Optimality

www.opt.reapress.com

Opt. Vol. 2, No. 4 (2025) 266–279.

Paper Type: Original Article

Ann-Magdm Problem Solving with Spectral Theory-based Weight Determination Methods

Wilson Arul Prakash S[1,*] **John Robinson P**[1]

[1] Department of Mathematics, Bishop Heber College, Affiliated to Bharathidasan University, Tiruchirappalli, 620017, Tamilnadu, India; wilson.loyola2@gmail.com.

Citation:

Received: 13 January 2025 *Revised: 16 March 2025* *Accepted: 24 May 2025*	Wilson Arul Prakash, S., & John Robinson, P. (2025). Ann-magdm problem solving with spectral theory-based weight determination methods. *Optimality*, 2(4), 266-279.

Abstract

This paper explores the extension of matrix eigenvalue theory to the spectral theory of operators on Banach and Hilbert spaces, focusing on finite-dimensional Hilbert spaces as a foundational step. Since matrices uniquely represent linear operators in finite-dimensional spaces, the relationship between linear operators and matrices is used to bridge the understanding of spectral theory. The study begins with a detailed examination of the spectrum of an operator, highlighting its properties and implications. The core contribution of this work lies in addressing a challenging problem: deriving the decision-maker's weight vector from concepts rooted in spectral theory. This approach is applied to solve Multiple Attribute Group Decision Making (MAGDM) problems, demonstrating its theoretical robustness and practical relevance. Furthermore, the same MAGDM problem is tackled using well-known methods from Artificial Neural Networks (ANNs). A comparative analysis is conducted to evaluate the practical aspects, productivity, and advantages of the proposed methods compared with existing solutions. This comprehensive investigation aims to provide deeper insights into the interplay among spectral theory, decision-making, and ANN techniques, advancing both mathematical theory and practical applications.

Keywords: Eigen values, Spectral theory, MAGDM, Artificial neural network, Operator theory.

1 | Introduction

The central aim of spectral theory is to "classify" all linear operators, a task that naturally directs its focus toward Hilbert spaces. This restriction arises because the general framework for Banach spaces, while inclusive, remains poorly understood in many respects even today. The clarity and depth achievable in Hilbert spaces are not only theoretically appealing but also practically significant. Most pivotal applications of spectral theory tend to reside within this more structured and accessible context. At first glance, this alignment with Hilbert spaces might appear to be a fortunate coincidence. However, more profound reflection reveals a deeper reason: Hilbert spaces are intricately connected to the Euclidean plane and spatial geometry. Unlike Banach spaces, which may diverge significantly from geometric intuitions, Hilbert spaces retain a natural

Corresponding Author: wilson.loyola2@gmail.com

https://doi.org/10.22105/opt.v2i4.95

affinity for the geometric principles that underpin much of our understanding of the physical world. Since Euclidean geometry serves as an accurate description of the universe across many scales, it is hardly surprising that infinite-dimensional mathematical constructs, when applied to real-world problems, align closely with the geometric intuition embodied by Hilbert spaces.

Functional analysis equips us with tools to rigorously analyze and classify linear operators, while also providing a pathway to generalize finite-dimensional results to the infinite-dimensional setting. In some instances, a firm grasp of integration theory on general measure spaces becomes indispensable. This requirement stems from the need to work with spaces and operators that arise in various applied and theoretical contexts, particularly those involving measures and function spaces. Integration theory provides the mathematical foundation for dealing with such operators, ensuring the necessary rigor and precision. This work begins by revisiting and condensing the essential knowledge required for further exploration. We start with a review of the general spectrum of Banach algebras, a fundamental concept that bridges the study of algebraic and topological properties of operators. The spectrum provides critical insights into operator behavior, enabling us to classify and analyze them within a broader framework. Next, we delve into the fundamentals of compact operators on Hilbert spaces. Compact operators play a vital role in spectral theory, offering a rich structure and numerous applications. Their properties, such as the discreteness of the spectrum and the accumulation points of eigenvalues, make them particularly interesting for both theoretical and practical purposes. Before proceeding with the detailed discussion, some preliminary notation and conventions are introduced to ensure clarity and consistency. These foundational elements provide the basis for a systematic, structured approach to subsequent analysis.

The data set used in this work was introduced and applied in ANN by [1] and [2]. The Intuitionistic fuzzy domain was used in ANN-based robotics by [3], and time series forecasting using an IFS-based ANN was performed in [4]. An intuitionistic fuzzy neural network with a Gaussian membership function was proposed by [5], and a fully linguistic intuitionistic fuzzy ANN was proposed by [6], and an IFS evaluation of ANN was put forth in [7]. In [8], various learning rules for IFS ANNs were applied, and many comparisons with classical methods were made. The authors in [9] proposed a Gram-Schmidt Orthogonalized ANN in which the input vector to the ANN was an IFS matrix, the entries were orthogonalized, and the results were compared with those of existing methods. In this paper, the numerical illustration in [9] will be used for computations, and our proposed techniques will be compared with those in [9]. Medical applications of machine learning methods were proposed in [10], and a multi-criteria-based method was proposed in [11]. In [12–14], deep neural networks and complex neural systems were discussed, which are recent areas of interest for researchers. In this paper, the balancing (weight) vector provided by the decision makers will be used to solve MAGDM problems, and the same numerical illustration will be solved using some ANN techniques. Understanding the differences among the proposed techniques in this work will enable decision-makers to select the best ANN approach for the given situation, taking into account factors such as accuracy, computational efficiency, and interpretability of results. By adding this viewpoint, the comparative analysis is enhanced, and the methodological variation in using ANNs to solve decision-making problems is highlighted.

2 | Spectrum of an Operator on a Finite-Dimensional Hilbert Space

We will define the analogues of operators and their eigenvectors, since we apply matrix eigenvalue theory to operators on a Hilbert space.

Definition 1. Let T be an operator on space H, which is Hilbert. If there is a non-zero vector x in H such that $Tx=\lambda x$, then a scalar λ is an eigenvalue (also known as a characteristic value or spectral value) of T.

Definition 2. Any non-zero vector x in H that $Tx=\lambda x$ is referred to as an eigen vector (also known as characteristic vector or spectral vector) of T if λ is an eigenvalue of T.

Note: The entire theory becomes trivial if there are no non-zero vectors in the Hilbert space, which means that T cannot have any eigen vectors. Thus, we will assume $H\neq\{0\}$ throughout the task. The following property is derived from the definition of eigenvalues and eigenvectors. Let λ be an operator T's eigenvalue on Hilbert space H. M_λ is a non-zero closed linear subspace of H that is invariant under T if it is the set of all eigenvectors of T that correspond to the eigenvalue λ and the zero vector 0. Based on the known properties of $M_{\lambda,}$: $M_\lambda = \{x \in H : Tx = \lambda x\} = \{x \in H : (T - \lambda I)x = 0\}$. Also, M_λ is the null space of the continuous transformation $T - \lambda I$, because T and I are continuous. Hereby M_λ should be closed. Next, i $x \in M_\lambda$, f then $Tx \in M_\lambda$ and hence we should have $Tx = \lambda x$. As we know that M_λ is a linear subspace of H, $x \in M_\lambda$ which leads to the result $\lambda x = T_x \in M_\lambda$. Altogether, it is now interesting to see that M_λ is invariant under T.

