SoftwareX 28 (2024) 101889

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

PyDEMATEL: A Python-based tool implementing DEMATEL and fuzzy DEMATEL methods for improved decision making

Abderrahman Chekry [a], Jamal Bakkas [a], Mohamed Hanine [b], Elizabeth Caro Montero [c,d,e], Mirtha Silvana Garat de Marin [c,f,g], Imran Ashraf [h,*]

[a] *LAPSSII Laboratory, Graduate School of Technology, Cadi Ayyad University, Safi, Morocco*
[b] *LTI Laboratory, ENSA, Chouaib Doukkali University, El Jadida, Morocco*
[c] *Universidad Europea del Atlantico, Isabel Torres 21, 39011 Santander, Spain*
[d] *Universidad Internacional Iberoamericana, Campeche 24560, Mexico*
[e] *Universidad de La Romana, La Romana, Dominican Republic*
[f] *Universidad Internacional Iberoamericana, Arecibo, PR 00613, USA*
[g] *Universidade Internacional do Cuanza, Cuito, Bie, Angola*
[h] *Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, South Korea*

ARTICLE INFO

Keywords:
DEMATEL
Fuzzy DEMATEL
MCDM
Python
Decision-making

ABSTRACT

In the context of decision-making, the DEMATEL (Decision Making Trial and Evaluation Laboratory) method stands out for its systematic approach to complex systems. By incorporating fuzzy logic, the DEMATEL fuzzy method takes traditional techniques a step further, effectively managing the uncertainties and imprecision inherent in expert assessments. This hybrid method has proved useful in a variety of fields, including business, engineering, healthcare, environmental management, and education. Its ability to refine subjective judgments into actionable information enables decision-makers to improve organizational performance, optimize resource allocation, and achieve more accurate results. The development of software tools for these methods makes them more accessible and practical, enabling more effective analysis and application. In this paper, we propose a flexible implementation that integrates seamlessly into Python-based applications, offering full access to all parameters, matrices, and intermediary calculations of the method. Additionally, the tool also provides a user-friendly graphical interface.

Code metadata

Current code version	0.2.2
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-24-00301
Code Ocean compute capsule	
Legal Code License	BSD-3-Clause
Code versioning system used	none
Software code languages, tools, and services used	Python3.11
Compilation requirements, operating environments & dependencies	numpy, matplotlib, openpyxl, tkinter
If available Link to developer documentation/manual	
Support email for questions	a.chekry@uca.ac.ma

Software metadata

Current software version	0.2.2
Permanent link to executables of this version	https://github.com/achekry/-Elsevier-SoftwareX-PyDEMATEL or https://pypi.org/project/pyDEMATEL/
Legal Software License	BSD-3-Clause
Computing platforms/Operating Systems	Linux, Microsoft Windows, macOS
Installation requirements & dependencies	numpy, matplotlib, openpyxl, tkinter
If available, link to user manual — if formally published include a reference to the publication in the reference list	
Support email for questions	a.chekry@uca.ac.ma

* Corresponding author.
E-mail address: imranashraf@ynu.ac.kr (Imran Ashraf).

Available online 9 October 2024
https://doi.org/10.1016/j.softx.2024.101889
Received 3 June 2024; Received in revised form 21 August 2024; Accepted 4 September 2024

1. Introduction

MCDM (Multi-Criteria Decision-Making) is an important field of decision-making that focuses on developing computational and mathematical models for the subjective evaluation of several options, with a consideration of various factors or criteria. One such pivotal technique in this field is the DEMATEL (Decision-Making Trial and Evaluation Laboratory) Method, which offers a structured approach to analyze complex systems and MCDM scenarios across diverse fields. Originating from the Battelle Memorial Institute in 1972 [1], The DEMATEL method uses graph theory and matrix algebra to identify and evaluate interdependencies and cause-and-effect relationships between different factors, providing decision-makers with insights for strategic decision-making, resource optimization, and risk mitigation to enhance organizational performance [2]. The technique aims to classify factors into two distinct groups: the “cause factors group”, composed of factors triggering or causing effects within a system or process, and the “effect factors group”, composed of factors influenced or affected by the actions of cause factors. Several enhancements have been made to the DEMATEL method, resulting in various variants, including the Fuzzy DEMATEL approach [3]. This method enhances the classic DEMATEL approach by incorporating fuzzy logic to better manage uncertainty and imprecision in evaluating relationships between criteria. This allows experts to express their opinions in linguistic terms rather than crisp values, leading to more realistic and robust relationship matrices. As a result, Fuzzy DEMATEL offers better aggregation of divergent opinions, more flexible reasoning, and broader applicability in complex contexts where human judgments are often vague and subjective.

DEMATEL, especially in its fuzzy extension, stands out for its ability to model interrelationships and causal effects between criteria, unlike methods such as AHP or TOPSIS which treat criteria as independent elements. Its ability to capture complex interdependencies provides a more stable basis for decision-making when these relationships are crucial, which improves decision-making in complex systems and makes DEMATEL very versatile and effective in various fields. For example, it was used to develop and analyze a set of sustainability indicators for Indian automotive component SMEs using expert interviews, identifying carbon management as the most influential indicator for measuring sustainability [4]. In another scenario, the integration of DEMATEL with Fermatean Fuzzy Sets (FFS) was employed to evaluate the implementation of Education 4.0 (EDUC4) in Philippine universities, identifying cost-related obstacles as critical barriers [5]. The fuzzy DEMATEL method was used in a study aimed at mitigating the risk of anchor loss in ships by analyzing the causal relationships of critical factors, including environmental conditions, crew competence, and anchoring methods [6]. It has also been used in combination with regret theory to develop a multi-attribute model for the selection of competent construction program managers in China, taking into account both psychological behaviors and interrelationships between attributes, leading to more comprehensive and scientific manager selections [7]. Another study uses this technique to develop and prioritize recovery action plans for ecotourism centers affected by the Covid-19 pandemic, identifying “standardization of centers” and “estimating the number of requests and increasing capacity” as top priorities [8]. In the healthcare field, Fuzzy DEMATEL analysis is used to identify 15 key factors influencing sustainable healthcare waste management, revealing that the top three crucial factors are a sound legal framework, sustainable management training, and positive attitudes, thus helping to improve the sustainability of healthcare waste management practices [9]. As a final example, we end this list with the field of education, where the Fuzzy DEMATEL method was used to determine the interrelationships between evaluation criteria for learning management systems (LMS), identifying user satisfaction, ease of learning, and ease of use as the most important of the twelve criteria considered [10].

