Distance Metrics Library for MCDA Methods - Supplementary Material

,

Abstract

This Supplementary Material provides background and formulas for the methods employed in this research and implemented in Python library called distance-metrics-mcda.

Keywords: Distance metrics, MCDA, TOPSIS, Decision support systems.

1. Introduction

The Supplementary Material provide fundamentals, basic assumptions, and detailed step-by-step descriptions of the methods implemented in the Python 3 library ‘distance-metrics-mcda‘. This library is available in the Python Package Index (PyPI) repository of software for the Python programming language [3] and on GitHub [4]. The contents of these materials include:

1. MCDA methods implemented in module ‘mcda_methods‘: TOPSIS
2. Correlation coefficients implemented in module ‘correlations‘
3. Objective weighting methods implemented in module ‘weighting_methods‘
4. Distance metrics implemented in module ‘distance_metrics‘
5. Normalization methods implemented in module ‘normalizations‘

2. The TOPSIS method

The following stages of the TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) method are provided below, based on [2].

Step 1. Normalization of the decision matrix represented by Equation (1).

$$X = [x_{ij}]_{m \times n} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{1}$$

where m denotes the number of alternatives, and n represents the number of criteria.

The Minimum-Maximum normalization method or another normalization method can be used for performing the normalization procedure. In Minimum-Maximum normalization r_{ij} normalized values are obtained applying Equation (2) for profit criteria and (3) for cost criteria.

$$r_{ij} = \frac{x_{ij} - min_j(x_{ij})}{max_j(x_{ij}) - min_j(x_{ij})} \tag{2}$$

$$r_{ij} = \frac{max_j(x_{ij}) - x_{ij}}{max_j(x_{ij}) - min_j(x_{ij})} \tag{3}$$

Step 2. Calculation of weighted normalized decision matrix using Equation (4).

$$v_{ij} = w_j r_{ij} \tag{4}$$

Step 3. Determination of Positive Ideal Solution with Equation (5) and Negative Ideal Solution with Equation (6). PIS contains the maximums of the weighted normalized decision matrix, while NIS contains its minimums. Due to the previous normalization application, there is no need to divide the criteria into profit and cost in this step.

$$v_j^+ = \{v_1^+, v_2^+, \ldots, v_n^+\} = \{max_j(v_{ij})\} \tag{5}$$

$$v_j^- = \{v_1^-, v_2^-, \ldots, v_n^-\} = \{min_j(v_{ij})\} \tag{6}$$

Step 4. Computation of distance from PIS (7) and NIS (8) for each alternative. The default metric for distance computation in TOPSIS algorithm is Euclidean distance.

$$D_i^+ = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^+)^2} \tag{7}$$

$$D_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^-)^2} \tag{8}$$

Step 5. Computation of the score for each considered alternative according to Equation (9). The C_i value is always between 0 to 1, and the alternative that has the highest C_i value is the best. It implies that for TOPSIS, the ranking is created by descending sorting of alternatives by preference value.

$$C_i = \frac{D_i^-}{D_i^- + D_i^+} \tag{9}$$

3. Correlation coefficients

3.1. Spearman Rank Correlation Coefficient - r_s

The Spearman Rank Correlation Coefficient is calculated to compare two rankings x and y as Equation (10) demonstrates

$$r_s = 1 - \frac{6 \cdot \sum_{i=1}^{N} (x_i - y_i)^2}{N \cdot (N^2 - 1)} \tag{10}$$

where N means size of vector x and y [7].

3.2. Weighted Spearman's Rank Correlation Coefficient - r_w

The r_w correlation coefficient is calculated in aim to compare two rankings x and y as Equation (11) demonstrates. N represents a number of rank values x_i and y_i.

$$r_w = 1 - \frac{6 \sum_{i=1}^{N} (x_i - y_i)^2 ((N - x_i + 1) + (N - y_i + 1))}{N^4 + N^3 - N^2 - N} \tag{11}$$

3.3. Rank Similarity Coefficient - WS

WS ranking similarity coefficient is calculated using Equation (12), where N denotes the size of compared rankings, x_i and y_i mean positions of x_i and y_i in the compared rankings x and y.

$$WS = 1 - \sum_{i=1}^{N} 2^{-x_i} \frac{|x_i - y_i|}{max(|x_i - 1|, |x_i - N|)} \tag{12}$$