Definition 3. The eigen space of T that corresponds to the eigenvalue λ is the closed subspace M_λ. Every eigenspace of T is a non-zero closed Linear subspace of H that is invariant under T.

Note: In general, an operator on a Hilbert space H need not necessarily have an eigenvalue, as will be discussed in the following illustration.

Example 1. Consider the Hilbert space on l_2 defined by $T(x_1, x_2, \ldots, x_n) = \{0, x_1, x_2, \ldots\}$. Suppose λ being an eigenvalue of T, then there exists a non-zero vector $(x_1, x_2, \ldots, x_n)$ such that $Tx = \lambda x$, which provides $(0, x_1, x_2, \ldots) = (\lambda x_1, \lambda x_2, \ldots, \lambda x_n)$ which in turn implies that $x_1 = 0, \lambda x_2 = x_1, \lambda x_n = x_{n-1}$ ... By hypothesis, $x = (x_n) \in l_2$ cannot be a zero vector, so that $x_n \neq 0$ for any n. Hence $\lambda x_1 = 0$ implies λ=0 and $\lambda x_2 = x_1$ implies $x_1 = 0$ contradicting x is a non-zero vector. Hence, it is clear that T must not have any eigenvalues.

Definition 4. The spectrum of T, represented by σ(T), is the collection of all of T's eigenvalues.

The non-emptiness of the spectrum of an operator on a finite-dimensional Hilbert space H is established by the following theorem.

Theorem 1. The spectrum of T, or σ(T), is a finite subset of the complex plane if T is an arbitrary operator on a finite-dimensional Hilbert space H. Additionally, the number of points in σ(T) does not exceed the dimension n of H.

Lemma 1. If and only if there is a non-zero vector x in a finite-dimensional Hilbert space H such that Tx=0, then the operator T on H is singular.

Note: A complex scalar linked with H has at least one point in σ(T). It can have up to n points, but no more than n. If a scalar field is real, then σ(T) might be empty. To obtain a richer theory, we typically use the complex scalar in spectral theory.

The following theorem gives some elementary properties of $\sigma(T)$.

Theorem 2. Suppose T is an operator function on a finite-dimensional Hilbert space, and the preceding statements hold:

I. T is singular iff $0 \in \sigma(T)$.

II. Suppose T is non-singular, then $\lambda \in \sigma(T)$ iff $\lambda^{-1} \in \sigma(T^{-1})$.

III. Suppose A is non-singular, then $\sigma(ATA^{-1}) = \sigma(T)$.

IV. Suppose $\lambda \in \sigma(T)$, and if P is a polynomial, then it can be proved that $P(\lambda) \in \sigma(P(T))$.

Next, the spectrum of operators can be expressed as functions of T.

Theorem 3. (Spectral mapping theorem (polynomials)) Suppose T is an operator on a complex Banach space B, and if p is a polynomial, then it can be proved that $\sigma(p(T)) = p(\sigma(T) = \{p(\lambda): \lambda \in \sigma(T)\}$.

Example 2. Consider the Spectrum of the idempotent operator function T on a Banach space.

By the property of idempotency, $T^2 = T$ or $T^2 - T = 0$. Let $p(T) = T^2 - T$.

Then $p(T) = 0$ by hypothesis. Hence $p(\sigma^2(T) = \sigma^2 T - \sigma(T) = \sigma(T)(\sigma(T) - 1) = 0$ so that $\sigma(T) = 1$ or $\sigma(T) = 0$. Hence $\sigma(T) = \{0,1\}$.

3 | Decision Maker Weight Determining Methods with Real Eigen Values

MAGDM problem-solving involves the effective involvement of the decision maker, who is responsible for providing consensus for the process. At one end, the decision maker's weight vector plays a crucial role, and in situations where the problem-solving techniques depend solely on it, a proper method to elicit it from the decision maker becomes crucial at the other end. In this paper, and in particular when dealing with a decision maker who provides weights in the form of linear space problems, we need means and methods to process this information and retrieve the weights in a proper sense that appropriately suits the decision problem involved. Here is a model of eliciting a weight vector from the decision maker.

Step 1. Let H be a Hilbert space and T_i be a linear operator function on H and $e_{i1}, e_{i2}, \ldots, e_{in}$ is the basis for H. Let $T_i(e_{i1}), T_i(e_{i2}), \ldots, T_i(e_{in})$ be values in H. Let $e_{11}, e_{12}, \ldots, e_{1n}$ basis for H, where $A_1 = (T_1(e_{11}) \quad T_1(e_{12}) \ldots T_1(e_{1n}))^T$. Let $e_{21}, e_{22}, \ldots, e_{2n}$ be another basis for H, where $A_2 = (T_1(e_{21}) \quad T_1(e_{22}) \ldots T_1(e_{2n}))^T$. Similarly, we can find A_n, where $A_n = (T_1(e_{n1}) \quad T_1(e_{n2}) \ldots T_1(e_{nn}))^T$.

Step 2. Let $\lambda_i = (\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{in})$ be the real eigenvalues for a matrix A_i, where the eigenvalues are in the spectrum of the space, and form the matrix $B_j = (\lambda_1 \quad \lambda_2 \quad \ldots \quad \lambda_n)$.

Step 3. Find $B_j * {B_j}^T$ and then normalize the matrix to elicit the weights $w_1, w_2, \ldots, w_n$ such that $\sum_{i=1}^{n} w_i = 1$.

Problem proposed by decision maker-1

Let us suppose that decision maker-1 provides the balancing (weight) information in the form of the following linear space problem, which needs to be unlocked by the stakeholder.

Step 1. Let T_1 be a linear operator function on Hilbert space $\mathfrak{R}^3$. Let $\beta_1 = \{e_1 = (1,0,0), e_2 = (0,1,0), e_3 = (0,0,1)\}$ be a basis set. Let T_1 be defined by, $T_1(e_1) = (x_1, x_1 - y_1, z_1) = (1,1,0)$; $T_1(e_2) = (x_2, 2y_2, z_2) = (0,2,0)$; $T_1(e_3) = (x_3, -z_3, 4z_3) = (0,-1,4)$. The corresponding matrix A_1 is, $A_1 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 4 \end{pmatrix}$, and eigenvalues are $\lambda_1 = \{1,2,4\}$. Let T_2 be defined by, $T_2(e_1) = (x_1, x_1 + y_1, z_1) = (1,1,0)$; $T_2(e_2) = (x_2, y_2, z_2) = (0,2,0)$; $T_2(e_3) = (x_3, -z_3, 2z_3) = (0,-1,2)$. The corresponding matrix A_2 is, $A_2 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 2 \end{pmatrix}$, and eigenvalues are $\lambda_2 = \{1,2,2\}$. Let T_3 be defined by, $T_3(e_1) = (z_1, y_1, 2x_1) = (0,0,2)$; $T_3(e_2) = (-3y_2 + x_2, y_2, 6y_2 + z_2) = (-3,1,6)$; $T_3(e_3) = (x_3, y_3, z_3) = (0,0,1)$.