Although DEMATEL and fuzzy DEMATEL methods are widely used in various fields to analyze complex interactions between different criteria, these methods do not yet have a sufficiently powerful, flexible, and widely usable tool to highlight the potential and effectiveness of these methods in wider and more varied application contexts. In this paper, we propose Python-based software to fill this gap.

2. Motivation

Several tools are available for MCDM Methods. For instance, the TOPSIS method (Technique for Order Preference by Similarity to Ideal Solution) [11] is implemented using the PyTOPSIS tool [12], and the OPA method (Order Preference by Analysis) [13] is available using the PyOPASolver tool [14]. Despite the widespread use of the DEMATEL method in various fields, there are only two available tools: PyDecision [15] and Julia library [16]. PyDecision offers two straightforward functions: one for implementing DEMATEL and another for Fuzzy DEMATEL. These functions directly return the prominences, relations, and weights without exposing intermediate matrices, such as the direct influence matrix, the total influence matrix, and others. On the other hand, the Julia library does not support the Fuzzy DEMATEL method. What is more, neither implementation offers a graphical interface. In this paper, we propose a flexible implementation that can be seamlessly integrated into Python-based applications providing access to all parameters, matrices, and intermediary calculations of the method. In addition, it can be run in graphical mode with a user-friendly, feature-rich interface designed for novice users. This interface allows users to view results and export them to Excel for use outside the software.

3. Description of DEMATEL and fuzzy DEMATEL methods

3.1. DEMATEL method

As detailed in [1], the DEMATEL method involves a set E of m expert evaluators $E = \{E_1, E_2, \ldots, E_m\}$ to assess the interdependence among a set F of n factors $F = \{F_1, F_2, \ldots, F_n\}$. Each expert E_k proposes a matrix Z_k, where all main diagonal elements are null, and Z_{ij}^k defines the degree of influence of factor F_i on factor F_j from the point of view of the $k^t h$ expert. The method proposes five values to evaluate the influence between two factors: 0: “no influence (NO)“, 1: ”very low influence (VL)“, 2: ”low influence (L)“, 3: ”high influence (H)“, and 4: ”very high influence (VH)”. belongs thus to the set 0,1,2,3,4.

Step 1: Direct influence matrix:

$$Z = \frac{1}{m} \sum_{k=1}^{m} Z_k \tag{1}$$

With:

$$z_{ij} = \frac{1}{m} \sum_{k=1}^{m} z_{ij}^k, i, j = 1, 2, \ldots, n \tag{2}$$

Step 2: Normalized direct-influence matrix: :

$$X = \frac{Z}{\max\left(\max\left(\sum_{i=1}^{n} z_{ij}\right), \max\left(\sum_{i=1}^{n} z_{ij}\right)\right)} \tag{3}$$

Step 3: Total influence matrix :

$$T = X(I - X)^{-1} \tag{4}$$

Step 4: Influential relation map (IRM) :

$$R = [r_{ij}] = \left(\sum_{j=1}^{n} t_{ij}\right)_{nx1} \tag{5}$$

And C is calculated using Eq. (6)

$$C = [c_{ij}] = \left(\sum_{i=1}^{n} t_{ij}\right)_{1xn} \tag{6}$$

The sum of R+C, also called “Prominence”, reflects the significance or importance of a factor within the system or process analyzed. Factors with higher prominence values are considered more influential or critical in driving changes or effects within the system. While the R-C subtraction, also called “Relation”, measures how strongly factors are interconnected or dependent on each other.

Table 1
Fuzzy triangular numbers used to express linguistic terms.

Verbal variables influence	No.	Very low	Low	High	Very high
Fuzzy numbers	(0.0, 0.0, 0.25)	(0.0, 0.25, 0.5)	(0.25, 0.5, 0.75)	(0.5, 0.75, 1.0)	(0.75, 1.0, 1.0)

3.2. Fuzzy DEMATEL method

As detailed in [3], the fuzzy DEMATEL method follows the same steps as the classic DEMATEL method, except that it uses fuzzy triangular numbers instead of crisp values.

The linguistic scale selected for this tool is the one most frequently used in the literature [17–20], This scale is converted into triangular fuzzy numbers (TFN), as outlined in Table 1.

In this context, $\tilde{Z}$, $\tilde{X}$, $\tilde{T}$, ($\tilde{R}$, $\tilde{C}$) represent the Fuzzy Direct Influence Matrix, the Normalized Direct Influence Fuzzy Matrix, the Total Influence Fuzzy Matrix, and the Influential Relation Map (IRM), respectively. These matrices are calculated similarly to Z (Eq. (1)), X (Eq. (3)), T (Eq. (4)), and (R, C) (Eq. (5), (6)), but using fuzzy values instead of crisp values.

After these calculations, the defuzzification of $\tilde{R}$ and $\tilde{C}$ is performed to obtain the crisp values for R and C. These values are used to calculate the "Prominence" $(R + C)$ and "Relation" $(R - C)$. In this step, we use BNP Eq. (7)(Best Non-fuzzy Performance) a widely used defuzzification method for transforming fuzzy values into crisp values, particularly effective for triangular fuzzy numbers.

$$BNP = \frac{(u - l) + (m - l)}{3} + l \quad (7)$$

Algorithm 1: Algorithm followed by PyDEMATEL using the DEMATEL method in pseudo-code form

1. Inputs :
   (a) Read number of experts: nbExperts
   (b) Read number of factors: nbFactors
   (c) For each expert E_i , read name
   (d) For each factor F_i , read name
   (e) For each expert E_k, generate a tab to insert the Z_k matrix elements
   (f) For each expert E_k , fill the individual direct-influence matrix Z_k
2. Step 1: Generate the direct influence matrix Z using Eq. (1)
3. Step 2: Generate the normalized direct-influence matrix X using Eq. (3)
4. Step 3: Generate the total-influence matrix T using Eq. (4)
5. Step 4: Generate the influential Relation Map IRM using Eq.((5) and (6))

4. Software architecture

The PyDEMATEL software comprises three distinct classes: DEMATELSolver, FuzzyDEMATELSolver, and DEMATELWindow. The DEMATELWindow class serves as the tool's graphical interface, as shown in Fig. 1, while the DEMATELSolver and FuzzyDEMATELSolver classes implement the DEMATEL and Fuzzy DEMATEL methods respectively. The tool offers two modes of operation: decision-makers can use it directly through a user-friendly graphical interface, while developers can integrate the DEMATELSolver or FuzzyDEMATELSolver packages into their applications without the need for the graphical interface. PyDEMATEL is available from the Python Package Index (PyPI) under a BSD license. It can be installed using the command `pip install pyDEMATEL`.

