


Article

Utilisation of Machine Learning Techniques to Model Creep Behaviour of Low-Carbon Concretes

Yanni Bouras * and Le Li 

Institute of Sustainable Industries and Liveable Cities, College of Sport, Health and Engineering, Victoria University, Melbourne 3011, Australia; le.li@vu.edu.au

* Correspondence: yanni.bouras@vu.edu.au

Abstract: Low-carbon concrete mixes that incorporate high volumes of fly ash and slag as cement replacements are becoming increasingly more common as part of efforts to decarbonise the construction industry. Though environmental benefits are offered, concretes containing supplementary cementitious materials exhibit different creep behaviour when compared to conventional concrete. Creep can significantly impact long-term structural behaviour and influence the overall serviceability and durability of concrete structures. This paper develops a creep compliance prediction model using supervised machine learning techniques for concretes containing fly ash and slag as cement substitutes. Gaussian process regression (GPR), artificial neural networks (ANN), random forest regression (RFR) and decision tree regression (DTR) models were all considered. The dataset for model training was developed by mining relevant data from the Infrastructure Technology Institute of Northwestern University's comprehensive creep dataset in addition to extracting data from the literature. Holdout validation was adopted with the data partitioned into training (70%) and validation (30%) sets. Based on statistical indicators, all machine learning models can accurately model creep compliance with the RFR and GPR found to be the best-performing models. The sensitivity of the GPR model's performance to training repetitions, input variable selection and validation methodology was assessed, with the results indicating small variability. The importance of the selected input variables was analysed using the Shapley additive explanation. It was found that time was the most significant parameter, with loading age, compressive strength, elastic modulus, volume-to-surface ratio and relative humidity also showing high importance. Fly ash and silica fume content featured the least influence on creep prediction. Furthermore, the predictions of the trained models were compared to experimental data, which showed that the GPR, RFR and ANN models can accurately reflect creep behaviour and that the DTR model does not give accurate predictions.

Keywords: artificial neural networks; concrete; creep; machine learning; sustainability



Citation: Bouras, Y.; Li, L. Utilisation of Machine Learning Techniques to Model Creep Behaviour of Low-Carbon Concretes. *Buildings* **2023**, *13*, 2252. <https://doi.org/10.3390/buildings13092252>

Academic Editor: Jan Fořt

Received: 7 August 2023

Revised: 28 August 2023

Accepted: 4 September 2023

Published: 5 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

It is estimated that the construction industry accounts for 39% of global carbon dioxide (CO₂) emissions [1]. Cement alone makes up approximately 5% of global anthropogenic CO₂ emissions [2] due to its high embodied carbon and concrete's great consumption rates—concrete is the second most consumed material on the planet, second only to water. With the introduction of the United Nation's (UN) Sustainable Development Goals (SDGs) and increasing national and global sustainability targets, the construction industry must adopt greener construction practices. One approach to reducing the embodied carbon of cement and concrete-based materials is through the partial substitution of cement with materials featuring reduced environmental impacts and cementitious properties, referred to as Supplementary Cementitious Materials (SCMs). Fly ash and slag, bi-products of coal burning and steel production, respectively, have been studied extensively and are commonly adopted as SCMs in the industry. Replacing high volumes of ordinary Portland cement (OPC) with SCMs is becoming increasingly common in the construction industry due to the high CO₂ emission savings that can be achieved.

Blending fly ash or slag with OPC leads to the creation of additional hydration products, particularly calcium silicate hydrate (C-S-H), which is the primary component of strength and durability in concrete. However, negative changes to the microstructure of the cement matrix may be observed, including larger pore volumes with a finer pore distribution. This can reduce durability performance due to the greater potential for chloride ion permeability and water absorption [3]. Creep and shrinkage are important durability properties of concrete that significantly impact long-term performance and are influenced by the incorporation of mineral admixtures. Creep in concrete is a time-dependent, macroscopic deformation that occurs due to sustained loading [4]. Total creep strain is characterised by two components: drying creep and basic creep. Basic creep is the component of total creep strain that grows with time under sustained loading and is not influenced by drying effects. The drying creep element is the portion of creep strain that occurs when loaded concrete loses moisture and dries. Substitution of OPC with SCMs has been found to have various effects on creep strain. Shariq et al. [5] experimentally investigated creep and shrinkage strains in concrete containing Ground Granulated Blast-Furnace Slag (GGBFS) at replacement ratios of 20%, 40% and 60%. Creep and shrinkage strains were found to increase with GGBFS content. Average creep coefficients for 20%, 40% and 60% GGBFS concrete were 16.3%, 33.3% and 55.2% higher than concrete with no GGBFS concrete, respectively, after 150 days of loading. Conversely, many researchers have found that SCMs may lead to reduced creep strains. Qin et al. [6] analysed creep and shrinkage in self-compacting concrete (SCC) containing recycled aggregates and various blends of fly ash, slag and silica fume and observed that incorporating combinations of SCMs led to reductions in creep strains. Similar behaviour has been reported for the inclusion of ultra-fine GGBFS and silica fume in high-performance concrete (HPC) by Jianyong and Yan [7]. Creep strain more than halved when substituting 30% of OPC with ultra-fine GBFS and reduced by approximately 65% when a blend of silica fume and GBFS were used at 10% and 30% substitution ratios, respectively. Gedam et al. [8] experimentally investigated four different HPC mixes that included either fly ash, silica fume or GGBFS as an SCM. After 500 days of loading, total creep in mixes containing GGBFS, fly ash and silica fume were 40.12%, 22.49%, and 57.11% lower, respectively, than that of the reference mix containing OPC only. Reduced creep due to SCM incorporation has been attributed to additional C-S-H gel formation, which effectively refines early-age micro-cracks and improves the physical performance of HPC [8]. Chern and Chan [9] examined basic and drying creep in blast furnace slag concrete. The experimental test results suggested that basic creep is lower in concrete containing GGBFS when compared to OPC concrete. However, drying creep and total deformation were found to increase when incorporating the SCM. It is clear from the above that the effect of SCMs on the time-dependent behaviour of concrete is complex and depends on a multitude of factors, including the material properties of cement and SCMs, mix design, and curing and mixing conditions.

The application of artificial intelligence (AI) in many fields is growing rapidly, including in engineering and material sciences [10,11]. Complex systems and behaviour such as concrete creep can be accurately predicted using machine learning (ML) algorithms trained on comprehensive datasets [12–14]. Additionally, ML can be employed to elucidate underlying mechanisms governing behaviour. Molecular dynamics simulation is another emerging technique to predict material properties and behaviour [15]. Artificial Neural Networks (ANNs) and regression analysis are commonly adopted ML techniques for civil engineering systems. The Infrastructure Technology Institute of Northwestern University's (NU-ITI) concrete creep and shrinkage database [16] is a valuable tool that can be harnessed for deep analysis of time-dependent behaviour in concrete. A number of researchers have utilised the NU-ITI dataset to develop accurate prediction models for concrete creep using ML techniques [17–20]. Developing creep models for sustainable concrete mixes using ML has also received some research attention. Back-propagation neural network (BPNN) and support vector machine (SVM) models were utilised by Rong et al. [21] to investigate the drying creep of recycled aggregate concrete (RAC). Various ML models were

compared for their suitability in predicting total creep in RAC by Feng et al. [22,23]. Few research works have attempted to analyse creep in concrete containing SCMs using ML. Sadowski et al. [24] employed a firefly-algorithm (FA)-based ANN technique for modelling creep in concrete containing GGBFS. The developed model was highly accurate based on an assessment of the following statistical indicators: Correlation coefficient R^2 , mean absolute error (MAE), mean squared error (MSE) and root mean square error (RMSE). However, the dataset utilised for ML model training was limited to 12 different mix designs and only 132 individual data points. All specimens were loaded at 28 days at 50% of first-crack strength, with loading sustained for 150 days. OPC replacement ratios ranged from 0–60% at increments of 20%. Gedam et al. [25] applied ANNs to develop drying shrinkage and specific creep prediction models for HPC. A set of 106 experimental results, consisting of 2176 data points, was used to develop the dataset for model training for both drying shrinkage and creep prediction. It is unclear how many data points were specifically for creep measurements. The academic references used to source the data are listed, though many of them are not available in the open literature. Compressive strengths ranged from 20–83 MPa, and loading ages from 7–73 days. The developed models were able to accurately reflect creep behaviour and outperformed existing creep equations. A modified ACI model that considers the effect of SCMs on creep strain was developed by Chen et al. [26] using particle-swarm optimisation. Liu et al. [27] analysed creep in concrete containing SCMs using linear regression, support vector regression (SVR), random forest (RF) and extreme gradient boosting (XGB) techniques. A large database was constructed using experimental results reported in the literature. The influence of each adopted input parameter on creep prediction was investigated by conducting a sensitivity analysis based on the Shapley additive explanation (SHAP). The strength-to-stress ratio was determined to be the most influential input parameter, followed by time. This is likely due to compressive strength not being adopted as an input parameter directly and is instead reflected through applied stress and strength-to-stress ratio. Furthermore, the XGB model was found to accurately predict the creep compliance of SCM concrete and performed better than the other ML models considered. This study, however, did not consider curing conditions on creep behaviour aside from loading age.

In this paper, ML techniques are employed to predict and analyse creep in concretes containing SCMs, including fly ash, slag and silica fume. ANN, RF, and decision tree regression (DTR) are considered in this study as they are commonly adopted for concrete property prediction [28]. Gaussian process regression (GPR) model is also considered, as the ML techniques mentioned above may lead to overfitting. This paper builds on the existing literature by considering high volumes of cement substitution with fly ash and slag and through investigation of various input variables not considered in previous ML studies. Additionally, a sizable database is developed, consisting of creep data that are available either publicly or in the open literature. This study is necessitated by the increasing use of concrete with high-volume fly ash and/or slag as cement substitutes and the complexities associated with accurately modelling creep. The paper first presents a discussion on the development of the database and the selection of input variables. Subsequently, the background theory for each adopted ML technique is outlined. The performances of the ANN, RF, DTR and GPR models are then analysed and compared. Sensitivity and Shapley value additive (SHAP) analyses are also conducted, which assess the influence of input variables on model performance and creep behaviour, respectively.

2. Concrete Creep Database and Input Variables

A database was compiled by supplementing the Infrastructure Technology Institute of Northwestern University's (NU-ITI) database for concrete creep and shrinkage [16] with experimental results from the literature. Data relating to concrete containing fly ash and/or slag were first extracted from the NU-ITI dataset, including any related control tests. This included nine creep curves for concrete containing slag (185 data points) and six control creep curves (97 data points) that did not feature any slag content. For fly ash-based

concrete, 64 curves (1111 data points) and 18 control curves (159 data points) were extracted. The following changes were made to filter and clean the extracted data;

- ‘Suspicious data’, as labelled in the NU-ITI database, was removed
- Negative time and creep data points were removed
- Tests conducted at temperatures exceeding 50 °C were removed
- Data points at times $t > 1000$ days were removed
- Corrections were made to data for tests conducted by Li and Yao [7] (CT_id 1173 and 1174). Slag and superplasticiser quantities changed to 180 kg/m³ and 9.6 kg/m³, respectively.
- Results by Vincent et al. [29] (CT_id 1374 and 1375) removed. This was due to exceptionally large creep strains when compared to other experiments, which also appeared to differ from the source article.
- Results by Collins et al. [30] (CT_id 1140–1154) removed. This was due to exceptionally large creep strains when compared to other experiments.

Due to the small size of available data for slag concrete, the experimental results of Shariq et al. [5] and Qin et al. [6] were incorporated into the dataset. Shariq et al. [5] experimentally investigated creep and shrinkage in GBFS concrete with cement substitution ranging from 0–60%. In total, 12 creep curves were reported, for which 3 were controls (0% GBFS). Manual extraction of these data resulted in the generation of 111 data points. Qin et al. [6] analysed drying shrinkage and creep deformations in concrete containing combinations of fly ash, slag and silica fume. SCM replacement ratios of 50% and 75% were considered. Three control curves (35 data points), six curves for concrete containing fly ash and slag (69 data points) and two curves for concrete with fly ash only (28 data points) were added to the dataset. After making the aforementioned modifications, 465 data points were available for creep in concrete with slag or slag + fly ash as SCMs (including related control tests), and 991 data points for fly ash concrete. Thus, the database consisted of 1456 data points in total.

