

Determine OWA operator weights using kernel density estimation

Mingwei Lin, Wenshu Xu, Zhanpeng Lin & Riqing Chen

To cite this article: Mingwei Lin, Wenshu Xu, Zhanpeng Lin & Riqing Chen (2020) Determine OWA operator weights using kernel density estimation, Economic Research-Ekonomiska Istraživanja, 33:1, 1441-1464, DOI: [10.1080/1331677X.2020.1748509](https://doi.org/10.1080/1331677X.2020.1748509)

To link to this article: <https://doi.org/10.1080/1331677X.2020.1748509>



© 2020 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 02 May 2020.



Submit your article to this journal [↗](#)



Article views: 477



View related articles [↗](#)



View Crossmark data [↗](#)

Determine OWA operator weights using kernel density estimation

Mingwei Lin^{a,b} , Wenshu Xu^{a,c}, Zhanpeng Lin^{a,c} and Riqing Chen^b

^aCollege of Mathematics and Informatics, Fujian Normal University, Fuzhou, Fujian, China; ^bDigital Fujian Institute of Big Data for Agriculture and Forestry, Fujian Agriculture and Forestry University, Fuzhou, China; ^cDigital Fujian Internet-of-Things Laboratory of Environmental Monitoring, Fujian Normal University, Fuzhou, Fujian, China

ABSTRACT

Some subjective methods should divide input values into local clusters before determining the ordered weighted averaging (OWA) operator weights based on the data distribution characteristics of input values. However, the process of clustering input values is complex. In this paper, a novel probability density based OWA (PDOWA) operator is put forward based on the data distribution characteristics of input values. To capture the local cluster structures of input values, the kernel density estimation (KDE) is used to estimate the probability density function (PDF), which fits to the input values. The derived PDF contains the density information of input values, which reflects the importance of input values. Therefore, the input values with high probability densities (PDs) should be assigned with large weights, while the ones with low PDs should be assigned with small weights. Afterwards, the desirable properties of the proposed PDOWA operator are investigated. Finally, the proposed PDOWA operator is applied to handle the multicriteria decision making problem concerning the evaluation of smart phones and it is compared with some existing OWA operators. The comparative analysis shows that the proposed PDOWA operator is simpler and more efficient than the existing OWA operators.

ARTICLE HISTORY

Received 4 November 2019
Accepted 24 March 2020

KEYWORDS

Aggregation operator; ordered weighted averaging; multi-criteria decision making (MCDM)

JEL CLASSIFICATIONS

C43; C61; D81

1. Introduction

During multi-criteria decision making (MCDM) processes (Garg, 2018a; 2018b; 2019; Lin et al., 2018; Lin et al., 2018), various decision methods can be chosen to handle the criteria evaluation information of alternatives for ranking them (He et al., 2019a, 2019b; Lin et al., 2019; Lin et al., 2018; Liu et al., 2019;; Wei et al., 2019; Zeng et al., 2019). The aggregation operators are considered as a simple yet efficient MCDM method (Garg, 2018c; Garg & Kaur, 2020), which aggregate the criteria evaluation information of alternatives into the overall criteria values (Lin et al., 2020; Lin et al.,

CONTACT Riqing Chen  linmwcs@163.com

© 2020 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

2018; Roy et al., 2019; Tang et al., 2019). According to the overall criteria values of alternatives, all the alternatives can be ranked and then the optimal one can be obtained (Mi et al., 2019). To improve the ranking results of MCDM problems, a variety of aggregation operators have been proposed (Kang et al., 2018; Liu et al., 2020; Riaz & Tehrim, 2019; Wei et al., 2019), such as weighted averaging operator, weighted geometric operator, and ordered weighted averaging (OWA) operator (Yager, 1988; 2019). The OWA operator was proposed by Yager (1988). It is a parameterized class of the mean type aggregation operators. If the OWA operator weights are determined, then special aggregation operators will be obtained, such as max, arithmetic average, and min operators (Amarante, 2018; Gong et al., 2019; Jin et al., 2019a; Merigó & Yager, 2019; Yager, 2019; Yi & Li, 2019). Since it appeared, it has attracted much attention from a large number of well-known research scholars (Beliakov et al., 2018; Leite & Škrjanc, 2019; Li et al., 2020; Mesiar et al., 2018). Due to its practicality, it has been widely applied in various fields such as machine learning (Maldonado et al., 2018), EEG signal improvement (Pander, 2019), time series data fusion (Liu & Xiao, 2019), and risk assessment (Lin et al., 2020; Ma & Cong, 2019).

The implementation process of the OWA operator is composed of three steps: (1) The input values are rearranged in the descending order. (2) The weights of the rearranged input values are determined using an efficient method. (3) Finally, according to the derived weights, these rearranged input values are aggregated into a single value (Yager, 1988). How to determine the weights of rearranged input values is important for the OWA operator since using different weight information to aggregate rearranged input values may derive different ranking results (Casanovas et al., 2020; Jin et al., 2019b; Liu et al., 2019). How to derive the weights of the OWA operator has become a hot research topic in the decision making analysis field and a large quantity of methods have been put forward, such as constraint optimization models (Filev & Yager, 1995; Fuller & Majlender, 2001), quantifier functions (Yager, 1999; Yager & Filev, 1994), and distribution assumption (Lenormand, 2018; Sadiq & Tesfamariam, 2007; Sha et al., 2019; Xu, 2005). For these methods, the weights of the OWA operator are determined in an objective way. They do not consider the complex data distribution characteristics of input values. For the distribution assumption methods, it is assumed that the distribution of the OWA operator weights follows various types of commonly used probability density functions. Although they own solid theoretical foundations and have desirable properties, the ideal hypothesis is unrealistic since the real OWA operator weights usually cannot fit the probability density functions in the practical situations.

Based on the data distribution characteristics of input values, several methods (Boongoen & Shen, 2008; Li et al., 2016; Xu, 2006; Yager, 1993) have been devised. The argument-dependent method (Xu, 2006; Yager, 1993) assigns the input values far away from the average value with small weights and the input values close to the average value with large weights. It treats all the input values as only one cluster, whose average value is solely used to derive the OWA operator weights. However, in the practical cases, the input values show complex data distribution characteristics, where there may exist two or more local clusters. To identify these local clusters, an agglomerative hierarchical clustering method was modified by Boongoen and Shen

(2008) to partition the input values into multiple local clusters. After that, a clus-DOWA operator was proposed to derive the weights of input values based on their distances to their nearest local clusters. Using the concept of majority clusters, a majority clusters DOWA (MC-DOWA) operator was proposed by Li et al. (2016). It computes the weights based on the ratio of the number of input values in the local clusters to the number of all the input values. The clus-DOWA and MC-DOWA operators are capable of generating relatively reasonable ranking results by considering the complex data distribution characteristic of the input values, but they have high time complexity since classification methods should be first adopted to partition input values into some local clusters before calculating the weights. Moreover, for the MC-DOWA operator, the weights of different input values within the same local clusters are assigned with equal weights. It does not make sense.

To overcome these drawbacks, a novel probability density method is proposed to derive the weights of the OWA operator. A powerful mathematical tool, called the kernel density estimation (KDE), is introduced to estimate the probability density function (PDF) for fitting all the input values. The KDE can effectively capture the complex data distribution characteristic of input values without using complicated classification methods. According to the derived PDF, the input values with high probability densities are assigned with large weights, while the input values with low probability densities are assigned with small weights. Based on the probability density method, a novel probability density based OWA (PDOWA) operator is proposed in this paper and its desirable properties are investigated. An application is also developed to implement the processes of reordering, weighting, and aggregating input values of the proposed PDOWA operator. Finally, a practical MCDM example of evaluating smart phones is provided to show the application of the proposed PDOWA operator in the MCDM problem and then the comparative analysis is also given.

The rest of this paper is organized as follows: Some basic knowledge of the OWA operator is provided in Section 2. Section 3 proposes a novel probability density method to determine the OWA weights. Section 4 presents a novel probability density based OWA (PDOWA) operator and develops a smart application for the proposed PDOWA operator. In Section 5, an illustrative MCDM example concerning the evaluation of smart phones is provided and then the comparative analysis is also given. Finally, the valuable conclusions are drawn in Section 6.

2. Preliminaries

An OWA operator (Yager, 1988) is actually a mapping function $OWA : R^n \rightarrow R$, which is associated with a weight vector $W = [w_1, w_2, \dots, w_n]$ satisfying $w_j \in [0, 1]$ and $\sum_{j=1}^n w_j \in [0, 1]$.

$$OWA(I_1, I_2, \dots, I_n) = \sum_{j=1}^n w_j v_j$$

where I_1, I_2, \dots, I_n are the input values to be aggregated and v_j denotes the j th largest one among the input values I_1, I_2, \dots, I_n .

