Table 5
Data from experiments published in literature.

Nr.	Reference	Number of Samples	Masonry Compressive Strength in MPa	Type of Masonry Units
1	[131]	48	0.73–2.80	Cement stabilized earth brick and burnt clay brick
2	[113]	1	5.80	Soft clay wire-cut brick
3	[125]	8	2.07–10.26	Burnt clay brick and concrete brick
4	[144]	12	5.48–14.60	Hollow concrete block
5	[21]	1	2.82	Poor quality brick used in south India with quite low compressive strength
6	[138]	1	4.42	Solid concrete block
7	[81]	12	6.19–30.79	Vintage solid clay brick
8	[102]	3	1.92–2.43	Handmade burnt clay brick
9	[132]	4	6.90–10.10	Hollow concrete block
10	[137]	12	0.65–3.20	Stabilized mud block
11	[114]	3	3.34–3.85	Compressed earth block
12	[69]	8	2.90–7.20	Local clay brick in India (designated as m, b, s, and o)
13	[64]	6	1.25–10.00	Table moulded brick and wire-cut brick
14	[96]	6	7.54–11.70	Hollow concrete block
15	[28]	2	9.90–13.50	Solid clay brick
16	[20]	4	10.89–16.89	Solid clay brick
17	[68]	1	3.50	Ohio stone
18	[67]	1	18.20	Burnt clay half size brick
19	[136]	29	3.90–26.90	Red soft mud, yellow wire-cut, Poriso perforated soft mud brick, calcium silicate brick and red soft mud bricks
20	[87]	2	19.70–27.00	Modular cored unit
21	[54]	23	8.40–37.49	Solid bricks and extruded bricks
22	[64]	6	1.18–12.60	Table moulded brick and wire-cut brick
23	[104]	26	6.98–16.46	Hollow concrete blocks
24	[42]	6	9.05–13.80	Solid clay bricks
25	[59]	16	1.80–11.15	Hollow concrete blocks
26	[139]	3	5.01–6.32	Compressed cement blocks
27	[22]	3	10.00–12.00	Hollow concrete blocks
28	[134]	20	1.22–7.27	Clay bricks and compressed earth blocks
29	[98]	1	7.66	Solid clay brick
30	[109]	6	5.96–7.90	Solid clay bricks
31	[101]	1	9.84	Solid clay bricks
32	[62]	1	22.90	Granite stone units
33	[143]	11	11.57–15.86	Hollow calcium silicate blocks
34	[30]	2	1.84–2.80	Compressed earth blocks
35	[107]	4	2.62–4.05	Compressed earth blocks
36	[78]	1	5.37	Perforated concrete blocks
37	[88]	1	2.80	Perforated concrete blocks
38	[79]	1	5.26	Perforated clay bricks
39	[65]	2	5.44–5.95	Perforated concrete blocks
40	[86]	2	5.98–7.54	Solid clay bricks
41	[95]	8	7.54–11.70	Hollow concrete blocks
42	[112]	1	2.40	Vertical hollow concrete bricks
43	[32]	3	7.42–9.05	Hollow clay brick
44	[33]	3	2.53–4.20	Hollow calcarenite bricks
45	[34]	1	1.74	Hollow lightweight concrete
46	[27]	3	6.93–15.38	Clay bricks
47	[63]	4	1.57–11.03	Clay bricks, dense aggregate concrete and autoclaved aerated concrete
48	[90]	38	1.09–6.07	Solid clay bricks type i, ii and iii (unplastered)
49	[56]	7	0.45–0.97	Horizontal hollow clay bricks
50	[55]	9	0.54–1.28	Horizontal hollow clay bricks
51	[124]	1	4.21	Hollow concrete blocks
52	[117]	1	4.49	Hollow concrete blocks
53	[83]	9	4.35–12.04	Solid wall ceramic unit, hollow walled ceramic unit and concrete unit
54	[97]	4	5.25–9.27	Concrete unit
55	[96]	1	8.24	Concrete unit
56	[110]	4	2.16–3.10	Clay bricks and concrete blocks
57	[105]	2	10.58–11.54	Solid burnt clay bricks
58	[35]	1	19.24	Hollow concrete blocks
59	[43]	12	12.76–25.41	Hollow concrete blocks
60	[71]	11	13.22–26.00	Hollow concrete blocks
61	[104]	28	13.70–31	Hollow concrete blocks
62	[106]	2	10.08–11.74	Hollow concrete blocks
63	[111]	11	6.63–13.76	Hollow concrete blocks
64	[115]	12	11.31–22.69	Hollow concrete blocks
65	[116]	9	14.25–27	Hollow concrete blocks
66	[122]	2	22.20–25.70	Hollow concrete blocks
67	[73]	12	3.71–6.84	First-class fire-burnt clay bricks
68	[127]	9	2.26–3.72	Autoclaved aerated concrete (AAC) blocks
69	[142]	12	5.80–1390	Solid concrete blocks and hollow concrete blocks
70	[108]	6	5.96–9.39	Hand-moulded solid clay bricks
71	[1]	6	23.38–31.16	Concrete blocks
72	[52]	11	19.50–33.30	High-strength concrete masonry units
73	[121]	8	5.88–7.31	Solid clay bricks
74	[6]	3	6.71–7.33	Concrete blocks
75	[120]	9	2.13–2.98	Solid fired-clay bricks
76	[85]	12	5.60–23.80	Hollow concrete blocks
77	[133]	4	2.80–9.80	Thin layer mortared clay masonry bricks
78	[103]	12	5.60–18.30	Hollow concrete blocks
79	[141]	16	4.94–6.00	Concrete bricks
80	[58]	1	0.98	Solid brick walls
81	[93]	3	1.61–1.77	Masonry bricks
	Total	**611**	**0.45–33.30**	

