applied sciences

MDPI

Article

Concrete Strength Prediction Using Machine Learning Methods CatBoost, k-Nearest Neighbors, Support Vector Regression

Alexey N. Beskopylny [1,*], Sergey A. Stel'makh [2], Evgenii M. Shcherban' [3], Levon R. Mailyan [4], Besarion Meskhi [5], Irina Razveeva [6], Andrei Chernil'nik [2] and Nikita Beskopylny [7]

[1] Department of Transport Systems, Faculty of Roads and Transport Systems, Don State Technical University, 344003 Rostov-on-Don, Russia
[2] Department of Unique Buildings and Constructions Engineering, Don State Technical University, Gagarin Sq. 1, 344003 Rostov-on-Don, Russia
[3] Department of Engineering Geology, Bases, and Foundations, Don State Technical University, 344003 Rostov-on-Don, Russia
[4] Department of Roads, Don State Technical University, 344003 Rostov-on-Don, Russia
[5] Department of Life Safety and Environmental Protection, Faculty of Life Safety and Environmental Engineering, Don State Technical University, 344003 Rostov-on-Don, Russia
[6] Department of Mathematics and Informatics, Faculty of IT-Systems and Technology, Don State Technical University, Gagarin Sq. 1, 344003 Rostov-on-Don, Russia
[7] Department Hardware and Software Engineering, Faculty of IT-Systems and Technology, Don State Technical University, 344003 Rostov-on-Don, Russia
* Correspondence: besk-an@yandex.ru; Tel.: +7-8632738454

Abstract: Currently, one of the topical areas of application of machine learning methods in the construction industry is the prediction of the mechanical properties of various building materials. In the future, algorithms with elements of artificial intelligence form the basis of systems for predicting the operational properties of products, structures, buildings and facilities, depending on the characteristics of the initial components and process parameters. Concrete production can be improved using artificial intelligence methods, in particular, the development, training and application of special algorithms to determine the characteristics of the resulting concrete. The aim of the study was to develop and compare three machine learning algorithms based on CatBoost gradient boosting, k-nearest neighbors and support vector regression to predict the compressive strength of concrete using our accumulated empirical database, and ultimately to improve the production processes in construction industry. It has been established that artificial intelligence methods can be applied to determine the compressive strength of self-compacting concrete. Of the three machine learning algorithms, the smallest errors and the highest coefficient of determination were observed in the KNN algorithm: MAE was 1.97; MSE, 6.85; RMSE, 2.62; MAPE, 6.15; and the coefficient of determination R^2, 0.99. The developed models showed an average absolute percentage error in the range 6.15−7.89% and can be successfully implemented in the production process and quality control of building materials, since they do not require serious computing resources.

Keywords: artificial intelligence; machine learning; regression; CatBoost; k-nearest neighbors; support vector regression; self-compacting concrete

check for updates

Citation: Beskopylny, A.N.; Stel'makh, S.A.; Shcherban', E.M.; Mailyan, L.R.; Meskhi, B.; Razveeva, I.; Chernil'nik, A.; Beskopylny, N. Concrete Strength Prediction Using Machine Learning Methods CatBoost, k-Nearest Neighbors, Support Vector Regression. *Appl. Sci.* **2022**, *12*, 10864. https://doi.org/10.3390/app122110864

Academic Editor: Doo-Yeol Yoo

Received: 9 October 2022
Accepted: 25 October 2022
Published: 26 October 2022

1. Introduction

The construction industry is currently one of the main engines of the economy. The requirements and levels of responsibility for buildings and structures are increasing; new cities and districts are growing, and densely populated regions continue to develop their urbanized territories. In this regard, the processes of production of building materials, products and structures should be singled out separately. The fact is that the production of building materials is at the junction between the manufacturing and construction industries. In particular, a concept such as concrete: concrete mixture refers simultaneously to the concept of the construction industry, that is, to the factory sector, and to the concept of construction technology, for example, in monolithic concreting. Owing to the fact that concrete is the main building material throughout the world, but at the same time is one of the most complex artificial composites created by man, the prediction of its properties is not always fully possible. There are a huge number of factors and criteria that affect the final quality of concrete, and ultimately, the safety of products, structures, buildings and structures created from it. Thus, one of the main tasks of process engineers and scientists in the field of materials science is the search for the most effective prescription and technological methods aimed at achieving the goals of controlling the structure and regulating the properties of concrete and products based on them. In this regard, it is obvious that the problem of modern production and construction requires a high degree of manual labor and the influence of a strong human factor. Often, the calculations of technologists, errors in the recipe and probable violations of technology lead to disasters in construction, accidents during the construction of buildings and structures, and premature collapse of load-bearing structures. In addition, enclosing structures made of various types of concrete also suffer significantly. Thus, the problem expressed in the influence of the human factor is relevant [1–7].

Currently, the construction industry is on the verge of digitalization, which is destroying traditional ideas about the construction process, and also opens up many opportunities. The construction industry lags behind other sectors in terms of the implementation of modern information technologies due to its size and heterogeneity, and it will take many more years for it to reach the level of automation that has already been achieved today, for example, mechanical engineering. However, the movement of the industry toward the introduction of modern information technologies is inevitable. Companies that do not think about using big data, data analysis and the use of artificial intelligence methods in their work after the crisis are at risk of leaving the market during the next crisis. Prospects for improving the quality of manufactured products, services provided, and the formation of a positive image of modern companies lie in the use of artificial intelligence methods for digitalization, systematization of accumulated and incoming information, and forecasting cost, time and technological parameters in construction. Artificial intelligence solutions, which are already successfully used in other industries, are gradually being introduced into the construction process at all stages, including quality control in the production of building materials [8–14].

Table 1 provides an overview of the application of different machine learning methods to predict various characteristics of concrete and concrete products and structures.

Table 1. Overview of the application of various machine learning methods for predicting the characteristics of concrete and products and structures from it.

Ref. Number	Object of Study	Predictable Characteristics	Prediction Method
[15]	Geopolymer concrete based on fly ash	Compressive strength, flexural tensile strength	Orthogonal experimental plan
[16]	Heavy concrete	Search for cracks on the surface of concrete	Convolutional neural network
[17]	Beams made of ultra-high-quality fiber-reinforced concrete	Shear strength	Artificial neural network, support vector regression, extreme gradient boosting
[18]	Geopolymer concrete	Water absorption, water permeability, density	Artificial neural network

Table 1. *Cont.*

Ref. Number	Object of Study	Predictable Characteristics	Prediction Method
[19]	Concrete with the addition of metakaolin as a partial replacement for cement	Compressive strength, tensile strength, flexural tensile strength	Gene expression programming, artificial neural network, M5P model tree algorithm, random forest
[20]	Heavy concrete	Compressive strength	M5P model tree algorithm
[21,22]	Heavy concrete with secondary aggregate	Elastic modulus, compressive strength	Model tree algorithm M5, artificial neural network
[23]	Concrete containing rice husk ash and reclaimed asphalt pavement as a partial replacement for Portland cement and primary aggregates, respectively	Compressive strength	Artificial neural network
[24]	Concrete with partial or complete replacement of natural aggregate with waste rubber	Compressive strength	Artificial neural network
[25]	Self-compacting concrete with recycled aggregate	Compressive strength	Artificial neural network algorithm: Levenberg–Marquardt, Bayesian regularization, scaled conjugate gradient back-propagation
[2,26]	Self-compacting concrete with fly ash	Compressive strength	Nonlinear dependency model, multiregression model, artificial neural network
[27]	Self-compacting concrete with recycled aggregates	Compressive strength	Ensemble methods: random forest, k-nearest neighbors, extremely randomized trees, extreme gradient boosting, gradient boosting, light gradient boosting machine
[28]	Double-wall tubular columns with metal and nonmetal composite materials	Axial compressive strength	Random forest regression, XGBoost regression, AdaBoost regression, lasso regression, ridge regression, ANN regression
[29]	Geopolymer concrete	Compressive strength, flexural tensile strength	Artificial neural network based on GDX (adaptive LR with gradient descent)
[30]	Fresh concrete mix	Plastic viscosity, yield strength	Artificial neural network, random forest
[31]	Round bounded concrete columns	Compressive strength	Multiphysics programming of genetic expressions
[32]	Reinforced concrete beams with collars	Shear strength	Artificial neural network
[33]	Self-compacting geopolymer concrete	Plastic viscosity, compressive strength	Hybrid artificial neural network combined with bat.
[34]	Ash concrete from rice husks	Compressive strength	Artificial neural network, artificial neurofuzzy inference system
[35]	Environmentally friendly concrete containing coal waste	Flexural tensile strength	Hybrid artificial neural network combined with response surface methodology

