International Journal of Approximate Reasoning 180 (2025) 109384

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

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

An unsupervised feature extraction and fusion framework for multi-source data based on copula theory

Xiuwei Chen, Li Lai *, Maokang Luo

School of Mathematics, Sichuan University, Chengdu 610064, PR China

ARTICLE INFO

Keywords:
Copula theory
Feature fusion
Feature extraction
Multi-source information system

ABSTRACT

With the development of big data technology, people are increasingly facing the challenge of dealing with massive amounts of multi-source or multi-sensor data. Therefore, it becomes crucial to extract valuable information from such data. Information fusion techniques provide effective solutions for handling multi-source data and can be categorized into three levels: data-level fusion, feature-level fusion, and decision-level fusion. Feature-level fusion combines features from multiple sources to create a consolidated feature, enhancing information richness. This paper proposes an unsupervised feature extraction and fusion method for multi-source data that utilizes the R-Vine copula, denoted as CF. The method starts by performing kernel density estimation to extract each data source's marginal density and distribution. Next, the maximum spanning tree is employed to select a vine structure for each attribute, and the corresponding copulas are chosen using maximum likelihood estimation and the AIC criterion. The joint probability density of each attribute across all information sources can be obtained by utilizing the relevant vine structure and copulas, serving as the final fusion feature. Finally, the proposed method is evaluated on eighteen simulated datasets and six real datasets. The results indicate that compared to several state-of-the-art fusion methods, the CF method can significantly enhance the classification accuracy of popular classifiers such as KNN, SVM, and Logistic Regression.

1. Introduction

Multi-source data refers to data collection from different data sources or acquisition devices. In practical applications, multi-source data can include data from various sensors, instruments, and sources. These data sources can offer different perspectives and information. Integrating and fusing these data makes obtaining more comprehensive, accurate, and reliable information possible. The processing and analysis of multi-source data involve data quality, consistency, and fusion to extract valuable knowledge and information that support decision-making and application needs.

Information fusion techniques encompass integrating and consolidating information derived from diverse sources [1–3]. The primary objective entails extracting comprehensive, accurate, and reliable knowledge and information through synthesizing data from disparate origins. These techniques find application across various domains, including anomaly monitoring [4,5], medical diagnosis [6,7], and intelligent transportation [8,9]. Information fusion techniques serve to address the challenges posed by data acquired from multiple sensors or sources, capitalizing on the inherent strengths of each source while mitigating the effects of noise, redundancy, and uncertainty. The information fusion techniques typically involve three levels: data-level fusion [10,11], decision-level

* Corresponding author.
E-mail addresses: xiuweichen1998@163.com (X. Chen), laili@scu.edu.cn (L. Lai), makaluo@scu.edu.cn (M. Luo).

https://doi.org/10.1016/j.ijar.2025.109384
Received 26 August 2024; Received in revised form 19 January 2025; Accepted 11 February 2025

Available online 13 February 2025

fusion [12,13], and feature-level fusion [14,15]. Data-level fusion integrates raw data from diverse sources to enhance data quality, reliability, and effectiveness, surpassing mere aggregation and aiming to facilitate improved data mining through a comprehensive representation. Decision-level fusion can consolidate decisions from multiple sources to form a unified and enhanced decision. This process improves the overall decision-making efficiency by leveraging diverse and complementary information. Feature-level fusion builds upon the foundation of data-level fusion by extracting, amalgamating, and refining features from a variety of sources, aiming to create a unified and enriched feature that captures the salient characteristics of the underlying data. This process involves extracting relevant features, combining them strategically, and encoding them into a cohesive representation, fostering a more detailed and informative data description. By integrating features from multiple sources, feature-level fusion enhances the depth and discriminative power of the data representation, facilitating more effective analysis and decision-making in complex data environments.

In terms of integrating multi-source data, rough set theory provides a mathematical framework for dealing with uncertainty and incompleteness in data [16]. It allows for the analysis and classification of objects based on their discernibility and dependency relationships. The rough set theory identifies essential attributes contributing to decision-making by dividing the data space into lower and upper approximations. In recent years, a multitude of fusion algorithms based on rough set theory has been proposed, addressing various types of data such as incomplete interval-valued data [17], interval-valued order data [18], and fuzzy incomplete data [19]. These methods quantify the uncertainty of data by defining information entropy and aim to minimize uncertainty to select the optimal information sources. However, these methods do not consider the joint effect among multiple information sources during the fusion process. Consequently, the fusion results are merely a single representation of the original multi-source information system, which may be limited in cases where multiple information sources are highly correlated. Therefore, it is worth studying how to incorporate the joint effect among multiple information sources into the final fusion result.

Copula theory is a powerful mathematical framework that enables the modeling and analysis of multivariate dependencies [20]. Another prevalent method for addressing data correlations involves employing multi-view representation techniques that leverage deep neural networks, like Deep CCA [21]. These methodologies are designed to comprehend intricate nonlinear relationships between two sets of variables to produce highly correlated representations. They emphasize discovering nonlinear transformations that optimize the correlation between the two sets of variables. Conversely, Copula theory primarily deals with characterizing the dependency structure among one set of variables. It provides a flexible approach to capturing the joint distribution of random variables, regardless of their individual marginal distributions. Copula theory allows for a more comprehensive understanding and modeling of multivariate dependencies by separating the marginal distributions from the dependence structure. Copulas have gained significant attention in various fields, including finance [22,23], process monitoring [24,25], and risk management [26,27]. One particular type of copula structure that has gained popularity is the vine copula, specifically the Regular Vine copula model [28–30]. The R-Vine copula model is a hierarchical structure that decomposes the multivariate dependence into a cascade of bivariate dependencies. This approach offers several advantages when dealing with complex dependence structures. Firstly, the underlying dependence can be more complex in real-world applications, making it computationally challenging to directly estimate and model the full joint distribution. The R-Vine copula model addresses this issue by breaking down the joint distribution into simpler bivariate copula relationships. Each bivariate copula captures a specific pairwise dependence, simplifying the estimation and interpretation of the overall dependence structure. Secondly, the R-Vine copula model allows for greater flexibility in modeling complex dependencies. It provides a rich set of copula families that can be combined in a customizable manner to capture various types of dependence patterns. This flexibility includes the ability to capture tail dependencies, asymmetry, and non-linear relationships, which are often present in real-world data. Lastly, the R-Vine copula structure offers improved interpretability and ease of implementation. Decomposing the dependence structure into bivariate copula relationships allows a clearer understanding of the underlying dependencies. Additionally, the modular nature of the R-Vine copula model provides the possibility of parallelization and computational efficiency.

Motivated by the limitations of rough-set-based fusion algorithms in capturing the joint effects of multiple information sources, this paper proposes a novel feature extraction and fusion algorithm based on the R-Vine copula. It utilizes the vine structure to construct the joint probability density of each attribute across multiple information sources, reflecting the joint effect of multiple information sources in the fusion feature. Firstly, kernel density estimation [31] is used to extract each information source's marginal density and distribution. Then, the maximum spanning tree algorithm [32] is employed to estimate the optimal vine structure. Additionally, the corresponding bivariate copulas are selected using the Maximum Likelihood Estimation and the AIC criterion [33]. Finally, the joint probability density of multiple information sources is obtained based on the vine structure and bivariate copulas. This joint probability density can serve as the fusion feature, capturing the joint effect of multiple information sources. An illustrated framework can be seen in Fig. 1.

The structure of this paper is as follows: Section 2 presents a comprehensive review of the relevant background information, including an explanation of copula theory and multi-source information systems. Section 3 proposes the information fusion approach based on the regular vine copula. The corresponding algorithm is presented, and its time complexity is analyzed. In Section 4, a series of experiments on the simulated datasets and real datasets are conducted to demonstrate the effectiveness of the proposed method compared to seven alternative fusion approaches and the raw sources. Finally, Section 5 summarizes the paper and discusses its limitations while providing an outlook on future work.

2. Preliminaries

This section introduces the foundational theories, including copula theory and multi-source information systems.

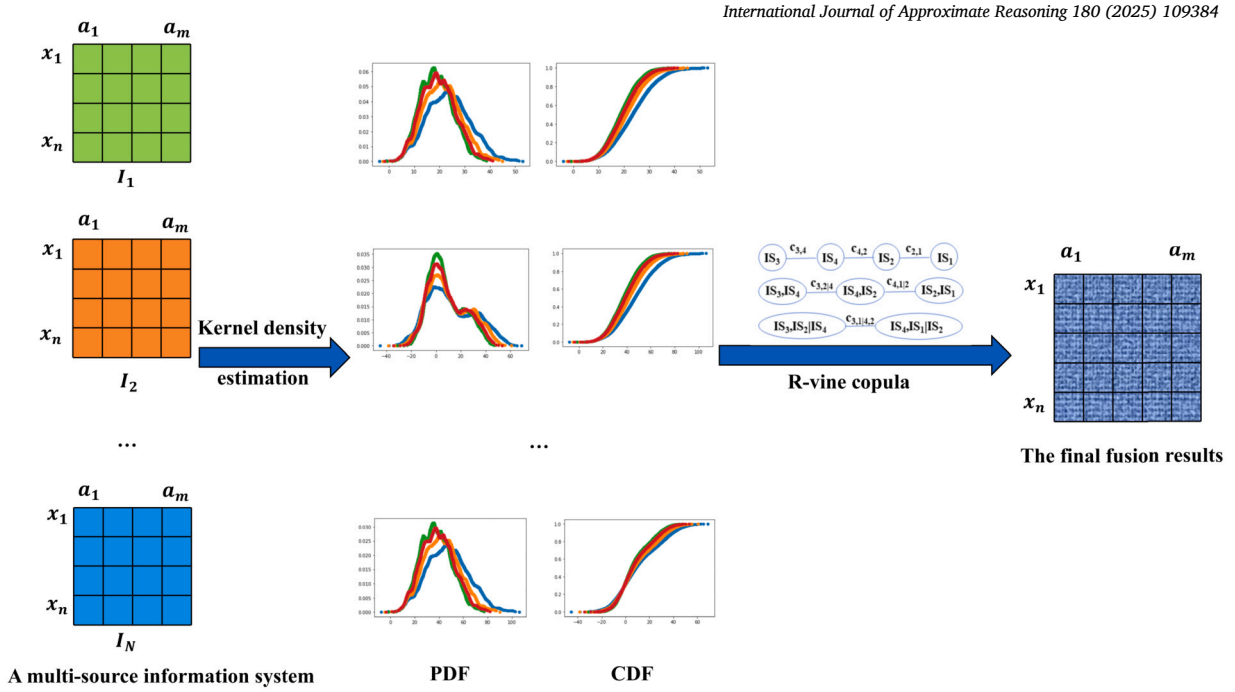

Fig. 1. The fusion framework of the proposed CF method. Firstly, kernel density estimation is used to extract each information source's marginal density (PDF) and distribution (CDF). Then, the maximum spanning tree algorithm is employed to estimate the optimal R-vine structure. Additionally, the corresponding bivariate copulas are selected using the Maximum Likelihood Estimation and the AIC criterion. Finally, the joint probability density of multiple information sources is obtained based on the vine structure and bivariate copulas. This joint probability density can serve as the fusion feature, capturing the joint effect of multiple information sources. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

2.1. Copula theory

A copula is a multivariate distribution with uniform marginal distributions, meaning each variable follows a uniform distribution over its respective ranges separately. Sklar's theorem [34] establishes the fundamental relationship between a multivariate copula and any multivariate distribution. It forms the basis of copula theory by stating that any multivariate distribution can be uniquely represented by its marginal distributions and a copula function.

Theorem 2.1. *Given a multivariate distribution function* $F\left(z_1,...,z_N\right)$ *with marginal distribution functions* $F_1\left(z_1\right)$,..., $F_N\left(z_N\right)$, *there exists a copula function* $C:[0,1]^N \to [0,1]$ *such that*

$$F\left(z_1,...,z_N\right) = C\left(F_1\left(z_1\right),...,F_N\left(z_N\right)\right). \tag{1}$$

Conversely, given a copula C *and univariate cumulative distribution functions* $F_1,...,F_N$, *then* $F\left(z_1,...,z_N\right)$ *is a valid multivariate cumulative distribution function with marginals* $F_1,...,F_N$. *Furthermore, for continuous distributions* F *and marginal distribution functions* F_1,..., F_N, *the joint probability density function of* z_1,..., z_N *can be obtained by*

$$f\left(z_1,...,z_N\right) = c\left(F_1\left(z_1\right),...,F_N\left(z_N\right)\right)\prod_{i=1}^{n} f_i\left(z_i\right), \tag{2}$$

where f_i *is the marginal densities of* z_i, *and* $c\left(u_1,...,u_N\right) = \frac{\partial^n C(u_1,...,u_N)}{\partial u_1,...,\partial u_N}$ *is the density of copula function* C.

