Hindawi
Advances in Civil Engineering
Volume 2021, Article ID 6658932, 10 pages
https://doi.org/10.1155/2021/6658932

Research Article

Prediction of Compressive Strength of Concrete and Rock Using an Elementary Instance-Based Learning Algorithm

Shun-Chieh Hsieh

Department of Land Management and Development, Chang Jung Christian University, Tainan City 71101, Taiwan

Correspondence should be addressed to Shun-Chieh Hsieh; sch@mail.cjcu.edu.tw

Received 12 December 2020; Revised 2 March 2021; Accepted 23 March 2021; Published 5 April 2021

Academic Editor: Weerachart Tangchirapat

The use of machine learning techniques to predict material strength is becoming popular. However, not much attention has been paid to instance-based learning (IBL) algorithms. Therefore, in order to predict material strength, as the direct method by conducting tests is time-consuming and expensive and experimental errors are inevitable, an indirect method based on elementary instance-based learning algorithm was proposed. The standard k-nearest neighbors (k-NN) with cross-validation were utilized to develop compressive strength prediction models for some concretes and rocks by considering indirect parameters such as physical and mechanical parameters. Results on applying this method to datasets from literature studies show that the values of RMSE for k-NN are modest, indicating adequacy to predict compressive strength with comprehensive range values of predictors. Additionally, the R^2-values of the k-NN models were high. In other words, the models were able to explain the variance in compressive strength for data with a wide range of input values.

1. Introduction

The relationship between material strength and its mixture and process can be complex. As such, the relationship which is usually determined empirically using experiments is problematic. Moreover, physical, mineralogical-petrographic, index, and mechanical tests are time-consuming and expensive, and experimental errors are inevitable. Machine learning (ML) techniques are increasingly used to model the strength of materials, such as concrete and rock and have become an important research area [1–10]. Additionally, the results of these studies can help engineers and practitioners determine the key components related to material strength performance.

Ensemble models can provide higher performance in predicting material strength compared to individual models. Nevertheless, no model has been proven to be superior all the time [11]. Moreover, if the link between input and output is important for description, then the ensemble models can lead to difficult interpretative problems [11, 12]. The ensemble model is designed based on specific and limited samples in relation to the nature and volume of the dataset; hence, the direct use of the ensemble model for strength prediction should be avoided to be used for other material types, until establishing after further complementary studies [13]. The main disadvantage of an ensemble is the resources it requires: calculations, software availability, and analyst's skills and time investment.

In general, it is better to use a simpler model rather than a more complex model, and selecting tuning parameters based on numerically optimal value may result in an overly complex model. Other options for choosing less complex models should be investigated, as they might lead to simpler models that provide acceptable performance. The use of ML techniques to predict material strength is becoming popular. However, not much attention has been paid to instance-based learning (IBL) algorithms [14, 15]. Therefore, the objective of the present study is to investigate the potential of an IBL algorithm for predicting material strength, with data obtained from the literature. The k-nearest neighbors (k-NNs) are among the simplest of all ML algorithms. The standard k-NN was utilized to develop compressive strength prediction models for some concretes and rocks by considering indirect parameters such as physical and mechanical parameters. To verify and validate the standard k-NN models, prediction results of models reported in the literature were compared using six datasets via the cross-validation method to minimize the bias.

The remainder of this paper is organized as follows. Section 2 describes the standard k-NN approach and resampling methods. Section 3 presents data description and summarization. Section 4 describes the results and discussion for prediction of compressive strength before our conclusions are provided in Section 5.

2. Methods

The model development procedure is presented in this section. In the first part, standard k-NN models with performance statistics are discussed. After that, in the second part, the validation procedures of the standard k-NN models are presented. The models used for this study were implemented by the high-level programming language R, an open-source statistical software [16].

2.1. k-Nearest Neighbors. The standard k-NN approach, which is an IBL algorithm, simply predicts a new sample using the k-closest samples from the training set [17]. The construction of the model is solely based on the individual samples from the training data. To predict a new response value y_i (i.e., *CS* for compressive strength, and *E* for Young's modulus) for regression, k-NN identifies only k-closest neighbors $\mathbf{x}_i$ $(\mathbf{x}_1', \mathbf{x}_2', \ldots, \mathbf{x}_k')$ in the space of the data attributes (i.e., mixture factors and process factors) and using a predefined function (i.e., average function) of the response values of the k-nearest neighbors [18].

The Euclidean distance is the most commonly used as a measure of closeness between observations $\mathbf{x}_i$ and $\mathbf{x}_j$ and is defined as follows:

$$d\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sqrt{\sum_{m=1}^{D} \left(x_{im} - x_{jm}\right)^2}, \quad j \neq i, j = 1, 2, \ldots N, \quad (1)$$

where D represents the number of attributes, x_{im} and x_{jm} are components of vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, and N is the number of observations. Minkowski distance is a generalization of Euclidean distance and is defined as

$$d\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sqrt[q]{\sum_{m=1}^{D} \left(x_{im} - x_{jm}\right)^{1/q}}, \quad (2)$$

where $q > 0$ [19]. It is easy to see that when $q = 2$, then Minkowski distance is the same as Euclidean distance. When $q = 1$, the Minkowski distance is equal to the Manhattan distance, which is a general metric used for samples with binary predictors. There are many other distance measures, such as Tanimoto, Hamming, and Cosine and are more suitable for specific types of predictors. For example, when using binary fingerprints to describe molecules, Tanimoto distance is often used in computational chemistry problems [20]. To show that the elementary version of k-NN is intuitive and straightforward and can produce decent predictions, the Euclidean distance was used in this paper. The average of the response values of the k-nearest neighbors is used for calculating the unknown response value y_i:

$$y_i = f\left(\mathbf{x}_i\right) = \frac{\sum_{m=1}^{k} f\left(\mathbf{x}_m'\right)}{k}. \quad (3)$$

Since distances between the observations are used as a measure of closeness, the data have to be preprocessed to have the same mean and variance for each predictor. All predictors are centered and scaled prior to performing k-NN. To center the predictors, all values minus the average predictor value. Because of centering, the mean of the predictors is zero. Similarly, to scale the data, each value of the predictor variable is divided by its standard deviation. Scaling the data will force the values to have a common standard deviation of one [17].

