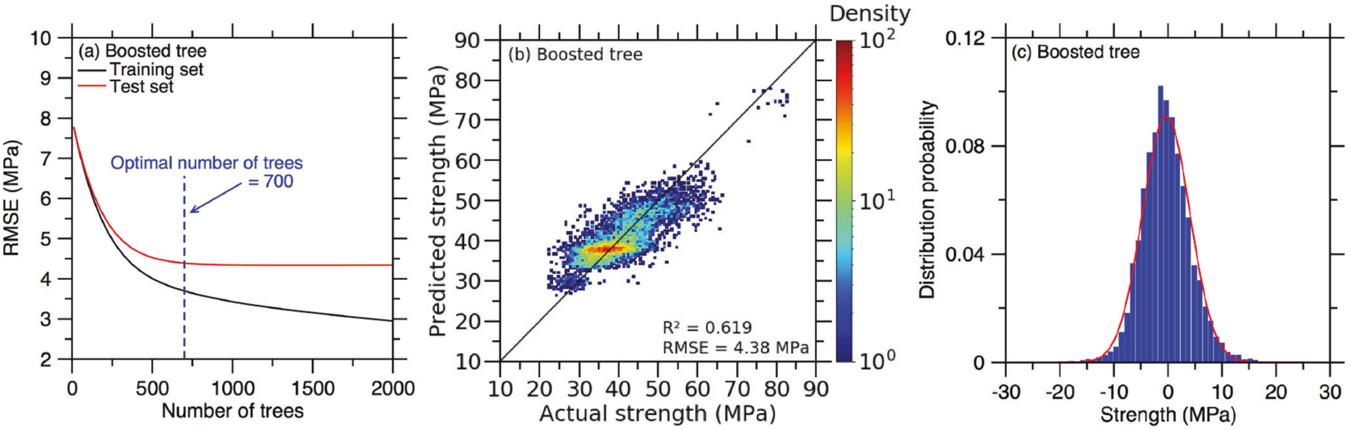

Fig. 4—(a) Accuracy (as captured by RMSE value) of BT models as function of number of trees considered in each model, as obtained for training and test sets, respectively. Optimal number of trees is determined as that for which RMSE of test set is one standard deviation away from minimum RMSE (that is, in plateau regime). (b) Comparison between true compressive strength values of test set and those predicted by optimal BT model (with 700 trees). (c) Error distribution of predicted strength values offered by optimal BT model. Data are fitted by Gaussian distribution.