2.2. Multi-source information systems

Let $IS = (U, A, V, f)$ represent an information system, where $U = \{x_1,...,x_n\}$ denotes the sample set of IS, $A = \{a_1,...,a_m\}$ represents the conditional attribute set of IS, V denotes the domain of A, and $f: U \times A \to V$ represents the information function of A.

Expanding on the definition of an IS, a MsIS can be represented as

$$MsIS = \{IS_i = (U, A, V_i, f_i) \,|\, i = 1,...,N\}, \tag{3}$$

where IS_i denotes the i-th subsystem of the MsIS.

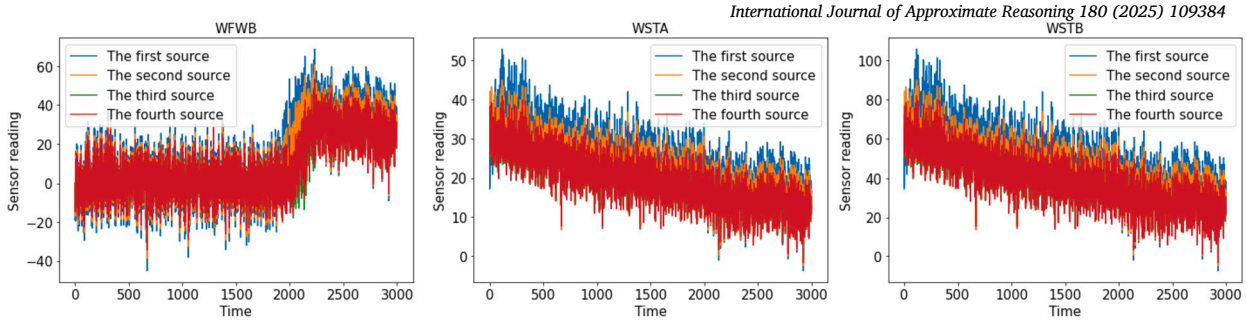

Fig. 2. A multi-source system from a nuclear power plant. The WFWB, WSTA, and WSTB represent the feedwater flow rate in evaporator B pipe, steam flow rate in evaporator A pipe, and steam flow rate in evaporator B pipe, respectively. The x-axis represents the time of data collection. And the y-axis denotes the sensor readings.

Example 2.1. In practical applications of state monitoring for nuclear power plants, multiple sensors are typically installed at key locations better to monitor the operating conditions of the nuclear power plant. These sensors collect operational data from the power plant, and the fusion of data from multiple sensors is employed to enhance monitoring accuracy. Fig. 2 illustrates a multi-source information system derived from a nuclear power plant, comprising three attributes and four information sources. Specifically, WFWB, WSTA, and WSTB represent the feedwater flow rate in evaporator B pipe, steam flow rate in evaporator A pipe, and steam flow rate in evaporator B pipe, respectively. The x-axis represents the time of data collection. And the y-axis denotes the sensor readings.

3. Feature extraction and fusion of MsIS based on regular vine copula

With the development of information technology, people are confronted with data from multiple sources. Studying how to extract useful information from multi-source data efficiently is of practical significance. This paper proposes a feature extraction and fusion method for a multi-source information system based on regular vine copula theory. By constructing the relevant vine structure, the joint density of each attribute under multiple information sources can be captured, reflecting the joint effects of multiple information sources.

Given a multi-source information system, the attributes under each information source can be treated as random variables. At the same time, the values observed for each object can be regarded as samples drawn from these variables. Consequently, the copula theory can be utilized to obtain the joint probability density of the attribute variables across all information sources, which can serve as the fusion result and reflect the collective effects of multiple information sources. Without loss of generality, in the subsequent discussion, let $\{a_s^1, ..., a_s^N\}$ denote the set of the attribute a_s under N sources, and $\left\{x^{a_s^1}, ..., x^{a_s^N}\right\}$ denote the corresponding set of sampling points.

Definition 3.1. Given a MsIS $MsIS = \{IS_i = (U, A, V_i, f_i) \,|\, i = 1, ..., N\}$. For any $a_s \in A$, let $\{a_s^1, ..., a_s^N\}$ denote the set of the attribute a_s under N sources and $\left\{x^{a_s^1}, ..., x^{a_s^N}\right\}$ denote the corresponding set of sampling points. The joint probability density of a_s under N sources can be denoted as

$$f_{a_s^1,...,a_s^N}\left(x^{a_s^1}, ..., x^{a_s^N}\right) = c\left(F_{a_s^1}\left(x^{a_s^1}\right), ..., F_{a_s^N}\left(x^{a_s^N}\right)\right)\prod_{i=1}^{N} f_{a_s^i}\left(x^{a_s^i}\right), \tag{4}$$

where c is the density of copula function.

Finding the function c directly in formula (4) can be challenging in practical applications. To address this issue, T. Bedford et al. [28,29] introduced the concept of vine structure, which decomposes the modeling of multivariate random variables into modeling the dependence relationships of a series of bivariate variables. In this study, we adopt the regular vine copula theory to effectively model the joint effects of multiple information sources.

Definition 3.2. Let $V = (T_1, ..., T_{N-1})$ be the set of $N-1$ trees, where the set of nodes and edges of the tree T_i are denoted as N_i and E_i, respectively. The tree set V is called an N-element vine structure if it satisfies [32]

1) $N_1 = \{1, 2, ..., N\}$;
2) $N_i = E_{i-1}, i = 2, ..., N-1$.

Specially, for any edge $\{v, w\} \in E_{i+1}$, the edges $v, w \in E_i$ must share a common node, then the tree set V is called an N-element regular vine.

Definition 3.3. Let $V = (T_1, ..., T_{N-1})$ be the set of $N-1$ trees, where the set of nodes and edges of the tree T_i are denoted as N_i and E_i, respectively. For any edge $e_i = \{v, w\} \in E_i$, the conditioning node set of the edge e_i is denoted as [32]

$$D_{e_i} = U_v \cap U_w, \tag{5}$$

where $U_{e_i} = \left\{ e \in N_1 \,\middle|\, \exists e_j \in E_j, j = 1, ..., i-1 \ st.\ e \in e_1 \in ... \in e_j \in e_i \right\}$. Furthermore, the conditioned nodes of the edge e_i are denoted as $\mathfrak{F}_{e_i,v} = U_v - D_{e_i}$ and $\mathfrak{F}_{e_i,w} = U_w - D_{e_i}$.

Based on the regular vine structure, let $N_1 = \left\{ a_s^1, ..., a_s^N \right\}$, the formula (4) can be rewritten as

$$f_{a_s^1,...,a_s^N}\left(x^{a_s^1}, ..., x^{a_s^N}\right) = \prod_{i=1}^{N-1} \prod_{e \in E_i} c_{\mathfrak{F}_{e,v},\mathfrak{F}_{e,w}|D_e}\left(F_{\mathfrak{F}_{e,v}|D_e}(x^{\mathfrak{F}_{e,v}} \,|\, x^{D_e}), F_{\mathfrak{F}_{e,w}|D_e}(x^{\mathfrak{F}_{e,w}} \,|\, x^{D_e})\right) \prod_{j=1}^{N} f_{a_s^j}\left(x^{a_s^j}\right), \tag{6}$$

where $x^{D_e} = \left\{ x^i \,|\, i \in D_e \right\}$, and the conditional distribution $F_{\mathfrak{F}_{e,v}|D_e}(x^{\mathfrak{F}_{e,v}} \,|\, x^{D_e})$ can be computed by [30]

$$F_{\mathfrak{F}_{e,v}|D_e}(x^{\mathfrak{F}_{e,v}} \,|\, x^{D_e}) = \frac{\partial C_{\mathfrak{F}_{v,v_1},\mathfrak{F}_{v,v_2}|D_v}\left(F_{\mathfrak{F}_{v,v_1}|D_v}(x^{\mathfrak{F}_{v,v_1}} \,|\, x^{D_v}), F_{\mathfrak{F}_{v,v_2}|D_v}(x^{\mathfrak{F}_{v,v_2}} \,|\, x^{D_v})\right)}{\partial F_{\mathfrak{F}_{v,v_2}|D_v}(x^{\mathfrak{F}_{v,v_2}} \,|\, x^{D_v})}, \tag{7}$$

where $v = \left\{ v_1, v_2 \right\}$.

To estimate the optimal R-Vine copula model for the set of random variables $\left\{ a_s^1, ..., a_s^N \right\}$, several key steps need to be taken, including the selection of the R-Vine tree structure V, the choice of bivariate copula function, and the estimation of their corresponding parameters. In order to determine the optimal R-Vine tree structure, the sequential maximum spanning tree algorithm [32] is employed. This sequential method utilizes Kendall's τ to identify the tree structure that best captures the pair-wise dependencies. These steps are as follows: 1) Generate a complete graph from N_1, where the weight of each edge is set as the absolute value of Kendall's rank correlation coefficient. 2) Apply the maximum spanning tree algorithm to obtain the maximum spanning tree T_1. 3) Repeat the above steps, sequentially selecting tree structures, until the last tree T_{N-1} is obtained, where the candidate nodes are the edges of the previous tree, and the candidate edges must satisfy the proximity condition of the regular vine.

As most candidate copula functions have unknown parameters, it is necessary to estimate the parameters of the copula functions before selecting the optimal one. For any $a_s \in A$, let $\left\{ x_i^{a_s^m} \,|\, i = 1, ..., n \right\}$ denote the n sampling points of the a_s under m-th source. Thus, for any edge $e_i = \{v, w\} \in E_i$, the maximum likelihood estimation is utilized to estimate the parameters of each candidate copula, as follows:

$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{n} \log c\left(\hat{F}_{\mathfrak{F}_{e,v}|D_e}(x_i^{\mathfrak{F}_{e,v}} \,|\, x_i^{D_e}), \hat{F}_{\mathfrak{F}_{e,w}|D_e}(x_i^{\mathfrak{F}_{e,w}} \,|\, x_i^{D_e}); \theta\right), \tag{8}$$

where $\hat{F}_{\mathfrak{F}_{e,v}|D_e}(x_i^{\mathfrak{F}_{e,v}} \,|\, x_i^{D_e})$ and $\hat{F}_{\mathfrak{F}_{e,w}|D_e}(x_i^{\mathfrak{F}_{e,w}} \,|\, x_i^{D_e})$ are computed based on formula (7) recursively.

After estimating the parameters of the copula function, the optimal bivariate copula c^* can be obtained by utilizing the Akaike Information Criterion [33], as follows:

$$c^* = \arg\min_{c}\left(-2 \sum_{i=1}^{n} \log c\left(\hat{F}_{\mathfrak{F}_{e,v}|D_e}(x_i^{\mathfrak{F}_{e,v}} \,|\, x_i^{D_e}), \hat{F}_{\mathfrak{F}_{e,w}|D_e}(x_i^{\mathfrak{F}_{e,w}} \,|\, x_i^{D_e}); \hat{\theta}\right) + 2k\right), \tag{9}$$

where k represents the number of parameters in the candidate copula function.

The fusion process is summarized as follows: first, each information source's marginal density and distribution are extracted using kernel density estimation [31]. Then, the vine structure for each attribute and corresponding copulas are estimated based on the sequential maximum spanning tree and formulas (8)-(9). Finally, formula (6) calculates each attribute's joint density across all information sources, representing the final fusion feature. The framework of the proposed CF method can be seen in Fig. 1. The corresponding algorithm for fitting the joint density function is presented in Algorithm 1. The time complexity of the step 4 is $O\left(n^2\right)$. The time complexity of the step 6 and 13 are $O\left(N^2 \times n \log n\right)$. The time complexity of the steps 7 and 14 are $O\left(N^3 \times \log N\right)$. The time complexity of the steps 9, 10, 16, 17, and 20 are $O(n)$. So the total time complexity of Algorithm 1 is $O\left(|A| \times \left(N \times n^2 + N^3 \times n \log n\right)\right)$.