In order to evaluate the prediction performances of the model, the coefficient of determination (R^2) and the root mean squared error (RMSE) were used:

$$R^2 = 1 - \frac{\sum_{m=1}^{M} \left(y_m - y_m^*\right)^2}{\sum_{m=1}^{M} \left(y_m - \overline{y}\right)^2},$$
$$\text{RMSE} = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \left(y_m - y_m^*\right)^2}, \quad (4)$$

where y_m and y_m^* are the observed and predicted values, respectively; $\overline{y}$ is the mean of the observed values; M is the number of data samples. R^2 provides information about the strength of correlation between observed and predicted values. RMSE evaluates the residual between the observed and predicted values. When predicting numeric values, RMSE is often used to evaluate the model. Quantitative evaluation of statistical information (i.e. RMSE) using resampling can help users understand how each technique performs on new data. The RMSE corresponding to the values of k within a range was used to select the optimal number of neighbors using the smallest RMSE. Furthermore, the observed and predicted values were plotted to discover areas of the data where the k-NN model did particularly good or bad.

2.2. Resampling Methods. When there is a large amount of data, the data can be split into training and test sets. The former is used to create a model, and the latter is used to evaluate the model performance. However, some researchers [21, 22] showed that validation using a single test set can be a poor choice. When the number of samples is small, a test set should be avoided, because each sample may be needed to build the model. In addition, the size of the test set may not have enough power or precision to make a reasonable judgment. Resampling methods can produce reasonable predictions about the model's performance in future samples. In this study, resampling methods, such as 10-fold cross-validation (CV), leave-one-out cross-validation (LOOCV), and repeated 10-fold cross-validation

(RepeatedCV), were used to minimize bias (overfitting and underfitting) associated with random sampling of training and hold out data samples and to determine the optimal number of neighbors to retain that minimize RMSE.

By evaluating how the model fits points that were not used to perform regression, we can understand how the model will function in future observations. It can evaluate the overall operation of the model, not just the observed data [23]. In CV, the sample was randomly divided into 10 groups of roughly equal size. A model was fit with all samples except for the first subset (called the first fold). The held-out samples were predicted by the model and used to estimate RMSE. The first subset was returned to the training set, the process was repeated with the second subset held out, and so on. The average RMSE of the 10 performance estimates would be the cross-validation estimate of model performance. Kohavi [24] confirmed that ten-fold validation testing yields the optimal computational time and reliable variance. In LOOCV, the number of fold was the number of samples (M). Because only one sample was held-out at a time, the final RMSE was calculated from the M individual held-out predictions. Some research studies [22, 25] indicated that RepeatedCV can effectively improve the accuracy of the estimation while still maintaining a small bias.

3. Data Description and Summarization

The datasets used in this study have been confirmed in some studies of predictive models (Table 1). Based on the standard k-NN model, this study used six experimental datasets to investigate the prediction performance of the elementary model. Table 2 lists the six datasets with descriptive statistics, i.e., maximum (Max), minimum (Min), average (Ave), and standard deviation (Std). The response/target was *CS*, and the predictor variables were the remaining attributes. The minimum and maximum values given in Table 2 of the different sample properties function as the boundary conditions of the some models, e.g., artificial neural networks (ANN).

Dataset 1. The test data of 144 different concrete mix-designs were gathered from Lam et al. [31]. The high-performance concrete (HPC) mixes were prepared at different ratios of water to cementitious materials, with low and high volumes of fly ash, and with or without addition of small amounts of silica fume. *CS* was 7.8–107.8 MPa. These samples consisted of 24 different mixes. In each mix series, the percentage of cement replacement by fly ash varied from 0% to 55%. The cementitious materials were Portland cement equivalent to ASTM type I, low-calcium fly ash equivalent to ASTM Class F, and a condensed silica fume commercially available in Hong Kong.

Dataset 2. Siddique et al. [32] collected the data of 80 concrete mixes with comparable physical and chemical composition properties from various studies. The self-compacting concrete mixes were made with water/powder ratios of 0.33–0.87 that contain from 0 to 261 kg/m^3 of fly ash. Coarse aggregate content varied from 621 to 923 kg/m^3. Fine aggregate content varied from 478 to 1079 kg/m^3. *CS* was 10.2–73.5 MPa. The content of superplasticizer was 0–100%.

Table 1: Sources of datasets in the literature.

Dataset	Data source	Laboratory	Sample size
1	Lam et al. [31]	Hong Kong	144
2	Siddique et al. [32]	Canada, USA, UK	80
3	Lim et al. [26]	South Korea	104
4	Teymen and Mengüç [27]	Turkey	93
5	Jahed Armaghani et al. [13]	Malaysia	71
6	Heidari et al. [33]	Iran	106

Dataset 3. A database of 104 concrete mixes used in this experiment that was carried out by Lim et al. [26] was produced in South Korea. The water to binder ratio of the HPC varies between 0.30 and 0.45, and the amount of fly ash used varied from 0% to 20% of the total binder, and the content of superplasticizer and air-entraining agent were 0.5–1.5% and 0.010–0.013%, respectively. Portland cement in accordance with ASTM type I was used. The coarse aggregate used was crushed granite (specific gravity, 2.7; fineness modulus, 7.2; maximum particle size, 19 mm). The fine aggregate was quartz sand (specific gravity, 2.61; fineness modulus, 2.94). *CS* was 38–74 MPa.

