INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OPEN ACCESS

Received: 25-04-2024

Accepted: 15-05-2024

Published: 13-06-2024

Citation: Robinson PJ, Saranraj A (2024) Intuitionistic Fuzzy Gram-Schmidt Orthogonalized Artificial Neural Network for Solving MAGDM Problems. Indian Journal of Science and Technology 17(24): 2529-2537. https://doi.org/10.17485/IJST/v17i24.1386

* **Corresponding author**.

johnrobinson.ma@bhc.edu.in

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Intuitionistic Fuzzy Gram-Schmidt Orthogonalized Artificial Neural Network for Solving MAGDM Problems

P John Robinson[1]*, A Saranraj[2]

1 PG & Research Department of Mathematics, Bishop Heber College (Affiliated to Bharathidasan University), Tiruchirappalli, 620017, Tamil Nadu, India
2 PG & Research Department of Mathematics, St. Joseph's College (Affiliated to Bharathidasan University), Tiruchirappalli, 620017, Tamil Nadu, India

Abstract

Objectives: To propose a suitable decision-making model based on Intuitionistic Fuzzy sets (IFSs) and Gram-Schmidt orthogonalization process for Artificial Neural Network (ANN). **Methods:** The IFS data sets appearing in the form of matrices are aggregated using the available aggregation operators in the literature and then the collective aggregated information is processed through Gram-Schmidt orthogonalization for the revised input vectors which is then fed into the ANN algorithm following Delta Learning Rule for the next phase. The weight updation is performed through the ANN and the output is improvised. **Findings:** The proposed Gram-Scmidt Orthogonalization process is utilized in Intuitionistic Fuzzy Artificial Neural Network model. The Delta learning rule is utilized in the process of the Neural Network, where the Intuitionistic Fuzzy nature of the input data is transformed into a fuzzy data and then the ranking of the alternatives is done based on the weights updation through the learning phase of the ANN. Once the vector is trained out of the learning phase, it is then processed through the activation function for the final selection of the best alternative required of the Multiple Attribute Group Decision Making (MAGDM) problem posed in this work. To demonstrate the usefulness and applicability of this new method with the Gram-Schmidt process, the numerical example also adds more insight to the proposed methodology of ANN with the application of some Linear Space techniques. **Novelty:** Most of the research done on Intuitionistic Fuzzy Artificial Neural Network model are based on learning rules or using some other calculations. The proposed Gram-Schmidt Orthogonalization process is used to find the orthogonal basis that are used as input training vectors in the Delta learning rule for ANN.

Keywords: MAGDM; ANN; Aggregation operators; Learning Rules; Intuitionistic Fuzzy sets; Gram-Scmidt Orthogonalization

1 Introduction

Multiple Attribute Group Decision Making (MAGDM) problems are often used in the area of commerce, business, medicine, logistics, supply chain etc. MAGDM problems under uncertain situations is a grey area which has to be dealt with lot of carefulness and precision due to its ambiguity. Many researches these days are concentrating on the field of MAGDM under Intuitionistic Fuzzy Set (IFS) environment. Again, the problem in dealing with IFS-MAGDM problems depends mostly on the type of aggregation operator employed which are hugely available in the literature. Furthermore, the bottle-neck problem underlying in solving real life MAGDM problems requires more precision in pinpointing the best alternatives to be chosen at the end of the study. In the traditional methods of MAGDM, the role of Artificial Intelligence is minimal, which we have overcome in this work by solving the MAGDM completely using the novel Gram-Schmidt orthogonalization process proposed to produce a better input vector for the Artificial Neural Network (ANN). Artificial Intelligence (AI) and Machine Learning (ML) are closely related fields within computer science that have gained significant attention and applications in recent years. Artificial Neural Networks (ANNs) represent a fundamental paradigm in machine learning and artificial intelligence. Inspired by the structure of the human brain, ANNs are a computational model composed of interconnected nodes, or "neurons," organized in layers. Providing the ANN with a proper input vector is a challenging task especially when dealing vague data sets like the IFS which is the type of data set in this work. In order to create a proper consensus with the MAGDM problem, the ANN technique is coupled with MAGDM to provide better decision-making strategies. The IFS information in this work is processed through the ANN layers, with each neuron performing simple computations. Through a process of learning from data, ANNs can discover complex patterns and relationships, making them adept at tasks like pattern recognition, classification, and regression. Intuitionistic Fuzzy (IF) ANN were extensively worked by authors in [(1)], [(2)], [(3)] and [(4)]. Aggregation operators for Linguistic Intuitionistic Fuzzy Sets are proposed and utilized in [(5)]. Basic concepts of Linear space techniques are detailed to a large extent in [(6)]. Triangular Intuitionistic Fuzzy MAGDM problem is discussed in [(7)]. In [(8)] the method that evaluates a trained ANN is proposed. In [(9)] and [(10)], some aggregation operators for Trapezoidal Intuitionistic Fuzzy numbers are presented. Aggregation operators for MAGDM problems and calculation of attribute weights were much developed and utilized by author in [(11)]. Authors in [(12)] and [(13)] utilized machine learning applications with fuzzy centered real-life problems and solved them using novel techniques. In this work, we have improvised the ANN model of [(14)] and have proposed a new ANN where the inputs can be derived using some linear space techniques like Gram-Schmidt orthogonalization process which seems to be more effective and reasonable especially when the decision maker provides incomplete information about the problem statement. When comparing with the earlier methods which directly employed the given data as input vectors for ANN, the current method proposed in this paper processes the IFS data using Gram-Schmidt Orthogonalization for ANN which is pioneering, novel and will relieve the ambiguity arising in the decision process.

