And., Intl.,J., Appl Sci., Eng., Tech, Vol 04, No 03, November 2024
e-ISSN: 2797-0442

Prediction of the Residual Compressive Strength of Rice Husk Ash Concrete after Exposure to Elevated Temperatures Using XGBoost Machine Learning Algorithm

Elvis Ang'ang'o*, Silvester Abuodha, Siphila Mumenya
Department of Civil & Construction Engineering, University of Nairobi, Kenya
*Corresponding author, e-mail: elvisangango80@students.uonbi.ac.ke

Abstract— The study aimed to assess the applicability of XGBoost in determining the residual compressive strength of rice husk ash concrete exposed to elevated temperature reducing the need for costly, time-consuming laboratory experiments. Data was collected from the available literature with 75% used for training and 25% for testing. Synthetic data was created using the Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP). The model accuracy was checked using statistical scores: coefficient of determination (R^2), root mean squared error (RMSE), and mean absolute error (MAE). SHAP values were used for feature importance analysis. Coding was done in Python using Jupyter Notebook. With the original data, the model produced RMSE, R^2, and MAE test values of 3.351, 0.939, and 2.994 respectively indicating excellent performance. The combined original and synthetic dataset gave RMSE, R^2, and MAE values of 0.071, 0.941, and 0.053 respectively signifying improved performance. The feature analysis identified higher temperature, unheated compressive strength, and water-cement ratio as the most significant factors in the XGBoost prediction. The exposure duration, alumina content, and iron oxide had minimal influence.

Keywords: Residual concrete compressive strength, Rice husk ash concrete, Machine learning, XGBoost, WGAN, SHAP.

1. Introduction

Climate change is a significant global issue resulting in detrimental effects, including elevated temperatures, severe droughts, intensified storms, increasing sea levels, and glacial melting, The production of cement is reported to be a very energy-intensive activity that produces a lot of CO_2 emissions that contribute to the greenhouse effect and cause global warming [1]. This process is also quite expensive thus subsequently increasing the cost of construction.

Higher temperatures greater than the normal atmospheric temperature will cause concrete to undergo thermal cracking whereby increased internal stresses in the concrete due to the heat causes it to crack. For reinforced concrete when cracks form, they expose steel reinforcement and in places such as coastal areas where there is chloride in the salt water, this exposes them to rust. The high temperatures cause the bending and spalling of concrete and eventually may lead to structural failure of concrete structures. These elevated temperatures are expected in industries with furnaces, airfield pavements, nuclear power plants, jet aircraft engine blasts, and building fires. Thus, there is a need to design and construct these concrete structures to perform adequately under elevated temperatures.

Rice husk ash is an environmentally safe waste material obtained from rice that has been proposed as a supplemental cementitious material. Its production does not involve the emission of CO_2 and thus will not have any adverse effect on the climate. Furthermore, being agro-waste, it is easily and readily obtained, reducing the cost of production, and leading to a decrease in construction cost. The literature

Received: October 05, 2024, Revised: October 22, 2024, Accepted: October 29, 2024
https://doi.org/10.25077/aijaset.v4i3.187

further implies the incorporation of rice husk ash into concrete has positive effects on its properties which include higher compressive strength and durability [2], [3], [4].

Rice husk ash concrete is eco-friendly, requires less energy to produce, and consists of numerous components. In service, it is affected by multiple factors, including elevated temperature in building fires, in industries close to furnaces, nuclear power plants, airfield pavements, and jet aircraft blast hangers. Machine learning can determine various patterns in the rice husk ash concrete data beyond simple analysis. It can provide a better understanding of concrete behavior and evaluate future performance, negating the need for extensive laboratory experiments that are expensive, time-consuming, and contribute to waste generation.

Previous studies [5], [6], [7] have employed various machine learning techniques to evaluate the residual concrete compressive strength following elevated temperature exposure. However, none has used the XGBoost to determine RHA concrete residual compressive strength considering the chemical composition and fineness of the RHA as input parameters. XGBoost is considered robust, rapid in training, and precise. This study improves the understanding of RHA concrete behavior under elevated temperatures. The study provides a basis for the algorithm application in other civil engineering data sets where the input and output parameters are clearly defined to obtain desired results or predict a given outcome as the machine learning algorithm can map patterns of a complex data set.

Cakiroglu [8] used the tabular generative adversarial network to produce synthetic data, finding that augmenting the data set with synthetic data enhanced the performance of machine learning models. Nevertheless, the study advocated additional research aimed at generating higher-quality synthetic data. Given the significant reliance of machine learning algorithms on data, the present research investigated the application of the Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP), an improved version of the traditional GAN, to generate continuous numerical synthetic data. This innovative methodology offers a fresh perspective for the civil engineering sector in addressing the challenges posed by limited data availability, thereby reducing the necessity for costly and labor-intensive laboratory experiments.

2. Method

The research approach encompassed data set collecting, the formulation of the XGBoost model, and statistical validation of the model accuracy utilizing mean absolute error, root mean square error, and coefficient of determination. An analysis of the importance of the feature was performed to ascertain the impact of the input parameters on the result. Synthetic data was generated using the Wasserstein Generative Adversarial Network with Gradient Penalty which was contrasted with the real data and employed in the XGBoost model.