Table 1. *Cont.*

Ref. Number	Object of Study	Predictable Characteristics	Prediction Method
[36]	Heavy concrete	Compressive strength	Artificial neural network RBF
[37–39]	Recycled concrete	Compressive strength	Artificial neural network, gene expression programming
[40]	Concrete based on ceramic waste	Mobility, compressive strength, density	Artificial neural network, decision tree
[12]	Concrete modified with eggshell powder	Compressive strength	Artificial neural network combined with ANL-SFL metaheuristic optimization algorithm
[13]	Geopolymer concrete based on fly ash with high calcium content	Compressive strength	Artificial neural network, boosting and AdaBoost ML
[41]	Concrete reinforced with carbon nanotubes/carbon nanofibers	Compressive strength, flexural tensile strength	Artificial neural network
[42]	Concrete curing in hot weather	Pulse velocity, compressive strength, depth of water penetration, split tensile strength	Artificial neural network, finite regression model
[43]	Self-compacting rubberized concrete	Compressive strength	Multilayered perceptron artificial neural network (MLP-ANN), ensembles of MLP-ANNs, regression tree ensembles (random forests, boosted and bagged regression trees), support vector regression and Gaussian process regression
[44]	Concrete at high temperatures	Compressive strength	Decision tree, artificial neural network, bagging, gradient boosting

In the production of building materials, researchers generate a large amount of data containing important information about the mechanical properties of the resulting material. Data such as the volumetric content of various components, together with the description of the process and results of experiments, often have an unstructured and complex form (in the form of texts in natural language, tables, graphs) [45]. The introduction of artificial intelligence methods, in particular machine learning, for the analysis of accumulated data arrays will improve the quality of construction technology and optimize costs by reducing time costs [46–54]. In this regard, the purpose of our study is the development and comparison of three machine learning algorithms based on CatBoost gradient boosting, k-nearest neighbors and support vector regression methods for predicting the compressive strength of concrete using our accumulated empirical database, and ultimately, improvement of production processes in the construction industry. The objectives of the study were:

- Deep analysis of existing machine learning methods in concrete technology, analysis of the experience of their application, evaluation of such experience and the conclusion of scientific and practical deficits from the information received.
- Docking of experimental empirical results obtained in the course of real physical experiments and training on their basis of special tools that allow control of the properties and predict the performance of concretes and structures based on using machine learning methods.
- After processing and applying the data of a physical experiment, the development of an algorithm is based on three methods of machine learning: CatBoost gradient

boosting, the k-nearest neighbors method and the support vector regression method, for processing the empirical base with further comparison of the results based on the values of the main metrics.

- Assessment of the prospects for applying the developed methods in practice and the possibility of translating and projecting the results obtained on various types of concrete, and developing specific proposals for construction industry enterprises.

The proposals developed must be tested and substantiated by verifying them against real data. Thus, the scientific novelty of our study is new relationships between real physical experimental data, empirical relationships and values based on them, together with an assessment of the applicability of machine learning methods in predicting the properties of similar concretes for given initial parameters comparable to the main and control ones. The practical significance of the study is the methodology developed for predicting the strength of concrete using machine learning methods, determining the rational parameters of such a methodology and identifying factors and criteria that affect the effectiveness of the proposed solutions.

2. Materials and Methods

2.1. CatBoost Algorithm

In gradient boosting, predictions are made based on an ensemble of weak learning algorithms, while decision trees are built sequentially. The previous trees in the model are not changed and the results of the previous step are used to improve the next one. In gradient boosting, decision trees are iteratively trained in order to minimize the loss function, as shown in Figure 1.

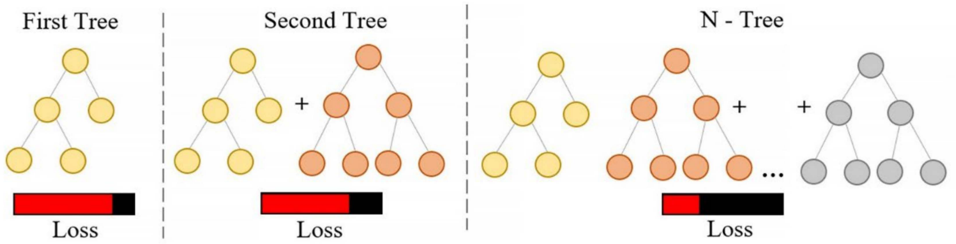


Figure 1. Iterative training of decision trees in gradient boosting.

In this study, the CatBoost method is used, which is a gradient boosting library created by Yandex. When implementing decision trees in this method, the same functions are used to create left and right splits at each level of the tree, as shown in Figure 2.

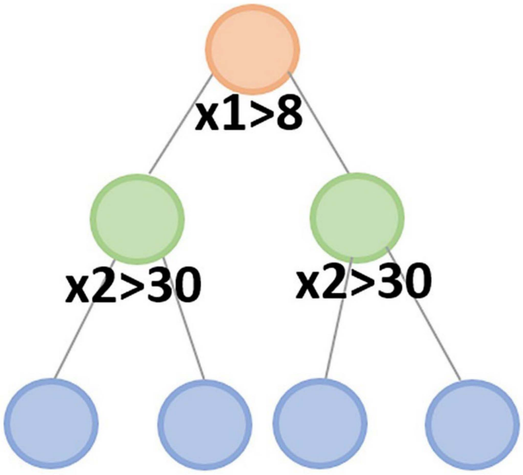


Figure 2. CatBoost decision tree for the regression problem.

Unlike some other machine learning algorithms, CatBoost works well with a small dataset; however, in such cases, you should be aware of overfitting. To avoid overfitting, the model parameters should be tuned.

2.2. k-Nearest Neighbors Method

The k-nearest neighbors method is a supervised machine learning algorithm used to solve a regression problem that performs well with a small amount of data.

In practice, the KNN method is more often used in classification problems, but currently the regression version of the k-nearest neighbors algorithm is also common. It is a good basic algorithm to try first before considering more advanced methods.

The algorithm finds the distances between the query and all examples in the data by choosing a certain number of examples (k) closest to the query, then averages the labels in the case of a regression problem.

The k-nearest neighbors algorithm follows:

1. Input:

   Training examples $\{x_i, y_i\}$
   x_i are values of training examples attributes;
   y_i are actual values of the output characteristic.
   Test point x for which we are making a prediction.

2. Forecasting:

   Calculating the distance $D(x, x_i)$ to each training example x_i;
   Selection of *k*-nearest instances and their labels $y_{i1}, \ldots, y_{ik}$;
   Determination of the mean value $\overline{y}$ for $y_{i1}, \ldots, y_{ik}$ by Formula (1):

$$\overline{y} = \frac{1}{k}\sum_{j=1}^{k} y_i \tag{1}$$

where *k* is the number of nearest instances, y_i is the actual value of the output parameter.

2.3. Support Vector Regression (SVR)

The support vector regression (SVR) was proposed based on the support vector machine (SVM) for a standard classification problem.

The SVR algorithm in its implementation as a whole is very similar to SRM, but there are several other features: SVR has an additional adjustable parameter ε (epsilon). The epsilon value determines the width of the "tube" around the evaluated function (hyperplane). Points falling inside this "tube" are considered correct predictions and are not penalized by the algorithm. Support vectors are points that extend outside the pipe, not just those that are on the edge, as in classification problems. The value of the additional sliding variable (ξ) measures the distance to points outside the pipe, which can be controlled by adjusting the regularization parameter C.