By assigning special values to the OWA operator weights, some special operators can be derived from the OWA operator as follows (Yager, 1988):

(1) When $W = [1, 0, \dots, 0]$, then the OWA operator reduces to the max operator and

$$OWA(I_1, I_2, \dots, I_n) = \max(I_1, I_2, \dots, I_n);$$

(2) When $W = [0, 0, \dots, 1]$, then the OWA operator reduces to the min operator and

$$OWA(I_1, I_2, \dots, I_n) = \min(I_1, I_2, \dots, I_n);$$

(3) When $W = [\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}]$, then the OWA operator reduces to the arithmetic averaging operator and

$$OWA(I_1, I_2, \dots, I_n) = \frac{1}{n} \sum_{i=1}^n I_i.$$

If an aggregation operator is considered as the OWA operator, then it should have the following four properties (Yager, 1988):

(1) Boundedness: The aggregated value from an OWA operator should fall in between the minimum value and maximum value among the input values, namely

$$\min(I_1, I_2, \dots, I_n) \leq OWA(I_1, I_2, \dots, I_n) \leq \max(I_1, I_2, \dots, I_n)$$

(2) Monotonicity: If $B_i \geq S_i$ for $\forall i$, then we have

$$OWA(B_1, B_2, \dots, B_n) \geq OWA(S_1, S_2, \dots, S_n)$$

(3) Commutativity: If $\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n$ is any permutation of I_1, I_2, \dots, I_n , then we have

$$OWA(I_1, I_2, \dots, I_n) = OWA(\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n)$$

(4) Idempotency: If $I_1 = I_2 = \dots = I_n = I$, then we have

$$OWA(I_1, I_2, \dots, I_n) = I$$

Two important measures have been defined by Yager (1988) to characterize the OWA operators, which are the orness and dispersion. The former measure, also called attitudinal character, is defined as

$$O(W) = \frac{1}{n-1} \sum_{j=1}^n (n-j)w_j \tag{1}$$

where $O(W)$ denotes the orness of the OWA operator.

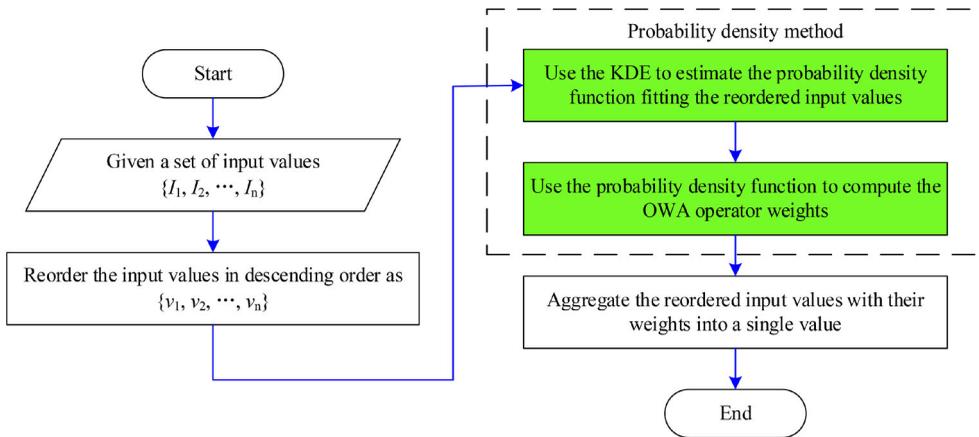


Figure 1. The flow chart of the probability density based OWA operator. *Source:* The authors' data.

Yager (1988) gave the orness measures of max, min, and arithmetic averaging operators as

$$O(W_{\max}) = 1, O(W_{\min}) = 0, O(W_{aa}) = 0.5$$

where W_{\max} , W_{\min} , W_{aa} denote the weight vectors of max, min, and arithmetic averaging operators.

It can be noted that the value of orness measure of the OWA operator falls in the unit interval $[0,1]$ and the orness measure of the OWA operator actually characterizes its similarity degree to the max operator.

The latter measure, also called entropy, is defined as

$$E(W) = - \sum_{j=1}^n w_j \ln(w_j) \quad (2)$$

where $E(W)$ denotes the dispersion of the OWA operator. It characterizes how uniformly the input values are being used.

3. Probability density method

As demonstrated in Figure 1, the implementation process of the proposed probability density based OWA (PDOWA) operator works as follows:

1. A given set of input values $\{I_1, I_2, \dots, I_n\}$ are reordered in descending order as $\{v_1, v_2, \dots, v_n\}$;
2. The mathematical tool of kernel density estimation (KDE) is used to estimate a probability density function (PDF), which can describe how the input values are distributed;
3. The estimated probability density function is used to compute the OWA operator weights;

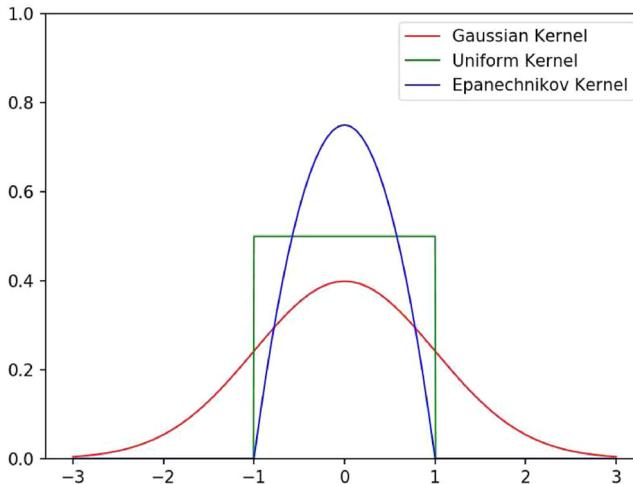


Figure 2. The graphical representation of some classical kernel functions. *Source:* The authors' data.

4. The reordered input values associated with their weights are aggregated into a single value.

The key to the PDOWA operator is how to identify the data distribution characteristics of input values. In the OWA operator studied by Xu (2005), the input values are supposed to be independent and identically distributed and follow the normal distribution. Nevertheless, in the real situations, the input values are often distributed irregularly and they cannot follow the ideal normal distribution. Therefore, it is unreasonable to assume that the input values follow the normal distribution. To avoid the unreasonable assumption, a novel probability density method is proposed, which uses the kernel density estimation to estimate the probability density function that describes how the input values are distributed. As a nonparametric method, the kernel density estimation can estimate the probability density function of the given input values without the priori knowledge about the data distribution characteristic. Based on the kernel density estimation, the probability density function of the reordered input values $\{v_1, v_2, \dots, v_n\}$ can be estimated as

$$\hat{p}(v) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{v-v_i}{h}\right) \quad (3)$$

where $\hat{p}(v)$ denotes the estimated probability density function of the random variable v , n denotes the number of input values, h is a smoothing parameter, also called bandwidth, $K(\cdot)$ is a kernel.

The kernel $K(\cdot)$ has the following three features: (1) $K(x)$ is symmetric; (2) $\int_{-\infty}^{\infty} K(x)dx = 1$; and (3) $K(x) \geq 0$ for all x . As depicted in Figure 2, there are some classical kernel functions: normal (Gaussian), uniform, and Epanechnikov. Because of its desirable mathematical properties, the Gaussian kernel is chosen in this paper, which is defined as $K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$, where $x = \frac{v-v_i}{h}$.

From Equation (3), it is seen that the accuracy of the estimated probability density function depends on the smoothing parameter h and kernel function $K(\cdot)$ when the number of input values are enough large. Through statistical experiments, Epanechnikov (1969) and Scott (1992) found that different types of kernel functions have slight influences on the accuracy when the smoothing parameter is fixed. However, different values for smoothing parameter show great impacts on the accuracy. The smoothing parameter h governs the smoothness degrees of the estimated probability density functions. Large values generate oversmoothed estimations, while small values generate undersmoothed estimations. Hence, how to choose an appropriate value for the smoothing parameter is crucial to kernel density estimation. To determine an optimal value for smoothing parameter, there are many methods that have been put forward such as Scott's rule (Scott, 1992), Silverman's rule (Silverman, 1986), and cross validation (Rudemo, 1982). Scott's rule and Silverman's rule assume that the underlying distribution of input values is unimodal and normal, despite the fact that the real distribution is multimodal and non-normal. Thus, it is to be expected that the derived optimal value will be too large for the multimodal distributions. In this case, it will produce the oversmoothed probability density functions. The cross validation is an empirical method, which can produce more trustworthy optimal values regardless of the underlying distribution characteristics of the input values. All of these methods have been implemented in Python. Scott's rule and Silverman's rule are implemented in SciPy module and the cross validation is implemented in the Statsmodels KDEMultivariate class.