2.1 Data Collection

The data was collected from previous experimental studies by Al-Majidi [9], Umasabor & Okovido [10], and Wang [11] which included 64 samples. The gathered data set exhibited high quality characterized by its accuracy in representing the pertinent issue, reliability through the absence of contradictory information, and completeness, ensuring no duplicates or missing entries. Additionally, the data was up to date, possessed an appropriate size, was at least an order of magnitude larger than the trainable parameters, demonstrated diversity, and maintained relevance to the subject matter. In contrast to deep learning and unsupervised learning algorithms, which necessitate substantial quantities of data to discern intrinsic patterns in the data without explicit guidance, XGBoost is a supervised learning algorithm utilizing a pre-labeled sample data set for training, allowing it to make predictions while iteratively refining its parameters to reduce error. The labeled data sets provide contextual information, equipping the model with the target output values necessary for generating accurate predictions. The effectiveness of the model is dependent on the availability of well-labeled data and a well-defined model objective. The data collected was well labeled, devoid of missing values, exhibited appropriate variance, and the volume was adequate for the machine learning model to discern the underlying patterns. The variables included the age of the specimen, water-cement ratio, rice husk ash percentage replacement, chemical composition of the RHA (silica SiO_2, alumina Al_2O_3, and iron oxide Fe_2O_3), RHA fineness, unheated compressive strength (at room temperature), intensity of the elevated temperature and duration

of exposure to elevated temperature. The elevated temperature ranged from 100^0C to 800^0C. The important data set components considered are shown in Table 1. The statistical analysis of the data reported in Table 2 reveals a range of variability and diversity. This characteristic facilitates the model's ability to effectively generalize to novel inputs and additional data.

Table 1. Important Data Set Components

Research / Citation	**RHA (%)**	**Chemical composition (%)**	**RHA Fineness 100% passing (Sieve size-µm)**	**w/c ratio**	**Unheated compressive strength (MPa)**	**Elevated temp (^{0}C)**	**Exposure duration (hours)**	**Age (days)**	**Data size**
Wang [11]	0 10 15 20	SiO_2-84.00 Al_2O_3-1.35 Fe_2O_3-1.45	90	0.3	64 65 66 67	200 400 600 800	2	59	28
Al-Majidi [9]	0 5 10 20 30	SiO_2-87.16 Al_2O_3-0.38 Fe_2O_3-0.35	45	0.55	35.96 39.12 39.49 41.14 46.5	200 400 600 800	1	28	20
Umasabor & Okovido [10]	0 5 10 15	SiO_2-87.22 Al_2O_3-0.70 Fe_2O_3-1.68	75	0.6	22 24.2 27.2 40	100 200 300 400 500 600 700	2	200	16

Table 2. Data Set Statistics

Variable	**Mean**	**Standard deviation**	**Minimum**	**25th Percentile**	**50th Percentile**	**75th Percentile**	**Maximum**
RHA	10.156	8.403	0	5	10	15	30
SiO_2	66.159	36.914	0	84	87.16	87.22	87.22
Al_2O_3	0.578	0.456	0	0.38	0.7	0.7	1.35
Fe_2O_3	0.911	0.728	0	0.35	1.45	1.68	1.68
RHA Fineness	52.734	33.273	0	45	75	75	90
w/c ratio	0.509	0.124	0.3	0.488	0.55	0.6	0.6
Unheated comp str	41.540	15.991	22	27.2	39.485	50.875	68
Elevated temp	456.25	221.019	100	200	400	600	800
Exposure duration	1.688	0.467	1	1	2	2	2
Age	111	79.963	28	28	59	200	200
Residual comp str	30.988	19.522	3.3	17.386	25.65	43.88	73

AIJASET – Vol. 04, No. 03, November 2024. 193-205
https://doi.org/10.25077/aijaset.v4i3.187

2.2 Model Training and Testing

The research flow chart is shown in Figure 1.

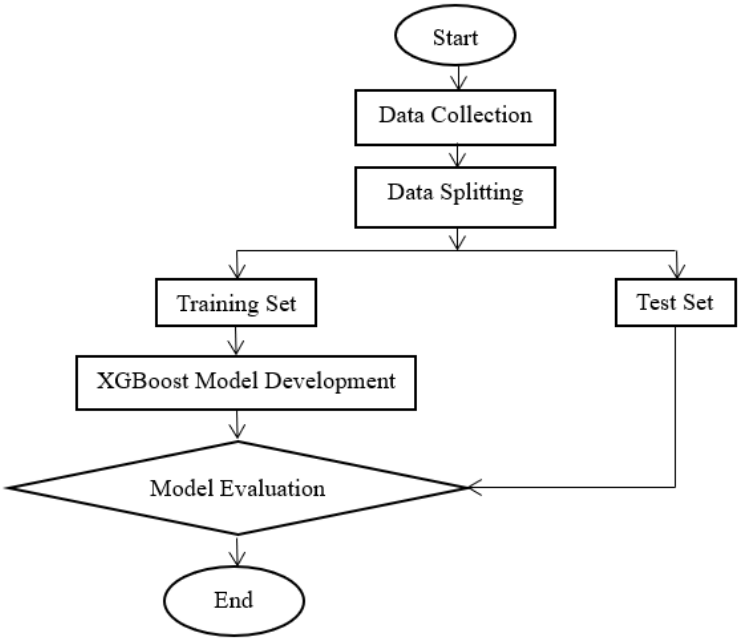

Figure 1. Research flow chart

Extreme gradient boosting (XGBoost) is a powerful and scalable machine learning based on gradient boosting. Gradient boosting is an ensemble machine-learning technique derivative of the Decision Tree algorithm. This technique employs a series of weak/base learners to obtain an arbitrarily strong learner, with the fundamental objective of reducing the overall error, referred to as model loss, through the iterative addition of learners to the model [12]. XGBoost effectiveness largely stems from its exceptional scalability, allowing it to efficiently handle billions of instances in both distributed and memory-listed scenarios while also maintaining rapid performance on individual systems [13].

