IRAQI Academic Scientific Journals

University of Kirkuk Journal For Administrative and Economic Science (2026) 16(2): 730-745

UKJAES

University of Kirkuk Journal For Administrative and Economic Science

ISSN:2222-2995 E-ISSN:3079-3521

University of Kirkuk Journal For Administrative and Economic Science

Ahmed Renas Abubaker& Hussein Mohammad Mahmood Faqe. Advancing Predictive Modeling using Fuzzy – Enhancing General Regression Neural Networks.*University of Kirkuk Journal For Administrative and Economic Science***(2026) 16(2):730-745.**

Advancing Predictive Modeling using Fuzzy – Enhancing General Regression Neural Networks

Renas Abubaker Ahmed[1], Mohammad Mahmood Faqe Hussein[2]

[1]*Department of Statistics and Informatics/University of Sulaimani - College of Administration and Economics, Sulaimani, Iraq*
[2]*Department of Statistics and Informatics/University of Sulaimani - College of Administration and Economics, Sulaimani, Iraq*

renas.ahmed@univsul.edu.iq[1], mohammad.faqe@univsul.edu.iq[2]

Abstract: Predictive modeling is essential for accurate forecasting across many fields but traditional regression models and most machine learning algorithms often struggle with complex nonlinear or imprecise data Generalized Regression Neural Networks GRNNs offer a promising solution yet methods for applying them to uncertain or fuzzy datasets are not well established The study develops a GRNN model which effectively handles fuzzy data The study develops a GRNN model which effectively handles fuzzy data The study develops a GRNN model which effectively handles fuzzy data The study develops a GRNN model which effectively handles fuzzy data The study employs a GRNN to predict outputs using a non-iterative kernel-based approach with input pattern summation and output layers The pattern layer calculates Gaussian-weighted distances between inputs and training samples while the summation layer aggregates these to produce predictions Fuzzy set theory enables the handling of imprecise data through its system which permits inputs and outputs to possess partial membership values The Fuzzy-GRNN framework combines GRNN predictive accuracy with fuzzy logic interpretability for regression tasks which involve uncertain or noisy data The study demonstrates that the Fuzzy GRNN model outperforms the conventional GRNN in predicting NOx levels achieving lower RMSE and MAE values The network uses fuzzy logic to model nonlinear data patterns while it handles the uncertainty present in the data The Fuzzy GRNN offers better accuracy and reliability for nonlinear regression and environmental forecasting tasks.

Keywords:GRNN, Fuzzy Logic, Defuzzification, Predictive Modeling.

تطوير النمذجة التنبوئية باستخدام المنطق الضبابي - تعزيز الشبكات العصبية للانحدار العام

م. ريناس ابوبكر احمد[1] ،أ.د. محمد محمود فقي حسين [2]

[1] جامعة السليمانية، السليمانية، العراق
[2]جامعة السليمانية، السليمانية، العراق

IRAQI Academic Scientific Journals
University of Kirkuk Journal For Administrative and Economic Science (2026) 16(2): 730-745

renas.ahmed@univsul.edu.iq[1], mohammad.faqe@univsul.edu.iq[2]

المستخلص: يُعدّ النمذجة التنبؤية ضرورية للتنبؤ الدقيق في العديد من المجالات، إلا أن نماذج الانحدار التقليدية ومعظم خوارزميات التعلم الآلي غالبًا ما تواجه صعوبة في التعامل مع البيانات المعقدة غير الخطية أو غير الدقيقة. تُقدّم الشبكات العصبية للانحدار المعمم (GRNN) حلًا واعدًا، ولكن طرق تطبيقها على مجموعات البيانات غير المؤكدة أو الضبابية لا تزال غير راسخة. تُطوّر هذه الدراسة نموذج GRNN يتعامل بفعالية مع البيانات الضبابية. تستخدم الدراسة GRNN للتنبؤ بالمخرجات باستخدام نهج غير تكراري قائم على النواة مع طبقات تجميع أنماط الإدخال والإخراج. تحسب طبقة الأنماط المسافات الموزونة غاوسيًا بين المدخلات وعينات التدريب، بينما تجمع طبقة التجميع هذه المسافات لإنتاج التنبؤات. تُمكّن نظرية المجموعات الضبابية من التعامل مع البيانات غير الدقيقة من خلال نظامها الذي يسمح للمدخلات والمخرجات بامتلاك قيم عضوية جزئية. يجمع إطار عمل Fuzzy-GRNN بين دقة التنبؤ لشبكة GRNN مع قابلية تفسير المنطق الضبابي لمهام الانحدار التي تتضمن بيانات غير مؤكدة أو مشوشة. تُظهر الدراسة أن نموذج Fuzzy GRNN يتفوق على نموذج GRNN التقليدي في التنبؤ بمستويات أكاسيد النيتروجين، محققًا قيمًا أقل في متوسط الجذر التربيعي للخطأ (RMSE) ومتوسط الخطأ المطلق (MAE). تستخدم الشبكة المنطق الضبابي لنمذجة أنماط البيانات غير الخطية مع التعامل مع عدم اليقين الموجود في البيانات. توفر شبكة Fuzzy GRNN دقة وموثوقية أفضل لمهام الانحدار غير الخطي والتنبؤ البيئي.

الكلمات المفتاحية: إزالة الغموض، المنطق الضبابي، الشبكة العصبية المعممة العامة، التنبؤ