3.4. Pearson correlation coefficient

The Pearson correlation coefficient determines the consistency of two vectors, x and y, including numerical values. For MCDA analysis, the Pearson correlation coefficient estimates the similarity between vectors with preference values or positions in rankings of assessed alternatives [18]. Pearson coefficient is computed according to Equation (13)

$$r_{xy} = \frac{\sum_{i=1}^{N}(x_i - \overline{x}) \sum_{i=1}^{N}(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \overline{x})^2} \sqrt{\sum_{i=1}^{N}(y_i - \overline{y})^2}} \tag{13}$$

where N means the size of compared vectors, $\overline{x} = \frac{1}{n}\sum_{i=1}^{N} x_i$ denotes the mean value of x and $\overline{y} = \frac{1}{n}\sum_{i=1}^{N} y_i$ express the mean value of y.

4. Objective weighting methods

4.1. Entropy weighting method

[9]

Step 1. Normalize decision matrix using sum normalization method to get normalized decision matrix $P = [p_{ij}]_{m \times n}$ where $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$, m denotes alternatives number and n represents criteria number.

Step 2. Calculate the entropy E_j for each jth criterion according to Equation (14).

$$E_j = -\frac{\sum_{i=1}^{m} p_{ij} ln p_{ij}}{ln m} \tag{14}$$

Step 3. Calculate d_j as Equation (15) shows.

$$d_j = 1 - E_j \tag{15}$$

Step 4. Calculate the entropy weights for each jth criterion.

$$w_j = \frac{d_j}{\sum_{j=1}^{n} d_j} \tag{16}$$

4.2. CRITIC weighting method

Criteria Importance Through Inter-criteria Correlation (CRITIC) is the objective weighting method. To determine criteria weights using CRITIC method, the decision matrix $X = [x_{ij}]_{m \times n}$ is required. This decision matrix contains m alternatives and n criteria, where x_{ij} represents the performance values of i^{th} alternative with respect of j^{th} criterion. To determine the weight of the j^{th} criterion w_j using CRITIC the calculations provided below are conducted [6].

Step 1. Decision matrix must be normalized using Equation (17) for profit criteria and (18) for cost criteria.

$$r_{ij} = \frac{x_{ij} - min_j(x_{ij})}{max_j(x_{ij}) - min_j(x_{ij})} \tag{17}$$

$$r_{ij} = \frac{max_j(x_{ij}) - x_{ij}}{max_j(x_{ij}) - min_j(x_{ij})} \tag{18}$$

Step 2. Calculation of the Pearson correlation coefficient between pairs of criteria as Equation (19) demonstrates.

$$\rho_{jk} = \frac{\sum_{i=1}^{m}(r_{ij} - \overline{r}_j)(r_{ik} - \overline{r}_k)}{\sqrt{\sum_{i=1}^{m}(r_{ij} - \overline{r}_j)^2 \sum_{i=1}^{m}(r_{ik} - \overline{r}_k)^2}}. \tag{19}$$

Step 3. Calculation of criteria weights using Equation (20) and (21),

$$c_j = \sigma_j \sum_{k=1}^{n}(1 - \rho_{jk}); \tag{20}$$

$$w_j = \frac{c_j}{\sum_{k=1}^{n} c_k}, \tag{21}$$

where $i = 1, 2, \ldots, m;\ j, k = 1, 2, \ldots, n$. In the formulas given above c_j represents the quantity of information contained in j^{th} criterion, σ_j express the standard deviation of the j^{th} criterion and ρ_{jk} is the correlation coefficient between the j^{th} and k^{th} criteria.

A high standard deviation and low correlation of given criterion with the others determine a high criterion weight. Thus, a high value of C_j provides more information from the considered criterion [15].

5. Distance metrics

This section gives foundations and formulas for eight distance metrics provided by this software. Sets between which distance is calculated are denoted by a and b, and n means the size of one-dimensional vectors a and b compared with distance metrics.

5.1. Euclidean distance

The Euclidean distance measured between two sets of points is obtained by calculating the square root of the sum of the squares of the differences between the corresponding points in compared sets a and b. Equation (22) is applied for computation of the Euclidean distance

$$d(a, b) = \sqrt{\sum_{j=1}^{n}(a_i - b_i)^2} \tag{22}$$

where n denotes the size of sets [12].