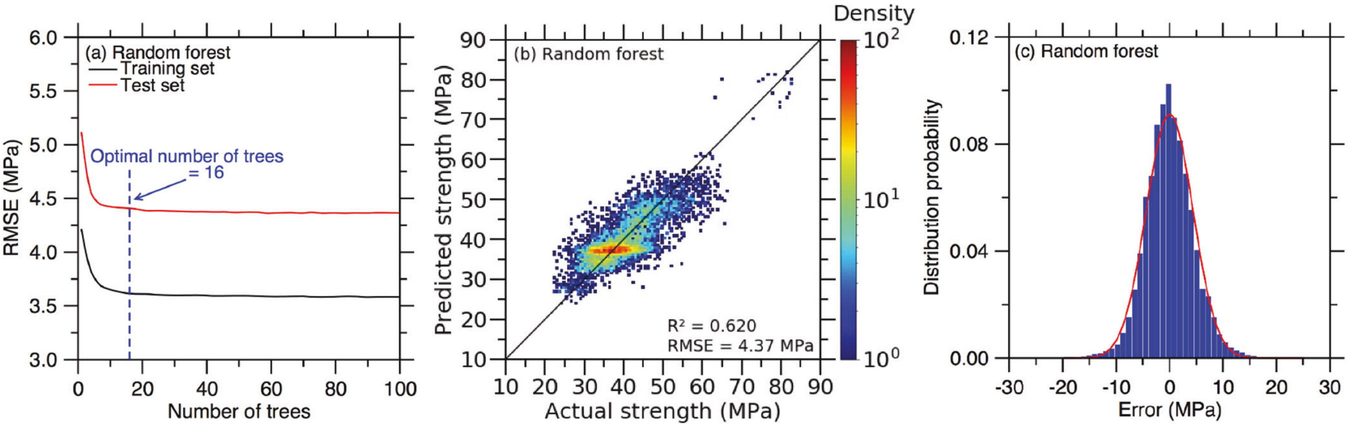

Fig. 5—(a) Accuracy (as captured by RMSE value) of RF models as function of number of trees considered in each model, as obtained for training and test sets, respectively. Optimal number of trees is determined as that for which RMSE of test set is one standard deviation away from minimum RMSE (that is, in plateau regime). (b) Comparison between true compressive strength values of test set and those predicted by optimal RF model (with 16 trees). (c) Error distribution of predicted strength values offered by optimal RF model. Data are fitted by Gaussian distribution.

training and test sets as a function of the number of trees (that is, which characterizes the complexity of the model). As observed in the case of ANN and BT, RF does not yield any noticeable signature of overfitting at high model complexity, that is, the RMSE of the test set only plateaus on increasing number of trees (Fig. 5). Here, 16 was selected as being the optimal number of trees (by adopting the same criterion as that used in the case of ANN and BT).

To quantify the accuracy of the predictions of the optimal RF model (that is, with 16 trees), Fig. 5(b) shows a comparison between the true compressive strength of the test set and those predicted by the ML model. The R^2 factor for the test set was found to be 0.620. Overall, among all the ML algorithms considered in this study, the RF algorithm offers the most accurate compressive strength predictions for unknown concrete (test set). Figure 5(c) shows the obtained error distribution. In contrast to BT, RF does not yield any notable bias because the mean error is –0.013 MPa. The standard deviation of the distribution is 4.39 MPa, and the 90% and 95% confidence intervals extend from –7.22 to 7.22 MPa and –8.60 to 8.60 MPa, respectively.

DISCUSSION

The following compares the overall performance of the different ML algorithms used in this study. An ideal ML model should offer: 1) high "accuracy" (that is, low error when predicting the strength of unknown concretes); 2) low "complexity"; and 3) good "interpretability." These features are often mutually exclusive, because more complex models are more accurate and less interpretable. However, accuracy is not the only criterion used to select an optimal model, because simpler and more interpretable models are 1) less likely to overfit small data sets; 2) usually more computationally efficient; and 3) their enhanced interpretability (that is, by not acting as a "black box") can potentially offer some mechanistic understanding about the nature of the relationship between inputs and outputs.[24-26]

Table 1—Comparison between levels of accuracy (captured by R^2, 95% confidence interval, mean error bias, and MAPE), complexity (roughly captured by number of nonzero parameters in PR, number of trees in RF and BT, and number of weight coefficients and threshold terms in ANN), and overall interpretability of model

ML algorithms	R^2	95% confidence interval, MPa	Systematic bias, MPa	MAPE, %	Complexity and interpretability
PR	0.596	[–8.86, 8.86]	0.007	9.00	Intermediate (41) High interpretability
ANN	0.591	[–8.88, 8.88]	0.009	9.06	Intermediate (49) Low interpretability
BT	0.619	[–8.88, 8.28]	–0.322	8.89	High (700) Intermediate interpretability
RF	0.620	[–8.60, 8.60]	–0.013	8.74	Low (16) Intermediate interpretability

To explore the balance between accuracy, complexity, and interpretability, Table 1 summarizes for each ML algorithm the coefficient of determination R^2 of the test set, the 95% confidence interval of the test set, the mean error bias (which all capture the accuracy of the model when predicting unknown compressive strength data), the MAPE of the test set, the degree of complexity (roughly captured by the number of nonzero parameters in PR, the number of trees in RF and BT, and the number of weight coefficients and threshold terms in ANN), and the overall degree of interpretability.