2 Methodology

The method of computation proposed in this work is novel when compared to the earlier methods in the literature ([(1)], [(3)], [(5)], [(8)] and [(14)]) since none of the methods utilized an orthogonalized vector for the input of ANN which contributes a benchmark in the field of ANN guided solving MAGDM problems. When it comes to solving MAGDM problems using ANN, which is only less concentrated so far, this work will lead to a new method of applying Gram-Schmidt process to ANN which will solve the MAGDM problems more effectively. The improvement in the methodology can be clearly viewed from the comparison of the current method with the one proposed in [(14)]. The existing method in [(14)] is modified with the input received from utilizing Gram-Schmidt orthogonalization process is much effective than the previous traditional techniques. The developed concepts are presented in the following sections.

2.1 Learning and Activation phase in ANN

An ANN's ability to learn is its most important feature. Learning, often known as training, is the process by which a neural network adjusts its parameters appropriately in response to a stimulus, producing the desired response. In Artificial Neural Networks (ANNs), supervised learning is a machine learning technique where the model is trained on labelled data with known input-output pairings. The architecture of the network, including its layers and activation procedures, is specified. Input data is sent through the network during training, and the output is compared to actual labels using a loss function. Gradients are calculated using backpropagation, which enables weight changes using optimization techniques like stochastic gradient descent. Unsupervised learning takes place without outside assistance, much like how a tadpole or early fish learns on their own. It entails independent learning without the influence of an instructor. In this method, similar Artificial Neural Networks (ANNs) arrange input vectors without taking into account individual appearances or group affiliations. The network groups input patterns into clusters during training. The neural network generates an output response when it receives a new input and categorizes it accordingly. In the context of Reinforced Learning, the network receives feedback indicating its performance, such as being "50% correct". Reinforcement gaming involves learning through negative reinforcement, while reinforcement learning, and reinforcement signal are based on positive rewards. An activation function in Artificial Neural Networks (ANNs) is a mathematical operation applied to the output of each neuron. Its primary role is to introduce non-linearity into the model, enabling the network to learn complex relationships. There are many activation functions available out of which a suitable can be chosen.

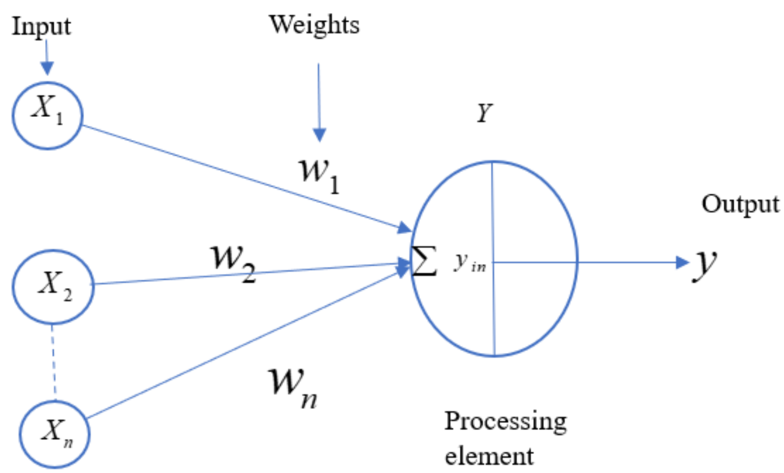

Fig 1. A simple architecture of Artificial Neural Network

2.2 Orthogonality and Orthonormality in Linear Space

Calculating the closest vector on a subspace to a given vector is typically required to solve the matrix equation Ax=b and find approximate solutions to other equations. The question then becomes one of orthogonality: identifying the vectors that are perpendicular to the subspace. The property of the nearest point is that the difference between the two points is perpendicular to, or orthogonal to, the subspace. We must thus create concepts of length, distance, and orthogonality. This section's fundamental construction is the dot product, which calculates a vector's length and measures the angles between vectors.