Corresponding Author: E-mail: renas.ahmed@univsul.edu.iq

Introduction

The domain of predictive modeling serves essential functions in multiple sectors which include financial services, energy usage control, environmental protection work, medical treatment systems, and industrial operations management because its value depends entirely on the quality of the fundamental data input and the strength of the applied modeling methods. The Common regression models which researchers frequently utilize, establish valid measurements for their output results through their requirement of precise numerical data, thus they cannot identify the intricate connections between multiple input variables which exist in real-world conditions. The Generalized Regression Neural Network (GRNN) which Specht developed in 1991 solves certain limitations through its non-parametric learning system which allows single-pass training, thus enabling users to create complex nonlinear models while achieving outstanding generalization results. The function of GRNN produces major advantages which application domains including system identification, time-series forecasting, dynamic process modeling, and function approximation. The standard GRNN function operates through direct input processing which results in its inability to manage the uncertainty and imprecision that exists in actual measurement data and expert human knowledge. GRNNs have been used in numerous applications which include system identification and dynamic control and prediction activities. The research by Al-Mahasneh et al. (2018) examined how GRNNs function in system identification and control processes while demonstrating their superior training efficiency and accuracy compared to standard back-propagation neural networks. The processing of real-world data becomes difficult for GRNNs despite their advantages because these systems need fuzzy logic systems to manage the uncertainty and vague details found in actual data. The Zadeh framework emerged as a complete system to model uncertainty because it allows variables to partially join several fuzzy sets, thus enabling models to process data which contains imprecise information or linguistic content or noisy structures. The development of fuzzy systems as a new field led to three primary systems which include genetic fuzzy systems and neuro-fuzzy systems and evolving fuzzy systems as methods to enhance system performance through better system adaptability and learning capabilities. The article by Zhang et al. (2024) [20] presents a comprehensive analysis of the technological advancements which the authors show to have applications in various fields of research. The fuzzy logic system combines with GRNN to create a hybrid system which enables neural networks to approximate functions while fuzzy systems provide interpretable results and manage uncertain data. Multiple approaches present different methods for integrating GRNNs with fuzzy systems which include the direct integration of fuzzy membership functions into the kernel function of GRNN. The research done by Castellano et al.

(2023) found that the combination of Graph Neural Networks with fuzzy logic results in improved interpretability for deep learning models. The researchers developed the Fuzzy-Enhanced GRNN framework to boost predictive modeling through its creation of fuzzy input layers which convert crisp data into fuzzy outputs and fuzzy kernel functions which the GRNN design uses to measure similarity between fuzzy vectors. The Fuzzy-GRNN model performance test on benchmark datasets demonstrates that the model outperforms both standard GRNN and traditional regression methods through its better predictive accuracy while showing how fuzzy reasoning enhances model robustness and interpretability. The research work establishes a link between predictive performance and uncertain data processing which drives progress in intelligent modeling systems and establishes a base for future hybrid fuzzy-neural network applications that need reliable and interpretable predictions in multiple fields with special emphasis on the MASS cement industry in Sulaimani Kurdistan-Iraq.

1st: Methodology

1- Generalized Regression Neural Network (GRNN) Architecture

The Generalized Regression Neural Network (GRNN) differs from traditional neural networks because it uses nonparametric probability density estimation method instead of requiring repeated weight training for its operations. The standard GRNN architecture consists of four layers: the input layer, pattern (radial basis) layer, summation layer, and output layer.

A. Input Layer

The input layer acts as an interface between the external data and the network. Each neuron corresponds to a single explanatory (predictor) variable. Let the input vector be denoted as

$$\mathbf{x} = [x_1, x_2, \dots, x_p]^T \quad (1)$$

where p is the number of input variables. The input layer performs no computation; instead, it simply transmits the input values directly to the pattern layer [4], [14].

B. Pattern Layer (Radial Basis Layer)

The main processing unit of the GRNN system operates through its pattern layer which serves as its primary computational component. The system consists of one neuron which represents each training observation. Each pattern neuron calculates the Euclidean distance between the input vector x and its corresponding training input vector x_i for all training samples between i=1 and i=n where n represents the total number of training samples. The i-th pattern neuron produces its output by transforming distance values through a Gaussian radial basis function.

$$P_i = \exp\left(-\frac{(\mathbf{x}-\mathbf{x}_i)^T(\mathbf{x}-\mathbf{x}_i)}{2\sigma^2}\right) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\sigma^2}\right) \quad (2)$$

Where:

- $\mathbf{x}$ is the input vector,
- $\mathbf{x}_i$ is the i-th training input vector,
- $\| \mathbf{x} - \mathbf{x}_i \|^2$ denotes the squared Euclidean distance,
- σ is the smoothing (spread) parameter, which controls the width of the Gaussian kernel.

The parameter σ plays a crucial role in model behavior: larger values yield smoother regression surfaces, whereas smaller values make the model more sensitive to individual training samples.

C. Summation Layer

The summation layer combines the outputs of the pattern layer through two separate summation units which include a denominator summation unit and a numerator summation unit [5], [11].

(1) Denominator Summation Unit

This unit computes the total contribution of all pattern neurons:

$$S_D = \sum_{i=1}^{n} P_i \tag{3}$$

where P_i is the output of the i-th pattern neuron. This term represents the overall similarity between the input vector and all training samples.

(2) Numerator Summation Unit

The numerator summation unit computes a weighted sum of the target outputs:

$$S_N = \sum_{i=1}^{n} y_i P_i \tag{4}$$

where y_i denotes the target output associated with the i-th training sample. If the output is multidimensional, this summation is performed in a vectorized manner.

D. Output Layer

The output layer produces the final GRNN prediction by taking the ratio of the numerator and denominator summations:

$$\hat{Y}(x) = \frac{S_N}{S_D} = \frac{\sum_{i=1}^{n} y_i P_i}{\sum_{i=1}^{n} P_i} = \frac{\sum_{i=1}^{n} y_i \exp\left(-\frac{\|x-x_i\|^2}{2\sigma^2}\right)}{\sum_{i=1}^{n} \exp\left(-\frac{\|x-x_i\|^2}{2\sigma^2}\right)} \tag{5}$$

This expression reveals that GRNN estimates the conditional expectation of the output given the input by computing a distance-weighted average of the training targets. Consequently, training samples closer to the input vector exert a stronger influence on the prediction, as governed by the Gaussian kernel and the smoothing parameter σ .

2- Fuzzy Set Theory

Classical set theory, originally developed by Cantor (1895), is grounded in binary logic, where an element either belongs to a set or does not. This relationship is described using a characteristic (indicator) function:

$$\chi_A(x) = \begin{cases} 1, & \text{if } x \in A, \\ 0, & \text{if } x \notin A. \end{cases} \tag{6}$$

Where:

- x: the element being tested for membership in set A.
- A: a classical (crisp) set.
- $\chi_A(x)$: the characteristic (indicator) function, which shows whether $x$belongs to A.

The typical set system works well for distinct concepts yet fails to create effective models for the ambiguous and unclear elements which people encounter in their daily life. The concepts of tall and young and approximately equal are vague because they do not have precise limits which prevent their representation through traditional crisp membership. Zadeh (1965) [19] developed fuzzy set theory to create a system which enables elements to have partial membership rights within a set . The system defines membership through a scale which ranges from 0 to 1 as its continuous measurement system . The fuzzy set $\tilde{A}$ established on the complete universe of discourse X defines its structure through the following relationship:

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}, \mu_{\tilde{A}}: X \to [0,1] \tag{7}$$

where $\mu_{\tilde{A}}(x)$ denotes the membership function, representing the degree to which element x belongs to the fuzzy set $\tilde{A}$.