On the one hand, the PR and ANN algorithms offer a fairly similar level of accuracy. This indicates that, despite its complex architecture, the ANN model does not offer any improvement in accuracy as compared with a simpler analytical polynomial model. This may arise from the fact that the number of data points per dimension is here too small to benefit from the ability of ANN to model complex, nonlinear functions.[25,27] In turn, thanks to its analytical form, PR offers significantly higher interpretability than ANN, which is essentially a black box for practical purposes.

On the other hand, we note that both the RF and the BT methods offer a notable increase in accuracy as compared with PR and ANN. This likely arises from the superior resistance of tree-based models to overfitting, which is key in the case of small and noisy data sets as the one considered herein. Nevertheless, despite its significantly higher degree of complexity (as compared with RF), BT does not offer any improvement in accuracy and, worse, shows a nonnegligible systematic bias when predicting the strength of unknown concrete. Overall, it is concluded that RF offers the most optimal balance between accuracy, complexity, and interpretability among the ML algorithms considered herein.

Overall, all the ML algorithms considered herein exhibit a MAPE that is lower than 10%, which compares favorably with the intrinsic uncertainty of the compressive strength experimental data obtained by ASTM C39.[28] This suggests that, overall, all these algorithms offer good predictive power, with a degree of accuracy that approaches the upper limit set by the uncertainty of the experimental data. Although incremental increases in accuracy can certainly be beneficial, other aspects (for example, computing burden, complexity, and interpretability) should be considered when comparing the performance of different ML approaches.

CONCLUSIONS

This study demonstrates that machine learning (ML) can offer a robust prediction of the 28-day compressive strength of concrete—by learning solely from the data set, without the need for any prerequisite domain knowledge. All the ML approaches investigated here achieved a satisfactory test set coefficient of determination that is between 0.59 and 0.62. As expected, the existence of a general balance between model accuracy and complexity was reported. For instance, the random forest (RF) model successfully predicts the strength of unknown concretes with an RMSE accuracy of approximately 4.4 MPa. In contrast, a simpler polynomial regression (PR) model achieves a lower RMSE accuracy of approximately 4.5 MPa. It should be noted, however, that higher model complexity does not systematically result in higher accuracy. For instance, despite its high complexity, the artificial neural network (ANN) presented herein yields an accuracy that is lower than that offered by the polynomial model. This highlights the fact that, despite the current popularity of advanced learning models (for example, deep learning), simpler models may still offer competitive, if not superior, accuracy—and, hence, such simple models should not be disregarded without investigation. This may be especially true for small and noisy data sets. This is significant because, in contrast to complex "black box" models, simpler models are often more interpretable and, thus, more suitable for inverse design problems—for example, predicting optimal concrete formulations with maximum strength. More generally, interpretable models offer some exclusive opportunities to decipher the underlying physics or chemistry that governs concrete's behavior. Overall, it is suggested that, rather than selecting ML algorithms solely on the basis of their balance between accuracy and complexity, one should also take into account their interpretability, which will be key to advancing the state of the art in concrete science.

It should be noted that, once trained, regression ML models can be used as surrogates to inform the "inverse-design" optimization approach, for example, to accelerate the discovery of optimized concrete mixture proportions featuring minimum cost, maximum strength, and/or minimum carbon footprint. In this regard, the level of complexity of the trained model can impact the nature and efficiency of such optimization tasks. First, simpler models are often faster to compute, which, in turn, can greatly reduce the associated computational burden (especially in the case of brute-force searches). Second, unlike simple analytical models (for example, PR), tree-based models are usually not differentiable, making it challenging to adopt gradient-based optimization methods (for example, gradient descend). Finally, thanks to their higher interpretability, simpler models offer more direct access to the importance of each feature. This knowledge can be used to filter out features of low importance and, in turn, reduce the dimensionality of the feature space—which is key to facilitating inverse-design approaches.[29]