The model proposed by Chen & Guestrin [13] gives the predicted result using Equation 1 which uses additive functions.

$$\overline{y_i} = y_i^0 + \eta \sum_{k=1}^{M} f_k(X_i) \quad (1)$$

The term $\overline{y_i}$ denotes the predicted outcome for the ith sample, *Xi* represents the feature vector associated with that sample, *M* indicates the total number of estimators, f_k refers to the individual estimators where k varies from 1 to M each linked to a distinct tree structure, y_i^0 signifies the average of the observed values within the training set serving as the initial estimate, and the learning rate is denoted as η that plays a crucial role in progressively refining the model while incorporating new trees and mitigating the risk of overfitting [14].

Training is done in an additive manner; therefore, a kth estimator is added to the model at the kth step, and the kth prediction is given by Equation 2.

$$\overline{y_i^k} = \overline{y}_i^{(k-1)} + \eta f_k \quad (2)$$

The value of f_k is established based on the leaf weights derived from the minimization of the objective function associated with the kth tree using Equation 3 [14].

$$obj = \gamma T + \sum_{j=1}^{T} [G_j \omega_j + \frac{1}{2}(H_j + \lambda)\omega_j^2] \quad (3)$$

T represents the total count of leaves in the kth tree, while ω_j denotes the weights of the leaves with j varying from 1 to T. γ and λ serve as regularization parameters that regulate the complexity of the

tree structure, thereby mitigating the risk of overfitting. *Gj* and *Hj* correspond to the summation and count of residual linked to the jth leaf, respectively [14].

The kth tree is constructed by partitioning the leaves, beginning with a single leaf, while maximizing the gain parameter defined by Equation 4. Following the division, the left leaf is designated G_L and H_L, while the right is assigned G_R and H_R [14].

$$gain = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \quad (4)$$

The model training and testing were done in Python using the scikit-learn machine learning library for implementing machine learning models. Jupyter Notebook was used as the integrated development environment (IDE) for efficient code development. Hyperparameter optimization was performed to determine the most effective hyperparameter settings for enhancing model performance.

Optuna framework which employs Bayesian optimization with an algorithm called TPE (Tree-Structured Parzen Estimator) was used for hyperparameter tuning. Iteratively sampling new hyperparameters, the TPE algorithm updates its internal probability distributions, assesses performance using the objective function, and discloses a hyperparameter combination that it anticipates will produce an even better outcome [15].

The tuning process aimed to reduce the root mean squared error. The model was developed with a focus on the following hyperparameters: n_estimators varied between 100 and 1600, max_depth was set from 1 to 11, learning_rate was adjusted within the range of 0.01 to 1.0, subsample values spanned from 0.4 to 1.1, colsample_bytree was configured between 0.01 and 1.0, regularization parameters included lambda ranging from 1 to 100, alpha from 0 to 11 and gamma from 1 to 11 [12], [14], [16].

2.3 Performance Evaluation of the Model

The assessment of the model's performance was conducted through the application of the coefficient of determination (R^2) as outlined in Equation 5, the root mean squared error (RMSE) as specified in Equation 6, and the mean absolute error (MAE) as described in Equation 7.

$$R^2 = 1 - \frac{\sum_{i=0}^{n}(y-\hat{y})^2}{\sum_{i=0}^{n}(y-\bar{y})^2} \quad (5)$$

$$RMSE = \sqrt{\frac{1}{n}}\sum_{i=0}^{n}(y-\hat{y})^2 \quad (6)$$

$$MAE = \frac{1}{n}\sum_{i=0}^{n}|y-\hat{y}| \quad (7)$$

y represents the actual value, $\hat{y}$ denotes the predicted value, $\bar{y}$ signifies the mean of the actual value and n indicates the total number of samples in the testing data set. The units for RMSE and MAE correspond to those of the output parameter which is MPa in this case. A lower value indicates a more precise model prediction. In contrast, R^2 is a dimensionless quantity; values approaching 1 signify improved predictive accuracy of the model [16].

R^2 quantifies the extent to which the model's predictions account for the overall variability observed in the target variable. RMSE was employed to compute the average squared deviation between the predicted and actual values, effectively imposing a penalty on larger discrepancies through the squaring process. The RMSE is sensitive to outliers and is a good indicator of a robust model that can handle unseen data. Unlike mean square error, MAE is less sensitive to outliers and quantifies the size of the discrepancies between predicted and observed values, disregarding the direction of these errors. A lower MAE denotes a better match between the model predictions and the true values, implying that the model's predictions are generally more accurate.

2.4 Feature Importance Analysis

The input variables were examined to assess their impact on the model. This was done using SHAP values. This prevents the "black box" problem associated with machine learning algorithms whereby how the model functions cannot be easily explained.