(CM), is proposed in [60,61].

The input parameters for the chosen machine learning techniques, the number of dataset samples and the coefficient of determination (R^2) corresponding to each of the presented studies, are also included in Table 4. As seen, most of the studies consider as input the compressive strength of the masonry unit and the mortar, f_{bc} and f_{mc}, respectively. In some studies, the height to thickness (slenderness) ratio of masonry, h_w/t_w, and the mortar thickness to masonry unit thickness ratio, t_m/t_b, are among the input parameters. In [92,91,90] non-destructive test parameters like the rebound hammer number (RH) and the ultrasonic pulse velocity (UPV) are considered as inputs.

For the majority of the studies in Table 4, fewer than 100 dataset samples are used to train and test the adopted machine learning methods, while the largest dataset sample is 540. In some of those studies, a formula is proposed to predict the compressive strength of masonry walls, as an output of the regression analysis.

Table 6
Statistics of the parameters involved in modelling masonry compressive strength.

Nr.	Variable	Symbol	Unit	Category	Statistics				
					Min	Average	Max	STD	CV
1	Masonry unit compressive strength	f_{bc}	MPa	Input	2.30	18.77	74.90	15.61	0.83
2	Mortar compressive strength	f_{mc}	MPa	Input	0.30	10.23	32.00	6.06	0.59
3	Masonry height to thickness ratio	h_w/t_w	-	Input	1.15	3.31	8.60	1.22	0.37
4	Mortar thickness to masonry unit thickness ratio	t_m/t_b	-	Input	0.01	0.11	0.33	0.07	0.61
5	Masonry compressive strength	f_{wc}	MPa	Output	0.45	9.42	33.30	7.46	0.79

3. Materials and methods

This section presents the methodology followed for the development and formulation of a computational mathematical model for estimating the compressive strength of masonry, as well as the database used for the training, development, and validation of the model. Special emphasis is given to the compilation of the experimental database and the parameters used to simulate the behaviour of masonry, particularly those parameters that influence its compressive strength. Additionally, the different techniques used for the formation of the different models are listed in detail. Finally, the statistical methodology used for establishing the optimal model is presented.

