

# Deep learning-based modelling of polyvinyl chloride tube-confined concrete columns under different load eccentricities

Li Shang<sup>a</sup>, Haytham F. Isleem<sup>b,c,\*</sup>, Mostafa M. Alsaadawi<sup>d</sup>

<sup>a</sup> School of Civil and Hydraulic Engineering, Xichang University, Xichang, 615000, China

<sup>b</sup> Department of Computer Science, University of York, York, YO10 5DD, United Kingdom

<sup>c</sup> Jadara University Research Center, Jadara University, Irbid, Jordan

<sup>d</sup> Civil Engineering Department, Faculty of Engineering, Horus University-Egypt, New Damietta, 34517, Egypt

## ARTICLE INFO

### Keywords:

Polyvinyl chloride tube  
Confined concrete column  
Finite element method  
Machine learning  
Deep learning  
Linear regression  
Support vector regression  
Decision tree  
Random forest  
Gradient boosting  
Transformer-convolutional neural network  
Graphical user interface

## ABSTRACT

This study presents a deep learning-based framework for predicting the load-carrying capacity of polyvinyl chloride (PVC) tube-confined concrete columns under various loading conditions. A comprehensive dataset of 200 samples was generated using finite element modeling, incorporating key parameters such as PVC tube thickness, concrete strength, and load eccentricity. Several machine learning algorithms, including Linear Regression (LR), Support Vector Regression (SVR), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), and a novel hybrid Transformer-Convolutional Neural Network (Transformer-CNN) model, were employed for the prediction task. The results demonstrate that the proposed Transformer-CNN model outperforms traditional methods, achieving the lowest root mean squared error of 27.15 kN and the highest coefficient of determination value of 0.9875. The model's robustness was further validated using cross-validation techniques, ensuring its reliability for practical applications. To facilitate usability, a Python-based graphical user interface (GUI) was developed, enabling engineers to apply the model efficiently in real-world scenarios. This study highlights the potential of deep learning in advancing the design and analysis of PVC-confined concrete columns, offering a more accurate and efficient alternative to conventional methods.

## 1. Introductions

The integration of various materials into concrete has been a focal point in enhancing the mechanical properties and durability of infrastructure. Traditionally, steel tubes have been utilized for their high strength and rigidity, particularly in the construction of concrete-filled steel tubes (CFSTs). These structures are valued for combining the compressive strength of concrete with the tensile strength of steel, offering superior mechanical properties and ease of construction, especially in high-rise buildings and long-span bridges (Xu et al., 2021; Eilbeigi et al., 2022). However, steel's susceptibility to corrosion and high maintenance costs poses significant challenges, particularly in harsh environmental conditions (Han et al., 2014; Fakharifar and Chen, 2016).

To address these issues, alternatives such as Fiber Reinforced Polymer (FRP) tubes and polyvinyl chloride (PVC) have been explored. FRP tubes offer high strength and corrosion resistance but are limited by their brittleness and higher costs, complicating their adoption in certain

applications (Ozbakkaloglu and Xie, 2016; Li et al., 2018; Jiang et al., 2019; Berg et al., 2006; Wang and Yang, 2012). Conversely, PVC provides a compelling alternative due to its superior corrosion resistance, lower cost, and enhanced performance in seismic conditions. PVC's non-metallic, thermoplastic nature, comprising a blend of polymer resin, lubricants, plasticizers, stabilizers, fillers, and pigments, allows for customization to meet specific needs (Wu et al., 2021; Raheemah and Resan, 2019). This material substitution not only mitigates corrosion-related issues but also reduces maintenance costs, making PVC a viable and economical choice for large-scale infrastructure projects.

The incorporation of hollow tubes in concrete structures has been a significant development in enhancing the mechanical properties of these materials. This method, which includes using tubes like PVC to confine concrete, provides notable benefits such as increased strength, stiffness, and ductility. These advantages are particularly valuable in densely built urban environments where reducing the size of structural elements without compromising performance is crucial.

\* Corresponding author. Department of computer Science, University of York, York, YO10 5DD, United Kingdom.

E-mail addresses: [1058451679@qq.com](mailto:1058451679@qq.com) (L. Shang), [mps565@york.ac.uk](mailto:mps565@york.ac.uk) (H.F. Isleem), [mostafa.m.alsaadawi@gmail.com](mailto:mostafa.m.alsaadawi@gmail.com) (M.M. Alsaadawi).

<https://doi.org/10.1016/j.engappai.2025.110217>

Received 12 October 2024; Received in revised form 23 January 2025; Accepted 2 February 2025

Available online 13 February 2025

0952-1976/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

The use of PVC tubes, in particular, has garnered attention due to their ability to synergistically interact with the concrete core. This interaction creates a composite material that exhibits enhanced ultimate strength, surpassing the combined strengths of the individual components. This improvement is primarily attributed to the triaxial stress state induced within the concrete core by the confining effect of the PVC tube, which significantly enhances the longitudinal compressive strength of the concrete (Isleem et al., 2022a; Abbas, 2023).

Over the past four decades, extensive research has focused on assessing the static performance of PVC tube-confined concrete. Findings have demonstrated that increasing the thickness of PVC tubes can substantially enhance the fracture energy of the composite material. Furthermore, studies have shown that reinforced concrete-filled PVC tubes maintain their structural integrity and ductility even under harsh environmental conditions, such as prolonged exposure to seawater, highlighting PVC's suitability for use in marine and saline environments (Gupta and Verma, 2016). This durability and resistance to environmental degradation make PVC a promising alternative to traditional materials like steel, particularly in applications requiring long-term durability and low maintenance. Moreover, the use of PVC tubes significantly enhances the ultimate strength and lateral resistance of concrete cores. The confinement provided by the PVC tube not only increases the longitudinal compressive strength of the concrete but also helps manage the stress distributions within the structure. This improved stress management leads to better overall performance of the PVC-concrete composite system under various loading conditions (Sharma et al., 2022; Abdulla, 2017; Isleem et al., 2024a). Further studies have also explored the influence of geometric factors on the performance of PVC-concrete columns under axial loads. The thickness-to-diameter ratio of PVC tubes is a critical parameter, with lower ratios being more susceptible to failure, while higher ratios tend to result in more ductile failure modes, such as bulging and buckling, rather than rupture (Sharma et al., 2022). Additionally, the slenderness ratio, defined by the column's height-to-diameter ratio, significantly affects failure modes; columns with low slenderness ratios may experience sudden fractures in the PVC tubes, while those with higher slenderness ratios are more likely to develop shear cracks (Osman and Soliman, 2015). These findings underscore the importance of considering both loading conditions and geometric design in the structural analysis and design of PVC-confined concrete columns.

Most existing research has focused on the behavior of PVC-concrete columns under concentric axial loads, a scenario that, while simplifying the analysis, does not fully represent the conditions these structures encounter in practice. In reality, concrete columns often experience eccentric loads, which can introduce significant bending moments and lead to instability, especially in slender columns (Christianto et al., 2019). The introduction of eccentricity alters the stress and strain distribution within the column, potentially diminishing the effectiveness of the confinement provided by the PVC tube. According to Eurocode 4, to maintain the beneficial effects of confinement, the ratio of load eccentricity to the outer diameter ( $e/d$ ) should be less than 0.1 for circular sections (Piscesa et al., 2017). Exceeding this limit can compromise the structural integrity and effectiveness of the confinement, particularly under high eccentric loading conditions.

Given the practical difficulties in obtaining experimental data under large eccentric load conditions, numerical methods have become invaluable in predicting the behavior of PVC-concrete columns. Using advanced finite element software like Abaqus, researchers can simulate various loading conditions and assess the structural response of these columns with high accuracy. The concrete damage plasticity model is typically employed to represent the behavior of the concrete under compression, while the Von Mises model is used for the PVC tube, capturing the material's response to stress and strain (Isleem et al., 2024a). These numerical models are crucial for understanding the failure mechanisms, load-carrying capacities, and overall structural performance of PVC-concrete columns, providing insights that guide the

design and optimization of such systems.

Several scholars have explored methods for predicting the bearing capacity or strength of PVC-concrete columns, primarily through experimental research and theoretical analysis (Sharma et al., 2022; Saadon and Jasim, 2017). However, these existing models predominantly focus on the behavior of columns under axial compression with concentric loading. As a result, they tend to be conservative, lacking the flexibility to accommodate the effects of varying loading conditions. This conservativeness stems from the models' failure to consider the complexities introduced by eccentric loading. Consequently, there is a pressing need for the development of more sophisticated and reliable models that can accurately predict the axial compressive load-carrying capacity of PVC-concrete columns under a range of eccentric loading conditions. Addressing this gap is crucial for enhancing the precision and applicability of these predictive models in practical engineering scenarios.

Machine learning (ML) has proven highly effective in addressing various structural engineering challenges, including structural health monitoring, damage detection, performance evaluation, and structural parameter identification (Wang et al., 2024; Cury and Cr mona, 2012; Shahin et al., 2023, 2024; Zhang and Burton, 2019; Elshaarawy et al., 2024; Feng et al., 2020). These techniques have also shown significant success in estimating the compressive strength of composite columns. The development of predictive models for these systems has increasingly incorporated advanced computational techniques. Specifically, ML algorithms such as Artificial Neural Networks (ANN) and Support Vector Regression (SVR) have been employed to predict key structural behaviors, including the compressive strength and stress-strain response of composite columns (Isleem et al., 2022b, 2022c, 2023; Naderpour et al., 2010; Abdulla, 2020; Oreta and Kawashima, 2003; Chen et al., 2022; Hamed et al., 2025). Many studies have employed ML techniques to estimate the axial load capacity of CFST (Asteris et al., 2021a, 2021b, 2024; Bardhan et al., 2022; Liao et al., 2021; Isleem et al., 2024b; Wang and Chan, 2023; Zhou et al., 2023; Lyu et al., 2021; Zarringol et al., 2023; Ma et al., 2022; Xu et al., 2024a).

In recent years, ML techniques have advanced further, leveraging sophisticated models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and Graph Neural Networks (GNNs) to predict the behavior of concrete columns. For instance, LSTM networks have been integrated with Kolmogorov–Arnold Networks (KANs) to predict the deformation of concrete dams, demonstrating their potential in capturing complex temporal dependencies in structural behavior (Xu et al., 2024b). Hybrid models have also gained attention in structural engineering due to their ability to combine the strengths of different algorithms. For example, a hybrid stacked machine learning model that integrates SVM, Decision Trees (DT), Random Forests (RF), Gradient Boosting (GB), and Extreme Gradient Boosting (XGBoost) has been developed to predict the compressive strength of high-performance concrete (Tipu et al., 2023). These ML-based models offer superior accuracy and versatility over traditional analytical methods, particularly in managing complex, multi-factorial problems without the need for simplifying assumptions.

## 2. Research significance

In recent studies, existing analytical models for predicting the load-carrying capacity of PVC-concrete columns have been found to lack accuracy, primarily due to their oversimplified assumptions and limited consideration of variables. While analytical models have been developed for PVC-concrete columns, they predominantly focus on concentric loading conditions, with limited attention given to eccentric loading scenarios. These conditions introduce additional complexities that analytical models often fail to address, such as nonlinear stress distributions and the interaction effects between PVC tubes and concrete cores. The scarcity of analytical models tailored for eccentric loading highlights a critical gap in the existing research. To address these

limitations, precise models that account for all relevant factors are essential for accurate predictions. While ML approaches have demonstrated promising results across various disciplines, their application in estimating the load-carrying capacity of PVC-concrete columns, particularly under different eccentric loading conditions, remains unexplored. This gap highlights the necessity for further investigation and refinement of ML techniques in this domain.

In this research, a comprehensive dataset consisting of 200 samples was generated using finite element modeling. The dataset includes critical features influencing the load-carrying capacity of composite columns, such as the thickness of the PVC tube, tensile strength of the PVC tube, standard cylinder concrete strength, concrete core diameter, specimen height, longitudinal and transverse reinforcement ratios, and eccentricity. To develop a robust predictive model, several ML algorithms were employed, including Linear Regression (LR), SVR, DT, RF, GB, and a newly proposed hybrid Transformer-Convolutional Neural Network (CNN) model. The ultimate goal is to create practical ML tools that can be readily applied in the field, reducing the need for costly and time-consuming experimental testing.

### 3. Methodology

The methodology adopted in this study encompasses several critical steps, starting with data collection and preprocessing. This is followed by the design and implementation of the hybrid Transformer-CNN model, which is then trained and validated using a rigorous cross-validation approach to ensure reliable performance.

#### 3.1. Data collection and preprocessing

In this study, the dataset comprises 200 samples produced using finite element modeling by Isleem et al., as detailed in reference (Isleem et al., 2024a). Appendix A provides the specific characteristics of PVC-concrete columns under different eccentric loading conditions. These data were used to develop an efficient and robust predictive deep learning model for predicting the load carrying capacity of composite columns. The dataset includes several critical features that influence the load carrying capacity, denoted as  $Y$ . These features, designated as  $X1$  through  $X8$ , are as follows:

- **X1:** Thickness of PVC tube (mm)
- **X2:** Tensile strength of PVC tube (MPa)
- **X3:** Standard cylinder concrete strength (MPa)
- **X4:** Concrete core diameter (mm)
- **X5:** Height of the tested specimen (mm)
- **X6:** Longitudinal reinforcement ratio multiplied by the tensile strength of the longitudinal steel rebars (MPa)
- **X7:** Transverse reinforcement ratio multiplied by the tensile strength of the transverse steel rebars (MPa)
- **X8:** Eccentricity (mm)

The target variable  $Y$  is the load carrying capacity ( $P_{cc}$ ) measured in

kilonewtons (kN).

The statistical descriptive analysis, as summarized in Table 1, provided comprehensive insights into the dataset's central tendency, dispersion, and overall distribution. This analysis is critical for understanding the characteristics of the data and identifying any anomalies or outliers that could impact the performance of the predictive model.

The mean values of the features indicate the central tendency, with  $X1$  averaging 3.54 mm and  $X2$  averaging 46.25 MPa. The  $Y$  has a mean of 352.5 kN, indicating a wide range of capacities among the samples.

The standard deviation values highlight the variability in the data. For instance,  $X5$  exhibits significant variability with a standard deviation of 293.89 mm, reflecting a wide range of specimen heights in the dataset. Similarly, the  $Y$  also shows high variability, with a standard deviation of 255.20 kN.

The minimum and maximum values provide the range of the data.  $X4$  ranges from 58.00 mm to 184.40 mm, while  $X8$  varies from 0.00 mm to 50.00 mm. The  $Y$  spans from 27.71 kN to 1210.56 kN, indicating a broad range of structural capacities.

The skewness and kurtosis values offer insights into the distribution shape. Most features exhibit skewness values close to zero, suggesting that the data is fairly symmetrical. However, features such as  $X1$  and  $X5$  have skewness values of 1.71 and 0.82, respectively, indicating a right-skewed distribution. Kurtosis values close to zero indicate that most features have a relatively normal distribution, except for  $X1$  and  $X3$ , which have higher kurtosis values, indicating a more peaked distribution.

The interquartile range (IQR), which measures the spread of the middle 50% of the data, highlights the dispersion of each feature. For instance,  $X4$  has an IQR of 58.60 mm, suggesting significant variability within the middle range of concrete core diameters. The IQR for the load carrying capacity ( $Y$ ) is 409.34 kN, reflecting substantial variability in the structural capacities of the samples.

Following the descriptive analysis, the dataset underwent normalization to scale the input features. This process, known as standard scaling, transforms the features such that they have a mean of zero and a standard deviation of one. Normalization is essential in deep learning models as it ensures that the features contribute equally to the model's learning process and prevents any single feature from dominating due to its scale.

To visualize the data distribution, a histogram of normality tests for the entire dataset was generated, as shown in Fig. 1. This histogram helps in assessing whether the data follows a Gaussian distribution, which is often a desirable property for many machine learning algorithms. The plots for each feature ( $X1$  to  $X8$ ) and the target variable  $Y$  are displayed, providing insights into their individual distributions.

The statistical descriptive analysis, as summarized in Table 1, complements these visual findings by offering detailed numerical insights into each feature's distribution. For instance, the histogram for  $X1$  reveals a right-skewed distribution, which is supported by the skewness value of 1.71 in the statistical analysis. Similarly,  $X3$  shows a moderate skewness of 0.81 and a peaked distribution indicated by a kurtosis of 1.95, both reflected in the histogram.