Example 1. Suppose that seven experts are invited to provide their preferences for a cloud storage service production with respect to its criterion performance. The collected preferences are expressed as

$$\{I_1 = 8.0, I_2 = 9.8, I_3 = 5.5, I_4 = 9.5, I_5 = 2.8, I_6 = 8.6, I_7 = 3.2\}$$

The above preferences are reordered in descending order as

$$\{v_1 = 9.8, v_2 = 9.5, v_3 = 8.6, v_4 = 8.0, v_5 = 5.5, v_6 = 3.2, v_7 = 2.8\}$$

The estimated probability density functions are depicted in Figures 3–5 when different methods are used.

As shown in Figures 3 and 4, it is seen that the estimated probability density functions are oversmoothed and they are incapable of identifying the local cluster structures of the input values when the Scott's rule and Silverman's rule are used to determine the optimal values for the smoothing parameter. As shown in Figure 5, when the cross validation is used, two local cluster structures can be identified in the estimated probability density function. Hence, the optimal value derived from the cross validation method is more trustworthy.

We can also estimate the desirable probability density function by manually adjusting the value of the smoothing parameter. As the value of the smoothing parameter varies, various estimated probability density functions can be generated as depicted in Figure 6.

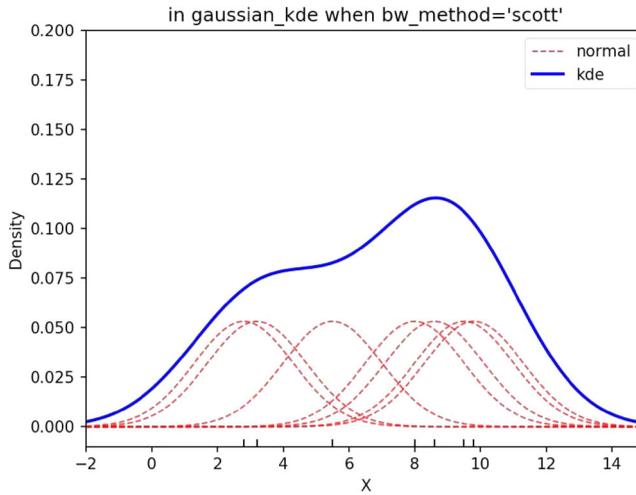


Figure 3. The estimated probability density function of input values when using Scott's rule. *Source:* The authors' data.

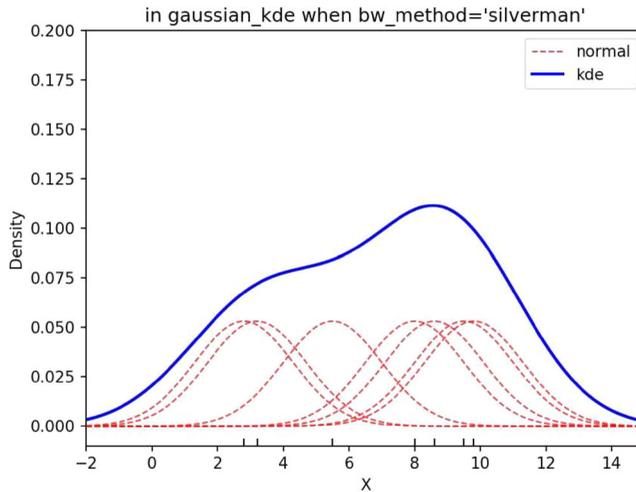


Figure 4. The estimated probability density function of input values when using Silverman's rule. *Source:* The authors' data.

As shown in Figure 6, the probability density function shown as the blue curve is undersmoothed since it fits “biased” data when the value of smoothing parameter $h = 0.6$. The probability density function shown as the black curve is oversmoothed since it is incapable of identifying the underlying local cluster structures of the input values when $h = 2.0$. The probability density function shown as the green curve is considered to be optimally smoothed since it approximates the real data distribution characteristic of input values when $h = 1.4$.

Through the above analysis, it can be seen that the estimated probability density function can not only capture the local densities of given input values, but also identify their local cluster structures without using time-consuming classification methods. It is well known that the input values with high probability densities have large importance and the input values with low probability densities have small importance.

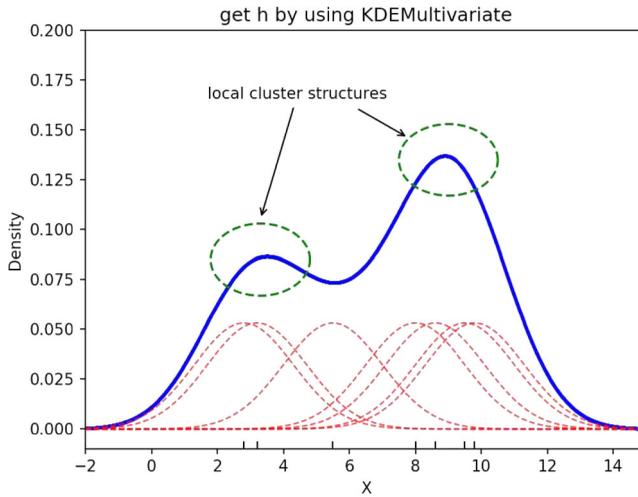


Figure 5. The estimated probability density function of input values when using cross validation. Source: The authors' data.

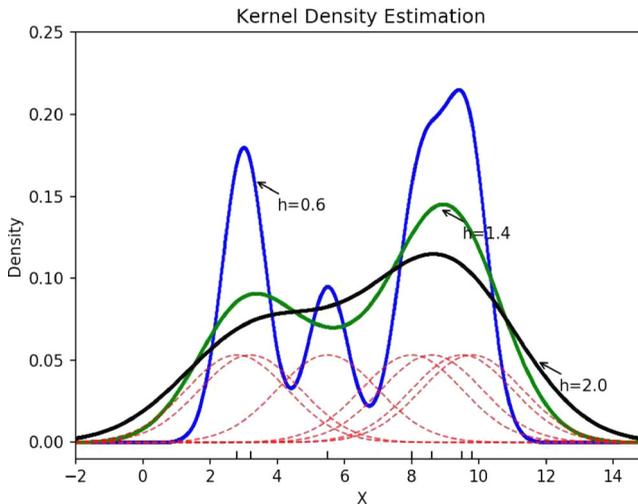


Figure 6. The estimated probability density functions for different values of smoothing parameter. Source: The authors' data.

Following this rule, the input values with high probability densities should be assigned with large weights and the input values with low probability densities should be assigned with relatively small weights. Therefore, based on the estimated probability density function, the weights of the OWA operator can be derived as

$$\begin{aligned}
 w_j &= \frac{\hat{p}(v_j)}{\sum_{l=1}^n \hat{p}(v_l)} = \frac{\frac{1}{nh} \sum_{i=1}^n K\left(\frac{v_j - v_i}{h}\right)}{\sum_{l=1}^n \left(\frac{1}{nh} \sum_{i=1}^n K\left(\frac{v_l - v_i}{h}\right)\right)} \\
 &= \frac{\sum_{i=1}^n K\left(\frac{v_j - v_i}{h}\right)}{\sum_{l=1}^n \sum_{i=1}^n K\left(\frac{v_l - v_i}{h}\right)} = \frac{\sum_{i=1}^n e^{-\frac{1}{2}\left(\frac{v_j - v_i}{h}\right)^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}\left(\frac{v_l - v_i}{h}\right)^2}}
 \end{aligned}
 \tag{4}$$

where $W = [w_1, w_2, \dots, w_j, \dots, w_n]$ denotes the weight vector of the OWA operator that satisfies $w_j \in [0, 1]$ and $\sum_{j=1}^n w_j \in [0, 1]$.

Let $d(v_j) = \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2}$ denote the proximity density (PD) of the input value v_j to the other input values, then we can obtain the following theorem.

Theorem 1. *Given a set of input values $\{I_1, I_2, \dots, I_n\}$, then its descending ordered set is $\{v_1, v_2, \dots, v_n\}$. If the proximity densities of input values v_j and v_k satisfy that $d(v_j) \geq d(v_k)$, then $w(v_j) \geq w(v_k)$.*

Proof. According to equation (4), we can have

$$w(v_j)-w(v_k) = \frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2} - \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_k-v_i}{h})^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}} = \frac{d(v_j)-d(v_k)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}}$$

Since $d(v_j) \geq d(v_k)$, then $w(v_j)-w(v_k) \geq 0$. It completes the proof of Theorem 1. It can be observed that the OWA operator weights of descending ordered input values depend on their proximity densities.