4.1. DEMATELWindow

Fig. 1 shows the DEMATELWindow, the GUI for the PyDEMATEL application. This window is divided into several sections: inputs are handled in the first three, while the last section is reserved for results and intermediate calculations. The initial input section allows users to specify the number of experts and factors involved. The second one, which is automatically generated from the first one, allows users to enter the names of experts or evaluators and the names of factors. In addition, this block contains two buttons that allow users to select the DEMATEL or fuzzy DEMATEL method to be used for calculations. The third input section, automatically generated based on the first two sections, presents a series of tabs, each one representing a specific expert. Within each expert's tab, there is an interface for entering their evaluation matrix. The output section displays intermediate calculations such as the "Direct-Influence Matrix Z", the "Normalized Direct-Influence Matrix X", and the "Total-Influence Matrix T". It also presents the method's results, including the "Influential Relation Map (IRM)" and the "Causal Diagram" as a graph, each in a separate tab. For the Fuzzy DEMATEL method, there is an additional tab labeled "Fuzzy IRM". Furthermore, it is possible to export all these results in Excel format.

4.2. DEMATELSolver

The DEMATELSolver class provides a complete, streamlined implementation of the DEMATEL method, including attributes and methods that manage each step, from acquiring data and creating intermediate models to presenting the final results. This class also includes a method for plotting the causal influence graph. Appendix A describes the principal methods of the DEMATELSolver class, divided into three sets. The first set comprises input methods, which involve acquiring the numbers and names of experts and factors, as well as the evaluation matrix values provided by each expert. Once the input has been acquired, DEMATELSolver generates several matrices in the following steps: the first step produces "direct-influence matrix Z" using Eq. (1), the second step generates "normalized direct-influence matrix X" using Eq. (3), the third step creates the "total-influence matrix T" using Eq. (4), and the fourth step produces an "influential Relation Map IRM" using Eq. (5) and (6). Each step is supported by a specific method, which guarantees a complete and accurate implementation of the DEMATEL method. The last set is dedicated to methods that present outputs within a variety of formats, including graphics and the possibility of generating an Excel document.

4.3. FuzzyDEMATELSolver

Similar to the DEMATELSolver class described in the previous section, the FuzzyDEMATELSolver class follows nearly the same steps. It encompasses all stages of the process, including data acquisition, intermediate model creation, and final result presentation. This class also includes a method for plotting the causal influence graph. Appendix A presents the principal methods of the FuzzyDEMATELSolver class, organized into three sets. The first set of these methods gathers the numbers and names of the experts and factors, as well as the evaluation matrix values for each expert. The difference with the previous method lies in the use of triangular fuzzy numbers instead of crisp values. once these data has been acquired, FuzzyDEMATELSolver generates several matrices following these steps: the first step produces "fuzzy direct

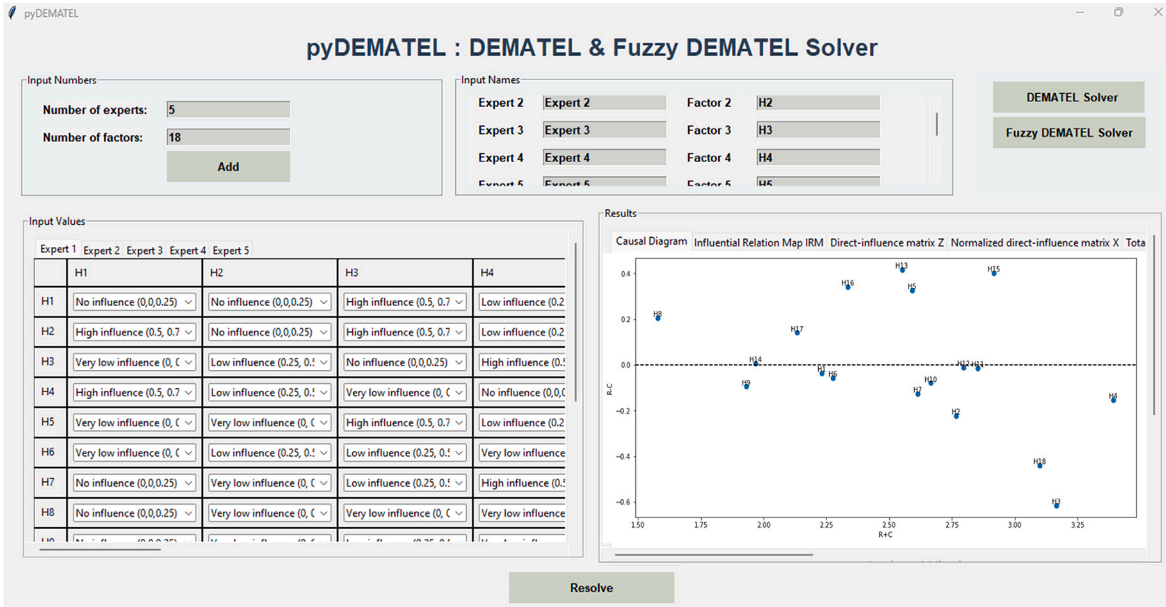


Fig. 1. The graphical interface of PyDEMATEL.

influence matrix $\tilde{Z}$", the second step generates a "normalized direct-influence fuzzy matrix $\tilde{X}$", the third step creates "Total influence fuzzy matrix $\tilde{T}$", and the fourth step produces an "influential Relation Map IRM". Each step is supported by a specific method, ensuring a complete and accurate implementation of the Fuzzy DEMATEL method. The last set is dedicated to methods that present outputs within a variety of formats, including graphics and the possibility of generating an Excel document.

Designed primarily for integration into Python-based programs using the DEMATEL and Fuzzy DEMATEL approach, the DEMATELSolver and FuzzyDEMATELSolver classes offer developers a flexible and robust toolbox, thanks to the diverse and rich range of its methods.

5. Illustrative example

To assess and validate the PyDEMATEL tool, we used data from the example presented in the application section presented in [18] to compare the results obtained. This example aims to make an objective decision regarding the management of risks associated with the transport of cargo by oil tankers, which pose significant risks to human life and the marine environment due to the dangerous nature of the cargo. Oil tanker degassing is a critical and risky operation, involving toxic and flammable gasses that can seriously harm the environment and the crew. Awareness of the consequences of these risks is essential for the crew. Management companies and professional managers pay particular attention to managing these operational risks. In this example, emphasis is placed on eighteen critical operational risks which express the input factors for our PyDEMATEL named F1, F2, …, F18, that can have catastrophic consequences during the degassing of a tanker. We used as input to our software the linguistic evaluation matrix of marine experts' consensus Table 2, which is considered as the individual direct influence matrix of a group of five experts.