Across key properties relating to concrete creep modelling, there were missing data in the compiled dataset for relative humidity during creep testing (RH), relative humidity during preconditioning ($H0$) and 28-day elastic modulus $E_{c,28}$. Missing data for RH and $H0$ were estimated using the mode imputation method. The mode for RH was 40% for the slag-based concrete data and 60% for fly ash concrete. The data were separated for mode determination to capture potential differences in local conditions between tests on slag and fly ash concrete. Note that only 31 and 129 RH data points were missing for slag and fly ash concrete, respectively. In total, 45 data points were missing for $H0$ data relating to slag concrete, and no data were missing for fly ash concrete. The mode for $H0$ was 98%. Median imputation was not applicable due to the partial categorical nature of the RH and $H0$ data: 99% means sealed specimen, 100% means storage in water, 101% represents steam and 85% moist conditions. The 28-day elastic modulus was estimated using the following ACI-209 equation;

$$E_{c,28} = 4733\sqrt{f_c} \quad (1)$$

The ACI-209 equation was selected due to its common adoption and dependence on compressive strength only, which, for the case of this study, was convenient due to data availability. The creep compliance reported in the NU-ITI database was split into two groups: measurement of elastic strain at the beginning of the creep test and measurement of creep strain only. The majority of the slag concrete creep data did not record elastic strain. Conversely, most of the data for fly ash concrete did measure creep strain. During preliminary analyses, it was observed that model development using ML was more successful when elastic strain was considered. Hence, $E_{c,28}$ was utilised to estimate elastic strain for relevant data points, ensuring uniformity across the dataset. This approach introduces two potential sources of error: inaccurate estimation of $E_{c,28}$ and use of 28-day modulus for elastic strain calculation despite loading age. These errors are, however, assumed to be

small. In total, only 142 data points (9.39% of the dataset) did not measure elastic strain, report 28-day elastic modulus and were not loaded at 28 days.

The adopted input variables and their ranges, mean and standard deviation are listed in Table 1. Input variables selection considered the significance of creep, limitations of the available data and parameters required by codified creep equations. Histograms for each input parameter are shown in Figures 1 and 2. It can be seen that the number of data points for time decreases exponentially as time increases. Most of the data were measured at times $t \leq 150$ days, with limited data available at $t > 500$ days. The majority of the data featured a loading age of 28 days (899 data points) or 7 days (405 data points). Only a few experiments adopted different loading ages. V/S ratios fell mostly into three ranges: 12.0, 20.0–22.22 and 30.0–33.5 mm. Ratios of 50.8 and 60 mm are also present in the data, albeit to a lesser extent. The most common relative humidity ranges were 40–45%, followed by 60–65% and 50–55%. A smaller number of tests were conducted on sealed specimens (99%) and on specimens stored in water (100%). The 28-day compressive strengths ranged from 15.5–188.9 MPa, with the 80–90 MPa band being the most frequently occurring in the dataset by a large margin. Consequently, water-to-binder ratios were commonly between 20–30% and elastic moduli between 40,000–45,000 MPa. There is a relatively uniform distribution of other strength grades. Slag substitution ratios varied from 0–60% of total binder content, see Figure 2c. Note that 0% is omitted for clarity due to the high number of data points without slag content (1149). Higher replacement ratios (40–42% and 50–60%) are better represented in the data than lower ratios (20% and 30–35%). The fly ash-to-total binder ratios are shown in Figure 2d and ranged from 0–50%. Unlike slag, lower substitution ratios were more common for fly ash, with 10–20% appearing the most frequently followed by 30–32%. Only a small section of the dataset features ratios above 32%. In total, 464 data points featured silica fume content (SF), of which more than half (261 points) equated to 29.37 kg/m³, with the remaining points evenly distributed from 34–89 kg/m³. The impact of adopting relative humidity H_0 during preconditioning as an input variable on model accuracy is explored in the Sensitivity Analysis presented in Section 5.

Table 1. Statistical properties of adopted input variables in dataset.

Input Variable	Notation	Range	Mean	Standard Deviation	Unit
Time since loading	t	0–977	109.22	138.10	days
Loading age	t_0	7–730	40.22	87.94	days
Relative humidity during test	RH	40–100	52.54	15.41	%
Compressive strength at 28 days	$f_{c,28}$	15.5–118.9	69.44	23.53	MPa
Elastic modulus at 28 days	$E_{c,28}$	18,651.9–50,800.0	39155	6863	MPa
Cement content	C	124–620	398.41	116.90	kg/m ³
Water content	W	85.00–228.24	150.81	40.48	kg/m ³
Total aggregate content	A	1046.4–1874.0	1636.53	175.98	kg/m ³
Fly ash content	FA	0–517	78.50	68.73	kg/m ³
Slag content	S	0–355	42.76	90.86	kg/m ³
Volume-to-Surface ratio	V/S	12–60	25.13	13.31	mm
Silica fume content	SF	0–89	12.86	21.14	kg/m ³
Relative humidity during preconditioning	H_0	65–100	95.25	10.25	%

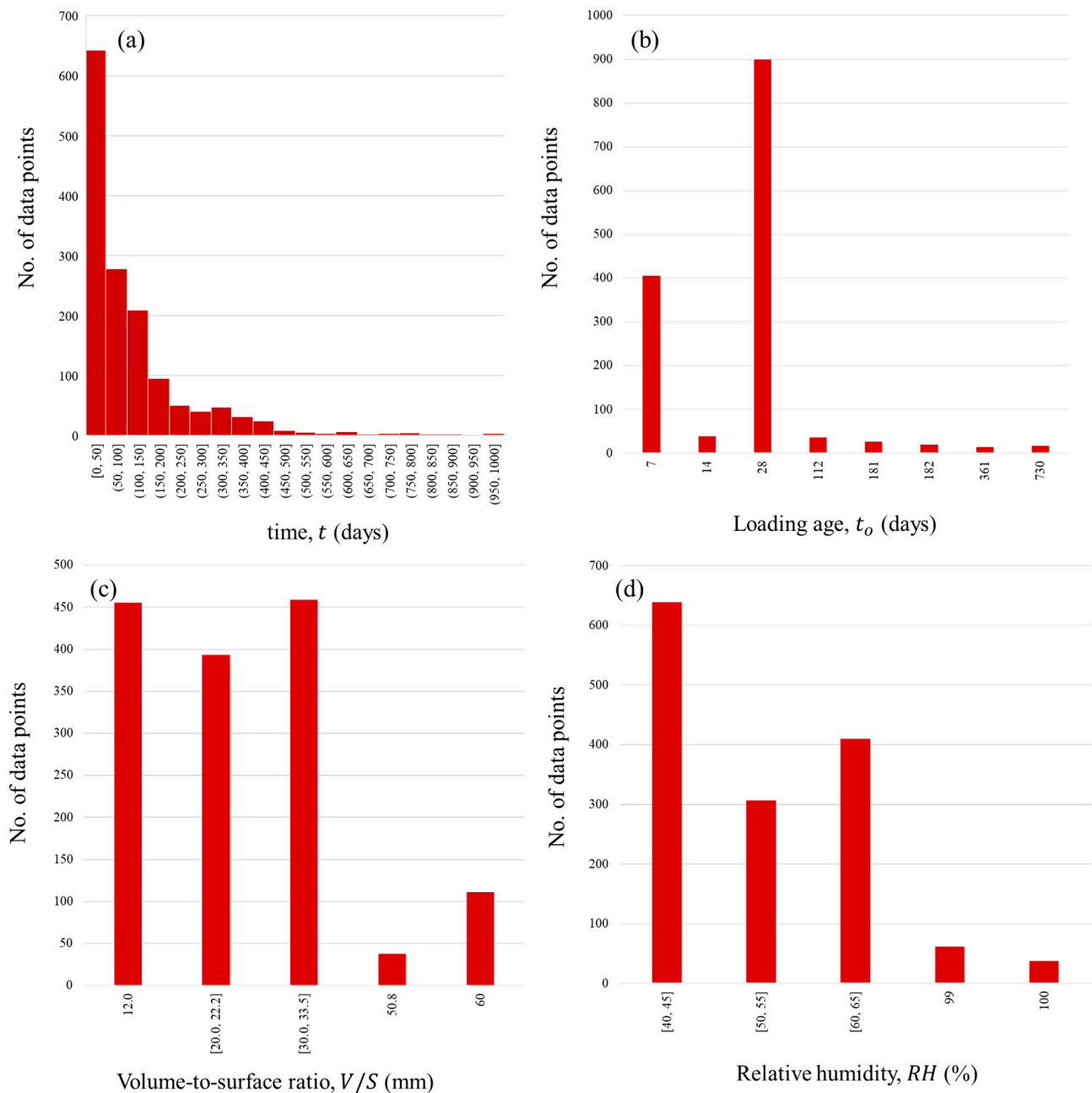


Figure 1. Histograms of input data relating to testing parameters: (a) time, (b) loading age, (c) volume-to-surface ratio and (d) relative humidity.

Table 2 compares the input variables adopted herein with those considered in studies in the literature on training ML models to predict the creep in concrete containing SCMs [24,25,27] and the ACI and B3 creep equations. It can be seen that mix proportions, compressive strength, t_0 , RH , and V/S are commonly considered. Elastic modulus of concrete, at 28-days or at t_0 , has not been adopted as an input variable by other studies, though it is considered by design equations. Additionally, the relative humidity of preconditioning is only considered herein. Various factors considered by design equations, including the curing method, cement type, air content and unit weight of concrete, could not be adopted here, and likely in other studies, due to lack of available information.

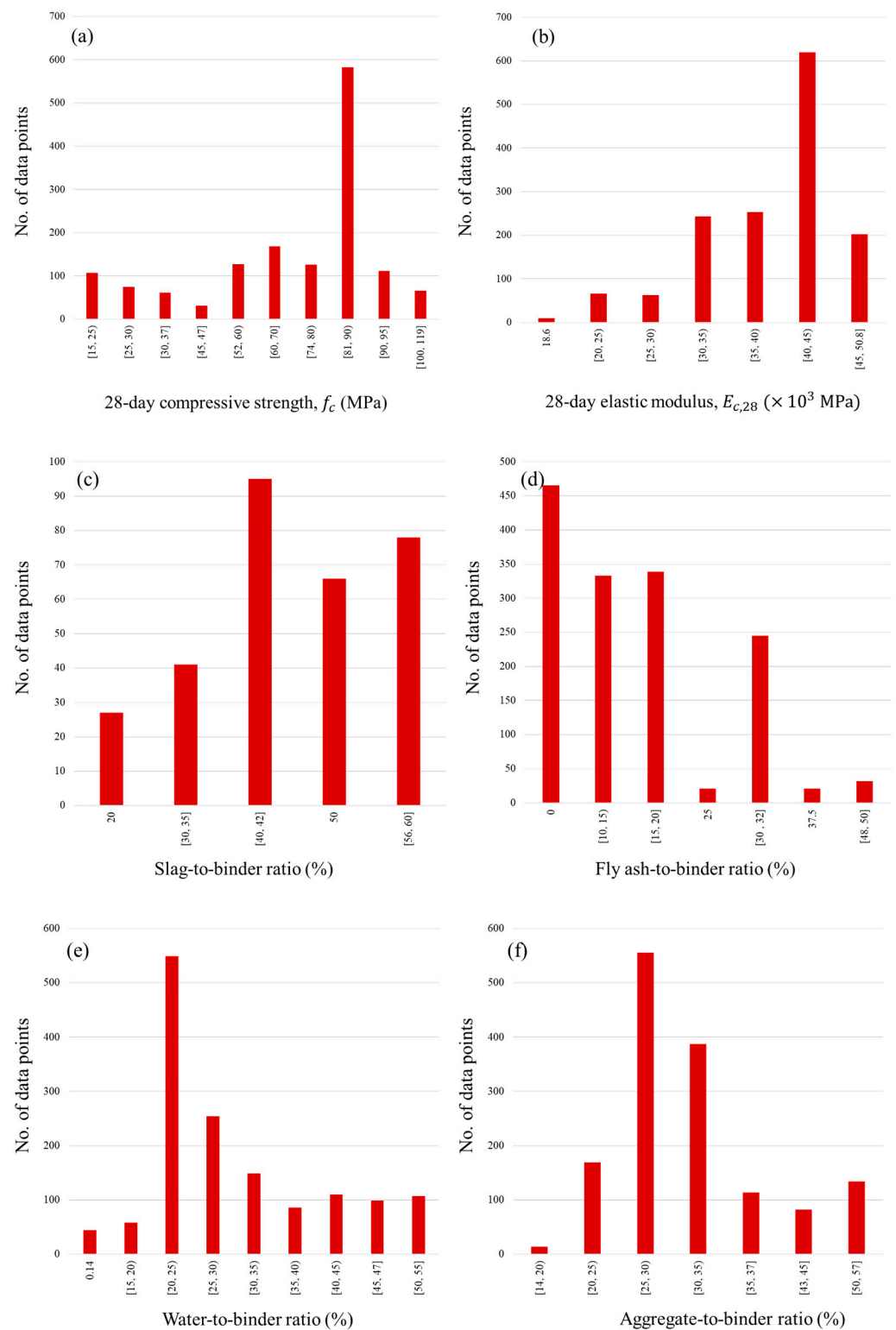


Figure 2. Histograms of input variables relating to concrete properties: (a) 28-day compressive strength, (b) 28-day elastic modulus, (c) slag-to-binder ratio, (d) fly ash-to-binder ratio, (e) water-to-binder ratio and (f) aggregate-to-binder ratio.