3- Membership Functions

The membership function (MF) is the core component of fuzzy set theory. It assigns a membership degree to each element of the universe of discourse. The selection of an appropriate MF depends on the problem context, empirical data, or expert knowledge [10]. Common Types of Membership Functions

A. Triangular Membership Function

$$\mu(x;a,b,c) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a \le x \le b, \\ \frac{c-x}{c-b}, & b \le x \le c, \\ 0, & x \ge c. \end{cases} \quad (8)$$

Where:

- x: the input value whose membership degree is being calculated.
- a: the lower bound of the triangular fuzzy set, where membership starts to increase from 0.
- b: the peak value of the triangular fuzzy set, where membership reaches 1 (full membership).
- c: the upper bound of the triangular fuzzy set, where membership decreases back to 0.
- $\mu(x;a,b,c)$: the membership function, giving a value in [0، 1]that represents the degree of membership of $x$in the fuzzy set.

B. Trapezoidal Membership Function

$$\mu(x;a,b,c,d) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a \le x \le b, \\ 1, & b \le x \le c, \\ \frac{d-x}{d-c}, & c \le x \le d, \\ 0, & x \ge d. \end{cases} \quad (9)$$

Where:

- x: the input value being evaluated for membership.
- a: the lower limit where the membership starts to increase from 0.
- b: the start of the plateau, where membership reaches 1 (full membership).
- c: the end of the plateau, where membership remains at 1.
- d: the upper limit where membership decreases back to 0.
- $\mu(x;a,b,c,d)$: the membership function, representing the degree of membership of $x$in the fuzzy set ([0، 1]).

C. Gaussian Membership Function

$$\mu(x;c,\sigma) = \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \quad (10)$$

Where:

- x: the input value being evaluated for membership.
- c: the center (mean) of the Gaussian function; the point where membership is maximum ($\mu = 1$).
- σ: the standard deviation (spread) of the Gaussian; controls the width of the curve. A larger σmeans a wider curve (more gradual decrease), and a smaller σmeans a narrower curve (steeper decrease).
- $\mu(x;c,\sigma)$: the membership function, giving the degree of membership of $x$in the fuzzy set, ranging from 0 to 1.

Gaussian membership functions provide smooth transitions and are often used to represent concepts such as “approximately equal to 50,” with center $c = 50$ and spread controlled by σ.

4- Fuzzy Numbers

A fuzzy number is a special class of fuzzy sets defined on the real line $\mathbb{R}$, satisfying the following properties:

A. Normality:

$$\exists x \in \mathbb{R} \text{ such that } \mu(x) = 1.$$

Where:

- $\exists$: "there exists"; a mathematical quantifier indicating that at least one value satisfies the condition.
- $x \in \mathbb{R}$: $x$is a real number, i.e., an element of the real number set $\mathbb{R}$.
- $\mu(x)$: the membership function of a fuzzy set, which assigns a degree of membership to x, ranging from 0 to 1.

B. Convexity:

$$\mu(\lambda x_1 + (1-\lambda)x_2) \geq \min\{\mu(x_1), \mu(x_2)\}, \forall x_1, x_2 \in \mathbb{R}, \lambda \in [0,1] \quad (11)$$

Where:

- $\mu(x)$: the membership function of a fuzzy set, giving the degree of membership of x(between 0 and 1).
- $x_1, x_2 \in \mathbb{R}$: any two real numbers in the universe of discourse.
- $\lambda \in [0,1]$: a weighting factor that determines a point on the line segment between $x_1$and x_2.

C. Upper semi-continuity.

Common Types of Fuzzy Numbers

- Triangular Fuzzy Number (TFN)

$$\tilde{A} = (a, b, c)$$

- Trapezoidal Fuzzy Number (TrFN)

$$\tilde{A} = (a, b, c, d)$$

For example, the triangular fuzzy number $\tilde{A} = (2,5,8)$ represents a quantity centered around 5, with possible variation between 2 and 8. Arithmetic operations on fuzzy numbers are commonly defined using Zadeh's extension principle (Zadeh, 1975). Given two fuzzy numbers $\tilde{A}$ and $\tilde{B}$ and a binary operation $\oplus$, the resulting fuzzy set $\tilde{C}$ is defined as:

$$\mu_{\tilde{C}}(z) = \sup_{x \oplus y = z} \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)) \quad (12)$$

Where:

- $\mu_{\tilde{C}}(z)$: the membership function of the resulting fuzzy number $\tilde{C}$after performing the operation $\oplus$(e.g., addition, subtraction) on fuzzy numbers $\tilde{A}$and $\tilde{B}$.
- $\tilde{A}, \tilde{B}$: fuzzy numbers or fuzzy sets with membership functions $\mu_{\tilde{A}}(x)$and $\mu_{\tilde{B}}(y)$, respectively.
- z: a possible value of the resulting fuzzy number after applying $\oplus$to elements of $\tilde{A}$and $\tilde{B}$.
- $x \oplus y = z$: the arithmetic operation defining the relationship between elements $x$of $\tilde{A}$and $y$of $\tilde{B}$that produce z. For example, if $\oplus$is addition, then $x + y = z$.

5- Linguistic Variables

Another fundamental concept introduced by Zadeh (1965) [19] is the linguistic variable, whose values are words or phrases rather than numerical values. A linguistic variable is formally defined as a 5-tuple:

$$L = (x, T(x), U, G, M) \quad (13)$$

where:

- x: name of the variable (e.g., *Temperature*),

- $T(x)$: set of linguistic terms (e.g., {cold, warm, hot}),
- U: universe of discourse (e.g., [0، 100] °C),
- G: syntactic rule for generating new terms (e.g., modifiers such as *very* or *slightly*),
- M: semantic rule mapping each linguistic term to a fuzzy set.

Fuzzy set theory extends classical set theory by allowing gradual membership, providing a powerful framework for modeling vagueness and uncertainty. The fundamental components (membership functions, fuzzy numbers, and linguistic variables) form the basis for fuzzy inference systems and intelligent decision-making models widely used in science and engineering.