SHAP (Shapely Additive exPlanation) values is a model-agnostic feature importance analysis tool proposed by Lundberg & Lee [17] which is designed to elucidate a specific prediction by quantifying the contribution of each feature to the prediction. SHAP values employ an explanatory model g that is represented as a linear function of binary variables as illustrated in Equation 8.

$$g(z') = \phi_0 + \sum_{I=1}^{M} \phi_i \, z'_i \tag{8}$$

z' represents the simplified features belonging to the set $\{0,1\}^M$, M denotes the total number of input features, ϕ_i are the feature attribution values (SHAP values) and z'_i are variables indicating a feature being observed when $z'_i = 1$ (corresponding feature value is present) and unknown (corresponding feature value is absent) when $z'_i = 0$ [18].

The SHAP value ϕ_i attributed to each feature is obtained using Equation 9.

$$\phi_i = \sum_{S \subseteq N\{i\}} \frac{|S|!(M-|S|-1)!}{M!} [f_x(S \cup \{i\}) - f_x(S)] \tag{9}$$

Where $f_x(S) = E[f(x) \mid xS]$, S represents the collection of non-zero indices in *z'*, $E[\, f(x) \mid xS\,]$ denotes the expected value of the function given a specific subset S of the input features, M refers to the total number of input features, and N encompasses the entirety of the input features [18].

2.5 Synthetic Data Generation

Generative Adversarial Networks (GANs) developed by Goodfellow [19] is a framework used to generate synthetic data. It comprises two distinct neural network architectures: the generator and the discriminator. The generator produces data that resembles real data by utilizing random noise, whereas the discriminator is tasked with distinguishing between this generated data and genuine data. The two models operate in a competitive framework, and this adversarial training facilitates their progressive enhancement over time.

The GANs face challenges such as convergence failure and mode collapse. Convergence failure is whereby the GANs fail to provide good quality results and mode collapse occurs when the generator only focuses on certain modes in the real data and fails to consider the different variations in the data distribution thereby producing synthetic data that is not diverse.

Wasserstein GANs proposed by Arjovsky [20] address these issues of mode collapse and convergence in the original GAN by considering the Wasserstein distance as the loss function, in contrast to the binary cross entropy that relies on the Jensen-Shannon divergence. The Jensen-Shannon divergence quantifies the similarity between the two distributions, exhibits symmetry, and always has a value. The WGAN differentiates the disparity between the distribution of real data and that of generated data, a measure referred to as the Earth Mover distance which calculates the distance required to move the mass of features from one point to another so that the two distributions are similar. The GAN classifies the output as either real or fake. In contrast, the WGAN discriminator is designed to assess the disparity between the actual and generated distributions, which is why it is commonly termed the critic.

The WGAN facilitates a more stable learning environment, enhances the convergence of gradients, minimizes the occurrence of mode collapse, and yields superior optimal outcomes. The loss function of the WGAN is shown in Equation 10.

$$\max_{w \in W} E_{x \sim pr}[f_w(x)] - E_{z \sim p(z)}[f_w(g_\theta(x))] \tag{10}$$

$E_{x \sim pr}[f_w(x)]$ represents the distribution of real data, $E_{z \sim p(z)}[f_w(g_\theta(x))]$ denotes the distribution of data generated from random noise z, and W signifies the Wasserstein distance [20]. The critic is optimized to increase this loss while the generator is optimized to decrease it.

To ensure the generator gets enough gradient information for training, a gradient penalty is introduced in the WGAN as proposed by Gulrajani [21]. The new loss function is defined by Equation 11.

$$L = \mathop{\mathrm{E}}_{\tilde{x} \sim P_g}[D(\tilde{x})] - \mathop{\mathrm{E}}_{x \sim P_r}[D(x)] + \lambda \mathop{\mathrm{E}}_{\hat{x} \sim P_{\hat{x}}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2] \tag{11}$$

The expression $\underset{\tilde{x}\sim P_g}{\mathrm{E}}[D(\tilde{x})] - \underset{x\sim P_r}{\mathrm{E}}[D(x)]$ represents the original critic's loss where $\underset{\tilde{x}\sim P_g}{\mathrm{E}}[D(\tilde{x})]$ corresponds to the generated data and $\underset{x\sim P_r}{\mathrm{E}}[D(x)]$ the real data. $\lambda \underset{\hat{x}\sim P_{\hat{x}}}{\mathrm{E}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2]$ denotes the gradient penalty with λ indicating the gradient penalty coefficient and $\hat{x}$ representing the interpolated data given by Equation 12.

$$\hat{x} = \epsilon x + (1 - \epsilon)\tilde{x} \tag{12}$$

Synthetic data was produced utilizing WGAN-GP. The original data was first scaled. Data scaling ensures that every feature contributes equally to the model and prevents the dominance of input variables with higher values. Normalization was used to scale the data which limits the value of each feature between 0 (minimum) and 1 (maximum) using Equation 13 where *X'* represents the value of the feature after scaling, *x* denotes the original value of the unscaled feature, x_{min} indicates the minimum feature value and x_{max} signifies the maximum feature value.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{13}$$

A heuristic approach was used to determine the WGAN-GP network parameters. The WGAN-GP was created with a batch size of 64. The batch size specifies how many rows of data will be passed through the network in one go (cycle) after which the neural network modifies its weights based on the errors. A small batch size makes the process take a long time while a large batch size may reduce the model accuracy. For a small data set the number of samples can be taken as the batch size therefore 64 was used. Epoch represents the number of times data is passed through the network. The number of epochs chosen was 200,000. This takes a long time to implement but improves accuracy.