Table 2. Comparison of input variables adopted herein with relevant literature and codified creep equations.

Input Variable	This Paper	[25]	[24]	[27]	ACI-209	B3
Time since loading	X	X	X	X	-	-
Loading age	X	X	-	X	X	X
Age at drying commencement	-	X	-	-	X	X
RH	X	X	-	X	X	X
$f_{c,28}$	X	X	-	-	-	X
$E_{c,28}$	X	-	-	-	-	X
Compressive strength at t_0	-	-	-	-	X	-
Elastic modulus at t_0	-	-	-	-	X	-
Applied stress	-	-	-	X	-	-
Stress-to-strength ratio	-	-	-	X	-	-
C	X	X	X	X	X	X
W (or W/C)	X	X	-	X	-	X
A (or A/C)	X	X	-	X	-	X
FA	X	X	-	X	-	-
S	X	X	X	X	-	-
Volume-to-Surface ratio	X	X	-	X	X	X
SF	X	X	-	X	-	-
Water-to-binder ratio	-	-	X	-	-	-
Fine aggregate content	-	-	X	-	X	-
Course aggregate content	-	-	X	-	-	-
Slump	-	-	X	-	X	-
Compaction factor	-	-	X	-	-	-
Air content	-	-	-	-	X	-
Cement type	-	-	-	-	X	X
Curing method	-	-	-	-	X	X
Relative humidity of preconditioning	X	-	-	-	-	-
Temperature	-	-	-	X	-	-
Unit weight of concrete	-	-	-	-	X	-
Ultimate shrinkage strain	-	-	-	-	-	X
Pore relative humidity	-	-	-	-	-	X
Shrinkage half time	-	-	-	-	-	X

3. AI Regression Models

Supervised ML is a subcategory of ML and AI. Typical supervised learning includes learning (or training) and application procedures with the framework shown in Figure 3. It is a standard method for predicting concrete properties, including those relating to strength and durability. In total, 12 factors that are relevant to the concrete and creep test features are used as supervised learning inputs (Table 2). Creep deformation (ϵ_{cr}) is the output of the supervised machine learning algorithms.

3.1. Artificial Neural Network (ANN)

An ANN is a standard supervised ML method composed of artificial neurons (nodes) that simulate our brains' information transformation. There are various ANN algorithms, and the Back-propagation Neural Network (BPNN) algorithm is one of the most widely used to optimise the neural network training through back-propagation of the error gradient. BPNN's structure is shown in Figure 4. Hornik et al. [31] proved that a single hidden layer network is good enough to develop reliable predictions for BPNN. Thus, this paper also uses a single hidden layer.

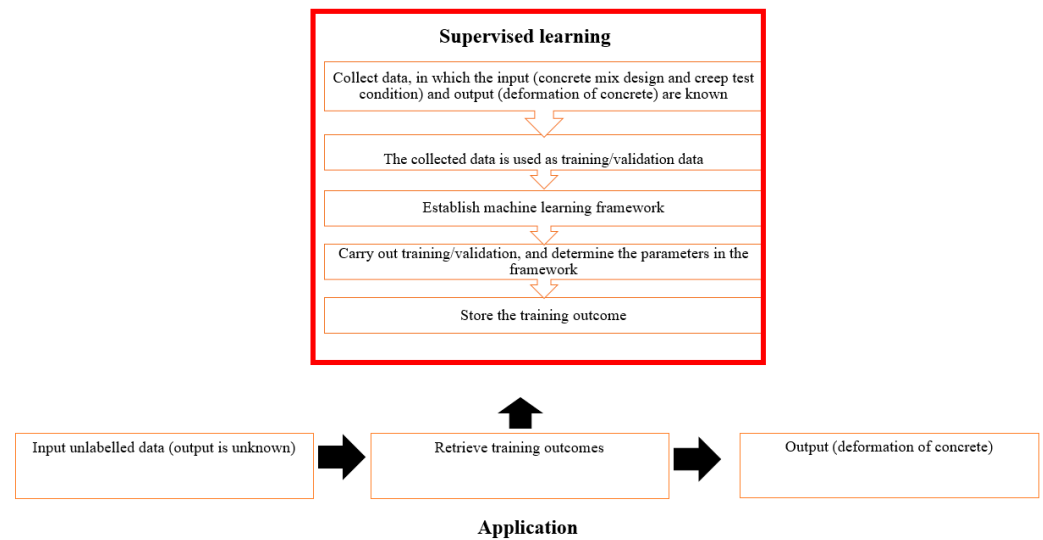


Figure 3. Framework of supervised learning.

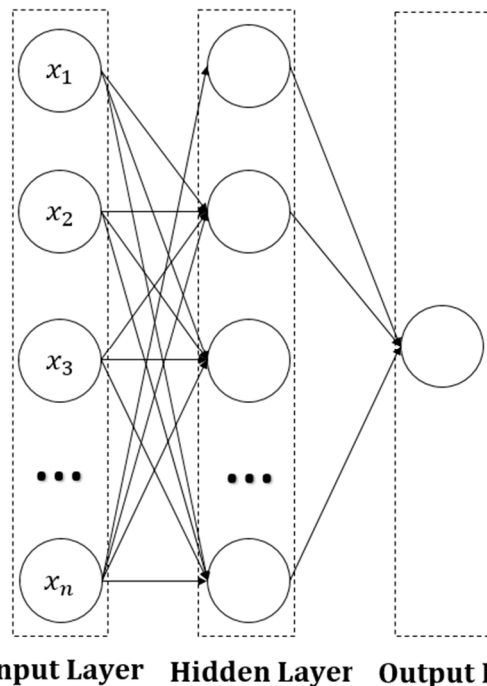


Figure 4. Architecture of artificial neural network [32].

BPNN contains many neurons in each layer, and the connection between neurons is assigned weight, representing the connection's importance. BPNN is trained through feed-forward stages and back-propagation stages. In the feed-forward stages, input layer neurons pass the input into the hidden layer. Neurons in the hidden layer execute a weighted summation of all inputs and compute activation value according to an activation function. Each neuron's activation value in hidden layers is then used as the input for neurons in the output layer. The output layer also uses activation functions to determine the final results. Activation function is shown in Equation (2) below:

$$f_j = \frac{1}{1 + \exp(-\sum w_{ji}o_i + t)} \quad (2)$$

where w_{ji} is the connection weight of neurons i to neuron j . Neuron i is located in the layer before neuron j . o_i and i are output from neuron i and the bias value. In BPNN, the network error is passed backwards from the output layer to the input layer, and weights are adjusted to reduce the network error. This paper uses the Levenberg–Marquardt (LM) algorithm to adjust the neurons’ weights and improve the BPNN’s performance. Details of LM are explained in Smith et al. [33]. Collected data are categorised into training and testing subsets (70% and 30%, respectively) to ensure the training quality of ANN.

3.2. Decision Tree Regression (DTR)

Using a decision tree, continuous data (input) are repeatedly partitioned into nodes to determine the output values. The partition follows binary decision rules with the structure shown in Figure 5. Nodes with and without outgoing edges are called interior nodes and leaves accordingly. The binary decision rules are explained below:

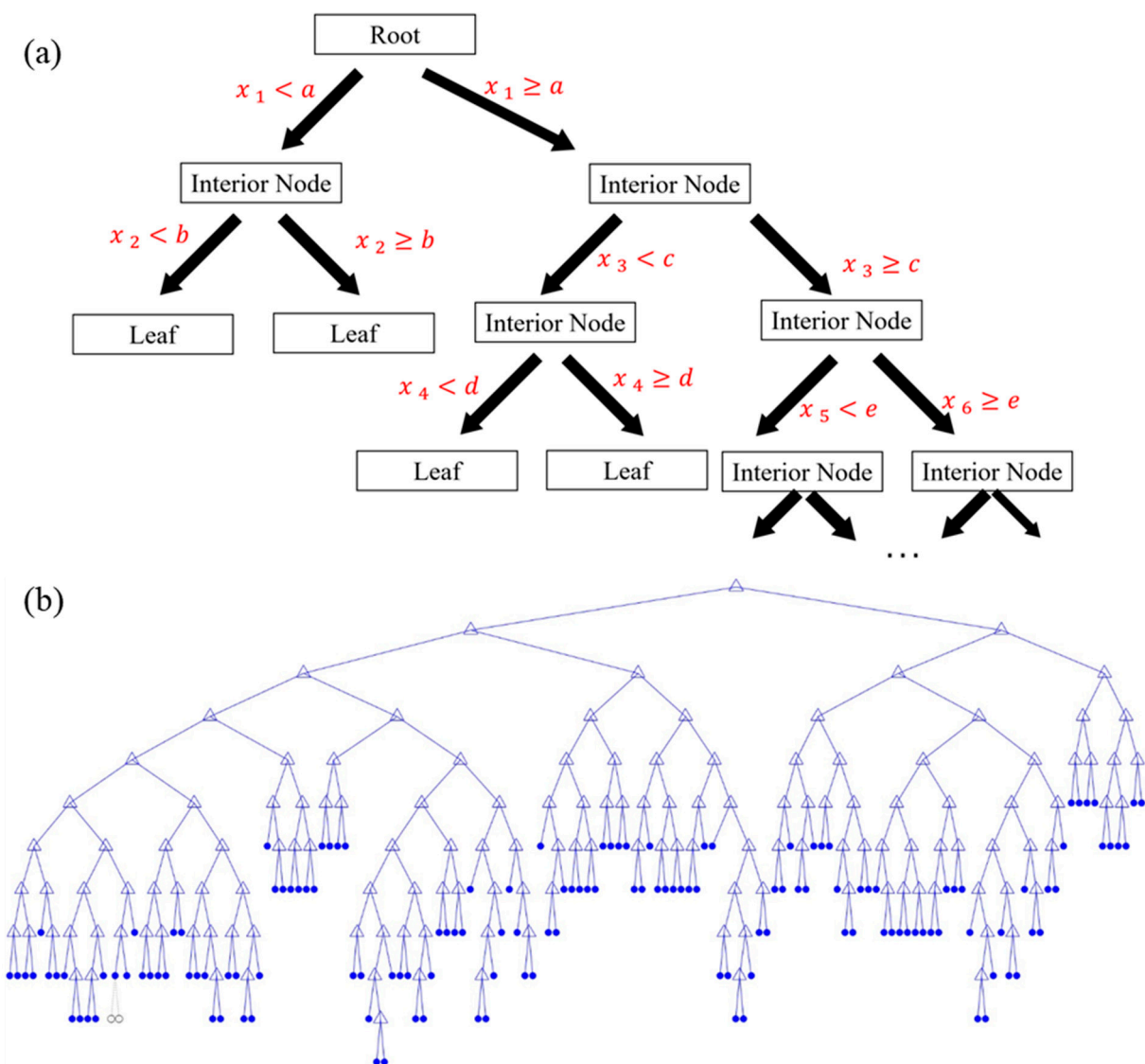


Figure 5. Structure of DTR: General Structure (a) and structure developed herein for creep prediction (b).