Definition 2.2.1 [6] The dot product of two vectors *u, v* in $\mathbb{R}^n$ is

$$u.v = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} . \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} = u_1v_1 + u_2v_2 + \cdots + u_nv_n$$

Think of $u,\ v$ as column vectors, this is the same as u^Tv.

For example,

$$\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} . \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} = 1.2\ +\ 2.3\ +\ 1.1\ =\ 9.$$

Notice that the dot product of two vectors is a scalar. As long as you keep in mind that you can only dot two vectors together and that the outcome is a scalar, you may perform arithmetic with dot products essentially as normal. Let $u,\ v,\ w$ be vectors in R^n and let c be a scalar.

1. Commutativity: $u.v = v.u$.

2. Distributivity with addition:$(u+v).w = u.w+v.w$.
3. Distributivity with scalar multiplication: $(cu).v = c(u.v)$.

The vector's dot product with itself is a significant particular case:

$$\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = u_1^2+u_2^2+\cdots+u_n^2.$$

Therefore, for any vector u, we have: i)$u.u \geq 0$ ii) $u.u = 0 \Leftrightarrow u = 0$. A vector u's length in $\mathbb{R}^n$ is equal to the number $\|u\| = \sqrt{(u.u)} = \sqrt{(u_1^2+u_2^2+\cdots+u_n^2)}$..

If u.v=0, then two vectors, x,y in $\mathbb{R}^n$, are orthogonal or perpendicular. Let W be a subspace of $\mathbb{R}^n$. Its orthogonal complement is the subspace $W^{\perp} = \{u \text{ in } \mathbb{R}^n |\, u.w = 0 \, for\, all\, w \text{ in } W\}$. The perpendicular line W is the orthogonal counterpart of a line W across the origin in R^2. Let W be a $\mathbb{R}^n$ subspace. Then the following are regarding orthogonal complements: i)$W^{\perp}$ is also a subspace of $\mathbb{R}^n$, ii) $(W^{\perp})^{\perp} = W$, iii) $dim(W)+dim(W^{\perp}) = n$.

2.2.1 Orthogonal Decomposition

Let W be subspace of $\mathbb{R}^n$ and let u be a vector in $\mathbb{R}^n$. Then compute the closest vector u_W to u in W, where the vector u_W is called the orthogonal projection of u onto W. The closest vector to u on W means that the difference $u-u_W$ is orthogonal to the vectors in W. In other words, if $u_{W^{\perp}} = u-u_W$, then we have $u = u_W+u_{W^{\perp}}$, where u_W is in W and $u_{W^{\perp}}$ is in $W^{\perp}$. The first order of business is to prove that the closest vector always exists.

Theorem 2.2.1: [6] **(Orthogonal decomposition).**

Let W be a subspace in $\mathbb{R}^n$ and let u be a vector in $\mathbb{R}^n$. Then u can be uniquely represented as: $u = u_W+u_{W^{\perp}}$, where u_W is the closest vector to u on W and $u_{W^{\perp}}$ is in $W^{\perp}$. If $L = Span\{x\}$ is a line, then $u_L = \frac{x.u}{x.x}x$ and $u_{L^{\perp}} = u-u_L$ for any vector u.

As a simple example we shall compute the orthogonal projection of $u = \begin{pmatrix} -6 \\ 4 \end{pmatrix}$ onto the line L spanned by $x = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$, and find the distance from u to L. First, we find

$u_L = \frac{u.x}{x.x}X = \frac{-18+8}{9+4}\begin{pmatrix} 3 \\ 2 \end{pmatrix} = -\frac{10}{13}\begin{pmatrix} 3 \\ 2 \end{pmatrix}$ and $u_L^{\perp}) = u-u_L = \frac{1}{13}\begin{pmatrix} -48 \\ 72 \end{pmatrix}$

The distance from u to L is $\|u_L^{\perp}\| = \frac{1}{13}\sqrt{48^2+72^2} \approx 6.656$..