AUTHOR BIOS

Boya Ouyang *is a PhD Student in the Department of Materials Science and Engineering at the University of California, Los Angeles (UCLA), Los Angeles, CA. He received his BS from Central South University, Changsha, Hunan, China, and his MS from Stanford University, Stanford, CA. His research interests include applying artificial intelligence and machine learning to cement and concrete manufacturing.*

Yu Song *is a Postdoctoral Researcher in the Department of Civil and Environmental Engineering at UCLA. He received his BS in civil engineering from Beijing University of Civil Engineering and Architecture, Beijing, China, and his MS and PhD in construction materials from the University of Illinois Urbana-Champaign, Urbana, IL. His research interests include the characterization and modeling of concrete materials.*

Yuhai Li *is a PhD Student in the Department of Civil and Environmental Engineering at UCLA, where he received his BS in computer science. His research interests include data-driven informatics and deep learning applied to materials science.*

Feishu Wu *is a Research Assistant in the Physics of AmoRphous and Inorganic Solids Laboratory (PARISlab) at UCLA, where she received her BS in statistics. Her research interests include using machine learning to identify the most influential features governing concrete strength.*

Huizi Yu *is an Undergraduate Student double majoring in statistics and economics at UCLA. Her research interests include identifying the features governing the propensity for under- or overdesign in concrete.*

Yongzhe Wang *is an Undergraduate Student in the Department of Statistics at UCLA. His research interests include using machine learning to predict the strength development kinetics in cement-based materials.*

ACI member **Gaurav Sant** *is a Professor and Henry Samueli Fellow in the Department of Civil and Environmental Engineering, and Materials Science and Engineering at UCLA. He received his BS, MS, and PhD in civil engineering from Purdue University, West Lafayette, IN. Sant is a member of ACI Committees 231, Properties of Concrete at Early Ages, and 236, Material Science of Concrete. He is a recipient of the ACI Walter P. Moore, Jr. Faculty Achievement Award and the Jean-Claude Roumain Innovation in Concrete Award. His research interests include reducing the CO_2 impact and intensity of the materials of modern construction.*

Mathieu Bauchy *is an Associate Professor in the Department of Civil and Environmental Engineering at UCLA, Los Angeles, CA. He received his BS in physics and his MS in condensed matter from the Ecole Normale Supérieure, Cachan, France, and his PhD in condensed matter from the Université Pierre et Marie Curie, Paris, France. His research interests include applying atomistic simulations and machine learning to decipher the behavior of engineering materials and systems.*

ACKNOWLEDGMENTS

The authors acknowledge financial support for this research provided by the U.S. Department of Transportation through the Federal Highway Administration (Grant #: 693JJ31950021) and the U.S. National Science Foundation (DMREF: 1922167).

NOTATION

E_{loss} = loss function in ANNs
$F(X)$ = output of BT models
$h_i(X)$ = new tree learner that fits to pseudo-residuals in BT models
$L(\,)$ = loss function in BT models
M = number of input neurons in previous layer in ANN model
m_i = multiplier in BT models
n_t = number of trees in RF models
r_i = pseudo-residuals in BT models
$s(\,)$ = activation function of ANN model
sign( )= sign function
T_i^{hid} = threshold term (bias) of hidden neurons
w_i = weight associated with each edge of ANN
$w_i^{(k)}$ = initial weight before update
$w_i^{(k+1)}$ = updated weight for step $k + 1$
X = input variables
Y = output of ML models
Y' = concrete strength true values
$Y_i(X)$ = individual value predicted by tree for input vector X
y_i = output of neuron i in ANN
β_i = parameters of polynomial regression model
$\Delta^{(k)}w_i$ = increment in weight
$\gamma_i^{(k)}$ = learning rate data used for training in ANNs
λ = learning rate in BT models
$\nabla_i E^k$ = partial derivative of error function (k) with respect to weights w at k-th iteration

REFERENCES

1. Rodríguez de Sensale, G., "Strength Development of Concrete with Rice-Husk Ash," *Cement and Concrete Composites*, V. 28, No. 2, 2006, pp. 158-160. doi: 10.1016/j.cemconcomp.2005.09.005

2. Ashour, S. A., and Wafa, F. F., "Flexural Behavior of High-Strength Fiber Reinforced Concrete Beams," *ACI Structural Journal*, V. 90, No. 3, May-June 1993, pp. 279-287.