6- Fuzzy Output Interpretation in GRNN

When a Generalized Regression Neural Network (GRNN) is integrated with fuzzy logic, the network output may not be a single crisp value but a fuzzy output [15].

- Fuzzy outputs represent uncertainty, vagueness, or linguistic information
- Examples: *Low risk*, *medium load*, *High temperature*
- To support decision-making, fuzzy outputs must be interpreted or defuzzified

Fuzzy output interpretation is the final stage of a Fuzzy-GRNN system. Let the output variable Y be described by a fuzzy set:

$$\tilde{Y} = \{(y_i, \mu_{\tilde{Y}}(y_i)) \mid y_i \in U\} \quad (14)$$

where:

- U : universe of discourse of the output
- $\mu_{\tilde{Y}}(y_i) \in [0,1]$: membership degree of y_i

In a Fuzzy-GRNN, the fuzzy output membership is computed as a kernel-weighted aggregation:

$$\mu_{\tilde{Y}}(y) = \frac{\sum_{k=1}^{N} \phi_k(\tilde{x})\, \mu_{B_k}(y)}{\sum_{k=1}^{N} \phi_k(\tilde{x})} \quad (14)$$

where:

- $\phi_k(\tilde{x})$: fuzzy kernel response of the k-th neuron
- $\mu_{B_k}(y)$: membership of training output y_k in fuzzy set B_k
- N : number of training samples

This produces a fuzzy output distribution instead of a single numeric value.

7- Defuzzification Methods

To obtain a crisp output y^*, the fuzzy set $\tilde{Y}$ must be defuzzified.

A. Centroid (Center of Gravity) Method

$$y^* = \frac{\sum_i y_i\, \mu_{\tilde{Y}}(y_i)}{\sum_i \mu_{\tilde{Y}}(y_i)} \quad (15)$$

Where:

- y^*: the crisp output value obtained after defuzzifying the fuzzy set $\tilde{Y}$.
- y_i: the discrete values in the universe of discourse of the output variable.
- $\mu_{\tilde{Y}}(y_i)$: the membership degree of $y_i$in the fuzzy output set $\tilde{Y}$, ranging from 0 to 1.

The method under consideration is highly regarded for several key characteristics. The system establishes itself as the leading method of its field because it demonstrates reliable performance which practitioners use as a standard. The system produces outputs which maintain smoothness and stability to enable better result interpretation and consistent output handling. The method works best for regression tasks which enables it to function as an all-purpose tool for studying variable relationships. The system achieves widespread use in different analytical applications through its practical utility which emerges from its combination of essential features.

B. Mean of Maximum (MOM)

$$y^* = \frac{\sum_{i \in \arg\max \mu_{\tilde{Y}}(y_i)} y_i}{|\arg\max \mu_{\tilde{Y}}(y_i)|} \quad (16)$$

Where:

- y^*: the crisp output value obtained after defuzzifying the fuzzy set $\tilde{Y}$.
- y_i: the discrete values in the universe of discourse of the output variable.
- $\mu_{\tilde{Y}}(y_i)$: the membership degree of $y_i$in the fuzzy output set $\tilde{Y}$.

The method approaches its objectives through two main principles which include simple design and efficient function. The system achieves its operational efficiency through implementation of maximum membership value extraction. The method requires less computational resources than centroid-based methods because it generates results through direct physical measurements from the target area. The method offers practical advantages because of its high operational efficiency and simple handling requirements in multiple different use cases.

C. Bisector Method

The method establishes y^*by determining the value which divides the membership function's area into two equal parts. The method delivers symmetrical results through its ability to produce equal fuzzy set representations. The method uses area measurements instead of peak points and particular data points to deliver an accurate and natural estimate of central tendency in symmetric distributions.

D. Linguistic Interpretation of Fuzzy Outputs

Fuzzy outputs need numerical defuzzification for processing but the results can be expressed through linguistic terms which enable users to understand the outcomes better. The system outputs multiple fuzzy set memberships which include Low Medium and High to represent its built-in system uncertainties. The membership values show how much trust exists in each category which enables detailed understanding of the data. The linguistic system enables users to read the reasoning behind system decisions while making it simpler to explain operational choices to others.

8- Advantages of Fuzzy Output Interpretation in GRNN

The fuzzy–GRNN framework provides multiple essential benefits which improve both prediction performance and result comprehension. The system demonstrates its effectiveness in predicting uncertain outcomes through its ability to show different membership degrees instead of providing fixed prediction values. The system provides output level changes which occur through gradual transitions instead of sudden jumps, which enables users to move between three different linguistic states, which include low, medium, and high [13]. The model establishes a connection between numerical results and fuzzy linguistic expressions, which enhances the ability to understand neural network outputs, making them easier to comprehend for human viewers. The fuzzy–GRNN provides interpretable results which enable control systems and decision support systems and expert systems to operate with transparent reasoning processes. The system creates a framework which uses numerical accuracy to perform linguistic reasoning while it combines the neural network learning function with the human-like reasoning framework of fuzzy logic.