Example 3.1. (Continued to Example 2.1) First, utilizing the kernel density estimate method with the Epanchnikov kernel function, the probability densities and cumulative distributions of the three attributes under four sources can be extracted, presented in Figs. 3 and 4. Next, the optimal vine structures and the corresponding copulas are estimated, shown in Table 1. In this example, the vine structures of the three attributes are the same, as depicted in Fig. 5. Finally, based on the formula (6), the final fusion features, that is, the joint probability densities, can be computed, as shown in Fig. 6. The joint probability density of each attribute can be regarded as a new feature, reflecting the combined effects of multiple sensors.

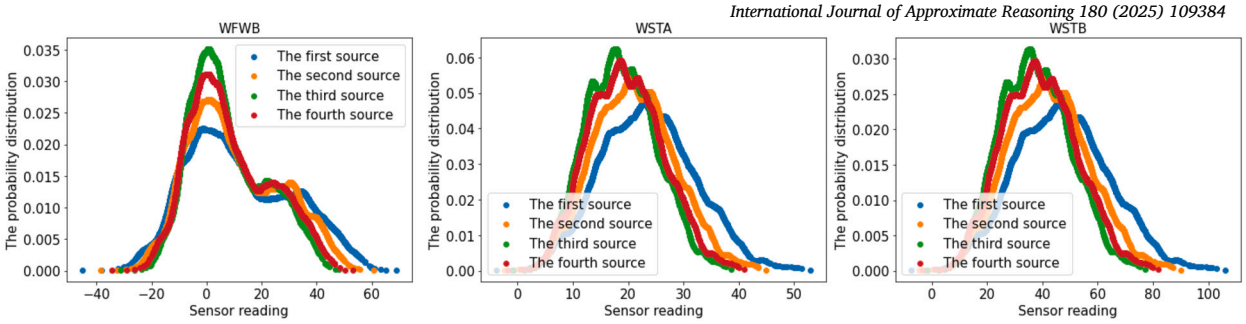

Fig. 3. The probability densities of the attributes WFWB, WSTA, and WSTB. These attributes correspond to the feedwater flow rate in evaporator B pipe, steam flow rate in evaporator A pipe, and steam flow rate in evaporator B pipe, respectively. By employing the kernel density estimate method with the Epanchnikov kernel function, one can extract the probability densities of these three attributes from four different sources. On the x-axis, sensor readings are represented, while the y-axis signifies the probability densities of the attributes.

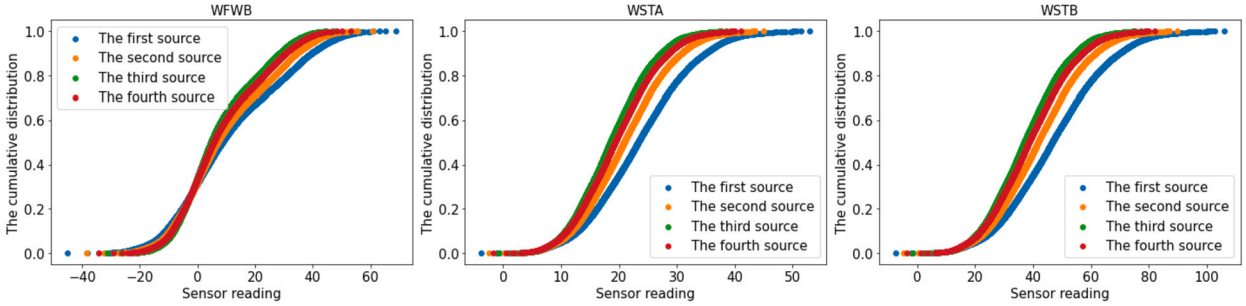

Fig. 4. The cumulative distributions of the attributes WFWB, WSTA, and WSTB. These attributes correspond to the feedwater flow rate in evaporator B pipe, steam flow rate in evaporator A pipe, and steam flow rate in evaporator B pipe, respectively. By employing the kernel density estimate method with the Epanchnikov kernel function, one can extract the cumulative distributions of these three attributes from four different sources. The x-axis represents the sensor readings, while the y-axis represents the cumulative distributions of the attributes.

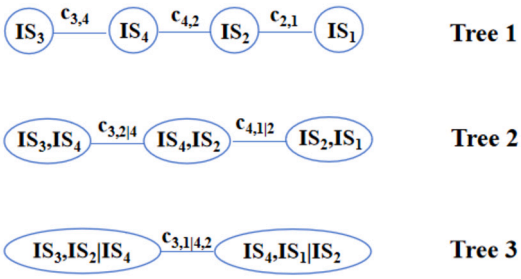

Fig. 5. The estimated vine structure of the attributes WFWB, WSTA, and WSTB. Tree 1 contains four nodes, namely IS_1-IS_4, with their copula functions being $c_{3,4}$, $c_{4,2}$, and $c_{2,1}$, respectively. Using the edges of the first tree as nodes for the next tree, Tree 2 consists of three nodes with copula functions $c_{3,2|4}$ and $c_{4,1|2}$. Then, the edges of the second tree are used as nodes for the third tree. Tree 3 has two nodes with a copula function of $c_{3,1|4,2}$.

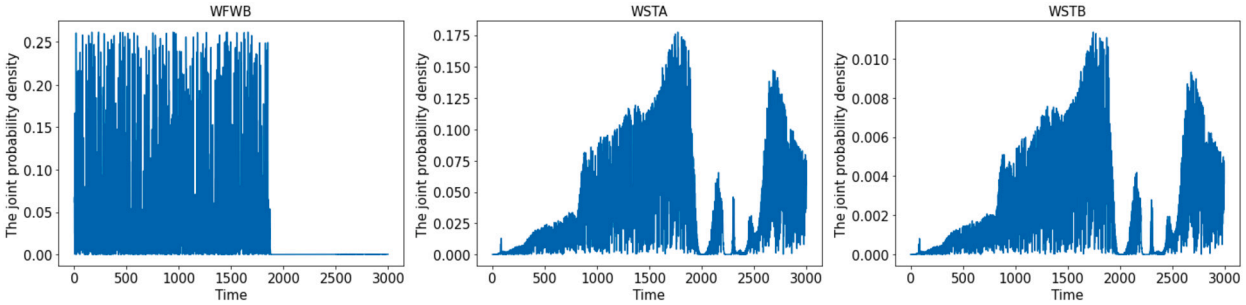

Fig. 6. The fusion results of the attributes WFWB, WSTA, and WSTB. Based on the formula (6), the final fusion features, that is, the joint probability densities, can be computed. The x-axis represents the time points, while the y-axis represents the joint probability densities of the attributes. The joint probability density of each attribute can be regarded as a new feature, reflecting the combined effects of multiple sensors.

Algorithm 1: The fusion algorithm of the proposed CF.

Input : A MsIS $MsIS = \{IS_i = (U, A, V_i, f_i) | i = 1, ..., N\}$
Output : A new fused information table with joint density

begin
  for $s = 1 : |A|$ **do**
    for $i = 1 : N$ **do**
      Estimate the marginal density $f_{a_s^i}(x^{a_s^i})$ and distribution $F_{a_s^i}(x^{a_s^i})$ based on the kernel density estimation.
    end
    Generate a complete graph from $\{a_s^1, ..., a_s^N\}$, where the weight of each edge is set as the absolute value of Kendall's rank correlation coefficient ;
    Apply the maximum spanning tree algorithm to obtain the maximum spanning tree T_1;
    for *each edge* $v = \{v_1, v_2\} \in T_1$ **do**
      Estimate the parameter of the candidate copula function and select the optimal copula based on the formula (8) and (9);
      Compute node $F_{\Im_{e,v}|D_e}(x^{\Im_{e,v}} | x^{D_e})$ for the next tree based on formula (7);
    end
    for $i = 2 : N - 1$ **do**
      Compute the Kendall's rank correlation coefficient of all candidate edges of T_i;
      Apply the maximum spanning tree algorithm to obtain the maximum spanning tree T_i;
      for *each edge* $v = \{v_1, v_2\} \in T_i$ **do**
        Estimate the parameter of the candidate copula function and select the optimal copula based on the formula (8) and (9);
        Compute node $F_{\Im_{e,v}|D_e}(x^{\Im_{e,v}} | x^{D_e})$ for the next tree based on formula (7);
      end
    end
    Compute the final joint density of a_s under all sources $f_{a_s^1,...,a_s^N}(x^{a_s^1}, ..., x^{a_s^N})$ based on formula (6);
  end
  return : $\{f_{a_1^1,...,a_1^N}(x^{a_1^1}, ..., x^{a_1^N}), ..., f_{a_{|A|}^1,...,a_{|A|}^N}(x^{a_{|A|}^1}, ..., x^{a_{|A|}^N})\}$
end

Table 1
The vine structures and copulas of three attributes under four sources.

Tree	Edge	WFWB		WSTA		WSTB	
		Copula	Parameter	Copula	Parameter	Copula	Parameter
1	IS_2, IS_1	Clayton	28.00	Clayton	28.00	Clayton	28.00
	IS_4, IS_2	Clayton	23.72	Gaussian	1.00	Gaussian	1.00
	IS_4, IS_3	Clayton	28.00	Gaussian	1.00	Gaussian	1.00
2	$IS_4, IS_1; IS_2$	Frank	-16.07	Gaussian	-0.35	Gaussian	-0.35
	$IS_3, IS_2; IS_4$	Frank	-14.05	Frank	-10.45	Frank	-10.51
3	$IS_3, IS_1; IS_4, IS_2$	Joe	1.12	Frank	-1.98	Frank	-1.95

4. Experimental analysis

This section conducted a series of experiments based on simulated datasets and real multi-source datasets to validate the performance of the proposed method. For the statistical analysis of regular vine copula models, the VineCopula[1] package proves to be an excellent tool. It offers an extensive range of functions encompassing tree structure selection, parameter estimation, and copula selection. All programs in this paper were implemented using Python, incorporating the utilization of the VineCopula package. All experiments were performed on private computers with Intel(R) Xeon(R) W-2123 CPU @3.60 GHz CPU, 64 GB RAM, and 64-bit Windows operating system.

To demonstrate the effectiveness of the proposed method, we compare it with widely adopted and state-of-the-art data fusion techniques. The details of these comparative approaches are elaborated below:

(1) Mean Fusion (MF): [35] $MF(x, a_s) = \frac{1}{N}\sum_{i=1}^{N} f_i(x, a_s^i)$, where $f_i(x, a_s^i)$ denotes the value of x under a_s^i, and $MF(x, a_s)$ denotes the value of x under a_s after fusion.
(2) Neighborhood Entropy Fusion (NRE-FS): [10] This method presents an unsupervised reduction framework for multi-source data. A novel rough entropy is introduced alongside the utilization of the Sup-Inf function to determine the most suitable information source. Subsequently, the feature selection algorithm is employed to acquire a more precise feature subset. In the following experiments, the top half of the sorted result of attributes is taken as the reduction result, and δ is fine-tuned within the range of 0.05 to 0.5 with

[1] https://github.com/tnagler/VineCopula

Table 2
The details of the used datasets.

No.	Dataset	Abbreviation	Sample	Attribute	Class
1	Chemical Composition of Ceramic Samples	CCC	88	17	2
2	Breast Tissue	BT	106	9	6
3	Breast Cancer Coimbra	BCC	116	9	2
4	Iris	Iris	150	4	3
5	Wine	Wine	178	13	3
6	BME	BME	180	128	3
7	seeds	seeds	210	7	3
8	Plane	Plane	210	144	7
9	Glass Identification	GI	214	9	7
10	Vertebral Column	VC	310	6	3
11	Ecoli	Ecoli	336	7	8
12	Libras Movement	LM	360	90	15
13	Raisin	Raisin	900	7	2
14	Rice (Cammeo and Osmancik)	RCO	3810	7	2
15	Wine Quality Wine	WQW	4898	11	7
16	Wall-Following Robot Navigation Data	WFRN	5456	24	4
17	MAGIC Gamma Telescope	MGT	19020	10	2
18	Occupancy Detection	OD	20560	6	2

increments of 0.05.

(3) Fuzzy Dominated Entropy Fusion (FDEF): [18] This method employs fuzzy set theory to establish the concept of fuzzy dominating conditional entropy, specifically for interval-valued data. The fusion outcome is determined by selecting the information source with the lowest conditional entropy.