Each data cluster (x_j) arriving in a node is split into two sub-clusters, following Equation (3).

$$R_1(j, s) = \{ x | x_j < s \} \text{ and } R_2(j, s) = \{ x | x_j \geq s \} \tag{3}$$

where X_j is referring to $X_{1,2,3 \dots}$ and s is referring to $a, b, c \dots$ in Figure 5. The aim of binary decision rules is to find values of x_j and s that minimise Equation (4).

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \quad (4)$$

where \hat{y}_{R_1} is the average response of training observations in $R_1(j, s)$, and \hat{y}_{R_2} is the average response of training observations in $R_2(j, s)$. In this study, data have been divided into training and testing data (70% and 30%, respectively). The study also uses the `fitree` function in MATLAB (version R2021b) for DTR construction. De Azevedo et al. [34] show the details of DTR.

3.3. Random Forest Regression (RFR)

RFR conglomerates many DTR models' outcomes and determines the average value from each DTR prediction. Therefore, the prediction accuracy is higher than DTR's in Section 3.2. Figure 6 shows the configuration of RFR. The Least Squares Boost (LSB) is the algorithm that combines DTR and achieves an acceptable accuracy for DTR training. Alatefi and Almeshal [35] explained the details of LSB, which is the algorithm used in this paper. Again, data have been divided into training and testing data (70% and 30%, respectively).

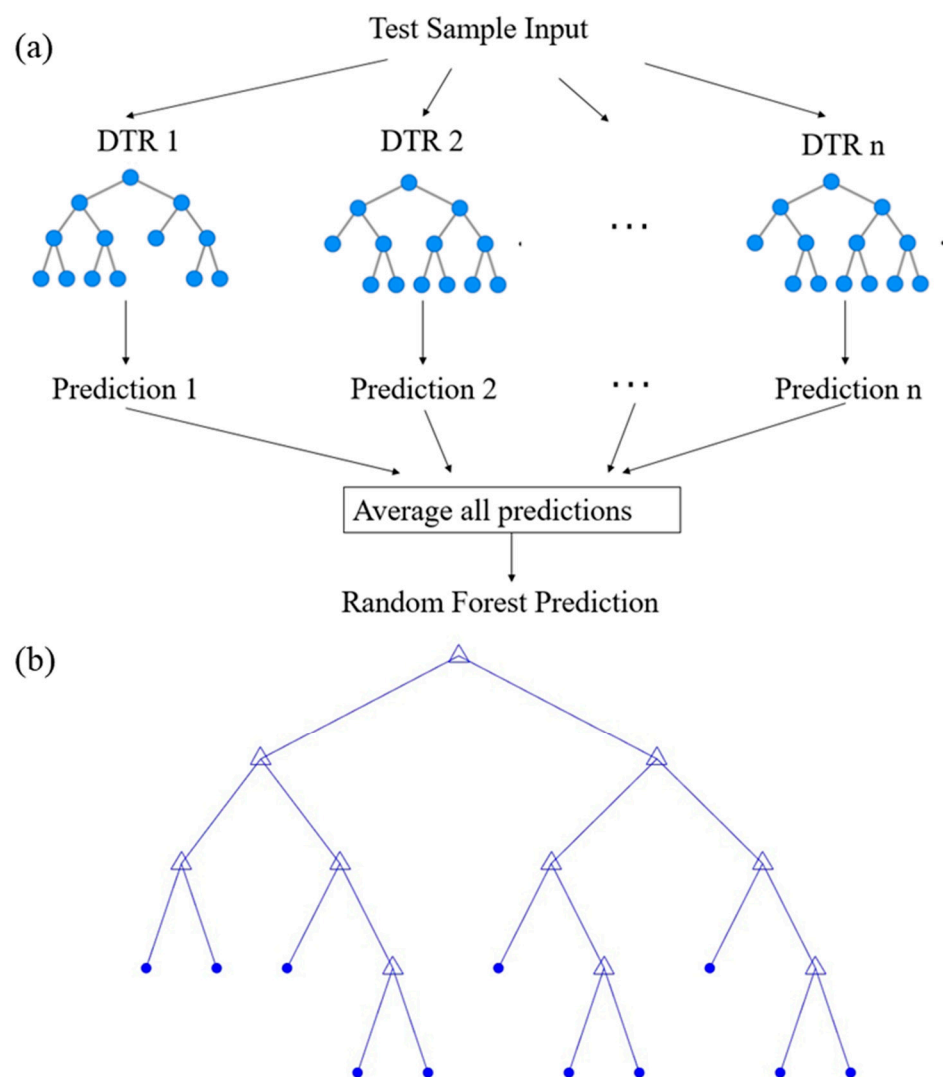


Figure 6. Structure of random forest prediction: General Structure (a) and structure developed herein for creep prediction (b).

3.4. Gaussian Process Regression (GPR)

Gaussian process regression (GPR) is a non-linear, non-parametric machine learning technique based on Bayesian probability theory [36]. It provides a flexible and probabilistic approach for regression tasks, allowing for the estimation of complex relationships between input variables and output values. One of the key advantages of GPR is its ability to handle small datasets, provide robust estimates even in the presence of noise and provide a reliable response to input data.

Consider a dataset $D = \{(x_i; y_i) \mid i = 1 \dots n\}$ of observations where x_i is the input vector, and y_i is a scalar output. The core assumption of GPR is that the output y is determined as;

$$y = f(x) + \epsilon \quad (5)$$

where ϵ denotes noise (random variable) that is governed by an identically distributed Gaussian distribution, $\epsilon \sim N(0, \sigma^2)$. The observations in the data are considered as generated from a Gaussian process, which is a collection of random variables such that any finite subset follows a multivariate Gaussian distribution featuring a mean of zeros. The kernel function $k(x, x')$, also known as the covariance or similarity function, plays an important role and governs the relationship between observations. Hence, the selection of kernel function is critical and different kernel functions can impact model performance. As it is assumed that the training and testing data are multivariate Gaussian distributed, the following distribution can be expressed;

$$\begin{bmatrix} y \\ y_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix}\right) \quad (6)$$

where K is the covariant matrix defined as;

$$K = \begin{bmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & \dots & k(x_2, x_n) \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{bmatrix} \quad (7)$$

K_* is the $n \times 1$ covariant matrix of the test point x' and all input data points x and K_{**} is the self-covariant matrix of x' .

Once the kernel function and hyperparameters are selected, the GPR model is trained by iteratively updating its prior distribution over functions using Bayesian inference and the observed training data. Post-training, Bayesian inference is utilised to estimate the predicted output value y_* for a test point x' , along with its associated uncertainty, by conditioning the posterior distribution on the observed training data and the input of the test point. This iterative learning process enables the GPR model to generate predictions and quantify their uncertainties based on the learned information from the training data.

4. Model Performance and Results

4.1. Artificial Neural Network

The training and testing are carried out in a MATLAB environment through the Neural Net Fitting Designer toolbox [37]. A sigmoidal tangent function with a linear activation function was considered for hidden nodes in the model, and a linear activation function for the output layer in the model setup, see Equation (1). The ANN model was trained with one hidden layer and one output layer through the LM algorithm. The number of hidden neurons was determined as $2(n + 1) = 26$, as it has 12 inputs. Figure 7 shows the ANN's construction in MATLAB.

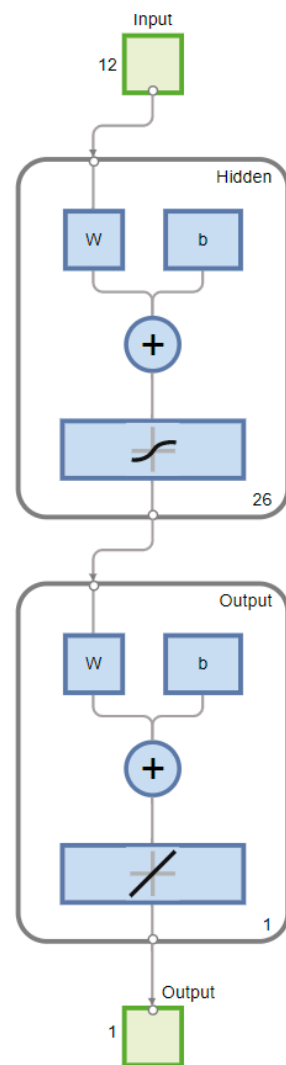


Figure 7. ANN model establishment in MATLAB.

Table 3 shows the R-squared (R^2), root-mean-square deviation (RMSE), mean absolute error (MAE) and Mental State Examination (MSE) achieved for the training and testing subsets. Figure 8a,b shows the predicted and true responses for the ANN model. The models achieved $R^2 = 0.99$ and 0.91 and RMSE of 2.032 and 3.028 for training and testing. The high R^2 and RMSE values indicate that ANN can give a reliable prediction for concrete creep behaviour.

Table 3. Achieved prediction accuracies of trained ANN, DTR, RFR and GPR models.

Model	Set	RMSE	R^2	MSE	MAE
ANN	Training	2.032	0.989	4.128	1.333
ANN	Testing	3.028	0.908	9.172	2.194
DTR	Training	3.791	0.962	14.374	2.339
DTR	Testing	5.679	0.902	32.254	3.557
RFR	Training	0.722	0.999	0.521	0.517
RFR	Testing	2.218	0.985	4.918	1.414
GPR: Squared Exponential	Validation	2.946	0.980	8.676	1.990
GPR: Matern 5/2	Validation	2.419	0.990	5.853	1.592
GPR: Exponential	Validation	2.428	0.990	5.897	1.532
GPR: Rational Quadratic	Validation	2.295	0.990	5.265	1.516

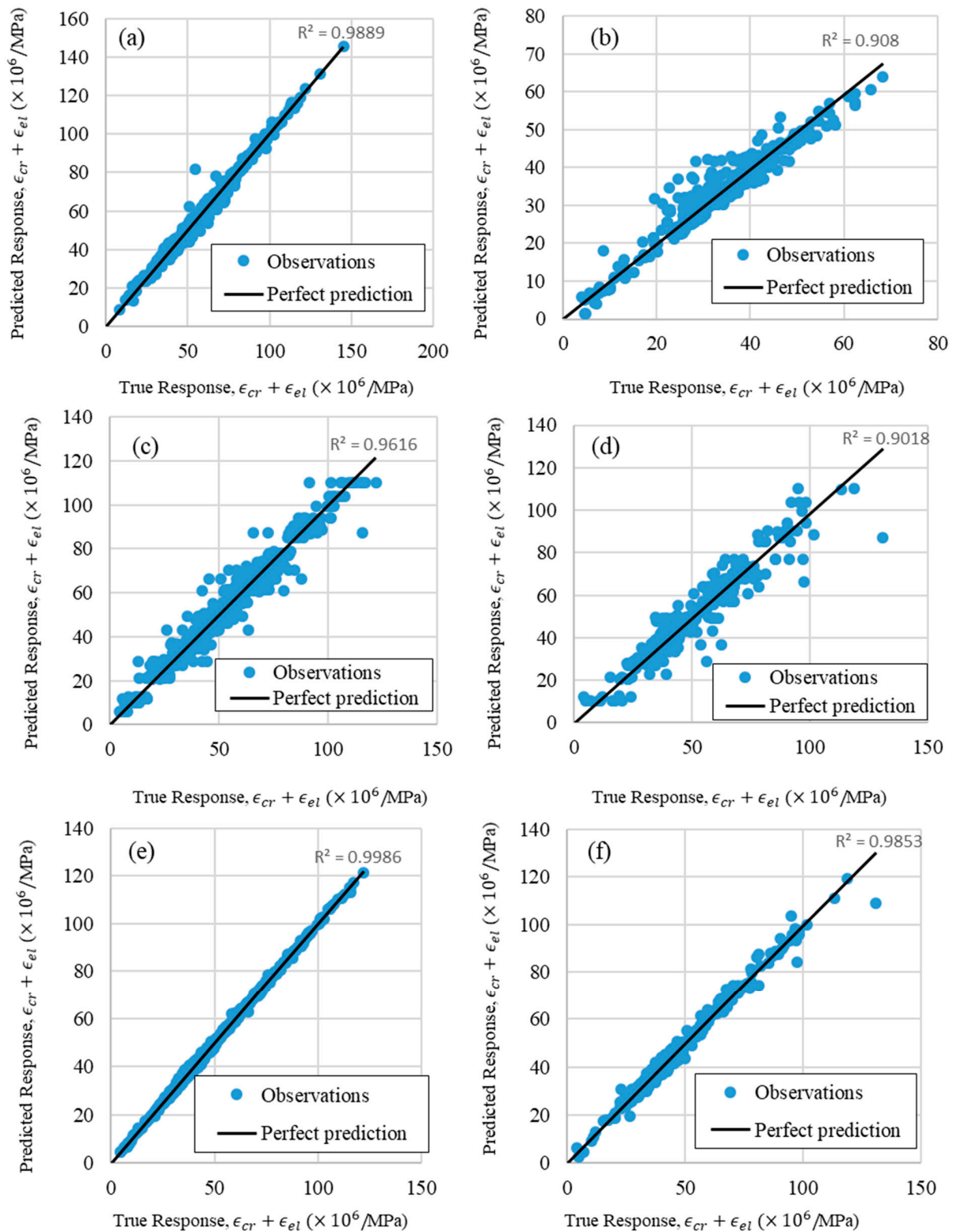


Figure 8. Predicted versus true responses for ANN model with training data (a) and testing data (b), DTR model with training data (c) and testing data (d), and for RFR model with training data (e) and testing data (f).