The corresponding matrix A_3 is, $A_3 = \begin{pmatrix} 0 & 0 & 2 \\ -3 & 1 & 6 \\ 0 & 0 & 1 \end{pmatrix}$, and eigenvalues are $\lambda_3 = \{0,1,1\}$.

Step 2. The matrix B_1 is, $B_1 = (\lambda_1 \quad \lambda_2 \quad \lambda_3)$, where $B_1 = \begin{pmatrix} 1 & 1 & 0 \\ 2 & 2 & 1 \\ 4 & 2 & 1 \end{pmatrix}$.

Step 3. Finding $B_1 * {B_1}^T$, we have the following computation: $B_1 * {B_1}^T = \begin{pmatrix} 2 & 4 & 6 \\ 4 & 9 & 13 \\ 6 & 13 & 21 \end{pmatrix}$.

Normalizing $B_j * {B_j}^T$, we get the weight vector of Decision Maker-1 as:

$$\gamma = (0.156837, 0.334829, 0.508333).$$

Problem proposed by Decision maker-2: Suppose the decision maker-2 proposes balancing (weight) information that the stakeholder must unlock.

Step 1. Let T_4 be a linear operator function on Hilbert space $\mathfrak{R}^3$ and let $\beta_2 = \{b_1 = (4,2,0), b_2 = (2,1,5), b_3 = (1,0,0)\}$ be a basis set. The operator T_4 is defined by, $T_4(b_1) = \left(z_1, \frac{y_1}{2}, \frac{x_1}{2}\right) = (0,1,2)$; $T_4(b_2) =$

$\left(-2x_2, y_2, \frac{x_2+y_2+z_2}{2}\right) = (-4,1,4)$; $T_4(b_3) = (-5x_3, x_3, 7x_3 + z_3) = (-5,1,7)$. The corresponding eigenvalues are $\lambda_4 = \{1,2,5\}$. Let the operator T_5 is defined by, $T_5(b_1) = \left(x_1, \frac{y_1}{2}, x_1 + y_1\right) = (4,1,6)$; $T_5(b_2) = (z_2 - 2x_2 - y_2, x_2, z_2 - x_2) = (0,2,3)$; $T_5(b_3) = (z_3, y_3, 9x_3) = (0,0,9)$. The corresponding eigenvalues are $\lambda_5 = \{2,4,9\}$. Let the operator T_6 be defined by, $T_6(b_1) = \left(z_1, \frac{y_1}{2}, \frac{x_1}{4}\right) = (0,1,1)$; $T_6(b_2) = (-y_2, x_2, z_2 - 2x_2) = (-1,2,1)$;$T_6(b_3) = (-x_3, -x_3, 4x_3) = (-1,-1,4)$. The corresponding eigenvalues are $\lambda_6 = \{1,2,3\}$.

Step 2. The matrix B_2 is formed as: $B_2 = (\lambda_4 \quad \lambda_5 \quad \lambda_6)$, where $B_2 = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 5 & 9 & 3 \end{pmatrix}$.

Step 3. Finding $B_2 * B_2{}^T$, we have the following computation: $B_2 * B_2^T = \begin{pmatrix} 6 & 12 & 26 \\ 12 & 24 & 152 \\ 26 & 52 & 115 \end{pmatrix}$.

Normalizing $B_j * B_j{}^T$, we get the weight vector of decision maker-2 as:

$w = (0.135814, 0.271628, 0.592557)$.

Problem proposed by decision maker-3

Let us suppose that decision maker-3 proposes balancing (weight) information, which the stakeholder must unlock.

Step 1. Let T_7 be a linear operator function on Hilbert space $\mathfrak{R}^4$ and $\beta_3 = \{c_1 = (1,0,0,0), c_2 = (0,1,0,0), c_3 = (0,0,1,0), c_4 = (0,0,0,1)\}$ be the basis. The operator T_7 defined by, $T_7(c_1) = (2x_1, x_1, x_1, r_1) = (2,1,1,0)$; $T_7(c_2) = (y_2, y_2, z_2, y_2 + x_2) = (1,2,0,1)$; $T_7(c_3) = (z_3, x_3, 2z_3, z_3) = (1,0,2,1)$; $T_7(c_4) = (z_3, r_4, r_4, 2r_4) = (0,1,1,2)$. The corresponding eigenvalues are $\lambda_7 = \{4,2,0,2\}$. Let the operator T_8 be defined by,$T_8(c_1) = (x_1, 4x_1, y_1, r_1) = (1,4,0,0)$; $T_8(c_2) = (x_2, y_2, z_2, r_2) = (0,2,0,0)$; $T_8(c_3) = (5z_3, 3z_3, z_3 + x_3, z_3) = (5,3,1,1)$; $T_8(c_4) = (4r_4, 7r_4, 2r_4, 2r_4) = (4,7,2,2)$. The corresponding eigenvalues are $\lambda_8 = \{0,1,2,3\}$. Let the operator T_9 be defined by, $T_9(c_1) = (2x_1, x_1, 3x_1, 4x_1 + r_1) = (2,1,3,4)$; $T_9(c_2) = (x_2, 2y_2, y_2, 3y_2) = (0,2,1,3)$; $T_9(c_3) = (2z_3, z_3, 6z_3 + x_3, 5z_3) = (2,1,6,5)$; $T_9(c_4) = (r_4, 2r_4, 4r_4, 8r_4) = (1,2,4,8)$. The corresponding eigenvalues are $\lambda_9 = \{13.0990, 1, 2.9010, 1\}$. Let the operator T_{10} be defined by, $T_{10}(c_1) = (4x_1, -x_1, -x_1, -x_1 + r_1) = (4,-1,-1,-1)$; $T_{10}(c_2) = (-y_2, 4y_2, -y_2, -y_2) = (-1,4,-1,-1)$; $T_{10}(c_3) = (-z_3, -z_3, 4z_3, -z_3) = (-1,-1,4,-1)$; $T_{10}(c_4) = (-r_4, -r_4, -r_4, 4r_4) = (-1,-1,-1,4)$. The corresponding eigenvalues are $\lambda_{10} = \{1,5,5,5\}$.

Step 2. The matrix B_3 is formed as: $B_3 = (\lambda_7 \quad \lambda_8 \quad \lambda_9 \quad \lambda_{10})$, and hence

$$B_3 = \begin{pmatrix} 4 & 0 & 13.0990 & 1 \\ 2 & 1 & 1 & 5 \\ 0 & 2 & 2.9010 & 5 \\ 2 & 3 & 1 & 5 \end{pmatrix}.$$

Step 3. Finding $B_3 * B_3{}^T$ and normalizing $B_3 * B_3{}^T$, we get the weight vector of Decision Maker-3 as: $\omega = (0.343122, 0.191853, 0.226570, 0.238453)$.