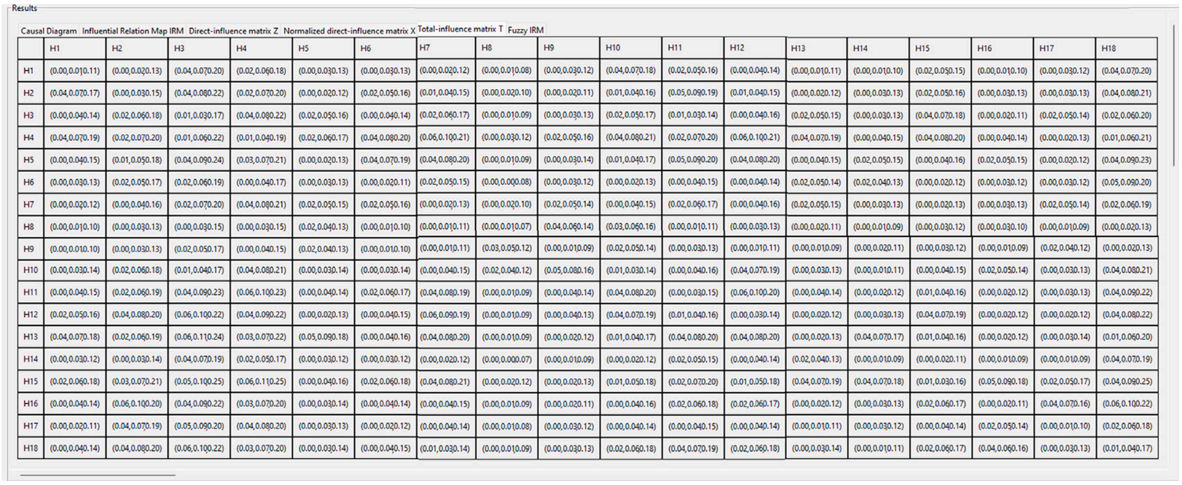
Once all the necessary data has been entered, the tool generates a series of tabs in the results section containing both intermediate and final outcomes. It produces the normalized direct influence matrix X, the direct influence matrix Z, and the total influence matrix T as shown in Fig. 2. Additionally, it generates the influence relationship map IRM depicted in Fig. 3 and the causality diagram presented in Fig. 1. All results obtained using the PyDEMATEL tool are accurate and identical to those presented by the authors in their article, confirming that the tool correctly implements the DEMATEL method.

6. Impact

PyDEMATEL is designed to offer a robust and flexible solution for implementing the DEMATEL and Fuzzy DEMATEL methods, ready to be used and integrated in various domains to solve MCDM problems. The tool has been tested to solve decision-making problems in the maritime industry and used to identify critical operational risks related to tanker degassing by visualizing the interdependencies and cause–effect relationships between the various risks. As a result, it was easier to implement more effective and targeted management strategies to minimize the risks and their potential consequences. A python-based tool that is dedicated to MCDM DEMATEL and Fuzzy DEMATEL methods. It offers a user-friendly interface, targeting various user groups and facilitating informed decision-making processes. Its intuitive graphical interface enables users from different fields to make decisions based on factors and expert assessments. The interface simplifies data entry and presents results both numerically and graphically with the ability to export them in Excel format, improving decision-making. Beyond its user-oriented features, PyDEMATEL extends its usefulness to researchers and Python developers looking to integrate decision-making capabilities into their projects. By integrating its package, users have access to a complete toolbox including data acquisition, problem-solving, and result output methods. This seamless integration not only simplifies the development process but also ensures its robustness and flexibility across a variety of applications. The software is licensed under the BSD license, allowing it to be used in academic contexts for educational purposes. Using this tool in education can enhance the learning experience by addressing real-world MCDM challenges. This versatile software bridges the gap between theory and practice, enabling users at different levels to make informed decisions with confidence.

7. Conclusion

In this study, we present PyDEMATEL, a software tool that implements both the DEMATEL and fuzzy DEMATEL decision-making methods. This tool is designed to solve MCDM problems across various domains and offers a user-friendly, feature-rich graphical interface. PyDEMATEL allows users to generate the matrices required for decision-making processes, display a causality graph that illustrates the dependencies between factors to easily identify the most critical or