Dataset 4. The rocks were collected from the factories, outcrops, and quarries in different locations of Turkey [27]. A series of laboratory tests including physical test, ultrasonic velocity test, point load strength test, Schmidt hammer test, Brazilian tensile strength test, Shore hardness test, and uniaxial *CS* test were conducted on blocks or pieces taken from fresh parts of 93 different rocks from 32 rock types. *CS* was 6.64–303.67 MPa.

Dataset 5. The granite samples were taken from the face of the Pahang–Selangor raw water transfer tunnel in Malaysia [13]. A series of laboratory tests including physical test, ultrasonic velocity test, point load strength test, Schmidt hammer test and uniaxial *CS* test were conducted on 71 samples of granite. *CS* was 28.0–211.9 MPa. *E* was 22.0–183.3 GPa.

Dataset 6. A variety of sedimentary rocks including grainstone, wackestone-mudstone, boundstone, gypsum, and silty marl were collected from quarries in Qom Province, central Iran [33]. A series of laboratory tests including physical test, ultrasonic velocity test, point load strength test, and uniaxial *CS* test were conducted on 106 data sets. *CS* was 6.21–160.32 MPa.

4. Results and Discussion

This section depicts predictive accuracy of proposed models by comparing different models from literature studies. There are some strong between-predictor correlations, as shown in

Table 2: Descriptive statistics of datasets.

	Predictor	Min	Ave	Max	Std
Dataset 1					
Output	Concrete compressive strength, *CS* (MPa)	7.8	56.63	107.8	23.77
Input	Fly ash replacement ratio, *FA* (%)	0.0	25.00	55.0	19.11
	Silica fume replacement ratio, *SF* (%)	0.0	1.88	5.0	2.43
	Total cementitious material, *TCM* (kg/m^3)	400.0	436.67	500.0	45.13
	Fine aggregate, *ssa* (kg/m^3)	536.0	639.38	724.0	54.87
	Coarse aggregate, *ca* (kg/m^3)	1086.0	1125.00	1157.0	29.51
	Water content, *W* (lt/m^3)	150.0	171.67	205.0	24.00
	High rate water reducing agent, *HRWRA* (lt/m^3)	0.0	4.87	13.0	4.05
	Age of samples, *age* (days)	3.0	60.67	180.0	61.31
Dataset 2					
Output	Concrete compressive strength, *CS* (MPa)	10.2	38.52	73.5	14.13
Input	Cement content, *C* (kg/m^3)	160.0	271.75	427.0	66.25
	Fly ash content, *FA* (kg/m^3)	0.0	151.03	261.0	61.24
	Fine aggregate (sand) content, *ssa* (kg/m^3)	478.0	836.90	1079.0	107.34
	Coarse aggregate content, *ca* (kg/m^3)	621.0	856.05	923.0	44.06
	Water to powder ratio, *W/P* (%)	33.0	52.05	87.0	13.12
	Superplasticizer dosage, *SP* (%)	0.0	35.43	100.0	20.09
Dataset 3					
Output	Concrete compressive strength, *CS* (MPa)	38.0	52.68	74.0	9.43
Input	Water to binder ratio, *W/B* (%)	30.0	37.60	45.0	5.57
	Water content, *W* (kg/m^3)	160.0	170.00	180.0	8.24
	Fine aggregate ratio, *s/a* (%)	37.0	46.00	53.0	3.64
	Fly ash replacement ratio, *FA* (%)	0.0	10.10	20.0	8.30
	Air-entraining agent content, *AE* (kg/m^3)	0.036	0.054	0.078	0.015
	Superplasticizer content, *SP* (kg/m^3)	1.89	4.48	8.5	2.30
Dataset 4					
Output	Concrete compressive strength, *CS* (MPa)	6.64	104.77	303.67	64.93
Input	Point load strength index, I_s (MPa)	1.15	5.44	15.73	3.00
	P-wave velocity, V_p (km/s)	0.85	4.70	6.66	1.40
	Brazilian tensile strength, BTS (MPa)	1.02	8.45	21.32	4.34
	Schmidt hardness, SHH	16.85	47.31	65.12	11.31
	Shore hardness, SSH	7.2	59.06	99.0	21.98
	Unit weight, UW (g/cm^3)	1.05	2.49	2.96	0.36
Dataset 5: sample size: 71					
Output	Granite compressive strength, *CS* (MPa)	28.0	115.86	211.9	42.23
	Young's modulus, *E* (GPa)	22.0	87.99	183.3	35.17
Input	Porosity, *n* (%)	0.1	0.37	0.57	0.13
	Schmidt hammer rebound number, R_n	37.0	49.58	61.0	6.01
	P-wave velocity, V_p (m/s)	2823.0	5586.31	7943.0	1097.18
	Point load strength index, I_s (MPa)	0.89	3.32	7.1	1.51
Dataset 6					
Output	Rock compressive strength, *CS* (MPa)	6.21	68.69	160.32	38.24
Input	Schmidt rebound hardness number, *SHN*	17.0	31.53	47.0	6.23
	Block punch index, *BPI* (MPa)	1.89	7.84	19.02	3.75
	Point load strength index, I_s (MPa)	1.13	2.87	4.8	0.76
	P-wave velocity, V_p (m/s)	2592.0	4345.39	6231.0	1137.96

Tables 3 and 4. However, the percent of variance accounted for by each principal component is more than 37% of the variance, as shown in Table 5, indicating that there are no redundant predictors. In order to evaluate and compare the performance of different models, the prediction performances of the standard *k*-NN models and the models developed in the literature studies are presented in Table 6. A plot of the observed values against the predicted values helps one to understand how well the model fits. Also, a plot of the residuals versus the predicted values can help uncover systematic patterns in the model predictions, such as the trend. These plots for *k*-NN models are shown in Figures 1 and 2.

Dataset 1. Pala et al. [35] studied the impact of fly ash and silica fume replacement content on the long-term strength of concrete by ANN. Chou and Pham [15] used six data mining techniques, ANN, classification and regression trees (CART), chi-squared automatic interaction detector (CHAID), multiple linear regressions (MLR), generalized

Table 3: Correlation matrices of concrete datasets.

