Label-Aware Causal Feature Selection

Zhaolong Ling, Jingxuan Wu, Yiwen Zhang, Peng Zhou, *Senior Member, IEEE*, Xingyu Wu, Kui Yu, *Member, IEEE*, and Xindong Wu, *Fellow, IEEE*

***Abstract*—Causal feature selection has recently received increasing attention in machine learning and data mining, especially in the era of Big Data. Existing causal feature selection algorithms select unique causal features of the single class label as the optimal feature subset. However, a single class label usually has multiple classes, and it is unreasonable to select the same causal features for different classes of a single class label. To address this problem, we employ the class-specific mutual information to evaluate the causal information carried by each class of the single class label, and theoretically analyze the unique relationship between each class and the causal features. Based on this, a Label-aware Causal Feature Selection algorithm (LaCFS) is proposed to identifies the causal features for each class of the class label. Specifically, LaCFS uses the pairwise comparisons of class-specific mutual information and the size of class-specific mutual information values from the perspective of each class, and follows a divide-and-conquer framework to find causal features. The correctness and application condition of LaCFS are theoretically proved, and extensive experiments are conducted to demonstrate the efficiency and superiority of LaCFS compared to the state-of-the-art approaches.**

***Index Terms*—Bayesian network, causal feature selection, class-specific mutual information, Markov blanket.**

I. Introduction

FEATURE selection is a critical step in high-dimensional data analysis, which aims to identify features that are essential for predictive models [1], [2]. However, traditional feature selection methods ignore the potential causal relationships between features and the class label, which can lead to selection bias and spurious correlations [3]. To build more interpretable and robust predictive models, causal feature selection has emerged, which aims to learn the Markov blanket (MB) of the class label [4], [5], [6]. Under the faithfulness assumption, the MB of a target variable in a Bayesian network (BN) consists of its parents (direct causes), children (direct effects), and spouses (other parents of these children) [7], [8]. The MB of a class label indicates the local causal relationship between the class label and the features in its MB [9], [10]. Furthermore, since all other variables are probabilistically independent of the class label conditioning on its MB, the MB of a class label is the optimal feature subset for classification [11], [12].

In real-world applications, when the class label is as the cause, its different values may correspond to different treatment options or interventions, and understanding the information corresponding to each class value can help identify the causal features associated with it a particular treatment option [13]; when it is as an effect, its different values may correspond to different expected goals, and knowing the in-formation corresponding to each class value can filter out the predicted features for each expected goal [14], [15]. At the same time, considering the information of a particular class further extends the interpretability of causal feature selection, i.e., being able to interpret features that have predictive power in the context of a particular intervention or a particular goal [16], [17]. It should be clear that the research discussion is not about multi-label feature selection. Multi-label feature selection focuses on processing multiple class labels at the same time, whereas this study only focuses on different classes of the single class label. For example, in the medical domain, multi-label causal feature selection addresses multi-label datasets, allowing for feature selection across multiple class labels, such as “patient type” and “treatment plan”. However, this study focuses on single-label datasets, exemplified by datasets with only one class label, “disease status”, which has two attribute values: “diseased” and “not diseased”. The corresponding causal features are selected for the attribute values of the single class label, “diseased” and “not diseased”. However, existing algorithms [5], [18] use the MB of the class label as the optimal feature subset for each class. Thus, it is an interesting challenge to consider the causal features for each class of the class label, and further provide an accurate classification of the corresponding class.

To battle this challenge, we try to identify the causal features of each class in a class label by introducing class-specific mutual information. Class-specific mutual information focuses on a specific class of a variable, and measures the relationship between it and other variables by considering the information contained in this class [19], [20]. Meanwhile, owing to the probabilistic correlation of class-specific mutual information,

Received 30 May 2024; revised 13 November 2024; accepted 22 December 2024. Date of publication 25 December 2024; date of current version 5 February 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2021ZD0111801, in part by the National Natural Science Foundation of China under Grant 62306002, Grant 62272001, Grant 62176001, Grant 62376087, and Grant 62120106008, and in part by the Natural Science Project of Anhui Provincial Education Department under Grant 2023AH030004, and in part by Xunfei Zhiyuan Digital Transformation Innovation Research Special for Universities under Grant 2023ZY001. Recommended for acceptance by R. Akbarinia. *(Corresponding author: Peng Zhou.)*

Zhaolong Ling, Jingxuan Wu, Yiwen Zhang, and Peng Zhou are with the School of Computer Science and Technology, Anhui University, Hefei 230601, China (e-mail: zlling@ahu.edu.cn; ahu_wujingxuan@163.com; zhangyiwen@ahu.edu.cn; zhoupeng@ahu.edu.cn).