5.2. Manhattan distance

Manhattan distance is also known as the Taxicab distance. It is used for the determination of the distance between two sets a and b by calculating the sum of absolute differences among respective particular points included in compared sets, as Equation (23) shows [12].

$$d(a, b) = \sum_{j=1}^{n} |a_i - b_i| \tag{23}$$

5.3. Hausdorff distance

Hausdorff distance is a nonlinear metric used to measure the mismatch between two compared sets. In contrast to most methods used to distance determination, this metric is not based on measuring the distance between corresponding points. Its algorithm measures proximity rather than exact superposition, so it is more tolerant of perturbations that appear in points' positions.

Besides, this metric is sensitive to extreme values [16]. The Hausdorff distance is computed by Equation (24)

$$H(A, B) = max(h(A, B), h(B, A)) \tag{24}$$

where $h(A, B)$ and $h(B, A)$ represent directed distances between two sets of points $A = \{a_1, \ldots a_{N_a}\}$ and $B = \{b_1, \ldots b_{N_b}\}$. Directed distance is calculated as Equation (25) presents

$$h(A, B) = \underset{a \in A}{max}(\underset{b \in B}{min}(d(a, b))) \tag{25}$$

where distance between a and b is determined as $d(a, b) = \|a - b\|$. Distance between these two points is most commonly calculated applying Euclidean distance, Manhattan distance, and Chebyshev distance, also known as chessboard distance. For each point, a from set A, find the nearest point b from set B and determine the distance. The distance with the maximum value is the final directed distance.

5.4. Correlation distance

The correlation metric considers points as sequences of values. The distance is computed by subtracting the correlation values between points from 1, according to Equation (26) [17]

$$d(a, b) = 1 - \frac{\sum_{i=1}^{n}(a_i - \bar{a}_i)(b_i - \bar{b}_i)}{\sqrt{\sum_{i=1}^{n}(a_i - \bar{a}_i)^2}\sqrt{\sum_{i=1}^{n}(b_i - \bar{b}_i)^2}} \tag{26}$$

where

$$\bar{a} = \frac{1}{n}(\sum_{i=1}^{n} a_i), \tag{27}$$

$$\bar{b} = \frac{1}{n}(\sum_{i=1}^{n} b_i). \tag{28}$$

5.5. Chebyshev distance

The Chebyshev distance between two sets of points is calculated with Equation (29).

$$d(a, b) = \underset{i=1,\ldots,n}{max}\{|a_i - b_i|\} \tag{29}$$

This metric is also known as chessboard distance [14].

5.6. Standardized Euclidean distance

Standardized Euclidean distance determines the distance between two points in a multidimensional space by applying differential contributions of individual point coordinate components and considering correlations between them [19]. This distance is computed according to Equation (30).

$$d(a, b) = \sqrt{\sum_{i=1}^{n}(\frac{a_i - b_i}{\sigma_i})^2} \tag{30}$$

5.7. Cosine distance

Cosine distance considers all points as vectors and is computed by subtracting the cosine of the angle between vectors from 1 like Equation (31) demonstrates [10, 8].

$$d(a,b) = 1 - \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}} \tag{31}$$

5.8. Cosine similarity measure

The Cosine similarity is an angle-based measure for estimation of similarity between two sets of points [13]. The cosine similarity measure (csm) applied for two one-dimensional vectors a and b is calculated with Equation (32).

$$csm(a,b) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i}\sqrt{\sum_{i=1}^{n} b_i}} \tag{32}$$

Distance metrics most commonly used by decision-makers to determine the distance to reference points in MCDA methods or to measure the distance between one-dimensional value vectors are Euclidean, Manhattan, and Chebyshev distance. However, many metrics can be used for the above purposes. Therefore, MCDA analysis can be extended to the distance metrics proposed and detailed in paper [11]. The formulas for calculating the following distance metrics are given below. In each, $n(i = 1, 2, \ldots, n)$ is the number of values in the compared sets a and b.