Now, as we have elicited the weighting vectors from the 1st, 2nd and 3rd decision makers, $\gamma = (0.156837, 0.334829, 0.508333)$, $w = (0.135814, 0.271628, 0.592557)$

and $\omega = (0.343122, 0.191853, 0.226570, 0.238453)$ respectively, we can make use of these vectors in the aggregation computation of the decision matrices of the given MAGDM problem to select the best alternatives.

4 | Decision Maker Weight Determining Methods with Complex Eigen Values

In the previous case, we used real eigenvalues to determine decision-maker weights; in this section, we consider complex eigenvalues for determining the decision-maker weight vector.

Step 1. Let H be a Hilbert space and T_i be a linear operator function on H and $e_{i1}, e_{i2}, \dots, e_{in}$ is the basis for H. Let $T_i(e_{i1}), T_i(e_{i2}), \dots, T_i(e_{in})$ be values in H. Let $e_{11}, e_{12}, \dots, e_{1n}$ basis for H, where $A_1 = (T_1(e_{11}) \quad T_1(e_{12}) \dots T_1(e_{1n}))^T$. Let $e_{21}, e_{22}, \dots, e_{2n}$ be another basis for H, where $A_2 = (T_1(e_{21}) \quad T_1(e_{22}) \dots T_1(e_{2n}))^T$. Similarly, we can find A_n, where $A_n = (T_1(e_{n1}) \quad T_1(e_{n2}) \dots T_1(e_{nn}))^T$.

Step 2. Let $\lambda_i = (\lambda_{i1}, \lambda_{i2}, \dots, \lambda_{in})$ be the complex eigenvalues for a matrix A_i, where the eigenvalues are in the spectrum of the space, and form the matrix $\lambda_{in} = a + ib$, $|\lambda_{in}| = \sqrt{a^2 + b^2}$, $B_j = (|\lambda_1| \quad |\lambda_2| \quad \dots \quad |\lambda_n|)$.

Step 3. Find $B_j * {B_j}^T$ and then normalize the matrix to elicit the weights $w_1, w_2, \dots, w_n$ such that $\sum_{i=1}^{n} w_i = 1$.

Problem proposed by decision maker-1

Step1. Let T_1 be a linear operator function on Hilbert space $\Re^3$, and $\beta_1 = \{e_1 = (1,0,0), e_2 = (1,1,0), e_3 = (1,1,1)\}$ be a basis set. Let T_1 be defined by, $T_1(e_1) = (7x_1, 3x_1 + y_1, 8x_1 + z_1) = (7,3,8)$; $T_1(e_2) = (11x_2, 9y_2, 2x_2 + 3y_2) = (11,9,5)$; $T_1(e_3) = (6x_3, 8y_3, 4z_3) = (6,8,4)$. The corresponding eigenvalues are $\lambda_1 = \{20.1981, -0.0990 + i3.1609, -0.0990 - i3.1609\}$. Let T_2 be defined by, $T_2(e_1) = (4x_1, -2x_1, 3x_1 + z_1) = (4, -2,3)$; $T_2(e_2) = (8x_2, -3y_2, 2x_2 + 3y_2) = (8, -3,5)$; $T_2(e_3) = (7x_3, -2z_3, 4z_3) = (7, -2,4)$.

The corresponding eigenvalues are $\lambda_2 = \{-0.2854 + i0.3132, -0.2854 + i0.3132, 5.5708\}$. Let T_3 be defined by, $T_3(e_1) = (x_1, 2x_1, 2x_1) = (1,2,2)$; $T_3(e_2) = (2x_2, 3y_2, 3y_2 + 2x_2) = (2,3,5)$; $T_3(e_3) = (3x_3, 2y_3, 3z_3) = (3,2,3)$.

The corresponding eigenvalues are $\lambda_3 = \{-0.3803 + i0.8703, 7.7605, -0.3803 - i0.8703\}$.

Step2. The matrix B_1 is formed as follows:

$$\lambda_1 = \{20.1981, -0.0990 + i3.1609, -0.0990 - i3.1609\},$$

$$\lambda_{11} = 20.1981 \Rightarrow |\lambda_{11}| = \sqrt{a^2 + b^2} = \sqrt{(20.1981)^2 + 0^2} = 20.1981,$$

$$\lambda_{12} = -0.0990 + i3.1609 \Rightarrow |\lambda_{12}| = \sqrt{(-0.0990)^2 + (3.1609)^2} = 3.1624,$$

$$\lambda_{13} = -0.0990 - i3.1609 \Rightarrow |\lambda_{13}| = \sqrt{(-0.0990)^2 + (-3.1609)^2} = 3.1624,$$

$$\lambda_2 = \{-0.2854 + i0.3132, -0.2854 + i0.3132, 5.5708\},$$

$$\lambda_{21} = -0.2854 + i0.3132 \Rightarrow |\lambda_{21}| = \sqrt{(-0.2854)^2 + (0.3132)^2} = 0.4237,$$

$$\lambda_{22} = -0.2854 - i0.3132 \Rightarrow |\lambda_{22}| = \sqrt{(-0.2854)^2 + (-0.3132)^2} = 0.4237,$$

$$\lambda_{23} = 5.5708 \Rightarrow |\lambda_{23}| = \sqrt{a^2 + b^2} = \sqrt{(5.5708)^2 + 0^2} = 5.5708,$$

$$\lambda_3 = \{-0.3803 + i0.8703, 7.7605, -0.3803 - i0.8703\},$$

$$\lambda_{31} = -0.3803 + i0.8703 \Rightarrow |\lambda_{21}| = \sqrt{a^2 + b^2} = \sqrt{(-0.3803)^2 + (0.8703)^2} = 0.9497,$$

$$\lambda_{32} = 7.7605 \Rightarrow |\lambda_{32}| = \sqrt{(7.7605)^2 + 0^2} = 7.7605,$$

$$\lambda_{33} = -0.3803 + i0.8703 \Rightarrow |\lambda_{21}| = \sqrt{a^2 + b^2} = \sqrt{(-0.3803)^2 + (0.8703)^2} = 0.9497.$$

Now the matrix B_1 is: $B_1 = (|\lambda_1| \quad |\lambda_2| \quad |\lambda_3|)$, where $B_1 = \begin{pmatrix} 20.1981 & 0.4237 & 0.9497 \\ 3.1624 & 0.4237 & 7.7605 \\ 3.1624 & 5.5708 & 0.9497 \end{pmatrix}$.