Xingyu Wu is with the Department of Computing, School of Hong Kong Polytechnic University, Hong Kong 999077, China (e-mail: xingy.wu@polyu.edu.hk).

Kui Yu and Xindong Wu are with the Key Laboratory of Knowledge Engineering with Big Data (the Ministry of Education of China), School of Computer Science and Information Technology, Hefei University of Technology, Hefei 230009, China (e-mail: yukui@hfut.edu.cn; xwu@hfut.edu.cn).

Digital Object Identifier 10.1109/TKDE.2024.3522580

it can retain the features relevant to a specific class and remove the redundant and irrelevant features [21], [22].

Thus, in this paper we propose a Label-aware Causal Feature Selection algorithm, called LaCFS. LaCFS identifies the unique relationships between each class in the class label and its causal features by evaluating the class-specific mutual information. Specifically, LaCFS instantiates this unique relationship in several steps and follows the divide-and-conquer framework to respectively learn the parent-child variables and spouse variables of a specific class. In this way, LaCFS can retrieve most of the specific MB variables for each class and take them as the optimal feature subset, instead of selecting the MB for the entire class label as existing causal feature selection algorithms perform. The main contributions of this paper are summarised as follows:

- We formally analyze the characteristics of class-specific mutual information relationships between each class and its causal features, which inspires us to design a strategy to distinguish causal and non-causal features.
- The proposed LaCFS algorithm learns the MB for each class by measuring the class-specific mutual information, achieving a more accurate manner. We prove the correctness and application condition of LaCFS to facilitate the theoretical guarantee.
- We conduct the experiments on five benchmark BNs and eight real-world datasets to show that LaCFS has comparable efficiency but achieves higher accuracy than the six state-of-the-art causal feature selection algorithms.

The rest of the paper is organized as follows. Section II reviews existing causal feature selection algorithms; Section III introduces the basic definitions and notations; and Section IV describes the class-specific mutual information available to identify the MB of each class in a class label. In Section V, we discuss the results of the experiments and explain the associated outcomes. Finally, Section VI concludes the paper.

II. Related Work

Recently, many causal feature selection algorithms have been proposed. Moreover, owing to the MB of a class label being the optimal and minimal feature subset with maximum classification predictive property [11], [12], causal feature selection methods based on the MB have received increasing attention from researchers [18], [23], [24], [25].

Existing causal feature selection algorithms can be broadly classified into two categories: simultaneous and divide-and-conquer algorithms [18]. The simultaneous algorithms use the currently selected Markov blanket as the condition set for the conditional independence test. Koller et al. [4] proposed the KS algorithm, finding the Markov blanket by minimizing the cross-entropy loss, but without theoretical guarantees of the soundness of the algorithm. GS [26] was the first sound algorithm, and its framework with a growth phase and a contraction phase has become the basic strategy for subsequent algorithms. Tsamardinos et al. [27] proposed the IAMB algorithm, which improves GS by reordering the variables in each iteration; later, IAMB variants (e.g., inter-IAMB [27] and FBED [28]) were successively proposed. These methods are efficient, but the number of samples required is exponentially positively related to the number of nodes in the Markov blanket, i.e., simultaneous Markov blanket learning methods can lead to performance degradation when data samples are insufficient [29], [30].

To solve this problem, the divide-and-conquer algorithm is proposed. It divides the Markov blanket learning process into PC (parents and children) discovery and spouses discovery to reduce the data sample requirement by reducing the set of conditions. MMMB [31] and HITON-MB [32] were two early algorithms that removed erroneous parent-child variables as early as possible by intersecting the growth and contraction phases in the PC discovery process and storing the separating set of variables that are independent of the target variable, and identifying the independent variables as spouses if they are dependent on the target variable given the concatenation of the separation set with the target variable's children. PCMB [29] and IPCMB [33] removed non-children variables that cannot be removed by MMMB and HITON-MB by adding a double-check policy. Gao et al. [34] discovered the coexistence property of spouse and error parent-child variables and proposed a relatively efficient algorithm, STMB. Wu et al. [35] theoretically analyzed that the PCMasking phenomenon can interfere with the identification of variables and proposed the CCMB algorithm to improve the accuracy of Markov blanket learning. BAMB [36] and EEMB [37] balanced learning efficiency and accuracy by unifying PC and spouses discovery in one loop. Recently, a new learning framework, CFS [38], was proposed, and CFS improved the efficiency of Markov blanket discovery to some extent by optimizing the order of spouse discovery.