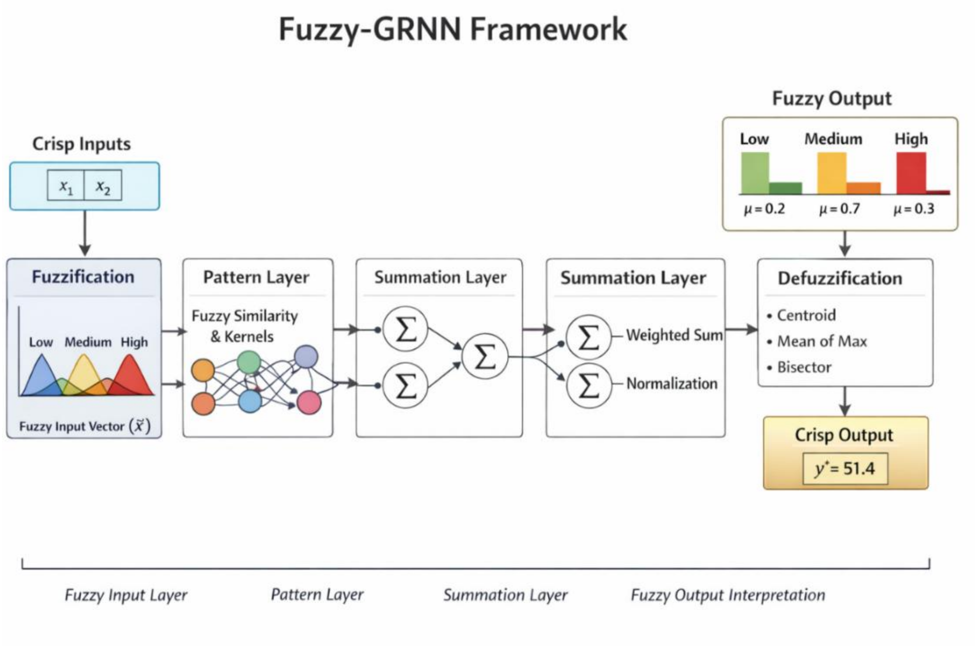

Figure (1): Fuzzy-GRNN framework

The system employs fuzzy logic together with Generalized Regression Neural Networks to handle uncertain input data while generating results which people can comprehend. The process starts with actual input data which undergoes fuzzification to generate fuzzy input vectors that display membership values for Low, Medium, and High categories. The system processes fuzzy inputs through its pattern layer by applying fuzzy similarity measures and kernel functions while the system aggregates results through its summation layers using weighted sum and normalization methods. The network produces a fuzzy output which contains membership degrees for each category and defuzzification methods such as Centroid, Mean of Max, or Bisector convert this output into a single crisp value. The method combines GRNN's predictive accuracy with fuzzy logic's understandable explanation which makes it suitable for regression analysis that needs both uncertainty handling and result interpretation [9].

2nd: Data Analysis and Results

1- Data Description

The research examines daily data which the MASS Cement Industry in Sulaimanyah, Kurdistan-Iraq collected between January 1 and December 31 of 2023. The study utilized six explanatory variables which were measured on a ratio scale are considered: **Limestone (LS)** (high in $CaCO_3$, crucial for clinker), **Clay** (provides SiO_2, Al_2O_3, and Fe_2O_3), **Iron** (Fe_2O_3, essential for clinker phases), **Sand** (SiO_2-rich, key for silicate formation), and **HFO Consumption L1** (Heavy Fuel Oil used in kilns). The response variable is level of **Nox 450 mg/nm3 emissions**, a critical environmental measure indicating combustion efficiency and safety concerns in cement production.

2- GRNN Results

The output of neural network weight initialization, the object contains the randomly initialized weights for a neural network model. A **6x10 matrix** in the table (1) corresponds to the weights between the **input layer** (6 neurons) and the **hidden layer** (10 neurons)

Table (1): Represents the weight matrices and biases for the input and hidden layer of the neural network.

	Hidden neuron									
Input neuron	−1.644	−2.005	0.195	3.422	−12.217	29.169	0.581	−3.776	4.207	−0.088
	−0.895	5.93	−1.420	−3.141	21.095	2.697	−1.621	7.557	−2.996	2.223
	11.295	1.743	0.829	1.001	−267.233	16.22	−1.622	−53.666	0.703	−7.174
	5.945	6.484	4.87	−0.384	176.442	12.801	−5.587	48.995	−0.156	−3.167
	14.971	6.173	3.025	0.014	−120.162	−8.753	−4.487	176.897	−0.802	−7.838
	−25.978	−2.919	−2.440	2.947	333.812	−9.738	2.579	51.431	6.476	17.825

The input layer of the neural network connects to the hidden layer through weight matrix estimates and bias terms, which Table (1) presents as its explanation. The model demonstrates its ability to capture strong nonlinear relationships between inputs and hidden neurons through its wide range of weight values, which show different signs and magnitudes. The connection between the fifth and sixth rows to multiple hidden neurons contains links that reach very high absolute values, which show that these links have direct power to manage hidden layer activation, because they create the main pathways for the network to form its internal model. The network demonstrates its capacity to learn asymmetric and competing effects from predictors through positive weights, which activate hidden neurons, and negative weights, which inhibit their function. The model gains improved capacity to represent complicated patterns through non-zero bias terms, which move the hidden neuron activation thresholds, thereby increasing its activated potential in complex scenarios.

Table (2): Represents the weight matrices and biases for the hidden and output layer of the neural network.

	Output Layer
Hidden Layer	0.332
	−1.333
	−1.781
	−2.008
	−4.883
	0.355
	5.165
	−3.240
	0.489
	3.886
	−1.501

The hidden layer to output layer connections of the neural network show their estimated weight values and bias values in Table 2. The bias term establishes the baseline level of the output, while the hidden–output weights quantify the relative contribution of each hidden neuron to the final prediction. The positive and negative coefficients show that different hidden neurons have either strengthening or weakening effects on the output. The model output depends on neurons with larger absolute weights because they determine the output value through their encoded nonlinear patterns that reflect the most important data learned from the model. Hidden neurons with smaller magnitudes make their contributions through their active engagement in other tasks.

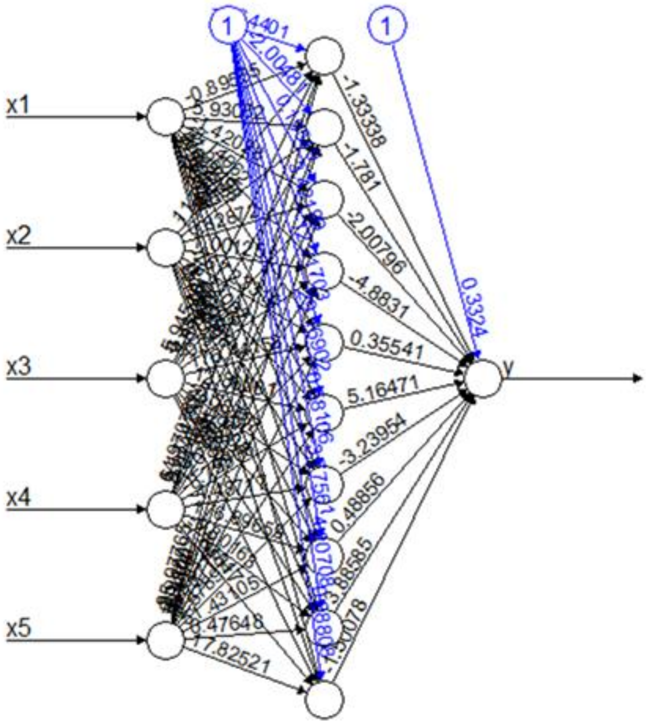

Figure (2): Represent a Generalized Regression Neural Network (GRNN)

The multilayer perceptron designed to map five input variables from X1 to X5 towards a single output Y is depicted in Figure 2 as a feedforward artificial neural network. The system uses three layers which comprise an input layer containing five nodes, a hidden layer with nine neurons, and an output layer that includes one neuron. The connection between these two layers carries a weight value that shows how strong the connection is between the nodes, while the blue lines from nodes 1 show that bias terms were added which let the model change its activation function to better match the data (0.3324 for the output layer). The system functions as a supervised learning system which enables the network to learn non-linear patterns in input data for predicting both continuous and categorical target values.