4.2. Decision Tree Regression

DTR was carried out using “fitrtree” functions in MATLAB with default settings. Recursive binary splitting was adopted as the training algorithm with 291 nodes. The mini-

imum leaf size was determined as two through hyperparameter optimisation. Figure 8c,d plots predicted responses against true responses for training and testing data. Table 3 also shows the RMSE, R-squared, MSE, and MAE based on predicted and true responses, which shows the model achieved $R^2 = 0.96$ and 0.90 and RMSE of 3.791 and 5.67 for training and testing. DTR's prediction is less accurate than ANN's based on the RMSE and R^2 values. The outcome confirms Dietterich [38]'s finding that DTR can be unsteady and the insignificant input variable variation can significantly affect tree structure.

4.3. Random Forest Regression

MATLAB carried out RFR through "fitrensemble" functions. Least-squares boosting was adopted as the training algorithm with 1001 observations. Hyperparameters optimisation indicates that the optimum number of learning cycles (trees) is 482, the learning rate is 0.19021, and the minimum leaf size is 8. Figure 8e,f plots predicted responses against true responses for training and testing data. Based on Table 3, RFR's prediction is more reliable than DTR as it considers the performance of many DTR algorithms together. It also achieves a higher R-squared and a lower RMSE than ANN. This is because RFR is less prone to overfitting compared with ANN [39].

4.4. Gaussian Process Regression

The creep compliance prediction model was developed in MATLAB software using the regression learner tool. The GPR algorithm with various kernel functions was adopted for model training. The length scale and noise standard deviation hyperparameters were taken as 589.9 and 13.77, respectively. A constant basis function and the isotropic kernel were adopted. The input parameters were scaled (standardised), and numeric parameters were locally optimised using the MATLAB advanced options. The dataset was split into training and validation subsets of sizes 70% and 30%, respectively. Table 3 compares the accuracies achieved by each model. High accuracies were observed for all adopted kernel functions with the Matern 5/2, rational quadratic (RQ) and exponential models, each achieving $R^2 = 0.99$ and providing close results for other indicators. The Matern 5/2-GPR model's predicted response versus the true response for the validation data is shown in Figure 9.

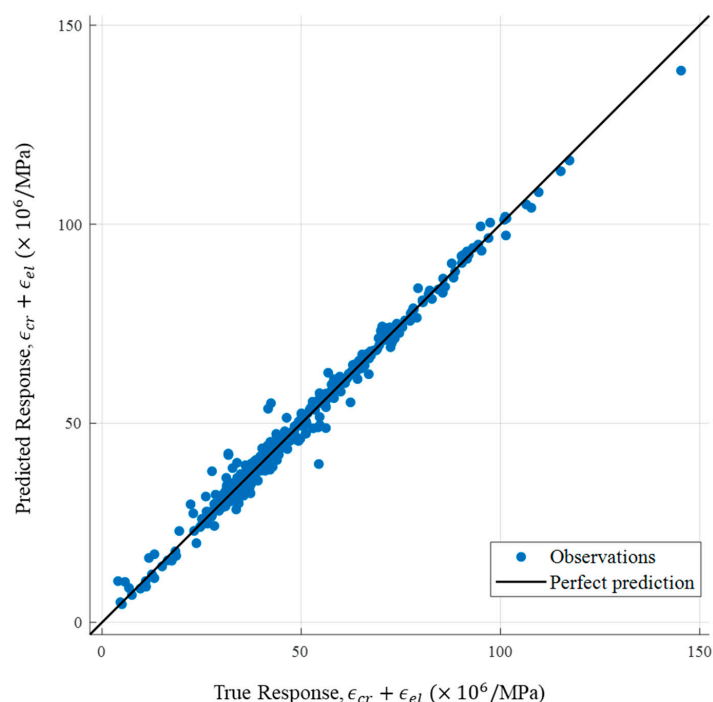


Figure 9. Predicted response versus true response for Matern 5/2-GPR model (validation data).

5. Sensitivity Analysis

The random selection of validation can influence the performance of the ML models. To examine this, the RQ and Matern 5/2 GPR models were trained an additional five times, and the changes in the accuracy indicators were examined. The results of this analysis are presented in Table 4, revealing that the selected GPR models demonstrated consistent performance with small reductions in accuracy shown across multiple training repetitions. The sensitivity of the GPR models to input variables and validation methodology is also analysed. Training was conducted with consideration of relative humidity during preconditioning. The performance of the GPR models when considering this variable is provided in Table 5. The results provided in Tables 3–5 show that the accuracy indicators do not noticeably change when adopting H_0 as an input variable and that these small changes are within variations observed due to training repetitions. Changing the validation methodology from holdout (30%) to cross-validation (5-fold) caused a small reduction in model performance based on the accuracy indicators. However, the accuracy of the trained models is still high. The additional input variable had a negligible impact on the accuracy of the GPR models when adopting cross-validation.

Table 4. Sensitivity of Matern 5/2 and RQ GPR model performance to training repetitions.

Kernel Function	Repeat No.	RMSE	R^2	MSE	MAE
Matern 5/2	1	2.505	0.98	6.275	1.694
RQ	1	2.526	0.98	6.383	1.692
Matern 5/2	2	2.688	0.98	7.227	1.694
RQ	2	2.980	0.98	8.878	1.769
Matern 5/2	3	3.094	0.98	9.572	1.801
RQ	3	2.892	0.98	8.365	1.747
Matern 5/2	4	2.430	0.98	5.907	1.581
RQ	4	2.437	0.98	5.939	1.577
Matern 5/2	5	3.100	0.97	9.610	1.801
RQ	5	2.951	0.98	8.709	1.746

Table 5. Sensitivity of GPR model accuracy to input variables and validation methodology.

Changed Input Variables	Validation	Algorithm and Kernel Function	RMSE	R^2	MSE	MAE
+ H_0	Holdout—30%	GPR: Squared Exponential	2.852	0.98	8.134	1.993
		GPR: Matern 5/2	2.498	0.98	6.238	1.685
		GPR: Exponential	2.476	0.98	6.132	1.542
		GPR: RQ	2.490	0.98	6.198	1.669
-	Cross—5-fold	GPR: Squared Exponential	3.974	0.96	15.796	2.100
		GPR: Matern 5/2	3.723	0.96	13.864	1.759
		GPR: Exponential	2.580	0.98	6.656	1.450
		GPR: RQ	3.401	0.97	11.566	1.729
+ H_0	Cross—5-fold	GPR: Squared Exponential	4.074	0.96	16.596	2.131
		GPR: Matern 5/2	3.530	0.97	12.460	1.776
		GPR: Exponential	2.574	0.98	6.624	1.441
		GPR: RQ	2.979	0.98	8.872	1.677

The effect of dataset size is an important parameter that requires further validation. This was examined herein by varying the size of the dataset used for model training and assessing the effect on prediction accuracy. The R^2 indicator is plotted against training data size, which ranges from 436 (30% of the total dataset) to 1019 data points (70% of the total dataset) for the RQ-GPR, ANN, DTR and RFR models, see Figure 10a. Data points were removed at random to reduce dataset size. It can be seen that R^2 converges for the DTR, RFR and GPR models when the training data size reaches approximately 800 data points. The ANN model converges at approximately 860 data points. In this paper, the number of

training data is 1019. Therefore, the training of ANN and DTR can achieve a high accuracy for concrete creep prediction. The R^2 for RFR is always above 0.98 for the data size from 436 to 1019. This indicates that RFR performs better when there is a limited sample size.

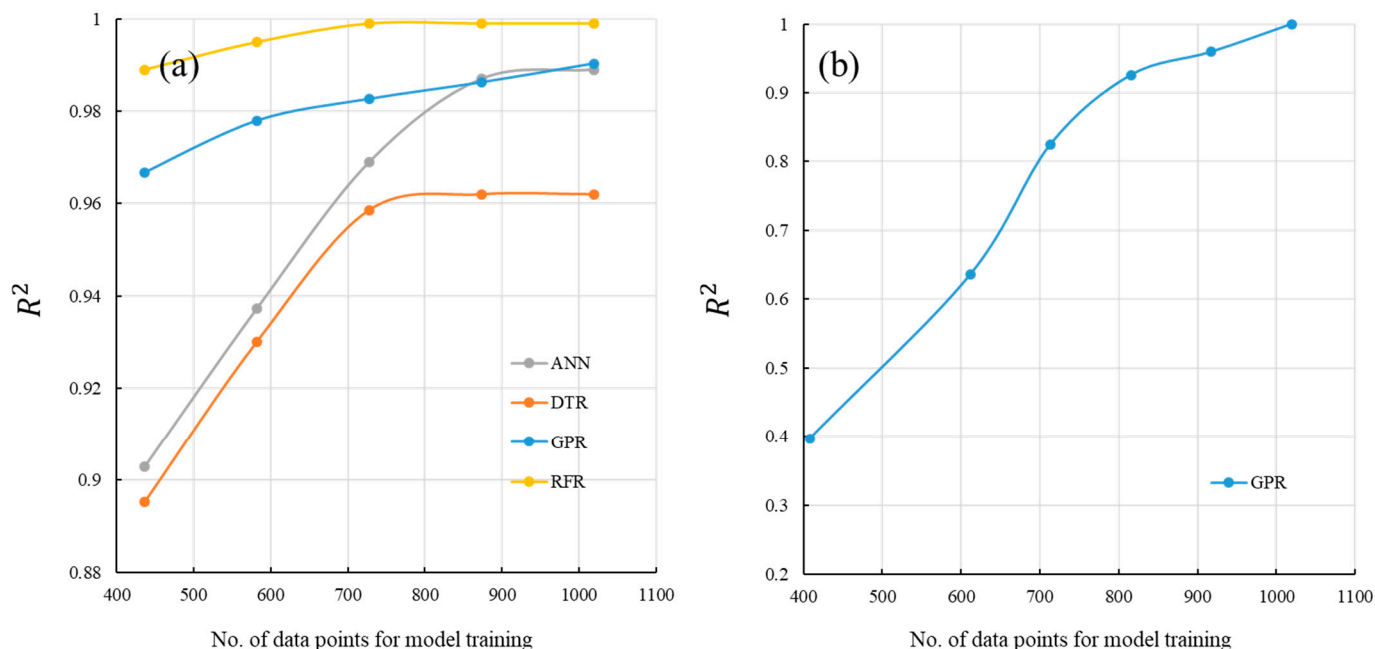


Figure 10. Effect on training data size on prediction accuracy for all ML models with random individual data point selection (a) and for the RQ-GPR model with random creep curve selection (b).

The random selection of data points may, however, not cause a significant reduction in accuracies of model prediction due to the input variable of time, which causes every creep curve to feature multiple data points. In order to investigate this, a second analysis was conducted where entire creep curves were removed at random as opposed to removing individual data points. The results of this analysis are presented in Figure 10b for the RQ-GPR model. It can be seen that the effect of dataset size now has a much larger impact on model accuracy (R^2), though the model accuracy still plateaus at 80% of the dataset size, which validates the size of the dataset used for model development.