Step3. Finding $B_1 * {B_1}^T$, we have the following computation:

$$B_1 * B_1^T = \begin{pmatrix} 409.0447 & 71.4241 & 67.1367 \\ 71.4241 & 70.4057 & 19.7313 \\ 67.1367 & 19.7313 & 41.9365 \end{pmatrix}.$$

Normalizing $B_j * {B_j}^T$, we get the matrix as: $(B_1 * B_1^T)^N = \begin{pmatrix} 0.7470 & 0.4421 & 0.5212 \\ 0.1304 & 0.4358 & 0.1532 \\ 0.1226 & 0.1221 & 0.3256 \end{pmatrix}$, where N represents the normalized matrix. Further, taking the average of the entries, we get the weights as $w = \{0.5701, 0.2398, 0.1901\}$.

Problem proposed by decision maker-2

Let T_4 be a linear operator function on Hilbert space $\mathfrak{R}^3$ and $\beta_2 = \{b_1 = (4,2,0), b_2 = (6,3,5), b_3 = (7,3,5)\}$ is a basis set. The operator T_4 be defined by, $T_4(b_1) = \left(\frac{x_1}{4}, \frac{y_1}{2}, \frac{x_1+y_1}{6}\right) = (1,3,1)$; $T_4(b_2) = \left(-\frac{x_2}{6}, y_2 - 1, \frac{z_3}{5}\right) = (-1,2,-1)$; $T_4(b_3) = (x_3 - 5, 2y_3, 7x_3 + 5y_3 - 3z_3) = (2,6,0)$. The collection of eigenvalues is $\lambda_4 = \{-0.8232, 1.9116 + i2.9143, 1.9116 - i2.9143\}$. The operator T_5 defined by, $T_5(b_1) = (y_1, x_1, x_1 + z_1) = (2,4,4)$; $T_5(b_2) = \left(\frac{z_2}{5}, \frac{x_2}{3}, y_2\right) = (1,2,3)$; $T_5(b_3) = \left(-y_3, \frac{y_3+z_3}{-2}, -z_3\right) = (-3,-4,-5)$. The collection of eigenvalues is $\lambda_5 = \{+2i, -2i, -1\}$. The operator T_6defined by, $T_6(b_1) = (3x_1 + 1, -3x_1, 4y_1) = (13,-12,7)$; $T_6(b_2) = (y_2 + 1, -z_2, x_2) = (4,-5,6)$; $T_6(b_3) = (-y_3, -z_3 - 1, -x_3 - 1) = (-3,-6,-8)$. The collection of eigenvalues is $\lambda_6 = \{-3.5883 + i7.8240, -0.8233, -3.5883 - i7.8240\}$. Following similar computations as for decision maker-1, we get the weights as $\gamma = (0.3828, 0.1727, 0.4445)$.

Problem proposed by decision maker-3

Let T_7 be a linear operator on Hilbert space $\mathfrak{R}^4$ and $\beta_3 = \{c_1 = (1,0,0,0), c_2 = (1,1,0,0), c_3 = (1,1,1,0), c_4 = (1,1,1,1)\}$ be a basis set. The operator T_7 defined by, $T_7(c_1) = (x_1, 2x_1, -x_1, r_1) = (1,2,-1,0)$; $T_7(c_2) = (z_2, 5x_2, 3y_2, r_2) = (0,5,3,0)$; $T_7(c_3) = (-2x_3, r_3, x_3 - y_3, 4z_3) = (-2,0,0,4)$; $T_7(c_4) = (x_4 - z_4, 6y_4, -4z_4, -3r_4) = (0,6,-4,-3)$. The operator T_8 defined by, $T_8(c_1) = (9x_1, 13x_1, 5x_1, 2x_1) = (9,13,5,2)$; $T_8(c_2) = (x_2, 11y_2, 4x_2 + 3y_2, 3x_2 + 3y_2) = (1,11,7,6)$; $(c_3) = (3x_3, 7y_3, 4z_3, y_3) = (3,7,4,1)$; $T_8(c_4) = (6x_4, y_4 - z_4, 7z_4, 10r_4) = (6,0,7,10)$. The operator T_9 defined by, $T_9(c_1) = (x_1, 4x_1 + y_1, 2x_1 + z_1, 3x_1 + r_1) = (1,4,2,3)$; $T_9(c_2) = (x_2y_2, y_2, 4x_2, 4y_2) = (0,1,4,4)$; $T_9(c_3) = (-x_3, y_3 - z_3, z_3, r_3) = (-1,0,4,1)$; $T_9(c_4) = (2x_4, z_4 - r_4, 4z_4, r_4) = (2,0,4,1)$. Following similar computations, we get the weights as:

$\omega = (0.2853, 0.2133, 0.38225, 0.119175)$.

Similar to the computations with real eigen values, as we have elicited the weighting vectors from the total decision makers involved in the problem, from the first $w = (0.5701, 0.2398, 0.1901)$, from the second $\gamma = (0.3828, 0.1727, 0.4445)$ and from the third decision maker $\omega = (0.2853, 0.2133, 0.38225, 0.119175)$, we can make use of these vectors in the aggregation computation of the decision matrices of the given MAGDM problem to select the best alternatives.

5 | Algorithm for the Classical Decision-Making Method

Step 1. Compress or reduce the number of columns of all matrices, keeping the number of matrices unchanged, using the IFWAA aggregation operator and the balancing vector (weight) provided by the decision maker.

Step 2. Compress or reduce the number of matrices into a single matrix with a single column using the IFHA aggregation operator and the balancing vector (weight) provided by the decision maker.

Step 3. In this step, the correlations are computed between the final m_i's, and the ideal IFS value, where $\widetilde{m}_i = (0,1)$. The correlation of $G, H \in IFSs(X)$ is given by a formula $Cr_{ZL}(G,H) = \frac{1}{n}\sum_{i=1}^{n}[u_G(x_i)u_H(x_i) + \gamma_G(x_i)\gamma_H(x_i) + \pi_G(x_i)\pi_H(x_i)]$.

Step 4. Here, the correlation coefficient is displayed for the selection of the best option: $\rho r_{ZL}(m_1, \widetilde{m}_i)$, where $\rho r_{ZL}(G,H) = \frac{Cr_{ZL}(G,H)}{\sqrt{Cr_{ZL}(G,G)Cr_{ZL}(H,H)}}$.

Step 5. Arranging the options based on the computed $\rho r_{ZL}(G,H)$, we can obtain the most preferred choice based on the highest relationship between the variables.