3- Fuzzy GRNN Results

The output of neural network weight initialization after entering fuzzy input to the model, the object contains the randomly initialized weights for a neural network model. A **11x10 matrix** in the table (3) corresponds to the weights between the **input layer** (11 neurons) and the **hidden layer** (10 neurons)

Table (3): Represents the weight matrices and biases for the fuzzy input and hidden layer of the neural network.

	Hidden neuron									
Input neuron	−0.405	0.773	0.627	1.049	−1.188	−0.279	2.298	−1.885	−1.244	−1.448
	0.522	−35.899	−2731.936	30.495	−31.495	−1598.722	133.661	27.377	−4.333	−184.273
	2.071	7.438	−0.262	−13.291	2.931	1.516	141.475	9.148	24.105	2.543
	0.002	0.203	0.203	37.499	7.153	−1.061	−13.745	−77.137	−51.871	56.993
	−3.684	51.609	−13.662	−29.613	3.386	1.135	−203.002	41.313	64.938	−282.903
	−0.057	−0.106	0.115	11.093	1.872	0.202	3.242	−4.170	−6.982	3.779
	55.46	−21.223	−16.834	−25.715	−62.857	70.397	−506.191	95.821	−2.352	−419.177
	−1.126	−0.572	−0.652	1.785	−1.644	−0.197	−0.256	−2.493	0.861	17.166
	1.304	5.458	4.305	56.151	−20.328	7.27	61.734	81.885	10.164	−555.300
	−0.446	−1.687	−0.208	0.585	−0.709	1.149	−0.045	−0.947	0.74	−1.008
	18.016	40.214	−4.531	−64.113	10.068	−2.329	−97.660	52.822	25.12	99.263

The table displays the neural network's fuzzy-transformed input layer weight and bias measurements which connect to the hidden layer. The weight distribution between the two networks shows a greater degree of spread because the fuzzy network entails more weight variations which give rise to increased sensitivity compared to traditional systems. The fuzzy input-hidden connections demonstrate excessive absolute weight values which exceed 1000 because particular membership-function inputs fully activate or block hidden neurons to extract regime-based data patterns and uncertainty patterns from the information. The presence of both large positive and negative weights demonstrates how fuzzy inputs produce dual effects which enhance and diminish hidden neuron activation, which helps the network create better models of complex nonlinear relationships. The bias terms maintain stable learning by modifying activation thresholds which enable the system to handle the variable nature of fuzzy inputs.

Table (4): Represents the weight matrices and biases for the hidden and output layer of the neural network for Fuzzy-GRNN

	Output Layer
Hidden Layer	0.735
	−1.596
	−1.185
	2.218
	−2.748
	2.498
	3.77
	−1.012
	0.756
	−2.308
	−0.433

The weight matrix and bias parameters which link the hidden layer to the output layer in the Fuzzy–GRNN model are presented in Table (4). The weights show positive and negative values which indicate that hidden neurons create output through their excitatory and inhibitory functions. The hidden units which have large positive weights of 3.77 and 2.498 produce strong positive effects on network output. The neurons with large negative weights of -2.748 and -2.308 produce strong negative effects on network output. The network uses diverse signs and varying magnitudes to combine information which fuzzy-activated hidden neurons provide in flexible and nonlinear ways. The substantial weight magnitudes indicate that the fuzzy inference system improves feature discrimination by enabling dominant fuzzy rules to shape the defuzzification and output generation process. The hidden-to-output weight structure proves that the Fuzzy–GRNN method successfully uses fuzzy membership representations to enhance nonlinear mapping accuracy and predictive performance when compared to traditional GRNN methods.

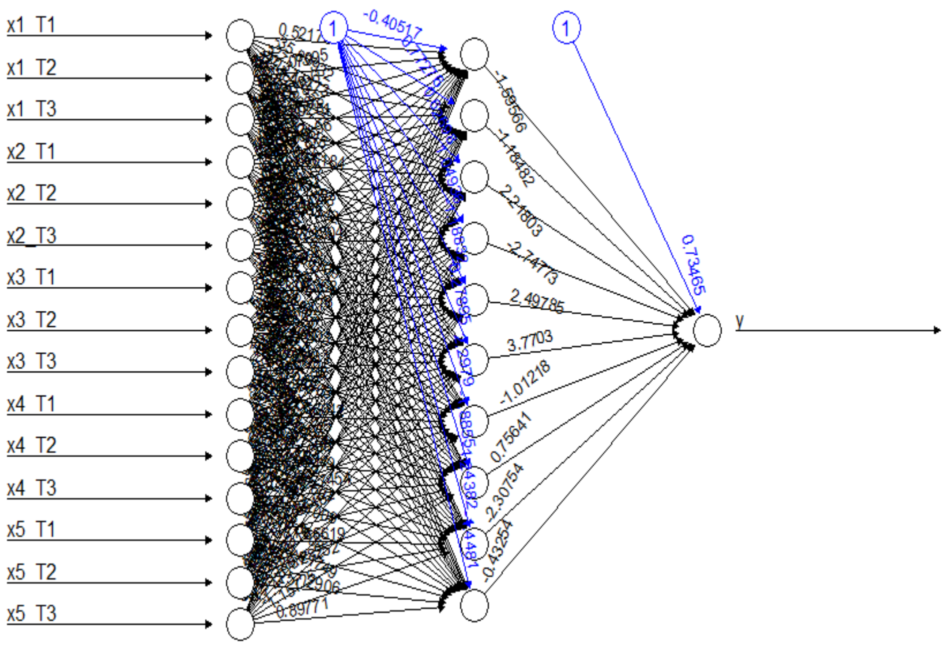

Figure (3): Represent a Fuzzy Generalized Regression Neural Network (Fuzzy-GRNN)

The advanced feedforward neural network shown in Figure 3 serves both time-series forecasting and dynamic system modelling requirements through its fifteen input nodes which operate with five different variables from three time points (T1, T2, T3) in the time series. The system consists of a hidden layer which contains ten neurons that handle the time-dependent data through multiple connections which lead to a single output Y. The current network design uses bias nodes in the same way as the earlier model, which connects through blue lines to nodes marked as "1" while the output layer uses a bias weight of 0.73465 to control prediction thresholds. The network can identify immediate input effects together with their historical patterns through this particular setup which includes lagged versions of all variables.