3. Materials and Dataset

3.1. Dataset Description

The data set is a table of experimental values (a series of 249 experiments) obtained from the development of laboratory compositions of self-compacting concretes. The main raw materials used were: Portland cement grade CEM I 42.5N; quartz sand with fineness modulus 1.78 (M_k = 1.78); crushed granite fraction 5–20 mm with a crushability grade of 1200; ground blast furnace granulated slag (SiO_2 = 30 ± 1%; Al_2O_3 = 9.2 ± 0.9%; Fe_2O_3 = 0.86 ± 0.08%; CaO = 33 ± 1%; SO_3 = 1.6 ± 0.2%; MgO = 5.0 ± 0.5%; Na_2O = 0.18 ± 0.02%, K_2O = 0.62 ± 0.06%, MnO = 0.82 ± 0.08%, TiO_2 = 0.46 ± 0.05%, P_2O_5 = 0.018 ± 0.002%, Cl = 0.003 ± 0.001%). As an additive, a hyperplasticizer based on polycarboxylate esters "Rheoplast PCE3240" was used. The compressive strength was determined according to GOST 10180-2012 "Concretes. Methods for strength determination using reference specimens".

The analyzed data set is presented in Supplementary Materials.

The features of machine learning models are the content of cement (kg/m^3), slag (kg/m^3), water (L), sand (kg/m^3), crushed stone (kg/m^3) and additives (kg). The predicted parameter is compressive strength (MPa).

Figure 3 shows the correlation between the variables. It is observed that the linear correlation between the individual input variables and the output variable is strong (>0.5). There is also a negative correlation, in which an increase in one variable is associated with a decrease in another. The statistical characteristics of the dataset are shown in Table 2.

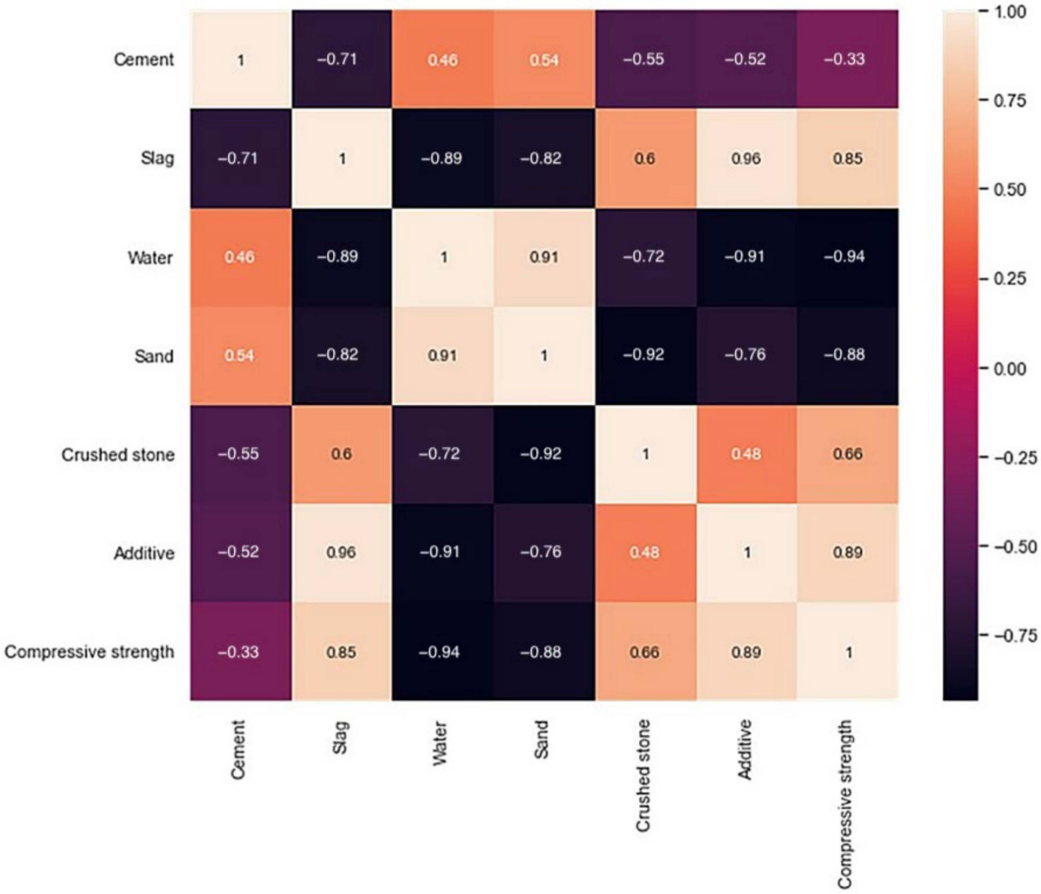


Figure 3. Correlation matrix.

Table 2. Statistical characteristics of the original dataset.

Variable	Cement	Slag	Water	Sand	Crushed Stone	Additive	Compressive Strength
Unit	kg/m^3	kg/m^3	liter	kg/m^3	kg/m^3	kg	MPa
count	249.00	249.00	249.00	249.00	249.00	249.00	249.00
mean	198.04	140.32	171.49	1027.04	805.36	4.26	38.79
std	42.27	99.57	10.47	126.76	104.96	2.16	21.87
min	150.00	47.00	150.00	790.00	715.00	2.31	9.60
max	286.00	309.00	186.00	1143.00	987.00	8.30	85.80

3.2. Performance Evaluation Methods

When analyzing regression models, it is important to use various evaluation metrics to evaluate their performance. This study uses five metrics: mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE), mean absolute percentage error (MAPE) and the coefficient of determination R^2. These metrics are defined as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3}$$

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{\hat{y}_i}\right| \times 100 \tag{5}$$

$$R^2 = \frac{\left(\sum\limits_{i=1}^{n}(y_i - \overline{y})(\hat{y}_i - \overline{\hat{y}}_i)\right)^2}{\sum\limits_{i=1}^{n}(y_i - \overline{y})^2 \sum\limits_{i=1}^{n}(\hat{y}_i - \overline{\hat{y}}_i)^2} \tag{6}$$

where y_i is the actual measured compressive strength; $\hat{y}_i$ is predicted value of compressive strength; $\overline{y}_i$ is the average value for y_i; $\overline{\hat{y}}_i$ is the mean value for $\hat{y}_i$.

4. Model Building and Training

In this study, algorithms based on machine learning methods are developed in the Jupyter Notebook interactive computing web platform in the high-level Python programming language.

The search for the optimal values of the main parameters of the model is one of the key points for achieving the best generalizing ability. In this study, the grid search method was used in combination with five-block cross-validation, which allows us to analyze all possible combinations of parameters of interest for each of the implemented models.

The general workflow of the model in the case of using cross-validation and a grid of parameters is shown in Figure 4.

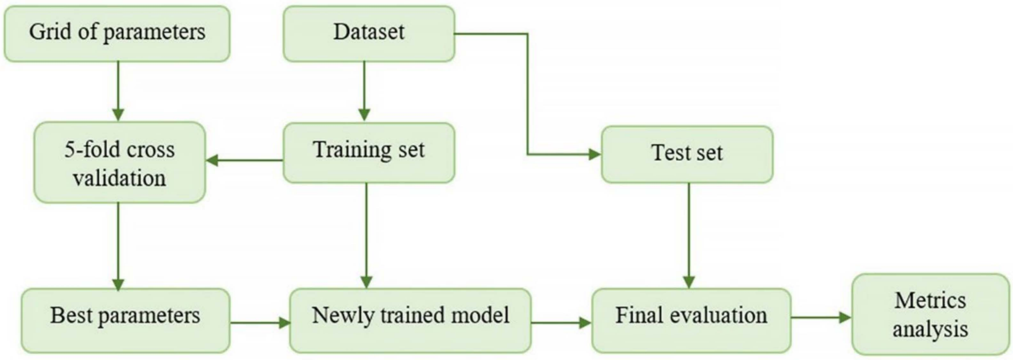


Figure 4. Parameter selection and model evaluation process using parameter grid and five-box cross-validation.

4.1. Model Building

4.1.1. Model Building for CatBoost

For the algorithm based on the CatBoost method, learning rate and tree depth are selected as adjustable parameters.

The learning rate factor is a parameter that allows control of the amount of weight correction at each iteration. In practice, the learning rate coefficient is usually selected experimentally; its tuning allows for achieving the highest possible quality of the model.

The second adjustable coefficient is the depth of the tree. In most cases, the optimal value is between 4 and 10, so this range of values is used in the parameter lattice. All possible combinations form a table–grid of model parameter settings, as shown in Table 3.

Table 3. Parameter grid for the CatBoost model.