(4) Tolerance Incomplete Entropy Fusion (TIEF): [17] This framework presents a novel tolerance relation tailored for handling incomplete interval-valued data. It utilizes this relation to define a new uncertainty measurement. By minimizing the measurement, the framework enables the selection of the most crucial information source. In the subsequent experiments, the threshold α undergoes tuning within the interval of 0.05 to 0.5, incrementing by 0.05.

(5) Fuzzy Incomplete Entropy Fusion (FIEF): [19] This method introduces a novel fusion approach explicitly designed for fuzzy incomplete data. It leverages the rough set theory to define the conditional entropy for fuzzy incomplete data. The fusion result is obtained by selecting the information source with the minimum entropy. The parameter L_a is fine-tuned between 0.05 and 0.5 with steps of 0.05 in the following experiments.

(6) Significance Degree Fusion (SDF-FS): [36] This method presents a feature selection framework for multi-label information systems. The concept of information source significance is introduced, and the best source is obtained by minimizing this significance measure. Subsequently, the positive region is employed to select a suitable feature subset.

(7) Entropy-Based Unsupervised Fusion (EUF): [37] The proposed method introduces an unsupervised fusion approach for incomplete interval-valued information systems. It defines the information entropy for incomplete interval-valued data and selects the most significant information source by minimizing the information entropy. The threshold β is fine-tuned from 0.05 to 0.5 in increments of 0.05 in the subsequent experiments.

4.1. Performance on simulated datasets

Several simulation experiments are done in this section using publicly available datasets to show that the proposed model works better than a number of commonly used algorithms. All datasets were sourced from the UCI[2] and UCR[3] repositories and have been thoroughly detailed in Table 2. The simulation methods in [35] and [38] are employed to construct a multi-source information system. Initially, N random numbers following a normal distribution $N(0, 0.1)$ and uniform distribution $U(0, 0.1)$ are generated separately. Then, 40% of the data is randomly selected and added to the normal distribution random numbers (the sample value remains unchanged when the absolute value of the random number is larger than 1), while 20% of the data is added to the uniform distribution random numbers. The remaining data remains unchanged. This process generates a multi-source information system consisting of N subsystems. In this paper, each generated multi-source information system comprises three subsystems.

To demonstrate the effectiveness of the proposed fusion method in improving classifier accuracy, the proposed approach, alongside the seven comparative algorithms, is implemented to obtain fusion results. Subsequently, these fusion results underwent classification using KNN ($k = 3$), SVM ($\gamma = 0.001$), and Logistic Regression ($C = 0.0001$) classifiers, followed by ten-fold cross-validation. By comparing the classification accuracy of the proposed method against the other algorithms, the efficacy of the proposed approach in enhancing classifier performance can be accessed. The detailed comparison results are presented in Table 3-5. Regarding the KNN classifier, the proposed CF method consistently surpasses the other seven fusion approaches, along with the mean and maximum values of information sources, across the Iris, MGT, Wine, Raisin, BCC, CCC, WFRN, and VC datasets. The CF method demonstrates

[2] http://archive.ics.uci.edu

[3] https://www.timeseriesclassification.com/index.php

Table 3
The classification accuracy of the CF and comparative algorithms based on KNN.

Datasets	CF	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	45.7±11.7	35.2±10.3	59.5±11.7	59.0±14.2	**69.0±11.1**	52.4±11.3	34.8±11.1	62.9±11.2	55.2±10.9	52.4±11.3
Iris	**64.7±8.5**	28.7±4.3	54.0±9.2	54.7±8.8	50.7±14.4	33.3±8.9	44.7±10.3	55.3±6.0	48.0±10.2	40.0±13.0
MGT	**91.9±0.8**	56.1±0.9	56.3±0.9	56.3±1.0	57.5±1.1	56.0±0.9	56.5±1.3	56.0±1.1	56.2±1.0	56.1±0.9
Wine	**55.6±7.3**	35.4±10.3	36.5±10.3	38.2±10.7	47.3±10.3	35.9±9.8	33.8±10.5	35.4±10.0	36.0±9.1	35.9±9.8
LM	27.5±6.4	17.8±4.3	23.1±6.3	23.1±6.3	28.9±6.9	22.2±6.2	8.1±2.6	**27.8±5.0**	24.2±5.1	23.3±6.1
Raisin	**71.1±5.8**	49.3±4.0	49.3±4.0	49.3±4.0	55.8±4.9	49.3±4.0	49.4±4.5	49.3±4.0	49.3±4.0	49.3±4.0
RCO	73.7±2.4	51.0±2.8	51.4±2.8	51.3±2.9	**96.3±0.8**	51.1±2.9	53.1±2.4	51.6±2.7	51.4±2.8	51.1±2.9
BCC	**80.4±19.8**	57.0±11.8	57.0±11.8	58.0±13.2	59.7±14.7	56.1±11.6	49.2±15.2	58.0±13.2	57.0±11.8	57.0±11.8
Ecoli	73.8±8.5	53.0±4.0	69.1±5.8	69.0±5.1	**75.0±6.6**	64.3±4.9	47.0±4.2	70.8±5.5	65.8±3.2	65.5±2.9
BT	24.2±11.3	22.5±11.2	22.5±11.2	22.5±11.2	**25.6±12.3**	22.5±11.2	21.8±9.7	22.5±11.2	22.5±11.2	22.5±11.2
CCC	**80.6±10.2**	44.9±20.9	46.0±20.5	46.0±20.5	61.4±8.6	44.9±20.9	53.2±16.6	46.0±20.5	46.0±20.5	44.9±20.9
WFRN	**46.3±1.6**	43.9±1.5	44.3±1.4	44.9±1.5	45.5±2.7	44.9±1.5	38.8±2.6	44.9±1.5	44.9±1.5	43.5±1.5
GI	52.2±16.8	39.7±8.1	59.2±15.5	59.2±13.4	59.8±9.1	59.7±15.6	**64.4±14.7**	62.5±15.7	59.7±15.6	59.7±15.9
VC	**55.2±7.0**	45.8±9.1	46.1±9.0	46.5±8.3	41.9±8.7	45.8±9.2	34.5±7.1	46.8±7.9	45.8±9.1	45.8±9.2
OD	73.2±0.8	74.4±0.9	75.8±1.0	75.6±1.0	**80.2±0.9**	74.9±1.0	49.6±1.1	75.8±0.7	75.9±0.8	75.5±0.9
BME	55.6±12.2	32.8±10.1	50.6±12.8	50.6±13.3	61.1±11.1	48.9±11.9	**62.2±10.8**	51.1±11.9	49.4±12.5	48.9±11.9
WQW	33.9±2.7	34.2±2.3	34.4±2.4	34.5±2.3	34.6±1.4	33.9±2.4	**36.4±2.1**	34.4±2.4	34.2±2.2	33.9±2.4
Plane	16.2±9.3	11.4±7.7	**21.4±8.0**	**21.4±7.5**	**21.4±9.3**	21.0±8.0	11.9±7.8	21.0±7.4	21.0±8.0	21.0±8.0
Average	**56.8**	40.7	47.6	47.8	54.0	45.4	41.6	48.5	46.8	45.9

Table 4
The classification accuracy of the CF and comparative algorithms based on SVM.

Datasets	CF	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	**49.5±8.8**	19.0±5.6	19.0±5.6	19.0±5.6	19.5±6.5	19.0±5.6	19.0±5.6	19.5±5.4	20.5±6.0	19.0±5.6
Iris	**45.3±11.5**	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8
MGT	**90.1±0.8**	64.3±1.1	64.3±1.1	64.3±1.1	64.6±1.1	64.2±1.1	64.8±1.1	64.3±1.1	64.2±1.1	64.2±1.1
Wine	**51.1±11.4**	37.1±7.7	38.2±7.5	38.2±7.5	43.3±8.6	37.6±7.0	39.8±7.0	37.1±7.7	37.6±7.0	37.1±7.7
LM	**21.1±5.6**	0.0±0.0	1.1±1.8	1.4±1.9	0.6±1.7	0.6±1.1	0.3±0.8	0.3±0.8	1.1±1.8	0.6±1.1
Raisin	**74.7±4.2**	47.3±3.5	47.3±3.5	47.3±3.5	52.9±3.4	47.4±3.5	47.2±3.7	47.4±3.5	47.4±3.5	47.4±3.5
RCO	**74.1±3.0**	53.7±2.7	53.7±2.8	53.8±2.8	57.2±3.4	53.6±2.7	54.6±3.1	53.7±2.8	53.6±2.7	53.5±2.7
BCC	55.0±14.9	54.5±18.3	54.5±18.3	54.5±18.3	**56.7±12.9**	54.5±18.3	51.7±13.7	54.5±18.3	54.5±18.3	54.5±18.3
Ecoli	**75.0±8.3**	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6
BT	**27.6±16.6**	9.4±7.1	9.4±7.1	9.4±7.1	19.5±11.9	9.4±7.1	16.1±9.6	9.4±7.1	9.4±7.1	9.4±7.1
CCC	**87.4±6.5**	39.2±16.3	39.2±16.3	39.2±16.3	60.1±14.1	39.2±16.3	37.8±13.3	39.2±16.3	39.2±16.3	39.2±16.3
WFRN	**42.8±1.8**	40.9±1.9	41.3±1.9	41.2±1.9	40.6±2.3	40.9±1.7	40.4±2.6	41.2±1.9	41.1±1.9	40.9±1.7
GI	**38.8±11.2**	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6
VC	48.4±10.1	48.1±9.3	47.7±8.8	47.4±8.8	50.3±11.1	47.7±9.0	**51.0±11.2**	48.4±9.3	47.7±9.0	47.7±9.0
OD	**77.1±0.4**	76.9±0.5	76.9±0.5	76.9±0.5	76.9±0.5	76.9±0.5	76.9±0.5	77.0±0.5	76.9±0.5	76.9±0.5
BME	16.1±8.0	16.1±8.0	43.3±14.9	38.3±13.3	27.2±16.2	31.1±10.0	19.4±16.7	**46.1±15.9**	37.8±12.9	31.1±10.0
WQW	43.0±2.2	44.8±2.2	44.8±2.2	44.8±2.2	**44.9±2.2**	44.8±2.2	**44.9±2.2**	44.8±2.2	44.8±2.2	44.8±2.2
Plane	**14.8±6.5**	5.7±6.3	5.2±6.5	5.2±6.5	6.2±5.2	4.8±6.4	5.7±4.2	5.7±6.3	5.7±6.3	4.8±6.4
Average	**51.8**	36.1	37.7	37.4	39.6	36.9	36.8	37.8	37.4	36.9

superior performance for the seeds and Plane datasets when contrasted with the MF and SDF-FS methods. Within the LM and Ecoli datasets, the CF method outshines the MF, TIEF, FDEF, FIEF, and SDF-FS methods, as well as the raw sources. Similarly, in the RCO and BT datasets, the CF methodology performs better than the MF, TIEF, FDEF, FIEF, SDF-FS, and EUF methods, alongside the raw sources. Notably, the CF method outperforms the MF approach in the GI dataset, while in the OD dataset, it surpasses the SDF-FS approach. Additionally, in the BME dataset, the proposed CF method excels compared to the MF, TIEF, FIEF, FDEF, and EUF methods, along with the raw sources. On average, utilizing the KNN classifier, the CF method enhances accuracy by 2.8% to 16.0% when compared to other fusion methods and raw sources.

Similarly, for the SVM classifier, the CF method consistently surpasses other widely employed fusion algorithms, along with the mean and maximum values of information sources, across datasets encompassing seeds, Iris, MGT, Wine, LM, Raisin, RCO, Ecoli, BT, CCC, WFRN, GI, OD, and Plane. Notably, in the BCC dataset, the CF method outshines the MF, TIEF, FIEF, FDEF, SDF-FS, and EUF methods, as well as the raw sources. Similarly, in the VC dataset, the CF method outperforms the MF, TIEF, FIEF, and FDEF methods, along with the raw sources. On average, employing the SVM classifier, the CF method enhances accuracy by 12.2% to 15.7% when compared to alternative fusion methods and raw sources. Likewise, according to the Logistic Regression classifier, the CF method demonstrates superiority over other techniques and raw sources in datasets including seeds, Iris, MGT, LM, Raisin, RCO, Ecoli, BT, CCC, WFRN, GI, BME, and Plane. Noteworthy is the performance of the CF method in the Wine dataset, which surpasses the MF, NRE-FS, SDF-FS, and EUF methods, as well as the mean value of sources. In the BCC dataset, the CF method excels beyond the MF,

Table 5
The classification accuracy of the CF and comparative algorithms based on Logistic Regression.