5- Models Compression:

Table (5) shows performance metrics for two models (GRNN and Fuzzy-GRNN). Two evaluation criteria are used and the results are:

Table (5): Criteria Compression for GRNN and Fuzzy-GRNN models with Relative Improvement

Metric	GRNN	Fuzzy–GRNN	Improvement (%)
RMSE	0.8501	0.6122	**27.99% ↓**
MAE	0.4917	0.3832	**22.07% ↓**

The table 5 comparison assesses the predictive accuracy between standard GRNN and Fuzzy–GRNN model through RMSE and MAE measurements. The results demonstrate that using fuzzy logic results in significant enhancements to prediction accuracy. The RMSE value decreased from 0.8501 for standard GRNN to 0.6122 for Fuzzy–GRNN which results in a 27.99% improvement. The MAE value decreased from 0.4917 to 0.3832 which resulted in a 22.07% improvement. The Fuzzy–GRNN model shows better prediction accuracy because it produces results which match actual data points at a higher level while decreasing the total error range. The fuzzy enhancement demonstrates its effectiveness through improvements which show better results in both error metrics because fuzzy membership functions work successfully with the GRNN system. The Fuzzy–GRNN model demonstrates better performance than traditional GRNN methods because it provides higher reliability and accuracy for tasks which involve nonlinear regression and predictive modelling.

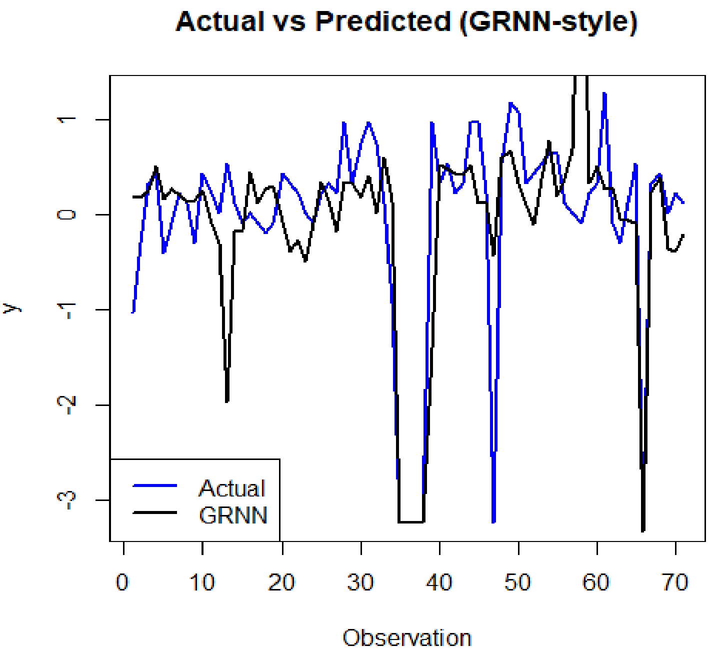

Figure (4): Explains Actual, GRNN values

The performance plot in Figure 4 shows the actual observed values which are represented by the blue line while the General Regression Neural Network (GRNN) model predictions are shown by the black line across 70 observations. The GRNN model demonstrates its capability to accurately predict main trends and major data direction changes through its ability to track more than just oscillatory data patterns which include the complete "trough" period from observation 35 to 40. The model demonstrates high accuracy in tracking actual data movement yet there are specific areas where peak values show differences and the model reacts too strongly to sudden data changes which include the downward spikes that show more model response than actual data. The GRNN model has achieved successful pattern recognition from the dataset because both lines remain closely aligned which enables accurate modeling of non-linear time-series and regression tasks.

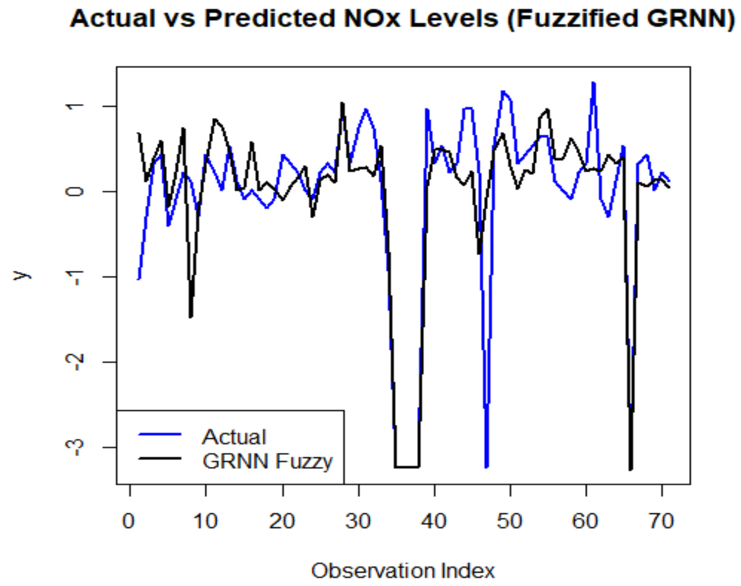

Figure (5): Explains Actual, Fuzzy-GRNN values

The Fuzzified General Regression Neural Network (GRNN) model shows its predictive power through its estimation of NOx levels across 70 observations which it presents through the model output (black line) and the actual recorded values (blue line). The GRNN framework shows improved environmental emissions data processing through fuzzy logic which enables the model to deal with uncertainty and non-linear relationships in the data as seen in its ability to track extreme fluctuations and major drops during the period of sustained low-level emissions between indices 35 and 40. The model shows slight deviations from actual data during peak intensity periods and shows minor delays in phase which results from its fast transition periods but the model still maintains strong correlation with actual data through its ability to track both the sudden spikes and the normal base patterns. Hybrid fuzzified approach enables sophisticated monitoring and forecasting of complex atmospheric pollutants through its high level of alignment which establishes a dependable tool for NOx tracking.