**Table 1**  
Statistical descriptive analysis of the dataset.

Variable	Mean	Standard Deviation	Minimum	Median	Maximum	Variance	Skewness	Kurtosis	1st Quartile (Q1)	3rd Quartile (Q3)	Interquartile Range (Q3 - Q1)
$X1$	3.54	1.35	2.00	3.00	7.80	1.81	1.71	2.77	3.00	3.90	0.90
$X2$	46.25	7.38	33.40	49.74	62.00	54.41	0.13	-0.41	39.79	49.74	9.95
$X3$	23.00	8.14	10.50	25.04	51.50	66.34	0.81	1.95	16.89	28.50	11.61
$X4$	123.4	36.64	58.00	134.00	184.40	1342.37	-0.45	-1.00	94.00	152.60	58.60
$X5$	441	293.89	126.00	280.00	1000.00	86371.21	0.82	-0.62	220.00	544.00	324.00
$X6$	4.06	5.93	0.00	0.00	13.48	35.22	0.81	-1.32	0.00	12.42	12.42
$X7$	1.39	2.07	0.00	0.00	4.98	4.30	0.87	-1.22	0.00	4.31	4.31
$X8$	25.80	16.54	0.00	30.00	50.00	273.73	-0.33	-1.00	20.00	40.00	20.00
$Y$	352.5	255.20	27.71	287.33	1210.56	65125.17	0.74	-0.18	129.11	538.46	409.34

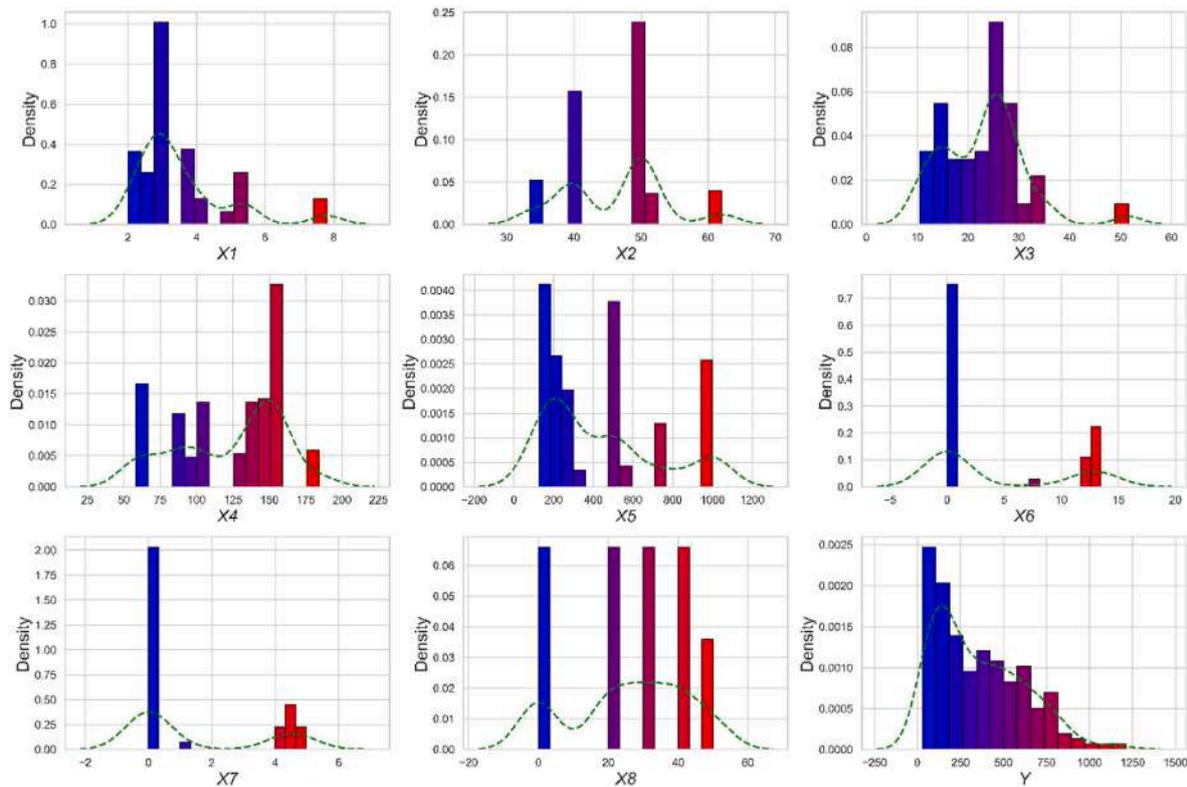


Fig. 1. Histogram showing the distribution of each feature and the target variable.

The histogram for X5 indicates a high degree of variability, which is consistent with the large standard deviation of 293.89 mm and a wide IQR of 324.00 mm observed in the statistical analysis. This high variability is also evident in the histogram's spread and the presence of multiple peaks.

The target variable Y shows a right-skewed distribution in the histogram, with a skewness of 0.74. The wide range of values, from 27.71 kN to 1210.56 kN, and the large standard deviation of 255.20 kN are visually confirmed by the histogram's long tail to the right.

Features such as X6 and X7 exhibit significant skewness and kurtosis values in the statistical table, which are mirrored in their respective histograms. X6 has a skewness of 0.81 and a kurtosis of  $-1.32$ , indicating a right-skewed but less peaked distribution, while X7 shows similar characteristics with a skewness of 0.87 and a kurtosis of  $-1.22$ .

To further understand the interaction between the features and the target variable Y, 3D histograms were generated to display the frequency distribution of the features in relation to Y.

Fig. 2 presents the 3D histogram displaying the frequency distribution of features X1 to X8 with the target variable Y. Each subplot highlights how different ranges of the feature values correlate with the load carrying capacity:

- **X1**: The histogram shows a concentration of higher counts in lower Y values, indicating that thinner PVC tubes are more common in the samples with lower load carrying capacities.
- **X2**: There is a noticeable peak where the tensile strength is around 45–50 MPa, correlating with a mid-range Y. This suggests a significant number of samples within this strength range and their corresponding load capacities.
- **X3**: The distribution shows higher counts at lower to mid-range concrete strengths, with a gradual decrease as the strength increases.
- **X4**: Most samples fall within the lower to mid-range diameters, with a higher count observed for specimens with smaller diameters.
- **X5**: There is a concentration of counts at lower heights, with these samples predominantly having lower Y values. This suggests that

shorter specimens are more common in the dataset and are associated with lower load carrying capacities.

- **X6**: The distribution indicates a higher frequency of samples with low to moderate reinforcement ratios, correlating with a wide range of Y values, but with a noticeable concentration at lower capacities.
- **X7**: Similar to X6, this feature shows a concentration of samples with lower reinforcement ratios, primarily associated with lower to moderate load capacities.
- **X8**: The histogram shows that most samples have low to moderate eccentricity, with higher frequencies at lower Y values.

Through careful data collection and preprocessing, study have established a robust foundation for developing a high-performing predictive model. The normalized and well-understood dataset is now ready for the subsequent phases of model architecture design and training, ensuring that the deep learning model can achieve optimal results in predicting the load carrying capacity of composite columns.

Fig. 3 presents the pair plot matrix of all variables (X1 to X8 and the target variable Y). This matrix is instrumental in visually examining the relationships and potential correlations between each pair of variables.

The diagonal plots show the distribution of each individual feature, indicating their spread and central tendency. For instance, the distribution of X1 shows a skewed pattern, aligning with previous observations from the statistical analysis.

The off-diagonal scatter plots provide a visual representation of the relationships between pairs of variables. Each scatter plot helps identify linear and non-linear relationships, clusters, and potential outliers. For example, there is a clear positive linear relationship between X4 and X5, as well as between X4 and Y. These relationships are indicated by the upward trend in the scatter plots.

The scatter plot between X3 and Y shows a moderate positive correlation, with data points forming a loosely aligned pattern along the positive slope. Similarly, X6 and X7 exhibit a clustered pattern, suggesting certain values of reinforcement ratios are more common.

In contrast, variables such as X2 and X7 show no clear pattern,

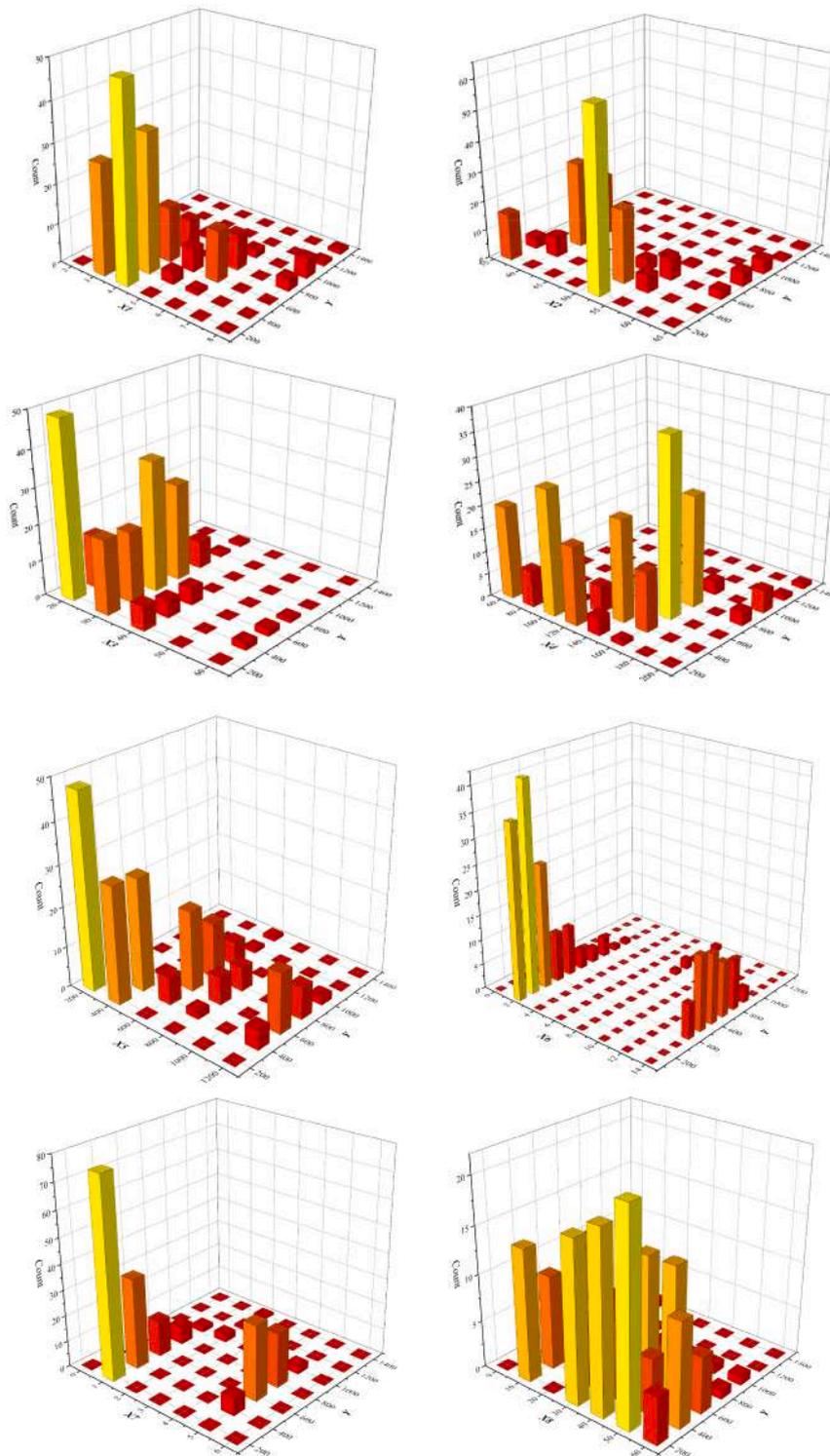


Fig. 2. 3D histogram displaying the frequency distribution of features X1-X4 with Y.

indicating a lack of strong correlation. The distribution and scatter plots for X8 reveal a more scattered relationship with Y, suggesting that higher eccentricities do not strongly predict the load carrying capacity.

Principal Component Analysis (PCA) (Abdi and Williams, 2010) is a dimensionality reduction technique that transforms a large set of variables into a smaller one that still contains most of the information in the large set. PCA achieves this by identifying the principal components, which are the directions of maximum variance in the dataset. These principal components are orthogonal to each other, ensuring that the

new features (principal components) are uncorrelated.

Fig. 4 illustrates the PCA plot, where the first three principal components (PC1, PC2, and PC3) are shown. The red dots represent the scores of the samples on the principal components, and the blue arrows represent the loadings, which indicate the direction and magnitude of each original feature's contribution to the principal components. The 95% confidence ellipse provides a visual boundary within which most of the data points are expected to lie, indicating the overall data distribution in the reduced dimension space.

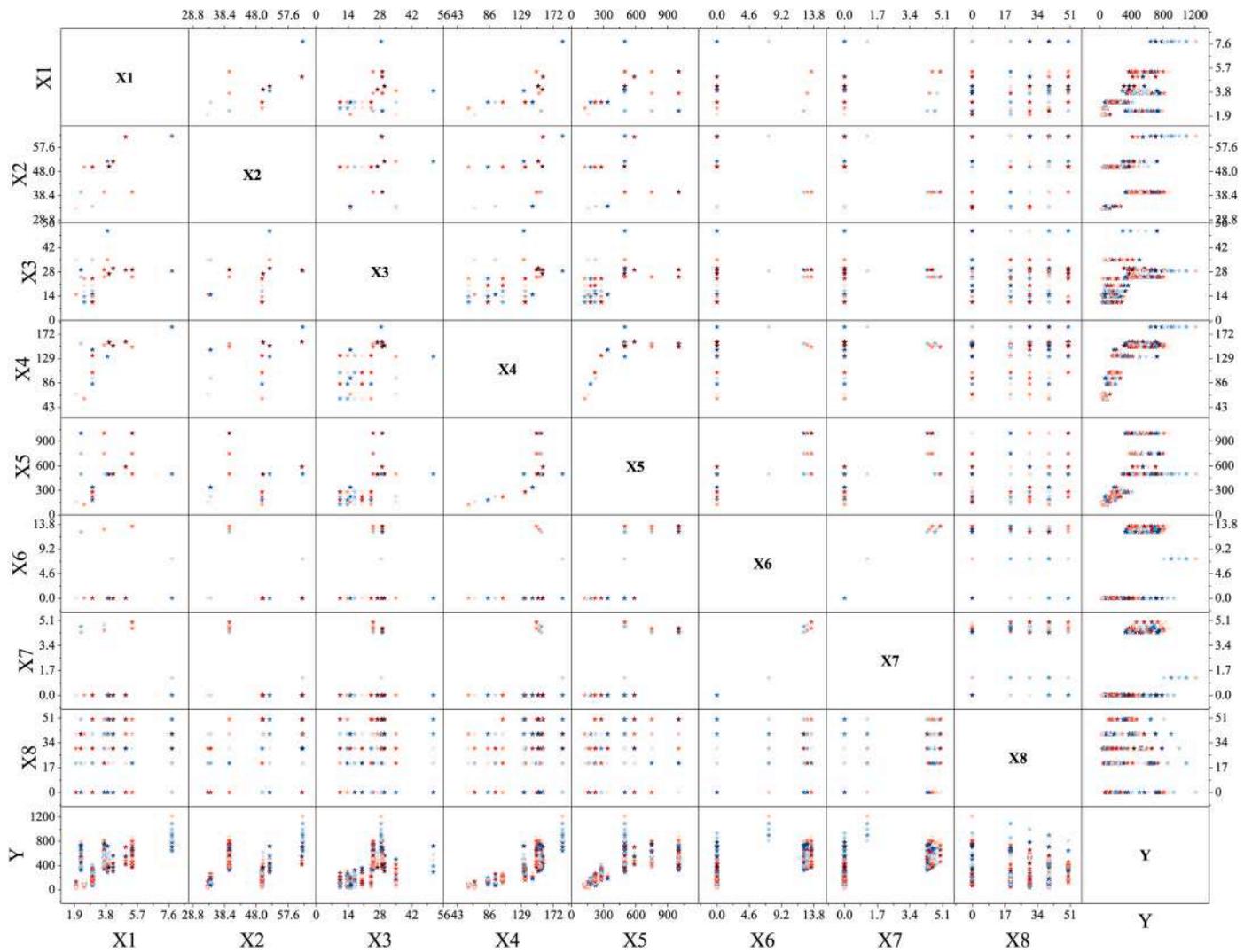


Fig. 3. Pair plot matrix of correlation of all variables.

From the PCA plot, it can observe that certain features, such as X8 and X5, have longer loading vectors, suggesting they contribute significantly to the variation captured by the principal components. In contrast, features with shorter loading vectors contribute less to the overall variance.

The PCA plot is valuable for visualizing the underlying structure of the data, identifying patterns, and detecting potential outliers. By reducing the dimensionality, PCA simplifies the complexity of the data while retaining the essential information, which is crucial for efficient and effective predictive modeling.

### 3.2. Correlation analysis

The Pearson correlation coefficient (Pearson, 1920), which ranges between  $-1$  and  $1$ , measures the linear correlation between two variables. A positive number indicates a positive correlation, while a negative number indicates a negative correlation. The closer the absolute value is to  $1$ , the stronger the linear relationship, assuming the normality is significant. An absolute value of  $0$  denotes no linear relationship, while an absolute value of  $1$  represents a perfect linear correlation. For the Pearson correlation coefficient to be accurate, the variables must follow a Gaussian distribution.

The Pearson correlation coefficient between variable  $X$  and variable  $Y$  is calculated using Eq. (1).

$$r_{xy} = \text{Corr}(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X) \cdot \text{Var}(Y)}} = \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{X})^2 \cdot \sum_{i=1}^n (y_i - \bar{Y})^2}} \quad (1)$$

where  $\text{Cov}(X, Y)$  is the covariance of  $X$  and  $Y$ ;  $\text{Var}(X)$  and  $\text{Var}(Y)$  are the variances of  $X$  and  $Y$ , respectively; and  $\bar{X}$  and  $\bar{Y}$  are the means of  $X$  and  $Y$ , respectively.