2.3 Computation of Gram-Schmidt Process

This subsection provides the Gram-Schmidt Process, a method for determining a subspace's orthogonal basis. Let the basis for subspace W of $\mathbb{R}^n$ be $\{a_1, a_2, \ldots, a_m\}$. Then the Orthogonal Basis by the process of Gram-Schmidt is given by:

i) $a_1 = b_1$ ii) $a_2 = (b_2)_{Span\{a_1\}^{\perp}} = b_2 - \frac{b_2.a_1}{a_1.a_1}a_1$

iii) $a_3 = (b_3)_{Span\{a_1,a_2\}^{\perp}} = b_3 - \frac{b_3.a_1}{a_1.a_1}a_1 - \frac{b_3.a_2}{a_2.a_2}a_2$

and proceeding further $a_m = (b_m)_{Span\{a_1,a_2,\ldots,a_{m-1}\}^{\perp}} = b_m - \sum_{i=1}^{m-1} \frac{b_m.a_i}{a_i.a_i}a_i$.

Then $\{a_1, a_2, \ldots, a_m\}$ is an orthogonal basis for the same subspace W.

2.4 MAGDM Problem Solving Using Intuitionistic Fuzzy (IF) Gram-Schmidt ANN

Pseudo-code for IF-Gram_Schmidt-ANN:

C_n: *n Matrix itemset of size k x m*
Input {*Intuitionistic Fuzzy Decision Matrices*}
A_n = *{Collection of n Matrices of size k};*
//* Aggregation Phase*//
Compute {*P-IFWG aggregator & the Initial Weight Vector*}
For$(n = 1;\ A_n \neq \emptyset;\ n++)$ **do begin**
Generate {*Individual Preference Intuitionistic Fuzzy Decision Matrices,* X_n}
//* X_N *is the collection of Individual Preference IF-Decision Matrices* *//
Generate {*Intuitionistic Fuzzy Attribute Weight Vector*}
While $i \leq m$**do** {*Defuzzify the IF column matrix into Fuzzy Column matrix*}

Generate {*Collective Overall Preference Linguistic Intuitionistic Fuzzy Decision Matrices using Fully Linguistic Intuitionistic Fuzzy Weight Vector, W^T*}
//*Improvise the input vector by GRAM-SCHMIDT Orthogonalization*//
Input vector $\left\{u_m = (v_m)_{Span\{u_1,u_2,\ldots,u_{m-1}\}^{\perp}} = v_m - \sum_{i=1}^{m-1} \frac{v_m \cdot u_i}{u_i \cdot u_i} u_i\right\}$
//* Learning Phase*//
Generate {*Weight Matrix by IF-Delta Rule*}
$net^n = (w^n)^T X_n, O^n = \frac{2}{1+exp(-net^n)} - 1, f^{'}(\mathrm{net}^n) = \frac{1}{2}\left[1-(O^n)^2\right],$
Update weights for next step $\{W^{n+1} = c\,(d_n - O^n) f^{'}(\mathrm{net}^n) X_n + W^n\}$
Continue the weight updation until the error is minimized to a desired level
//*Activation function*//
Fix {*The Threshold Value-Binary Step Function*}
While $Activated\ values \geq Threshold$ **do**
Generate {*Binary Matrix for final Decision with values exceeding the Threshold*}
Output {*Best Alternative(s) to be chosen*}
{the final decision variable can be converted into crisp variable and computations can be performed}.
End

3 Results and Discussion

The numerical illustration which will be presented in the following is the same one followed in [14] and the advantage here is that the data will be orthogonalized using Gram-Schmidt process and which in turn will be treated as the input for ANN process. Comparison of the numerical illustration with the earlier methods are presented at the end of the section.

3.1 Numerical Illustration of ANN With Gram-Schmidt Process & Delta rule

Let us suppose there is an investment company and a panel with five possible alternatives to invest the money: A_1 is a car company; A_2 is a food company; A_3 is a computer company; A_4 is an arms company; A_5 is a TV company. There are four attributes: G_1 is the risk analysis; G_2 is the growth analysis; G_3 is the social-political Impact analysis; G_4 is the environmental impact analysis. There are three decision makers and the decision matrices as listed in the following matrices $\widetilde{R}_k = \left(\tilde{r}_{ij}^{(k)}\right)_{m\times n}, k = (1,2,3)$ as follows [14]:

$$\widetilde{R_1} = \begin{pmatrix} (0.4,0.3)(0.5,0.2)(0.2,0.5)(0.1,0.6) \\ (0.6,0.2)(0.6,0.1)(0.6,0.1)(0.3,0.4) \\ (0.5,0.3)(0.4,0.3)(0.4,0.2)(0.5,0.2) \\ (0.7,0.1)(0.5,0.2)(0.2,0.3)(0.1,0.5) \\ (0.5,0.1)(0.3,0.2)(0.6,0.2)(0.4,0.2) \end{pmatrix};$$