6 | Numerical Illustration: MAGDM with Weights Derived From Linear Space Methods with Real and Complex Eigen Values

Consider the numerical illustration in [9] of the Risk Investment company, with five investment options based on criteria that must be evaluated against the alternatives. The balancing vectors (weights) are $w = (0.156837, 0.334829, 0.508333), \gamma = (0.135814, 0.271628, 0.592557)$ and $\omega = (0.343122, 0.191853, 0.226570, 0.238453)$, which are evaluated from the process of the information given by the decision makers and derived from the real eigenvalues. The decision matrices provided by the decision makers are:

$$R_1 = \begin{pmatrix} (0.5,0.4) & (0.6,0.3) & (0.3,0.6) & (0.2,0.7) \\ (0.7,0.3) & (0.7,0.2) & (0.7,0.2) & (0.4,0.5) \\ (0.6,0.4) & (0.5,0.4) & (0.5,0.3) & (0.2,0.3) \\ (0.8,0.1) & (0.6,0.3) & (0.3,0.4) & (0.2,0.6) \\ (0.6,0.2) & (0.4,0.3) & (0.7,0.1) & (0.1,0.3) \end{pmatrix},$$

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

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

Step 1. The IFWAA operator compresses the columns and reduces each matrix to a single column.

$$IFWAA(\widetilde{a_1}, \widetilde{a_2}, \dots, \widetilde{a_n}) = \sum_{j=1}^{n} \widetilde{a_j}\omega_j = \left(1 - \prod_{j=1}^{n}(1-\mu_{a_j})^{\omega_j},\ \prod_{j=1}^{n}(\gamma_{a_j})^{\omega_j}\right).$$

$$\tilde{r}_1^{(1)} = \left(1 - \left[(1-\mu_{a_1})^{\omega_1}(1-\mu_{a_2})^{\omega_2}(1-\mu_{a_3})^{\omega_3}(1-\mu_{a_4})^{\omega_4}\right], (\gamma_{a_1})^{\omega_1}(\gamma_{a_2})^{\omega_2}(\gamma_{a_3})^{\omega_3}(\gamma_{a_4})^{\omega_4}\right).$$

$$r_1^{(1)} = \left(\left(1 - \begin{bmatrix}(1-0.5)^{0.343122} * (1-0.6)^{0.191853} * \\ (1-0.3)^{0.226570} * (1-0.2)^{0.238453}\end{bmatrix}\right), \begin{pmatrix}(0.4)^{0.343122} * (0.3)^{0.191853} * \\ (0.6)^{0.226570} * (0.7)^{0.238453}\end{pmatrix}\right).$$

$r_1^{(1)} = (0.4217, 0.4742)$. Similarly, all the other values can be computed.

$r_2^{(1)} = (0.6461, 0.2860)$; $r_3^{(1)} = (0.4819, 0.3499)$; $r_4^{(1)} = (0.5777, 0.2591)$,

$r_5^{(1)} = (0.5085, 0.2035)$; $r_1^{(2)} = (0.3180, 0.3676)$; $r_2^{(2)} = (0.5429, 0.1765)$,

$r_3^{(2)} = (0.4604, 0.2485)$; $r_4^{(2)} = (0.4630, 0.2150)$; $r_5^{(2)} = (0.4704, 0.1577)$,

$r_1^{(3)} = (0.3188, 0.5783)$; $r_2^{(3)} = (0.5429, 0.3906)$; $r_3^{(3)} = (0.4604, 0.4507)$,

$r_4^{(3)} = (0.4630, 0.3790)$; $r_5^{(3)} = (0.4704, 0.3097)$.

Step 2. The IFHA operator compresses the three matrices into a single final matrix, ready for decision-making. Then for $r_1^{(1)} = (0.4217, 0.4742)$; $r_1^{(2)} = (0.3180, 0.3676)$; $r_1^{(3)} = (0.3188, 0.5783)$, where $w = (0.156837, 0.334829, 0.508333)$ and $\gamma^T = (0.592557, 0.271628, 0.135814)$, we have the computations as follows:

$\tilde{\mu}_1 = b^{(n\times\gamma_1)} = (0.4217)^{(3*0.135814)} = 0.7034$; $\tilde{\mu}_2 = b^{(n\times\gamma_2)} = (0.3180)^{(3*0.271628)} = 0.3931$; $\tilde{\mu}_3 = b^{(n\times\gamma_3)} = (0.3188)^{(3*0.592557)} = 0.1310$; $\tilde{\gamma}_1 = b^{(n\times\gamma_1)} = (0.4742)^{(3*0.135814)} = 0.7379$; $\tilde{\gamma}_2 = b^{(n\times\gamma_2)} = (0.3676)^{(3*0.271628)} = 0.4424$; $\tilde{\gamma}_3 = b^{(n\times\gamma_3)} = (0.5783)^{(3*0.592557)} = 0.3777$.

Utilizing the IFHA operator, we get,

$$\mathrm{IFHA}_{\omega,w}(\widetilde{a_1}, \widetilde{a_2}, \ldots, \widetilde{a_n}) = \left[1 - \prod_{j=1}^{n}(1 - \dot{\mu}_{\tilde{a}_{\sigma(j)}})^{w_j},\ \prod_{j=1}^{n}(\dot{\gamma}_{\tilde{a}_{\sigma(j)}})^{w_j}\right],$$

$$m_1 = \left(1 - \left[\left(1 - \mu_{a_1}\right)^{w_1}\left(1 - \mu_{a_2}\right)^{w_2}\left(1 - \mu_{a_3}\right)^{w_3}\right], \left(\gamma_{a_1}\right)^{w_1}\left(\gamma_{a_2}\right)^{w_2}\left(\gamma_{a_3}\right)^{w_3}\right),$$

$$m_1 = \left(\left(1 - \begin{bmatrix}(1-0.7034)^{0.508333} * \\ (1-0.3931)^{0.334829} * \\ (1-0.1310)^{0.156837}\end{bmatrix}\right), \begin{pmatrix}(0.7379)^{0.508333} * \\ (0.4424)^{0.334829} * \\ (0.3777)^{0.156837}\end{pmatrix}\right).$$

$m_1 = (0.5538, 0.5598)$. Similarly, we can compute all other values, and hence, $m_1 = (0.5538, 0.5598)$; $m_2 = (0.7275, 0.3699)$; $m_3 = (0.6282, 0.4406)$;$m_4 = (0.6734, 0.3792)$; $m_5 = (0.6437, 0.3133)$.

Step 3. In this step, the correlations are computed between the final $m_i's$, and the ideal IFS value $\tilde{m}_i = (0,1)$. $Cr_{ZL}(m_1, \tilde{m}_i) = 0.5598$; $Cr_{ZL}(m_2, \tilde{m}_i) = 0.1850$; $Cr_{ZL}(m_3, \tilde{m}_i) = 0.1469$; $Cr_{ZL}(m_4, \tilde{m}_i) = 0.0948$; $Cr_{ZL}(m_5, \tilde{m}_i) = 0.0627$.

Step 4. Now, the correlation coefficient is displayed for the selection of the best option: $\rho r_{ZL}(m_1, \tilde{m}_i) = 0.7036$; $\rho r_{ZL}(m_2, \tilde{m}_i) = 0.3183$; $\rho r_{ZL}(m_3, \tilde{m}_i) = 0.3303$; $\rho r_{ZL}(m_4, \tilde{m}_i) = 0.2448$; $\rho r_{ZL}(r_5, \tilde{r}_5) = 0.1955$.

Step 5. Arranging the options based on the computed $\rho r_{ZL}(m_i, \tilde{m}_i)$, we can observe that A_1 is the most preferred choice.