Datasets	CF	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	**48.1±11.9**	19.0±5.6	19.0±5.6	19.0±5.6	19.5±6.5	19.0±5.6	19.0±5.6	19.0±5.6	19.0±5.6	19.0±5.6
Iris	**36.0±11.6**	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8	18.7±5.8
MGT	**86.4±0.7**	64.8±1.1	64.9±1.1	64.9±1.1	64.8±1.1	64.8±1.1	64.8±1.1	64.8±1.1	64.8±1.1	64.8±1.1
Wine	44.3±11.0	42.6±8.1	44.9±9.8	**45.5±10.2**	43.8±7.8	44.9±8.0	39.8±7.0	43.2±8.1	44.3±8.4	43.8±9.3
LM	**21.4±8.0**	0.0±0.0	0.3±0.8	0.3±0.8	0.6±1.7	0.6±1.7	0.3±0.8	0.6±1.7	0.6±1.7	0.3±0.8
Raisin	**63.0±7.5**	47.0±3.7	47.8±4.7	47.4±4.5	50.9±7.5	47.7±4.3	47.1±3.3	49.3±4.0	48.2±4.6	47.7±4.3
RCO	**66.8±2.4**	57.2±3.4	57.3±3.4	57.2±3.4	57.3±3.3	57.2±3.4	57.2±3.4	57.5±3.4	57.2±3.4	57.2±3.4
BCC	55.0±14.9	51.6±12.2	53.3±11.5	53.3±10.0	**55.9±15.3**	52.4±11.7	55.0±14.9	52.5±12.7	52.4±11.7	52.4±11.7
Ecoli	**64.9±6.3**	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6	42.5±6.6
BT	**26.4±15.0**	22.7±6.7	25.6±7.8	24.6±6.5	24.4±8.1	22.7±4.9	25.1±13.2	23.6±6.3	23.6±6.3	23.6±7.8
CCC	**69.7±15.6**	40.8±15.8	43.1±14.6	41.9±14.8	47.6±23.3	41.9±14.8	36.5±11.4	41.9±14.8	41.9±14.8	41.9±14.8
WFRN	**41.9±2.0**	40.4±1.8	40.5±1.7	40.5±1.7	40.6±2.2	40.3±1.8	40.4±2.5	40.5±1.8	40.7±1.8	40.5±1.6
GI	**53.3±9.7**	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.3±9.6	31.8±10.2	31.3±9.6	31.3±9.6
VC	48.4±10.1	49.7±10.9	50.0±11.3	49.7±10.9	51.3±9.9	49.7±10.9	**51.3±11.9**	50.0±11.3	50.0±10.9	49.7±10.9
OD	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**	**76.9±0.5**
BME	**54.4±9.6**	16.1±8.0	25.0±10.3	24.4±10.6	20.0±16.9	18.9±9.7	16.1±8.0	23.9±15.3	27.8±11.1	18.9±9.7
WQW	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**	**44.9±2.2**
Plane	**12.4±6.5**	5.7±4.7	6.2±5.2	6.2±5.2	6.7±5.3	5.7±5.1	5.2±4.0	6.7±5.3	5.7±4.7	5.7±4.7
Average	**50.8**	37.3	38.5	38.3	38.8	37.8	37.3	38.2	38.4	37.8

Table 6
The Pvalues of the Wilcoxon test based on KNN.

Datasets	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	**<0.05**	0.9758	0.9625	1.0000	0.9756	**<0.05**	0.9971	0.9678	0.9756
Iris	**<0.001**	**<0.05**	**<0.05**	**<0.05**	**<0.001**	**<0.001**	**<0.05**	**<0.001**	**<0.01**
MGT	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Wine	**<0.01**	**<0.01**	**<0.01**	**<0.05**	**<0.001**	**<0.001**	**<0.01**	**<0.01**	**<0.001**
LM	**<0.01**	**<0.1**	**<0.1**	0.7798	**<0.1**	**<0.001**	0.5236	0.1170	**<0.1**
Raisin	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
RCO	**<0.001**	**<0.001**	**<0.001**	1.0000	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
BCC	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.01**	**<0.05**	**<0.05**	**<0.05**
Ecoli	**<0.001**	**<0.1**	**<0.1**	0.7239	**<0.01**	**<0.001**	0.4525	**<0.01**	**<0.01**
BT	0.4764	0.4764	0.4764	0.6943	0.4764	0.5279	0.4764	0.4764	0.4764
CCC	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
WFRN	**<0.05**	**<0.01**	**<0.05**	0.2461	**<0.05**	**<0.001**	**<0.05**	**<0.05**	**<0.05**
GI	**<0.05**	0.9199	0.9033	0.9714	0.9228	1.0000	0.9756	0.9228	0.9138
VC	**<0.05**	**<0.05**	**<0.05**	**<0.01**	**<0.05**	**<0.001**	**<0.05**	**<0.05**	**<0.05**
OD	0.9951	1.0000	1.0000	1.0000	1.0000	**<0.001**	1.0000	1.0000	1.0000
BME	**<0.01**	0.1162	0.1175	0.9045	**<0.1**	0.9526	0.1283	**<0.1**	**<0.1**
WQW	0.6875	0.7539	0.8125	0.7217	0.5391	0.9814	0.5391	0.6166	0.5391
Plane	**<0.1**	0.9155	0.9250	0.9152	0.9065	0.1969	0.9104	0.9065	0.9065

TIEF, FIEF, FDEF, and EUF methods, alongside the raw sources. On average, utilizing the Logistic Regression classifier, the CF method enhances accuracy by 12.0% to 13.5% compared to alternative fusion methods and raw sources.

Furthermore, considering the inherent randomness of cross-validation, the classification results were compared using the Wilcoxon signed-rank hypothesis test. This statistical test can evaluate the significance of differences between the CF method and the other comparative algorithms. The null hypothesis ($H_0 : \mu_{CF} \leq \mu_{other\ algorithms}$) assumes that the median of the CF method's classification results was less than or equal to the other comparative algorithms. In comparison, the alternative hypothesis ($H_1 : \mu_{CF} > \mu_{other\ algorithms}$) posited that the median of the CF method's classification results was greater than the other methods. By calculating the p-value and comparing it with predefined significance levels of 0.1, 0.05, 0.01, and 0.001, it can be determined whether there is sufficient evidence to reject the null hypothesis. The results of the Wilcoxon signed-rank tests are presented in Table 6-8.

Through the application of the Wilcoxon signed-rank test, substantial insights have emerged. For the KNN classifier, statistical significance underscores the consistent outperformance of the CF method over the other seven methodologies, as well as the mean and maximum values of information sources, within datasets including Iris, MGT, Wine, Raisin, BCC, CCC, and VC. Notably, within the seeds dataset, the CF method exhibits statistically significant superiority over the MF and SDF-FS methods. Similarly, in the LM dataset, statistical analysis reveals the CF method's significant performance advantage compared to the MF, TIEF, FIEF, FDEF, and SDF-FS methods, alongside the mean value extracted from sources. Within the Ecoli dataset, statistical significance indicates the CF method's superiority over the MF, TIEF, FIEF, FDEF, and SDF-FS methods, as well as the raw sources. In both the RCO and WFRN datasets, the CF approach statistically outshines the MF, TIEF, FIEF, FDEF, SDF-FS, and EUF methods, in addition to the raw sources. Similarly, the CF method statistically surpasses the SDF-FS method in the OD dataset. Furthermore, within the BME dataset, statistical

Table 7
The Pvalues of the Wilcoxon test based on SVM.

Datasets	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Iris	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
MGT	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Wine	**<0.05**	**<0.05**	**<0.05**	**<0.1**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**
LM	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Raisin	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
RCO	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
BCC	0.6167	0.6167	0.6167	0.8654	0.6167	0.5000	0.6167	0.6167	0.6167
Ecoli	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
BT	**<0.05**	**<0.05**	**<0.05**	0.1298	**<0.05**	**<0.1**	**<0.05**	**<0.05**	**<0.05**
CCC	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
WFRN	**<0.05**	**<0.05**	**<0.01**	**<0.01**	**<0.01**	**<0.05**	**<0.01**	**<0.01**	**<0.01**
GI	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
VC	0.5280	0.3895	0.4167	0.7239	0.4326	0.8984	0.6111	0.4326	0.4326
OD	**<0.1**	0.2001	0.1462	0.1799	0.1177	**<0.1**	0.1712	0.1377	0.2001
BME	/	1.0000	1.0000	0.9930	0.9969	0.9772	1.0000	1.0000	0.9969
WQW	0.9990	0.9990	0.9990	0.9990	0.9990	0.9990	0.9990	0.9990	0.9990
Plane	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**

Table 8
The Pvalues of the Wilcoxon test based on Logistic Regression.

Datasets	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF	Maximum value of sources	Mean value of sources
seeds	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Iris	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
MGT	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Wine	0.3630	0.5473	0.4720	0.4720	0.5941	0.1557	0.3630	0.5000	0.4720
LM	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Raisin	**<0.001**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.001**	**<0.001**	**<0.01**	**<0.01**
RCO	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
BCC	0.1473	0.3375	0.3371	0.8618	0.3375	/	0.1721	0.2092	0.2092
Ecoli	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
BT	0.2491	0.5338	0.4662	0.6643	0.3375	0.5672	0.3366	0.4328	0.3116
CCC	**<0.01**	**<0.01**	**<0.01**	**<0.05**	**<0.01**	**<0.001**	**<0.01**	**<0.01**	**<0.01**
WFRN	**<0.05**	**<0.05**	**<0.05**	**<0.1**	**<0.05**	**<0.1**	**<0.05**	**<0.1**	**<0.05**
GI	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
VC	0.8716	0.8862	0.8716	0.9244	0.8716	0.9122	0.8862	0.8716	0.8716
OD	/	/	/	/	/	/	/	/	/
BME	**<0.001**	**<0.001**	**<0.001**	**<0.01**	**<0.001**	**<0.001**	**<0.01**	**<0.01**	**<0.001**
WQW	0.9772	0.9772	0.9772	0.9772	0.9772	0.9772	0.9772	0.9772	0.9772
Plane	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.05**

analysis reveals the CF method's superiority over the MF and FDEF methods, as well as the raw sources. Moreover, in the GI and Plane datasets, the CF method significantly outperforms the MF method. Overall, the results indicate that in 62.3% of cases for the KNN classifiers, the CF method significantly surpasses the other seven algorithms, as well as the mean and maximum values of the information sources.

In a parallel manner, for the SVM classifier, statistical significance is evident as the CF method consistently outperforms other algorithms across datasets such as seeds, Iris, MGT, Wine, LM, Raisin, RCO, Ecoli, CCC, WFRN, GI, and Plane. In the BT dataset, statistical analysis highlights the CF method's significant superiority over the MF, TIEF, FIEF, FDEF, SDF-FS, and EUF methods, alongside the raw sources. Additionally, in the OD dataset, the CF method statistically outperforms the MF and SDF-FS methods. Overall, the findings reveal that in 72.8% of cases for the SVM classifiers, the CF method significantly surpasses the other seven algorithms, as well as the mean and maximum values of the information sources. Similarly, according to the Logistic Regression classifier, statistical significance underscores the superiority of the CF method over other techniques and raw sources within datasets including seeds, Iris, MGT, LM, Raisin, RCO, Ecoli, CCC, WFRN, GI, BME, and Plane. Overall, the results indicate that in 66.7% of cases for the Logistic Regression classifiers, the CF method significantly outperforms the other seven algorithms, as well as the mean and maximum values of the information sources.

For various evaluation metrics, the Nemenyi test is employed to assess the superiority of the proposed CF method compared to other approaches. This statistical test contrasts the disparities in average rankings among different algorithms against a designated critical difference (CD). If a certain disparity exceeds the critical difference, it signifies that the algorithm with the higher average ranking statistically outperforms its counterpart with the lower ranking; conversely, a lack of statistical significance is inferred. The critical difference CD is determined by [39]

Table 9
The results of the Nemenyi test based on KNN, SVM, and Logistic Regression.