In contrast, Spearman's rank correlation coefficient (Zar, 2005) evaluates the correlation between two statistical variables using a monotonic equation. If there are no repeated values in the data, the two variables are perfectly monotonically positively or negatively correlated, making the Spearman's rank correlation coefficient equal to  $+1$  or  $-1$ . This coefficient is parameter-free and relies solely on the ranking of values rather than their specific magnitudes. Spearman's rank correlation coefficient between variable  $X$  and variable  $Y$  can be expressed using Eq. (2):

$$\rho_{xy} = \frac{\sum_{i=1}^n (R(x_i) - R(\bar{X}))(R(y_i) - R(\bar{Y}))}{\sqrt{\sum_{i=1}^n (R(x_i) - R(\bar{X}))^2 \sum_{i=1}^n (R(y_i) - R(\bar{Y}))^2}} = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)} \quad (2)$$

where  $R(x_i)$  and  $R(y_i)$  are the rank orders of the variables  $x_i$  and  $y_i$ ,

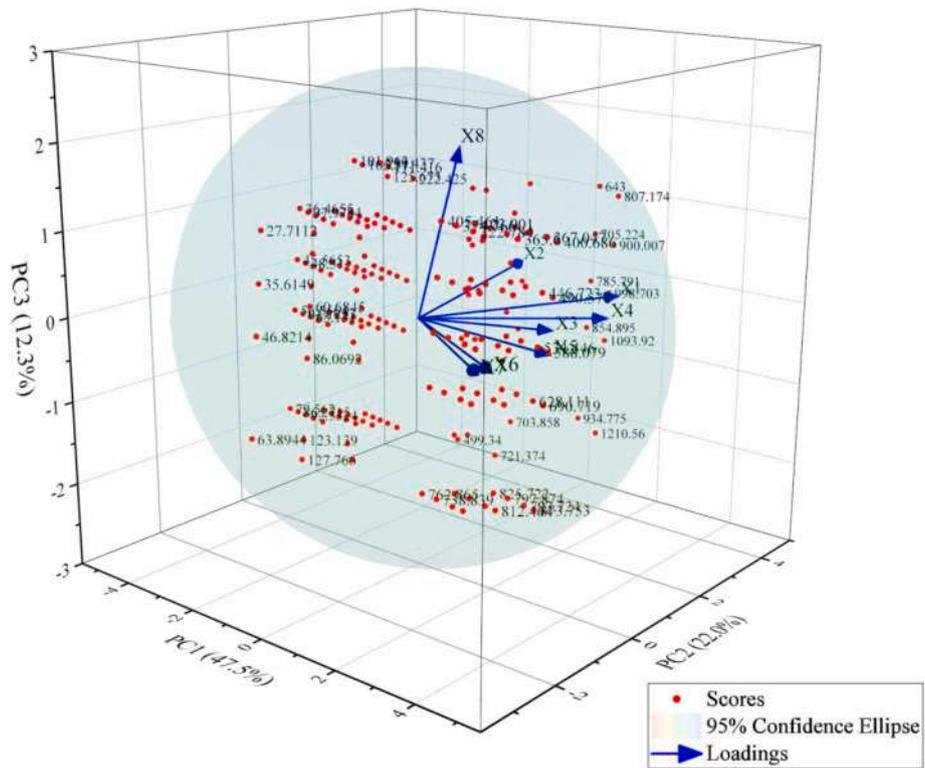


Fig. 4. Principal component analysis (PCA) plot.

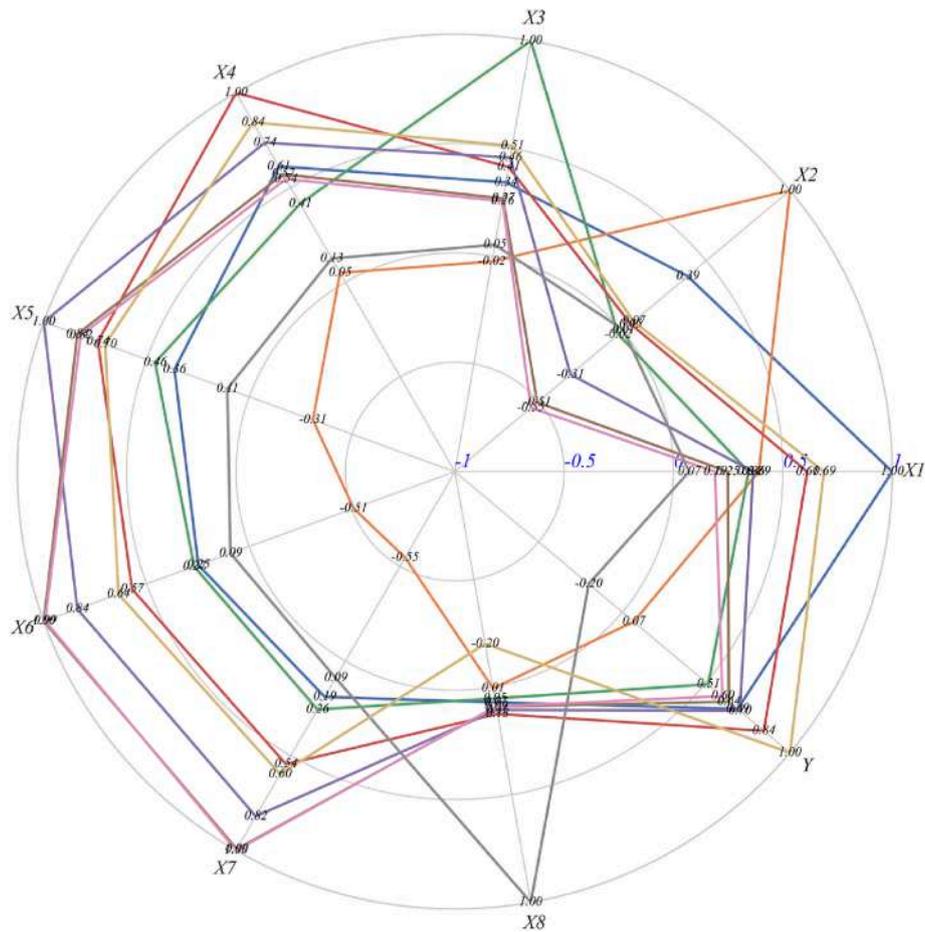


Fig. 5. Polar Pearson correlation coefficient heatmap.

respectively;  $R(\bar{X})$  and  $R(\bar{Y})$  are the average rank orders of the variables  $x_i$  and  $y_i$ , respectively;  $n$  is the total number of variables; and  $d_i = |R(x_i) - R(y_i)|$ .

Fig. 5 illustrates the Polar Pearson correlation coefficient heatmap, a visual representation of the Pearson correlation coefficients between each pair of variables. The plot uses a polar coordinate system to display the strength and direction of the linear relationships between variables.

In the polar heatmap, each axis represents a different variable, and the lines connecting the variables indicate the Pearson correlation coefficients. The values are plotted along the radii, where positive correlations are shown closer to the outer edge, and negative correlations are plotted towards the center. The magnitude of the correlation is indicated by the distance from the origin: the further from the center, the stronger the correlation.

Several key observations can be made from the polar heatmap. Strong positive correlations are evident between variables such as  $X4$  and  $X5$ , as well as  $X4$  and  $Y$ , which align with the observations from previous analyses. These variables have lines that extend far from the center, indicating a strong positive linear relationship.

Conversely, negative correlations are represented by lines closer to the center. For instance,  $X6$  shows a negative correlation with  $X2$ , indicating an inverse relationship between these variables.

The polar Pearson correlation coefficient heatmap provides a clear and concise way to visualize and interpret the linear relationships between multiple variables simultaneously. This visualization is particularly useful for identifying which variables are most strongly associated with the target variable, guiding the feature selection process for building predictive models.

Fig. 6 illustrates the Spearman rank correlation heatmap of all the variables ( $X1$  to  $X8$  and the target variable  $Y$ ). This heatmap is crucial for understanding the monotonic relationships between the features and the target variable.

The heatmap reveals several key insights. Strong positive correlations are observed between  $X4$  and  $X5$  with  $Y$ , showing correlation coefficients of 0.89 and 0.87, respectively. This indicates that as the concrete core diameter and specimen height increase, the load carrying capacity tends to increase as well. Additionally,  $X6$  and  $X3$  display moderate positive correlations with  $Y$ , with coefficients of 0.65 and 0.64, respectively, suggesting that higher reinforcement ratios and

concrete strengths are generally associated with higher load carrying capacities.

The analysis also highlights negative correlations, particularly between  $X6$  and  $X7$  with  $X2$ , having coefficients of  $-0.52$  each. These negative relationships indicate that higher reinforcement ratios correspond to lower tensile strengths of the PVC tube. Furthermore,  $X8$  shows a slight negative correlation with  $Y$  ( $-0.16$ ), suggesting that higher eccentricities might slightly decrease the load carrying capacity.

#### 4. Methodology

The overall methodology of this study encompasses several key steps to develop an efficient and robust predictive model for estimating the load carrying capacity of composite columns. The process begins with data collection and preprocessing, followed by the design and implementation of a sophisticated hybrid Transformer-CNNs model. The model is then trained and validated using a robust cross-validation approach, and its performance is evaluated. Hyperparameter tuning is performed to optimize the model, and finally, the model is deployed for practical applications.

##### 4.1. Model architecture design

The proposed predictive model for estimating the load carrying capacity of composite columns employs a sophisticated hybrid architecture that combines CNNs (Gu et al., 2018; Li et al., 2022; O'Shea and Nash, 2015; Vedaldi and Lenc, 2014) and Transformer models. This hybrid approach is designed to leverage the strengths of both CNNs for effective local feature extraction and Transformers for capturing complex dependencies between features. The following sections provide a detailed explanation of each component of the model architecture, including the CNN feature extractor, Transformer encoder, fully connected layers, and the output layer.

##### 4.1.1. Detailed architecture of the hybrid Transformer-CNN model

The architecture of the proposed predictive model for estimating the load carrying capacity of composite columns is a sophisticated hybrid design that integrates CNNs and Transformer models. This hybrid architecture leverages the unique strengths of both CNNs and

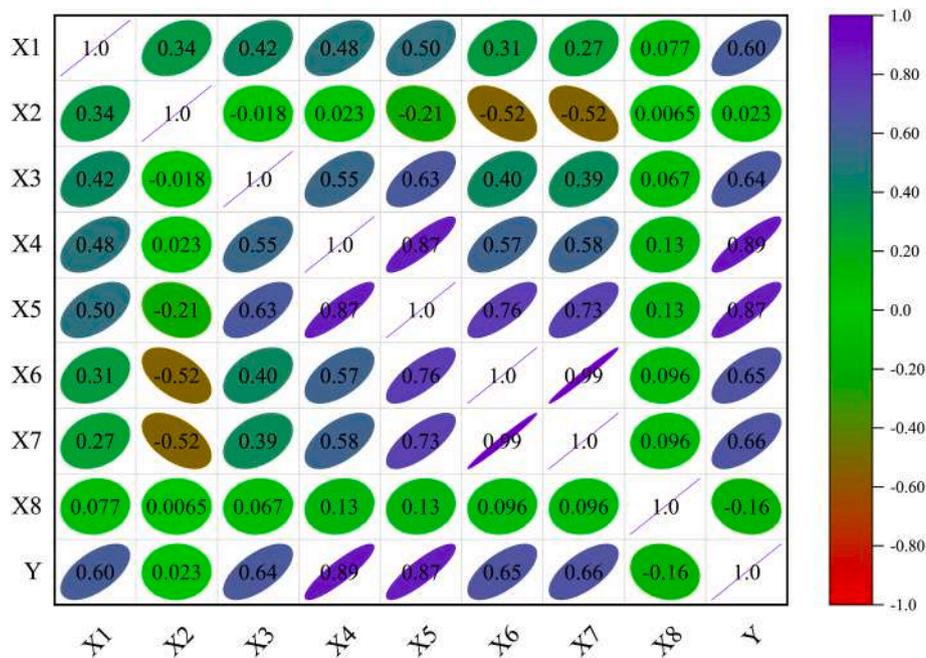


Fig. 6. Spearman rank correlation heatmap of all the variables.

Transformers to achieve accurate and reliable predictions. Here, we provide a detailed explanation of each component of the model, describing how they work together to process the input data and generate the final prediction. The architecture integrates CNNs to extract local patterns from the input features and Transformers to model long-range dependencies and interactions. The main components of the model are depicted in Fig. 7, which illustrates the flow from the input layer through the CNN and Transformer components to the fully connected layers and the output layer.

The CNN feature extractor is the first major component of the model. It begins with a convolutional layer that employs 32 filters with a kernel size of 3x3. This layer scans through the input features to extract local patterns, a process that is crucial for understanding the detailed structural properties of the columns. The ReLU (Rectified Linear Unit) activation function (Dahl et al., 2013) is applied here to introduce non-linearity into the model, enabling it to learn more complex patterns. The mathematical expression for the ReLU function is given by Eq. (3).

$$\text{ReLU}(x) = \max(0, x) \quad (3)$$

Following this, a **MaxPooling** layer with a pool size of 2x2 is used to reduce the dimensionality of the feature maps. This layer effectively down-samples the input representation, preserving the most significant information while reducing the computational complexity. This is followed by a second convolutional layer with 64 filters and a 3x3 kernel size, further refining the extracted features. Again, the ReLU activation function is applied to introduce non-linearity. Another **MaxPooling** layer with a 2x2 pool size is then applied, further reducing the dimensionality of the feature maps. The output of these layers is a set of 2D feature maps, which are then flattened into a 1D feature vector, making them suitable for input into the subsequent Transformer encoder. The 2D feature maps produced by these layers are then flattened into a 1D feature vector, as illustrated in Table 2.

Next, the Transformer encoder processes the flattened feature vector. This component begins with positional encoding, which provides sequence information to the model. This step is critical because it allows the Transformer to understand the order of the features, which is important for capturing the dependencies between them. The positional

**Table 2**  
Dimensions of feature maps through CNN layers.

Layer	Output Shape
Input Layer	(8)
Conv Layer 1	(32, 6, 6)
Pool Layer 1	(32, 3, 3)
Conv Layer 2	(64, 1, 1)
Pool Layer 2	(64, 1, 1)
Flatten Layer	(64)

encoding is computed using sine (Eq. (4)) and cosine functions (Eq. (5)) of different frequencies.

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d}}\right) \quad (4)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d}}\right) \quad (5)$$

The core of the Transformer encoder is the multi-head self-attention mechanism. This mechanism applies eight attention heads to capture complex dependencies between the features, allowing the model to focus on different parts of the input simultaneously. The attention mechanism is mathematically represented using Eq. (6).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

where  $Q$ ,  $K$ , and  $V$  are the query, key, and value matrices, respectively, and  $d_k$  is the dimension of the key vectors.

Following the self-attention mechanism, the output is passed through a feed-forward network using Eq. (7), which consists of a fully connected layer followed by a ReLU activation function. This network introduces additional non-linearity and allows the model to learn complex representations of the input data.

$$FFN(x) = \max(0, xW_1 + b_1) + W_2 + b_2 \quad (7)$$

To stabilize training and improve convergence, layer normalization and residual connections are implemented within the Transformer

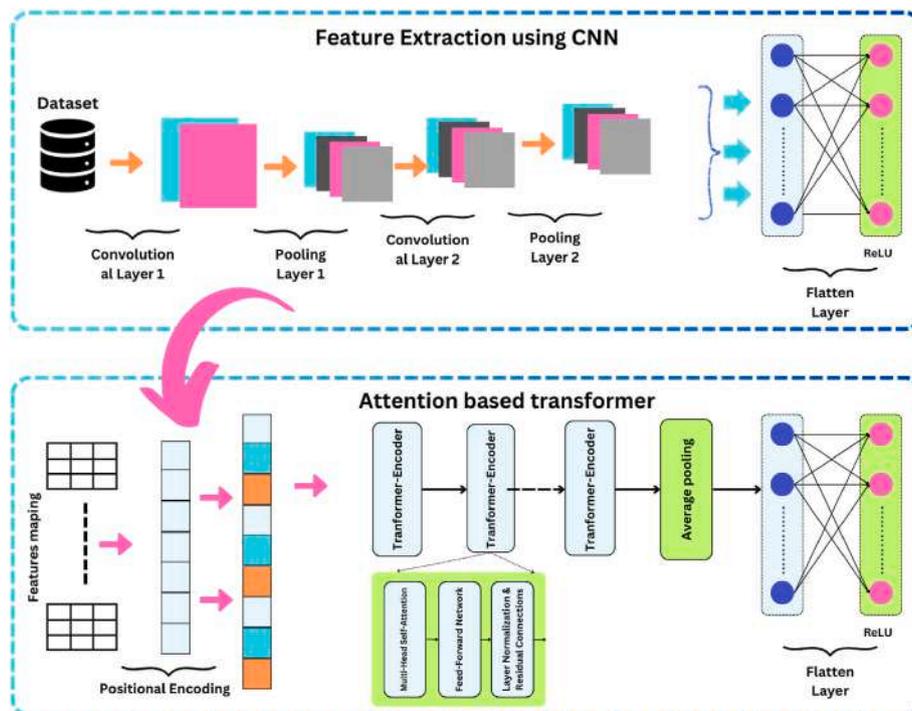


Fig. 7. Schematic diagram of the hybrid Transformer-CNN model architecture.

encoder. Layer normalization helps maintain consistent learning rates across different layers, and residual connections facilitate the flow of gradients during backpropagation, preventing vanishing or exploding gradient problems.

After processing through the Transformer encoder, the transformed features are fed into a series of fully connected layers. The first dense layer consists of 128 neurons and applies the ReLU activation function as given by Eq. (8).

$$\text{Dense}_1(x) = \text{ReLU}(W_1x + b_1) \quad (8)$$

This is followed by a dropout layer with a dropout rate of 0.5 to prevent overfitting by randomly disabling a fraction of neurons during training. Another dense layer with 64 neurons, also using the ReLU activation function, further refines the learned features as calculated using Eq. (9).

$$\text{Dense}_2(x) = \text{ReLU}(W_2x + b_2) \quad (9)$$

Another dropout layer with a 0.5 dropout rate is applied for additional regularization.

The final component of the model is the **output layer**, which consists of a single neuron with a linear activation function. This layer generates the final prediction of the Load Carrying Capacity ( $P_{cc}$ ), which is a continuous value appropriate for the regression task at hand

$$\text{Output}(x) = W_{\text{out}}x + b_{\text{out}} \quad (10)$$

In brief, the hybrid Transformer-CNN model is accurately designed to leverage the strengths of CNNs for effective feature extraction and Transformers for capturing complex dependencies. This comprehensive architecture ensures that the model can learn from the input features and make accurate predictions of the load carrying capacity of composite columns, thereby achieving high reliability and performance. By combining these advanced techniques, the model is capable of learning from the input features and making robust predictions. The complete pseudocode for the research methodology is presented in [Algorithm 1](#).