$$\widetilde{R_2} = \begin{pmatrix} (0.5,0.4)(0.6,0.3)(0.3,0.6)(0.2,0.7) \\ (0.7,0.3)(0.7,0.2)(0.7,0.2)(0.4,0.5) \\ (0.6,0.4)(0.5,0.4)(0.5,0.3)(0.6,0.3) \\ (0.8,0.1)(0.6,0.3)(0.3,0.4)(0.2,0.6) \\ (0.6,0.2)(0.4,0.3)(0.7,0.1)(0.5,0.3) \end{pmatrix}:$$

$$\widetilde{R_3} = \begin{pmatrix} (0.4,0.5)(0.5,0.4)(0.2,0.7)(0.1,0.8) \\ (0.6,0.4)(0.6,0.3)(0.6,0.3)(0.3,0.6) \\ (0.5,0.5)(0.4,0.5)(0.4,0.4)(0.5,0.4) \\ (0.7,0.2)(0.5,0.4)(0.2,0.5)(0.1,0.7) \\ (0.5,0.3)(0.3,0.4)(0.6,0.2)(0.4,0.4) \end{pmatrix}.$$

Following the computations of P-IFWG[14], the collective overall preference values are:

$$\widetilde{R_1} = \begin{pmatrix} (0.20907, 0.46232) \\ (0.45470, 0.24905) \\ (0.45322, 0.24061) \\ (0.22865, 0.34319) \\ (0.42247, 0.18478) \end{pmatrix}; \widetilde{R_3} = \begin{pmatrix} (0.32464, 0.56734) \\ (0.55958, 0.35113) \\ (0.55373, 0.34085) \\ (0.34999, 0.43600) \\ (0.52483, 0.24614) \end{pmatrix}; \widetilde{R_3} = \begin{pmatrix} (0.20907, 0.67535) \\ (0.45470, 0.45404) \\ (0.45322, 0.44119) \\ (0.22865, 0.54176) \\ (0.42247, 0.34673) \end{pmatrix}$$

By defuzzification method (r_{ij} = 1-μ-γ) we get the defuzzified values of the collective overall preference values as follows:

$$\widetilde{R_1} = \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix}; \widetilde{R_3} = \begin{pmatrix} 0.10802 \\ 0.08927 \\ 0.10541 \\ 0.214 \\ 0.22902 \end{pmatrix}; \widetilde{R_3} = \begin{pmatrix} 0.11557 \\ 0.09126 \\ 0.10558 \\ 0.22959 \\ 0.2308 \end{pmatrix}$$

The defuzzified collective overall preference values are treated as the Input Training Vectors and denoted as follows:

$$X_1 = \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix}; X_2 = \begin{pmatrix} 0.10802 \\ 0.08927 \\ 0.10541 \\ 0.214 \\ 0.22902 \end{pmatrix}; X_3 = \begin{pmatrix} 0.11557 \\ 0.09126 \\ 0.10558 \\ 0.22959 \\ 0.2308 \end{pmatrix}$$

Applying Gramm-Schmidt Process, let,

$w_1 = X_1^T = [\ 0.32861 \quad 0.29625 \quad 0.30616 \quad 0.42816 \quad 0.39274\ ]$,

$w_2 = X_2^T = [\ 0.10802 \quad 0.08927 \quad 0.10541 \quad 0.214 \quad 0.22902\ ]$,

$w_3 = X_3^T = [\ 0.11557 \quad 0.09126 \quad 0.10558 \quad 0.22959 \quad 0.2308\ ]$.

Now, $v_1 = w_1$. i.e., $v_1 = [0.32861\ ,\ 0.29625\ ,\ 0.30616\ ,\ 0.42816\ ,\ 0.39274\,]$

$$v_2 = w_2 - \tfrac{\langle w_2, v_1 \rangle}{\|v_1\|^2} v_1 \, \langle w_2, v_1 \rangle$$
$$= 0.035496 + 0.026446 + 0.032272 + 0.091626 + 0.089945 = 0.275785$$

$$\|v_1\|^2 = \langle v_1, v_1 \rangle$$

$$\|v_1\|^2 = 0.107984 + 0.087764 + 0.093734 + 0.183321 + 0.154248 = 0.626961$$

$$\frac{\langle w_2, v_1 \rangle}{\|v_1\|^2} = 0.439876$$

$$\frac{\langle w_2, v_1 \rangle}{\|v_1\|^2}.v_1 = [0.144548\ \ 0.130313\ \ 0.134672\ \ 0.188337\ \ 0.172757\,]$$

$$v_2 = w_2 - \left(\frac{\langle w_2, v_1 \rangle}{\|v_1\|^2}.v_1 \right) = [-0.036528 \quad -0.041043 \quad -0.029262 \quad 0.025663 \quad 0.056263\ ]$$

$$Now,\ v_3 = w_3 - \left(\frac{\langle w_3, v_1 \rangle}{\|v_1\|^2}.v_1 \right) - \left(\frac{\langle w_3, v_2 \rangle}{\|v_2\|^2}.v_2 \right)$$