3. Purnell, P., and Black, L., "Embodied Carbon Dioxide in Concrete: Variation with Common Mix Design Parameters," *Cement and Concrete Research*, V. 42, No. 6, 2012, pp. 874-877. doi: 10.1016/j.cemconres.2012.02.005

4. Biernacki, J. J.; Bullard, J. W.; Sant, G.; Brown, K.; Glasser, F. P.; Jones, S.; Ley, T.; Livingston, R.; Nicoleau, L.; Olek, J.; Sanchez, F.; Shahsavari, R.; Stutzman, P. E.; Sobolev, K.; and Prater, T., "Cements in the 21st Century: Challenges, Perspectives, and Opportunities," *Journal of the American Ceramic Society*, V. 100, No. 7, 2017, pp. 2746-2773. doi: 10.1111/jace.14948

5. Provis, J. L., "Grand Challenges in Structural Materials," *Frontiers in Materials*, V. 2, No. 31, 2015.

6. Wild, S.; Sabir, B. B.; and Khatib, J. M., "Factors Influencing Strength Development of Concrete Containing Silica Fume," *Cement and Concrete Research*, V. 25, No. 7, 1995, pp. 1567-1580. doi: 10.1016/0008-8846(95)00150-B

7. Burris, L. E.; Alapati, P.; Moser, R. D.; Ley, T.; Berke, N.; and Kurtis, K. E., "Alternative Cementitious Materials: Challenges and Opportunities," *Durability and Sustainability of Concrete Structures*, SP-305, American Concrete Institute, Farmington Hills, MI, 2015, 10 pp.

8. Khoury, G. A.; Majorana, C. E.; Pesavento, F.; and Schrefler, B. A., "Modelling of Heated Concrete," *Magazine of Concrete Research*, V. 54, No. 2, 2002, pp. 77-101. doi: 10.1680/macr.2002.54.2.77

9. Breysse, D., and Martínez-Fernández, J. L., "Assessing Concrete Strength with Rebound Hammer: Review of Key Issues and Ideas for More Reliable Conclusions," *Materials and Structures*, V. 47, No. 9, 2014, pp. 1589-1604. doi: 10.1617/s11527-013-0139-9

10. Yeh, I.-C., "Modeling of Strength of High-Performance Concrete Using Artificial Neural Networks," *Cement and Concrete Research*, V. 28, No. 12, 1998, pp. 1797-1808. doi: 10.1016/S0008-8846(98)00165-3

11. Young, B. A.; Hall, A.; Pilon, L.; Gupta, P.; and Sant, G., "Can the Compressive Strength of Concrete Be Estimated from Knowledge of the Mixture Proportions?: New Insights from Statistical Analysis and Machine Learning Methods," *Cement and Concrete Research*, V. 115, 2019, pp. 379-388. doi: 10.1016/j.cemconres.2018.09.006

12. Oey, T.; Jones, S.; Bullard, J. W.; and Sant, G., "Machine Learning Can Predict Setting Behavior and Strength Evolution of Hydrating Cement Systems," *Journal of the American Ceramic Society*, V. 103, No. 1, 2020, pp. 480-490. doi: 10.1111/jace.16706

13. Breiman, L., "Random Forests," *Machine Learning*, V. 45, No. 1, 2001, pp. 5-32. doi: 10.1023/A:1010933404324

14. Stone, M., "Cross-Validatory Choice and Assessment of Statistical Predictions," *Journal of the Royal Statistical Society. Series B. Methodological*, V. 36, No. 2, 1974, pp. 111-133. doi: 10.1111/j.2517-6161.1974.tb00994.x

15. Sinha, P., "Multivariate Polynomial Regression in Data Mining: Methodology, Problems and Solutions," *International Journal of Scientific and Engineering Research*, V. 4, No. 12, 2013.