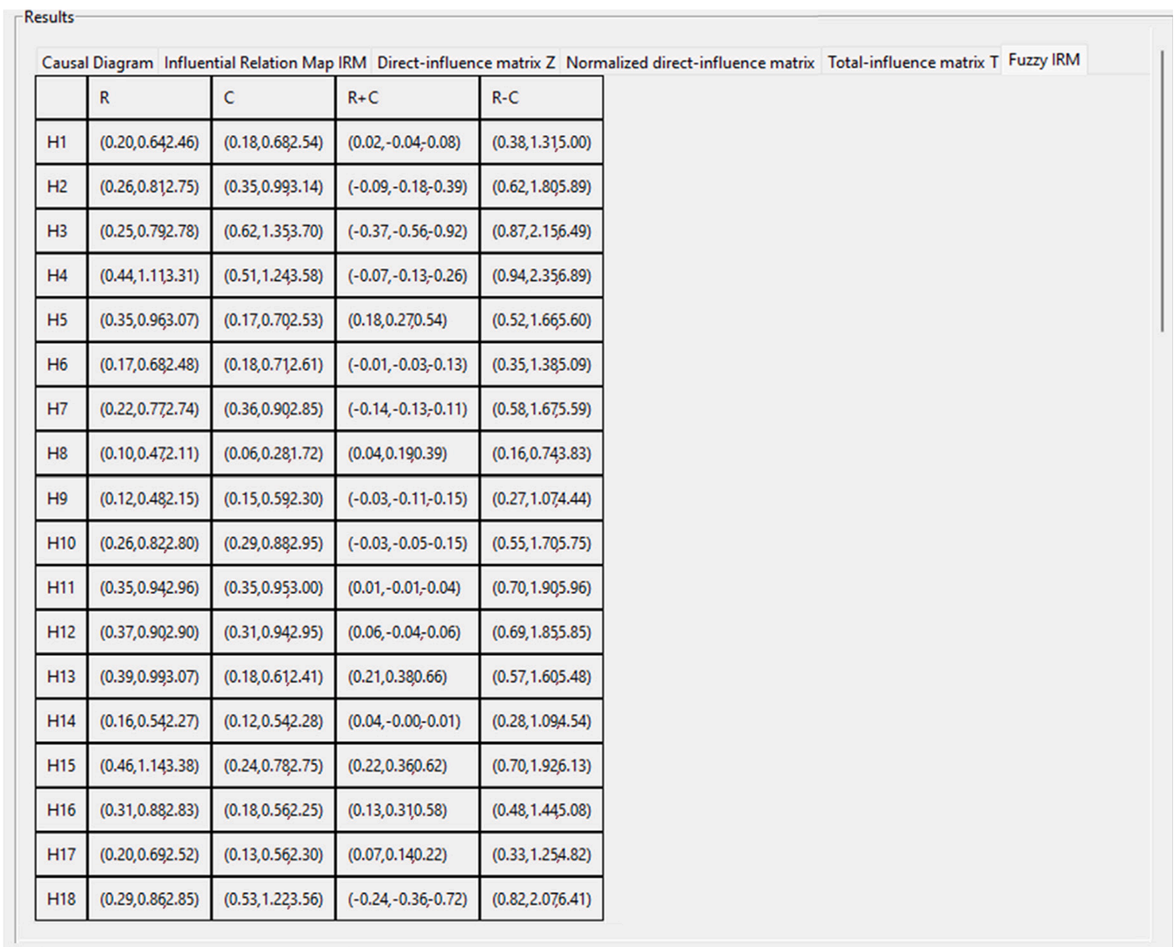

Results

Causal Diagram | Influential Relation Map IRM | Direct-influence matrix Z | Normalized direct-influence matrix X | Total-influence matrix T | Fuzzy IRM

	H1	H2	H3	H4	H5	H6	H7	H8	H9
H1	(0.00,0.01,0.11)	(0.00,0.02,0.13)	(0.04,0.07,0.20)	(0.02,0.06,0.18)	(0.00,0.03,0.13)	(0.00,0.03,0.13)	(0.00,0.02,0.12)	(0.00,0.01,0.08)	(0.00,0.03,0.12)
H2	(0.04,0.07,0.17)	(0.00,0.03,0.15)	(0.04,0.08,0.22)	(0.02,0.07,0.20)	(0.00,0.02,0.12)	(0.02,0.05,0.16)	(0.01,0.04,0.15)	(0.00,0.02,0.10)	(0.00,0.02,0.11)
H3	(0.00,0.04,0.14)	(0.02,0.06,0.18)	(0.01,0.03,0.17)	(0.04,0.08,0.22)	(0.02,0.05,0.16)	(0.00,0.04,0.14)	(0.02,0.06,0.17)	(0.00,0.01,0.09)	(0.00,0.03,0.13)
H4	(0.04,0.07,0.19)	(0.02,0.07,0.20)	(0.01,0.06,0.22)	(0.01,0.04,0.19)	(0.02,0.06,0.17)	(0.04,0.08,0.20)	(0.06,0.10,0.21)	(0.00,0.03,0.12)	(0.02,0.05,0.16)
H5	(0.00,0.04,0.15)	(0.01,0.05,0.18)	(0.04,0.09,0.24)	(0.03,0.07,0.21)	(0.00,0.02,0.13)	(0.04,0.07,0.19)	(0.04,0.08,0.20)	(0.00,0.01,0.09)	(0.00,0.03,0.14)
H6	(0.00,0.03,0.13)	(0.02,0.05,0.17)	(0.02,0.06,0.19)	(0.00,0.04,0.17)	(0.00,0.03,0.13)	(0.00,0.02,0.11)	(0.02,0.05,0.15)	(0.00,0.00,0.08)	(0.00,0.03,0.12)
H7	(0.00,0.02,0.12)	(0.00,0.04,0.16)	(0.02,0.07,0.20)	(0.04,0.08,0.21)	(0.02,0.05,0.15)	(0.02,0.05,0.16)	(0.00,0.02,0.13)	(0.00,0.02,0.10)	(0.02,0.05,0.14)
H8	(0.00,0.01,0.10)	(0.00,0.03,0.13)	(0.00,0.03,0.15)	(0.00,0.03,0.15)	(0.02,0.04,0.13)	(0.00,0.01,0.10)	(0.00,0.01,0.11)	(0.00,0.01,0.07)	(0.04,0.06,0.14)
H9	(0.00,0.01,0.10)	(0.00,0.03,0.13)	(0.02,0.05,0.17)	(0.00,0.04,0.15)	(0.02,0.04,0.13)	(0.00,0.01,0.10)	(0.00,0.01,0.11)	(0.03,0.05,0.12)	(0.00,0.01,0.09)
H10	(0.00,0.03,0.14)	(0.02,0.06,0.18)	(0.01,0.04,0.17)	(0.04,0.08,0.21)	(0.00,0.03,0.14)	(0.00,0.03,0.14)	(0.00,0.04,0.15)	(0.02,0.04,0.12)	(0.05,0.08,0.16)
H11	(0.00,0.04,0.15)	(0.02,0.06,0.19)	(0.04,0.09,0.23)	(0.06,0.10,0.23)	(0.00,0.04,0.14)	(0.02,0.06,0.17)	(0.04,0.08,0.19)	(0.00,0.01,0.09)	(0.00,0.04,0.14)
H12	(0.02,0.05,0.16)	(0.04,0.08,0.20)	(0.06,0.10,0.22)	(0.04,0.09,0.22)	(0.00,0.02,0.13)	(0.00,0.04,0.15)	(0.06,0.09,0.19)	(0.00,0.01,0.09)	(0.00,0.04,0.13)
H13	(0.04,0.07,0.18)	(0.02,0.06,0.19)	(0.06,0.11,0.24)	(0.03,0.07,0.22)	(0.05,0.09,0.18)	(0.00,0.04,0.16)	(0.04,0.08,0.20)	(0.00,0.01,0.09)	(0.00,0.02,0.12)
H14	(0.00,0.03,0.12)	(0.00,0.03,0.14)	(0.04,0.07,0.19)	(0.02,0.05,0.17)	(0.00,0.03,0.12)	(0.00,0.03,0.12)	(0.00,0.02,0.12)	(0.00,0.00,0.07)	(0.00,0.01,0.09)
H15	(0.02,0.06,0.18)	(0.03,0.07,0.21)	(0.05,0.10,0.25)	(0.06,0.11,0.25)	(0.00,0.04,0.16)	(0.02,0.06,0.18)	(0.04,0.08,0.21)	(0.00,0.02,0.12)	(0.00,0.02,0.13)
H16	(0.00,0.04,0.14)	(0.06,0.10,0.20)	(0.04,0.09,0.22)	(0.03,0.07,0.20)	(0.00,0.03,0.14)	(0.00,0.04,0.14)	(0.00,0.04,0.15)	(0.00,0.01,0.09)	(0.00,0.02,0.11)
H17	(0.00,0.02,0.11)	(0.04,0.07,0.19)	(0.05,0.09,0.20)	(0.04,0.08,0.20)	(0.00,0.03,0.13)	(0.00,0.02,0.12)	(0.00,0.04,0.14)	(0.00,0.01,0.08)	(0.00,0.03,0.12)
H18	(0.00,0.04,0.14)	(0.04,0.08,0.20)	(0.06,0.10,0.22)	(0.03,0.07,0.20)	(0.00,0.03,0.14)	(0.00,0.04,0.15)	(0.01,0.03,0.14)	(0.00,0.01,0.09)	(0.00,0.03,0.13)