Dataset 1	*ssa*	*SF*	*W*	*ca*	*age*	*TCM*	*HRWRA*	*FA*	*CS*
ssa	1	0.040	−0.854	−0.279	0	0.479	0.682	−0.511	0.584
SF	—	1	0	0	0	0	0.030	−0.203	0.082
W	—	—	1	0.334	0	−0.568	−0.917	0	−0.485
ca	—	—	—	1	0	−0.965	−0.581	0	−0.484
age	—	—	—	—	1	0	0	0	0.578
TCM	—	—	—	—	—	1	0.761	0	0.556
HRWRA	—	—	—	—	—	—	1	0.166	0.477
FA	—	—	—	—	—	—	—	1	−0.342
CS	—	—	—	—	—	—	—	—	1
Dataset 2	*W*/*P*	*SP*	*ca*	*FA*	*C*	*ssa*	*CS*		
W/*P*	1	0.138	−0.306	−0.518	0.461	−0.350	−0.466		
SP	—	1	0.059	−0.236	−0.124	−0.028	−0.442		
ca	—	—	1	0.052	−0.379	0.141	−0.027		
FA	—	—	—	1	−0.612	−0.215	0.214		
C	—	—	—	—	1	−0.163	0.288		
ssa	—	—	—	—	—	1	0.375		
CS	—	—	—	—	—	—	1		
Dataset 3	*W*/*B*	*s*/*a*	*W*	*FA*	*AE*	*SP*	*CS*		
W/*B*	1	0.572	−0.032	−0.016	−0.958	−0.898	−0.909		
s/*a*	—	1	−0.382	−0.122	−0.622	−0.482	−0.333		
W	—	—	1	−0.014	0.208	−0.209	−0.286		
FA	—	—	—	1	0.012	0.024	−0.068		
AE	—	—	—	—	1	0.863	0.841		
SP	—	—	—	—	—	1	0.922		
CS	—	—	—	—	—	—	1		

The parameters listed in the first row of datasets 1 to 3 are defined in Table 2.

Table 4: Correlation matrices of rock datasets.

Dataset 4	UW	V_P	*SHH*	*SSH*	*BTS*	I_s	*CS*
UW	1	0.866	0.801	0.647	0.629	0.545	0.589
V_P	—	1	0.799	0.609	0.702	0.614	0.702
SHH	—	—	1	0.818	0.785	0.728	0.781
SSH	—	—	—	1	0.784	0.812	0.807
BTS	—	—	—	—	1	0.884	0.947
I_s	—	—	—	—	—	1	0.896
CS	—	—	—	—	—	—	1
Dataset 5	I_s	V_P	R_n	*n*	*E*	*CS*	
I_s	1	0.622	0.629	−0.728	0.617	0.814	
V_P	—	1	0.615	−0.663	0.673	0.789	
R_n	—	—	1	−0.631	0.696	0.701	
n	—	—	—	1	−0.562	−0.885	
E	—	—	—	—	1	0.739	
CS	—	—	—	—	—	1	
Dataset 6	V_P	*SHN*	I_s	*BPI*	*CS*		
V_P	1	0.697	0.629	0.693	0.819		
SHN	—	1	0.812	0.813	0.872		
I_s	—	—	1	0.822	0.873		
BPI	—	—	—	1	0.873		
CS	—	—	—	—	1		

The parameters listed in the first row of datasets 4 to 6 are defined in Table 2.

Table 5: Proportion of variance of the first three principal components for each dataset.

Dataset	PC1	PC2	PC3
1	0.4489	0.1810	0.1314
2	0.3767	0.2165	0.1834
3	0.5417	0.2103	0.1685
4	0.7799	0.1171	0.0470
5	0.7363	0.1005	0.0964
6	0.8096	0.1009	0.0470

linear model (GENLIN), and support vector machines (SVM) to construct individual and ensemble models. The number of parameter settings for the aforementioned six single models varies from 5 to 10. Table 6 shows that top three performing models are ANNs, CART, and CHAID. The standard *k*-NN shows modest result as MLR and GENLIN. The observed *CS* values against those predicted from the *4*-NN (RepeatedCV), as shown in Figure 1, shows modest concordance between the observed and predicted values, and there are about as many positive as negative residuals and they do not show any strong patterns. There are some observations fairly far from the horizontal axis, but many more close to it. The majority of the residuals are within ±10 MPa for *CS* ranging from 7.8 to 107.8 MPa.

Dataset 2. Table 6 shows that top two performing models are MLR and GENLIN. The standard *k*-NN shows modest result as ANN and CART. The observed *CS* values against those predicted from the *7*-NN (CV), as shown in Figure 1, shows modest correlation between the observed and predicted values, and there are about as many positive residuals as negative residuals, and they do not show any strong patterns. The majority of the residuals are within ±10 MPa for *CS* ranging from 10.2 to 73.5 MPa.

Table 6: Prediction performances of individual models.