	Depth = 4	**Depth = 6**	**Depth = 8**	**Depth = 10**
learning rate = 0.03	model (depth = 4, learning rate = 0.03)	model (depth = 6, learning rate = 0.03)	model (depth = 8, learning rate = 0.03)	model (depth = 10, learning rate = 0.03)
learning rate = 0.1	model (depth = 4, learning rate = 0.1)	model (depth = 6, learning rate = 0.1)	model (depth = 8, learning rate = 0.1)	model (depth = 10, learning rate = 0.1)
learning rate = 0.5	model (depth = 4, learning rate = 0.5)	model (depth = 6, learning rate = 0.5)	model (depth = 8, learning rate = 0.5)	model (depth = 10, learning rate = 0.5)

As a result of five-box cross-validation, for all combinations of learning rate and tree depth, we need to train 60 models (3 × 4 × 5).

Figure 5 shows a heatmap for the cross-validation average R^2 expressed as a function of two parameters: tree depth and learning rate.

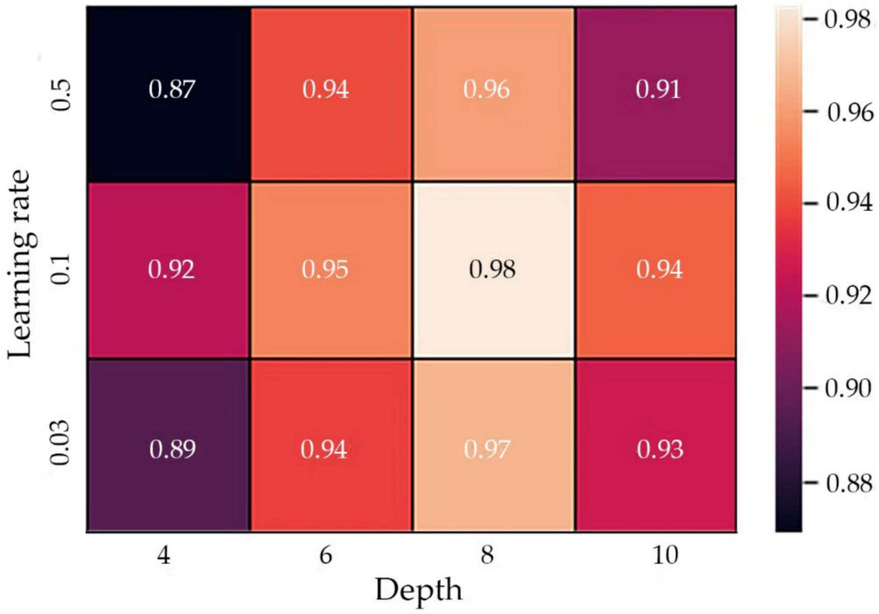


Figure 5. Heat map of R^2 value from two parameters: tree depth and learning rate.

Each heatmap value corresponds to an R^2 value for a specific combination of parameters, where light tones correspond to a high value and dark tones to a low value. It can be seen from the graph that the implemented CatBoost algorithm is sensitive to parameter settings, so their optimization is necessary to obtain good generalization ability. Various combinations of learning rates and tree depths increase R^2 from 87% (learning rate 0.5, tree depth 4) to 98% (learning rate 0.1, tree depth 8).

As a result of lattice search and cross-validation, the best parameters of the model were determined: tree depth equal to 8 and learning rate 0.1.

4.1.2. Model Building for k-Nearest Neighbors Algorithm

For the k-nearest neighbors algorithm, the following parameters were selected as adjustable parameters: the number of neighbors, the leaf size and the weight function (Table 4).

Table 4. Parameters for the k-nearest neighbor model.

Num	Parameter	Value
1	Number of neighbors	2, 5, 7, 10, 15, 20
2	Sheet size	1, 3, 5, 10, 20
3	weight function	”uniform” ”distance”

As a result of the five-block cross-validation, for all combinations of variable parameter values, we need to check the performance of 60 models (6 × 5 × 2).

An important component of the k-nearest neighbors method is normalization. Different attributes typically have different ranges of represented values in the sample, so distance values can be highly dependent on attributes with larger ranges. Therefore, the data were normalized (Z-normalization).

4.1.3. Model Building for SVR Algorithm

For the support vector machine, the following parameters are selected as adjustable parameters (Table 5):

- Kernel type: using this parameter, you can determine the type of hyperplane used for data separation; when using ”linear” a linear hyperplane is applied; a nonlinear hyperplane can also be used.
- Regularization parameter C: the strength of regularization is inversely proportional to C.
- Epsilon (ε): acceptable margin of error ε allows deviations within some threshold value.

Table 5. Parameters for SVR model.

Num	Parameter	Value
1	Kernel type	”linear” ”poly” ”rbf” ”sigmoid”
2	Regularization parameter C	1, 2, 3, 4, 5
3	Epsilon	0.1, 0.2, 0.5, 1, 1.5, 2, 3

As a result of the five-block cross-validation, for all combinations of variable parameter values, we need to check the performance of 140 models (4 × 5 × 7).

4.2. Model Training

4.2.1. Model Training CatBoost

Table 6 shows the parameters of the final CatBoost model: the number of iterations corresponding to the number of decision trees is 500; tree depth and learning rate are defined in Section 4.1.1; RMSE (3) is used as the loss function; the greedy search algorithm provides for sequential deepening of the tree; training is stopped when the error value does not decrease within 30 iterations.

Table 6. Model parameters based on CatBoost.

Num	Parameter	Value	Optional Description
1	Number of iterations	500	Number of decision trees
2	Tree depth	8	Tree structure depth
3	Learning rate	0.1	A parameter that determines the step size at each iteration when moving toward the minimum of the loss function
4	Metric used for learning	RMSE	Formula (4)
5	Greedy search algorithm	Symmetric tree	The tree is built level by level until it reaches the required depth
6	Type of overfitting detector	Early stopping	Stops training when the error value does not decrease within 30 iterations

Figure 6 shows the training schedule, according to which 65 iterations are sufficient for the model, determined by setting the overfitting detector.

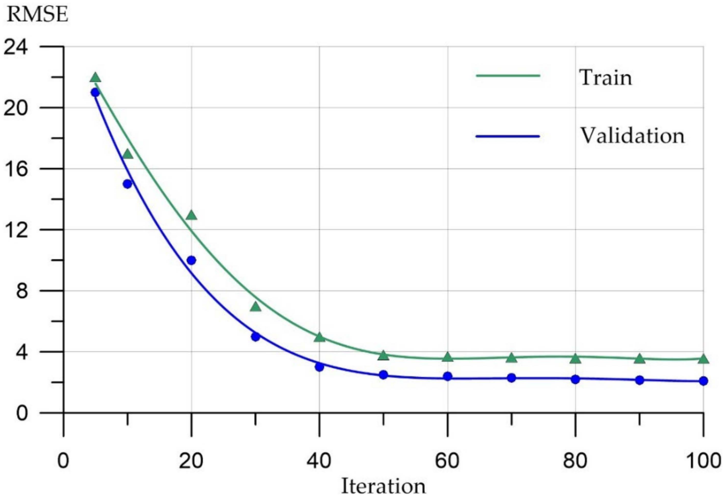


Figure 6. Model training graph based on CatBoost.

The interpretation of the gradient boosting algorithm is facilitated by the ability to represent the decision rules in the form of a visual tree structure. Figure 7 shows part of one of the decision trees. As you can see from the figure, the same functions are used to create left and right splits at each level of the tree. Owing to the peculiarities of the structure of decision trees, gradient boosting is able to cope with nonlinearities.

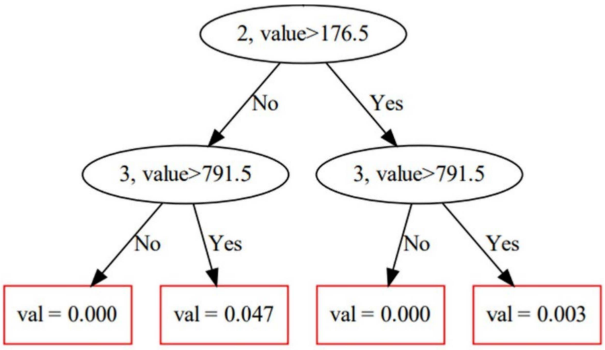


Figure 7. Visualization of the tree structure.