**Algorithm 1.** Pseudocode of the research methodology.

- 1: Data Collection and Preprocessing
- 2: Collect dataset with features X1 to X8 and target Y
- 3: Normalize the input features (X1 to X8)
- 4: Model Architecture Design
- 5: Define input layer to accept 8 numerical features
- 6: Design CNN Feature Extractor
- 7: Convolutional Layer 1: 32 filters, kernel size 3x3, ReLU activation
- 8: MaxPooling Layer 1: pool size 2x2
- 9: Convolutional Layer 2: 64 filters, kernel size 3x3, ReLU activation
- 10: MaxPooling Layer 2: pool size 2x2
- 11: Flatten the feature maps to 1D vector
- 12: Design Transformer Encoder
- 13: Add positional encoding
- 14: Apply multi-head self-attention: 8 attention heads
- 15: Feed-Forward Network with ReLU activation
- 16: Layer Normalization and Residual Connections
- 17: Design Fully Connected Layers
- 18: Dense Layer 1: 128 neurons, ReLU activation
- 19: Dropout Layer 1: dropout rate 0.5
- 20: Dense Layer 2: 64 neurons, ReLU activation
- 21: Dropout Layer 2: dropout rate 0.5
- 22: Output Layer: 1 neuron, linear activation
- 23: Model Training and Cross-Validation
- 24: Use 5-Fold Cross-Validation
- 25: for each fold do
- 26: Split data into training and validation sets
- 27: Train the model for defined epochs (e.g., 100) and batch size (e.g., 32)
- 28: Validate the model on validation set
- 29: Record performance metrics for each fold
- 30: end for
- 31: Model Evaluation
- 32: Calculate average performance metrics across all folds
- 33: Analyze prediction errors
- 34: Hyperparameter Tuning
- 35: Perform grid search within cross-validation framework

(continued on next column)

(continued)

- 
- 36: Optimize hyperparameters: number of filters, kernel size, number of attention heads, dropout rates
  - 37: Use k-fold cross-validation for robustness
  - 38: Model GUI Development
  - 39: Develop the GUI of developed model to predict load carrying capacity
  - 40: Host the GUI at GitHub
- 

## 4.2. Model training and cross-validation

The model training and validation process is a critical phase in the development of the predictive model for estimating the load carrying capacity of composite columns. This section details the methodology for training the model, including the use of K-Fold cross-validation ([Berrar, 2019](#); [Refaeilzadeh et al., 2009](#)) and the metrics used to evaluate performance.

K-Fold cross-validation is a robust method for evaluating the performance of a predictive model by dividing the dataset into  $K$  equally sized folds. In this study, a 5-Fold cross-validation approach is employed, as illustrated in [Fig. 8](#). The dataset is split into five subsets, and the model is trained and validated five times, each time using a different fold as the validation set and the remaining folds as the training set. This process ensures that every data point is used for both training and validation, providing a comprehensive evaluation of the model's performance.

To prevent overfitting in the optimal developed Transformer-CNN model, several strategies were employed ([Benzaamia et al., 2024](#)). Firstly, the 5-Fold cross-validation approach inherently reduces overfitting by ensuring that the model is validated on multiple subsets of the data. Additionally, early stopping was implemented to terminate training when the validation performance no longer improved, preventing the model from becoming excessively tailored to the training data. Regularization techniques, including L2 regularization, were applied to penalize large weights, promoting simpler and more generalizable models. Furthermore, dropout layers were incorporated into the network architecture to randomly deactivate neurons during training, thereby enhancing the model's robustness and reducing reliance on specific pathways. By carefully managing model complexity and incorporating these regularization methods, we ensure that the model maintains high predictive accuracy on unseen data without overfitting the training dataset.

The key steps in the 5-Fold cross-validation process are as follows.

1. Split the dataset into 5 equal-sized folds.
2. For each fold, train the model using the data from the other 4 folds and validate it using the data from the remaining fold.
3. Rotate the validation fold and repeat the process until each fold has been used as the validation set once.
4. Aggregate the performance metrics from each fold to obtain an overall assessment of the model's efficacy.

During each fold of the cross-validation process, the model undergoes a rigorous training phase followed by validation using the reserved fold. The training process involves optimizing the model parameters to minimize the loss function, the Mean Squared Error (MSE) for regression tasks:

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (11)$$

where  $y_i$  is the actual value and  $\hat{y}_i$  is the predicted value.

To ensure the model's reliability and accuracy, various performance metrics are calculated during the validation phase. These metrics include.

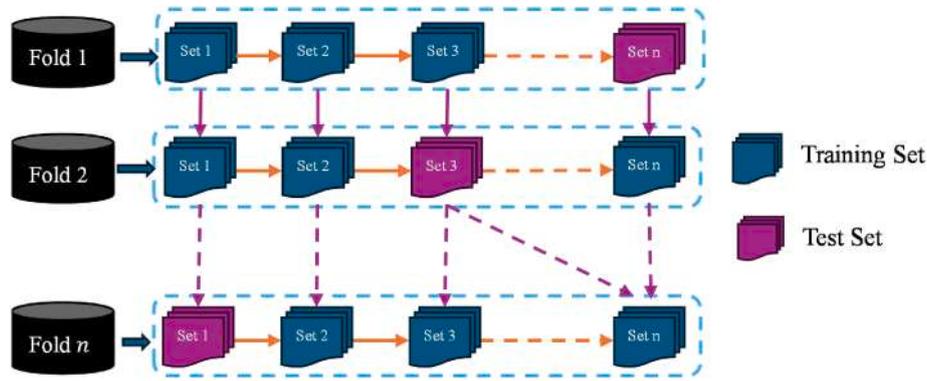


Fig. 8. Schematic of 5-fold cross-validation.

- **Coefficient of Determination ( $R^2$ ):** This metric measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It indicates how well the data points fit the statistical model. An  $R^2$  value close to 1 implies that the model explains a large portion of the variance in the output variable.  $R^2$  is given by Eq. (12).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (12)$$

where  $\bar{y}$  is the mean of the actual values.

- **Root Mean Square Error (RMSE):** RMSE provides the square root of the average squared differences between predicted and actual values, indicating the magnitude of the prediction errors. It is sensitive to outliers and gives a higher weight to larger errors, given by Eq. (13).

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (13)$$

- **Mean Absolute Error (MAE):** This metric measures the average magnitude of the errors in a set of predictions, without considering their direction as calculated by Eq. (14). It is less sensitive to outliers compared to RMSE.

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (14)$$

- **Mean Absolute Percentage Error (MAPE):** MAPE expresses the prediction accuracy as a percentage of the actual values, calculated using Eq. (15). It provides an intuitive measure of the prediction accuracy relative to the magnitude of the actual values.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (15)$$

- **Mean Squared Logarithmic Error (MSLE):** MSLE measures the ratio between the true and predicted values using logarithmic scale differences. It is particularly useful when the values span several orders of magnitude, as it penalizes underestimations more than overestimations given by Eq. (16).

$$MSLE = \frac{1}{n} \sum_{i=1}^n (\log(1 + y_i) - \log(1 + \hat{y}_i))^2 \quad (16)$$

- **Median Absolute Error:** Provides the median of all absolute differences between the predicted and actual values given by Eq. (17), offering a robust measure of central tendency.

$$\text{Median Absolute Error} = \text{median}(|y_1 - \hat{y}_1|, |y_2 - \hat{y}_2|, \dots, |y_n - \hat{y}_n|) \quad (17)$$

- **Coefficient of Variation of the Root Mean Square Error (CV-RMSE):** CV-RMSE normalizes RMSE by the mean of the observed values, providing a relative measure of the error. It is useful for comparing the performance of models across different scales of output, calculated using Eq. (18).

$$CV - RMSE = \frac{RMSE}{\bar{y}} \quad (18)$$

By aggregating these performance metrics across all folds, the model's predictive accuracy and reliability are comprehensively assessed. The training process ensures that the model learns effectively from the data, while the validation metrics provide a robust evaluation of its performance. This meticulous approach guarantees that the final model is both accurate and reliable, capable of making precise predictions of the load carrying capacity of composite columns.

### 4.3. Hyperparameter tuning

Hyperparameter tuning is a crucial step in optimizing the performance of a predictive model. It involves selecting the best set of hyperparameters that maximize the model's accuracy and reliability. This section details the methodology for hyperparameter tuning, including the model definition for tuning, hyperparameter grid specification, and the grid search process for selecting the best hyperparameters.

#### 4.3.1. Model definition for hyperparameter tuning

The first step in hyperparameter tuning is defining the model with tunable hyperparameters. In the context of the hybrid Transformer-CNN model, several hyperparameters are optimized to improve performance. These include.

1. **Number of Filters in CNN Layers:** The number of filters in the convolutional layers determines the model's capacity to extract features from the input data. The values to tune include 32, 64, and 128 filters.
2. **Kernel Size in CNN Layers:** The kernel size defines the dimensions of the convolutional filters. The choices are 3x3, 5x5, and 7x7.
3. **Dropout Rate:** Dropout is used to prevent overfitting by randomly dropping neurons during training. The dropout rates to tune include 0.2, 0.3, 0.5, and 0.7.

4. **Number of Attention Heads in Transformer Encoder:** The number of attention heads in the multi-head self-attention mechanism can significantly impact the model's ability to capture dependencies. The values considered in the study are 4, 8, and 16 heads.
5. **Feed-Forward Network Dimension in Transformer Encoder:** The size of the feed-forward network in the Transformer encoder is another critical parameter. The values include 128, 256, and 512 neurons.

#### 4.3.2. Hyperparameter grid specification

Once the tunable hyperparameters are identified, the next step is to specify a grid of possible values for each hyperparameter. This grid defines the search space for the hyperparameter tuning process. The selected values of the hyperparameters grid for the hybrid Transformer-CNN model is shown in Table 3.

This grid defines the combinations of hyperparameters that have been evaluated during the tuning process.

#### 4.3.3. Grid search and selection of best hyperparameters

Grid search is a systematic method for exploring the hyperparameter space defined by the grid specification. It involves training and evaluating the model for each combination of hyperparameters in the grid. The steps involved in grid search are as follows.

1. **Initialize GridSearchCV:** The `GridSearchCV` function from the scikit-learn library is used to perform the grid search. It systematically trains the model on each combination of hyperparameters and evaluates its performance using cross-validation. The initialization of `GridSearchCV` is given using Eq. (19):

$$\text{grid\_search} = \text{GridSearchCV}(\text{estimator} = \text{model}, \text{param\_grid} = \text{grid}, \text{cv} = 5, \text{scoring} = \text{'neg\_mean\_squared\_error'}) \quad (19)$$

2. **Fit Grid Search:** The grid search process involves fitting the model on the training data for each combination of hyperparameters and evaluating it using cross-validation. The model's performance is assessed, using Eq. (20), based on the specified scoring metric, such as the negative MSE in this study.

$$\text{grid\_search.fit}(X_{\text{train}}, y_{\text{train}}) \quad (20)$$

3. **Select Best Hyperparameters:** After evaluating all combinations, the best set of hyperparameters is selected based on the cross-validation performance. The `GridSearchCV` object stores the best hyperparameters and the corresponding model as given by Eqs. 21 and 22.

$$\text{best\_parameters} = \text{grid\_search.best\_params\_} \quad (21)$$

$$\text{best\_model} = \text{grid\_search.best\_estimator\_} \quad (22)$$

The comprehensive approach to hyperparameter tuning ensures that the final model is optimized for performance, leveraging the best

**Table 3**  
Hyperparameter range specification.

Hyperparameter	Range
Number of Filters in Conv Layers	[32, 64, 128]
Kernel Size in Conv Layers	[(3,3), (5,5), (7,7)]
Dropout Rate	[0.2, 0.3, 0.5, 0.7]
Number of Attention Heads	(Fakharifar and Chen, 2016; Berg et al., 2006; Abdulla, 2017)
Feed-Forward Network Dimension	[128, 256, 512]

combination of hyperparameters. This meticulous process helps achieve a model that is both accurate and robust, capable of making precise predictions for the load carrying capacity of composite columns.

#### 4.4. Model GUI deployment

To make the predictive model for estimating the load carrying capacity of composite columns accessible and user-friendly, a graphical user interface (GUI) is developed. This section details the development and deployment of the GUI, including the use of the Tkinter library in Python and hosting the GUI on GitHub.

##### 4.4.1. Development of a GUI using tkinter library in python

Tkinter (Lundh, 1999) is a standard GUI library in Python that provides a fast and easy way to create interactive applications. The development process of the GUI involves several steps, ensuring that the interface is intuitive and easy to use for end-users.

1. **Setting Up the Environment:** First, the Python environment is set up with Tkinter installed. This library is included in the standard Python distribution, making it readily available without the need for additional installations.
2. **Designing the Interface:** The interface design is a crucial step. It includes creating input fields for users to enter the required features ( $X_1$  to  $X_8$ ) and a button to trigger the prediction. Labels and instructions are added to guide the users on how to use the application. Fig. 9 provides a layout of the GUI design.
3. **Coding the Functionality:** The core functionality of the GUI is implemented using Python. This includes:
  - **Input Fields:** `TextEntry` widgets are used to create input fields for the features  $X_1$  to  $X_8$ .
  - **Prediction Button:** A Button widget is implemented to trigger the prediction process when clicked.
  - **Output Display:** A Label or Text widget is used to display the predicted load carrying capacity ( $P_{cc}$ ).
4. **Integrating the Model:** The trained predictive model is integrated into the GUI. This involves loading the model using a library pickle and using it to make predictions based on the user inputs.

##### 4.4.2. Hosting the GUI on GitHub

To make the GUI application easily accessible, it is hosted on GitHub. Hosting the application on GitHub provides version control, collaboration features, and easy distribution. The steps to host the GUI on GitHub are as follows.

1. **Creating a GitHub Repository:** A new repository is created on GitHub to host the project files. This includes the GUI code, the trained model file, and any additional resources or documentation.
2. **Uploading the Project Files:** The project files are uploaded to the GitHub repository. This can be done using Git commands from the terminal or directly through the GitHub web interface.
3. **Version Control:** GitHub provides robust version control, allowing for tracking changes, rolling back to previous versions, and collaborative development. Commit messages are used to document changes, making it easy to manage the project over time.
4. **Providing Documentation:** Documentation is added to the repository to guide users on how to download, install, and run the GUI application. This includes a README file with step-by-step instructions and screenshots.
5. **Sharing the Repository:** Once the project is uploaded and documented, the repository can be shared with users. They can clone or download the repository to their local machines and run the application using Python.

Hosting the GUI on GitHub ensures that the application is accessible to a wide audience, facilitates collaboration and version control, and

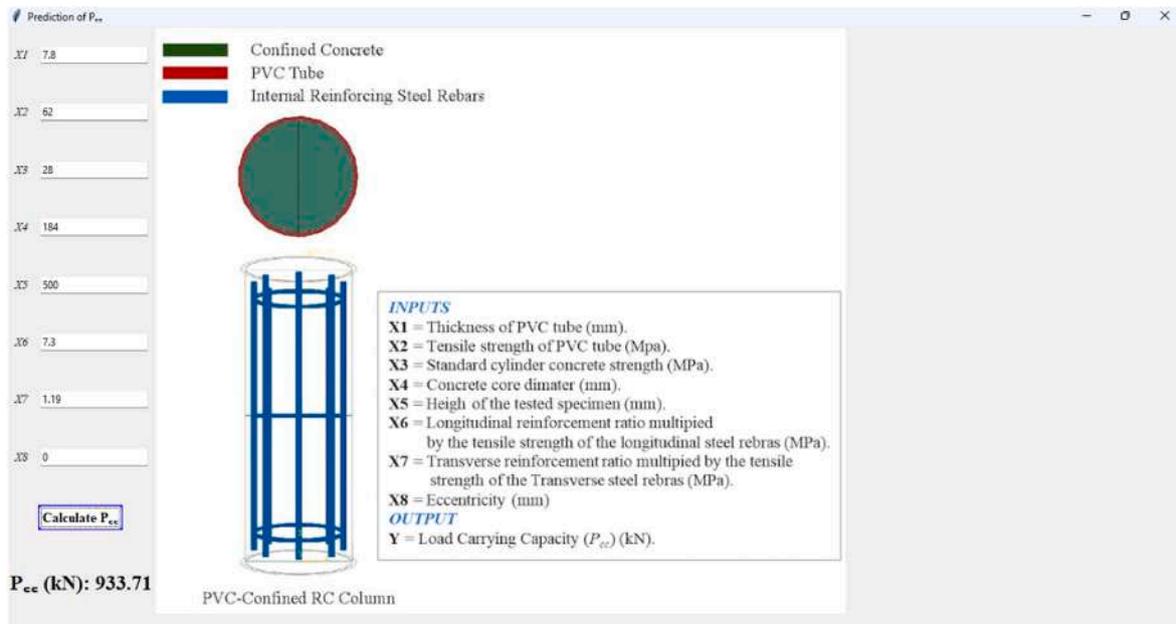


Fig. 9. Screenshot of the GitHub repository hosting the GUI project.

Table 4  
Models performance across all the metrics and cross folds.