Proceeding with the same algorithm, but utilizing the weight vectors from complex eigenvalues, the ranking of all the alternatives using the correlations obtained, it is easy to note that even here, A_1 is the preferred choice of the available options, with a high correlation.

7 | Artificial Neural Network-MAGDM for the Decision Matrices

The numerical decision problem in the above section is attempted to be solved using different ANN techniques, thereby reducing the time and workforce required for the computational procedures.

Pseudo-code for ANN-MAGDM

C_n: n Matrix IFS dataset of size k x m

Input {Decision Matrices comprising Intuitionistic Fuzzy data values}

A_n = {Collection of n Matrices of size k};

//* Aggregation Phase*//

Compute {P-IFWG/ IFWG/OWA/OWG/G-OWA 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 a Fuzzy Column matrix}

Generate {Collective Overall Preference Intuitionistic Fuzzy Decision Matrices/ Weights (Real /Complex Eigen values based normalized vector)}

//*Improvise the input vector by IFHA operator*//

Input vector $\left\{\mathrm{IFHA}_{\omega,w}(\widetilde{a_1}, \widetilde{a_2}, \ldots, \widetilde{a_n}) = \left[1 - \prod_{j=1}^{n}(1 - \dot{\mu}_{\tilde{a}_{\sigma(j)}})^{w_j},\ \prod_{j=1}^{n}(\dot{\gamma}_{\tilde{a}_{\sigma(j)}})^{w_j}\right]\right\}$

//* Learning Phase*//

Generate {Weight Matrix by IF-Delta/Perceptron/Hebb Rule}

Update weights for next step

Continue the weight updation until the target is achieved

//*Activation function*//

Fix {The Threshold Value-Based on (Delta/Perceptron/Hebb Rule)}

While Activated values $\geq$ Threshold do

Generate {Matrix for final Decision with values exceeding the Threshold}

Output{Best Alternative(s) to be chosen}

{The final decision variable can be converted into a crisp variable, and computations can be performed}. End.

Table 1. Ranking of alternatives of proposed methods and existing models.

Sl. No.	MAGDM Method	Ranking of Alternatives
1	Proposed MAGDM: Real eigenvalues	$A_1 > A_3 > A_2 > A_4 > A_5$ The high-ranked option is A_1.
2	Proposed MAGDM: Complex eigenvalues	$A_1 > A_3 > A_2 > A_4 > A_5$ The high-ranked option is A_1.
3	ANN-MAGDM [34]: P-IFWG operator	A4, A5
4	ANN-MAGDM [34]: IFWG operator	A4, A5
5	ANN-MAGDM [34]: OWA operator	A1, A2, A3
6	ANN-MAGDM [34]: OWG operator	A2, A3
7	ANN-MAGDM [34]: G-OWA operator	A1, A2, A3
8	Classical MAGDM methods [20, 41, 42, 43] accuracy functions	$A_5 > A_2 > A_3 > A_1 > A_4$ The most desirable alternative is A_5.
9	Classical MAGDM methods [20, 41, 42, 43] Hamming distance function excluding intuitionistic degree	$A_1 > A_4 > A_3 > A_2 > A_5$ The most desirable alternative is A_5.
10	Classical MAGDM methods [20, 41, 42, 43] Hamming distance function, including intuitionistic degree	$A_1 > A_4 > A_2 > A_3 > A_5$. The most desirable alternative is A_5.

Table 1. Continued.

Ann with the delta learning rule and the input values			
Sl. No.:	**Target Values**	**Threshold Value**	**Selected Alternatives**
11	t1= -1; t2= 1; t3= -1	0.20395	A1, A4
12	t1= 1; t2= -1; t3= -1	0.20266	A1, A4
13	t1= -1; t2= -1; t3= -1	0.1915	A1, A4
14	t1= 1; t2= -1; t3= 1	0.19432	A2, A3, A5
Ann with the delta learning rule and defuzzified input values			
	Target Values	**Threshold Value**	**Selected Alternatives**
15	t1= -1; t2= 1; t3= -1	0.11884	A1, A2
16	t1= 1; t2= -1; t3= -1	0.11309	A1, A2
17	t1= -1; t2= -1; t3= -1	0.03679	A1, A2
18	t1= 1; t2= -1; t3= 1	0.19329	A1, A2
Ann with the perceptron learning rule and its input values			
	Target Values	**Threshold Value**	**Selected Alternatives**
19	t1= -1; t2= 1; t3= -1	0.17755	A1, A4
20	t1= 1; t2= -1; t3= -1	0.15263	A1, A4
21	t1= -1; t2= -1; t3= -1	0.13017	A1, A4
22	t1= 1; t2= -1; t3= 1	0.11099	A1, A4
Ann with the perceptron learning rule and defuzzified input values			
	Target Values	**Threshold Value**	**Selected Alternatives**
23	t1= -1; t2= 1; t3= -1	-0.1726	A1, A3, A5
2	t1= 1; t2= -1; t3= -1	-0.17255	A1, A3, A5
3	t1= -1; t2= -1; t3= -1	-0.17255	A1, A3, A5
4	t1= 1; t2= -1; t3= 1	0.01377	A1, A5
Ann with the Hebb learning rule			
	Data Set	**Threshold Value**	**Selected Alternatives**
1	IFS	0.35331	A2, A3, A5
2	De-fuzzified Values	2.99435	A2, A5

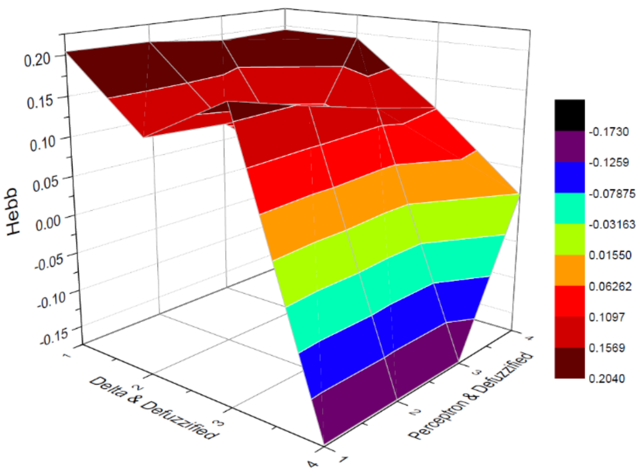

Fig. 1. Comparison of ANN thresholds with delta, perceptron, and Hebb learning rule with IFS and defuzzified input values

8 | Discussion

This study uses a variety of training techniques, including the traditional MAGDM method and guided MAGDM methods based on Artificial Neural Networks (ANNs), by calculating weight vectors from decision makers' provided linear operators in Hilbert space and thereby identifying the best options. Various ANN techniques, such as the Delta Learning Rule, Perceptron Learning Rule, and Hebbian Learning Rule, are included in the numerical example of selecting the optimal option from the list of options. Based on performance measurements and the corresponding learning methodology, each method determines the optimal option or the set of alternatives. The following are the main results and top options for each method: 1. Delta Learning Rule: Iterative weight updates converge towards optimal predictions, making all the alternatives the best option (depends on the choice of values of targets d1, d2, d3). 2. Perceptron Learning Rule: Use binary outputs to categorize acceptability; best options are A1, A3, A4, and A5. 3. Hebbian Learning Rule: A2, A3, and A5 are the best options, employing weight updates that are directly impacted by correlations between input and output.