$\|v_2\|^2 = 0.001334 + 0.001684 + 0.000856 + 0.000658 + 0.003165 = 0.007697$

$\langle w_3, v_1 \rangle = [0.037977 + 0.027036 + 0.032324 + 0.098301 + 0.090644] = 0.286282$

$\langle w_3, v_2 \rangle = [-0.004222 - 0.003746 - 0.003089 + 0.005892 + 0.012986] = 0.007821$

$\frac{\langle w_3, v_1 \rangle}{\|v_1\|^2} = 0.456618, \ \frac{\langle w_3, v_2 \rangle}{\|v_2\|^2} = 1.016110$

$\frac{\langle w_3, v_1 \rangle}{\|v_1\|^2}.v_1 = [0.150049, 0.135273, 0.139798, 0.195506, 0.179332]$

$\frac{\langle w_3, v_2 \rangle}{\|v_2\|^2}.v_2 = [-0.037116, -0.041704, -0.029733, 0.026076, 0.057169]$

$v_3 = w_3 - \frac{\langle w_3, v_1 \rangle}{\|v_1\|^2}.v_1 - \frac{\langle w_3, v_2 \rangle}{\|v_2\|^2}.v_2$

$= [0.002637 , -0.002309 , -0.004485 , 0.008008 , -0.005701]$

The Orthogonal Basis is given by:

$$v_1 = \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix}; \ v_2 = \begin{pmatrix} -0.036528 \\ -0.041043 \\ -0.029262 \\ 0.025663 \\ 0.056263 \end{pmatrix}; \ v_3 = \begin{pmatrix} 0.002637 \\ -0.002309 \\ -0.004485 \\ 0.008008 \\ -0.005701 \end{pmatrix}$$

The Orthogonal Basis values are treated as the Input Training Vector and denoted as follows:

$$X_1 = \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix}; \ X_2 = \begin{pmatrix} -0.036528 \\ -0.041043 \\ -0.029262 \\ 0.025663 \\ 0.056263 \end{pmatrix}; \ X_3 = \begin{pmatrix} 0.002637 \\ -0.002309 \\ -0.004485 \\ 0.008008 \\ -0.005701 \end{pmatrix}$$

To find the d values: $d_i = 1 - \sqrt{\frac{\sum_{i=1}^{n} (S_i(x))^2}{n}}$

$d_1 = 0.645868, \ d_2 = 0.960759, \ d_3 = 0.994920.$

Apply the Learning rule (Delta learning Rule) for the defuzzified individual Column matrices to obtain the weight vector.

Assume the initial weight vector derived using Gaussian distribution method.

$w^1 = (0.2 \ 0.2 \ 0.2 \ 0.2 \ 0.2)^T$. By applying delta learning rule to these weights we get: when $d_1 = 0.645868$, $d_2 = 0.960759$, $d_3 = 0.994920$

$$net^1 = (w^1)^T X_1 = (0.2 \ 0.2 \ 0.2 \ 0.2 \ 0.2) \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix} = 0.350384$$

$$O^1 = \frac{2}{1 + exp(-0.350384)} - 1 = 0.173421$$

$$f'(net^1) = \frac{1}{2}[1 - (O^1)^2] = \frac{1}{2}[1 - (0.173421)^2] = 0.484963$$

$$W^2 = c\,(d_1 - O^1) f'(net^1) X_1 + W^1$$

$$= (0.1)(0.645868 - 0.173421)(0.484963) \begin{pmatrix} 0.32861 \\ 0.29625 \\ 0.30616 \\ 0.42816 \\ 0.39274 \end{pmatrix} + \begin{pmatrix} 0.2 \\ 0.2 \\ 0.2 \\ 0.2 \\ 0.2 \end{pmatrix}$$

$$W^2 = \begin{pmatrix} 0.207529 \\ 0.206788 \\ 0.207015 \\ 0.209810 \\ 0.208998 \end{pmatrix}$$

Following the procedure of Delta Rule we can compute up to:

$$W^4 = c(d_3 - o^3)f^{'}(net^3)X_3 + W^3 = \begin{pmatrix} 0.205901 \\ 0.204696 \\ 0.205383 \\ 0.211444 \\ 0.211423 \end{pmatrix}$$

Computing the mean for the weight vector to fix the threshold, we get the threshold value of the given function as 0.2077694.