Fold	Model	MSE	RMSE (kN)	MAE (kN)	R <sup>2</sup>	MAPE	MSLE	MedAE (kN)	CV-RMSE
1	LR	3482.53	59.01	46.67	0.95	52.03	NA	38.46	0.19
2	LR	2865.32	53.53	42.02	0.93	18.47	NA	33.51	0.16
3	LR	4899.68	70.00	55.05	0.94	21.20	0.12	42.40	0.19
4	LR	5199.46	72.11	59.99	0.92	48.54	NA	65.39	0.19
5	LR	3033.11	55.07	45.77	0.95	22.69	NA	39.06	0.15
	Min	2865.32	53.53	42.02	0.92	18.47	0.12	33.51	0.15
	Max	5199.46	72.11	59.99	0.95	52.03	0.12	65.39	0.19
	Mean	3896.02	61.94	49.90	0.94	32.59	0.12	43.76	0.18
	Std. Dev.	967.91	7.68	6.60	0.01	14.56	0.00	11.18	0.02
1	SVR	55752.09	236.12	197.44	0.16	163.93	1.01	170.03	0.78
2	SVR	33137.18	182.04	145.51	0.15	62.78	0.36	121.99	0.54
3	SVR	79964.06	282.78	203.80	0.06	91.72	0.64	147.52	0.78
4	SVR	66926.79	258.70	218.96	0.00	126.34	0.85	197.52	0.68
5	SVR	60890.00	246.76	192.31	0.04	79.78	0.54	161.51	0.66
	Mean	59334.03	241.28	191.60	0.08	104.91	0.68	159.71	0.69
	Std. Dev.	15393.34	33.44	24.72	0.06	36.12	0.23	24.94	0.09
1	DT	6690.48	81.80	56.62	0.90	24.22	0.07	32.83	0.27
2	DT	7676.94	87.62	64.73	0.80	20.87	0.07	51.10	0.26
3	DT	18710.03	136.78	98.49	0.78	27.38	0.14	67.91	0.38
4	DT	13264.13	115.17	93.57	0.80	45.05	0.20	72.19	0.30
5	DT	9348.76	96.69	71.09	0.85	21.63	0.06	38.50	0.26
	Mean	11138.07	103.61	76.90	0.83	27.83	0.11	52.51	0.29
	Std. Dev.	4400.38	20.07	16.35	0.04	8.91	0.05	15.56	0.04
1	RF	838.84	28.96	18.00	0.99	6.88	0.01	14.00	0.10
2	RF	2637.59	51.36	31.67	0.93	9.55	0.02	16.24	0.15
3	RF	3730.60	61.08	30.58	0.96	8.36	0.01	14.68	0.17
4	RF	1664.96	40.80	28.15	0.98	9.80	0.02	18.11	0.11
5	RF	1225.92	35.01	25.08	0.98	8.10	0.01	20.06	0.09
	Mean	2019.58	43.44	26.70	0.97	8.54	0.01	16.62	0.12
	Std. Dev.	1044.92	11.50	4.90	0.02	1.06	0.00	2.23	0.03
1	GB	398.57	19.46	12.68	0.94	4.97	0.01	7.02	0.06
2	GB	722.99	26.21	16.54	0.93	5.23	0.01	8.88	0.08
3	GB	1521.45	38.02	19.60	0.93	5.51	0.01	7.56	0.10
4	GB	602.70	23.93	15.30	0.94	5.24	0.01	11.11	0.06
5	GB	479.55	21.34	14.33	0.94	5.31	0.01	9.73	0.06
	Mean	745.05	25.79	15.69	0.94	5.25	0.01	8.86	0.07
	Std. Dev.	403.51	6.53	2.32	0.01	0.17	0.00	1.48	0.02
1	Hybrid Transformer-CNN model	419.54	20.48	13.35	0.99	5.23	0.01	7.39	0.07
2	Hybrid Transformer-CNN model	761.04	27.59	17.41	0.98	5.51	0.01	9.34	0.08
3	Hybrid Transformer-CNN model	1601.52	40.02	20.63	0.98	5.80	0.01	7.96	0.11
4	Hybrid Transformer-CNN model	634.42	25.19	16.11	0.99	5.52	0.01	11.69	0.07
5	Hybrid Transformer-CNN model	504.78	22.47	15.09	0.99	5.59	0.01	10.24	0.06
	Mean	784.26	27.15	16.52	0.99	5.53	0.01	9.32	0.08
	Std. Dev.	424.74	6.87	2.45	0.01	0.18	0.00	1.55	0.02

provides a platform for ongoing development and improvement.

### 5. Results and discussion

This section presents the performance metrics of various predictive models across different folds, compares these models with existing methods, and discusses the significance of the findings in the context of existing research. The potential applications and implications of these results in structural engineering are also considered.

#### 5.1. Performance metrics across different folds

The predictive performance of several models was evaluated using 5-Fold cross-validation. The models assessed include Linear Regression (LR), Support Vector Regression (SVR), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), and the Proposed Hybrid Transformer-CNN model. Table 4 summarizes the performance metrics for each model, including MSE, RMSE, MAE,  $R^2$ , MAPE, MSLE, MedAE, and CV-RMSE.

The Proposed Model consistently outperformed other models across most metrics. For instance, the Hybrid Transformer-CNN model achieved a mean MSE of 784.26, RMSE of 27.15, MAE of 16.52, and  $R^2$  of 0.9875, indicating high accuracy and reliability. In contrast, SVR and Decision Tree models showed significantly higher errors and lower  $R^2$  values, demonstrating poorer performance. The Proposed Model's superior performance is also reflected in its lower standard deviations, suggesting stable and consistent predictions across different folds.

#### 5.2. Comparison of model performance with other existing methods

When comparing the Hybrid Transformer-CNN model to other existing models, the results clearly indicate its superior predictive capability. The Linear Regression model, for instance, had a mean MSE of 3896.02 and an  $R^2$  of 0.9380, significantly worse than the Hybrid Transformer-CNN model. Similarly, SVR and Decision Tree models had mean  $R^2$  values of 0.0794 and 0.8269, respectively, compared to the Hybrid Transformer-CNN model's 0.9875. These comparisons highlight the effectiveness of the hybrid Transformer-CNN architecture in capturing complex patterns and dependencies within the data.

Fig. 10 illustrate the actual versus model-predicted values for the training and test sets, respectively, with a 20% deviation margin band. The Hybrid Transformer-CNN Model's predictions closely follow the actual values, staying well within the deviation margin, further demonstrating its accuracy.

Fig. 11 depicts the number of epochs versus loss and validation loss during training and validation of hybrid Transformer-CNN model, showing a stable convergence with minimal overfitting.

Fig. 12 presents a radar plot comparing the performance of different models across various metrics. The hybrid Transformer-CNN Model shows superior performance across most metrics, emphasizing its robustness and efficacy.

#### 5.3. Model interpretability

Understanding the decision-making process of deep learning models is crucial, especially in structural engineering applications where interpretability can provide valuable insights into the underlying physical phenomena (Zhang et al., 2024). To address the black-box nature of the proposed Transformer-CNN hybrid model, we employed SHAP (SHapley Additive exPlanations) to analyze and interpret the model's predictions. SHAP values quantify the contribution of each feature to the prediction for individual samples, offering both global and local interpretability (Zhang et al., 2023).

The SHAP summary plot (beeswarm) reveals the distribution and impact of each feature across all predictions. From the beeswarm plot (see Fig. 13), feature X5 shows the most substantial positive impact on

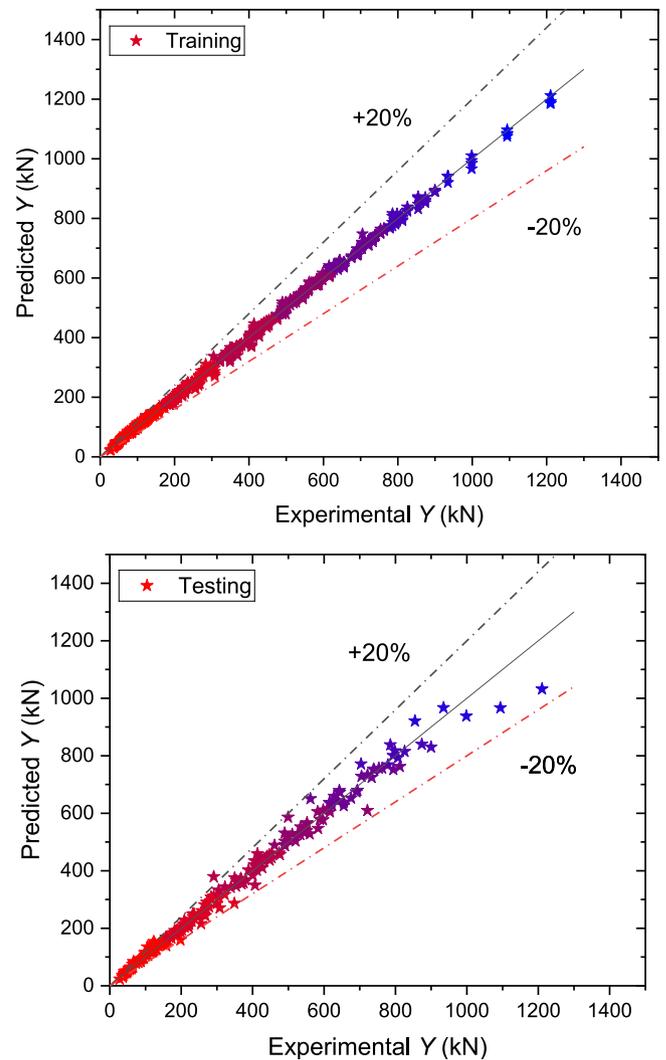


Fig. 10. Actual versus model predicted Y plot with 20 % deviation margin band on training and test set.

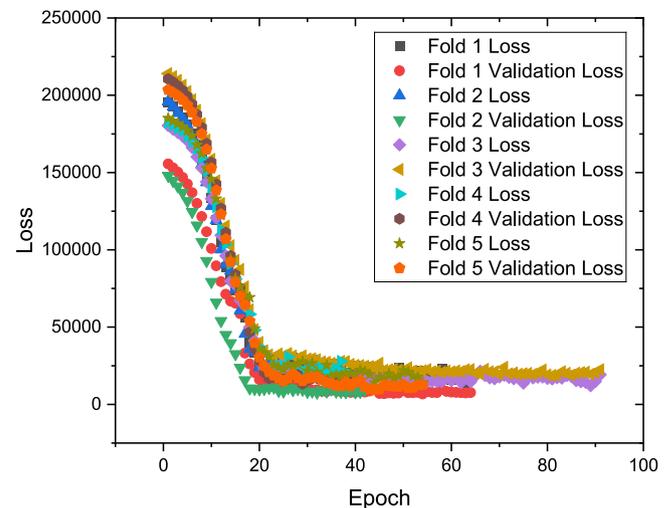


Fig. 11. Number of epochs versus loss and validation loss.

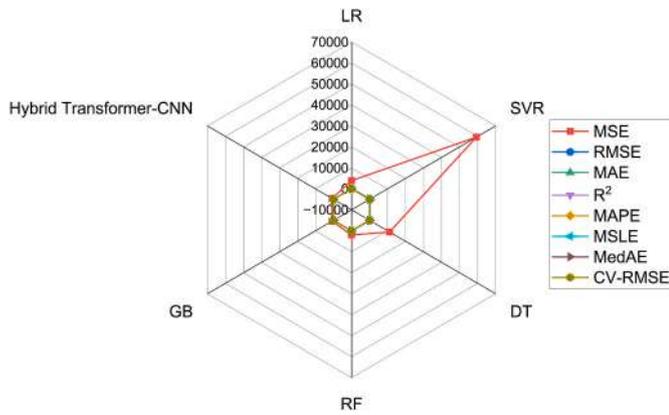


Fig. 12. Radar plot of model's performance and comparison.

the model's predictions, indicating a strong direct relationship with the target values. Conversely, X8 displays both positive and negative impacts, suggesting more complex interactions where the context or interaction with other features determines its effect on the model's output. The SHAP bar plot further quantifies these influences, showing that X5, X8, X4, and X3 have the most substantial average impacts on model output, highlighting their critical roles in the predictive accuracy of the model (see Fig. 14).

To delve deeper into the interactions of specific features, dependence plots were analyzed. The SHAP dependence plot for X5 (see Fig. 15) illustrates a predominantly linear relationship with the model's output, reaffirming its strong influence. Higher values of X5 generally lead to higher predictions, which is consistent with its engineering implications, such as material strength or geometric properties. In contrast, the dependence plot for X8 (see Fig. 16) shows clusters of both high positive and negative SHAP values across its range, indicating variable effects depending on the interaction with other features, which is indicative of operational conditions or external environmental factors.

The dependence plots for X4 and X3 (see Figs. 17 and 18,

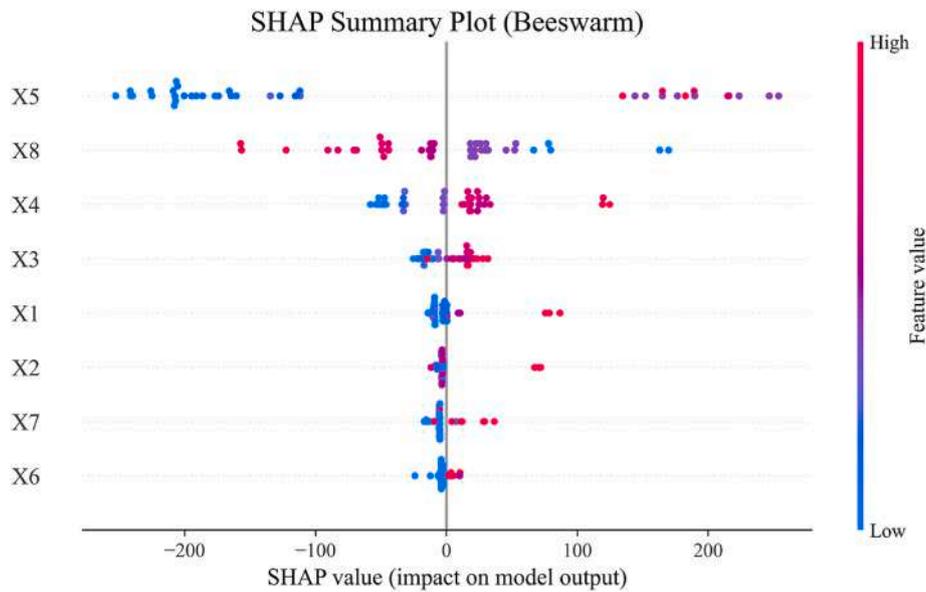


Fig. 13. SHAP summary beeswarm plot.

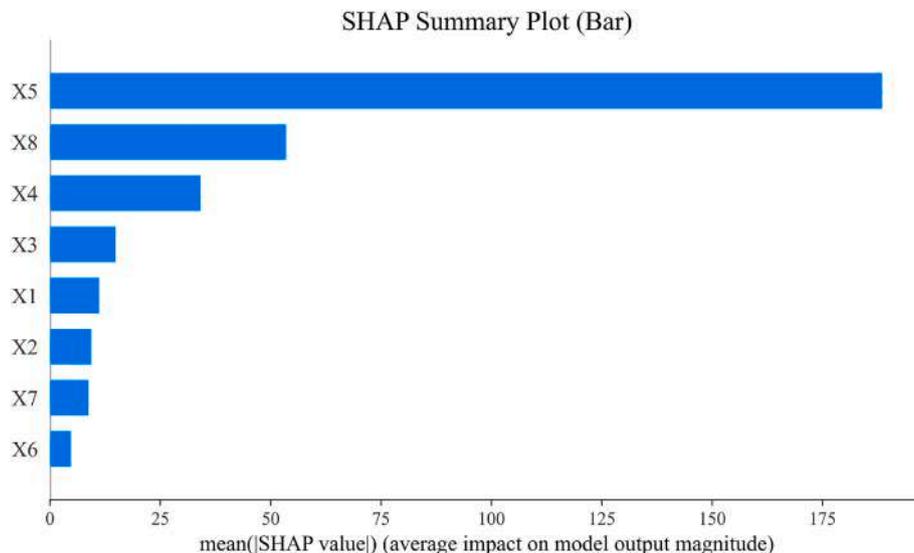


Fig. 14. SHAP summary bar plot.

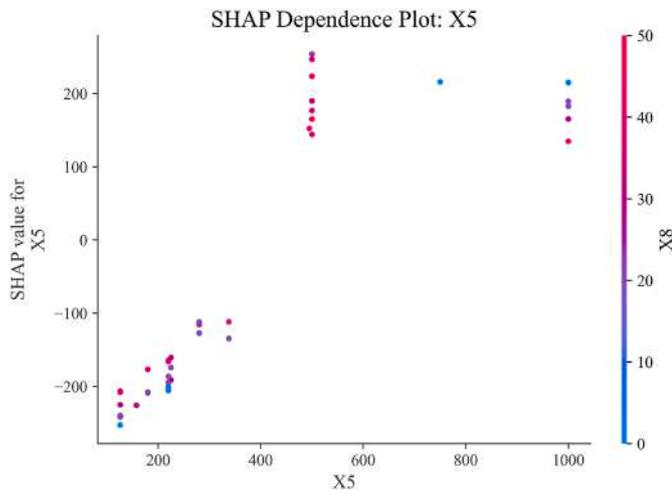


Fig. 15. SHAP dependence plot for X5 (standard cylinder concrete strength).

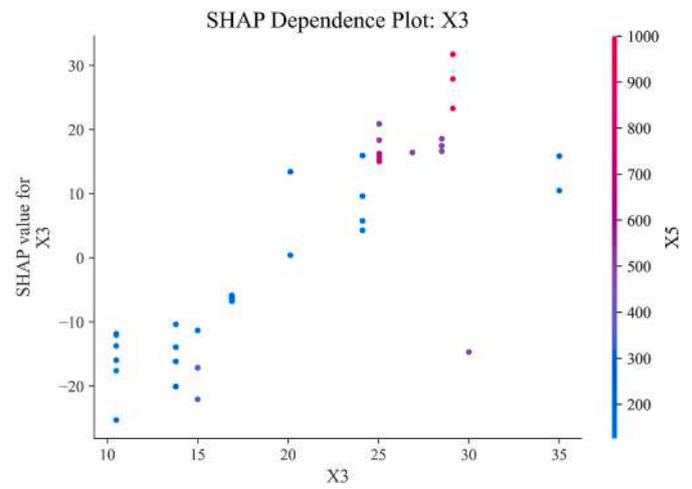


Fig. 18. SHAP dependence plot for X3 (Concrete Core Diameter).

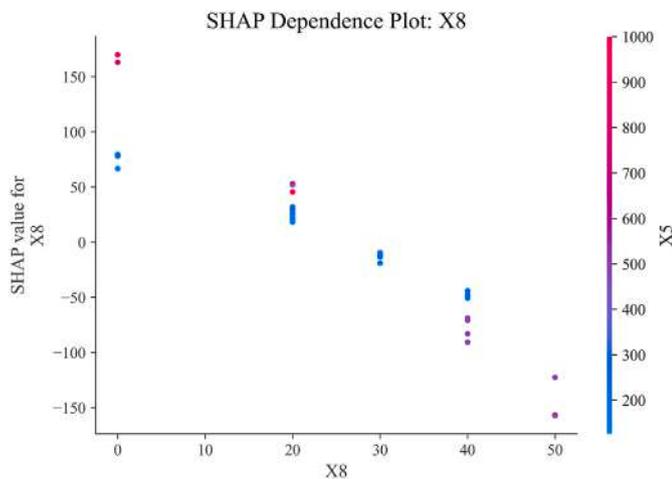


Fig. 16. SHAP dependence plot for X8 (Eccentricity).