6. Shapley Additive Explanation (SHAP) Analysis

The significance of each input parameter on creep compliance prediction is assessed by conducting a SHAP analysis [40]. This involves computing and analysing Shapley values for a set of query points across a dataset. Shapley values, a concept originally from cooperative game theory, assist with ML model interpretability and explain predictions by quantifying the contribution of each feature (input variable) to the prediction [41]. Shapley values consider all possible combinations of inputs/features as well as interactions and dependencies between them. For a given prediction, the Shapley values (contributions) of each feature sum to the value of the prediction, and the Shapley value for a specific feature is the difference between the actual prediction and the mean prediction.

In order to ensure a high representation of the data, the query points were selected herein using 3–4 data points from every creep curve available in the dataset that included the initial time and final time of the creep test. The Shapley values were computed using the developed RQ-GPR model and the “explainer” function in MATLAB. Figure 11a,b depicts the averaged Shapley values (absolute) and the Shapley value distribution, respectively, for slag concretes. In order of decreasing significance, the input variables are found to be t , V/S , $E_{c,28}$, $f_{c,28}$, S , W , C , H_0 , FA , A , t_0 , RH and SF . Aside from time, V/S was the most influential variable and was mostly negatively correlated with creep compliance. This indicates that an increasing V/S results in decreasing creep strain. The averaged

Shapley values for $E_{c,28}$ and $f_{c,28}$ were similar with both parameters featuring a strong influence on creep compliance prediction. The remaining input variables were noticeably less influential, particularly $H0$, FA , A , t_0 , RH and SF . The lower importance of t_0 and RH may be attributed to their low variability in the dataset. Most tests on the slag concrete commenced at 28 days, and the RH varied only from 45–65%.

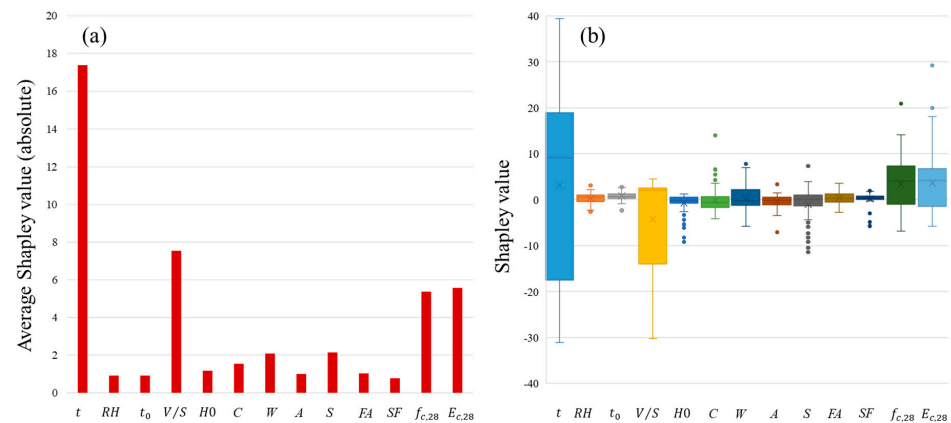


Figure 11. Shapley value absolute averages (a) and a Box and Whisker plot (b) showing Shapley value distribution for slag concretes.

Figure 12a depicts the averaged Shapley values (absolute), and Figure 12b displays the Shapley value distribution for fly ash-based concretes. For fly ash concretes, the input variables by order of reducing influence are t , $E_{c,28}$, t_0 , $f_{c,28}$, V/S , RH , W , FA , SF , C , FA , A , S and $H0$. As with slag concrete, $E_{c,28}$, V/S and $f_{c,28}$ were highly influential. However, the absolute Shapley values for t_0 and RH are substantially higher in fly ash concrete than in slag concrete. This is to be expected and is now reflected in the Shapley values due to the higher variability of these parameters in the dataset. Across both slag and fly ash datasets, the Shapley values for $H0$, C , A , FA and SF are small. This suggests that these parameters are not as important in creep compliance prediction as the other parameters. Conversely, water content W shows high significance in both datasets. Slag content S was important for predicting creep in slag concrete, whereas fly ash content showed a low impact on creep for fly ash concrete. Liu et al. [27] also observed the low importance of SCM content when compared to other variables on creep prediction in concrete through Shapley value additive analysis. However, the significance of slag content appears to be higher in the analysis presented herein. Interestingly, there are differences in the significance rankings of parameters t , A and V/S . Aggregate content was reported to have a major influence on creep compliance prediction, with the volume-to-surface ratio showing low importance in [27]. The opposite effect of these two parameters is observed herein. Furthermore, time since loading was only the 6th most influential parameter in [27], whereas in this analysis, it is determined to be the most significant. These differences may be due to the adoption of different input variables and datasets adopted for model training. For example, there is a higher maximum quantity of slag content in the dataset developed in this paper.

Figures 13 and 14 depict the dependence analysis of the Shapley values. This involves plotting the input parameters against their individual Shapley values, which enables an assessment to be conducted on the influence of parameter value on creep prediction. An obvious and expected trend is observed in Figure 13a for time since loading, where the Shapley values rise with time in a fashion resembling a creep curve. Figure 13d–f shows that increasing loading age, compressive strength and elastic modulus lead to a reduction in Shapley values, indicating that increasing these parameters reduces creep strain development. No clear trend can be seen for relative humidity and volume-to-surface ratio, aside from the negative Shapley values occurring at the maximum input values ($RH = 100\%$ and $V/S = 60$ mm). These query points are the outliers (dots) shown in Figure 12b, meaning the input values are greater than 1.5 times the interquartile range

below the first quartile. Similar behaviour is shown for slag and fly ash, as the largest input values give negative Shapley scores without an evident trend occurring at lower input values. This suggests that high SCM ratios lead to a reduction in creep strain. However, this effect is likely eclipsed by the changes in compressive strength and elastic modulus that occur when increasing SCM content based on the significance analysis discussed earlier. There is no noticeable relationship between cement, total aggregate or silica fume content on Shapley values. Increasing water content is, however, shown to lead to an increase in Shapley values and, therefore, creep strain.

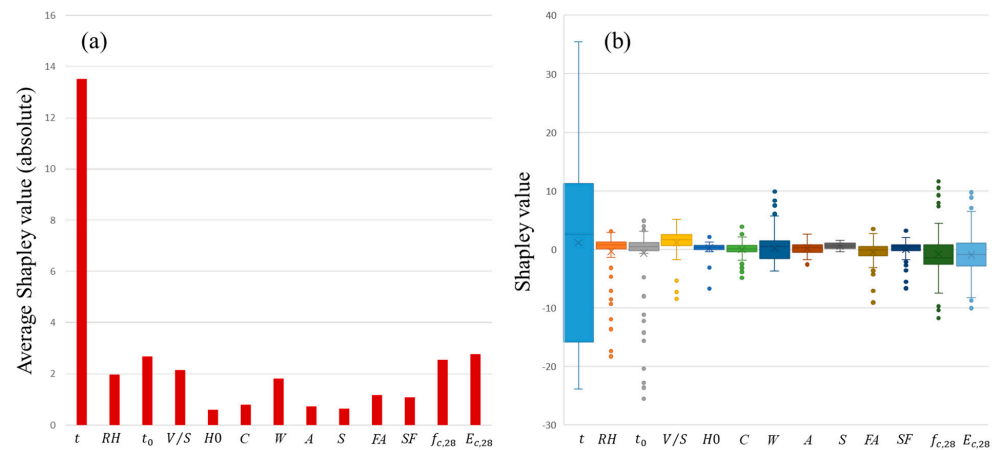


Figure 12. Shapley value absolute averages (a) and a Box and Whisker plot (b) showing Shapley value distribution for fly ash concrete.

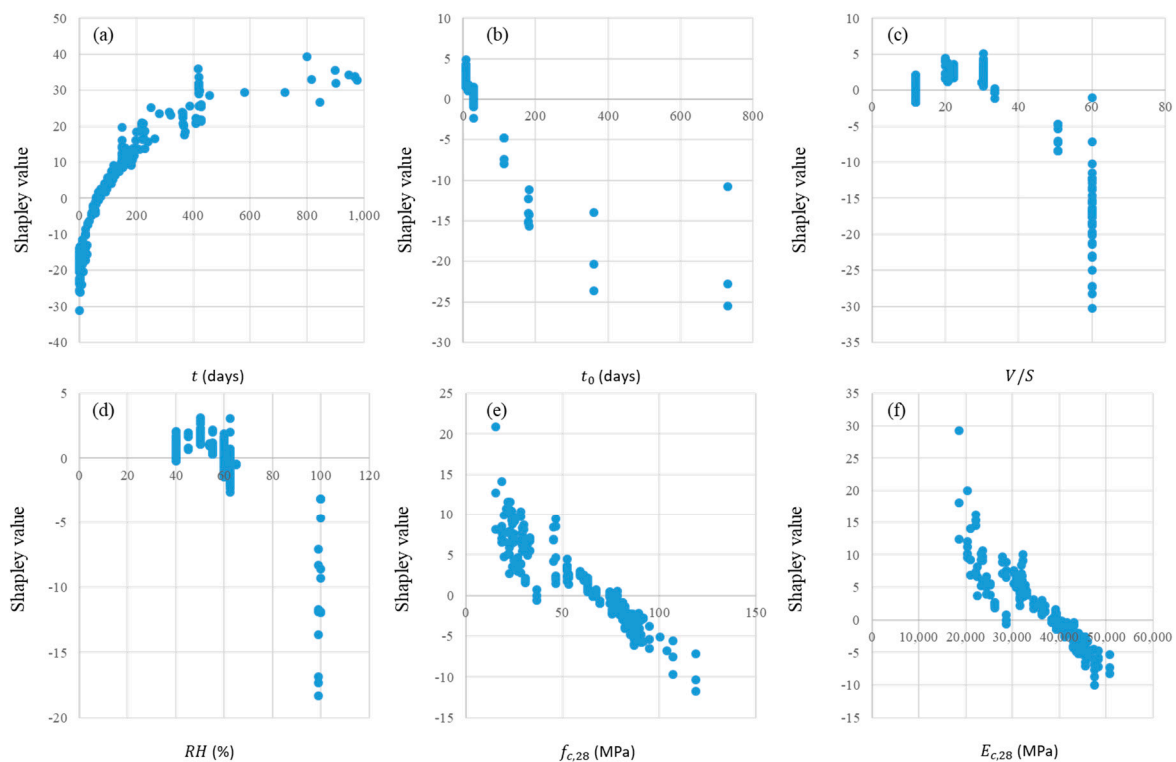


Figure 13. Shapley values versus input parameters: time since loading (a), age at loading (b), volume-to-surface ratio (c), relative humidity (d), 28-day compressive strength (e) and 28-day elastic modulus (f).