4. Probability density based OWA operator

Based on the above weight determining method, in this paper, the PDOWA operator is defined as:

$$PDOWA(I_1, I_2, \dots, I_n) = \sum_{j=1}^n w_j v_j = \sum_{j=1}^n \left(\frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2} v_j}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}} \right) \quad (5)$$

According to Equations (1) and (2), the orness and dispersion of the PDOWA operator are computed as

$$O(W) = \frac{1}{n-1} \sum_{j=1}^n (n-j) \left(\frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}} \right) \quad (6)$$

$$E(W) = - \sum_{j=1}^n \left(\frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}} \right) \ln \left(\frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_j-v_i}{h})^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_l-v_i}{h})^2}} \right) \quad (7)$$

when $I_1 = I_2 = \dots = I_n$, then we have $O(W) = \frac{1}{2}$ and $E(W) = \ln n$.

In the following part, some properties of the proposed PDOWA operator are discussed.

Property 1 (Boundedness) Given a set of input values $\{I_1, I_2, \dots, I_n\}$, then we have

$$\min(I_1, I_2, \dots, I_n) \leq PDOWA(I_1, I_2, \dots, I_n) \leq \max(I_1, I_2, \dots, I_n)$$

Proof. According to equation (5), we have

$$\begin{aligned} PDOWA(I_1, I_2, \dots, I_n) &= \sum_{j=1}^n w_j v_j = \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} v_j \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} \\ &\geq \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} \min(v_1, v_2, \dots, v_n) \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} \\ &= \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} \min(I_1, I_2, \dots, I_n) \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} = \min(I_1, I_2, \dots, I_n) \end{aligned}$$

Similarly,

$$\begin{aligned} PDOWA(I_1, I_2, \dots, I_n) &= \sum_{j=1}^n w_j v_j = \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} v_j \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} \\ &\leq \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} \max(v_1, v_2, \dots, v_n) \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} \\ &= \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} \max(I_1, I_2, \dots, I_n) \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} = \max(I_1, I_2, \dots, I_n) \end{aligned}$$

which completes the proof of property 1.

Property 2 (Commutativity) If $\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n$ is any permutation of I_1, I_2, \dots, I_n , then we have

$$PDOWA(I_1, I_2, \dots, I_n) = PDOWA(\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n)$$

Proof. If $\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n$ is any permutation of given input values I_1, I_2, \dots, I_n , then their descending ordered sets are equal, which is v_1, v_2, \dots, v_n . Therefore, we have

$$\begin{aligned} PDOWA(I_1, I_2, \dots, I_n) &= \sum_{j=1}^n w_j v_j = \frac{\sum_{j=1}^n \left(\sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_i - v_j}{h} \right)^2} v_j \right)}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2} \left(\frac{v_l - v_i}{h} \right)^2}} \\ &= PDOWA(\tilde{I}_1, \tilde{I}_2, \dots, \tilde{I}_n) \end{aligned}$$

which completes the proof of property 2.

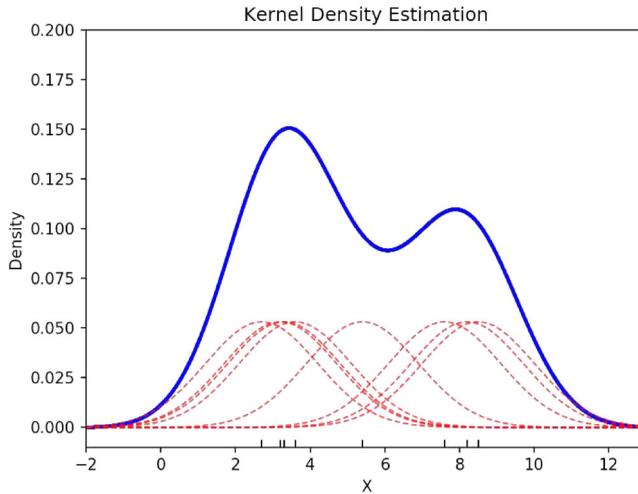


Figure 7. The estimated probability density function fitting online reviews. *Source:* The authors' data.

Property 3 (Idempotency) If $I_1 = I_2 = \dots = I_n = I$, then we have

$$PDOWA(I_1, I_2, \dots, I_n) = I$$

Proof. If $I_1 = I_2 = \dots = I_n = I$, then we have $v_1 = v_2 = \dots = v_n = I$. According to Equation (4), the weights of input values are

$$w_j = \frac{\sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_i-v_j}{h})^2}}{\sum_{l=1}^n \sum_{i=1}^n e^{-\frac{1}{2}(\frac{v_i-v_l}{h})^2}} = \frac{1}{n}$$

Using Equation (5), we have

$$PDOWA(I_1, I_2, \dots, I_n) = \sum_{j=1}^n w_j v_j = \sum_{j=1}^n \frac{1}{n} * I = I.$$

which completes the proof of property 3.

Example 2. The online reviews of a brand car are collected as $\{2.7, 3.2, 3.6, 8.2, 8.5, 5.4, 7.6, 3.3\}$. When the proposed PDOWA operator is used, the implementation processes are listed as follows:

1. They are reordered in descending order as $\{8.5, 8.2, 7.6, 5.4, 3.6, 3.3, 3.2, 2.7\}$;
2. The probability density function of the online reviews is estimated as depicted in Figure 7.
3. The estimated probability density function is used to computed the weights of the PDOWA operator as

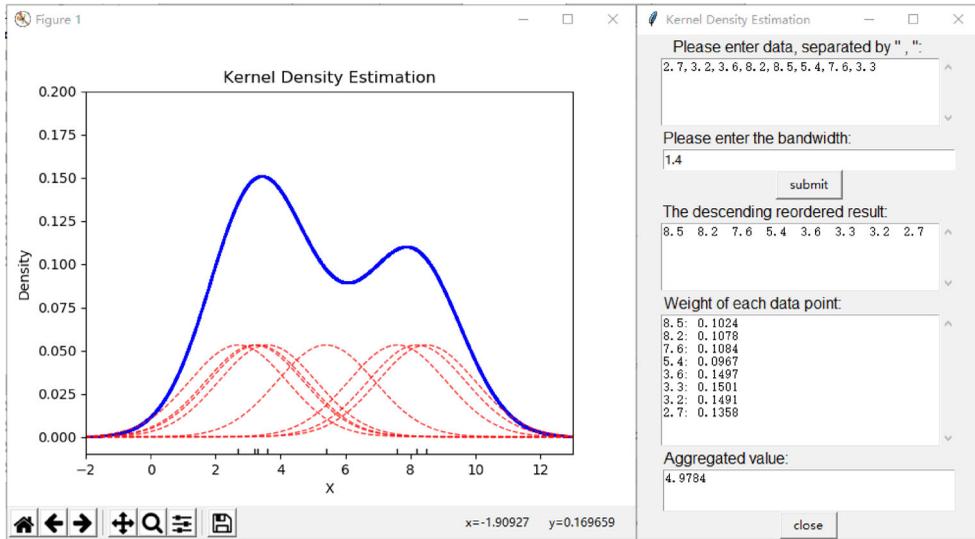


Figure 8. The smart application performing the implementation processes of the PDOWA operator. Source: The authors' data.

$$w_1(v_1) = 0.1024, w_2(v_2) = 0.1078, w_3(v_3) = 0.1084, w_4(v_4) = 0.0967, \\ w_5(v_5) = 0.1497, w_6(v_6) = 0.1501, w_7(v_7) = 0.1491, w_8(v_8) = 0.1358$$

(4) The reordered online reviews associated with their weights are aggregated into a single value as

$$PDOWA(I_1, I_2, \dots, I_n) = 4.9784$$

To automatically perform the implementation processes of the PDOWA operator, a smart application is developed using Python. Tkinter, the standard GUI (Graphical User Interface) library in Python, is adopted to develop the GUI for this smart application. As shown in Figure 8, after running the smart application, users can enter a series of input values separated by commas and specify a value for the smoothing parameter h . When the button “submit” is clicked, the processes for reordering, weighting, and aggregating input values can be performed automatically along with the estimation of probability density function.

5. Illustrative example and comparative analysis

In this section, we present an illustrative example to illustrate the application of the proposed PDOWA operator in the MCDM problem and then the comparative analysis is also given.

5.1. Illustrative example

In this subsection, the proposed PDOWA operator is applied to aggregate the evaluation information of multiple experts since the experts usually cannot reach

Table 1. Decision matrix provided by the first expert.

	c_1	c_2	c_3	c_4
a_1	8.0	7.8	7.8	7.5
a_2	7.9	7.7	7.5	7.6
a_3	8.5	8.3	8.5	8.8
a_4	8.7	8.5	8.9	8.8

Source: The authors' data.