4.2.2. Model Training k-Nearest Neighbors

The selection of the number of neighbors parameter affects the generalizing ability of the developed model. The choice of the parameter k is important for obtaining correct model results. If the value of the parameter is small, then an overfitting effect occurs when the decision on the output characteristic is made on the basis of a small number of examples and has low significance, and it should be taken into account that the use of small values of k increases the influence of noise on the results. On the contrary, if the value of the parameter is too high, then objects that poorly reflect the local features of the data set take part in the process of solving the regression problem. Thus, the choice of the parameter k significantly affects the generalizing ability of the model.

The leaf size parameter is also significant for the model, as it affects the speed of its work along with the amount of memory used by the algorithm.

Under some circumstances, it may be beneficial to weight points so that nearby points contribute more to the regression than distant points. The "uniform" weight function setting assigns equal weights to all points, while "distance" assigns weights proportional to the reciprocal distance from the query point.

As a result of the five-box cross-validation in Section 4.1.2, the best parameters for the k-nearest neighbors model were determined (Table 7).

Table 7. Parameters of the k-nearest neighbors model.

Num	Parameter	Value
1	Number of neighbors	15
2	Sheet size	5
3	Weight function	″uniform″

4.2.3. Model Training SVR

One of the main advantages of SVR is that its computational complexity does not depend on the dimension of the input space. In addition, it has excellent generalization capabilities with high predictive accuracy when the parameters are properly tuned.

In practice, for the SVR method, the most commonly used kernel, which provides good generalization capabilities, is the radial basis function (RBF), also known as the Gaussian kernel.

There is no rule of thumb for choosing the value of C—it depends entirely on the data. The best option is to search through a grid of parameters, as in Section 4.1.3, where it is suggested to use several different values and choose the one that gives the lowest level of error in testing.

SVR is a powerful algorithm that allows us to choose how error tolerant we are with an acceptable margin of error. The epsilon parameter defines the dead zone.

Adjustment of the penalty coefficient C and the threshold value of the error ε significantly affect the mean square error in the regression model. After conducting multiple experiments with the help of cross-validation, better training results were obtained, thereby choosing the optimal values of the model parameters.

Table 8 presents the parameters of the final SVR model.

Table 8. Model parameters based on SVR.

Num	Parameter	Value
1	Kernel type	″rbf″
2	Regularization parameter C	5
3	Epsilon	0.5

4.2.4. Parallelization of the Optimization Process and Model Training

Owing to the fact that the search for optimal model parameters using parameter grids and cross-validation leads to the creation and training of a large number of models, it is worth evaluating the time spent on the algorithms.

To reduce time costs, the grid search and cross-validation were parallelized across several processor cores. Tables 9–11 show the values of two characteristics (CPU times and Wall time) depending on the number of cores involved. Loading eight processor cores allows you to reduce CPU times~15 times, and Wall time~3 times.

Table 9. The result of parallelizing the learning process across CPU cores for the CatBoost model.

Number of Cores Involved	CPU Times, s	Wall Time, s
1	16	31.6
2	2.06	22.6
4	1.73	15.2
8	1.1	10.0

Table 10. The result of parallelizing the learning process across CPU cores for the k-nearest neighbors model.

Number of Cores Involved	CPU Times, s	Wall Time, s
1	8	11.1
2	2.06	10.2
4	0.72	7.2
8	0.4	3.0

Table 11. The result of parallelizing the learning process across CPU cores for the SVR model.

Number of Cores Involved	CPU Times, s	Wall Time, s
1	9	12.4
2	2.18	9.4
4	0.75	6.1
8	0.6	3.0

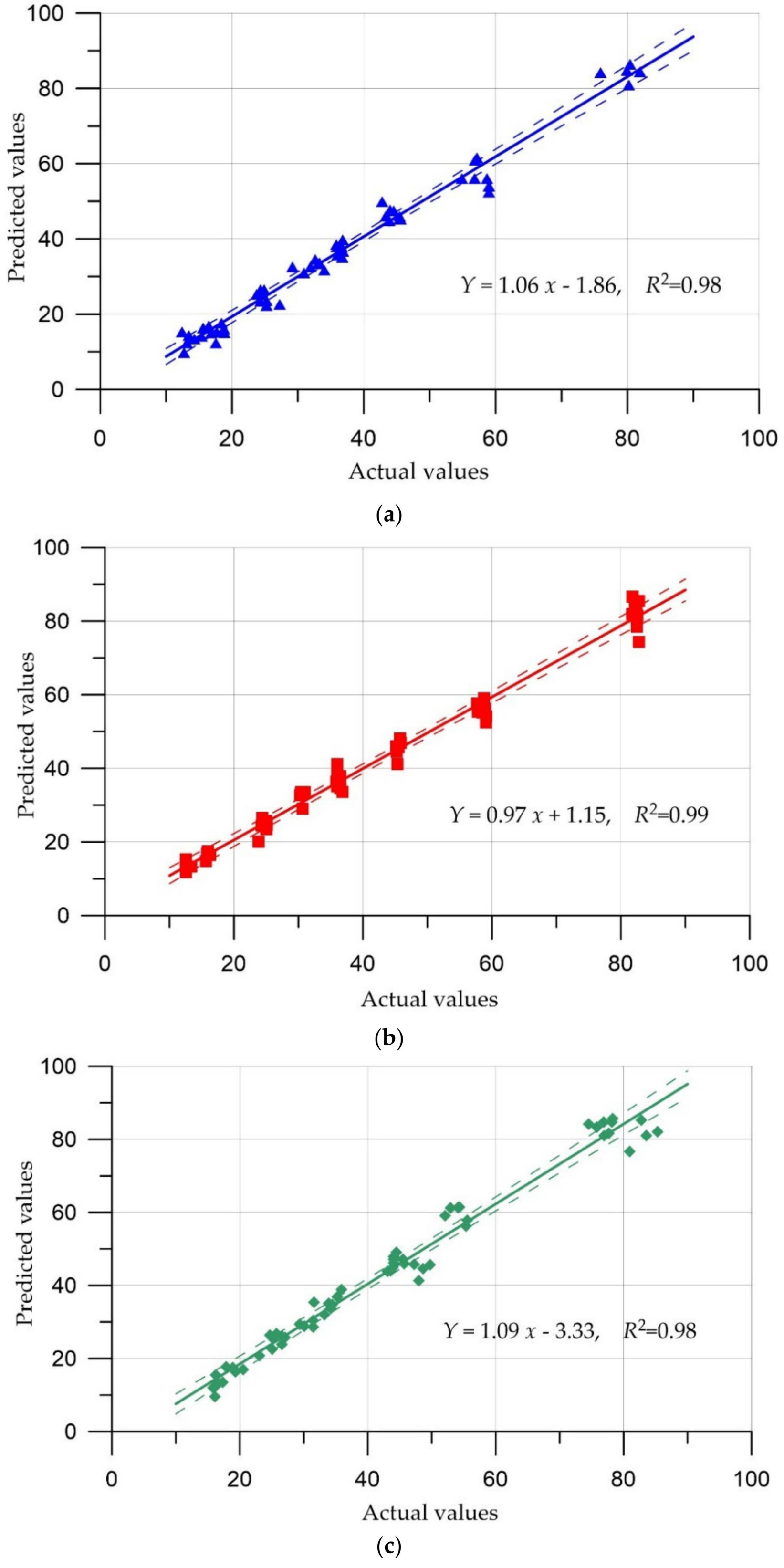
5. Comparison of Prediction Results

Prediction error plots (Figure 8) show the actual values from the dataset versus the predicted values generated by our model. This visualization method allows you to see how large the variance is in the model.

Table 12 presents the values of the metrics selected to evaluate the developed models. Figure 9 shows graphs visualizing this table.

Table 12. Metrics of the developed models.