The Generator and Discriminator networks had 3 layers each: the input, hidden, and output layers. The size of the input layer for both the Generator and Discriminator was 11 units (neurons), representing the number of features to be produced. The hidden layer units for both networks were set to 512, as an increase in neurons introduces complexity, whilst a decrease results in diminished accuracy. The output layer for the Generator network had 11 units, which represents the total amount of features produced while the output layer of the Discriminator had just one unit as its purpose is to differentiate the data produced as either real or fake.

Layer Normalization, which normalizes all features within each sample was used after the input and hidden layer as opposed to the Batch Normalization employed in the original WGAN-GP paper [21] which normalizes each feature independently within a batch of samples (mini-batch). Layer Normalization was seen to improve training speed and is suitable for small batch sizes as it is independent of the batch size.

The network employs activation functions to identify intricate nonlinear relationships among the inputs, transforming them into a more practical output. The generator network employed the Rectified Linear Unit (ReLU) activation function across its layers while the Leaky Rectified Linear Unit (Leaky ReLU) was used for the discriminator layers. ReLU limits the outputs from the layers to 0 and above thereby ensuring that no negative value is obtained from the layers. The neurons are not activated by the ReLU function simultaneously, rather they are deactivated only when the linear transformation's output is less than zero. This however may lead to the neurons with a negative bias never getting activated resulting in a dead neuron leading to poor network performance as it is unable to learn complex patterns in the data. To combat this, Leaky ReLU is employed whereby for the positive outputs it functions just like ReLU but for negative outputs instead of returning zero it returns a small negative value proportional to the given output preventing the dying ReLU problem. This ensures that all the neurons in the network contribute to the output even though they have negative inputs.

The WGAN-GP model employed the Adam optimizer, featuring a learning rate (α) of 0.0001, a decay rate for momentum (β_1) of 0 to prevent high weighting for the more recent gradients, and a decay rate for squared gradients (β_2) of 0.9 which helps to capture the long-term memory of the gradients thus a stable estimate of variance. The optimizer adjusts the model's weights per the gradient derived from

the loss function. The gradient penalty coefficient (λ) employed was 10 and the number of critic operations during each generator operation (n_{crit}) was selected as 5.

1000 synthetic data points were generated. The evaluation of the synthetic data involved a comparison with the original data set, focusing on the mean and standard deviations, as well as the minimum and maximum values of the variables. The Mann-Whitney U test was performed to assess the similarity between the two data sets. The null hypothesis was that the two samples were obtained from the same population. If the p-value obtained from the test is below the significance threshold of 0.05, it is permissible to reject the null hypothesis, thereby indicating the two data sets exhibit dissimilarity. The Mann-Whitney U test was implemented using the SciPy Python library statistical functions.

The effectiveness of the synthetic data was assessed by training the XGBoost model using the data generated by WGAN-GP and subsequently evaluating it against the original data set. The model performance solely using the original data was compared to the model utilizing the combined synthetic and original data.

3. Results and Discussion

3.1 Performance Evaluation of the Model

The validation technique employed was the train-test split method. This approach guarantees that the model demonstrates effective performance on data that it has not previously encountered, rather than solely on the training data set. The process consisted of randomly dividing the data set into two subsets, allocating 75% for model training and reserving 25% for testing. The XGBoost model performance evaluation metrics for the train and test data set are shown in Table 3.

Table 3. Performance Evaluation Metrics of the XGBoost Model for the Train and Test Set

Performance Metric	Train Set	Test Set
RMSE	1.887	3.351
R^2	0.992	0.939
MAE	1.420	2.994

The performance evaluation metrics of the training set are superior and of a good fit, as this was the data set on which the model was trained. There is not a huge difference between the train and test set values, which indicates low variance therefore the model is robust and can perform well on new unseen data, not just the data it was trained on. The low RMSE value indicates that the model can handle outliers well. RMSE penalizes large errors due to the squaring and aims to minimize the impact of outliers. The low MAE values suggest that the model demonstrates a good fit, indicating that the predictions are, on average, close to the actual values. The high R^2 value for both the training and testing phases indicates that the model effectively encapsulates the fundamental patterns present within the data.

For the unoptimized model, performance evaluation metrics RMSE, R^2, and MAE obtained were 5.505, 0.835, and 4.440 respectively. In contrast, the metrics obtained by optuna hyperparameter tuning were 3.351, 0.939, and 2.994. This indicated a great improvement in model performance by hyperparameter tuning. From the optuna hyperparameter search the ideal hyperparameter values were the number of trees (n_estimators) 541, subsample 0.993, max_depth 6, learning rate 0.07, lambda 2, gamma 1, colsample_bytree 0.2, and alpha 0.