3.1. Compilation of the database

It is common practice for most researchers involved in the formulation and development of predictive computational models to focus more on the computational methods and techniques used, and less on the database used for the development, training, and testing of the model's performance and reliability. The authors of this paper firmly believe the opposite. The authors maintain that the primary factor determining the reliability of a predictive computational model lies in the accuracy and comprehensiveness of the database used to describe the phenomenon under study, without neglecting the importance of the computational method. No matter how innovative or advanced a computational technique is, it cannot lead to a reliable predictive model unless it is supported by a reliable and representative database. This underlines and reaffirms the well-known adage from computer science: "garbage in, garbage out."

Given the importance of the database, it is considered useful and appropriate to briefly present the key principles that should be followed when compiling a reliable and comprehensive database. A reliable database consists of true and trustworthy data while also ensuring that the data adequately and statistically cover all possible values that each variable in the studied problem can take. Furthermore, when collecting experimental data, it is crucial to select data from experiments conducted in certified and credible laboratories, adhering to all relevant international standards, such as [16,18,15,48,45] and [46], including the preparation and storage of specimens under appropriate environmental conditions.

Based on the above considerations, a comprehensive experimental database was compiled to develop and formulate an optimal and reliable computational model for estimating the compressive strength of masonry. This database consists of 611 data points collected from 81 published experimental studies, which are listed in Table 5. To the best of the authors' knowledge, this experimental database is the largest ever compiled and used for studying the compressive strength of masonry. The database is defined by the four input (f_{bc}, f_{mc}, h_w/t_w and t_m/t_b.) and single output parameters (f_{wc}). In Table 5, besides the authors of each study used for the compilation of the experimental database, the number of samples and the range of compressive strength values studied are also provided.

Table 6 presents the aggregated statistical data for the entire database compiled from the 81 experimental studies listed in Table 5. Specifically, the table shows the minimum (Min), average, and maximum (Max) values, as well as the Standard Deviation (STD) and Coefficient of Variation (CV) for each parameter involved in the problem, which determines the compressive strength of masonry. In addition to the statistical parameters, Fig. 2 displays the histograms of the parameters used for predicting the compressive strength of masonry, as well as the relation between each parameter and the compressive strength of masonry, for all cases in the database. The histograms are accompanied by a fitted normal distribution for a clearer visualisation of the distribution of the incidence frequency of these parameters. Both Table 6 and Fig. 2 are capital as they define the validity range of soft computing models trained and developed for estimating masonry compressive strength. This is a crucial issue and will be further discussed in a subsequent section titled "*Limitations and Future Research.*"

3.2. Methodology

This section provides a detailed and in-depth description of the methodology followed to identify the optimal soft computing model for estimating the compressive strength of masonry. The key steps of the methodology are as follows:

I. Data Splitting and Normalization: The experimental database, consisting of 611 datasets, was divided into two subsets: 489 datasets (80 %) were used for model training, and the remaining 122 datasets (20 %) were used for model testing. It is important to note how the data was split. The most significant disadvantage of splitting data into a single training set and a single test set is the possibility that the test set may not follow the same class distribution as the overall data, and some numerical features may not have the same distribution in the training and test sets. Considering this, the datasets were initially sorted in descending order based on the masonry compressive strength value, which is the output parameter of the soft computing models. According to the authors' experience, this technique leads to more reliable results. Subsequently, the 80/20 split of the sorted database was performed using the k-fold cross-validation technique with 5 folds [5]. This data splitting process was carried out with normalization of the data using six different classical normalization methods.

II. Simulation Method: To estimate the compressive strength of masonry, Back Propagation Neural Networks (BPNN) models were utilized and trained using (i) four different purely mathematical and nature-inspired optimization algorithms widely used in the literature: Levenberg-Marquardt (LM) algorithm [76], Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [51], Particle Swarm Optimization (PSO) algorithm [70], and the Imperialist Competitive Algorithm (ICA) [19], (ii) architectures with 1 hidden layer, (iii) architectures with 1–30 neurons per hidden layer, with a step of 1, as opposed to using semi-empirical formulas that have been proposed for determining the number of neurons and are commonly used by most researchers [126], and (iv) 12 different transfer functions, leading to 144 (12^2) different combinations for architectures with 1 hidden layer. It is worth noting that most researchers use at most four transfer functions, which makes the evaluation of optimization algorithms