Table 2. Decision matrix provided by the second expert.

	c_1	c_2	c_3	c_4
a_1	8.0	8.0	7.9	8.1
a_2	8.1	7.9	7.8	7.7
a_3	8.9	8.5	9.0	8.9
a_4	8.2	8.6	8.0	8.8

Source: The authors' data.

Table 3. Decision matrix provided by the third expert.

	c_1	c_2	c_3	c_4
a_1	8.0	7.8	8.2	9.1
a_2	7.5	7.9	8.1	8.8
a_3	9.2	8.8	8.5	9.5
a_4	9.0	8.2	8.0	9.2

Source: The authors' data.

Table 4. Decision matrix provided by the fourth expert.

	c_1	c_2	c_3	c_4
a_1	8.5	8.3	7.8	9.0
a_2	7.9	8.1	8.1	8.9
a_3	6.2	6.8	6.1	6.9
a_4	6.6	6.6	5.8	7.4

Source: The authors' data.

Table 5. Decision matrix provided by the fifth expert.

	c_1	c_2	c_3	c_4
a_1	5.4	4.6	4.9	5.1
a_2	5.5	4.9	5.4	5.5
a_3	5.9	6.9	7.4	7.4
a_4	5.7	6.5	7.6	8.1

Source: The authors' data.

consensus so that their evaluation information shows the complex data distribution characteristics in terms of local clusters.

Example 3. An organization intends to purchase a batch of smart phones. After screening, four alternatives $\{a_1, a_2, a_3, a_4\}$ are selected to be further evaluated according to four criteria: price (c_1), performance (c_2), battery life (c_3), and after-sale service (c_4). The weight values of these four criteria are $(0.3, 0.3, 0.2, 0.2)$. Eight experts $\{e_1, e_2, \dots, e_8\}$ are called to evaluate these four alternatives with respect to their four criteria and then all the evaluation information is collected to construct eight decision matrices as shown in Tables 1–8.

In the following part, we show the application of the proposed PDOWA operator in this group MCDM problem.

Step 1: The criteria values of four alternatives in the above eight decision matrices are fused using the following equation:

Table 6. Decision matrix provided by the sixth expert.

	c_1	c_2	c_3	c_4
a_1	5.5	5.9	5.9	6.0
a_2	5.1	5.9	5.2	5.8
a_3	5.2	4.2	4.9	5.0
a_4	5.1	4.5	5.1	5.0

Source: The authors' data.

Table 7. Decision matrix provided by the seventh expert.

	c_1	c_2	c_3	c_4
a_1	5.8	5.0	5.9	5.4
a_2	5.5	5.1	6.2	5.9
a_3	5.1	4.7	5.2	5.1
a_4	5.2	4.8	5.4	5.6

Source: The authors' data.

Table 8. Decision matrix provided by the eighth expert.

	c_1	c_2	c_3	c_4
a_1	5.8	6.0	6.4	5.9
a_2	5.9	6.1	5.6	5.4
a_3	5.2	5.2	4.7	5.2
a_4	5.1	5.5	4.8	5.8

Source: The authors' data.

Table 9. The group aggregated criteria values.

	Group aggregated criteria values
a_1	$\{I_1^1, I_1^2, I_1^3, I_1^4, I_1^5, I_1^6, I_1^7, I_1^8\} = \{7.8, 8.0, 8.2, 8.4, 5.0, 5.8, 5.5, 6.0\}$
a_2	$\{I_2^1, I_2^2, I_2^3, I_2^4, I_2^5, I_2^6, I_2^7, I_2^8\} = \{7.7, 7.9, 8.0, 8.2, 5.3, 5.5, 5.6, 5.8\}$
a_3	$\{I_3^1, I_3^2, I_3^3, I_3^4, I_3^5, I_3^6, I_3^7, I_3^8\} = \{8.5, 8.8, 9.0, 6.5, 6.8, 4.8, 5.0, 5.1\}$
a_4	$\{I_4^1, I_4^2, I_4^3, I_4^4, I_4^5, I_4^6, I_4^7, I_4^8\} = \{8.7, 8.4, 8.6, 6.6, 6.8, 4.9, 5.2, 5.3\}$

Source: The authors' data.

Table 10. The reordered group aggregated criteria values.

	Reordered group aggregated criteria values
a_1	$\{v_1^1, v_1^2, v_1^3, v_1^4, v_1^5, v_1^6, v_1^7, v_1^8\} = \{8.4, 8.2, 8.0, 7.8, 6.0, 5.8, 5.5, 5.0\}$
a_2	$\{v_2^1, v_2^2, v_2^3, v_2^4, v_2^5, v_2^6, v_2^7, v_2^8\} = \{8.2, 8.0, 7.9, 7.7, 5.8, 5.6, 5.5, 5.3\}$
a_3	$\{v_3^1, v_3^2, v_3^3, v_3^4, v_3^5, v_3^6, v_3^7, v_3^8\} = \{9.0, 8.8, 8.5, 6.8, 6.5, 5.1, 5.0, 4.8\}$
a_4	$\{v_4^1, v_4^2, v_4^3, v_4^4, v_4^5, v_4^6, v_4^7, v_4^8\} = \{8.7, 8.6, 8.4, 6.8, 6.6, 5.3, 5.2, 4.9\}$

Source: The authors' data.

$$I_i^k = \omega_j r_{ij}^k$$

where r_{ij}^k is the evaluation information of alternative a_i with respect to criterion c_j that is provided by the k th expert. The term ω_j denotes the weight value of criterion c_j . The term I_i^k denotes the aggregated criteria value of alternative a_i in the decision matrix provided by the k th expert.

The aggregated criteria values of four alternatives provided by eight experts are shown in Table 9. For example, the set $\{7.8, 8.0, 8.2, 8.4, 5.0, 5.8, 5.5, 6.0\}$ contains all the aggregated criteria values of alternative a_1 provide by eight experts.

Step 2: All the aggregated criteria values of alternatives provided by eight experts are reordered in the descending order. The reordered aggregated criteria values are shown in Table 10.

Step 3: Using Equation (4), the OWA weights of all the aggregated criteria values of four alternatives are computed as

$$w(a_1) = (0.1184, 0.1251, 0.1306, 0.1348, 0.1340, 0.1301, 0.1224, 0.1047);$$

$$w(a_2) = (0.1174, 0.1240, 0.1269, 0.1318, 0.1318, 0.1269, 0.1240, 0.1174);$$

$$w(a_3) = (0.1086, 0.1146, 0.1215, 0.1372, 0.1393, 0.1303, 0.1275, 0.1210);$$

$$w(a_4) = (0.1120, 0.1152, 0.1210, 0.1415, 0.1421, 0.1282, 0.1252, 0.1147)$$

Step 4: Using Equation (5), the final aggregated criteria values of four alternatives are computed as

$$\begin{aligned} PDOWA(a_1) &= 6.8714, PDOWA(a_2) = 6.7500, PDOWA(a_3) = 6.7398, PDOWA(a_4) \\ &= 6.7748 \end{aligned}$$

Therefore, the ranking result of these four alternatives is $a_1 \succ a_4 \succ a_2 \succ a_3$.

5.2. Comparative analysis

To verify the superiority of our proposed PDOWA operator, we also use the normal distribution based OWA operator (Xu, 2005), clus-DOWA operator (Boongoen & Shen, 2008), and MC-DOWA operator (Li et al., 2016) to handle the above MCDM problem.

(1) When the normal distribution based OWA (NDOWA) operator is applied to process Example 3, the OWA weights of all the aggregated criteria values of four alternatives are computed as

$$w(a_1) = (0.0588, 0.1042, 0.1525, 0.1845, 0.1845, 0.1525, 0.1042, 0.0588);$$

$$w(a_2) = (0.0588, 0.1042, 0.1525, 0.1845, 0.1845, 0.1525, 0.1042, 0.0588);$$

$$w(a_3) = (0.0588, 0.1042, 0.1525, 0.1845, 0.1845, 0.1525, 0.1042, 0.0588);$$

$$w(a_4) = (0.0588, 0.1042, 0.1525, 0.1845, 0.1845, 0.1525, 0.1042, 0.0588)$$

It can be seen that the OWA weights of all the aggregated criteria values of four alternatives are equal. That is because it assumes that the OWA weights follow the normal distribution and then the corresponding probability density function is applied to compute the OWA weights of all the aggregated criteria values of four alternatives. However, this probability density function is instantiated using the number of aggregated criteria values of four alternatives instead of their aggregated

Table 11. The clusters for the clus-DOWA operator.