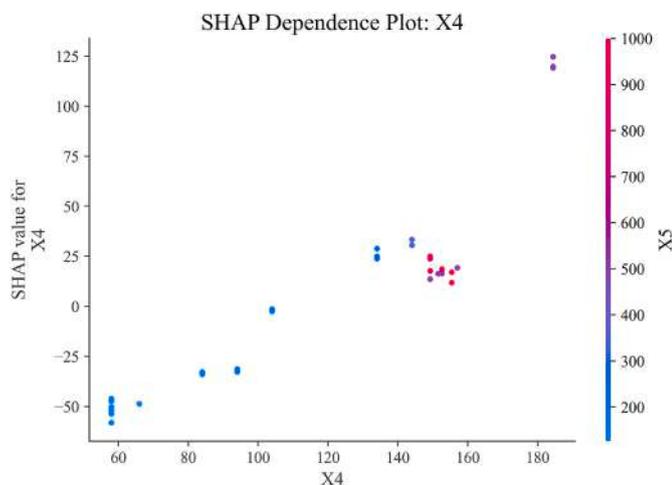


Fig. 17. SHAP dependence plot for X4 (Height of the Tested Specimen).

respectively) show more scattered distributions, suggesting that these features contribute to the model's output in more nuanced ways. For X4, higher values tend to lead to higher outputs, but with considerable variation that might be influenced by interactions with other features.

X3 shows a general trend where certain mid-range values lead to positive impacts on model outputs, hinting at an optimal range for this particular feature.

The SHAP analysis not only validates the model's predictions but also provides meaningful engineering insights. Material properties such as concrete strength and PVC tube tensile strength are critical determinants of load capacity, emphasizing the importance of material selection and quality in composite column design. Geometric parameters including core diameter and specimen height significantly influence structural performance, guiding optimal design configurations to enhance load-bearing capabilities. Moreover, loading conditions, particularly eccentricity, are crucial factors affecting load capacity, highlighting the need for careful consideration of loading scenarios in structural applications.

By integrating SHAP-based interpretability into our analysis, we bridge the gap between complex machine learning models and practical engineering applications. This approach not only enhances the transparency and trustworthiness of the model but also provides actionable insights that can inform design decisions and improve structural performance.

#### 5.4. Interpretation of results & significance of findings

The results indicate that the Proposed Hybrid Transformer-CNN model significantly outperforms traditional models such as Linear Regression, SVR, Decision Tree, Random Forest, and Gradient Boosting. The high  $R^2$  value suggests that the Proposed Model explains a large portion of the variance in the load carrying capacity, while the low MSE, RMSE, and MAE values indicate minimal prediction errors. Fig. 19 presents a box plot for composite metrics comparison, highlighting the distribution of errors for each model. The Proposed Model shows the least variance and the lowest median error across all metrics, confirming its robustness and reliability.

The superior performance of the Proposed Model underscores the importance of using advanced deep learning architectures like the hybrid Transformer-CNN in structural engineering applications. Traditional models often struggle to capture the complex interactions and dependencies in the data, leading to higher prediction errors. In contrast, the hybrid model's ability to integrate local feature extraction with global dependency modeling provides a significant advantage.

#### 5.5. Potential applications and implications in structural engineering

The findings from this study have several important implications for structural engineering. The ability to accurately predict the load carrying capacity of composite columns can lead to safer and more efficient

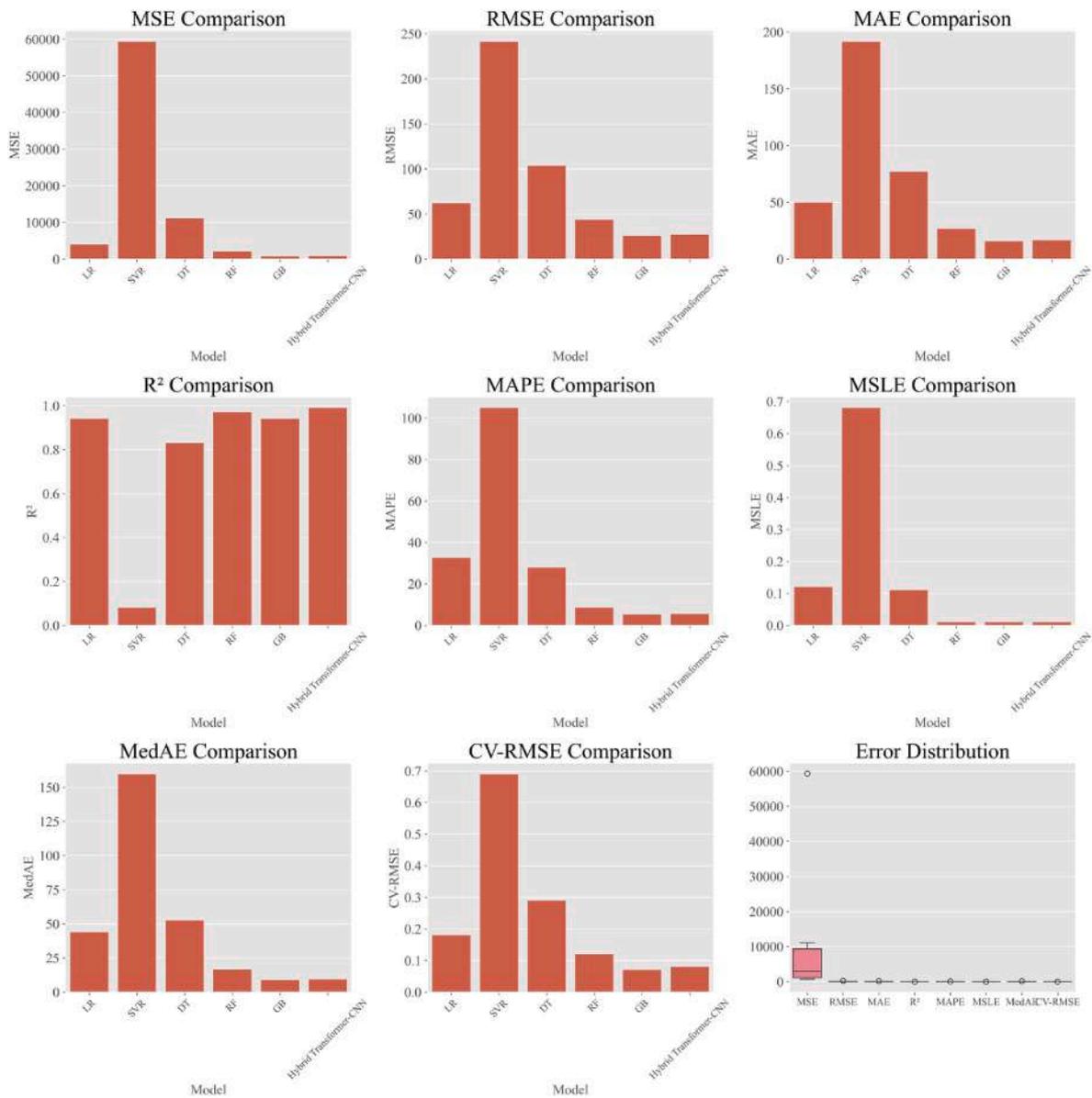


Fig. 19. Box plot for composite metrics comparison.

structural designs. Engineers can use the Proposed Model to assess the structural integrity of columns under various conditions, leading to better-informed decisions in the design and construction process. Additionally, the robustness and accuracy of the model can reduce the need for extensive physical testing, saving time and resources.

Overall, the results demonstrate that the Proposed Hybrid Transformer-CNN model is a powerful tool for predicting the load carrying capacity of composite columns, offering significant improvements over traditional methods and providing valuable insights for structural engineering applications.

### 6. Conclusions

A machine learning-based data-driven framework is presented in this study for predicting the load carrying capacity of composite columns. A comprehensive and reliable dataset was constructed, containing 200 samples with input features such as thickness of PVC tube, tensile strength of PVC tube, standard cylinder concrete strength, concrete core diameter, height of the tested specimen, longitudinal and transverse reinforcement ratios multiplied by the tensile strength of the rebars, and

eccentricity. Various machine learning algorithms, including Linear Regression, Support Vector Regression, Decision Tree, Random Forest, Gradient Boosting, and a Proposed Hybrid Transformer-CNN model, were employed for the prediction task. *K*-Fold cross-validation was utilized to ensure robust model performance evaluation. The main conclusions are summarized as follows:

1. The Proposed Hybrid Transformer-CNN model demonstrated superior performance across all metrics compared to traditional models. It achieved the lowest Mean Squared Error (MSE) and highest  $R^2$ , indicating high accuracy and reliability in predicting the load carrying capacity of composite columns.
2. Traditional models such as Linear Regression and SVR showed significantly higher prediction errors and lower  $R^2$  values, highlighting the limitations of these models in capturing the complex interactions and dependencies within the data.
3. The use of advanced deep learning architectures, specifically the hybrid Transformer-CNN, allowed for effective feature extraction and global dependency modeling, which are crucial for accurate predictions in structural engineering applications.

- The consistent performance of the Proposed Model across different folds, as evidenced by low standard deviations in metrics, underscores its robustness and reliability.
- The radar plot comparison of model performances revealed that the Proposed Model excels across multiple evaluation criteria, confirming its efficacy and suitability for practical applications in structural engineering.

The data-driven approach adopted in this study has shown high accuracy and rationality on the dataset, demonstrating the effectiveness of machine learning methods in structural engineering. However, the existence of class imbalances in the data presents challenges that prevent achieving 100% prediction accuracy.

Building upon the findings and limitations of this study, several advanced technologies and methodologies can be explored to enhance the predictive modelling of Polyvinyl Chloride Tube-Confined Concrete Columns.

- Investigating more sophisticated architectures such as Graph Neural Networks (GNNs) or Attention-based models could capture complex structural relationships and interactions more effectively.
- Combining multiple machine learning models through ensemble methods like stacking, boosting, or bagging can potentially improve prediction accuracy and model robustness.
- Employing automated hyperparameter tuning methods, such as Bayesian Optimization or Genetic Algorithms, may enhance model performance by identifying optimal configurations.
- Implementing techniques such as Synthetic Minority Over-sampling Technique (SMOTE), Adaptive Synthetic Sampling (ADASYN), or cost-sensitive learning can mitigate the effects of class imbalance, leading to more reliable predictions.
- Incorporating explainability frameworks can provide insights into the decision-making process of the models, fostering greater trust

and facilitating the integration of domain knowledge into the models.

- Combining machine learning models with traditional physics-based models can leverage the strengths of both approaches, enhancing predictive accuracy and ensuring consistency with physical laws.
- Exploring transfer learning techniques can allow models trained on related tasks or datasets to be adapted for predicting the load capacity of composite columns, reducing the need for extensive labelled data.
- Developing models capable of real-time data processing and prediction can be invaluable for structural health monitoring and proactive maintenance strategies.

By pursuing these advanced methodologies, future research can address the current limitations and further advance the application of machine learning in structural engineering, leading to more accurate, reliable, and interpretable models.

#### CRediT authorship contribution statement

**Li Shang:** Conceptualization, Data curation, Funding acquisition, Methodology, Resources, Software, Validation, Writing – review & editing. **Haytham F. Isleem:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization, Validation, Writing – original draft. **Mostafa M. Alsaadawi:** Writing – original draft, Visualization, Validation, Investigation, Conceptualization, Formal analysis, Writing – review & editing, Project administration.

#### Declaration of competing interest

The authors declares that they have no known competing financial interests or personal relationship that could have appeared to the work reported in this paper.

## Nomenclature

### Symbols

$d_k$	Dimension of the key vectors
$FFN(x)$	Feed-Forward Network function
$P_{cc}$	Load carrying capacity (kN)
$PE$	Positional Encoding
$Q, K, V$	Query, Key, and Value matrices used in attention mechanisms
$ReLU(x)$	Rectified Linear Unit function, $\max(0, x)$
$X_1$	Thickness of PVC tube (mm)
$X_2$	Tensile strength of PVC tube (MPa)
$X_3$	Standard cylinder concrete strength (MPa)
$X_4$	Concrete core diameter (mm)
$X_5$	Height of the tested specimen (mm)
$X_6$	Longitudinal reinforcement ratio multiplied by the tensile strength of the longitudinal steel rebars (MPa)
$X_7$	Transverse reinforcement ratio multiplied by the tensile strength of the transverse steel rebars (MPa)
$X_8$	Eccentricity (mm)

### Abbreviations and Acronyms

<b>ANN</b>	Artificial Neural Network
<b>CFST</b>	Concrete-Filled Steel Tubes
<b>CNN</b>	Convolutional Neural Network
<b>CV-RMSE</b>	Coefficient of Variation of the Root Mean Square Error
<b>DT</b>	Decision Tree
<b>GNN</b>	Graph Neural Network
<b>FRP</b>	Fiber Reinforced Polymer
<b>GB</b>	Gradient Boosting
<b>GUI</b>	Graphical User Interface
<b>KAN</b>	Kolmogorov–Arnold Network

- LSTM** Long Short-Term Memory
- LR** Linear Regression
- MAE** Mean Absolute Error
- MAPE** Mean Absolute Percentage Error
- MedAE** Median Absolute Error
- ML** Machine Learning
- MSE** Mean Squared Error
- MSLE** Mean Squared Logarithmic Error
- PCA** Principal Component Analysis
- PVC** Polyvinyl Chloride
- RF** Random Forest
- RMSE** Root Mean Squared Error
- R<sup>2</sup>** Coefficient of Determination
- SHAP** SHapley Additive exPlanations
- SVR** Support Vector Regression
- Transformer-CNN** Hybrid Transformer-Convolutional Neural Network
- XGBoost** Extreme Gradient Boosting

**APPENDIX A**

**Table A1**  
Database information.

No	Code	Axial Strain (mm/mm)	Load (KN)	e (mm)	t <sub>p</sub> (mm)	D (mm)	H (mm)	f <sub>yp</sub> (Mpa)	f <sub>c</sub> (MPa)	Longitudinal reinforcement		Transverse reinforcement	
										Ø (mm)	No.	Ø (mm)	S (mm)
1	E00_1	0.0215	1210.56	0	7.8	200	500	62.00	28.50	10	8	6	200
2	E00_2	0.0150	934.78	0	7.8	200	500	62.00	28.50	0	0	0	0
3	E00_3	0.0125	703.86	0	5.0	168	588	61.73	29.01	0	0	0	0
4	E00_4	0.0097	615.08	0	4.0	165	495	50.00	26.88	0	0	0	0
5	E00_5	0.0089	762.87	0	2.3	160	500	39.79	25.04	10	6	6	110
6	E00_6	0.0111	778.82	0	3.7	160	500	39.79	25.04	10	6	6	110
7	E00_7	0.0139	825.72	0	5.4	160	500	39.79	25.04	10	6	6	110
8	E00_8	0.0080	721.90	0	2.3	160	1000	39.79	25.04	10	6	6	120
9	E00_9	0.0116	740.89	0	3.7	160	1000	39.79	25.04	10	6	6	120
10	E00_10	0.0134	793.72	0	5.4	160	1000	39.79	25.04	10	6	6	120
11	E00_11	0.0082	794.17	0	2.3	160	1000	39.79	29.12	10	6	6	120
12	E00_12	0.0100	812.46	0	3.7	160	1000	39.79	29.12	10	6	6	120
13	E00_13	0.0141	873.75	0	5.4	160	1000	39.79	29.12	10	6	6	120
14	E00_14	0.0083	738.84	0	2.3	160	750	39.79	25.04	10	6	6	120
15	E00_15	0.0097	753.58	0	3.7	160	750	39.79	25.04	10	6	6	120
16	E00_16	0.0133	797.87	0	5.4	160	750	39.79	25.04	10	6	6	120
17	E00_17	0.0191	79.56	0	2.5	63	126	49.74	10.50	0	0	0	0
18	E00_18	0.0178	143.89	0	3.0	90	180	49.74	10.50	0	0	0	0
19	E00_19	0.0155	190.53	0	3.0	110	220	49.74	10.50	0	0	0	0
20	E00_20	0.0144	271.86	0	3.0	140	280	49.74	10.50	0	0	0	0
21	E00_21	0.0179	86.27	0	2.5	63	126	49.74	13.79	0	0	0	0
22	E00_22	0.0159	154.97	0	3.0	90	180	49.74	13.79	0	0	0	0
23	E00_23	0.0140	209.86	0	3.0	110	220	49.74	13.79	0	0	0	0
24	E00_24	0.0118	299.65	0	3.0	140	280	49.74	13.79	0	0	0	0
25	E00_25	0.0165	92.33	0	2.5	63	126	49.74	16.89	0	0	0	0
26	E00_26	0.0138	167.09	0	3.0	90	180	49.74	16.89	0	0	0	0
27	E00_27	0.0123	229.00	0	3.0	110	220	49.74	16.89	0	0	0	0
28	E00_28	0.0114	321.87	0	3.0	140	280	49.74	16.89	0	0	0	0
29	E00_29	0.0151	97.88	0	2.5	63	126	49.74	20.13	0	0	0	0
30	E00_30	0.0127	182.15	0	3.0	90	180	49.74	20.13	0	0	0	0
31	E00_31	0.0116	245.99	0	3.0	110	220	49.74	20.13	0	0	0	0
32	E00_32	0.0069	353.71	0	3.0	140	280	49.74	20.13	0	0	0	0
33	E00_33	0.0136	105.70	0	2.5	63	126	49.74	24.12	0	0	0	0
34	E00_34	0.0118	196.95	0	3.0	90	180	49.74	24.12	0	0	0	0
35	E00_35	0.0089	267.80	0	3.0	110	220	49.74	24.12	0	0	0	0
36	E00_36	0.0061	406.87	0	3.0	140	280	49.74	24.12	0	0	0	0
37	E00_37	0.0102	63.89	0	2.0	70	158	33.40	15.00	0	0	0	0
38	E00_38	0.0090	123.14	0	3.0	100	225	34.20	15.00	0	0	0	0
39	E00_39	0.0065	265.73	0	3.0	150	338	34.20	15.00	0	0	0	0
40	E00_40	0.0046	127.77	0	2.0	70	158	33.40	35.00	0	0	0	0
41	E00_41	0.0059	255.51	0	3.0	100	225	34.20	35.00	0	0	0	0
42	E00_42	0.0070	499.34	0	3.9	140	500	52.00	35.00	0	0	0	0
43	E00_43	0.0079	561.75	0	4.3	160	500	52.00	30.00	0	0	0	0
44	E00_44	0.0054	721.37	0	3.9	140	500	52.00	51.50	0	0	0	0
45	E20_1	0.0149	1093.92	20	7.8	200	500	62.00	28.50	10	8	6	200