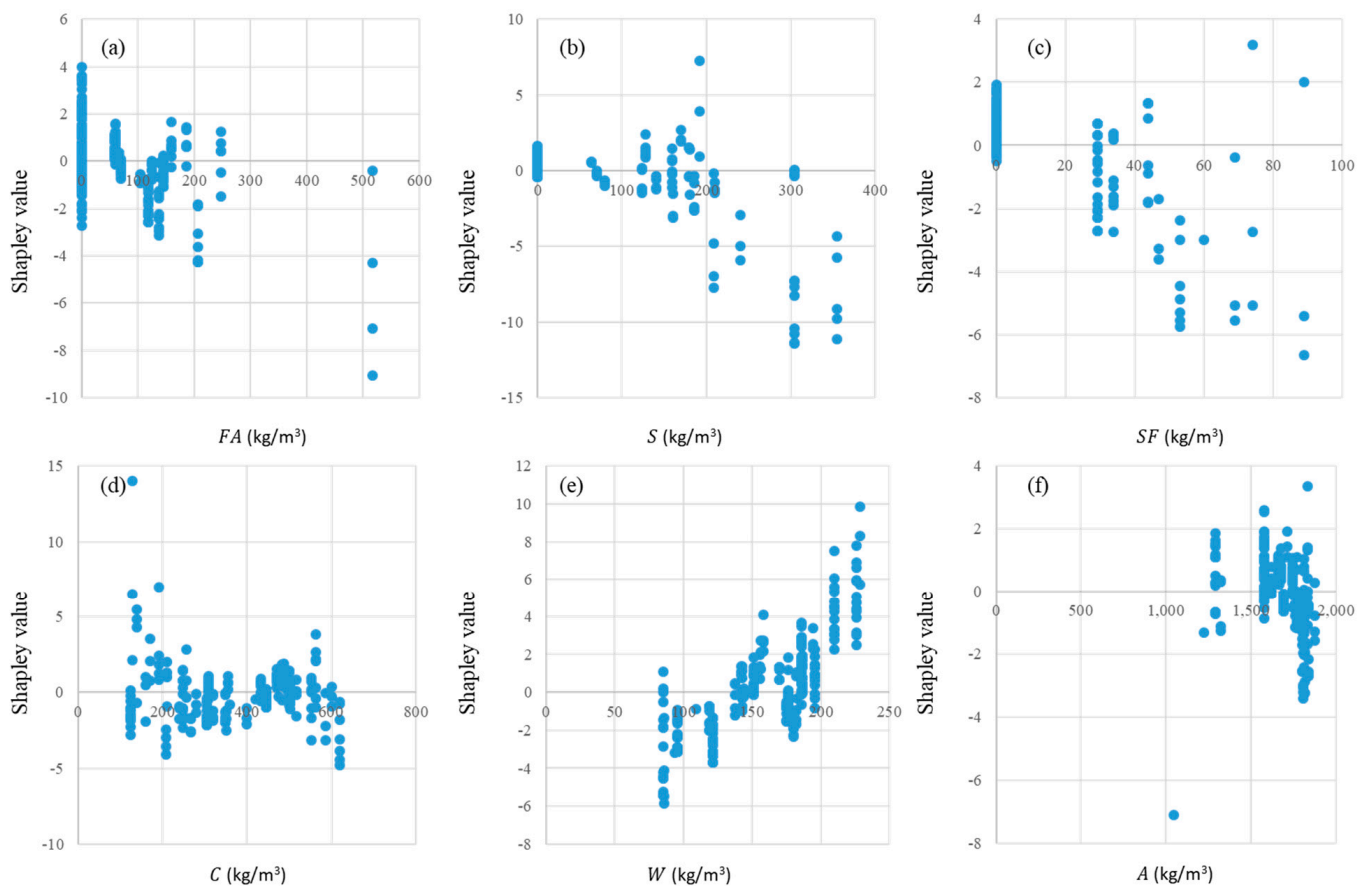


Figure 14. Shapley values versus input parameters: fly ash content (a), slag content (b), silica fume content (c), cement content (d), water content (e) and total aggregate content (f).

7. Model Comparison

The performances of the trained ML models are investigated further by comparing creep strain predictions to experimental data used for model training and by examining the effect of varying input parameters. Figures 15 and 16 show a select set of creep curves for fly ash and slag concrete, respectively, with varying cement substitution ratios and the predictions of the RQ-GPR, RFR, ANN and DTR models. The RQ-GPR, RFR and ANN models achieve highly accurate representations of experimental data. However, it is observed that the DTR model is not capable of reflecting creep behaviour and produces unrealistic results despite showing promising performance based on the examined statistical indicators in Section 4.2. This may be attributed to instabilities with DTR caused by small variations in data, which can result in the generation of completely different trees. Figure 15 also depicts the effect of changing the loading age from 28 days, as conducted in the experiments, to 7 days. In this case, the RQ-GPR, RFR and ANN give similar results, which matches the expectation of an increased creep strain. Figure 16 shows creep strain curves produced by the trained ML models for cement replacement ratios not conducted in the experiments (10%, 30% and 50%). Compressive strength and elastic modulus at intermediate cement substitution ratios were obtained via linear interpolation. Again, the RQ-GPR, RFR and ANN models yield almost identical predictions that satisfy intuitive expectations. The input variables for creep data shown in Figures 15 and 16 are provided in Table 6.

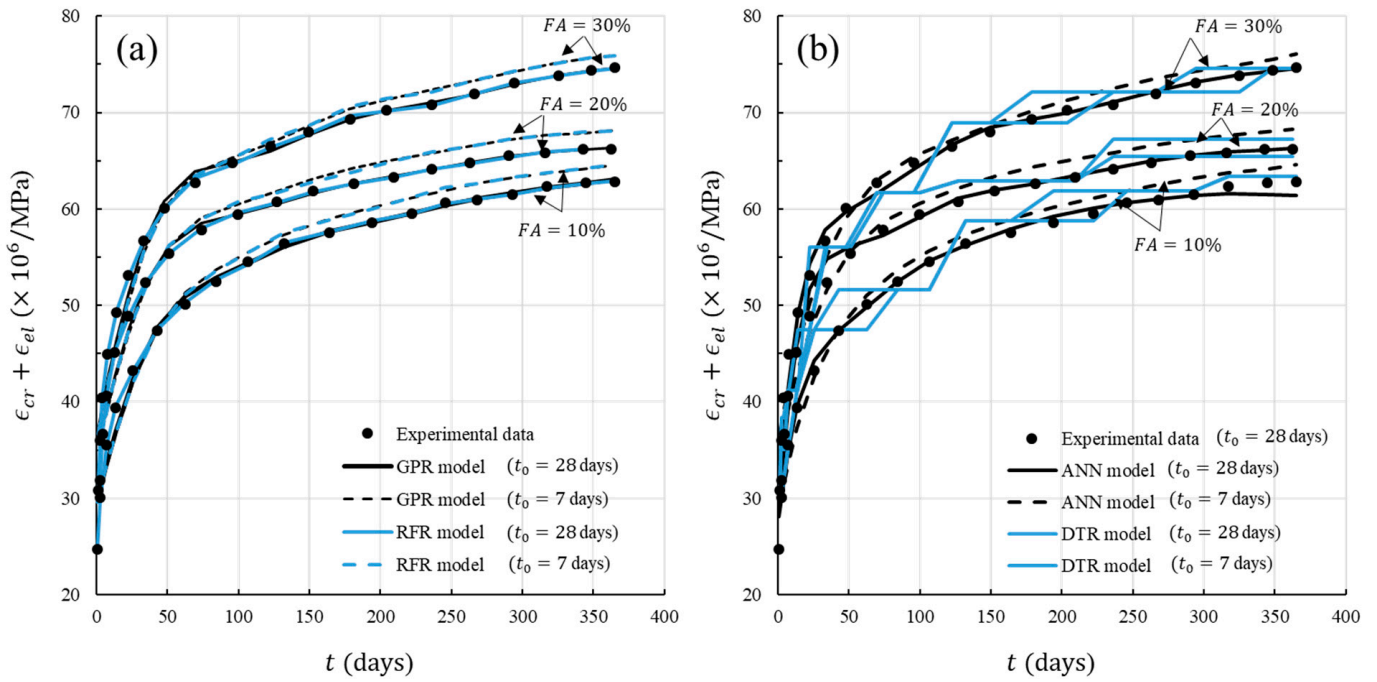


Figure 15. Predictions of GPR and RFR (a) and ANN and DTR (b) model compared to experimental data for fly ash concrete including varied loading age.

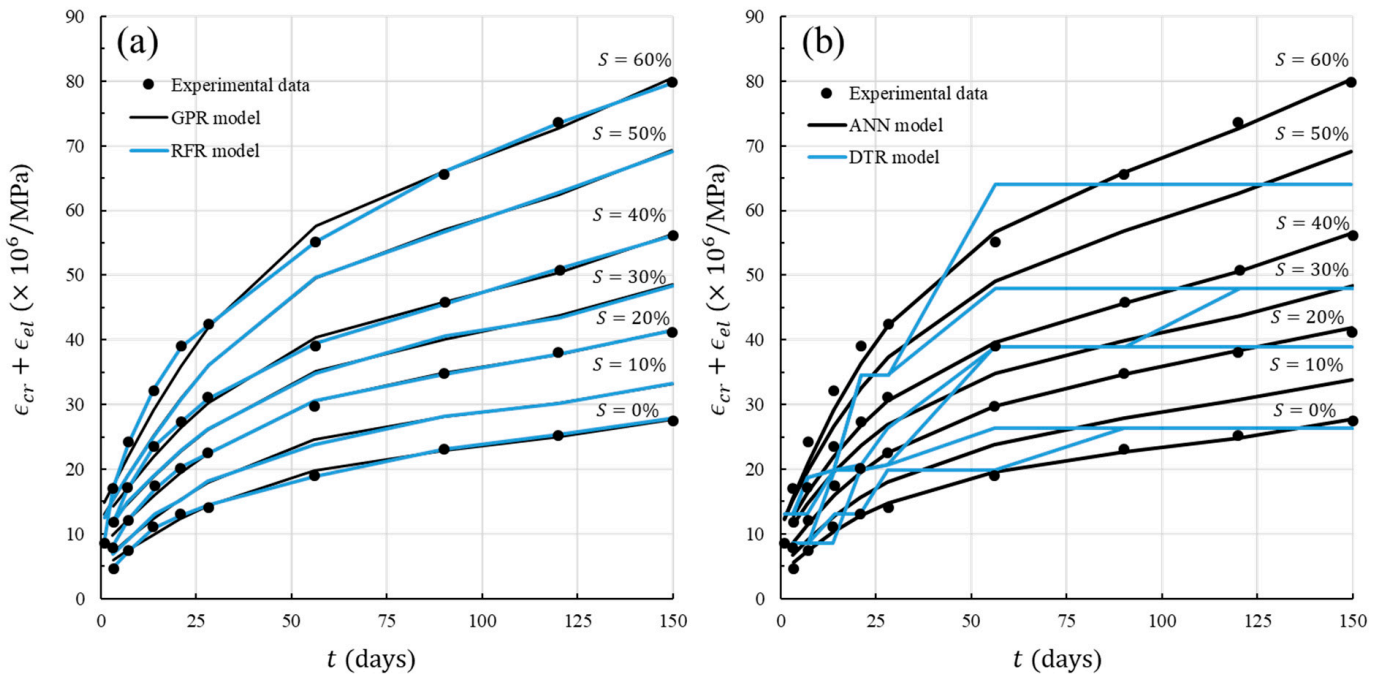


Figure 16. Predictions of GPR and RFR (a) and ANN and DTR (b) model compared to experimental data for slag concrete including varied slag mass.

Table 6. Input variables for model comparisons.

Figure	<i>RH</i> (%)	<i>t</i> ₀ (days)	<i>V/S</i> (mm)	<i>C</i> (kg/m ³)	<i>W</i> (kg/m ³)	<i>A</i> (kg/m ³)	<i>S</i> (kg/m ³)	<i>FA</i> (kg/m ³)	<i>SF</i> (kg/m ³)	<i>f</i> _{<i>c</i>,28} MPa	<i>E</i> _{<i>c</i>,28} MPa
Figure 15	40	28	30.42	620	186	1576	0	69	0	82.3	41,627
Figure 15	40	28	30.42	552	186	1576	0	138	0	76.3	39,444
Figure 15	40	28	30.42	483	186	1576	0	207	0	74.5	39,163
Figure 15	40	7	30.42	620	186	1576	0	69	0	82.3	41,627
Figure 15	40	7	30.42	552	186	1576	0	138	0	76.3	39,444
Figure 15	40	7	30.42	483	186	1576	0	207	0	74.5	39,163
Figure 16	62.5	28	60	350	175	1812	0	0	0	30.8	26,254
Figure 16	62.5	28	60	280	175	1812	70	0	0	26.8	24,479
Figure 16	62.5	28	60	210	175	1812	140	0	0	25.02	23,674
Figure 16	62.5	28	60	140	175	1812	210	0	0	21.86	22,129
Figure 16	62.5	28	60	315	175	1812	35	0	0	28.76	25,367
Figure 16	62.5	28	60	245	175	1812	105	0	0	25.89	24,077
Figure 16	62.5	28	60	175	175	1812	175	0	0	23.44	22,902

8. Conclusions

A creep compliance prediction model was developed for sustainable concretes containing slag and fly ash using ANN-, DTR-, RFR- and GPR-supervised ML algorithms. A dataset of concrete creep was constructed using data extracted from the NU-ITI creep database and from experimental results reported in the literature. The selected input variables were time, loading age, 28-day compressive strength, 28-day elastic modulus, relative humidity during creep testing, volume-to-surface ratio, cement content, water content, total aggregate mass, slag content, fly ash content and silica fume content. The effects of considering humidity during specimen preconditioning as an input variable were also explored. Mode imputation was conducted for missing data on relative humidity and humidity during preconditioning, whereas a 28-day elastic modulus was estimated using the AC1 equation when required. Model training utilised 70% of the dataset, while the remaining 30% was reserved for model verification. A sensitivity analysis was conducted on the GPR model to analyse changes in performance due to training repetitions, input variable selection and validation methodology. A SHAP analysis was performed to investigate the contribution, and hence significance, of each input variable on creep prediction. Additionally, the predictions of the trained ML models were compared to experimental data and the effect of varying input parameters was examined. The following conclusions have been drawn from this work;

- RFR and GPR models were the best-performing ML techniques, achieving R-squared values of 0.99 for validation data. ANN and DTR also achieved accurate results and obtained R-squared values for validation data of 0.91 and 0.90, respectively.
- The selection of kernel function for the GPR model was not found to greatly impact accuracy indicators. Exponential, squared exponential, Matern 5/2 and RQ kernel functions of the GPR model all demonstrated consistent and accurate performance.
- The performance of the GPR model was proven to not be sensitive to training repetitions and random splitting of training and testing data. Only small reductions in accuracy indicators were observed across five training repetitions.
- Consideration of humidity during preconditioning as an input variable for the GPR models did not noticeably improve or reduce accuracy.
- Time since loading was the most significant parameter for creep prediction, followed by compressive strength, elastic modulus, volume-to-surface ratio, loading age, relative humidity and water content.
- Cement content, aggregate content and humidity during preconditioning did not show great importance in creep prediction. Similarly, fly ash and silica fume content were found to have a low influence.