	H10	H11	H12	H13	H14	H15	H16	H17	H18
H1	(0.04,0.07,0.18)	(0.02,0.05,0.16)	(0.00,0.04,0.14)	(0.00,0.01,0.11)	(0.00,0.01,0.10)	(0.02,0.05,0.15)	(0.00,0.01,0.10)	(0.00,0.03,0.12)	(0.04,0.07,0.20)
H2	(0.01,0.04,0.16)	(0.05,0.09,0.19)	(0.01,0.04,0.15)	(0.00,0.02,0.12)	(0.00,0.03,0.13)	(0.02,0.05,0.16)	(0.00,0.03,0.13)	(0.00,0.03,0.13)	(0.04,0.08,0.21)
H3	(0.02,0.05,0.17)	(0.01,0.03,0.14)	(0.00,0.04,0.16)	(0.02,0.05,0.15)	(0.00,0.03,0.13)	(0.04,0.07,0.18)	(0.00,0.02,0.11)	(0.02,0.05,0.14)	(0.02,0.06,0.20)
H4	(0.04,0.08,0.21)	(0.02,0.07,0.20)	(0.06,0.10,0.21)	(0.04,0.07,0.19)	(0.00,0.04,0.15)	(0.04,0.08,0.20)	(0.00,0.04,0.14)	(0.00,0.02,0.13)	(0.01,0.06,0.21)
H5	(0.01,0.04,0.17)	(0.05,0.09,0.20)	(0.04,0.08,0.20)	(0.00,0.04,0.15)	(0.02,0.05,0.15)	(0.00,0.04,0.16)	(0.02,0.05,0.15)	(0.00,0.02,0.12)	(0.04,0.09,0.23)
H6	(0.00,0.02,0.13)	(0.00,0.04,0.15)	(0.00,0.04,0.14)	(0.02,0.05,0.14)	(0.02,0.04,0.13)	(0.00,0.02,0.12)	(0.00,0.03,0.12)	(0.00,0.03,0.12)	(0.05,0.09,0.20)
H7	(0.00,0.04,0.15)	(0.02,0.06,0.17)	(0.00,0.04,0.16)	(0.02,0.05,0.15)	(0.00,0.03,0.13)	(0.00,0.02,0.13)	(0.00,0.03,0.13)	(0.02,0.05,0.14)	(0.02,0.06,0.19)
H8	(0.03,0.06,0.16)	(0.00,0.01,0.11)	(0.00,0.03,0.13)	(0.00,0.02,0.11)	(0.00,0.01,0.09)	(0.00,0.03,0.12)	(0.00,0.03,0.10)	(0.00,0.01,0.09)	(0.00,0.02,0.13)
H9	(0.02,0.05,0.14)	(0.00,0.03,0.13)	(0.00,0.01,0.11)	(0.00,0.01,0.09)	(0.00,0.02,0.11)	(0.00,0.03,0.12)	(0.00,0.01,0.09)	(0.02,0.04,0.12)	(0.00,0.02,0.13)
H10	(0.01,0.03,0.14)	(0.00,0.04,0.16)	(0.04,0.07,0.19)	(0.00,0.03,0.13)	(0.00,0.01,0.11)	(0.00,0.04,0.15)	(0.02,0.05,0.14)	(0.00,0.03,0.13)	(0.04,0.08,0.21)
H11	(0.04,0.08,0.20)	(0.00,0.03,0.15)	(0.06,0.10,0.20)	(0.00,0.04,0.14)	(0.00,0.02,0.12)	(0.01,0.04,0.16)	(0.00,0.02,0.12)	(0.00,0.03,0.13)	(0.04,0.09,0.22)
H12	(0.04,0.07,0.19)	(0.01,0.04,0.16)	(0.00,0.03,0.14)	(0.00,0.02,0.12)	(0.00,0.03,0.13)	(0.04,0.07,0.19)	(0.00,0.02,0.12)	(0.00,0.02,0.12)	(0.04,0.08,0.22)
H13	(0.01,0.04,0.17)	(0.04,0.08,0.20)	(0.04,0.08,0.20)	(0.00,0.02,0.13)	(0.04,0.07,0.17)	(0.01,0.04,0.16)	(0.00,0.02,0.12)	(0.00,0.03,0.14)	(0.01,0.06,0.20)
H14	(0.00,0.02,0.12)	(0.02,0.05,0.15)	(0.00,0.04,0.14)	(0.02,0.04,0.13)	(0.00,0.01,0.09)	(0.00,0.02,0.11)	(0.00,0.01,0.09)	(0.00,0.01,0.09)	(0.04,0.07,0.19)
H15	(0.01,0.05,0.18)	(0.02,0.07,0.20)	(0.01,0.05,0.18)	(0.04,0.07,0.19)	(0.04,0.07,0.18)	(0.01,0.03,0.16)	(0.05,0.09,0.18)	(0.02,0.05,0.17)	(0.04,0.09,0.25)
H16	(0.00,0.04,0.16)	(0.02,0.06,0.18)	(0.02,0.06,0.17)	(0.00,0.02,0.12)	(0.00,0.03,0.13)	(0.02,0.06,0.17)	(0.00,0.02,0.11)	(0.04,0.07,0.16)	(0.06,0.10,0.22)
H17	(0.00,0.04,0.14)	(0.00,0.04,0.15)	(0.00,0.04,0.14)	(0.00,0.01,0.11)	(0.00,0.03,0.12)	(0.00,0.04,0.14)	(0.02,0.05,0.14)	(0.00,0.01,0.10)	(0.02,0.06,0.18)
H18	(0.02,0.06,0.18)	(0.04,0.07,0.19)	(0.02,0.06,0.18)	(0.00,0.03,0.14)	(0.00,0.01,0.11)	(0.02,0.06,0.17)	(0.04,0.06,0.16)	(0.00,0.03,0.13)	(0.01,0.04,0.17)

Fig. 2. Resulting total influence fuzzy matrix T.