Dataset	ML technique	RMSE	R^2	Literature
1	Concrete compressive strength			Chou & Pham [11]
	ANN	5.867	0.993	
	CART	7.860	0.983	
	CHAID	8.307	0.980	
	MLR	11.735	0.961	
	GENLIN	11.596	0.962	
	SVM	13.145	0.950	
	***4*-NN (RepeatedCV)**	**10.693**	**0.808**	
	***4*-NN (CV)**	**10.782**	**0.805**	
	***5*-NN (LOOCV)**	**10.829**	**0.803**	
2	Concrete compressive strength			Chou & Pham [11]
	ANN	7.104	0.832	
	CART	7.312	0.848	
	CHAID	10.040	0.699	
	MLR	5.161	0.917	
	GENLIN	5.163	0.917	
	SVM	10.954	0.726	
	***7*-NN (CV)**	**7.437**	**0.756**	
	***8*-NN (RepeatedCV)**	**7.525**	**0.762**	
	***8*-NN (LOOCV)**	**7.734**	**0.743**	
3	Concrete compressive strength			Chou & Pham [11]
	ANN	1.548	0.987	
	CART	1.837	0.982	
	CHAID	1.994	0.979	
	MLR	1.996	0.979	
	GENLIN	1.996	0.980	
	SVM	2.012	0.981	
	Stepwise regression	2.020	0.956	Ahmadi-Nedushan [38]
	***2*-NN (LOOCV)**	**1.747**	**0.966**	
	***2*-NN (CV)**	**1.834**	**0.967**	
	***2*-NN (RepeatedCV)**	**1.856**	**0.965**	
4	Rock compressive strength			Teymen and Mengüç [27]
	ANN	N/A	0.921	
	MRA	N/A	0.953	
	***8*-NN (CV)**	21.954	0.896	
	***9*-NN (RepeatedCV)**	21.956	0.899	
	***9*-NN (LOOCV)**	22.476	0.887	
5	Granite compressive strength			Jahed Armaghani et al. [13]
	ANN	26.203	0.804	
	MRA	13.818	0.891	
	3-**NN (CV)**	**14.509**	**0.885**	
	3-**NN (RepeatedCV)**	**14.867**	**0.872**	
	***3*-NN (LOOCV)**	**15.576**	**0.862**	
	Granite Young's modulus			Jahed Armaghani et al. [13]
	ANN	21.022	0.643	
	MRA	22.192	0.596	
	***7*-NN (CV)**	**21.850**	**0.630**	
	***7*-NN (RepeatedCV)**	**21.876**	**0.638**	
	***7*-NN (LOOCV)**	**22.654**	**0.580**	
6	Rock compressive strength			
	ANN	7.58	0.96	Jalali et al. [30]
	MRA	10.80	0.91	Heidari et al. [33]
	***2*-NN (LOOCV)**	**10.387**	**0.926**	
	***2*-NN (RepeatedCV)**	**10.466**	**0.929**	
	***2*-NN (CV)**	**10.758**	**0.927**	

Highlighted ones in bold denote the *k*-NN model and performance measure.

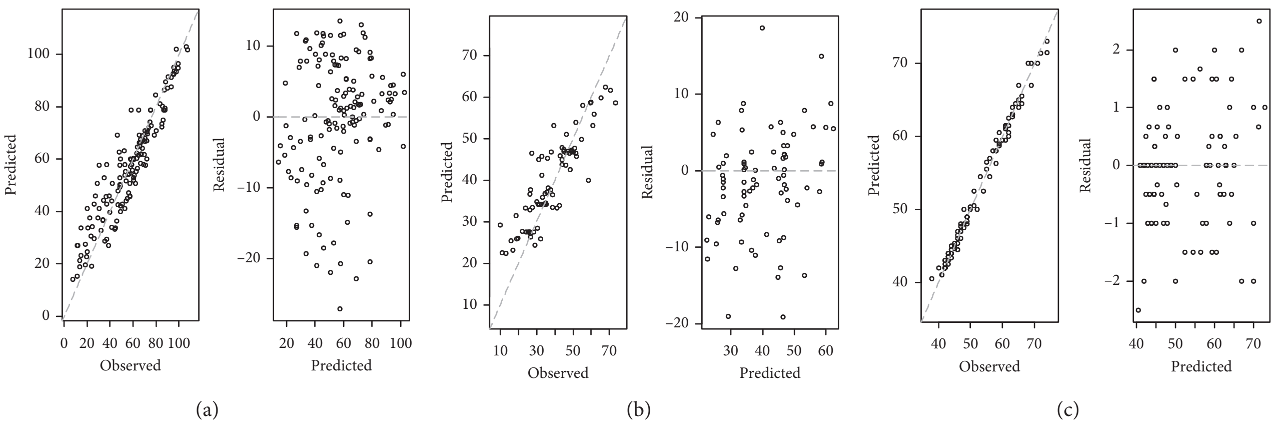

FIGURE 1: Left: predicted versus observed *CS* values for concrete datasets; right: residuals versus the predicted values. (a) Dataset 1. (b) Dataset 2. (c) Dataset 3.

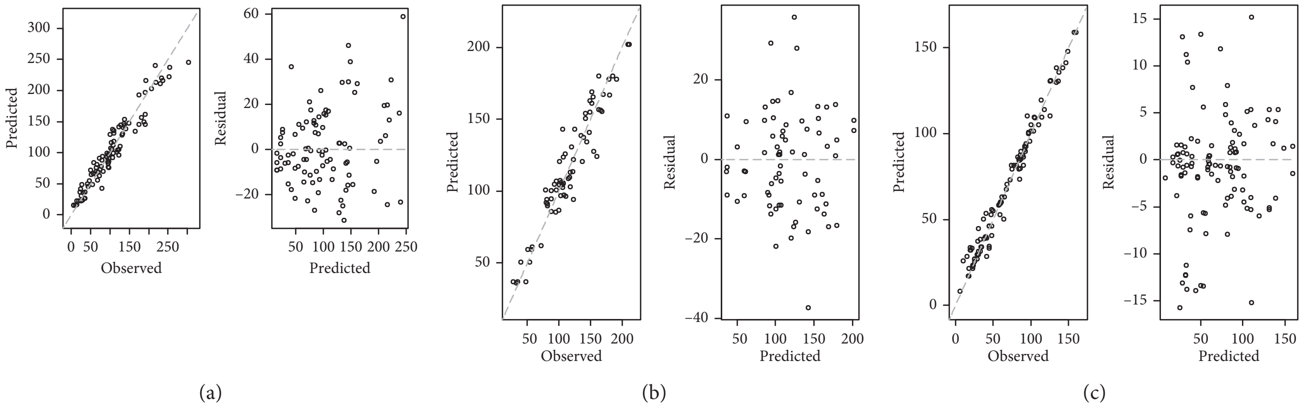

FIGURE 2: Left: predicted versus observed *CS* values for rock datasets; right: residuals versus the predicted values. (a) Dataset 4. (b) Dataset 5. (c) Dataset 6.