(continued on next page)

Table A1 (continued)

No	Code	Axial Strain (mm/mm)	Load (KN)	$e$ (mm)	$t_p$ (mm)	$D$ (mm)	$H$ (mm)	$f_{yp}$ (Mpa)	$f_c$ (MPa)	Longitudinal reinforcement		Transverse reinforcement	
										$\emptyset$ (mm)	No.	$\emptyset$ (mm)	$S$ (mm)
46	E20_2	0.0115	854.90	20	7.8	200	500	62.00	28.50	0	0	0	0
47	E20_3	0.0072	617.28	20	5.0	168	588	61.73	29.01	0	0	0	0
48	E20_4	0.0068	552.17	20	4.0	165	495	50.00	26.88	0	0	0	0
49	E20_5	0.0067	675.77	20	2.3	160	500	39.79	25.04	10	6	6	110
50	E20_6	0.0086	693.60	20	3.7	160	500	39.79	25.04	10	6	6	110
51	E20_7	0.0110	733.30	20	5.4	160	500	39.79	25.04	10	6	6	110
52	E20_8	0.0027	582.12	20	2.3	160	1000	39.79	25.04	10	6	6	120
53	E20_9	0.0033	597.63	20	3.7	160	1000	39.79	25.04	10	6	6	120
54	E20_10	0.0039	628.11	20	5.4	160	1000	39.79	25.04	10	6	6	120
55	E20_11	0.0026	639.26	20	2.3	160	1000	39.79	29.12	10	6	6	120
56	E20_12	0.0032	655.50	20	3.7	160	1000	39.79	29.12	10	6	6	120
57	E20_13	0.0044	690.72	20	5.4	160	1000	39.79	29.12	10	6	6	120
58	E20_14	0.0033	614.37	20	2.3	160	750	39.79	25.04	10	6	6	120
59	E20_15	0.0038	622.30	20	3.7	160	750	39.79	25.04	10	6	6	120
60	E20_16	0.0044	643.98	20	5.4	160	750	39.79	25.04	10	6	6	120
61	E20_17	0.0154	56.56	20	2.5	63	0	49.74	10.50	0	0	0	0
62	E20_18	0.0131	113.25	20	3.0	90	0	49.74	10.50	0	0	0	0
63	E20_19	0.0128	157.30	20	3.0	110	0	49.74	10.50	0	0	0	0
64	E20_20	0.0119	235.36	20	3.0	140	0	49.74	10.50	0	0	0	0
65	E20_21	0.0123	61.12	20	2.5	63	0	49.74	13.79	0	0	0	0
66	E20_22	0.0114	121.58	20	3.0	90	0	49.74	13.79	0	0	0	0
67	E20_23	0.0107	172.72	20	3.0	110	0	49.74	13.79	0	0	0	0
68	E20_24	0.0095	258.19	20	3.0	140	0	49.74	13.79	0	0	0	0
69	E20_25	0.0116	65.13	20	2.5	63	0	49.74	16.89	0	0	0	0
70	E20_26	0.0098	130.46	20	3.0	90	0	49.74	16.89	0	0	0	0
71	E20_27	0.0092	187.80	20	3.0	110	0	49.74	16.89	0	0	0	0
72	E20_28	0.0078	278.34	20	3.0	140	0	49.74	16.89	0	0	0	0
73	E20_29	0.0099	68.58	20	2.5	63	0	49.74	20.13	0	0	0	0
74	E20_30	0.0091	141.84	20	3.0	90	0	49.74	20.13	0	0	0	0
75	E20_31	0.0080	201.70	20	3.0	110	0	49.74	20.13	0	0	0	0
76	E20_32	0.0053	305.30	20	3.0	140	0	49.74	20.13	0	0	0	0
77	E20_33	0.0084	73.47	20	2.5	63	0	49.74	24.12	0	0	0	0
78	E20_34	0.0077	153.36	20	3.0	90	0	49.74	24.12	0	0	0	0
79	E20_35	0.0065	221.04	20	3.0	110	0	49.74	24.12	0	0	0	0
80	E20_36	0.0049	349.07	20	3.0	140	0	49.74	24.12	0	0	0	0
81	E20_37	0.0067	46.82	20	2.0	70	158	33.40	15.00	0	0	0	0
82	E20_38	0.0060	99.69	20	3.0	100	225	34.20	15.00	0	0	0	0
83	E20_39	0.0050	232.48	20	3.0	150	338	34.20	15.00	0	0	0	0
84	E20_40	0.0029	86.07	20	2.0	70	158	33.40	35.00	0	0	0	0
85	E20_41	0.0045	197.91	20	3.0	100	225	34.20	35.00	0	0	0	0
86	E20_42	0.0042	413.50	20	3.9	140	500	52.00	35.00	0	0	0	0
87	E20_43	0.0054	489.88	20	4.3	160	500	52.00	30.00	0	0	0	0
88	E20_44	0.0033	583.23	20	3.9	140	500	52.00	51.50	0	0	0	0
89	E30_1	0.0115	998.70	30	7.8	200	500	62.00	28.50	10	8	6	200
90	E30_2	0.0096	785.79	30	7.8	200	500	62.00	28.50	0	0	0	0
91	E30_3	0.0051	545.21	30	5.0	168	588	61.73	29.01	0	0	0	0
92	E30_4	0.0053	490.84	30	4.0	165	495	50.00	26.88	0	0	0	0
93	E30_5	0.0051	596.96	30	2.3	160	500	39.79	25.04	10	6	6	110
94	E30_6	0.0065	616.46	30	3.7	160	500	39.79	25.04	10	6	6	110
95	E30_7	0.0077	655.22	30	5.4	160	500	39.79	25.04	10	6	6	110
96	E30_8	0.0019	497.74	30	2.3	160	1000	39.79	25.04	10	6	6	120
97	E30_9	0.0023	510.98	30	3.7	160	1000	39.79	25.04	10	6	6	120
98	E30_10	0.0026	534.45	30	5.4	160	1000	39.79	25.04	10	6	6	120
99	E30_11	0.0019	542.47	30	2.3	160	1000	39.79	29.12	10	6	6	120
100	E30_12	0.0022	551.13	30	3.7	160	1000	39.79	29.12	10	6	6	120
101	E30_13	0.0028	588.08	30	5.4	160	1000	39.79	29.12	10	6	6	120
102	E30_14	0.0023	526.66	30	2.3	160	750	39.79	25.04	10	6	6	120
103	E30_15	0.0025	533.57	30	3.7	160	750	39.79	25.04	10	6	6	120
104	E30_16	0.0029	552.39	30	5.4	160	750	39.79	25.04	10	6	6	120
105	E30_17	0.0130	45.57	30	2.5	63	0	49.74	10.50	0	0	0	0
106	E30_18	0.0122	95.09	30	3.0	90	0	49.74	10.50	0	0	0	0
107	E30_19	0.0114	135.28	30	3.0	110	0	49.74	10.50	0	0	0	0
108	E30_20	0.0103	208.75	30	3.0	140	0	49.74	10.50	0	0	0	0
109	E30_21	0.0106	48.51	30	2.5	63	0	49.74	13.79	0	0	0	0
110	E30_22	0.0101	101.00	30	3.0	90	0	49.74	13.79	0	0	0	0
111	E30_23	0.0089	147.34	30	3.0	110	0	49.74	13.79	0	0	0	0
112	E30_24	0.0079	228.71	30	3.0	140	0	49.74	13.79	0	0	0	0
113	E30_25	0.0098	51.41	30	2.5	63	0	49.74	16.89	0	0	0	0
114	E30_26	0.0086	107.69	30	3.0	90	0	49.74	16.89	0	0	0	0
115	E30_27	0.0078	159.72	30	3.0	110	0	49.74	16.89	0	0	0	0
116	E30_28	0.0063	247.11	30	3.0	140	0	49.74	16.89	0	0	0	0
117	E30_29	0.0089	53.44	30	2.5	63	0	49.74	20.13	0	0	0	0
118	E30_30	0.0075	116.28	30	3.0	90	0	49.74	20.13	0	0	0	0

(continued on next page)

Table A1 (continued)

No	Code	Axial Strain (mm/mm)	Load (KN)	e (mm)	t <sub>p</sub> (mm)	D (mm)	H (mm)	f <sub>yp</sub> (Mpa)	f <sub>c</sub> (MPa)	Longitudinal reinforcement		Transverse reinforcement	
										Ø (mm)	No.	Ø (mm)	S (mm)
119	E30_31	0.0062	171.48	30	3.0	110	0	49.74	20.13	0	0	0	0
120	E30_32	0.0048	270.70	30	3.0	140	0	49.74	20.13	0	0	0	0
121	E30_33	0.0075	56.35	30	2.5	63	0	49.74	24.12	0	0	0	0
122	E30_34	0.0061	124.70	30	3.0	90	0	49.74	24.12	0	0	0	0
123	E30_35	0.0052	187.34	30	3.0	110	0	49.74	24.12	0	0	0	0
124	E30_36	0.0043	308.07	30	3.0	140	0	49.74	24.12	0	0	0	0
125	E30_37	0.0060	35.61	30	2.0	70	158	33.40	15.00	0	0	0	0
126	E30_38	0.0050	82.14	30	3.0	100	225	34.20	15.00	0	0	0	0
127	E30_39	0.0043	206.85	30	3.0	150	338	34.20	15.00	0	0	0	0
128	E30_40	0.0022	60.68	30	2.0	70	158	33.40	35.00	0	0	0	0
129	E30_41	0.0035	158.21	30	3.0	100	225	34.20	35.00	0	0	0	0
130	E30_42	0.0029	349.65	30	3.9	140	500	52.00	35.00	0	0	0	0
131	E30_43	0.0043	432.70	30	4.3	160	500	52.00	30.00	0	0	0	0
132	E30_44	0.0024	489.74	30	3.9	140	500	52.00	51.50	0	0	0	0
133	E40_1	0.0099	900.01	40	7.8	200	500	62.00	28.50	10	8	6	200
134	E40_2	0.0096	705.22	40	7.8	200	500	62.00	28.50	0	0	0	0
135	E40_3	0.0041	476.27	40	5.0	168	588	61.73	29.01	0	0	0	0
136	E40_4	0.0041	427.15	40	4.0	165	495	50.00	26.88	0	0	0	0
137	E40_5	0.0033	502.59	40	2.3	160	500	39.79	25.04	10	6	6	110
138	E40_6	0.0045	520.24	40	3.7	160	500	39.79	25.04	10	6	6	110
139	E40_7	0.0056	559.84	40	5.4	160	500	39.79	25.04	10	6	6	110
140	E40_8	0.0014	404.68	40	2.3	160	1000	39.79	25.04	10	6	6	120
141	E40_9	0.0016	417.60	40	3.7	160	1000	39.79	25.04	10	6	6	120
142	E40_10	0.0019	446.72	40	5.4	160	1000	39.79	25.04	10	6	6	120
143	E40_11	0.0013	440.37	40	2.3	160	1000	39.79	29.12	10	6	6	120
144	E40_12	0.0015	453.60	40	3.7	160	1000	39.79	29.12	10	6	6	120
145	E40_13	0.0020	490.57	40	5.4	160	1000	39.79	29.12	10	6	6	120
146	E40_14	0.0018	448.07	40	2.3	160	750	39.79	25.04	10	6	6	120
147	E40_15	0.0019	453.22	40	3.7	160	750	39.79	25.04	10	6	6	120
148	E40_16	0.0022	474.12	40	5.4	160	750	39.79	25.04	10	6	6	120
149	E40_17	0.0110	36.47	40	2.5	63	0	49.74	10.50	0	0	0	0
150	E40_18	0.0107	79.90	40	3.0	90	0	49.74	10.50	0	0	0	0
151	E40_19	0.0524	117.58	40	3.0	110	0	49.74	10.50	0	0	0	0
152	E40_20	0.0092	182.90	40	3.0	140	0	49.74	10.50	0	0	0	0
153	E40_21	0.0085	37.98	40	2.5	63	0	49.74	13.79	0	0	0	0
154	E40_22	0.0090	83.89	40	3.0	90	0	49.74	13.79	0	0	0	0
155	E40_23	0.0079	124.67	40	3.0	110	0	49.74	13.79	0	0	0	0
156	E40_24	0.0063	198.65	40	3.0	140	0	49.74	13.79	0	0	0	0
157	E40_25	0.0083	40.22	40	2.5	63	0	49.74	16.89	0	0	0	0
158	E40_26	0.0075	88.37	40	3.0	90	0	49.74	16.89	0	0	0	0
159	E40_27	0.0067	133.74	40	3.0	110	0	49.74	16.89	0	0	0	0
160	E40_28	0.0051	214.71	40	3.0	140	0	49.74	16.89	0	0	0	0
161	E40_29	0.0075	41.54	40	2.5	63	0	49.74	20.13	0	0	0	0
162	E40_30	0.0068	94.32	40	3.0	90	0	49.74	20.13	0	0	0	0
163	E40_31	0.0052	142.14	40	3.0	110	0	49.74	20.13	0	0	0	0
164	E40_32	0.0042	233.54	40	3.0	140	0	49.74	20.13	0	0	0	0
165	E40_33	0.0065	43.34	40	2.5	63	0	49.74	24.12	0	0	0	0
166	E40_34	0.0056	99.64	40	3.0	90	0	49.74	24.12	0	0	0	0
167	E40_35	0.0042	153.57	40	3.0	110	0	49.74	24.12	0	0	0	0
168	E40_36	0.0038	264.56	40	3.0	140	0	49.74	24.12	0	0	0	0
169	E40_37	0.0069	27.71	40	2.0	70	158	33.40	15.00	0	0	0	0
170	E40_38	0.0048	66.30	40	3.0	100	225	34.20	15.00	0	0	0	0
171	E40_39	0.0037	178.81	40	3.0	150	338	34.20	15.00	0	0	0	0
172	E40_40	0.0018	41.39	40	2.0	70	158	33.40	35.00	0	0	0	0
173	E40_41	0.0026	119.45	40	3.0	100	225	34.20	35.00	0	0	0	0
174	E40_42	0.0019	283.54	40	3.9	140	500	52.00	35.00	0	0	0	0
175	E40_43	0.0033	369.35	40	4.3	160	500	52.00	30.00	0	0	0	0
176	E40_44	0.0014	388.50	40	3.9	140	500	52.00	51.50	0	0	0	0
177	E50_1	0.0094	807.17	50	7.8	200	500	62.00	28.50	10	8	6	200
178	E50_2	0.0088	643.00	50	7.8	200	500	62.00	28.50	0	0	0	0
179	E50_3	0.0033	415.16	50	5.0	168	588	61.73	29.01	0	0	0	0
180	E50_4	0.0032	368.07	50	4.0	165	495	50.00	26.88	0	0	0	0
181	E50_5	0.0024	405.46	50	2.3	160	500	39.79	25.04	10	6	6	110
182	E50_6	0.0031	423.90	50	3.7	160	500	39.79	25.04	10	6	6	110
183	E50_7	0.0037	461.32	50	5.4	160	500	39.79	25.04	10	6	6	110
184	E50_8	0.0009	322.78	50	2.3	160	1000	39.79	25.04	10	6	6	120
185	E50_9	0.0011	339.00	50	3.7	160	1000	39.79	25.04	10	6	6	120
186	E50_10	0.0014	367.05	50	5.4	160	1000	39.79	25.04	10	6	6	120
187	E50_11	0.0009	354.78	50	2.3	160	1000	39.79	29.12	10	6	6	120
188	E50_12	0.0011	365.60	50	3.7	160	1000	39.79	29.12	10	6	6	120
189	E50_13	0.0014	400.69	50	5.4	160	1000	39.79	29.12	10	6	6	120
190	E50_14	0.0013	374.77	50	2.3	160	750	39.79	25.04	10	6	6	120
191	E50_15	0.0015	379.93	50	3.7	160	750	39.79	25.04	10	6	6	120

(continued on next page)

Table A1 (continued)

No	Code	Axial Strain (mm/mm)	Load (KN)	$e$ (mm)	$t_p$ (mm)	$D$ (mm)	$H$ (mm)	$f_{yp}$ (Mpa)	$f_c$ (MPa)	Longitudinal reinforcement		Transverse reinforcement	
										$\emptyset$ (mm)	No.	$\emptyset$ (mm)	$S$ (mm)
192	E50_16	0.0018	412.04	50	5.4	160	750	39.79	25.04	10	6	6	120
193	E50_19	0.0401	101.96	50	3.0	110	220	49.74	10.50	0	0	0	0
194	E50_20	0.0082	159.44	50	3.0	140	280	49.74	10.50	0	0	0	0
195	E50_23	0.0071	105.51	50	3.0	110	220	49.74	13.79	0	0	0	0
196	E50_24	0.0059	171.42	50	3.0	140	280	49.74	13.79	0	0	0	0
197	E50_35	0.0038	125.67	50	3.0	110	220	49.74	24.12	0	0	0	0
198	E50_36	0.0031	222.43	50	3.0	140	280	49.74	24.12	0	0	0	0
199	E50_43	0.0022	304.84	50	4.3	160	500	52.00	30.00	0	0	0	0
200	E50_44	0.0008	291.13	50	3.9	140	500	52.00	51.50	0	0	0	0

## Data availability

Data will be made available on request.