5.9. Squared Euclidean distance

$$d(a,b) = \sum_{i=1}^{n} (a_i - b_i)^2 \tag{33}$$

5.10. Bray-Curtis distance

$$d(a,b) = \frac{\sum_{i=1}^{n} |a_i - b_i|}{\sum_{i=1}^{n} (a_i + b_i)} \tag{34}$$

5.11. Canberra distance

$$d(a,b) = \sum_{i=1}^{n} \frac{a_i - b_i}{a_i + b_i} \tag{35}$$

5.12. Lorentzian distance

$$d(a,b) = \sum_{i=1}^{n} ln(1 + |a_i - b_i|) \tag{36}$$

5.13. Jaccard distance

$$d(a,b) = \frac{\sum_{i=1}^{n} (a - b)^2}{\sum_{i=1}^{n} a_i^2 + \sum_{i=1}^{n} b_i^2 - \sum_{i=1}^{n} a_i b_i} \tag{37}$$

5.14. Dice distance

$$d(a,b) = \frac{\sum_{i=1}^{n} (a_i - b_i)^2}{\sum_{i=1}^{n} a_i^2 + \sum_{i=1}^{n} b_i^2} \tag{38}$$

5.15. Bhattacharyya distance

$$d(a,b) = -ln(\sum_{i=1}^{n}\sqrt{a_i b_i})^2 \tag{39}$$

5.16. Hellinger distance

$$d(a,b) = 2\sqrt{1 - \sum_{i=1}^{n}\sqrt{a_i b_i}} \tag{40}$$

5.17. Matusita distance

$$d(a,b) = \sqrt{2 - 2\sum_{i=1}^{n}\sqrt{a_i b_i}} \tag{41}$$

5.18. Squared-chord distance

$$d(a,b) = \sum_{i=1}^{n}(\sqrt{a_i} - \sqrt{b_i})^2 \tag{42}$$

5.19. Pearson χ_2 distance

$$d(a,b) = \sum_{i=1}^{n}\frac{(a_i - b_i)^2}{b_i} \tag{43}$$

5.20. Square χ_2 distance

$$d(a,b) = \sum_{i=1}^{n}\frac{(a_i - b_i)^2}{a_i + b_i} \tag{44}$$

6. Normalization methods

Normalization methods implemented in pyrepo-mcda library were described based on [1].

6.1. Minimum-maximum normalization

In the case of minimum-maximum method normalization, the normalized values of r_{ij} are determined for the profit criteria using the Equation (45), while for the cost criteria using the Equation (46). This method is used by default for the MABAC method (Multi-Attributive Border Approximation area Comparison).

$$r_{ij} = \frac{x_{ij} - min_j(x_{ij})}{max_j(x_{ij}) - min_j(x_{ij})} \tag{45}$$

$$r_{ij} = \frac{max_j(x_{ij}) - x_{ij}}{max_j(x_{ij}) - min_j(x_{ij})} \tag{46}$$

This procedure has the advantage that the measurement scale is precisely between 0 and 1 for each attribute. Therefore, this procedure is also appropriate for data containing negative or zero values.

6.2. Maximum normalization

The maximum method is a technique in which only the most enormous value is used for profit (47) and cost criteria (48).

$$r_{ij} = \frac{x_{ij}}{max_j(x_{ij})} \tag{47}$$

$$r_{ij} = 1 - \frac{x_{ij}}{max_j(x_{ij})} \tag{48}$$

6.3. Sum normalization

In the sum method, all values in the estimated set are summed. This normalization method is applied by default for Complex Proportional Assessment (COPRAS) and Additive Ratio Assessment (ARAS) method [5]. Equations used for profit (49) and cost criteria (50) are given below.

$$r_{ij} = \frac{x_{ij}}{\sum_{i=1}^{m} x_{ij}} \tag{49}$$

$$r_{ij} = \frac{\frac{1}{x_{ij}}}{\sum_{i=1}^{m} \frac{1}{x_{ij}}} \tag{50}$$

6.4. Linear normalization

Linear normalization is performed for profit criteria as Equation (51) shows and for cost criteria according to Equation (52). This is default normalization method for CODAS and WASPAS.

$$r_{ij} = \frac{x_{ij}}{max_j(x_{ij})} \tag{51}$$

$$r_{ij} = \frac{min_j(x_{ij})}{x_{ij}} \tag{52}$$

6.5. Vector normalization

In the vector method, the square root of all values is calculated. Formulas used for profit (53) and cost criteria (54) are presented below. This normalization technique is the default for TOPSIS, MOORA, and MULTIMOORA methods.