	Clusters
a_1	$\{8.4, 8.2\}, \{8.0, 7.8\}, \{6.0, 5.8, 5.5, 5.0\}$
a_2	$\{8.2, 8.0, 7.9, 7.7\}, \{5.8, 5.6, 5.5, 5.3\}$
a_3	$\{9.0, 8.8, 8.5\}, \{6.8, 6.5\}, \{5.1, 5.0, 4.8\}$
a_4	$\{8.7, 8.6, 8.4\}, \{6.8, 6.6\}, \{5.3, 5.2, 4.9\}$

Source: The authors' data.

criteria values. Hence, the probability density function cannot identify local clusters of aggregated criteria values as the proposed PDOWA operator.

The final aggregated criteria values of four alternatives are computed as

$$NDOWA(a_1) = 6.8661, NDOWA(a_2) = 6.7500, NDOWA(a_3) = 6.7772, NDOWA(a_4) = 6.7992$$

Therefore, the ranking result of these four alternatives is $a_1 \succ a_4 \succ a_3 \succ a_2$.

(2) When the clus-DOWA operator is applied to process [Example 3](#), then the modified agglomerative hierarchical clustering algorithm should be first used to divide the aggregated criteria values of alternatives into clusters as shown in [Table 11](#).

Then, the OWA weights of aggregated criteria values of four alternatives are computed as

$$w(a_1) = (0.1304, 0.1304, 0.1304, 0.1304, 0.1304, 0.1304, 0.1180, 0.0994);$$

$$w(a_2) = (0.1212, 0.1342, 0.1342, 0.1104, 0.1212, 0.1342, 0.1342, 0.1104);$$

$$w(a_3) = (0.1274, 0.1274, 0.1120, 0.1197, 0.1197, 0.1351, 0.1351, 0.1236);$$

$$w(a_4) = (0.1327, 0.1327, 0.1173, 0.1224, 0.1224, 0.1327, 0.1327, 0.1071)$$

For alternative a_1 , it can be noted that most of the OWA weights are 0.1304 except the weights of the last two aggregated values. It is unreasonable. That is because the clustering algorithm mistakenly partitions the aggregated criteria values $\{8.4, 8.2, 8.0, 7.8\}$ into two clusters: $\{8.4, 8.2\}$ and $\{8.0, 7.8\}$.

Finally, the final aggregated criteria values of four alternatives are computed as

$$clus-DOWA(a_1) = 6.9097, clus-DOWA(a_2) = 6.7554, clus-DOWA(a_3) = 6.7695, clus-DOWA(a_4) = 6.8393$$

Therefore, the ranking result of these four alternatives is $a_1 \succ a_4 \succ a_3 \succ a_2$.

(3) When the MC-DOWA operator is used to handle [Example 3](#), then the classification method should be first used to group the aggregated criteria values of four alternatives into some clusters. Here, we use the k-means method to divide the aggregated criteria values of four alternatives into clusters as shown in [Table 12](#).

Table 12. The clusters for the MC-DOWA operator.

	Clusters
a_1	$\{8.4, 8.2, 8.0, 7.8\}, \{6.0, 5.8, 5.5, 5.0\}$
a_2	$\{8.2, 8.0, 7.9, 7.7\}, \{5.8, 5.6, 5.5, 5.3\}$
a_3	$\{9.0, 8.8, 8.5\}, \{6.8, 6.5\}, \{5.1, 5.0, 4.8\}$
a_4	$\{8.7, 8.6, 8.4\}, \{6.8, 6.6\}, \{5.3, 5.2, 4.9\}$

Source: The authors' data.

Then, the local weights of the aggregated criteria values of alternative a_1 are derived as $w_l(8.4) = w_l(8.2) = w_l(8.0) = w_l(7.8) = 0.25, w_l(6.0) = w_l(5.8) = w_l(5.5) = w_l(5.0) = 0.25$

The global weights of two clusters of alternative a_1 are derived as

$$w_g(\{8.4, 8.2, 8.0, 7.8\}) = 0.5, w_g(\{6.0, 5.8, 5.5, 5.0\}) = 0.5$$

The local weights of the aggregated criteria values of alternative a_2 are derived as $w_l(8.2) = w_l(8.0) = w_l(7.9) = w_l(7.7) = 0.25, w_l(5.8) = w_l(5.6) = w_l(5.5) = w_l(5.3) = 0.25$

The global weights of two clusters of alternative a_2 are derived as

$$w_g(\{8.2, 8.0, 7.9, 7.7\}) = 0.5, w_g(\{5.8, 5.6, 5.5, 5.3\}) = 0.5$$

The local weights of the aggregated criteria values of alternative a_3 are derived as

$$w_l(9.0) = w_l(8.8) = w_l(8.5) = 0.33, w_l(6.8) = w_l(6.5) = 0.5, w_l(5.1) = w_l(5.0) = w_l(4.8) = 0.33$$

The global weights of three clusters of alternative a_3 are derived as

$$w_g(\{9.0, 8.8, 8.5\}) = 0.42, w_g(\{6.8, 6.5\}) = 0.42, w_g(\{5.1, 5.0, 4.8\}) = 0.15$$

The local weights of the aggregated criteria values of alternative a_4 are derived as

$$w_l(8.7) = w_l(8.6) = w_l(8.4) = 0.33, w_l(6.8) = w_l(6.6) = 0.5, w_l(5.3) = w_l(5.2) = w_l(4.9) = 0.33$$

The global weights of three clusters of alternative a_4 are derived as

$$w_g(\{8.7, 8.6, 8.4\}) = 0.42, w_g(\{6.8, 6.6\}) = 0.42, w_g(\{5.3, 5.2, 4.9\}) = 0.15$$

Finally, the final aggregated criteria values of four alternatives are computed as

$$MC-DOWA(a_1) = 6.8375, MC-DOWA(a_2) = 6.7500, MC-DOWA(a_3) = 6.7655, MC-DOWA(a_4) = 6.7590$$

Therefore, the ranking result of these four alternatives is $a_1 \succ a_3 \succ a_4 \succ a_2$.

Table 13. The ranking results obtained from different operators.

Operators	Ranking results
The proposed PDOWA	$a_1 \succ a_4 \succ a_2 \succ a_3$
NDOWA (Xu, 2005)	$a_1 \succ a_4 \succ a_3 \succ a_2$
clus-DOWA (Boongoen & Shen, 2008)	$a_1 \succ a_4 \succ a_3 \succ a_2$
MC-DOWA (Li et al., 2016)	$a_1 \succ a_3 \succ a_4 \succ a_2$

Source: The authors' data.

The ranking results obtained from the above four operators are summarized in Table 13.

From Table 13, it can be noted that the proposed PDOWA operator obtains the same best alternative as the NDOWA, clus-DOWA, and MC-DOWA operators. It shows the effectiveness of the proposed PDOWA operator. However, the ranking result obtained from the proposed PDOWA operator is different from those that are obtained from the other operators. The reasons are analyzed as follows:

1. For the NDOWA operator, it is assumed that the OWA weights of aggregated criteria values follow the normal distribution and the corresponding probability density function is applied to determine the OWA weights. However, this probability density function need to be instantiated using the number of aggregated criteria values instead of aggregated criteria values. Hence, the probability density function cannot identify local clusters of aggregated criteria values so that it derives the same set of OWA weights for different sets of aggregated criteria values with the same number of aggregated criteria values. As depicted in Example 3, it is noted that the sets of OWA weights for four alternatives $\{a_1, a_2, a_3, a_4\}$ are equal.
2. For the clus-DOWA operator, the modified agglomerative hierarchical clustering algorithm should be first used to divide the aggregated criteria values of alternatives into clusters. Nevertheless, the algorithm may derive wrong clustering results, which greatly influence the OWA weights. For example, it mistakenly divides the aggregated criteria values $\{8.4, 8.2, 8.0, 7.8\}$ of alternative a_1 into two clusters: $\{8.4, 8.2\}$ and $\{8.0, 7.8\}$.
3. For the MC-DOWA operator, the aggregated criteria values of each alternative in the same clusters own the same weight even if the aggregated criteria values in the same clusters are different. Therefore, the obtained OWA weights are unreasonable.
4. The proposed PDOWA operator uses the mathematical tool of kernel density estimation to identify the underlying local cluster structures among the aggregated criteria values. Therefore, it can overcome the defects of the other operators and the derived OWA weights are more reasonable.