№	Model	MAE	MSE	RMSE	MAPE, %	R^2
1	CatBoost (CB)	2.17	7.8	2.79	6.84	0.98
2	K-nearest neighbors (KNN)	1.97	6.85	2.62	6.15	0.99
3	SVR	2.61	11.39	3.37	7.89	0.98



(**a**)

(**b**)

(**c**)

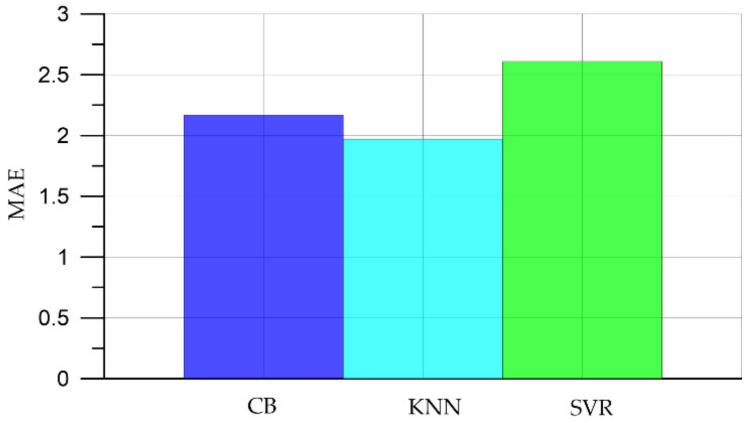
Figure 8. Relationship between actual compressive strength and calculated values (**a**) for the CatBoost model; (**b**) for the k-nearest neighbors model; (**c**) for the SVR model.

Considering the fact that the developed machine learning algorithms were applied on a series of experimental data obtained when testing concrete, which is a heterogeneous material that depends on a large number of factors and significantly differs in properties and structure in its volume, the following should be noted. The scatter of data when measuring the characteristics of such a material exists regardless of knowledge about them, many of the heterogeneities in concrete are uncontrollable either from the point of view of either the recipe or the technology. Therefore, there is always a data error, which is within 10% and is an acceptable norm in the production of concrete.

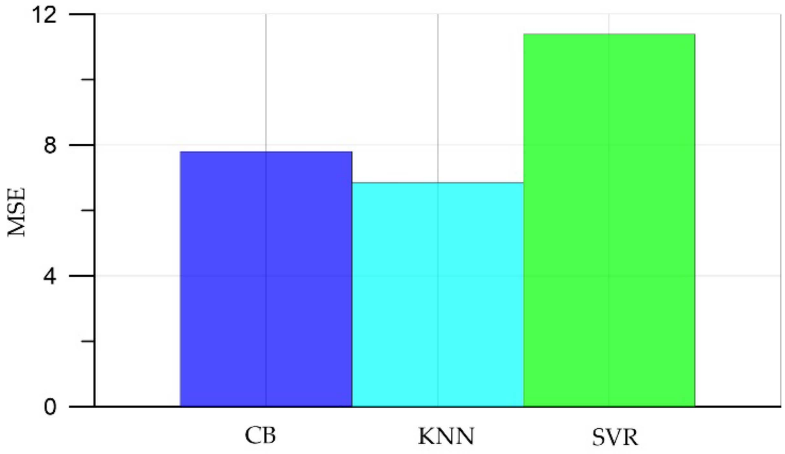
The results of the study showed that the coefficient of determination for the developed models is quite high, 0.98–0.99, while the observed value is higher than that reported in [27], which is explained by the homogeneity of the initial data set and the tuning of the hyperparameters of the models.

MAE values are in the range from 1.97 to 2.61, MSE from 6.85 to 11.39 and RMSE from 2.62 to 3.37, which are consistent with the results of previous studies by other authors [25,26].

The MAPE value (6.15–7.89%) obtained by testing the developed machine learning models is acceptable; the models can be verified and accepted for use in determining the compressive strength of self-compacting concrete, considering all available data. The accuracy of the models is comparable to the normative and technical documents for concrete in global practice.

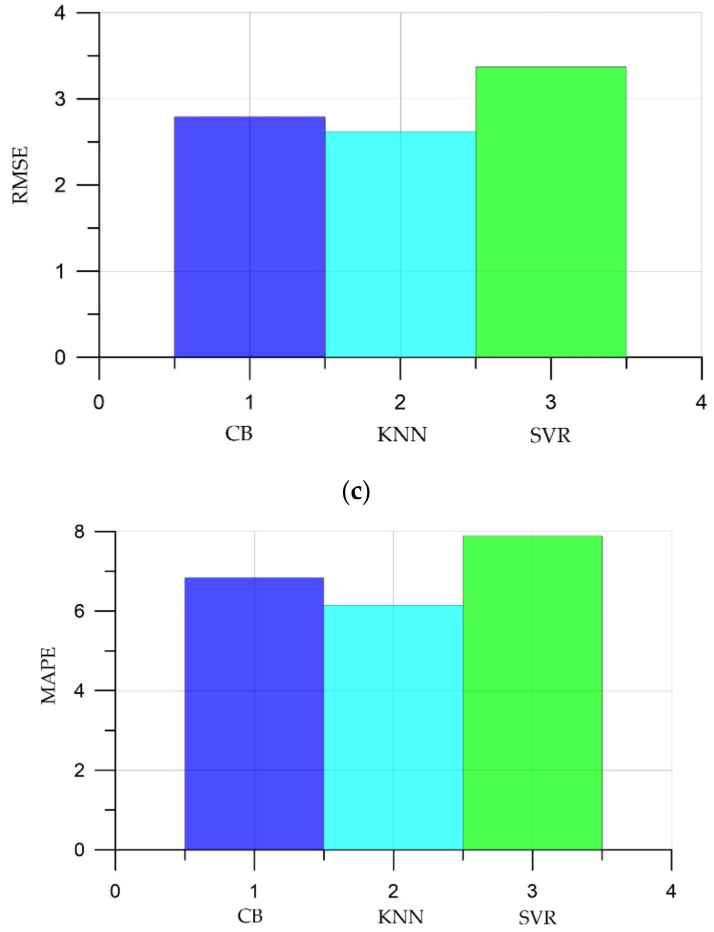


(**a**)



(**b**)

Figure 9. *Cont.*



(**c**)

(**d**)

Figure 9. Metric values for the developed regression models: (**a**) MAE; (**b**) MSE; (**d**) RMSE; (**c**) MAPE.

6. Conclusions

(1) Development and comparison of three machine learning algorithms based on CatBoost gradient boosting, k-nearest neighbors (KNN) and support vector regression (SVR) were used to predict the compressive strength of self-compacting concrete by applying our accumulated empirical database and data.

(2) It has been established that artificial intelligence methods can be applied to determine the compressive strength of self-compacting concrete. The developed models showed a mean absolute percentage error (MAPE) in the range 6.15–7.89%.

(3) Of the three machine learning algorithms, the smallest errors and the largest coefficient of determination were observed in the KNN algorithm: MAE was 1.97; MSE, 6.85; RMSE, 2.62; MAPE, 6.15; and the coefficient of determination R^2, 0.99.

(4) Models can be verified and accepted for use in determining the compressive strength of self-compacting concrete, taking into account all available data.

(5) The developed methods can be successfully implemented in the process of production and quality control of building materials, since they do not require serious computing resources and, in the future, based on artificial intelligence, an expert system can be created to summarize all of the accumulated experimental data, which can be located in an electronic environment university and provide data to interested workers and researchers for the development of the industry.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app122110864/s1, Table S1: Dataset.

Author Contributions: Conceptualization, S.A.S., E.M.S., A.N.B. and I.R.; methodology, S.A.S., E.M.S. and I.R.; software, I.R., A.C. and N.B.; validation, I.R., S.A.S., E.M.S. and A.N.B.; formal analysis, I.R., A.C. and N.B.; investigation, L.R.M., S.A.S., E.M.S., A.N.B. and I.R.; resources, B.M.; data curation, S.A.S., E.M.S. and I.R.; writing—original draft preparation, I.R., S.A.S., E.M.S. and A.N.B.; writing—review and editing, I.R., S.A.S., E.M.S. and A.N.B.; visualization, I.R., S.A.S., E.M.S., A.N.B. and N.B.; supervision, L.R.M. and B.M.; project administration, L.R.M. and B.M.; funding acquisition, A.N.B. and B.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The study did not report any data.