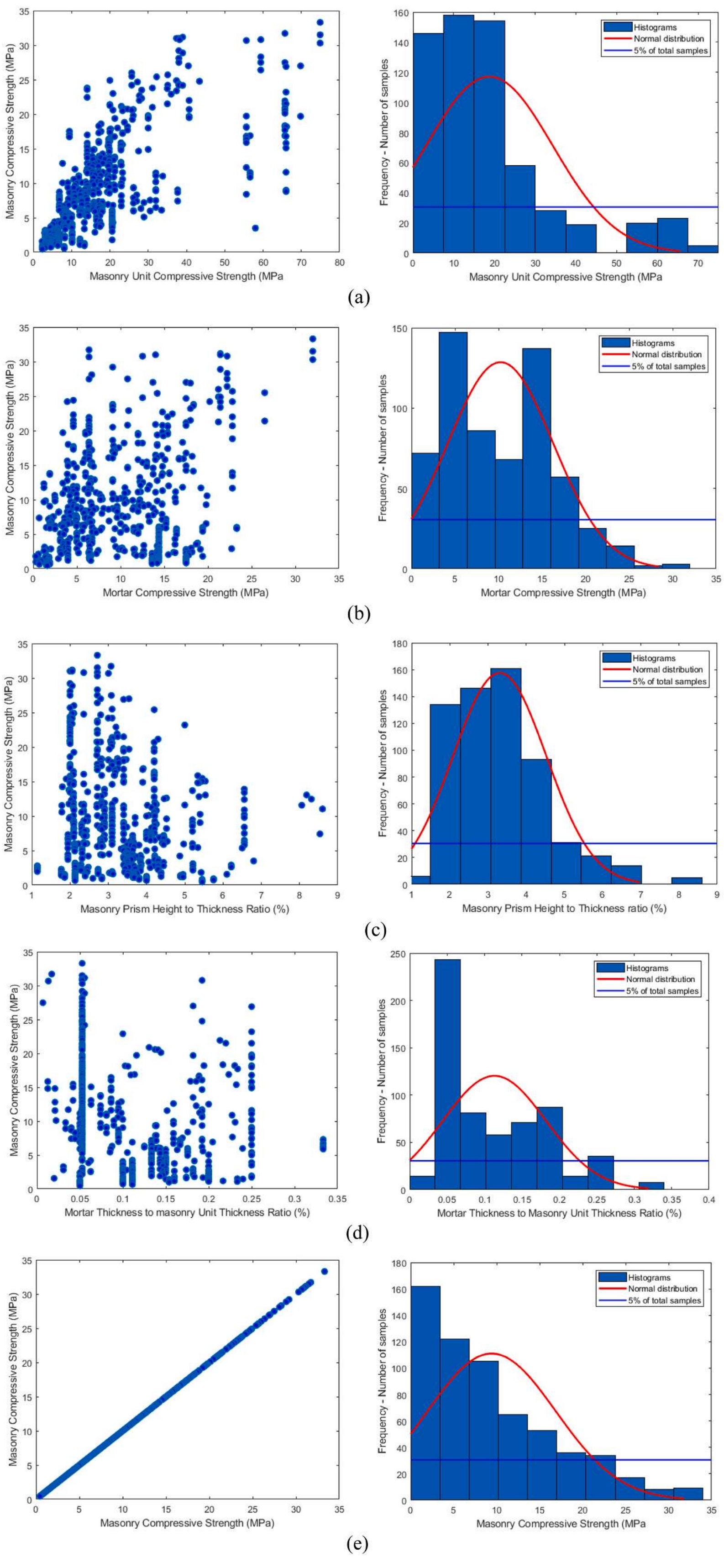

Fig. 2. Histograms of parameters used for the prediction of masonry compressive strength for: (a) Masonry compressive strength (f_{wc}) vs Unit compressive strength, (b) Masonry compressive strength (f_{wc}) vs Mortar compressive strength, (c) Masonry compressive strength (f_{wc}) vs Masonry height to thickness (h_w/t_w) ratio, (d) Masonry compressive strength (f_{wc}) vs Mortar thickness to masonry unit thickness (t_m/t_b) ratio, and (e) Masonry compressive strength (f_{wc}).