From the above analysis, the advantages of the proposed PDOWA operator are summarized as follows:

1. The clus-DOWA and MC-DOWA operators use the classification methods to divide the aggregated criteria values into clusters before deriving the OWA operator weights. However, the process for clustering aggregated criteria values is

time-consuming. Moreover, additional parameter should be considered during the clustering process. For example, if the k-means method is used to cluster the aggregated criteria values, then how to determine the value for the parameter k is a key problem since it can influence the clustering result. The NDOWA operator introduces the probability density function of normal distribution to determine the OWA weights. Since the probability density function only depends on the number of aggregated criteria values, it is incapable of identifying the local cluster structures of aggregated criteria values. The proposed PDOWA operator is capable of identifying the underlying local cluster structures among aggregated criteria values without using the time-consuming classification methods. Therefore, the proposed PDOWA operator is simple but efficient.

2. The clustering results generated by the modified agglomerative hierarchical clustering algorithm in the clus-DOWA operator may be wrong, which can lead to unreasonable OWA weights and ranking results. The OWA weights obtained by the MC-DOWA operator are also unreasonable since the aggregated criteria values in each cluster are assigned with equal weights. The proposed PDOWA operator computes the OWA weights of aggregated criteria values according to their probability densities. Hence, the OWA weights and ranking results obtained from the proposed PDOWA operator are more reasonable.

6. Conclusions

In this paper, a novel PDOWA operator is proposed to determine the OWA weights by considering the data distribution characteristics of input values. The kernel density estimation is first applied to estimate the PDF, which fits to input values. Using the estimated PDF, the weights of the OWA operator can be derived. Afterwards, some desirable properties are discussed. Finally, a practical example concerning the evaluation of smart phones is provided to show the application of the proposed PDOWA operator in MCDM problems.

The proposed PDOWA operator is capable of identifying the underlying local cluster structures among input values without using complicated classification methods. It is simple but efficient. Moreover, its OWA weights and ranking results are more reasonable than the existing OWA operators. However, it also has the limitations: 1) It does not consider the subjective weights of input values when aggregating input values; 2) It cannot be used to derive the OWA weights for interval-based input values.

In the future studies, we will extend the OWA operator to process rough sets (Sharma et al., 2018, 2020) and its applications in multicriteria evaluation (Roy et al., 2019; Sharma et al., 2018)

Disclosure statement

The authors declare that they have no conflict of interest.

Funding

This research work was supported by the National Natural Science Foundation of China under Grant Nos. 61872086, 61972093, and U1805263, National Innovation and Entrepreneurship Training Program for College Students under Grant No. 201910394012.

ORCID

Mingwei Lin  <http://orcid.org/0000-0003-2026-7178>

Data Availability

The data used to support the findings of this study are included within the article.

References

- Amarante, M. (2018). Mm-OWA: A generalization of OWA operators. *IEEE Transactions on Fuzzy Systems*, 26(4), 2099–2106. doi:10.1109/TFUZZ.2017.2762637
- Beliakov, G., James, S., Wilkin, T., & Calvo, T. (2018). Robustifying OWA operators for aggregating data with outliers. *IEEE Transactions on Fuzzy Systems*, 26(4), 1823–1832. doi:10.1109/TFUZZ.2017.2752861
- Boongoen, T., Shen, Q. (2008). Clus-DOWA: A New Dependent OWA Operator, In Proceedings of 2008 IEEE International Conference on Fuzzy Systems, 1057–1063.
- Casanovas, M., Torres-Martínez, A., & Merigó, J. M. (2020). Multi-person and multi-criteria decision making with the induced probabilistic ordered weighted average distance. *Soft Computing*, 24(2), 1435–1446. doi:10.1007/s00500-019-03977-6
- Epanechnikov, V. A. (1969). Nonparametric estimation of a multidimensional probability density. *Theory of Probability and Application*, 14, 153–158.
- Filev, D. P., & Yager, R. R. (1995). Analytic properties of maximum entropy OWA operators. *Information Sciences*, 85(1-3), 11–27. doi:10.1016/0020-0255(94)00109-0
- Fuller, R., & Majlender, P. (2001). An analytic approach for obtaining maximal entropy OWA operator weights. *Fuzzy Sets and Systems*, 124(1), 53–57. doi:10.1016/S0165-0114(01)00007-0
- Garg, H. (2018a). Generalized Pythagorean fuzzy geometric interactive aggregation operators using Einstein operations and their application to decision making. *Journal of Experimental & Theoretical Artificial Intelligence*, 30(6), 763–794. doi:10.1080/0952813X.2018.1467497
- Garg, H. (2018b). New exponential operational laws and their aggregation operators of interval-valued Pythagorean fuzzy information. *International Journal of Intelligent Systems*, 33(3), 653–683. doi:10.1002/int.21966
- Garg, H. (2018c). Some methods for strategic decision-making problems with immediate probabilities in Pythagorean fuzzy environment. *International Journal of Intelligent Systems*, 33(4), 687–712. doi:10.1002/int.21949
- Garg, H. (2019). Novel neutrality operations based Pythagorean fuzzy geometric aggregation operators for multiple attribute group decision analysis. *International Journal of Intelligent Systems*, 34(10), 2459–2489. doi:10.1002/int.22157
- Garg, H., & Kaur, G. (2020). Quantifying gesture information in brain hemorrhage patients using probabilistic dual hesitant fuzzy sets with unknown probability information. *Computers & Industrial Engineering*, 140, 106211. doi:10.1016/j.cie.2019.106211
- Gong, C., Li, W., & Yi, P. (2019). Rank-based analysis method to determine OWA weights and its application in group decision making. *International Journal of Intelligent Systems*, 34(7), 1685–1699. doi:10.1002/int.22116

- He, T. T., Wei, G. W., Lu, J. P., Wei, C., & Lin, R. (2019a). Pythagorean 2-tuple linguistic VIKOR method for evaluating human factors in construction project management. *Mathematics*, 7(12), 1149. doi:10.3390/math7121149
- He, T. T., Wei, G. W., Lu, J. P., Wei, C., & Lin, R. (2019b). Pythagorean 2-tuple linguistic Taxonomy method for supplier selection in medical instrument industries. *International Journal of Environmental Research and Public Health*, 16(23), 487. doi:10.3390/ijerph16234875
- Jin, L., Mesiar, R., & Yager, R. R. (2019a). Melting probability measure with OWA operator to generate fuzzy measure: the crescent method. *IEEE Transactions on Fuzzy Systems*, 27(6), 1309–1316. doi:10.1109/TFUZZ.2018.2877605
- Jin, L., Mesiar, R., & Yager, R. R. (2019b). Ordered weighted averaging aggregation on convex poset. *IEEE Transactions on Fuzzy Systems*, 27(3), 612–617. doi:10.1109/TFUZZ.2019.2893371
- Kang, B., Deng, Y., Hewage, K., & Sadiq, R. (2018). Generating Z-number based on OWA weights using maximum entropy. *International Journal of Intelligent Systems*, 33(8), 1745–1755. doi:10.1002/int.21995
- Leite, D., & Škrjanc, I. (2019). Ensemble of evolving optimal granular experts, OWA aggregation, and time series prediction. *Information Sciences*, (504), 95–112. doi:10.1016/j.ins.2019.07.053
- Lenormand, M. (2018). Generating OWA weights using truncated distributions. *International Journal of Intelligent Systems*, 33(4), 791–801. doi:10.1002/int.21963
- Lin, M. W., Chen, Z. Y., Liao, H. C., & Xu, Z. S. (2019). ELECTRE II method to deal with probabilistic linguistic term sets and its application to edge computing. *Nonlinear Dynamics*, 96(3), 2125–2143. doi:10.1007/s11071-019-04910-0
- Lin, M. W., Huang, C., & Xu, Z. S. (2020). MULTIMOORA based MCDM model for site selection of car sharing station under picture fuzzy environment. *Sustainable Cities and Society*, 53, 101873. doi:10.1016/j.scs.2019.101873
- Lin, M. W., Li, X. M., & Chen, L. F. (2020). Linguistic q-rung orthopair fuzzy sets and their interactional partitioned Heronian mean aggregation operators. *International Journal of Intelligent Systems*, 35(2), 217–249. doi:10.1002/int.22136
- Lin, M. W., Wang, H. B., Xu, Z. S., Yao, Z. Q., & Huang, J. L. (2018). Clustering algorithms based on correlation coefficients for probabilistic linguistic term sets. *International Journal of Intelligent Systems*, 33(12), 2402–2424. doi:10.1002/int.22040
- Lin, M. W., Wei, J. H., Xu, Z. S., & Chen, R. Q. (2018). Multiattribute group decision-making based on linguistic pythagorean fuzzy interaction partitioned bonferroni mean aggregation operators. *Complexity*, 2018, 1–24. doi:10.1155/2018/9531064
- Lin, M. W., Xu, Z. S., Zhai, Y. L., & Yao, Z. Q. (2018). Multi-attribute group decision-making under probabilistic uncertain linguistic environment. *Journal of the Operational Research Society*, 69(2), 157–170. doi:10.1057/s41274-017-0182-y
- Lin, M. W., Zhan, Q. S., Xu, Z. S., & Chen, R. Q. (2018). Group decision making with probabilistic hesitant multiplicative preference relations based on consistency and consensus. *IEEE Access*, 6, 63329–63344. doi:10.1109/ACCESS.2018.2876403
- Liu, P. D., Chen, S. M., & Wang, Y. M. (2020). Multiattribute group decision making based on intuitionistic fuzzy partitioned Maclaurin symmetric mean operators. *Information Sciences*, 512, 830–854. doi:10.1016/j.ins.2019.10.013
- Liu, X., Han, B., Chen, H. Y., & Zhou, L. G. (2019). The probabilistic ordered weighted continuous OWA operator and its application in group decision making. *International Journal of Machine Learning and Cybernetics*, 10(4), 705–715. doi:10.1007/s13042-017-0752-y
- Liu, J., Wang, M. T., Xu, P., Zeng, S. Z., & Liu, M. L. (2019). Intuitionistic linguistic multi-attribute decision making algorithm based on integrated distance measure. *Economic Research-Ekonomska Istraživanja*, 32(1), 3667–3683. doi:10.1080/1331677X.2019.1646146
- Liu, G., & Xiao, F. Y. (2019). Time series data fusion based on evidence theory and OWA operator. *Sensors (Switzerland)*, 19(5), 1171. doi:10.3390/s19051171