3rd: Conclusions and Recommendations

Conclusions

The research established that the six selected explanatory variables function as essential elements which determine cement production efficiency while they control Nox 450 mg/nm3 emission rates. The study found that the raw materials which included limestone and clay along with iron oxide were essential for clinker production because they determined the Nox emissions which resulted from the burning process. The HFO consumption variable indicated its importance in regulating the fuel efficiency and emissions in the kiln. The model (GRNN, before and after fuzzified input data) provided distinct insights into the prediction of level of Nox emissions, with the GRNN model with fuzzified input data showing better performance in both RMSE and MAE results compared to all other individual models. The research demonstrates that both GRNN and Fuzzy–GRNN can accurately model the nonlinear relationships which exist in NOx data, with Fuzzy–GRNN showing better results than standard GRNN. Fuzzification improves input sensitivity through weight analysis which enables complex pattern recognition. The performance metrics RMSE and MAE showed improvements of 28 percent and 22 percent respectively while visual comparisons confirm that Fuzzy–GRNN better tracks extreme fluctuations and nonlinear trends. The combination of fuzzy logic with GRNN improves predictive accuracy and operational resilience, which makes this method superior for nonlinear regression and environmental forecasting.

Recommendations

The research showed that Fuzzy–GRNN should be used to forecast NOx emissions and other environmental data which shows nonlinear and uncertain behavior. The network gains improved forecasting accuracy through fuzzy membership function integration because it enables better handling of both typical and extreme value fluctuations. The hybrid method developed in this study can monitor industrial processes while forecasting energy load and detecting air pollutants which require decision-making in scenarios with both nonlinear patterns and uncertainty.

References:

1. **Al-Mahasneh, A. J., Anavatti, S., Garratt, M., & Pratama, M.** (2018). Applications of general regression neural networks in dynamic systems. *Digital Systems, 1*, 133–154.
2. **Castellano, G., Scaringi, R., Vessio, G., & Zaza, G.** (2024, December). Integrating graph neural networks and fuzzy logic to enhance deep learning interpretability. In *Proceedings of the Fourth International Workshop on Multilingual Semantic Web (MSW 2024)*, Paris, France (pp. 11–13).
3. **Cigizoglu, H. K., & Alp, M.** (2006). Generalized regression neural network in modelling river sediment yield. *Advances in Engineering Software, 37*(2), 63–68.
4. **Enke, D., & Mehdiyev, N.** (2013). Stock market prediction using a combination of stepwise regression analysis, differential evolution-based fuzzy clustering, and a fuzzy inference neural network. *Intelligent Automation & Soft Computing, 19*(4), 636-648.
5. **He, H., Kojima, S., Omura, T., Maruta, K., & Chang-Jun, A. H. N.** (2021). Generalized regression neural network based channel identification and compensation using scattered pilot. *Radioengineering, 30*(4).
6. **Hussein, M. M. F.** (2022). Comparison of time series models before and after using wavelet shrinkage filtering to forecast the amount of natural gas in Iraq. *Cihan University–Erbil Scientific Journal, 6*, 32–46.
7. **Hussein, M. M. F., Saeed, A. A., & Mohamad, S. H.** (2023). Comparison of Markov chain and neural network models for forecasting population growth data in Iraq. *University of Kirkuk Journal of Administrative and Economic Sciences, 13*.
8. **Irfan, M., Koj, A., Thomas, H., & Sedighi, M.** (2016). Geographical general regression neural network (GGRNN) tool for geographically weighted regression analysis. *Geoprocessing2016*, 154–159.
9. **Jang, J. S. R., Sun, C. T., & Mizutani, E.** (1997). Neuro-fuzzy and soft computing—A computational approach to learning and machine intelligence [Book review]. *IEEE Transactions on Automatic Control, 42*(10), 1482–1484.
10. **Klir, G., & Yuan, B.** (1995). *Fuzzy sets and fuzzy logic*. Prentice Hall.
11. **Kopal, I., Labaj, I., Vršková, J., Harničárová, M., Valíček, J., Ondrušová, D., ... & Palková, Z.** (2022). A generalized regression neural network model for predicting the curing characteristics of carbon black-filled rubber blends. *Polymers, 14*(4), 653.
12. **Mamdani, E. H.** (1974, December). Application of fuzzy algorithms for control of simple dynamic plant. In *Proceedings of the Institution of Electrical Engineers* (Vol. 121, No. 12, pp. 1585–1588). IEE.
13. **Mhamad, A. J., Taher, H. A., & Taha, A. A.** (2025). A hybrid BAT-SVR methodology for forecasting Iraq's interest rates of commercial banks (IRCB) time series data. *Academic Journal of International University Erbil, 2*, 347–355.
14. **Nader, R. A., Karim, A. J., & Hussien, M. M. (2018).** Using Artificial Neural Networks and SPI Measure Techniques to Forecast the Risk of Drought in Iraq and Its Impact on Environment. *Journal of University of Human Development*, *4*(2), 69-77**.**
15. **Ross, T. J.** (2005). *Fuzzy logic with engineering applications*. John Wiley & Sons.
16. **Saeed, H. M. A. J., & Taher, H. A.** (2025). Optimizing the Grey Model (1,1) through neural networks for forecasting the contribution of agriculture to GDP in Iraq. *University of Kirkuk Journal of Administrative and Economic Sciences, 15*.
17. **Specht, D. F.** (1991). A general regression neural network. *IEEE Transactions on Neural Networks, 2*(6), 568–576.
18. **Taher, H. A., Mhamad, A. J., & Taha, A. A.** (2025). Optimization of support vector regression for improved prediction accuracy in cross-sectional data using genetic algorithms. *Journal of University of Raparin, 12*, 842–864.
19. **Zadeh, L. A.** (1965). Fuzzy sets. *Information and Control, 8*(3), 338–353.
20. **Zheng, Y., Xu, Z., Wu, T., & Yi, Z.** (2024). A systematic survey of fuzzy deep learning for uncertain medical data. *Artificial Intelligence Review, 57*(9), 230.
21. **Zimmermann, H. J.** (2011). *Fuzzy set theory—and its applications*. Springer Science & Business Media.