- Slag content did, however, show a moderate importance level for creep modelling.
- The RQ-GPR, RFR and ANN models accurately reflect creep behaviour. However, the DTR model cannot accurately produce concrete creep curves.

Author Contributions: Conceptualization, Y.B. and L.L.; Methodology, Y.B. and L.L.; Software, Y.B. and L.L.; Validation, Y.B.; Formal analysis, Y.B. and L.L.; Writing—original draft, Y.B. and L.L.; Writing—review & editing, Y.B. and L.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. World Green Building Council. Global Status Report 2017. 2017. Available online: <https://worldgbc.org/article/global-status-report-2017/> (accessed on 21 July 2023).
2. Sousa, V.; Bogas, J.A.; Real, S.; Meireles, I. Industrial production of recycled cement: Energy consumption and carbon dioxide emission estimation. *Environ. Sci. Pollut. Res.* **2023**, *30*, 8778–8789. [[CrossRef](#)] [[PubMed](#)]
3. Sandanayake, M.; Bouras, Y.; Haigh, R.; Vrcelj, Z. Current sustainable trends of using waste materials in concrete—A decade review. *Sustainability* **2020**, *12*, 9622. [[CrossRef](#)]
4. Bouras, Y.; Vrcelj, Z. Fractional and fractal derivative-based creep models for concrete under constant and time-varying loading. *Constr. Build. Mater.* **2023**, *367*, 130324. [[CrossRef](#)]
5. Shariq, M.; Prasad, J.; Abbas, H. Creep and drying shrinkage of concrete containing GGBFS. *Cem. Concr. Compos.* **2016**, *68*, 35–45. [[CrossRef](#)]
6. Qin, D.; Dong, C.; Zong, Z.; Guo, Z.; Xiong, Y.; Jiang, T. Shrinkage and creep of sustainable self-compacting concrete with recycled concrete aggregates, fly ash, slag, and silica fume. *J. Mater. Civ. Eng.* **2022**, *34*, 04022236. [[CrossRef](#)]
7. Li, J.; Yao, Y. A study on creep and drying shrinkage of high performance concrete. *Cem. Concr. Res.* **2001**, *31*, 1203–1206. [[CrossRef](#)]
8. Gedam, B.A.; Bhandari, N.; Upadhyay, A. Influence of supplementary cementitious materials on shrinkage, creep, and durability of high-performance concrete. *J. Mater. Civ. Eng.* **2016**, *28*, 04015173. [[CrossRef](#)]
9. Chern, J.-C.; Chan, Y.-W. Deformations of concretes made with blast-furnace slag cement and ordinary portland cement. *Mater. J.* **1989**, *86*, 372–382.
10. Salimpour, M.R.; Darvanjooghi, M.H.K.; Abdollahi, A.; Karimipour, A.; Goodarzi, M. Providing a model for Csf according to pool boiling convection heat transfer of water/ferrous oxide nanofluid using sensitivity analysis. *J. Numer. Methods Heat Fluid. Flow.* **2020**, *30*, 2867–2881. [[CrossRef](#)]
11. He, W.; Bagherzadeh, S.A.; Shahrajabian, H.; Karimipour, A.; Jadidi, H.; Bach, Q.-V. Controlled elitist multi-objective genetic algorithm joined with neural network to study the effects of nano-clay percentage on cell size and polymer foams density of PVC/clay nanocomposites. *J. Therm. Anal. Calorim.* **2020**, *139*, 2801–2810. [[CrossRef](#)]
12. Kazemi, F.; Asgarkhani, N.; Jankowski, R. Machine learning-based seismic response and performance assessment of reinforced concrete buildings. *Arch. Civ. Mech. Eng.* **2023**, *23*, 94. [[CrossRef](#)]
13. Bagherzadeh, F.; Shafighfard, T.; Khan, R.M.A.; Szczuko, P.; Mieloszyk, M. Prediction of maximum tensile stress in plain-weave composite laminates with interacting holes via stacked machine learning algorithms: A comparative study. *Mech. Syst. Signal Process.* **2023**, *195*, 110315. [[CrossRef](#)]
14. Bouras, Y.; Li, L. Prediction of high-temperature creep in concrete using supervised machine learning algorithms. *Constr. Build. Mater.* **2023**, *400*, 132828. [[CrossRef](#)]
15. Shojaei, S.; Shahgholi, M.; Karimipour, A. The effects of atomic percentage and size of Zinc nanoparticles, and atomic porosity on thermal and mechanical properties of reinforced calcium phosphate cement by molecular dynamics simulation. *J. Mech. Behav. Biomed. Mater.* **2023**, *141*, 105785. [[CrossRef](#)] [[PubMed](#)]
16. Bažant, Z.P.; Li, G.-H. *Comprehensive Database on Concrete Creep and Shrinkage*; Infrastructure Technology Institute, McCormick School of Engineering and Applied Science, Northwestern University: Evanston, IL, USA, 2008.
17. Bal, L.; Buyle-Bodin, F.J.N.C. Artificial neural network for predicting creep of concrete. *Neural Comput. Appl.* **2014**, *25*, 1359–1367. [[CrossRef](#)]
18. Zhu, J.; Wang, Y. Convolutional neural networks for predicting creep and shrinkage of concrete. *Constr. Build. Mater.* **2021**, *306*, 124868. [[CrossRef](#)]
19. Gandomi, A.H.; Sajedi, S.; Kiani, B.; Huang, Q. Genetic programming for experimental big data mining: A case study on concrete creep formulation. *Autom. Constr.* **2016**, *70*, 89–97. [[CrossRef](#)]

20. Liang, M.; Chang, Z.; Wan, Z.; Gan, Y.; Schlangen, E.; Šavija, B. Interpretable Ensemble-Machine-Learning models for predicting creep behavior of concrete. *Cem. Concr. Compos.* **2022**, *125*, 104295. [[CrossRef](#)]
21. Rong, X.; Liu, Y.; Chen, P.; Lv, X.; Shen, C.; Yao, B. Prediction of creep of recycled aggregate concrete using back-propagation neural network and support vector machine. *Struct. Concr.* **2022**, *24*, 2229–2244. [[CrossRef](#)]
22. Feng, J.; Zhang, H.; Gao, K.; Liao, Y.; Gao, W.; Wu, G. Efficient creep prediction of recycled aggregate concrete via machine learning algorithms. *Constr. Build. Mater.* **2022**, *360*, 129497. [[CrossRef](#)]
23. Feng, J.; Zhang, H.; Gao, K.; Liao, Y.; Yang, J.; Wu, G. A machine learning and game theory-based approach for predicting creep behavior of recycled aggregate concrete. *Case Stud. Constr. Mater.* **2022**, *17*, e01653. [[CrossRef](#)]
24. Sadowski, Ł.; Nikoo, M.; Shariq, M.; Joker, E.; Czarnecki, S. The nature-inspired metaheuristic method for predicting the creep strain of green concrete containing ground granulated blast furnace slag. *Materials* **2019**, *12*, 293. [[CrossRef](#)]
25. Gedam, B.A.; Bhandari, N.; Upadhyay, A. An apt material model for drying shrinkage and specific creep of HPC using artificial neural network. *Struct. Eng. Mech.* **2014**, *52*, 97–113. [[CrossRef](#)]
26. Chen, P.; Zheng, W.; Wang, Y.; Chang, W. Creep model of high-strength concrete containing supplementary cementitious materials. *Constr. Build. Mater.* **2019**, *202*, 494–506. [[CrossRef](#)]
27. Liu, Y.; Li, Y.; Mu, J.; Li, H.; Shen, J.J.C. Modeling and analysis of creep in concrete containing supplementary cementitious materials based on machine learning. *Constr. Build. Mater.* **2023**, *392*, 131911. [[CrossRef](#)]
28. Chaabene, W.B.; Flah, M.; Nehdi, M.L. Machine learning prediction of mechanical properties of concrete: Critical review. *Constr. Build. Mater.* **2020**, *260*, 119889. [[CrossRef](#)]
29. Vincent, E.C.; Townsend, B.D.; Weyers, R.E.; Via, C. *Final Contract Report Creep of High-Strength Normal and Lightweight Concrete*; Virginia Transportation Research Council: Charlottesville, VA, USA, 2004.
30. Collins, T.M. Proportioning high-strength concrete to control creep and shrinkage. *Mater. J.* **1989**, *86*, 576–580.
31. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *J. Neural Netw.* **1989**, *2*, 359–366. [[CrossRef](#)]
32. Khademi, F.; Akbari, M.; Jamal, S.M. Prediction of compressive strength of concrete by data-driven models. *I-Manag. J. Civ. Eng.* **2015**, *5*, 16–23. [[CrossRef](#)]
33. Smith, J.S.; Wu, B.; Wilamowski, B.M. Neural network training with Levenberg–Marquardt and adaptable weight compression. *J. IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *30*, 580–587. [[CrossRef](#)]
34. de Azevedo, C.R.; von Stosch, M.; Costa, M.S.; Ramos, A.M.; Cardoso, M.M.; Danhier, F.; Pr at, V.; Oliveira, R. Modeling of the burst release from PLGA micro-and nanoparticles as function of physicochemical parameters and formulation characteristics. *J. Int. J. Pharm.* **2017**, *532*, 229–240. [[CrossRef](#)] [[PubMed](#)]
35. Alatefi, S.; Almeshal, A.M. A new model for estimation of bubble point pressure using a bayesian optimized least square gradient boosting ensemble. *J. Energ.* **2021**, *14*, 2653. [[CrossRef](#)]
36. Deringer, V.L.; Bart ok, A.P.; Bernstein, N.; Wilkins, D.M.; Ceriotti, M.; Cs anyi, G. Gaussian process regression for materials and molecules. *Chem. Rev.* **2021**, *121*, 10073–10141. [[CrossRef](#)] [[PubMed](#)]
37. Ali, P.J.M.; Faraj, R.H.; Koya, E.; Ali, P.J.M.; Faraj, R.H. Data normalization and standardization: A technical report. *J. Mach. Learn. Tech. Rep.* **2014**, *1*, 1–6.
38. Dietterich, T.G. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *J. Mach. Learn.* **2000**, *40*, 139–157. [[CrossRef](#)]
39. Rizvon, S.S.; Jayakumar, K. Strength prediction models for recycled aggregate concrete using Random Forests, ANN and LASSO. *J. Build. Pathol. Rehabil.* **2022**, *7*, 5. [[CrossRef](#)]
40. Alkadhim, H.A.; Amin, M.N.; Ahmad, W.; Khan, K.; Nazar, S.; Faraz, M.I.; Imran, M. Evaluating the strength and impact of raw ingredients of cement mortar incorporating waste glass powder using machine learning and SHapley additive ExPlanations (SHAP) methods. *Materials* **2022**, *15*, 7344. [[CrossRef](#)]
41. Lundberg, S.; Lee, S. *A Game Theoretic Approach to Explain the Output of Any Machine Learning Model*; Github: San Francisco, CA, USA, 2021.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.