- Li, W. W., Yi, P. T., & Guo, Y. J. (2016). Majority clusters-density ordered weighting averaging: A family of new aggregation operators in group decision making. *International Journal of Intelligent Systems*, 31(12), 1166–1180. doi:10.1002/int.21821
- Li, W. W., Yi, P. T., & Zhang, D. N. (2020). Quantile-induced vector-based heavy OWA operator and the application in dynamic decision making. *International Journal of Intelligent Systems*, 35(2), 250–266. doi:10.1002/int.22207
- Ma, L., & Cong, X. H. (2019). Social stability risk assessment of NIMBY major projects by OWA, matter- element, and cloud model. *Journal of Intelligent & Fuzzy Systems*, 36(3), 2545–2556. doi:10.3233/JIFS-181259
- Maldonado, S., Merigó, J., & Miranda, J. (2018). Redefining support vector machines with the ordered weighted average. *Knowledge-Based Systems*, 148, 41–46. doi:10.1016/j.knsys.2018.02.025
- Merigó, J. M., & Yager, R. R. (2019). Aggregation operators with moving averages. *Soft Computing*, 23(21), 10601–10615. doi:10.1007/s00500-019-03892-w
- Mesiar, R., Sipeky, L., Gupta, P., & LeSheng, J. (2018). Aggregation of OWA operators. *IEEE Transactions on Fuzzy Systems*, 26(1), 284–291. doi:10.1109/TFUZZ.2017.2654482
- Mi, X., Li, J., Liao, H., Zavadskas, E. K., Al-Barakati, A., Barnawi, A., Taylan, O., & Herrera-Viedma, E. (2019). Hospitality brand management by a score-based q-rung orthopair fuzzy V.I.K.O.R. method integrated with the best worst method. *Economic Research-Ekonomiska Istraživanja*, 32(1), 3272–3295. doi:10.1080/1331677X.2019.1658533
- Pander, T. (2019). EEG signal improvement with cascaded filter based on OWA operator. *Signal, Image and Video Processing*, 13(6), 1165–1171. doi:10.1007/s11760-019-01458-9
- Riaz, M., & Tehrim, S. T. (2019). Multi-attribute group decision making based on cubic bipolar fuzzy information using averaging aggregation operators. *Journal of Intelligent & Fuzzy Systems*, 37(2), 2473–2494. doi:10.3233/JIFS-182751
- Roy, J., Sharma, H. K., Kar, S., Zavadskas, E. K., & Saparuskas, J. (2019). An extended COPRAS model for multi-criteria decision-making problems and its application in web-based hotel evaluation and selection. *Economic Research-Ekonomiska Istraživanja*, 32(1), 219–253. doi:10.1080/1331677X.2018.1543054
- Rudemo, M. (1982). Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics*, 9(2), 65–78.
- Sadiq, R., & Tesfamariam, S. (2007). Probability density functions based weights for ordered weighted averaging (OWA) operators: An example of water quality indices. *European Journal of Operational Research*, 182(3), 1350–1368. doi:10.1016/j.ejor.2006.09.041
- Scott, D. W. (1992). *Multivariate density estimation: Theory practice and visualization*. John Wiley and Sons.
- Sha, X. Y., Xu, Z. S., & Yin, C. (2019). Elliptical distribution-based weigh-determining method for ordered weighted averaging operator. *International Journal of Intelligent Systems*, 34(5), 858–877. doi:10.1002/int.22078
- Sharma, H. K., Kumari, K., & Kar, S. (2020). A rough set theory application in forecasting models. *Decision Making: Applications in Management and Engineering*, 3(2), 1–21. doi:10.31181/dmame2003001s
- Sharma, H., Kumari, K., & Kar, S. (2018). Short-term forecasting of air passengers based on hybrid rough set and double exponential smoothing models. *Intelligent Automation and Soft Computing*, 25(1), 1–14. doi:10.31209/2018.100000036
- Sharma, H., Roy, J., Kar, S., & Prentkovskis, O. (2018). Multi criteria evaluation framework for prioritizing Indian railway stations using modified rough AHP-Mabac method. *Transport and Telecommunication Journal*, 19(2), 113–127. doi:10.2478/ttj-2018-0010
- Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis, Monographs on Statistics and Applied Probability*. Chapman and Hall.
- Tang, X., Wei, G. W., & Gao, H. (2019). Pythagorean fuzzy Muirhead mean operators in multiple attribute decision making for evaluating of emerging technology commercialization. *Economic Research-Ekonomiska Istraživanja*, 32(1), 1667–1696. doi:10.1080/1331677X.2019.1638808

- Wei, G. W., Wei, C., Wu, J., & Wang, H. J. (2019). Supplier selection of medical consumption products with a probabilistic linguistic MABAC method. *International Journal of Environmental Research and Public Health*, 16(24), 5082. doi:[10.3390/ijerph16245082](https://doi.org/10.3390/ijerph16245082)
- Wei, G. W., Wu, J., Wei, C., Wang, J., & Lu, J. (2019). Models for MADM with 2-tuple linguistic neutrosophic Dombi Bonferroni mean operators. *IEEE Access*, 7, 108878–108905. doi:[10.1109/ACCESS.2019.2930324](https://doi.org/10.1109/ACCESS.2019.2930324)
- Xu, Z. S. (2005). An overview of methods for determining OWA weights. *International Journal of Intelligent Systems*, 20(8), 843–865. doi:[10.1002/int.20097](https://doi.org/10.1002/int.20097)
- Xu, Z. S. (2006). Dependent OWA operators. *Lecture Notes in Computer Science*, 3885, 172–178.
- Yager, R. R. (1988). On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1), 183–190. doi:[10.1109/21.87068](https://doi.org/10.1109/21.87068)
- Yager, R. R. (1993). Families of OWA operators. *Fuzzy Sets and Systems*, 59(2), 125–148. doi:[10.1016/0165-0114\(93\)90194-M](https://doi.org/10.1016/0165-0114(93)90194-M)
- Yager, R. R. (1999). Nonmonotonic OWA operators. *Soft Computing*, 3(3), 187–196. doi:[10.1007/s005000050068](https://doi.org/10.1007/s005000050068)
- Yager, R. R. (2019). OWA aggregation with an uncertainty over the arguments. *Information Fusion*, 52, 206–212. doi:[10.1016/j.inffus.2018.12.009](https://doi.org/10.1016/j.inffus.2018.12.009)
- Yager, R. R., & Filev, D. P. (1994). *Essentials of fuzzy modeling and control*. John Wiley and Sons.
- Yi, P., & Li, W. (2019). Induced cluster-based OWA operators with reliability measures and the application in group decision-making. *International Journal of Intelligent Systems*, 34(4), 527–540. doi:[10.1002/int.22063](https://doi.org/10.1002/int.22063)
- Zeng, S. Z., Peng, X. M., Balezentis, T., & Streimikiene, D. (2019). Prioritization of low-carbon suppliers based on Pythagorean fuzzy group decision making with self-confidence level. *Economic Research- Ekonomska Istraživanja*, 32(1), 1073–1087. doi:[10.1080/1331677X.2019.1615971](https://doi.org/10.1080/1331677X.2019.1615971)