Results

Causal Diagram | Influential Relation Map IRM | Direct-influence matrix Z | Normalized direct-influence matrix | Total-influence matrix T | Fuzzy IRM

	R	C	R+C	R-C
H1	(0.20,0.64,2.46)	(0.18,0.68,2.54)	(0.02,-0.04,-0.08)	(0.38,1.31,5.00)
H2	(0.26,0.81,2.75)	(0.35,0.99,3.14)	(-0.09,-0.18,-0.39)	(0.62,1.80,5.89)
H3	(0.25,0.79,2.78)	(0.62,1.35,3.70)	(-0.37,-0.56,-0.92)	(0.87,2.15,6.49)
H4	(0.44,1.11,3.31)	(0.51,1.24,3.58)	(-0.07,-0.13,-0.26)	(0.94,2.35,6.89)
H5	(0.35,0.96,3.07)	(0.17,0.70,2.53)	(0.18,0.27,0.54)	(0.52,1.66,5.60)
H6	(0.17,0.68,2.48)	(0.18,0.71,2.61)	(-0.01,-0.03,-0.13)	(0.35,1.38,5.09)
H7	(0.22,0.77,2.74)	(0.36,0.90,2.85)	(-0.14,-0.13,-0.11)	(0.58,1.67,5.59)
H8	(0.10,0.47,2.11)	(0.06,0.28,1.72)	(0.04,0.19,0.39)	(0.16,0.74,3.83)
H9	(0.12,0.48,2.15)	(0.15,0.59,2.30)	(-0.03,-0.11,-0.15)	(0.27,1.07,4.44)
H10	(0.26,0.82,2.80)	(0.29,0.88,2.95)	(-0.03,-0.05,-0.15)	(0.55,1.70,5.75)
H11	(0.35,0.94,2.96)	(0.35,0.95,3.00)	(0.01,-0.01,-0.04)	(0.70,1.90,5.96)
H12	(0.37,0.90,2.90)	(0.31,0.94,2.95)	(0.06,-0.04,-0.06)	(0.69,1.85,5.85)
H13	(0.39,0.99,3.07)	(0.18,0.61,2.41)	(0.21,0.38,0.66)	(0.57,1.60,5.48)
H14	(0.16,0.54,2.27)	(0.12,0.54,2.28)	(0.04,-0.00,-0.01)	(0.28,1.09,4.54)
H15	(0.46,1.14,3.38)	(0.24,0.78,2.75)	(0.22,0.36,0.62)	(0.70,1.92,6.13)
H16	(0.31,0.88,2.83)	(0.18,0.56,2.25)	(0.13,0.31,0.58)	(0.48,1.44,5.08)
H17	(0.20,0.69,2.52)	(0.13,0.56,2.30)	(0.07,0.14,0.22)	(0.33,1.25,4.82)
H18	(0.29,0.86,2.85)	(0.53,1.22,3.56)	(-0.24,-0.36,-0.72)	(0.82,2.07,6.41)

Fig. 3. Resulting influential relation map (IRM).

influential ones, and export results in Excel format for further analysis. Additionally, PyDEMATEL is available as a Python package, complete with documentation of its main methods provided in Appendix A. This makes it convenient for developers to integrate the tool into their decision-making solution projects. The features of PyDEMATEL have been tested and validated using data from an example in the maritime industry, with results consistent with those found in the referenced study. Furthermore, PyDEMATEL can be further improved by incorporating additional variants of the DEMATEL method, thus extending its applicability to a wider range of decision-making scenarios. We also aim to refine the graphical interface, based on feedback from future users and developers. These improvements will enable PyDEMATEL to continue to evolve as a valuable resource for solving complex MCDM problems.

Table 2
Linguistic evaluation matrix of marine experts' consensus.

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18
H1	No	No	H	L	VL	VL	No	No	VL	H	L	VL	No	No	L	No	VL	H
H2	H	No	H	L	No	L	VL	VL	No	VL	VH	VL	No	VL	L	VL	VL	H
H3	VL	L	No	H	L	VL	L	No	VL	L	No	VL	L	VL	H	No	L	L
H4	H	L	VL	No	L	H	VH	VL	L	H	L	VH	H	VL	H	VL	No	VL
H5	VL	VL	H	L	No	H	H	No	VL	VL	VH	H	VL	L	VL	L	No	H
H6	VL	L	L	VL	VL	No	L	No	VL	No	VL	VL	L	L	No	VL	VL	VH
H7	No	VL	L	H	L	L	No	VL	L	VL	L	VL	L	VL	No	VL	L	L
H8	No	VL	VL	VL	L	No	No	No	H	H	No	VL	VL	No	VL	VL	No	No
H9	No	VL	L	VL	L	No	No	H	No	L	VL	No	No	VL	VL	No	L	No
H10	VL	L	No	H	VL	VL	VL	L	VH	No	VL	H	VL	No	VL	L	VL	H
H11	VL	L	H	VH	VL	L	H	No	VL	H	No	VH	VL	No	VL	No	VL	H
H12	L	H	VH	H	No	VL	VH	No	VL	H	VL	No	No	VL	H	No	No	H
H13	H	L	VH	L	VH	VL	H	No	No	VL	H	H	No	H	VL	No	VL	VL
H14	VL	VL	H	L	VL	VL	No	No	No	No	L	VL	L	No	No	No	No	H
H15	L	L	H	VH	VL	L	H	VL	No	VL	L	VL	H	H	No	VH	L	H
H16	VL	VH	H	L	VL	VL	VL	No	No	VL	L	L	No	VL	L	No	H	VH
H17	No	H	VH	H	VL	No	VL	No	VL	VL	VL	VL	No	VL	VL	L	No	L
H18	VL	H	VH	L	VL	VL	No	No	VL	L	H	L	VL	No	L	H	VL	No

CRediT authorship contribution statement

Abderrahman Chekry: Writing – original draft, Data curation, Conceptualization. **Jamal Bakkas:** Writing – original draft, Formal analysis, Conceptualization. **Mohamed Hanine:** Visualization, Methodology, Formal analysis. **Elizabeth Caro Montero:** Investigation, Funding acquisition, Data curation. **Mirtha Silvana Garat de Marin:** Visualization, Software, Project administration. **Imran Ashraf:** Writing – review & editing, Validation, Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This study was funded by the European University of Atlantic.