Algorithms	Average ranking			Pvalues		
	KNN	SVM	LR	KNN	SVM	LR
CF	7.5000	**8.8333**	**8.7222**			
MF	2.6389	4.2500	3.1667	**<0.001**	**<0.001**	**<0.001**
TIEF	5.9722	5.2500	6.3611	0.8871	**<0.05**	0.3632
FIEF	6.2500	5.1667	5.5278	0.9662	**<0.05**	**<0.05**
NRE-FS	**8.7222**	7.1389	6.8333	0.9708	0.8076	0.6885
FDEF	3.6111	4.2778	4.3889	**<0.01**	**<0.001**	**<0.001**
SDF-FS	4.2778	5.0833	4.2222	**<0.05**	**<0.01**	**<0.001**
EUF	6.6111	5.7500	5.9444	0.9970	**<0.1**	0.1531
Maximum value of sources	5.4722	5.1944	5.5000	0.5924	**<0.05**	**<0.05**
Mean value of sources	3.9444	4.0556	4.3333	**<0.05**	**<0.001**	**<0.001**
$CD_{0.1} = 2.9469$	$CD_{0.05} = 3.1932$		$CD_{0.01} = 3.6796$		$CD_{0.001} = 4.2629$	

$$CD_{\alpha} = q_{\alpha}\sqrt{\frac{k(k+1)}{6N}},$$

Where k denotes the number of algorithms under comparison, N represents the quantity of datasets, and the q_{α} signifies the critical value. The results of the Nemenyi test are presented in Table 9.

From the perspective of average ranking, the findings indicate that for both the SVM and Logistic Regression classifiers, the CF method consistently achieves the highest average ranking compared to other algorithms. In the case of the KNN classifier, the CF method secures a higher average ranking than the MF, TIEF, FIEF, FDEF, SDF-FS, and EUF methods, as well as the raw sources. Additionally, considering Pvalues, for the KNN classifier, the CF method significantly outperforms the MF, FDEF, SDF-FS methods, and the mean value of sources. Similarly, within the SVM classifier, the CF method demonstrates clear superiority over the MF, TIEF, FIEF, FDEF, SDF-FS, and EUF methods, as well as the raw information sources. For the Logistic Regression classifier, the CF method presents significant advantages over the MF, FIEF, FDEF, and SDF-FS methods, as well as the raw information sources.

Additionally, in order to provide further insights into the robustness and applicability of the proposed method, we evaluate its performance across varying levels of noise. Specifically, we systematically vary the variances within normal and uniform distributions in the simulated data, ranging from 0.1 to 1.0 with increments of 0.1, to assess its generalizability. The classification accuracy of the proposed CF method under varying levels of noise are shown in Figs. 7-9. The x-y axis represents the variances of normal and uniform distributions, and the z-axis denotes the classification accuracy. From Figs. 7-9, it is evident that, across a majority of datasets such as Iris, MGT, LM, BCC, Ecoli, WFRN, GI, VC, OD, WQW, and Plane, the classification accuracy of the CF method using the KNN classifier exhibits a stable performance with minimal fluctuation. Similarly, with the SVM classifier, datasets such as BCC, WFRN, VC, OD, WQW, and Plane also show consistent and stable performance for the CF method. Additionally, in the case of the Logistic Regression classifier, the accuracy of the CF method remains steady across datasets including Wine, LM, BCC, Ecoli, BT, WFRN, VC, OD, WQW, and Plane. Generally speaking, based on the three classifiers, the classification accuracy of the CF method remains within a stable fluctuation range under various noise conditions, indicating a certain level of robustness.

4.2. Performance on real multi-source datasets

This subsection presents a series of experiments conducted on real-world multi-source datasets. The details of these datasets are summarized in Table 10. The classification accuracy of the CF method was compared with that of other methods using three classifiers. Moreover, we directly combine multiple original views together as the raw data to compare whether the model has improved, referred to as Raw. The results can be found in Table 11.

Regarding the KNN classifier, the proposed CF method consistently surpasses the other seven fusion approaches as well as the raw data across the MF and Yale datasets. Within the Scene15 and MNIST datasets, the CF method outperforms various algorithms such as TIEF, FIEF, NRE-FS, FDEF, SDF-FS, and EUF. In the GSALC dataset, the CF method outperforms NRE-FS, FDEF, and SDF-FS. For the SVM classifier, the CF method demonstrates superior performance compared to other methods and the raw non-fused data in the PSMTSR and MNIST datasets. Across the MF, GSALC, Scene15, and Yale datasets, the CF method performs better than several approaches like FIEF, NRE-FS, FDEF, SDF-FS, and EUF. Furthermore, with the Logistic Regression classifier, the CF method exhibits better performance than other approaches and the raw non-fused data in the MF, Scene15, and MNIST datasets. In the datasets PSMTSR and GSALC, the CF method outperforms the other seven approaches. In general, in the majority of cases, the CF method outperforms other algorithms as well as the raw non-fused data.

Furthermore, the results of the Wilcoxon signed-rank tests are presented in Table 12. For the KNN classifier, the CF method is statistically better than the other seven methodologies and the raw non-fused data, within datasets MF and MNIST. Notably, within the PSMTSR, GSALC, and Yale datasets, the CF method exhibits statistically significant superiority over the other seven methods. And in the Scene15 dataset, the CF method is statistically better than the methods MF, FDEF, and SDF-FS. In a parallel manner, for the SVM classifier, statistical significance is evident as the CF method consistently outperforms other algorithms across datasets such as Scene15 and Yale. In the datasets PSMTSR and MNIST, the CF method statistically significantly performs better than the other seven approaches. And in the MF and GSALC datasets, the CF method is statistically better than the FDEF and SDF-FS methods.

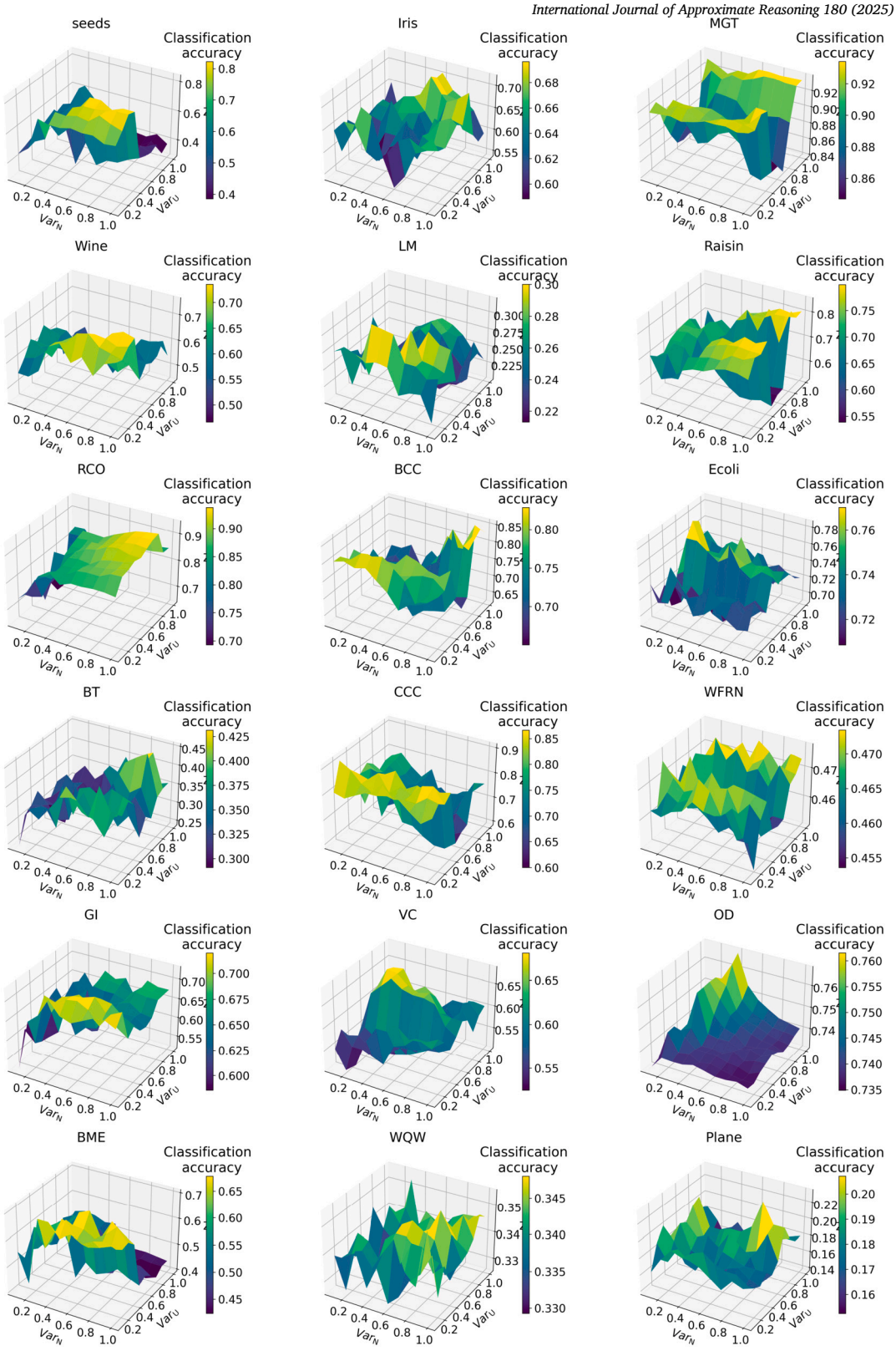

Fig. 7. The classification accuracy of the proposed CF method under varying levels of noise based on KNN. The variances in both normal and uniform distributions in the simulated data are adjusted from 0.1 to 1.0 in increments of 0.1, denoted as Var_N and Var_U, respectively. The x-y axis represents the variances of normal and uniform distributions, and the z-axis denotes the classification accuracy. Across a majority of datasets such as Iris, MGT, LM, BCC, Ecoli, WFRN, GI, VC, OD, WQW, and Plane, the classification accuracy of the CF method exhibits a stable performance with minimal fluctuation.