Acknowledgments: The authors would like to acknowledge the administration of Don State Technical University for their resources and financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chandra, S.; Björnström, J. Influence of superplasticizer type and dosage on the slump loss of Portland cement mortars—Part II. *Cem. Concr. Res.* **2002**, *32*, 1613–1619. [CrossRef]
2. Farooq, F.; Czarnecki, S.; Niewiadomski, P.; Aslam, F.; Alabduljabbar, H.; Ostrowski, K.A.; Śliwa-Wieczorek, K.; Nowobilski, T.; Malazdrewicz, S. A Comparative Study for the Prediction of the Compressive Strength of Self-Compacting Concrete Modified with Fly Ash. *Materials* **2021**, *14*, 4934. [CrossRef] [PubMed]
3. Beskopylny, A.N.; Shcherban', E.M.; Stel'makh, S.A.; Mailyan, L.R.; Meskhi, B.; Evtushenko, A.; Varavka, V.; Beskopylny, N. Nano-Modified Vibrocentrifuged Concrete with Granulated Blast Slag: The Relationship between Mechanical Properties and Micro-Structural Analysis. *Materials* **2022**, *15*, 4254. [CrossRef]
4. Beskopylny, A.N.; Stel'makh, S.A.; Shcherban', E.M.; Mailyan, L.R.; Meskhi, B.; Beskopylny, N.; El'shaeva, D.; Kotenko, M. The Investigation of Compacting Cement Systems for Studying the Fundamental Process of Cement Gel Formation. *Gels* **2022**, *8*, 530. [CrossRef] [PubMed]
5. Stel'makh, S.A.; Shcherban', E.M.; Beskopylny, A.; Mailyan, L.R.; Meskhi, B.; Beskopylny, N.; Zherebtsov, Y. Development of High-Tech Self-Compacting Concrete Mixtures Based on Nano-Modifiers of Various Types. *Materials* **2022**, *15*, 2739. [CrossRef]
6. Beskopylny, A.N.; Stel'makh, S.A.; Shcherban', E.M.; Mailyan, L.R.; Meskhi, B.; Varavka, V.; Beskopylny, N.; El'shaeva, D. A Study on the Cement Gel Formation Process during the Creation of Nanomodified High-Performance Concrete Based on Nanosilica. *Gels* **2022**, *8*, 346. [CrossRef]
7. Beskopylny, A.N.; Meskhi, B.; Stel'makh, S.A.; Shcherban', E.M.; Mailyan, L.R.; Veremeenko, A.; Akopyan, V.; Shilov, A.V.; Chernil'nik, A.; Beskopylny, N. Numerical Simulation of the Bearing Capacity of Variotropic Short Concrete Beams Reinforced with Polymer Composite Reinforcing Bars. *Polymers* **2022**, *14*, 3051. [CrossRef]
8. Dudukalov, E.V.; Munister, V.D.; Zolkin, A.L.; Losev, A.N.; Knishov, A.V. The use of artificial intelligence and information technology for measurements in mechanical engineering and in process automation systems in Industry 4.0. *J. Phys. Conf. Ser.* **2021**, *1889*, 052011. [CrossRef]
9. Muhammad, W.; Brahme, A.P.; Ibragimova, O.; Kang, J.; Inal, K. A machine learning framework to predict local strain distribution and the evolution of plastic anisotropy & fracture in additively manufactured alloys. *Int. J. Plast.* **2021**, *136*, 102867. [CrossRef]
10. Oh, W.B.; Yun, T.J.; Lee, B.R.; Kim, C.G.; Liang, Z.L.; Kim, I.S. A Study on Intelligent Algorithm to Control Welding Parameters for Lap-joint. *Procedia Manuf.* **2019**, *30*, 48–55. [CrossRef]
11. Patel, A.R.; Ramaiya, K.K.; Bhatia, C.V. Artificial Intelligence: Prospect in Mechanical Engineering Field-A Review. *Lect. Notes Data Eng. Commun. Technol.* **2021**, *52*, 267–282. [CrossRef]
12. Tosee, S.V.R.; Faridmehr, I.; Bedon, C.; Sadowski, Ł.; Aalimahmoody, N.; Nikoo, M.; Nowobilski, T. Metaheuristic Prediction of the Compressive Strength of Environmentally Friendly Concrete Modified with Eggshell Powder Using the Hybrid ANN-SFL Optimization Algorithm. *Materials* **2021**, *14*, 6172. [CrossRef]
13. Ahmad, A.; Ahmad, W.; Chaiyasarn, K.; Ostrowski, K.A.; Aslam, F.; Zajdel, P.; Joyklad, P. Prediction of Geopolymer Concrete Compressive Strength Using Novel Machine Learning Algorithms. *Polymers* **2021**, *13*, 3389. [CrossRef] [PubMed]
14. Beskopylny, A.; Lyapin, A.; Anysz, H.; Meskhi, B.; Veremeenko, A.; Mozgovoy, A. Artificial Neural Networks in Classification of Steel Grades Based on Non-Destructive Tests. *Materials* **2020**, *13*, 2445. [CrossRef] [PubMed]
15. Liu, F.; Xu, J.; Tan, S.; Gong, A.; Li, H. Orthogonal Experiments and Neural Networks Analysis of Concrete Performance. *Water* **2022**, *14*, 2520. [CrossRef]