Appendix A. Description of principal PyDEMATEL methods

This appendix provides concise overviews of the PyDEMATEL package, comprising two primary classes: DEMATELSolver and FuzzyDEMATELSolver, each containing over 25 methods. The methods share identical names but operate differently: DEMATELSolver works with crisp values, while FuzzyDEMATELSolver operates with triangular fuzzy numbers. Here, we spotlight the most significant methods.

to install the package PyDEMATEL : `pip install pyDEMATEL`

to use DEMATELSolver : `from pyDEMATEL import DEMATELSolver`

to use FuzzyDEMATELSolver : `from pyDEMATEL import FuzzyDEMATELSolver`

to use DEMATELWindow : `from pyDEMATEL import DEMATELWindow`

`setNumberOfExperts(nb)`: change the number of experts

`getNumberOfExperts()`: returns the number of experts

`setNumberOfFactors(nb)`: change the number of factors

`getNumberOfFactors()`: returns the number of factors

`getMatrix()`: returns an array composed of the set of the expert's individual direct influence matrices.

`setMatrix(matrices)`: changes the array of individual expert direct influence matrices. "*matrices*" is an array of a set of expert's individual direct influence matrix

`AddMatrix(matrix)`: adds one matrix to the array of individual expert direct influence matrices.

`setExperts(experts)`: changes the list of experts. "*experts*" is a list of string

`getExperts()`: returns a list of experts

`addExpert(expert)`: adds one expert to the list of experts. *expert* is a string

`setFactors(factors)`: changes the list of factors. "*factors*" is a list of string

`getFactors()`: returns a list of factors

`addFactor(factor)`: adds one factor to the list of factors. *factor* is a string

`step1()`: computes the direct influence matrix

`step2()`: computes the normalized direct-influence matrix

`step3()`: computes the total influence matrix

`step4()`: computes the influential relation map

`drawCurve()`: draws the causal graph **`getDirectInfluenceMatrix()`**: returns the direct-influence matrix

`getNormalizedDirectInfluenceMatrix ()`: returns the normalized direct-influence matrix

`getTotalInfluenceMatrix()`: returns the total influence matrix

`getProminence()`: returns the prominence R+C

`getRalation()`: returns the relation R-C

`savexl(url)`: exports intermediary matrices and results in Excel format

References

[1] Gabus A, Fontela E. World problems, an invitation to further thought within the framework of DEMATEL, 1, (8):Geneva, Switzerland: Battelle Geneva Research Center; 1972, p. 12–4.

[2] Hanine M, Boutkhoum O, El Barakaz F, Lachgar M, Assad N, Rustam F, et al. An intuitionistic fuzzy approach for smart city development evaluation for developing countries: Moroccan context. Mathematics 2021;9(21):2668.

[3] Li R-J. Fuzzy method in group decision making. Comput Math Appl 1999;38(1):91–101.

[4] Li Y, Mathiyazhagan K. Application of DEMATEL approach to identify the influential indicators towards sustainable supply chain adoption in the auto components manufacturing sector. J Clean Prod 2018;172:2931–41.

[5] Gonzales G, Costan F, Suladay D, Gonzales R, Enriquez L, Costan E, et al. Fermatean fuzzy DEMATEL and MMDE algorithm for modelling the barriers of implementing education 4.0: insights from the Philippines. Appl Sci 2022;12(2):689.

[6] Kuzu AC. Application of fuzzy DEMATEL approach in maritime transportation: A risk analysis of anchor loss. Ocean Eng 2023;273:113786.

[7] Yan H, Yang Y, Lei X, Ye Q, Huang W, Gao C. Regret theory and fuzzy-DEMATEL-based model for construction program manager selection in China. Buildings 2023;13(4):838.
[8] Hosseini SM, Paydar MM, Hajiaghaei-Keshteli M. Recovery solutions for eco-tourism centers during the Covid-19 pandemic: Utilizing Fuzzy DEMATEL and Fuzzy VIKOR methods. Expert Syst Appl 2021;185:115594.
[9] Li H, Dietl H, Li J. Identifying key factors influencing sustainable element in healthcare waste management using the interval-valued fuzzy DEMATEL method. J Mater Cycles Waste Manag 2021;23:1777–90.
[10] Muhammad MN, Cavus N. Fuzzy DEMATEL method for identifying LMS evaluation criteria. Procedia Comput Sci 2017;120:742–9.
[11] Mardani A, Jusoh A, Nor K, Khalifah Z, Zakwan N, Valipour A. Multiple criteria decision-making techniques and their applications–a review of the literature from 2000 to 2014. Ekonomska istraživanja (Econ Res) 2015;28(1):516–71.
[12] Yadav V, Karmakar S, Kalbar PP, Dikshit AK. PyTOPS: A python based tool for TOPSIS. SoftwareX 2019;9:217–22.
[13] Ataei Y, Mahmoudi A, Feylizadeh MR, Li D-F. Ordinal priority approach (OPA) in multiple attribute decision-making. Appl Soft Comput 2020;86:105893.
[14] Bendarag A, Bakkas J, Hanine M, Boutkhoum O. PyOPAsolver: a Python based tool for ordinal priority approach operations and normalization. SoftwareX 2022;20:101226.
[15] Pereira V, Basilio MP, Santos CHTSHT. Enhancing decision analysis with a large language model: pyDecision a comprehensive library of MCDA methods in Python. 2024, arXiv preprint arXiv:2404.06370.
[16] Satman MH, Yıldırım BF, Kuruca E. JMcDM: A julia package for multiple-criteria decision-making tools. J Open Source Softw 2021;6(65):3430. http://dx.doi.org/10.21105/joss.03430.
[17] Gul M. Emergency department ergonomic design evaluation: A case study using fuzzy DEMATEL-focused two-stage methodology. Health Policy Technol 2019;8(4):365–76.
[18] Akyuz E, Celik E. A fuzzy DEMATEL method to evaluate critical operational hazards during gas freeing process in crude oil tankers. J Loss Prev Process Ind 2015;38:243–53.
[19] Jeong JS, Ramírez-Gómez Á. Development of a web graphic model with fuzzy-decision-making Trial and Evaluation Laboratory/Multi-criteria-Spatial Decision Support System (F-DEMATEL/MC-SDSS) for sustainable planning and construction of rural housings. J Clean Prod 2018;199:584–92.
[20] Bendarag A, Boutkhoum O, Abada D, Hanine M. Blockchain adoption barriers in Moroccan sustainable supply chain: a proposed approach. Indonesian J Electr Eng Comput Sci 2022;27(2):892–9.