Table 7
Best ANN architectures for each one optimization algorithm based on RMSE performance index for Testing Datasets.

Ranking	Model	Architecture	Performance Indices						Comments
			Testing Datasets		Training Datasets		All Datasets		
			R^2	RMSE (MPa)	R^2	RMSE (MPa)	R^2	RMSE (MPa)	
1	ANN-LM	4–9–1	0.9615	2.1083	0.9546	2.2235	0.9557	2.2009	Normalization Technique: Minmax [0.00, 1.00] Cost Function: MSE Transfer function at the hidden layer: radbas Transfer function at the output layer: logsig
2	ANN-BFGS	4–14–1	0.9560	2.2071	0.9623	2.0289	0.9609	2.0657	Normalization Technique: Minmax [−1.00, 1.00] Cost Function: SSE Transfer function at the hidden layer: tansig Transfer function at the output layer: purelin
3	ANN-PSO	4–9–1	0.9201	3.4068	0.8888	3.8681	0.8947	3.7811	Normalization Technique: Minmax [0.00, 1.00] Cost Function: MSE Transfer function at the hidden layer: tansig Transfer function at the output layer: satlins Population: 30
4	ANN-ICA	4–18–1	0.9182	3.4123	0.8917	3.8334	0.8967	3.7536	Normalization Technique: Minmax [0.00, 1.00] Cost Function: MSE Transfer function at the hidden layer: tansig Transfer function at the output layer: satlins Population: 50 Empires: 6

ANN-LM: Artificial Neural Network optimized by Levenberg-Marquardt algorithm
ANN-BFGS: Artificial Neural Network optimized by Broyden–Fletcher–Goldfarb–Shanno quasi-Newton algorithm
ANN-PSO: Artificial Neural Network optimized by Particle swarm optimization algorithm
ANN-ICA: Artificial Neural Network optimized by Imperialist Competitive Algorithm

incomplete and often leads to incorrect conclusions. This will be discussed and demonstrated in the next section, where the results of this study will be presented. For brevity, the full set of parameters used is thoroughly discussed in the document titled *Parameters of ANNs*, which has been appended as supplementary material to this manuscript.

III. Optimal Forecasting Model: The combination of the above parameters resulted in the training and development of many different architectures. All these architectures were evaluated and ranked based on their performance, which was determined using classic and widely accepted performance indices, such as the root mean square error (RMSE), the mean absolute percentage error (MAPE), and the coefficient of determination (R^2) [3]. Briefly, the root mean square error (RMSE) is used to provide the square root of the second power of the difference between predicted and experimental values. The mean absolute percentage error (MAPE) represents the absolute error of the prediction. The coefficient of determination (R^2) measures how close to the linear correlation the prediction and the target values are. VAF calculates the Variance Accounted For between predicted and target values. Additionally, the a20-index, recently proposed [13,7,8], and widely adopted, was applied for the assessment of the developed models. The a20-index is defined according to Eq. (5):

$$a20 - \text{index} = \frac{m20}{M} \tag{5}$$

where M is the number of dataset samples and m20 is the number of samples with a value of (experimental value)/(predicted value) ratio between 0.80 and 1.20. The adoption of the a20-index within the $\pm$ 20 % range is justified by the high coefficient of variation observed in the results of compression tests of masonry specimens. The a20-index ranges from 0 to 1, with higher values indicating better performance, and for an ideal predictive model, the a20-index is expected to be close to 1.

IV. Assessment of the Contribution of Each Feature to the Prediction: One of the primary goals of this study is to evaluate the parameters involved in the problem based on their influence on the compressive strength of masonry. For this purpose, using the optimal developed model from the previous step and the SHapley Additive exPlanations (SHAP) method proposed by Lundberg and Lee in 2017 [31,82], the importance of each input parameter on the output parameter is determined and ranked according to their influence from the most to the least significant.