16. Islam, M.M.; Hossain, M.B.; Akhtar, M.N.; Moni, M.A.; Hasan, K.F. CNN Based on Transfer Learning Models Using Data Augmentation and Transformation for Detection of Concrete Crack. *Algorithms* **2022**, *15*, 287. [CrossRef]
17. Ni, X.; Duan, K. Machine Learning-Based Models for Shear Strength Prediction of UHPFRC Beams. *Mathematics* **2022**, *10*, 2918. [CrossRef]
18. Rahman, S.K.; Al-Ameri, R. Experimental and Artificial Neural Network-Based Study on the Sorptivity Characteristics of Geopolymer Concrete with Recycled Cementitious Materials and Basalt Fibres. *Recycling* **2022**, *7*, 55. [CrossRef]
19. Shah, H.A.; Yuan, Q.; Akmal, U.; Shah, S.A.; Salmi, A.; Awad, Y.A.; Shah, L.A.; Iftikhar, Y.; Javed, M.H.; Khan, M.I. Application of Machine Learning Techniques for Predicting Compressive, Splitting Tensile, and Flexural Strengths of Concrete with Metakaolin. *Materials* **2022**, *15*, 5435. [CrossRef]
20. Behnood, A.; Behnood, V.; Gharehveran, M.M.; Alyamac, K.E. Prediction of the compressive strength of normal and high-performance concretes using M5P model tree algorithm. *Constr. Build. Mater.* **2017**, *142*, 199–207. [CrossRef]
21. Behnood, A.; Olek, J.; Glinicki, M.A. Predicting modulus elasticity of recycled aggregate concrete using M5′ model tree algorithm. *Constr. Build. Mater.* **2015**, *94*, 137–147. [CrossRef]
22. Naderpour, H.; Rafiean, A.H.; Fakharian, P. Compressive strength prediction of environmentally friendly concrete using artificial neural networks. *J. Build. Eng.* **2018**, *16*, 213–219. [CrossRef]
23. Getahun, M.A.; Shitote, S.M.; Gariy, Z.A. Artificial neural network based modelling approach for strength prediction of concrete incorporating agricultural and construction wastes. *Constr. Build. Mater.* **2018**, *190*, 517–525. [CrossRef]
24. Hadzima-Nyarko, M.; Nyarko, E.K.; Ademović, N.; Miličević, I.; Kalman Šipoš, T. Modelling the Influence of Waste Rubber on Compressive Strength of Concrete by Artificial Neural Networks. *Materials* **2019**, *12*, 561. [CrossRef]
25. De-Prado-Gil, J.; Palencia, C.; Jagadesh, P.; Martínez-García, R. A Study on the Prediction of Compressive Strength of Self-Compacting Recycled Aggregate Concrete Utilizing Novel Computational Approaches. *Materials* **2022**, *15*, 5232. [CrossRef]
26. Ghafor, K. Multifunctional Models, Including an Artificial Neural Network, to Predict the Compressive Strength of Self-Compacting Concrete. *Appl. Sci.* **2022**, *12*, 8161. [CrossRef]
27. De-Prado-Gil, J.; Palencia, C.; Silva-Monteiro, N.; Martínez-García, R. To predict the compressive strength of self compacting concrete with recycled aggregates utilizing ensemble machine learning models. *Case Stud. Constr. Mater.* **2022**, *16*, e01046. [CrossRef]
28. Chandramouli, P.; Jayaseelan, R.; Pandulu, G.; Sathish Kumar, V.; Murali, G.; Vatin, N.I. Estimating the Axial Compression Capacity of Concrete-Filled Double-Skin Tubular Columns with Metallic and Non-Metallic Composite Materials. *Materials* **2022**, *15*, 3567. [CrossRef]
29. Kuppusamy, Y.; Jayaseelan, R.; Pandulu, G.; Sathish Kumar, V.; Murali, G.; Dixit, S.; Vatin, N.I. Artificial Neural Network with a Cross-Validation Technique to Predict the Material Design of Eco-Friendly Engineered Geopolymer Composites. *Materials* **2022**, *15*, 3443. [CrossRef]
30. Amin, M.N.; Ahmad, A.; Khan, K.; Ahmad, W.; Ehsan, S.; Alabdullah, A.A. Predicting the Rheological Properties of Super-Plasticized Concrete Using Modeling Techniques. *Materials* **2022**, *15*, 5208. [CrossRef]
31. Ilyas, I.; Zafar, A.; Afzal, M.T.; Javed, M.F.; Alrowais, R.; Althoey, F.; Mohamed, A.M.; Mohamed, A.; Vatin, N.I. Advanced Machine Learning Modeling Approach for Prediction of Compressive Strength of FRP Confined Concrete Using Multiphysics Genetic Expression Programming. *Polymers* **2022**, *14*, 1789. [CrossRef] [PubMed]
32. Koo, S.; Shin, D.; Kim, C. Application of Principal Component Analysis Approach to Predict Shear Strength of Reinforced Concrete Beams with Stirrups. *Materials* **2021**, *14*, 3471. [CrossRef]
33. Faridmehr, I.; Nehdi, M.L.; Huseien, G.F.; Baghban, M.H.; Sam, A.R.M.; Algaifi, H.A. Experimental and Informational Modeling Study of Sustainable Self-Compacting Geopolymer Concrete. *Sustainability* **2021**, *13*, 7444. [CrossRef]
34. Amin, M.N.; Iqtidar, A.; Khan, K.; Javed, M.F.; Shalabi, F.I.; Qadir, M.G. Comparison of Machine Learning Approaches with Traditional Methods for Predicting the Compressive Strength of Rice Husk Ash Concrete. *Crystals* **2021**, *11*, 779. [CrossRef]
35. Dabbaghi, F.; Rashidi, M.; Nehdi, M.L.; Sadeghi, H.; Karimaei, M.; Rasekh, H.; Qaderi, F. Experimental and Informational Modeling Study on Flexural Strength of Eco-Friendly Concrete Incorporating Coal Waste. *Sustainability* **2021**, *13*, 7506. [CrossRef]
36. Wu, N.-J. Predicting the Compressive Strength of Concrete Using an RBF-ANN Model. *Appl. Sci.* **2021**, *11*, 6382. [CrossRef]
37. Bu, L.; Du, G.; Hou, Q. Prediction of the Compressive Strength of Recycled Aggregate Concrete Based on Artificial Neural Network. *Materials* **2021**, *14*, 3921. [CrossRef] [PubMed]
38. Ahmad, A.; Chaiyasarn, K.; Farooq, F.; Ahmad, W.; Suparp, S.; Aslam, F. Compressive Strength Prediction via Gene Expression Programming (GEP) and Artificial Neural Network (ANN) for Concrete Containing RCA. *Buildings* **2021**, *11*, 324. [CrossRef]
39. Suescum-Morales, D.; Salas-Morera, L.; Jiménez, J.R.; García-Hernández, L. A Novel Artificial Neural Network to Predict Compressive Strength of Recycled Aggregate Concrete. *Appl. Sci.* **2021**, *11*, 11077. [CrossRef]
40. Song, H.; Ahmad, A.; Ostrowski, K.A.; Dudek, M. Analyzing the Compressive Strength of Ceramic Waste-Based Concrete Using Experiment and Artificial Neural Network (ANN) Approach. *Materials* **2021**, *14*, 4518. [CrossRef]
41. Kekez, S.; Kubica, J. Application of Artificial Neural Networks for Prediction of Mechanical Properties of CNT/CNF Reinforced Concrete. *Materials* **2021**, *14*, 5637. [CrossRef] [PubMed]
42. Maqsoom, A.; Aslam, B.; Gul, M.E.; Ullah, F.; Kouzani, A.Z.; Mahmud, M.A.P.; Nawaz, A. Using Multivariate Regression and ANN Models to Predict Properties of Concrete Cured under Hot Weather. *Sustainability* **2021**, *13*, 10164. [CrossRef]

43. Kovačević, M.; Lozančić, S.; Nyarko, E.K.; Hadzima-Nyarko, M. Modeling of Compressive Strength of Self-Compacting Rubberized Concrete Using Machine Learning. *Materials* **2021**, *14*, 4346. [CrossRef]
44. Ahmad, A.; Ostrowski, K.A.; Maślak, M.; Farooq, F.; Mehmood, I.; Nafees, A. Comparative Study of Supervised Machine Learning Algorithms for Predicting the Compressive Strength of Concrete at High Temperature. *Materials* **2021**, *14*, 4222. [CrossRef]
45. Stel'makh, S.A.; Shcherban', E.M.; Beskopylny, A.N.; Mailyan, L.R.; Meskhi, B.; Razveeva, I.; Kozhakin, A.; Beskopylny, N. Prediction of Mechanical Properties of Highly Functional Lightweight Fiber-Reinforced Concrete Based on Deep Neural Network and Ensemble Regression Trees Methods. *Materials* **2022**, *15*, 6740. [CrossRef] [PubMed]
46. Rajadurai, R.-S.; Kang, S.-T. Automated Vision-Based Crack Detection on Concrete Surfaces Using Deep Learning. *Appl. Sci.* **2021**, *11*, 5229. [CrossRef]
47. Bin Khairul Anuar, M.A.R.; Ngamkhanong, C.; Wu, Y.; Kaewunruen, S. Recycled Aggregates Concrete Compressive Strength Prediction Using Artificial Neural Networks (ANNs). *Infrastructures* **2021**, *6*, 17. [CrossRef]
48. Palevičius, P.; Pal, M.; Landauskas, M.; Orinaitė, U.; Timofejeva, I.; Ragulskis, M. Automatic Detection of Cracks on Concrete Surfaces in the Presence of Shadows. *Sensors* **2022**, *22*, 3662. [CrossRef]
49. Sarir, P.; Armaghani, D.J.; Jiang, H.; Sabri, M.M.S.; He, B.; Ulrikh, D.V. Prediction of Bearing Capacity of the Square Concrete-Filled Steel Tube Columns: An Application of Metaheuristic-Based Neural Network Models. *Materials* **2022**, *15*, 3309. [CrossRef]
50. Deifalla, A.; Salem, N.M. A Machine Learning Model for Torsion Strength of Externally Bonded FRP-Reinforced Concrete Beams. *Polymers* **2022**, *14*, 1824. [CrossRef]
51. Kim, B.; Choi, S.-W.; Hu, G.; Lee, D.-E.; Serfa Juan, R.O. An Automated Image-Based Multivariant Concrete Defect Recognition Using a Convolutional Neural Network with an Integrated Pooling Module. *Sensors* **2022**, *22*, 3118. [CrossRef] [PubMed]
52. Khokhar, S.A.; Ahmed, T.; Khushnood, R.A.; Ali, S.M.; Shahnawaz. A Predictive Mimicker of Fracture Behavior in Fiber Reinforced Concrete Using Machine Learning. *Materials* **2021**, *14*, 7669. [CrossRef]
53. Lavercombe, A.; Huang, X.; Kaewunruen, S. Machine Learning Application to Eco-Friendly Concrete Design for Decarbonisation. *Sustainability* **2021**, *13*, 13663. [CrossRef]
54. Nafees, A.; Javed, M.F.; Khan, S.; Nazir, K.; Farooq, F.; Aslam, F.; Musarat, M.A.; Vatin, N.I. Predictive Modeling of Mechanical Properties of Silica Fume-Based Green Concrete Using Artificial Intelligence Approaches: MLPNN, ANFIS, and GEP. *Materials* **2021**, *14*, 7531. [CrossRef] [PubMed]