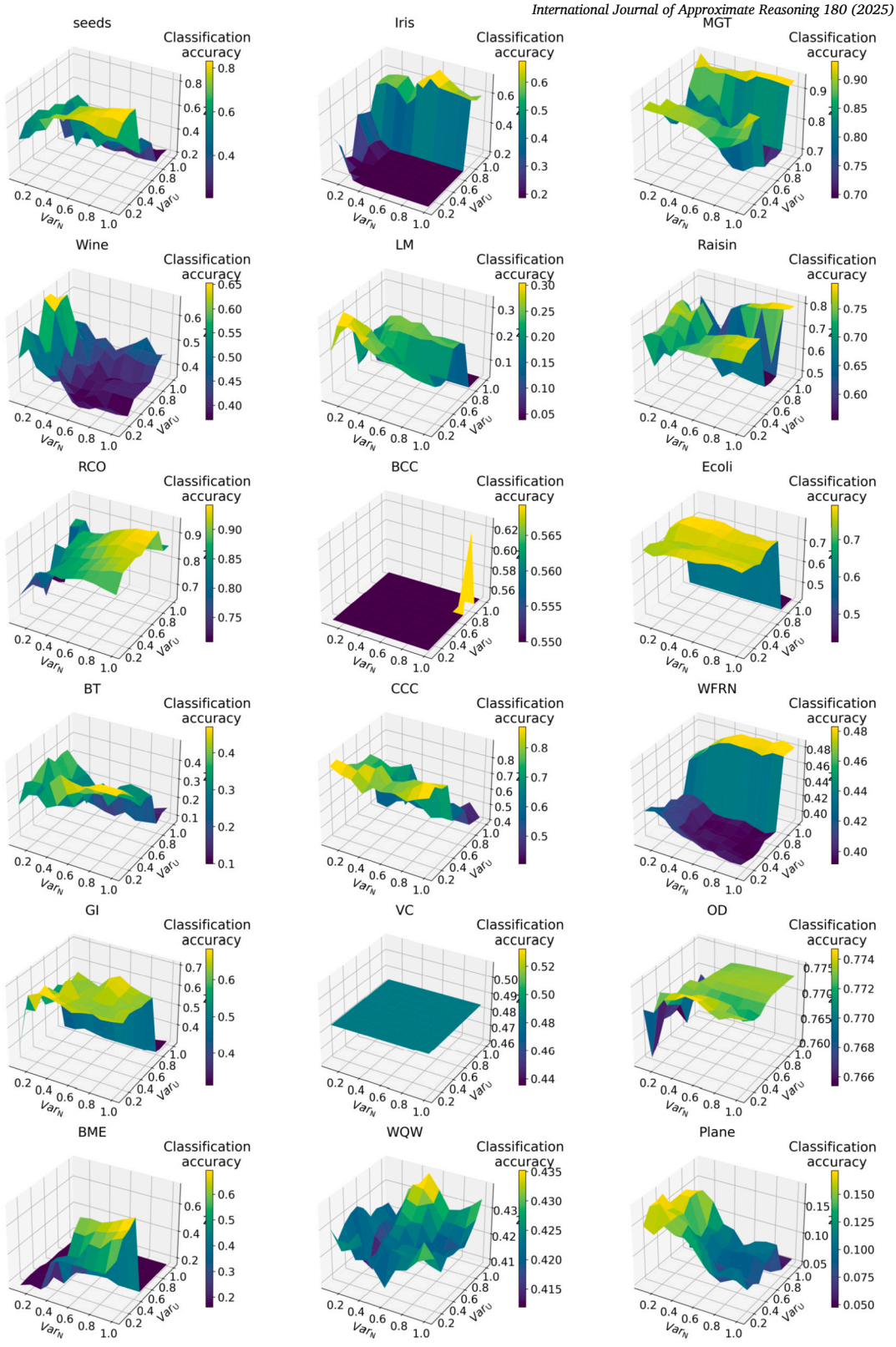

Fig. 8. The classification accuracy of the proposed CF method under varying levels of noise based on SVM. The variances in both normal and uniform distributions in the simulated data are adjusted from 0.1 to 1.0 in increments of 0.1, denoted as Var_N and Var_U, respectively. The x-y axis represents the variances of normal and uniform distributions, and the z-axis denotes the classification accuracy. Datasets such as BCC, WFRN, VC, OD, WQW, and Plane show consistent and stable performance for the CF method.

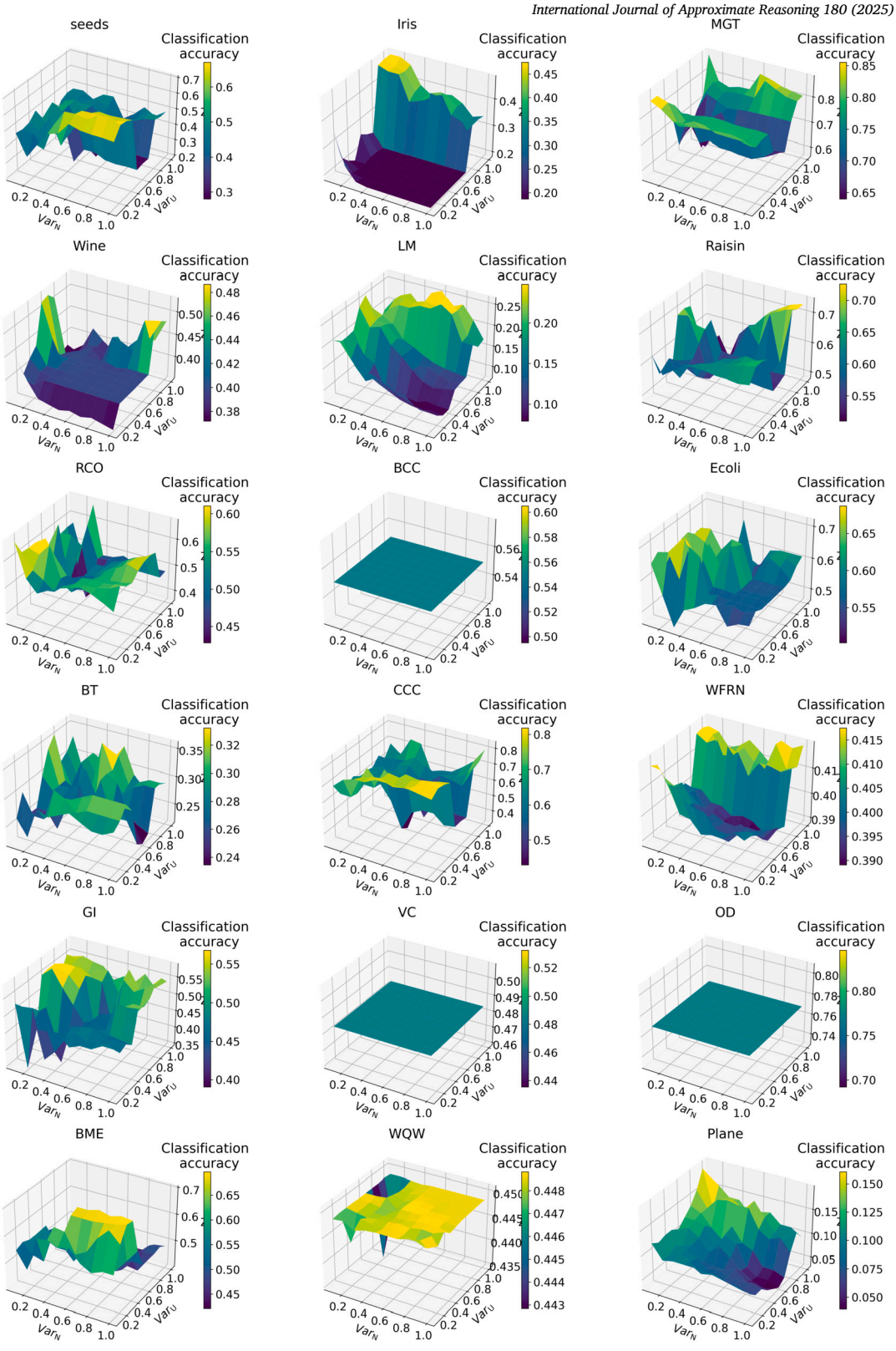

Fig. 9. The classification accuracy of the proposed CF method under varying levels of noise based on Logistic Regression. The variances in both normal and uniform distributions in the simulated data are adjusted from 0.1 to 1.0 in increments of 0.1, denoted as Var_N and Var_U, respectively. The x-y axis represents the variances of normal and uniform distributions, and the z-axis denotes the classification accuracy. The accuracy of the CF method remains steady across datasets, including Wine, LM, BCC, Ecoli, BT, WFRN, VC, OD, WQW, and Plane.

Table 10
The details of the real multi-source datasets.

No.	Dataset	Abbreviation	Sample	View	Class
1	Multiple Features [40]	MF	2000	FOU(76), FAC(216), KAR(64), PIX(240), ZER(47), MOR(6)	10
2	15-Scene Image Dataset [41]	Scene15	4485	GIST(1536), LBP(4096), HOG(1764)	15
3	Gas sensor array low-concentration [42]	GSALC	90	TGS2603(900),TGS2630(900),TGS813(900),TGS822(900), MQ-135(900), MQ-137(900), MQ-138(900), 2M012(900), VOCS-P(900), 2SH12(900)	3
4	Parkinson's Speech with Multiple Types of Sound Recordings [43]	PSMTSR	1040	View1(1)-View26(1)	2
5	MNIST-10k [44]	MNIST	10000	ISO(30), LDA(9), NPE(30)	10
6	Yale [45]	Yale	165	Intensity(4096), LBP(3304), Gabor(6750)	15

Table 11
The classification accuracy of the CF and comparative algorithms in the real datasets.

Classifier	Dataset	CF	Raw	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF
KNN	MF	**96.6±1.2**	95.2±1.3	75.1±3.0	85.4±2.9	83.8±2.4	60.2±2.8	79.2±3.2	34.9±2.8	60.8±3.7
	PSMTSR	68.9±3.1	68.6±5.6	50.8±4.1	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	56.5±2.6	60.9±3.2	**100.0±0.0**
	GSALC	80.0±14.7	83.3±14.3	80.0±16.3	83.3±17.4	82.2±17.4	75.6±19.8	68.9±8.3	42.2±8.3	**86.7±12.0**
	Scene15	24.4±1.5	19.7±1.5	**30.1±2.0**	14.6±1.7	14.3±1.6	14.7±1.9	13.4±1.6	9.3±1.3	15.1±1.8
	MNIST	94.5±0.6	**95.0±0.8**	40.7±1.9	60.2±0.9	59.1±1.7	53.2±2.0	64.0±1.6	19.2±1.6	44.3±1.8
	Yale	**73.2±7.6**	70.1±10.5	27.2±9.1	27.9±11.9	26.7±8.8	25.9±8.0	26.0±10.7	17.6±8.8	12.0±7.9
	Average	**72.9**	72.0	50.7	61.9	61.0	54.9	51.3	30.7	53.2
SVM	MF	**7.0±1.1**	**7.0±0.9**	6.5±0.8	6.6±0.8	6.5±0.8	6.5±0.8	6.5±0.8	6.5±0.8	6.5±0.8
	PSMTSR	**72.4±4.0**	69.2±3.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9
	GSALC	21.1±10.5	**32.2±11.6**	11.1±17.2	12.2±18.9	12.2±18.9	11.1±17.9	8.9±13.9	7.8±14.1	12.2±18.9
	Scene15	51.6±1.9	**52.4±1.9**	30.0±1.4	15.5±2.2	16.5±1.5	14.8±1.1	15.2±1.6	11.6±1.3	16.4±0.9
	MNIST	**95.7±0.7**	90.8±0.7	41.7±1.6	56.7±1.1	56.7±1.1	47.4±1.3	62.7±1.4	19.5±1.2	36.9±1.7
	Yale	**31.4±11.4**	**31.4±11.4**	12.6±7.7	**31.4±11.4**	26.5±11.1	24.0±12.0	25.3±11.4	7.2±5.2	4.2±4.6
	Average	46.5	**47.2**	26.5	29.9	29.2	26.8	29.3	18.3	22.2
Logistic regression	MF	**97.0±1.2**	96.7±1.0	73.8±4.4	80.2±2.8	79.8±3.3	45.5±4.2	79.4±2.0	36.8±3.9	61.9±2.4
	PSMTSR	62.7±3.6	**63.9±3.2**	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9	56.9±4.9
	GSALC	13.3±12.0	**23.3±10.5**	8.9±13.9	11.1±17.2	11.1±17.2	11.1±17.2	8.9±13.9	5.6±9.0	11.1±17.9
	Scene15	**50.1±2.3**	23.6±2.0	29.9±2.3	16.6±1.5	16.1±1.0	16.4±1.5	14.7±1.0	11.9±1.6	16.8±1.2
	MNIST	**91.4±1.0**	73.5±1.8	39.9±1.9	44.0±1.2	55.2±1.4	43.8±1.6	61.1±1.7	13.8±1.0	41.1±1.6
	Yale	3.0±4.0	3.0±4.0	15.0±11.2	**27.7±13.3**	24.1±9.5	24.1±13.3	24.7±11.6	7.8±5.3	7.2±6.0
	Average	**52.9**	47.3	37.4	39.4	40.5	33.0	41.0	22.1	32.5

Table 12
The Pvalues of the Wilcoxon test in the real datasets.

Classifier	Dataset	Raw	MF	TIEF	FIEF	NRE-FS	FDEF	SDF-FS	EUF
KNN	MF	**<0.01**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	PSMTSR	0.1855	**<0.01**	**<0.05**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
	GSALC	0.3990	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	Scene15	0.6152	**<0.001**	1.0000	1.0000	1.0000	**<0.001**	**<0.001**	1.0000
	MNIST	**<0.01**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	Yale	0.9969	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**	**<0.01**
SVM	MF	0.9716	0.5431	0.7748	0.7146	0.1714	**<0.05**	**<0.001**	0.9875
	PSMTSR	0.9853	**<0.05**	**<0.05**	**<0.05**	**<0.05**	**<0.01**	**<0.01**	**<0.05**
	GSALC	0.9974	**<0.05**	0.2420	0.2420	0.2420	**<0.1**	**<0.05**	0.2420
	Scene15	**<0.001**	1.0000	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	MNIST	0.9971	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	Yale	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
Logistic regression	MF	0.9951	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	PSMTSR	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	GSALC	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	Scene15	**<0.05**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**	**<0.001**
	MNIST	/	**<0.001**	0.5237	0.1468	**<0.1**	0.1875	**<0.001**	**<0.001**
	Yale	/	0.9953	1.0000	1.0000	0.9968	1.0000	0.9903	0.8565

Similarly, according to the Logistic Regression classifier, statistical significance underscores the superiority of the CF method over other techniques and raw data within datasets, including PSMTSR, GSALC, and Scene15. And in the MF and MNIST datasets, the CF method statistically outperforms the most approaches, such as the MF, NRE-FS, SDF-FS, and EUF methods. In general, in the majority of cases, the CF method significantly outperforms other methods as well as the raw data.