Observations regarding methodological variance: The underlying changes in the learning mechanisms, weight update rules, and evaluation criteria of the various ANN approaches account for the observed diversity in the best options. The leading causes of these variations are listed below:

Learning objectives and optimization: To suit the decision-making problem, each method optimizes weights in a unique way. For example, the Delta Learning Rule, offers a more straightforward weight adjustment process to reduce inconsistencies.

Weight update mechanisms: Hebbian learning is more sensitive to high-impact features since its weight updates are entirely dependent on the connection between inputs and outcomes. As seen with the Alternatives, this may result in the selection of options that have a significant impact on the weight calculation. Instead of producing a single option, the Perceptron Learning Rule produces many best alternatives (A1, A3, A4 and A5) by using binary classification (1 or 0) to find acceptable alternatives.

Sensitivity to input data: Iterative techniques that constantly modify weights throughout epochs include the delta learning rule. Depending on how well they match the desired outputs during optimization, their sensitivity to initial weights and error functions may result in the favouring of distinct alternatives.

Evaluation criteria: The ways in which the approaches rank and evaluate the options vary. For instance, Perceptron examines accept/reject classifications based on a threshold, but Hebbian Learning favours the option with the largest net input by evaluating alternatives based on their cumulative contributions to weight adjustments.

Approximation and complexity: While more sophisticated techniques capture subtle correlations between attributes and outputs, simpler techniques like Hebbian Learning may approximate solutions that match prominent patterns in the data. Different best choices may be chosen as a result of this approximation discrepancy.

By understanding these distinctions, decision-makers can choose the most suitable ANN method based on the problem's requirements, such as precision, computational efficiency, or interpretability of results. Including this perspective enriches the comparative analysis and highlights the methodological diversity in applying ANNs to decision-making problems.

9 | Conclusion

This study presents a novel ANN method using some linear space techniques namely the Spectral theory 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 some aggregation operators. In the subsequent sections this paper offers a thorough assessment of ANN methods for MAGDM applications, emphasizing their versatility and capacity to improve judgment in challenging situations. The adaptability and efficiency of several ANN training techniques in addressing MAGDM challenges are illustrated by this comparison study. The Hebbian Learning Rule offered an easy-to-use weight modification mechanism based on input-output correlations, whereas the Perceptron

Learning Rule provided a simple framework for decision-making with binary categorization, whereas the Delta Learning Rule demonstrated exceptional convergence efficiency.

The problem's complexity, the level of precision required, and the available computing power all influence the method selection. Finally applying ANN to MAGDM problem solving techniques significantly save time compared to traditional hand-calculated MAGDM methods by automating weight computations and iterative evaluations, enabling faster and more efficient decision-making for complex problems. 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.

Acknowledgments

The authors would like to express their sincere gratitude to all colleagues and institutions that provided support and guidance during the course of this research.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] Atanassov, K., Sotirov, S., & Angelova, N. (2020). Intuitionistic fuzzy neural networks with interval valued intuitionistic fuzzy conditions. In *Intuitionistic and type-2 fuzzy logic enhancements in neural and optimization algorithms: theory and applications* (pp. 99–106). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-35445-9_9

[2] Atanassov, K., Sotirov, S., & Pencheva, T. (2023). Intuitionistic fuzzy deep neural network. *Mathematics*, *11*(3), 1–14. https://doi.org/10.3390/math11030716

[3] Firouzkouhi, N., Amini, A., Nazari, M., Alkhatib, F., Bordbar, H., Cheng, C., … & Rashidi, M. (2023). Advanced artificial intelligence system by intuitionistic fuzzy Gamma-subring for automotive robotic manufacturing. *Artificial intelligence review*, *56*(9), 9639–9664. https://doi.org/10.1007/s10462-023-10396-5

[4] Hajek, P., Olej, V., Froelich, W., & Novotny, J. (2021). Intuitionistic fuzzy neural network for time series forecasting - the case of metal prices. *Artificial intelligence applications and innovations* (pp. 411–422). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-79150-6_33

[5] Kuo, R. J., & Cheng, W. C. (2019). An intuitionistic fuzzy neural network with gaussian membership function. *Journal of intelligent & fuzzy systems*, *36*(6), 6731–6741. https://doi.org/10.3233/JIFS-18998

[6] Leonishiya, A., & Robinson, P. J. (2023). A fully linguistic intuitionistic fuzzy artificial neural network model for decision support systems. *INDIAN journal of science and technology*, *16*, 29–36. https://doi.org/10.17485/IJST/v16iSP4.ICAMS136

[7] Petkov, T., Bureva, V., & Popov, S. (2021). Intuitionistic fuzzy evaluation of artificial neural network model. *Notes on intuitionistic fuzzy sets*, *27*(4), 71–77. http://dx.doi.org/10.7546/nifs.2021.27.4.71-77

[8] Robinson, P. J., & Leonishiya, A. (2024). Application of varieties of learning rules in intuitionistic fuzzy artificial neural network. *Machine intelligence for research and innovations* (pp. 35–45). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-8129-8_4

[9] Robinson, P. J., & Saranraj, A. (2024). Intuitionistic fuzzy gram-schmidt orthogonalized artificial neural network for solving MAGDM Problems. *Indian journal of science and technology*, *17*, 2529–2537. https://doi.org/10.17485/IJST/v17i24.1386

[10] Samet, S., Laouar, M. R., Bendib, I., & Eom, S. (2022). Analysis and prediction of diabetes disease using machine learning methods. *International journal of decision support system technology (IJDSST)*, *14*(1), 1–19. https://doi.org/10.4018/IJDSST.303943

[11] Santos Neto, J. B. S. dos, & Costa, A. P. (2023). A multi-criteria decision-making model for selecting a maturity model. *International journal of decision support system technology*, *15*, 1–15. https://doi.org/10.4018/IJDSST.319305

[12] Taherdoost, H. (2023). Deep learning and neural networks: Decision-making implications. *Symmetry*, *15*(9), 1–22. https://doi.org/10.3390/sym15091723

[13] Hussain, W., Merigó, J. M., Gil-Lafuente, J., & Gao, H. (2023). Complex nonlinear neural network prediction with IOWA layer. *Soft computing*, *27*(8), 4853–4863. https://doi.org/10.1007/s00500-023-07899-2

[14] Zhang, Q., Li, H., Lu, X., & Wu, C. (2022). A neural network-based approach to multi-attribute group decision-making with heterogeneous preference information. *Scientific programming*, *2022*(1), 9033237. https://doi.org/10.1155/2022/9033237