## References

- Abbas, J.L., 2023. Structural behavior of concrete filled double-skin PVC tubular columns confined by plain PVC sockets. *Open Eng.* 13. <https://doi.org/10.1515/eng-2022-0404>.
- Abdi, H., Williams, L.J., 2010. Principal component analysis. *WIREs Computational Statistics* 2, 433–459. <https://doi.org/10.1002/wics.101>.
- Abdulla, N.A., 2017. Concrete filled PVC tube: a review. *Constr. Build. Mater.* 156, 321–329. <https://doi.org/10.1016/j.conbuildmat.2017.08.156>.
- Abdulla, N.A., 2020. Mechanical behavior of slender composite columns under axial compression load. *KSCCE J. Civ. Eng.* 24, 208–218. <https://doi.org/10.1007/s12205-020-0669-y>.
- Asteris, P.G., Lemonis, M.E., Le, T.-T., Tsavdaridis, K.D., 2021a. Evaluation of the ultimate eccentric load of rectangular CFSTs using advanced neural network modeling. *Eng. Struct.* 248, 113297. <https://doi.org/10.1016/j.engstruct.2021.113297>.
- Asteris, P.G., Lemonis, M.E., Nguyen, T.-A., Van Le, H., Pham, B.T., 2021b. Soft computing-based estimation of ultimate axial load of rectangular concrete-filled steel tubes TT -, Steel and Composite Structures. *Int. J.* 39, 471–491. <https://www.kci.go.kr/kciportal/ci/sereArticleSearch/ciSereArtView.kci?sereArticleSearchBean.artid=ART002717021>.
- Asteris, P.G., Tsavdaridis, K.D., Lemonis, M.E., Ferreira, F.P.V., Le, T.-T., Gantes, C.J., Formisano, A., 2024. AI-powered GUI for prediction of axial compression capacity in concrete-filled steel tube columns. *Neural Comput. Appl.* 36, 22429–22459. <https://doi.org/10.1007/s00521-024-10405-w>.
- Bardhan, A., Biswas, R., Kardani, N., Iqbal, M., Samui, P., Singh, M.P., Asteris, P.G., 2022. A novel integrated approach of augmented grey wolf optimizer and ANN for estimating axial load carrying-capacity of concrete-filled steel tube columns. *Constr. Build. Mater.* 337, 127454. <https://doi.org/10.1016/j.conbuildmat.2022.127454>.
- Benzaamia, A., Ghrici, M., Rebouh, R., Zygoris, N., Asteris, P.G., 2024. Predicting the shear strength of rectangular RC beams strengthened with externally-bonded FRP composites using constrained monotonic neural networks. *Eng. Struct.* 313, 118192. <https://doi.org/10.1016/j.engstruct.2024.118192>.
- Berg, A.C., Bank, L.C., Oliva, M.G., Russell, J.S., 2006. Construction and cost analysis of an FRP reinforced concrete bridge deck. *Constr. Build. Mater.* 20, 515–526. <https://doi.org/10.1016/j.conbuildmat.2005.02.007>.
- Berrar, D., 2019. Cross-validation. In: *Encyclopedia of Bioinformatics and Computational Biology*. Elsevier, pp. 542–545. <https://doi.org/10.1016/B978-0-12-809633-8.20349-X>.
- Chen, P., Wang, H., Cao, S., Lv, X., 2022. Prediction of mechanical behaviours of FRP-confined circular concrete columns using artificial neural network and support vector regression: modelling and performance evaluation. *Materials* 15, 4971. <https://doi.org/10.3390/ma15144971>.
- Christianto, A.B., Piscesa, B., Faimun, F., Aji, P., 2019. Finite element modeling of circular reinforced concrete column confined with CFRP under eccentric loading. *J. Civ. Eng.* 34, 49. <https://doi.org/10.12962/j20861206.v34i2.6375>.
- Cury, A., Crémone, C., 2012. Pattern recognition of structural behaviors based on learning algorithms and symbolic data concepts. *Struct. Control Health Monit.* 19, 161–186. <https://doi.org/10.1002/stc.412>.
- Dahl, G.E., Sainath, T.N., Hinton, G.E., 2013. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, pp. 8609–8613. <https://doi.org/10.1109/ICASSP.2013.6639346>.
- Eilbeigi, S., Tavakkolizadeh, M., Masoodi, A.R., 2022. Nonlinear regression prediction of mechanical properties for SMA-confined concrete cylindrical specimens. *Buildings* 13, 112. <https://doi.org/10.3390/buildings13010112>.
- Elshaarawy, M.K., Alsaadawi, M.M., Hamed, A.K., 2024. Machine learning and interactive GUI for concrete compressive strength prediction. *Sci. Rep.* 14, 16694. <https://doi.org/10.1038/s41598-024-66957-3>.
- Fakharifar, M., Chen, G., 2016. Compressive behavior of FRP-confined concrete-filled PVC tubular columns. *Compos. Struct.* 141, 91–109. <https://doi.org/10.1016/j.compstruct.2016.01.004>.
- Feng, D.-C., Liu, Z.-T., Wang, X.-D., Chen, Y., Chang, J.-Q., Wei, D.-F., Jiang, Z.-M., 2020. Machine learning-based compressive strength prediction for concrete: an adaptive boosting approach. *Constr. Build. Mater.* 230, 117000. <https://doi.org/10.1016/j.conbuildmat.2019.117000>.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T., 2018. Recent advances in convolutional neural networks. *Pattern Recogn.* 77, 354–377. <https://doi.org/10.1016/j.patcog.2017.10.013>.
- Gupta, P.K., Verma, V.K., 2016. Study of concrete-filled unplasticized poly-vinyl chloride tubes in marine environment. *Proc. IME M J. Eng. Marit. Environ.* 230, 229–240. <https://doi.org/10.1177/1475090214560448>.
- Hamed, A.K., Elshaarawy, M.K., Alsaadawi, M.M., 2025. Stacked-based machine learning to predict the uniaxial compressive strength of concrete materials. *Comput. Struct.* 308, 107644. <https://doi.org/10.1016/j.compstruc.2025.107644>.
- Han, L.-H., Hou, C.-C., Wang, Q.-L., 2014. Behavior of circular CFST stub columns under sustained load and chloride corrosion. *Journal of Constructional Steel Research* 103, 23–36. <https://doi.org/10.1016/j.jcsr.2014.07.021>.
- Isleem, H.F., P., J., Ahmad, J., Qaidi, S., Althoey, F., Najm, H.M., Sabri, M.M.S., 2022a. Finite element and analytical modelling of PVC-confined concrete columns under axial compression. *Frontiers in Materials* 9. <https://doi.org/10.3389/fmats.2022.1011675>.
- Isleem, H.F., Tayeh, B.A., Abid, M., Iqbal, M., Mohamed, A.M., El Sherbiny, M.G., 2022b. Finite element and artificial neural network modeling of FRP-RC columns under axial compression loading. *Frontiers in Materials* 9. <https://doi.org/10.3389/fmats.2022.888909>.
- Isleem, H.F., Peng, F., Tayeh, B.A., 2022c. Confinement model for LRS FRP-confined concrete using conventional regression and artificial neural network techniques. *Compos. Struct.* 279, 114779. <https://doi.org/10.1016/j.compstruct.2021.114779>.
- Isleem, H.F., Augustino, D.S., Mohammed, A.S., Najemalden, A.M., Jagadesh, P., Qaidi, S., Sabri, M.M.S., 2023. Finite element, analytical, artificial neural network models for carbon fibre reinforced polymer confined concrete filled steel columns with elliptical cross sections. *Frontiers in Materials* 9. <https://doi.org/10.3389/fmats.2022.1115394>.
- Isleem, H.F., Yusuf, B.O., Xingchong, W., Qiong, T., Jagadesh, P., Augustino, D.S., 2024a. Analytical and numerical investigation of polyvinyl chloride (PVC) confined concrete columns under different loading conditions. *Aust. J. Struct. Eng.* 25, 69–97. <https://doi.org/10.1080/13287982.2023.2216566>.
- Isleem, H.F., Qiong, T., Alsaadawi, M.M., Elshaarawy, M.K., Mansour, D.M., Abdullah, F., Mandor, A., Sor, N.H., Jahami, A., 2024b. Numerical and machine learning modeling of GFRP confined concrete-steel hollow elliptical columns. *Sci. Rep.* 14, 18647. <https://doi.org/10.1038/s41598-024-68360-4>.
- Jiang, T., Wang, X.M., Chen, G.M., Zhang, J.J., Zhang, W.P., 2019. Behavior of recycled brick block concrete-filled FRP tubes under axial compression. *Eng. Struct.* 198, 109498. <https://doi.org/10.1016/j.engstruct.2019.109498>.
- Li, Y.L., Zhao, X.L., Singh Raman, R.K., 2018. Mechanical properties of seawater and sea sand concrete-filled FRP tubes in artificial seawater. *Constr. Build. Mater.* 191, 977–993. <https://doi.org/10.1016/j.conbuildmat.2018.10.059>.
- Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J., 2022. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE Transact. Neural Networks Learn. Syst.* 33, 6999–7019. <https://doi.org/10.1109/TNNLS.2021.3084827>.
- Liao, J., Asteris, P.G., Cavaleri, L., Mohammed, A.S., Lemonis, M.E., Tsoukalas, M.Z., Skentou, A.D., Maraveas, C., Koopialipoor, M., Armaghani, D.J., 2021. Novel fuzzy-based optimization approaches for the prediction of ultimate axial load of circular concrete-filled steel tubes. *Buildings* 11, 629. <https://doi.org/10.3390/buildings11120629>.
- Lundh, F., 1999. An Introduction to Tkinter, vol. 539, p. 540. URL: [www.pythonware.com/Library/Tkinter/Introduction/Index.Htm](http://www.pythonware.com/Library/Tkinter/Introduction/Index.Htm).
- Lyu, F., Fan, X., Ding, F., Chen, Z., 2021. Prediction of the axial compressive strength of circular concrete-filled steel tube columns using sine cosine algorithm-support vector regression. *Compos. Struct.* 273, 114282. <https://doi.org/10.1016/j.compstruct.2021.114282>.
- Ma, L., Zhou, C., Lee, D., Zhang, J., 2022. Prediction of axial compressive capacity of CFRP-confined concrete-filled steel tubular short columns based on XGBoost

- algorithm. *Eng. Struct.* 260, 114239. <https://doi.org/10.1016/j.engstruct.2022.114239>.
- Naderpour, H., Kheyroddin, A., Amiri, G.G., 2010. Prediction of FRP-confined compressive strength of concrete using artificial neural networks. *Compos. Struct.* 92, 2817–2829. <https://doi.org/10.1016/j.compstruct.2010.04.008>.
- Oreta, A.W.C., Kawashima, K., 2003. Neural network modeling of confined compressive strength and strain of circular concrete columns. *J. Struct. Eng.* 129, 554–561. <https://doi.org/10.1061/ASCE0733-94452003129:4554>.
- Osman, M., Soliman, A.-K., 2015. Behavior of confined columns under different techniques. *International Journal of Civil, Environmental, Structural, Construction and Architectural Engineering* 9, 70–78. <https://doi.org/10.5281/zenodo.1099456>.
- Ozbakkaloglu, T., Xie, T., 2016. Geopolymer concrete-filled FRP tubes: behavior of circular and square columns under axial compression. *Compos. B Eng.* 96, 215–230. <https://doi.org/10.1016/j.compositesb.2016.04.013>.
- O'Shea, K., Nash, R., 2015. An introduction to convolutional neural networks. <https://doi.org/10.48550/arXiv.1511.08458>.
- Pearson, K., 1920. Notes on the history of correlation. *Biometrika* 13, 25. <https://doi.org/10.2307/2331722>.
- Piscesca, B., Attard, M., Samani, A., Suprobo, P., 2017. Numerical investigation on the behaviour of concrete-filled-steel-tube column under eccentric loading. In: *The Third International Conference on Civil Engineering Research (ICCER)*, pp. 1–2. Surabaya, Indonesia, Indonesia.
- Raheemah, M.A., Resan, S.F., 2019. Experimental investigation of concrete columns enhanced by PVC tubes. *IOP Conf. Ser. Mater. Sci. Eng.* 584, 012047. <https://doi.org/10.1088/1757-899X/584/1/012047>.
- Refaeilzadeh, P., Tang, L., Liu, H., 2009. Cross-validation. In: *Encyclopedia of Database Systems*. Springer US, Boston, MA, MA, pp. 532–538. [https://doi.org/10.1007/978-0-387-39940-9\\_565](https://doi.org/10.1007/978-0-387-39940-9_565).
- Saadon, A.S., Jasim, N.A., 2017. Empirical design equations for PVC-concrete composite columns. *International Journal of Engineering Science Invention (IJESI)* 6, 71–78.
- Shahin, R.I., Ahmed, M., Yehia, S.A., Liang, Q.Q., 2023. ANN model for predicting the elastic critical buckling coefficients of prismatic tapered steel web plates under stress gradients. *Eng. Struct.* 294, 116794. <https://doi.org/10.1016/j.engstruct.2023.116794>.
- Shahin, R.I., Ahmed, M., Liang, Q.Q., Yehia, S.A., 2024. Predicting the web crippling capacity of cold-formed steel lipped channels using hybrid machine learning techniques. *Eng. Struct.* 309, 118061. <https://doi.org/10.1016/j.engstruct.2024.118061>.
- Sharma, M., Ansari, M.I., Islam, N., 2022. Study of concrete filled unplasticized polyvinyl chloride tubes as columns under axial loading. *Journal of Mechanical Materials and Mechanics Research* 5, 10–17. <https://doi.org/10.30564/jmmmr.v5i1.4494>.
- Tipu, R.K., Suman, Batra, V., 2023. Development of a hybrid stacked machine learning model for predicting compressive strength of high-performance concrete. *Asian Journal of Civil Engineering* 24, 2985–3000. <https://doi.org/10.1007/s42107-023-00689-z>.
- Vedaldi, A., Lenc, K., 2014. MatConvNet - Convolutional Neural Networks for MATLAB. <https://doi.org/10.48550/arXiv.1412.4564>.
- Wang, C., Chan, T.-M., 2023. Machine learning (ML) based models for predicting the ultimate strength of rectangular concrete-filled steel tube (CFST) columns under eccentric loading. *Eng. Struct.* 276, 115392. <https://doi.org/10.1016/j.engstruct.2022.115392>.
- Wang, J.-Y., Yang, Q.-B., 2012. Investigation on compressive behaviors of thermoplastic pipe confined concrete. *Constr. Build. Mater.* 35, 578–585. <https://doi.org/10.1016/j.conbuildmat.2012.04.017>.
- Wang, X., Zhao, Y., Wang, Z., Hu, N., 2024. An ultrafast and robust structural damage identification framework enabled by an optimized extreme learning machine. *Mech. Syst. Signal Process.* 216, 111509. <https://doi.org/10.1016/j.ymsp.2024.111509>.
- Wu, P., Liu, J., Yu, F., Fang, Y., Zhu, D., 2021. Testing and numerical modeling of PVC-cfrp confined concrete column-reinforced concrete beam joint under axial load. *Iranian Journal of Science and Technology, Transactions of Civil Engineering* 45, 639–651. <https://doi.org/10.1007/s40996-020-00412-x>.
- Xu, Y., Tang, H., Chen, J., Jia, Y., Liu, R., 2021. Numerical analysis of CFRP-confined concrete-filled stainless steel tubular stub columns under axial compression. *J. Build. Eng.* 37, 102130. <https://doi.org/10.1016/j.job.2020.102130>.
- Xu, C., Zhang, Y., Isleem, H.F., Qiu, D., Zhang, Y., Alsaadawi, M.M., Tipu, R.K., El-Demerdash, W.E., Hamed, A.Y., 2024a. Numerical and machine learning models for concentrically and eccentrically loaded CFST columns confined with FRP wraps. *Struct. Concr.* <https://doi.org/10.1002/suco.202400541>.
- Xu, R., Liu, X., Wei, J., Ai, X., Li, Z., He, H., 2024b. Predicting the deformation of a concrete dam using an integration of long short-term memory (LSTM) networks and Kolmogorov–arnold networks (KANs) with a dual-stage attention mechanism. *Water* 16, 3043. <https://doi.org/10.3390/w16213043>.
- Zar, J.H., 2005. Spearman rank correlation. In: *Encyclopedia of Biostatistics*. Wiley. <https://doi.org/10.1002/0470011815.b2a15150>.
- Zarringol, M., Patel, V.I., Liang, Q.Q., 2023. Artificial neural network model for strength predictions of CFST columns strengthened with CFRP. *Eng. Struct.* 281, 115784. <https://doi.org/10.1016/j.engstruct.2023.115784>.
- Zhang, Y., Burton, H.V., 2019. Pattern recognition approach to assess the residual structural capacity of damaged tall buildings. *Struct. Saf.* 78, 12–22. <https://doi.org/10.1016/j.strusafe.2018.12.004>.
- Zhang, S., Xu, J., Lai, T., Yu, Y., Xiong, W., 2023. Bond stress estimation of profiled steel-concrete in steel reinforced concrete composite structures using ensemble machine learning approaches. *Eng. Struct.* 294, 116725. <https://doi.org/10.1016/j.engstruct.2023.116725>.
- Zhang, S., Chen, W., Xu, J., Xie, T., 2024. Use of interpretable machine learning approaches for quantitatively understanding the performance of steel fiber-reinforced recycled aggregate concrete: from the perspective of compressive strength and splitting tensile strength. *Eng. Appl. Artif. Intell.* 137, 109170. <https://doi.org/10.1016/j.engappai.2024.109170>.
- Zhou, X.-G., Hou, C., Feng, W.-Q., 2023. Optimized data-driven machine learning models for axial strength prediction of rectangular CFST columns. *Structures* 47, 760–780. <https://doi.org/10.1016/j.istruc.2022.11.030>.