V. Mapping and Revealing the Nature of Masonry Compressive Strength: In this final step of the methodology, using the optimal developed model, a set of graphs is constructed. These graphs reveal the highly nonlinear nature of masonry and demonstrate their usefulness for practicing engineers in the design process of masonry structures.

4. Results and discussion

4.1. Training and development of ANN models

Following the methodology detailed in the previous section, BPNN (Back Propagation Neural Network) models were designed and trained for the estimation of masonry compressive strength. Specifically, using the parameters outlined in the supplementary materials titled *Parameters of ANNs*, a total of 1555,200 different neural network architectures were trained and developed.

The top 20 architectures, based on the RMSE (Root Mean Square Error) performance index for Testing Datasets, were identified for each of the four optimization algorithms. Additionally, the top 20 architectures across all four optimization algorithms are provided in the Spreadsheet file titled *Top 20 ANN Architectures*, which is appended as supplementary material to this work. Furthermore, Table 7 presents the optimal architecture for each of the four optimization algorithms.

Based on the results presented in the Top 20 architectures for each one of the four optimization algorithms, the following key findings are revealed during the modelling of masonry compressive strength:

- Optimization Algorithms: The best performing optimization algorithm was the Levenberg-Marquardt algorithm, followed by the Broyden–Fletcher–Goldfarb–Shanno (BFGS) quasi-Newton algorithm, the Particle Swarm Optimization (PSO) algorithm, and the Imperialist Competitive Algorithm (ICA),