3.2 Feature Importance Analysis

The summary plot of the shap values for the features is presented in Figure 2. The order of the features on the plot provides the ranking of the importance of the features to the model prediction; in this case, the elevated temperature has the highest effect while Fe_2O_3% has the least impact. Each feature sample is depicted as a dot with the color serving to denote its intensity. A red dot signifies a high feature value, whereas a blue dot represents a low feature value. The horizontal placement of the dots relative to the origin of the plot reflects the influence of various features on the model's output. Features located on the left side of the plot signify a detrimental effect on the predicted output, while those on the right

side denote a beneficial impact. Furthermore, the extent of deviation from the origin correlates with the magnitude of the feature's influence on the model's predictions.

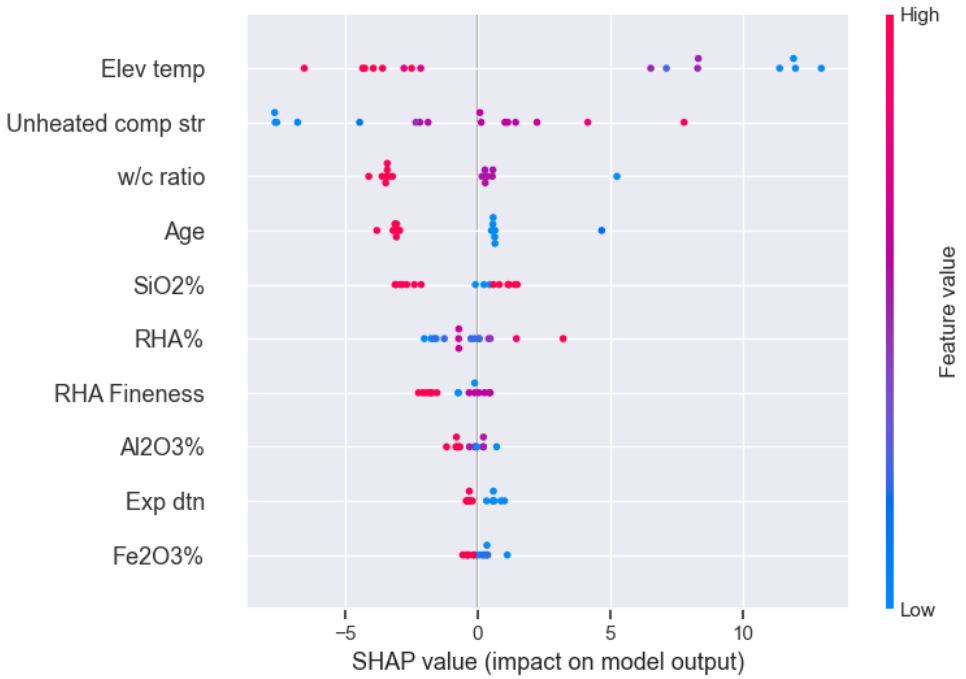

Figure 2. SHAP Values Summary Plot

Dividing the data set into two distinct subsets, one for training the model and the other for evaluating its predictive accuracy on novel observations, facilitated an assessment of the model's ability to generate accurate predictions with unfamiliar data. Consequently, this approach enables the model to effectively recognize patterns in unseen data, thereby enhancing its inferential capabilities. From the SHAP values summary plot of the model predictions on the test data, a high elevated temperature leads to a low predicted residual compressive strength and vice versa. The increase in temperature leads to the breakdown of calcium silicate hydrate (C-S-H) gel, which is responsible for the compressive strength of concrete. This decomposition results in the formation of cracks between aggregates and cement paste.

The unheated compressive strength is directly proportional to the residual compressive strength; as the former decreases, so does the latter. Concrete that initially exhibits low compressive strength is likely to demonstrate inadequate performance following exposure to high temperatures, resulting in diminished residual compressive strength.

The compressive strength of concrete incorporating rice husk ash is positively correlated with a reduction in the water-cement ratio. An elevated water-cement ratio is associated with a diminished anticipated residual compressive strength. Conversely, a lower water-cement ratio results in concrete that exhibits a greater amount of hydrated calcium silicate hydrate gel and cement particles. Therefore, when subjected to high temperatures, the dehydration of the concrete is relatively mild, allowing the material to preserve its structural integrity, resulting in only a minor decrease in compressive strength. However, care should be taken as a low water-cement ratio may cause concrete workability issues.

Concrete compressive strength increases with age. With an increase in age, there is an increase in the degree of hydration, improving concrete compressive strength. The hydration process is typically accelerated by initial exposure to high temperatures, resulting in a preliminary rise in compressive strength. Nevertheless, continued exposure to elevated temperatures subsequently diminishes the compressive strength. The compressive strength of aged concrete experiences a more significant decline following exposure to high temperatures which can be attributed to the concrete's microstructure becoming more complex with age and the increase in cement paste which is similar to what was reported by Seyam & Nemes [22]. Consequently, older concrete specimens are more susceptible to a significant reduction in compressive strength after being subjected to elevated temperatures therefore higher age leads to a lower predicted residual compressive strength by the model.

Higher silica content increases concrete compressive strength due to more formation of the calcium silicate hydrate gel. Elevated temperature causes the decomposition of the C-S-H gel thus reducing the concrete compressive strength. This phenomenon accounts for the higher predicted residual strength caused by high silica content. The lower the RHA% the lower the value of the predicted residual strength. This can be ascribed to the fact that the surface of RHA is multilayered, angular, and microporous, thereby after high-temperature exposure the concrete made with RHA is more compact than OPC concrete hence reduction in compressive strength is less severe.