16. Guo, J.; Xu, N.; Qian, K.; Shi, Y.; Xu, K.; Wu, Y.; and Alwan, A., "Deep Neural Network Based i-Vector Mapping for Speaker Verification Using Short Utterances," *Speech Communication*, V. 105, 2018, pp. 92-102. doi: 10.1016/j.specom.2018.10.004

17. Wasserman, P. D., *Advanced Methods in Neural Computing*, first edition, John Wiley & Sons, New York, 1993.

18. Li, J.; Cheng, J.; Shi, J.; and Huang, F., "Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement," *Advances in Computer Science and Information Engineering*, D. Jin and S. Lin, eds., Springer, Berlin, Germany, 2012, pp. 553-558.

19. Liaw, A., and Wiener, M., "Classification and Regression by randomForest," *R News*, V. 2/3, 2002, p. 6

20. Segal, M. R., "Machine Learning Benchmarks and Random Forest Regression," *UCSF: Center for Bioinformatics and Molecular Biostatistics*, Apr. 2004.

21. Chen, T., and Guestrin, C., "XGBoost: A Scalable Tree Boosting System," *Proceedings*, Twenty Second ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, ACM, 2016, pp. 785-794.

22. Ozturan, M.; Kutlu, B.; and Ozturan, T., "Comparison of Concrete Strength Prediction Techniques with Artificial Neural Network Approach," *Building Research Journal*, V. 56, No. 1, 2008, pp. 23-36.

23. Akande, K. O.; Owolabi, T. O.; Twaha, S.; and Olatunji, S. O., "Performance Comparison of SVM and ANN in Predicting Compressive Strength of Concrete," *IOSR Journal of Computer Engineering*, V. 16, No. 5, 2014, pp. 88-94. doi: 10.9790/0661-16518894

24. Liu, H.; Fu, Z.; Li, Y.; Sabri, N. F. A.; and Bauchy, M., "Balance between Accuracy and Simplicity in Empirical Forcefields for Glass Modeling: Insights from Machine Learning," *Journal of Non-Crystalline Solids*, V. 515, 2019, pp. 133-142. doi: 10.1016/j.jnoncrysol.2019.04.020

25. Liu, H.; Zhang, T.; Anoop Krishnan, N. M.; Smedskjaer, M. M.; Ryan, J. V.; Gin, S.; and Bauchy, M., "Predicting the Dissolution Kinetics of Silicate Glasses by Topology-Informed Machine Learning," *Nature Partner Journals Materials Degradation*, V. 3, No. 1, 2019, pp. 1-12. doi: 10.1038/s41529-019-0094-1

26. Yang, K.; Xu, X.; Yang, B.; Cook, B.; Ramos, J.; Anoop Krishnan, N. M.; Smedskjaer, M. M.; Hoover, C.; and Bauchy, M., "Predicting the Young's Modulus of Silicate Glasses Using High-Throughput Molecular Dynamics Simulations and Machine Learning," *Scientific Reports*, V. 9, No. 1, 2019, pp. 1-11. doi: 10.1038/s41598-019-45344-3

27. Bishnoi, S.; Singh, S.; Ravinder, R.; Bauchy, M.; Gosvami, N. N.; Kodamana, H.; and Anoop Krishnan, N. M., "Predicting Young's Modulus of Oxide Glasses with Sparse Datasets Using Machine Learning," *Journal of Non-Crystalline Solids*, V. 524, 2019, p. 119643 doi: 10.1016/j.jnoncrysol.2019.119643

28. ASTM C39/C39M-12a, "Standard Test Method for Compressive Strength of Cylindrical Concrete Specimens," ASTM International, West Conshohocken, PA, 2012, 7 pp.

29. Fonti, V., and Belitser, E., "Feature Selection Using Lasso," *VU Amsterdam: Research Paper in Business Analytics*, Vrije Universiteit, Amsterdam, Netherlands, 2017, pp. 1-25.