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \tag{53}$$

$$r_{ij} = 1 - \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \tag{54}$$

References

1. Aytekin, A.: Comparative analysis of the normalization techniques in the context of MCDM problems. Decision Making: Applications in Management and Engineering 4(2), pp. 1–25 (2021)
2. Bera, B., Shit, P.K., Sengupta, N., Saha, S., Bhattacharjee, S.: Susceptibility of deforestation hotspots in Terai-Dooars belt of Himalayan Foothills: A comparative analysis of VIKOR and TOPSIS models. Journal of King Saud University-Computer and Information Sciences (2021)

3. energyinpython: Python 3 library for Multi-Criteria Decision Analysis based on Distance Metrics (2022), `https://pypi.org/project/distance-metrics-mcda/`
4. energyinpython: Python 3 library for Multi-Criteria Decision Analysis based on Distance Metrics (2022), `https://github.com/energyinpython/distance-metrics-for-mcda`
5. Goswami, S., Mitra, S.: Selecting the best mobile model by applying AHP-COPRAS and AHP-ARAS decision making methodology. International Journal of Data and Network Science 4(1), pp. 27–42 (2020)
6. Jahan, A., Mustapha, F., Sapuan, S., Ismail, M.Y., Bahraminasab, M.: A framework for weighting of criteria in ranking stage of material selection process. The International Journal of Advanced Manufacturing Technology 58(1-4), pp. 411–420 (2012)
7. Kumar, A., Abirami, S.: Aspect-based opinion ranking framework for product reviews using a Spearman's rank correlation coefficient method. Information Sciences 460, pp. 23–41 (2018)
8. Liu, D., Chen, X., Peng, D.: Some cosine similarity measures and distance measures between q-rung orthopair fuzzy sets. International Journal of Intelligent Systems 34(7), pp. 1572–1587 (2019)
9. Lotfi, F.H., Fallahnejad, R.: Imprecise Shannon's entropy and multi attribute decision making. Entropy 12(1), pp. 53–62 (2010)
10. Pan, C., Huang, J., Hao, J., Gong, J.: Towards zero-shot learning generalization via a cosine distance loss. Neurocomputing 381, pp. 167–176 (2020)
11. Ploskas, N., Papathanasiou, J.: A decision support system for multiple criteria alternative ranking using TOPSIS and VIKOR in fuzzy and nonfuzzy environments. Fuzzy Sets and Systems 377, pp. 1–30 (2019)
12. Sharma, S.K., Kumar, S.: Comparative analysis of Manhattan and Euclidean distance metrics using A* algorithm. J. Res. Eng. Appl. Sci 1(4), pp. 196–198 (2016)
13. Stanujkić, D., Karabašević, D., Popović, G., Zavadskas, E.K., Saračević, M., Stanimirović, P.S., Ulutaş, A., Katsikis, V.N., Meidute-Kavaliauskiene, I.: Comparative Analysis of the Simple WISP and Some Prominent MCDM Methods: A Python Approach. Axioms 10(4), pp. 347 (2021)
14. Sun, Y., Li, S., Wang, X.: Bearing fault diagnosis based on EMD and improved Chebyshev distance in SDP image. Measurement 176, pp. 109100 (2021)
15. Tuş, A., Adalı, E.A.: The new combination with CRITIC and WASPAS methods for the time and attendance software selection problem. Opsearch 56(2), pp. 528–538 (2019)
16. Zhang, J., Pang, J., Yu, J., Wang, P.: An efficient assembly retrieval method based on Hausdorff distance. Robotics and Computer-Integrated Manufacturing 51, pp. 103–111 (2018)
17. Zhong, W., Zhu, L.: An iterative approach to distance correlation-based sure independence screening. Journal of Statistical Computation and Simulation 85(11), pp. 2331–2345 (2015)
18. Zhou, H., Deng, Z., Xia, Y., Fu, M.: A new sampling method in particle filter based on Pearson correlation coefficient. Neurocomputing 216, pp. 208–215 (2016)
19. Zou, Y., Chen, W., Tong, M., Tao, S.: DEA Cross-Efficiency Aggregation with Deviation Degree Based on Standardized Euclidean Distance. Mathematical Problems in Engineering 2021 (2021)