Finer RHA leads to a lower predicted residual compressive strength. Cementitious material with fine particles has a greater surface area, thereby increasing the hydration process for forming C-S-H, causing increased compressive strength. However, in the case of alkali-aggregate, the finer the cementitious material the more aggressive the reaction between the two, expediting the hydration process and may cause shrinkage of the concrete, therefore prone to cracking and reduction in compressive strength.

High alumina content causes a lower predicted residual compressive strength. The alumina in concrete assists in lowering the setting time of concrete, speeding up the reactions but reducing strength. Alumina acts as a filler, improving the cement paste's microstructure by decreasing the pore volume and increasing the compressive strength of concrete. It also acts as an activator promoting pozzolanic reactions. Excessive alumina, beyond the amount necessary for reaction with lime during hydration, leads to the leaching out of silica, which in turn results in compressive strength reduction as observed by Nazari [23]. Longer exposure duration of concrete to elevated temperature causes a lower predicted residual strength due to greater decomposition of the C-S-H gel.

High iron oxide content leads to low predicted residual strength and vice versa. However, the effect is negligible. Iron oxide functions as a filler, occupying the voids present in the cement paste increasing density, and improving the microstructure. Iron oxide is also highly reactive, quickening the hydration process and causing early strength gain in the concrete. Alumina and iron oxide contribute less to the compressive strength of concrete than silica.

The most significant features influencing the model were the elevated temperature, unheated compressive strength, and the water-cement ratio. In the temperature range of 200^0C to 400^0C, the compressive strength of RHA concrete is minimally reduced, and the predicted residual compressive strength remains comparatively high. However, once the temperature exceeds 400^0C, the residual compressive strength significantly declines. The residual compressive strength of RHA concrete diminished as the water-cement ratio increased, with a notably more pronounced reduction occurring beyond the 0.55 mark. The features with the least effect on the model were the exposure duration, Al_2O_3%, and Fe_2O_3%.

3.3 Evaluation of Generated Synthetic Data

The generated synthetic data was like the original data as seen from the comparison of the features' means, standard deviations, minimum, and maximum values shown in Table 4. The results of the Mann-Whitney U test revealed p-values for all features that exceeded the significance threshold of 0.05, suggesting that the two data sets exhibit similarity.

The XGBoost model underwent training using synthetic data and was subsequently evaluated on the original data set yielding RMSE, R^2, and MAE values of 0.065, 0.945, and 0.053 respectively. These results indicate that the synthetic data effectively represented the underlying patterns present in the original data set.

The combined dataset of both the original and synthetic data with a train-test split of 75% (training) to 25% (testing) gave an RMSE value of 0.071 representing a significant enhancement from the original value of 3.351, an improved R^2 value of 0.941 from a previous initial value of 0.939, and an MAE value of 0.053 which is an improvement from 2.994. Therefore, incorporating the synthetic data led to a notable performance improvement of the model.

Table 4. Comparison of the Original and Synthetic Data Statistics

Feature	Mann Whitney U Test p Value	Mean		Standard Deviation		Min		Max	
		Orig	Syn	Orig	Syn	Orig	Syn	Orig	Syn
RHA%	0.547	0.339	0.345	0.280	0.284	0.000	0.000	1.000	1.029
SiO_2%	0.100	0.759	0.751	0.423	0.423	0.000	0.000	1.000	1.017
Al_2O_3%	0.846	0.428	0.417	0.338	0.330	0.000	0.000	1.000	1.021
Fe_2O_3%	0.732	0.542	0.524	0.433	0.427	0.000	0.000	1.000	1.022
RHA Fineness	0.524	0.586	0.574	0.369	0.365	0.000	0.000	1.000	1.021
w/c ratio	0.084	0.698	0.707	0.412	0.401	0.000	0.000	1.000	1.023
Unheated compressive strength	0.852	0.425	0.420	0.348	0.336	0.000	0.000	1.000	1.012
Elevated temperature	0.907	0.509	0.499	0.316	0.304	0.000	0.000	1.000	1.020
Exposure duration	0.128	0.688	0.676	0.467	0.468	0.000	0.000	1.000	1.022
Age	0.882	0.483	0.479	0.465	0.458	0.000	0.000	1.000	1.046
Residual Compressive strength	0.975	0.397	0.400	0.280	0.277	0.000	0.000	1.000	1.006

4. Conclusion

The XGBoost demonstrates a high level of accuracy in forecasting the residual compressive strength of RHA concrete following exposure to elevated temperatures. It saves on time and expenses associated with extensive laboratory experiments. To improve the performance of the XGBoost algorithm, focus should be placed on the learning rate, number of estimators, and lambda hyperparameters by widening their search space and using default values for the other hyperparameters.

The key factors influencing predictions made by the XGBoost model include the elevated temperature, the unheated compressive strength, and the ratio of water to cement. The exposure duration, alumina content, and iron oxide have minimal effect on the prediction. A case can be made for their exclusion from the model creation which may reduce computation time and improve model performance. Increasing the dataset by augmenting it with synthetic data created by WGAN-GP improved the performance of the XGBoost machine learning algorithm. Although WGAN-GP has primarily been applied to text and image augmentation, the current study has demonstrated its utility in continuous numerical data instances. This proves useful in cases where data is scarce and the required data may take a long time to collect, such as in the study of the creep behavior of concrete and durability.