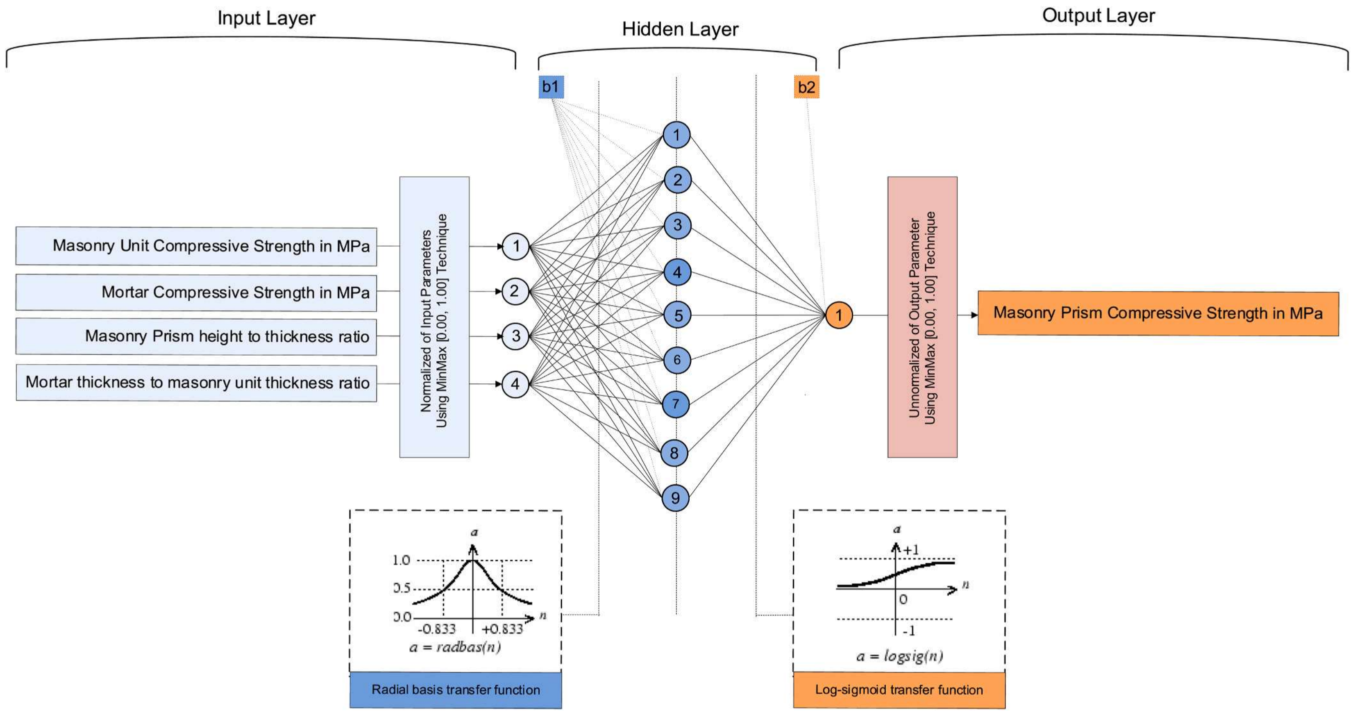

Fig. 3. Architecture of the optimal ANN-LM 4–9–1 model.

Table 8
Summary of prediction capability of the optimum ANN-LM 4-9-1 model.

Model	Datasets	Performance Indices				
		a20-index	R^2	RMSE (MPa)	MAPE	VAF
ANN-LM4–9–1	Training	0.5072	0.9546	2.2235	0.3159	91.1135
	Testing	0.5082	0.9615	2.1083	0.3391	92.0625
	All	0.5074	0.9557	2.2009	0.3205	91.2896

- Transfer Functions: The superior transfer function for the hidden layer was found to be the Radial Basis (RB) function, followed by the Softmax (SM) transfer function. It is worth noting that these two transfer functions are rarely used by researchers. For the output layer, the dominant transfer functions were, in order, the Log-sigmoid (LS) transfer function, the Radial Basis (RB) transfer function, the Symmetric Saturating Linear (SL) transfer function, and the Linear (Li) transfer function.
- Normalization Techniques: The Minmax normalization technique in the range [0.00, 1.00] was found to be the most effective for data normalization,
- Neurons Per Layer: The optimal number of neurons per layer varied significantly depending on the optimization algorithm used. This finding suggests that the semi-empirical formulas commonly proposed in the literature for determining the number of neurons are not reliable.

These results underscore the complexity and specificity required in optimizing ANN models for predicting masonry compressive strength, highlighting the importance of careful selection of network architecture, transfer functions, and normalization techniques.

4.2. *Optimal ANN model*

Among the trained and developed models presented in the previous subsection, the optimal ANN model, based on the RMSE value of the Testing Datasets, is the ANN-LM 4–9–1 model. This model was optimized using the Levenberg-Marquardt algorithm. Specifically, the optimal ANN-LM 4–9–1 model (Fig. 3) consists of a structure with four input parameters, 9 neurons in the hidden layer, and employs the Minmax normalization technique in the range [0.00–1.00]. The Radial Basis (RB) transfer function is used in the hidden layer, while the Log-sigmoid transfer function is utilized in the output layer.

Table 8 presents the values of five performance indices achieved by the developed and proposed optimal ANN-LM 4–9–1 model. It is noteworthy that all performance indices show better values for the Testing Datasets compared to those for the Training Datasets. This observation is particularly important as it indicates that the model does not suffer from the common overfitting problem. This finding will be further discussed in a subsequent section, along with additional evidence supporting that no overfitting occurred during the model training.

The diagram in Fig. 4, along with the performance indices, is used to assess the reliability and performance of the optimal ANN model. It is confirmed that the developed optimal model is a reliable tool for predicting masonry compressive strength, considering all five performance indices presented in Table 8, already introduced.

4.3. *ANN-based closed form equation for the estimation of masonry compressive strength*

It is a common practice in machine learning research for researchers to present only part of the details of the predictive models they develop and propose, along with their performance indices. However, without providing all the parameters required to create these models, it becomes impossible to verify their reliability, and even more so, to replicate or build upon these models by other researchers. To illustrate this point, it is worth noting that no proposed ANN model can be reproduced without providing, along with the architecture, the final values of weights and biases.

In light of the above, this subsection presents the closed-form equation Eq. (6) for estimating masonry compressive strength based on the developed and proposed optimal ANN-LM 4–9–1 model. Specifically, the masonry compressive strength (f_{wc}) in relation to the masonry unit compressive strength (f_{bc}), the mortar compressive strength (f_{mc}),