Dataset 3. Table 6 shows that top two performing models are ANN and *k*-NN. The observed *CS* values against those predicted from the *2*-NN (LOOCV), as shown in Figure 1, show good concordance between the observed and predicted values, and there are about as many positive residuals as negative residuals, and they do not show any strong patterns. The majority of the residuals are within ±2 MPa for *CS* ranging from 38 to 74 MPa.

In the study of Chou and Pham [11], the IBM SPSS modeler [36] was used in ANN analyses with the standard feedforward backpropagation, the gradient descent algorithm, and three hidden layers (20, 15, and 10 neurons). The best individual model in predicting HPC compressive strength using three experimental datasets was ANN, which achieved 45.1%, 4.5%, and 11.4% better error rates than those of the standard *k*-NN model for datasets 1 to 3, respectively. The standard *k*-NN model achieved 18.7%, 32.1%, and 13.2% better error rates than those of the lowest performing model, SVM, for datasets 1 to 3, respectively. One significant limitation of the work done by Chou and Pham is that it used the IBM SPSS modeler with default settings in the single and ensemble models. Therefore, further studies are needed to determine optimum values of parameters.

Siddique et al. [32] adopted the procedure for partitioning the neural-network connection weights proposed by Garson [37] to determine the relative importance of the various inputs. The enhanced ANN model yielded an RMSE of 5.557 MPa and achieved 21.8% better error rates than those of the standard ANN model for dataset 2.

Ahmadi-Nedushan [38] used differential evolution algorithm to find the optimal *k*-NN model parameters, such as number of neighbors, distance function, and attribute weights. The best enhanced model, with optimal attribute weighting, yielded an RMSE of 1.174 and achieved 32.8% better error rates than those of the standard *k*-NN model for dataset 3 because the Euclidian distance function in the standard *k*-NN model assumed that all the attributes were equally important. Sometimes, the right choice of neighbors depends on modifying the distance function to favor some predictors over others. This is easily accomplished by incorporating weights into the distance function.

Dataset 4. In the study of Teymen and Mengüç [27], MRA and ANN models were made with the help of IBM SPSS modeler and Matlab [39], respectively. Gradient descent with momentum and adaptive learning rate backpropagation algorithm and one hidden layer with three neurons were used. All binary combinations of the six independent variables were tried as input parameters. According to the performance index assessment, the weakest model was the one with V_p and *SSH* as input and yielded an R^2 of of 0.874 and 0.834 for ANN and MRA, respectively. The most successful model was the one with *BTS* and I_s as input and yielded an R^2 of 0.921 and 0.953 for ANN and MRA, respectively, as shown in Table 6. Nevertheless, *k*-NN with the six independent variables as input yielded a larger R^2 of 0.931, 0.917, and 0.905 for *7*-NN (CV), *5*-NN (ReapeatedCV), and *5*-NN (LOOCV), respectively. *k*-NN is a highly automated data-driven method. Therefore, the algorithm of *k*-NN is intuitive, straightforward, and easy to implement and can produce decent predictions. The observed *CS* values against those predicted from the *7*-NN (CV), as shown in Figure 2, show good concordance between the observed and predicted values, and there are about as many positive as negative residuals and they do not show any strong patterns. There are some observations fairly far from the horizontal axis, but many more close to it. The majority of the residuals are within ±20 MPa for *CS* ranging from 6.64 to 303.67 MPa.

Dataset 5. Table 6 shows that there is no clear winner or loser for predicting *E* in the three models, *k*-NN, ANN, and multivariate regression analysis (MRA). *k*-NN and MRA can predict *CS* with a high degree of accuracy. The observed *CS* values against those predicted from the *3*-NN (CV), as shown in Figure 2, show good concordance between the observed and predicted values, and there are about as many positive residuals as negative residuals, and they do not show any strong patterns. The majority of the residuals are within ±20 MPa for *CS* ranging from 28 to 211.9 MPa.

To overcome shortcomings such as the slow rate of learning and entrapment in local minima, Jahed Armaghani et al. [13] built an ANN enhanced with the imperialist competitive algorithm (ICA) [28, 29] to predict *CS* and *E*. The performance of the ICA-ANN can predict *CS* with a high degree of accuracy and *E* with a suitable degree of accuracy. The enhanced ANN model yielded an RMSE of 12.454 MPa and achieved 52.5% better error rates than those of conventional ANN for dataset 4. However, Jahed Armaghani et al. [13] mentioned that the proposed ICA-ANN predictive model is designed based on the *CS* of granite samples; hence, the direct use of the ICA-ANN model for *CS* prediction of other rock types is not suggested.

Dataset 6. In the study of Jalali et al. [30], Matlab software was used in ANN analyses with the standard feedforward network, the Levenberg–Marquardt algorithm, and one hidden layer with 5 neurons. Table 6 shows that ANN is top performing model. The predictive performances of MRA and *k*-NN show both models are comparable. The observed *CS* values against those predicted from the *2*-NN (LOOCV), as shown in Figure 2, show good concordance between the observed and predicted values, and there are about as many positive residuals as negative residuals, and they do not show any strong patterns. The majority of the residuals are within ±15 MPa for *CS* ranging from 6.21 to 160.32 MPa.

Generally, parametric methods will tend to outperform nonparametric approaches when there are a small number of observations per predictor. For example, MLR, GENLIN, and MRA outperform nonparametric approaches for datasets 2, 4, and 5. However, algorithms for predicting material strength based on conventional regression analysis and statistical models may be unsuitable because it is highly complex and correlations give good results only in similar materials [40–42].