Field studies should be conducted to get unregulated data different from controlled experimental data. Other than the generation of synthetic data through GANs, techniques for dimensionality reduction, including principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE), can be used to manage sparse datasets by facilitating the transformation of models into a lower-dimensional space.

References

[1] H. M. Owaid, R. B. Hamid, and M. R. Taha, "A Review of Sustainable Supplementary Cementitious Materials as an Alternative to All-Portland Cement Mortar and Concrete," *Australian Journal of Basic and Applied Sciences*, vol. 6, no. 9, pp. 287–303, 2012.

[2] A. Hossain, M. H. Rashid, O. U. Laz, and M. Rahman, "Effect of Rice Husk Ash on Concrete," in *WasteSafe 2011*, Khulna, Bangladesh, 2011.

[3] S. A. Zareei, F. Ameri, F. Dorostkar, and M. Ahmadi, "Rice husk ash as a partial replacement of cement in high strength concrete containing micro silica: Evaluating durability and mechanical properties," *Case Studies in Construction Materials*, vol. 7, no. 7, pp. 73–81, 2017.

[4] M.-H. Zhang and V. M. Malhotra, "High-Performance Concrete Incorporating Rice Husk Ash as a Supplementary Cementing Material," *ACI Materials Journal*, vol. 93, no. 6, pp. 629–636, 1996.

[5] A. Ahmad, K. A. Ostrowski, M. Maslak, F. Farooq, I. Mehmood, and A. Nafees, "Comparative Study of Supervised Machine Learning Algorithms for Predicting the Compressive Strength of Concrete at High Temperature," *Materials*, vol. 14, no. 4222, Jul. 2021.

[6] M. Ahmad *et al.*, "Supervised Learning Methods for Modeling Concrete Compressive Strength Prediction at High Temperature," *Materials*, vol. 14, no. 1983, Apr. 2021.

[7] M. Rahmati and V. Toufigh, "Evaluation of Geopolymer Concrete at High Temperatures: An Experimental Study using Machine Learning," *Journal of Cleaner Production*, vol. 372, 2022.

[8] C. Cakiroglu, "Explainable Data-Driven Ensemble Learning Models for the Mechanical Properties Prediction of Concrete Confined by Aramid Fiber-Reinforced Polymer Wraps Using Generative Adversarial Networks," *Appl. Sci.*, vol. 13, no. 11991, Nov. 2023.

[9] M. K. H. Al-Majidi, "Effects of Exposure to Elevated Temperatures on Properties of Concrete Containing Rice Husk Ash," University of Malaysia, 2011.

[10] R. Umasabor and J. O. Okovido, "Fire Resistance Evaluation of Rice Husk Ash Concrete," *Heliyon*, vol. 4, no. e01035, Dec. 2018.

[11] W. H. Wang, Y. F. Meng, and D. Z. Wang, "Effect of Rice Husk Ash on High-Temperature Mechanical Properties and Microstructure of Concrete," *Kemija U Industriji*, vol. 66, no. 3–4, pp. 157–164, 2017.

[12] H. Nguyen, T. Vu, T. P. Vo, and H.-T. Thai, "Efficient Machine Learning Models for Prediction of Concrete Strengths," *Construction and Building Materials*, vol. 266, Oct. 2020.

[13] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference*, San Francisco, California, USA: ACM, Aug. 2016.

[14] T. Nguyen-Sy, J. Wakim, Q.-D. To, M.-N. Vu, T.-D. Nguyen, and T.-T. Nguyen, "Predicting the Compressive Strength of Concrete from its Compositions and Age using the Extreme Gradient Boosting Method," *Construction and Building Materials*, vol. 260, 2020.

[15] S. Watanabe, "Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance," Germany, Tutorial paper arXiv:2304.11127v3 [cs.LG], May 2023.

[16] N.-H. Nguyen, J. Abellan-Garcia, S. Lee, E. Garcia-Castano, and T. P. Vo, "Efficient Estimating Compressive Strength of Ultra-High-Performance Concrete using XGBoost Model," *Journal of Building Engineering*, vol. 52, Mar. 2022.

[17] S. Lundberg and S. Lee, "A Unified Approach to Interpreting Model Predictions," *arXiv:1705.07874v2*, vol. [cs.AI], Nov. 2017.

[18] S. Lundberg, G. Erion, and S. Lee, "Consistent Individualized Feature Attribution for Tree Ensembles," *arXiv:1802.03888*, vol. [cs.LG], Mar. 2019.

[19] I. J. Goodfellow *et al.*, "Generative Adversarial Nets," presented at the Advances in Neural Information Processing Systems, Jun. 2014.

[20] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein Generative Adversarial Networks," presented at the International Conference on Machine Learning, Dec. 2017.

[21] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs," arXiv preprint, 2017. [Online]. Available: arXiv preprint arXiv:1704.00028

[22] A. M. Seyam and R. Nemes, "Age influence on compressive strength for concrete made with different types of aggregates after exposure to high temperatures," *Materials Today: Proceedings*, 2023.

[23] A. Nazari, S. Riahi, S. Riahi, S. F. Shamekhi, and A. Khademno, "Influence of Al2O3 nanoparticles on the compressive strength and workability of blended concrete," *Journal of American Science*, vol. 6, no. 5, pp. 6–9, 2010.