Applying activation function:

Mathematically it can be represented as: Binary step function,

$$f(x) = \begin{cases} 1 \; for \; x > \; 0.2077694 \\ 0 \; for \; x \; \leq \; 0.2077694 \end{cases}$$

The decision variable of the function matrix is $(0 \;\; 0 \;\; 0 \;\; 1 \;\; 1)^T$.

Hence, the best alternatives are A_4, A_5.

Many operators namely Ordered Weighted Averaging (OWA) operator, Ordered Weighted Geometric (OWG) operator, Generalised Ordered Weighted Averaging (G-OWA) operator, Generalised Ordered Weighted Geometric (G-OWG), Intuitionistic Fuzzy Weighted Geometric (IFWG) operator and Intuitionistic Fuzzy Weighted Averaging (IFWA) operator which are already existing in the literature can also be utilized in the decision algorithm instead of the Probabilistic Intuitionistic Fuzzy Weighted Geometric (P-IFWG) operator used in the algorithm of this paper. A comparison between all the above said operators is presented in the following table.

Table 1. Comparison of the Proposed Methods with different Operators and some Traditional Ranking methods

Type of Operator	IFWG	IFWA	OWA	OWG	G-OWA	G-OWG
Inputs as Orthogonal Vectors	A_4, A_5	A_1, A_4	A_1, A_2, A_3	A_2, A_3	A_1, A_2, A_3	A_2, A_3, A_5
Inputs as Orthonormal Vectors	A_4, A_5	A_1, A_2, A_3, A_4	A_1, A_2, A_3	A_2, A_3	A_1, A_2, A_3	A_5
Traditional MAGDM Methods	**METHOD – 1:** Score & Accuracy Functions $A_5 > A_2 > A_3 > A_1 > A_4$ Most desirable alternative is A_5.		**METHOD – 2:** Hamming Distance Function Excluding Intuitionistic Degree $A_1 > A_4 > A_3 > A_2 > A_5$ Most desirable alternative is A_1.		**METHOD – 3:** Hamming Distance Function Including Intuitionistic Degree $A_1 > A_4 > A_2 > A_3 > A_5$ Most desirable alternative is A_1.	

3.2 Discussion

In this paper, a novel Intuitionistic Fuzzy Artificial Neural Network model based on Gram-Scmidt Orthogonalization process is proposed. In the process of the Neural Network, the Delta learning rule is utilized where the Intuitionistic Fuzzy nature of the input data is transformed into a fuzzy data and then the ranking the alternatives is done based on the weights updation through the learning phase of the ANN. Numerical illustration is presented for the proposed method which reveals the effective way of producing the input vector using the Gram-Scmidt process. Comparisons of the proposed method is done with computations followed with different aggregation operators and also with traditional methods of ranking the alternatives in the MAGDM problems. The model proposed in [14] is improvised in this paper by the application of Gram-Scmidt process. Table 1 reveals the different choice of the available alternatives when different aggregation operators are employed, which depends on the choice of the decision maker of the problem.

4 Conclusion

This study presents a novel ANN method using some linear space techniques for the generation of effective and efficient input vectors. The proposed method is unusual to the field of ANN as well as MAGDM, since the input vector is not directly taken from the data set unlike the earlier methods, rather processed using Gram-Schmidt orthogonalization process. When it comes to geometry, the real advantage of orthogonalizing the data set is that it neglects the lengths of the segments, whereby making it possible to apply various operations on the drawing without worrying about the exact geometry involved in the system. The aggregation operators which are employed in this work take the decision matrices to the next level of processing through ANN and finally based on the updated weights of the ANN after successful activation, the ranking of the decision alternatives are performed for solving the MAGDM problem. A novel algorithm was ultimately solved using some learning rules (Delta Rule) for ANN that was proposed in this paper using Gram-Scmidt orthogonalization process. The method suggested in this research was compared to the conventional MAGDM method for handling the same choice problem. The new method of ANN compared to the previous method[14] is very effective in a sense that the inputs are produced in such a way to release the complexities faced by the decision maker in the MAGDM situation. More linear space techniques namely the projection method can also be incorporated into ANN in the future research works for producing an effective input vector for the ANN. Since most of the real life and business situations are very vague and ambiguous in nature, the DSS coupling ANN will require much application of different types of IFSs in the future which will be an interesting area of study for business analysists and researchers.