Residual plots show that the resulting predictions of *k*-NN are always reasonable within a reasonable range. This is because the final prediction is based on the actual value of the neighbor. Keep in mind that regression and neural networks may produce impossible results because the prediction range is from negative infinity to positive infinity, and the range of reasonable values may not be so extreme. The *k*-NN technique produces reasonable values with many distinct values. However, the range of predicted values is narrower than the range in the dataset. This is due to the averaging combination function, which smooths out the maximum and minimum values.

No resampling method is uniformly better than another. However, putting computational issues aside, a less obvious but potentially more important advantage of 10-fold CV is that it often gives more accurate rate than does LOOCV. This has to do with a bias-variance trade-off. Since the mean value of many highly correlated predictors has a higher variance than the mean value of many non-highly correlated predictors, the test error estimate produced by LOOCV tends to have a higher variance than the test error estimate produced by 10-fold CV.

5. Conclusions

This study has examined the use of an elementary ML technique to predict compressive strengths of some concretes and rocks. As seen in Table 6, the values of RMSE for *k*-NN are modest, indicating adequacy to predict compressive strength with comprehensive range values of predictors. Additionally, the R^2-values of the *k*-NN models were high. In other words, the models were able to explain the variance in compressive strength for data with a wide range of input values. One benefit of this approach is its simplicity, which allows us to use rigorous analysis to guide our intuition and research goals. Furthermore, *k*-NN does not require that the data satisfy some predefined model. *k*-NN requires neither temporary parameters nor background knowledge. Aha [34] showed that when combined with noisy example pruning and attribute weighting, IBL performs well compared with other methods. In short, *k*-NN with cross-validation is a simple, general, effective technique which yields high quality predictions by combining the predicted values of the *k*-nearest neighbors and weighting them by distance. However, finding a computationally effective means for calculating these weights requires further research.

Removing irrelevant predictors is a key preprocessing step for *k*-NN. Expert knowledge should first be applied to obtain relevant data for the required research objectives.

Furthermore, in an attempt to remove noninformative or redundant predictors from the model and to find only the most relevant predictors in a given problem, many different types of feature selection methods have been proposed [43].

Table 6 indicates that the results obtained by ANN are better than *k*-NN techniques. However, the tuning of parameters such as momentum, learning rate, and number of hidden layers, makes ANN easy to overfit the data at hand. Furthermore, the application of some subject matter expertise to the data preparation improves model performance. In dataset 4, the most successful model was the one with BTS and I_s of the six independent variables as input and yielded an R^2 of 0.921 for ANN. Nevertheless, *k*-NN with the six independent variables as input yielded a larger R^2 of 0.931. Therefore, the algorithm of *k*-NN is more intuitive and straightforward.

Sometimes one of simple models will be the best predicting model available; but in many cases, these models will serve as benchmarks rather than the model of choice. That is, any predicting model might be compared to these simple models to ensure that the new model is better than these simple alternatives. If not, the new model is not worth considering.

In model selection, whenever possible, analysts should not rely on a single data mining method. There is no single model that will always do better than any other model. Choosing between multiple models largely depends on the characteristics of the data and the type of questions being answered. Therefore, it is customary to apply several different methods in data mining and then choose the most useful method for the current goal. The *k*-NN model that reasonably approximates the performance of the more complex methods could be used as a tool to support decision making because the standard *k*-NN is easy to implement and has potential applications in material science.

Data Availability

The data used in the study were collected from different research papers in modelling aspect.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] S. Aboutaleb, M. Behnia, R. Bagherpour, and B. Bluekian, "Using non-destructive tests for estimating uniaxial compressive strength and static Young's modulus of carbonate rocks via some modeling techniques," *Bulletin of Engineering Geology and the Environment*, vol. 77, no. 4, pp. 1717–1728, 2018.

[2] P. Chopra, R. K. Sharma, and M. Kumar, "Prediction of compressive strength of concrete using artificial neural network and genetic programming," *Advances in Materials Science and Engineering*, vol. 2016, Article ID 7648467, 10 pages, 2016.

[3] M. T. Cihan, "Prediction of concrete compressive strength and slump by machine learning methods," *Advances in Civil Engineering*, vol. 2019, Article ID 3069046, 11 pages, 2019.

[4] N. Madhubabu, P. K. Singh, A. Kainthola, B. Mahanta, A. Tripathy, and T. N. Singh, "Prediction of compressive strength and elastic modulus of carbonate rocks," *Measurement*, vol. 88, pp. 202–213, 2016.

[5] B. A. Omran, Q. Chen, and R. Jin, "Comparison of data mining techniques for predicting compressive strength of environmentally friendly concrete," *Journal of Computing in Civil Engineering*, vol. 30, no. 6, p. 13, Article ID 04016029, 2016.

[6] L. K. Sharma, V. Vishal, and T. N. Singh, "Developing novel models using neural networks and fuzzy systems for the prediction of strength of rocks from key geomechanical properties," *Measurement*, vol. 102, pp. 158–169, 2017.

[7] B. A. Young, A. Hall, L. Pilon, P. Gupta, and G. Sant, "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*, vol. 115, pp. 379–388, 2019.

[8] M. C. Kang, D. Y. Yoo, and R. Gupta, "Machine learning-based prediction for compressive and flexural strengths of steel fiber-reinforced concrete," *Construction and Building Materials*, vol. 266, p. 13, Article ID 121117, 2021.

[9] S. R. Salimbahrami and R. Shakeri, "Experimental investigation and comparative machine-learning prediction of compressive strength of recycled aggregate concrete," *Soft Computing*, vol. 25, no. 2, pp. 919–932, 2021.

[10] M. A. Khan, S. A. Memon, F. Farooq, M. F. Javed, F. Aslam, and R. Alyousef, "Compressive strength of fly-ash-based geopolymer concrete by gene expression programming and random forest," *Advances in Civil Engineering*, vol. 2021, Article ID 6618407, 17 pages, 2021.

[11] J.-S. Chou and A.-D. Pham, "Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength," *Construction and Building Materials*, vol. 49, pp. 554–563, 2013.