Table 13
The results of the Nemenyi test in the real datasets.

Algorithms	Average ranking			Pvalues		
	KNN	SVM	LR	KNN	SVM	LR
CF	**7.2500**	**8.4167**	**7.4167**			
Raw	**7.2500**	**8.4167**	7.0833	1.0000	1.0000	1.0000
MF	4.4167	4.0000	4.2500	0.6877	0.1172	0.5413
TIEF	6.5000	5.7500	5.9167	0.9999	0.7551	0.9901
FIEF	5.4167	5.1667	5.1667	0.9647	0.5042	0.8893
NRE-FS	4.0833	3.5000	4.3333	0.5413	$<$**0.05**	0.5786
FDEF	3.6667	4.0833	4.7500	0.3628	0.1337	0.7551
SDF-FS	1.5000	2.0833	2.0000	$<$**0.01**	$<$**0.01**	$<$**0.05**
EUF	4.9167	3.5833	4.0833	0.8671	$<$**0.1**	0.4676
$CD_{0.1} = 4.6169$	$CD_{0.05} = 5.0027$		$CD_{0.01} = 5.7648$		$CD_{0.001} = 6.6787$	

For various evaluation metrics, the Nemenyi test is also employed to assess the superiority of the proposed CF method compared to other approaches in the real datasets. The results of the Nemenyi test are presented in Table 13. From the perspective of average ranking, the findings indicate that for the Logistic Regression classifiers, the CF method achieves the highest average ranking compared to other algorithms and raw data. In the case of the KNN and SVM classifiers, the CF method secures a higher average ranking than the other seven methods. Additionally, considering Pvalues, for the KNN classifier, the CF method significantly outperforms the SDF-FS methods. Similarly, within the SVM classifier, the CF method demonstrates clear superiority over the NRE-FS, SDF-FS, and EUF methods. For the Logistic Regression classifier, the CF method presents significant advantages over the SDF-FS method. In summary, when considering average ranking, the CF method displays a certain advantage. However, based on the Pvalues, the CF method only significantly outperforms a limited number of methods such as NRE-FS, SDF-FS, and EUF.

5. Conclusion

In practical applications, extracting features from multi-source data is crucial. Information fusion techniques can assist in handling complex multi-source information systems and extracting valuable information. This paper proposes an unsupervised feature extraction and fusion method based on the R-Vine copula. Initially, kernel density estimation is employed to extract each information source's marginal density and distribution. Next, a vine structure for each attribute is generated, and the corresponding copula functions are estimated. Utilizing the relevant vine structure, the joint probability density of each attribute across all information sources can be determined, serving as the final fusion feature. Extensive experiments on both simulated and real datasets show that the proposed method can significantly enhance the performance of popular classifiers like KNN, SVM, and Logistic Regression compared to various state-of-the-art fusion methods and raw information sources.

However, the fusion method proposed in this paper has certain limitations. Firstly, although copula theory proves efficient in delineating data interdependencies across diverse sources, its efficacy diminishes notably in high-dimensional environments or when confronted with noisy data sources, prevalent in real-world contexts. In the future, it is meaningful to investigate the integration of filtering techniques and feature selection techniques with copula theory to enhance the robustness and accuracy of data analysis. Secondly, this study introduces a copula-based fusion method designed explicitly for multi-source homogeneous data without considering heterogeneous data. Thus, its application scope is limited to homogeneous data situations. In the future, it would be worth exploring the application of the R-Vine copula method for the fusion of multi-source heterogeneous data.

CRediT authorship contribution statement

Xiuwei Chen: Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Li Lai:** Validation, Resources, Methodology. **Maokang Luo:** Validation, Supervision, Project administration.

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.

References

[1] P. Zhang, T. Li, G. Wang, C. Luo, H. Chen, J. Zhang, D. Wang, Z. Yu, Multi-source information fusion based on rough set theory: a review, Inf. Fusion 68 (2021) 85–117.

[2] W. Wei, J. Liang, Information fusion in rough set theory: an overview, Inf. Fusion 48 (2019) 107–118.
[3] R. Li, M. Zhou, D. Zhang, Y. Yan, Q. Huo, A survey of multi-source image fusion, Multimed. Tools Appl. 83 (2024) 18573–18605.
[4] A. Chen, X. Tang, B. Cheng, J. He, Multi-source monitoring information fusion method for dam health diagnosis based on Wasserstein distance, Inf. Sci. 632 (2023) 378–389.
[5] P. Zhang, T. Li, G. Wang, D. Wang, P. Lai, F. Zhang, A multi-source information fusion model for outlier detection, Inf. Fusion 93 (2023) 192–208.
[6] R. Nie, J. Cao, D. Zhou, W. Qian, Multi-source information exchange encoding with pcnn for medical image fusion, IEEE Trans. Circuits Syst. Video Technol. 31 (3) (2021) 986–1000.
[7] O.H. Salman, M.F.A. Rasid, M.I. Saripan, S.K. Subramaniam, Multi-sources data fusion framework for remote triage prioritization in telehealth, J. Med. Syst. 38 (2014) 103.
[8] Y. Xiao, X. Li, J. Yin, W. Liang, Y. Hu, Adaptive multi-source data fusion vessel trajectory prediction model for intelligent maritime traffic, Knowl.-Based Syst. 277 (2023) 110799.
[9] Z. Chen, Q. Wang, Q. He, T. Yu, M. Zhang, P. Wang, Cufuse: camera and ultrasound data fusion for rail defect detection, IEEE Trans. Intell. Transp. Syst. 23 (11) (2022) 21971–21983.
[10] P. Zhang, T. Li, Z. Yuan, C. Luo, G. Wang, J. Liu, S. Du, A data-level fusion model for unsupervised attribute selection in multi-source homogeneous data, Inf. Fusion 80 (2022) 87–103.
[11] K. Cai, W. Xu, An efficient multi-source information fusion approach for dynamic interval-valued data via fuzzy approximate conditional entropy, Int. J. Mach. Learn. Cybern. 15 (2024) 3619–3645.
[12] X. Gong, Y. Dong, T. Zhang, Codf-net: coordinated-representation decision fusion network for emotion recognition with eeg and eye movement signals, Int. J. Mach. Learn. Cybern. 15 (2024) 1213–1226.
[13] Z. Liu, Q. Pan, J. Dezert, J.-W. Han, Y. He, Classifier fusion with contextual reliability evaluation, IEEE Trans. Cybern. 48 (5) (2018) 1605–1618.
[14] H. Jian, Y. Zhang, W. Gao, B. Wang, G. Wang, Dual-branch feature fusion dehazing network via multispectral channel attention, Int. J. Mach. Learn. Cybern. 15 (2024) 2655–2671.
[15] S. Huang, X. Wu, Y. Yang, W. Wan, X. Wang, A dual-encoder network based on multi-layer feature fusion for infrared and visible image fusion, Int. J. Mach. Learn. Cybern. (2024).
[16] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.
[17] X. Zhang, X. Chen, W. Xu, W. Ding, Dynamic information fusion in multi-source incomplete interval-valued information system with variation of information sources and attributes, Inf. Sci. 608 (2022) 1–27.
[18] W. Xu, Y. Pan, X. Chen, W. Ding, Y. Qian, A novel dynamic fusion approach using information entropy for interval-valued ordered datasets, IEEE Trans. Big Data 9 (3) (2023) 845–859.
[19] W. Xu, M. Li, X. Wang, Information fusion based on information entropy in fuzzy multi-source incomplete information system, Int. J. Fuzzy Syst. 19 (2017) 1200–1216.
[20] H. Joe, Dependence Modeling with Copulas, CRC Press, 2014.
[21] G. Andrew, R. Arora, J. Bilmes, K. Livescu, Deep canonical correlation analysis, in: International Conference on Machine Learning, PMLR, 2013, pp. 1247–1255.
[22] Y. Fang, L. Madsen, Modified Gaussian pseudo-copula: applications in insurance and finance, Insur. Math. Econ. 53 (1) (2013) 292–301.
[23] U. Cherubini, E. Luciano, W. Vecchiato, Copula Methods in Finance, John Wiley & Sons, 2004.
[24] X. Ren, Y. Tian, S. Li, Vine copula-based dependence description for multivariate multimode process monitoring, Ind. Eng. Chem. Res. 54 (41) (2015) 10001–10019.
[25] Q. Cui, S. Li, Process monitoring method based on correlation variable classification and vine copula, Can. J. Chem. Eng. 98 (6) (2020) 1411–1428.
[26] J. Su, E. Furman, Multiple risk factor dependence structures: copulas and related properties, Insur. Math. Econ. 74 (2017) 109–121.
[27] R. Jammazi, J.C. Reboredo, Dependence and risk management in oil and stock markets. A wavelet-copula analysis, Energy 107 (2016) 866–888.
[28] T. Bedford, R.M. Cooke, Probability density decomposition for conditionally dependent random variables modeled by Vines, Ann. Math. Artif. Intell. 32 (2001) 245–268.
[29] T. Bedford, R.M. Cooke, Vines: a new graphical model for dependent random variables, Ann. Stat. 30 (4) (2002) 1031–1068.
[30] K. Aas, C. Czado, A. Frigessi, H. Bakken, Pair-copula constructions of multiple dependence, Insur. Math. Econ. 44 (2) (2009) 182–198.
[31] L. Wasserman, All of Nonparametric Statistics, Springer Science & Business Media, 2006.
[32] J. Dissmann, E.C. Brechmann, C. Czado, D. Kurowicka, Selecting and estimating regular vine copulae and application to financial returns, Comput. Stat. Data Anal. 59 (2013) 52–69.
[33] H. Akaike, Information Theory and an Extension of the Maximum Likelihood Principle, Springer, New York, New York, NY, 1998, pp. 199–213.
[34] R.B. Nelsen, An introduction to copulas, 2006.
[35] X. Chen, W. Xu, Double-quantitative multigranulation rough fuzzy set based on logical operations in multi-source decision systems, Int. J. Mach. Learn. Cybern. 13 (4) (2022) 1021–1048.
[36] W. Qian, S. Yu, J. Yang, Y. Wang, J. Zhang, Multi-label feature selection based on information entropy fusion in multi-source decision system, Evol. Intell. 13 (2020) 255–268.
[37] X. Chen, M. Luo, Incremental information fusion in the presence of object variations for incomplete interval-valued data based on information entropy, Inf. Sci. (2024) 120479.
[38] B. Sang, L. Yang, H. Chen, W. Xu, Y. Guo, Z. Yuan, Generalized multi-granulation double-quantitative decision-theoretic rough set of multi-source information system, Int. J. Approx. Reason. 115 (2019) 157–179.
[39] J. Demiar, D. Schuurmans, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (1) (2006) 1–30.
[40] R. Duin, Multiple features, UCI machine learning repository, https://doi.org/10.24432/C5HC70, 1998.
[41] N. Ali, B. Zafar, 15-scene image dataset, https://doi.org/10.6084/m9.figshare.7007177.v1, 2018.
[42] F. Tian, L. Zhao, S. Deng, Gas sensor array low-concentration, UCI Machine Learning Repository, https://doi.org/10.24432/C5CK6F, 2024.
[43] O. Kursun, B. Sakar, M. Isenkul, C. Sakar, A. Sertbas, F. Gurgen, Parkinson's speech with multiple types of sound recordings, UCI Machine Learning, Repository, https://doi.org/10.24432/C5NC8M, 2014.
[44] L. Deng, The mnist database of handwritten digit images for machine learning research [best of the web], IEEE Signal Process. Mag. 29 (6) (2012) 141–142.
[45] M.S. Chen, C.D. Wang, J.H. Lai, Low-rank tensor based proximity learning for multi-view clustering, IEEE Trans. Knowl. Data Eng. (2023).