References

1) Atanassov K, Sotirov S, Pencheva T. Intuitionistic Fuzzy Deep Neural Network. *Mathematics*. 2023;11(3):1–14. Available from: https://dx.doi.org/10.3390/math11030716.
2) Ding J, Zhang C, Li D, Zhan J, Li W, Yao Y. Three-way decisions in generalized intuitionistic fuzzy environments: survey and challenges. *Artificial Intelligence Review*. 2024;57(2):1–45. Available from: https://dx.doi.org/10.1007/s10462-023-10647-5.
3) Firouzkouhi N, Amini A, Nazari M, Alkhatib F, Bordbar H, Cheng C, et al. Advanced artificial intelligence system by intuitionistic fuzzy Γ-subring for automotive robotic manufacturing. *Artificial Intelligence Review*. 2023;56(9):9639–9664. Available from: https://dx.doi.org/10.1007/s10462-023-10396-5.
4) Leonishiya A, Robinson J. Varieties of Linguistic Intuitionistic Fuzzy Distance Measures for Linguistic Intuitionistic Fuzzy TOPSIS Method. *Indian Journal Of Science And Technology*. 2023;16(33):2653–2662. Available from: https://dx.doi.org/10.17485/ijst/v16i33.640-icrsms.
5) Leonishiya A, Robinson PJ. A Fully Linguistic Intuitionistic Fuzzy Artificial Neural Network Model for Decision Support Systems. *Indian Journal of Science and Technology*. 2023;16(Special Issue 4):29–36. Available from: https://doi.org/10.17485/IJST/v16iSP4.ICAMS136.
6) Margalit D, Rabinoff J. Interactive Linear Algebra. Georgia Institute of Technology. 2019. Available from: https://textbooks.math.gatech.edu/ila/ila.pdf.
7) Nirmalsingh SS, Robinson PJ. Triangular Intuitionistic Fuzzy Linear Programming Problem with a New Ranking Method based on Correlation Coefficient. *Indian Journal Of Science And Technology*. 2023;16(Special Issue 3):75–83. Available from: https://dx.doi.org/10.17485/ijst/v16isp3.icrtam297.
8) Petkov T, Bureva V, Popov S. Intuitionistic fuzzy evaluation of artificial neural network model. *Notes on Intuitionistic Fuzzy Sets*. 2021;27(4):71–77. Available from: https://dx.doi.org/10.7546/nifs.2021.27.4.71-77.
9) Robinson PJ, Jeeva S. Intuitionistic Trapezoidal Fuzzy MAGDM Problems with Sumudu Transform in Numerical Methods. *International Journal of Fuzzy System Applications*. 2019;8(3):1–46. Available from: https://dx.doi.org/10.4018/ijfsa.2019070101.
10) Robinson PJ, Jeeva S. Application of Integrodifferential Equations Using Sumudu Transform in Intuitionistic Trapezoidal Fuzzy MAGDM Problems. In: Applied Mathematics and Scientific Computing. Trends in Mathematics;Springer International Publishing. 2019;p. 13–20. Available from: https://doi.org/10.1007/978-3-030-01123-9_2.
11) Robinson PJ, Indhumathi M, Manjumari M. Numerical Solution to Singularly Perturbed Differential Equation of Reaction-Diffusion Type in MAGDM Problems. In: Applied Mathematics and Scientific Computing . Trends in Mathematics;Springer International Publishing. 2019;p. 3–12. Available from: https://doi.org/10.1007/978-3-030-01123-9_1.
12) Kumar A, Sharma TK, Verma OP. Detection of Heart Failure by Using Machine Learning. In: Machine Intelligence for Research and Innovations ;vol. 831 of Lecture Notes in Networks and Systems. Singapore. Springer. 2024;p. 195–203. Available from: https://doi.org/10.1007/978-981-99-8135-9_17.
13) Sharma R, Verma OP, Kumari P. Application of Dragonfly Algorithm-Based Interval Type-2 Fuzzy Logic Closed-Loop Control System to Regulate the Mean Arterial Blood Pressure. In: Machine Intelligence for Research and Innovations ;vol. 831 of Lecture Notes in Networks and Systems. Machine Intelligence for Research and Innovations. Springer. 2024;p. 183–193. Available from: https://doi.org/10.1007/978-981-99-8135-9_16.
14) Robinson PJ, Leonishiya A. Application of Varieties of Learning Rules in Intuitionistic Fuzzy Artificial Neural Network. In: Machine Intelligence for Research and Innovations;vol. 832 of Lecture Notes in Networks and Systems. Singapore. Springer. 2024;p. 35–45. Available from: https://doi.org/10.1007/978-981-99-8129-8_4.