[12] R. Cook, J. Lapeyre, H. Ma, and A. Kumar, "Prediction of compressive strength of concrete: critical comparison of performance of a hybrid machine learning model with standalone models," *Journal of Materials in Civil Engineering*, vol. 31, no. 11, p. 15, Article ID 04019255, 2019.

[13] D. Jahed Armaghani, E. Tonnizam Mohamad, E. Momeni, M. Monjezi, and M. S. Narayanasamy, "Prediction of the strength and elasticity modulus of granite through an expert artificial neural network," *Arabian Journal of Geosciences*, vol. 9, p. 16, Article ID 48, 2016.

[14] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37–66, 1991.

[15] D. Kibler, D. W. Aha, and M. K. Albert, "Instance-based prediction of real-valued attributes," *Computational Intelligence*, vol. 5, no. 2, pp. 51–57, 1989.

[16] R Development Core Team R, *A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2014.

[17] D. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons, Hoboken, NJ, USA, 2005.

[18] G. Myatt, *Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining*, John Wiley & Sons, Hoboken, NJ, USA, 2007.

[19] B. Liu, *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*, Springer, Berlin, Germany, 2011.

[20] P. McCarren, C. Springer, and L. Whitehead, "An investigation into pharmaceutically relevant mutagenicity data and

the influence on Ames predictive potential," *Journal of Cheminformatics*, vol. 3, p. 20, 2011.

[21] D. M. Hawkins, S. C. Basak, and D. Mills, "Assessing model fit by cross-validation," *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 2, pp. 579–586, 2003.

[22] A. M. Molinaro, R. Simon, and R. M. Pfeiffer, "Prediction error estimation: a comparison of resampling methods," *Bioinformatics*, vol. 21, no. 15, pp. 3301–3307, 2005.

[23] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010.

[24] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, vol. 2, pp. 1137–1143, Montreal, Canada, August 1995.

[25] J.-H. Kim, "Estimating classification error rate: repeated cross-validation, repeated hold-out and bootstrap," *Computational Statistics & Data Analysis*, vol. 53, no. 11, pp. 3735–3745, 2009.

[26] C.-H. Lim, Y.-S. Yoon, and J.-H. Kim, "Genetic algorithm in mix proportioning of high-performance concrete," *Cement and Concrete Research*, vol. 34, no. 3, pp. 409–420, 2004.

[27] A. Teymen and E. C. Mengüç, "Comparative evaluation of different statistical tools for the prediction of uniaxial compressive strength of rocks," *International Journal of Mining Science and Technology*, vol. 30, 2020.

[28] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition," in *Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC)*, pp. 4661–4667, Singapore, January 2007.

[29] M. A. Ahmadi, M. Ebadi, A. Shokrollahi, and S. M. J. Majidi, "Evolving artificial neural network and imperialist competitive algorithm for prediction oil flow rate of the reservoir," *Applied Soft Computing*, vol. 13, no. 2, pp. 1085–1098, 2013.

[30] S. H. Jalali, M. Heidari, and H. Mohseni, "Comparison of models for estimating uniaxial compressive strength of some sedimentary rocks from Qom Formation," *Environmental Earth Sciences*, vol. 76, p. 15, Article ID 753, 2017.

[31] L. Lam, Y. L. Wong, and C. S. Poon, "Effect of fly ash and silica fume on compressive and fracture behaviors of concrete," *Cement and Concrete Research*, vol. 28, no. 2, pp. 271–283, 1998.

[32] R. Siddique, P. Aggarwal, and Y. Aggarwal, "Prediction of compressive strength of self-compacting concrete containing bottom ash using artificial neural networks," *Advances in Engineering Software*, vol. 42, no. 10, pp. 780–786, 2011.

[33] M. Heidari, H. Mohseni, and S. H. Jalali, "Prediction of uniaxial compressive strength of some sedimentary rocks by fuzzy and regression models," *Geotechnical and Geological Engineering*, vol. 36, no. 1, pp. 401–412, 2018.

[34] D. W. Aha, "Tolerating noisy, irrelevant and novel attributes in instance-based learning algorithms," *International Journal of Man-Machine Studies*, vol. 36, no. 2, pp. 267–287, 1992.

[35] M. Pala, E. Özbay, A. Öztaş, and M. I. Yuce, "Appraisal of long-term effects of fly ash and silica fume on compressive strength of concrete by neural networks," *Construction and Building Materials*, vol. 21, no. 2, pp. 384–394, 2007.

[36] IBM, *PASW Modeler*, IBM Corporation, New York, NY, USA, 2010.

[37] G. D. Garson, "Interpreting neural-network connection weights," *AI Expert*, vol. 6, no. 7, pp. 47–51, 1991.

[38] B. Ahmadi-Nedushan, "An optimized instance based learning algorithm for estimation of compressive strength of concrete," *Engineering Applications of Artificial Intelligence*, vol. 25, pp. 1073–1081, 2012.

[39] MathWorks, *Matlab R2015b: Software for Technical Computing and Model Based Design*, The MathWorks Inc., Natick, MA, USA, 2015.

[40] M. Rezaei, A. Majdi, and M. Monjezi, "An intelligent approach to predict unconfined compressive strength of rock surrounding access tunnels in longwall coal mining," *Neural Computing and Applications*, vol. 24, no. 1, pp. 233–241, 2014.

[41] I.-C. Yeh and L.-C. Lien, "Knowledge discovery of concrete material using genetic operation trees," *Expert Systems With Applications*, vol. 36, no. 3, pp. 5807–5812, 2009.

[42] I. Yilmaz and G. Yuksek, "Prediction of the strength and elasticity modulus of gypsum using multiple regression, ANN, and ANFIS models," *International Journal of Rock Mechanics and Mining Sciences*, vol. 46, no. 4, pp. 803–810, 2009.

[43] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, Springer, Berlin, Germany, 2016.