In summary, existing causal feature selection algorithms optimize the efficiency and accuracy of MB discovery. However, these algorithms unreasonable select the same causal features for different classes. Thus, to avoid the unreasonableness of these causal feature selection algorithms, this paper proposes a label-aware causal feature selection algorithm that selects the causal features of each class in the class label based on class-specific mutual information.

III. Definitions and Notations

In this section, we will describe basic definitions, theorems, and the notations used in this paper. With regard to notation, specifically, the capital letters (such as X, Y) denote random variables, the lower-case letters (such as x, y) denote their values, and the capital bold letters (such as **U**, **S**) represent the set of variables. Specifically, let **U** denote the set of all (discrete random) variables, C denote the class label, C_i represent the i-th class of the class label, and X, etc. denote random variables.

A. Bayesian Networks

In the next, we present some basic concepts and theorems of the BN network.

Definition 1 (D-separation) [39]: Given $\mathbf{S} \subseteq \mathbf{U} \setminus \{X, Y\}$, the path π between X and Y is open if and only if (1) every collider on π is in $\mathbf{S}$ or its descendant in $\mathbf{S}$, and (2) no other non-colliders on π are in $\mathbf{S}$. Otherwise, the path π is blocked. If every path between X and Y is blocked by $\mathbf{S}$, X and Y are conditionally

independent given $\mathbf{S}$, and $\mathbf{S}$ is called the separating set of X and Y.

Definition 1 can be used to determine whether two variables are conditionally independent.

Theorem 1 [39]: In a faithful BN, given variables $X, Y \in \mathbf{U}$, if there is an edge between X and Y, for $\forall \mathbf{S} \subseteq \mathbf{U} \setminus \{X, Y\}$, $X \not\perp\!\!\!\perp Y|\mathbf{S}$.

Theorem 1 shows that given $\forall \mathbf{S} \subseteq \mathbf{U} \backslash \{X, Y\}$, variable X and variable Y ($Y \in \mathbf{PC}_X$) are conditionally dependent. Here, $\mathbf{PC}_X$ denotes the parents and children of X, which are the variables directly connected to X.

Theorem 2 [40]: In a faithful BN, for three variables X is adjacent to Y, Y is adjacent to Z, and X is not adjacent to Z, $\exists \mathbf{S} \subseteq \mathbf{U} \backslash \{X, Y, Z\}$, if $X \perp\!\!\!\perp Z|\mathbf{S}$ and $X \not\perp\!\!\!\perp Z|\mathbf{S} \cup Y$, then Z is a spouse of X.

Theorem 2 indicates that a variable is conditionally dependent with its spouse given the common child.

B. Class-Specific Mutual Information

In the following, we introduce concepts related to class-specific information measures.

Definition 2 [19]: The mutual information between the specific class C_i and the variable X is defined as follows, denoted as $I(C_i; X)$:

$$I(C_i; X) = \sum_x p(c_i, x) \log \frac{p(c_i, x)}{p(c_i)p(x)} \tag{1}$$

the mutual information between variables X and Y in the context of the specific class C_i is defined as follows,denoted as $I_{C_i}(X; Y)$:

$$I_{C_i}(X; Y) = \sum_x \sum_y p(c_i, x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{2}$$

Eq. (1) measures the degree of correlation between the specific class C_i and the variable X, and (2) measures the degree of correlation between X and Y, in the case of the specific class C_i.

Definition 3 [19]: Given the variable Y, the conditional mutual information between the specific class C_i and X is defined as follows, denoted $I(C_i; X|Y)$:

$$I(C_i; X|Y) = \sum_x \sum_y p(c_i, x, y) \log \frac{p(c_i, x|y)}{p(c_i|y)p(x|y)} \tag{3}$$

Definition 4 [41]: The three-way interaction information between the specific class C_i and variables X, Y is defined as follows, denoted as $I(C_i; X; Y)$:

$$I(C_i; X; Y) = \sum_x \sum_y p(c_i, x, y) \log \frac{p(c_i|x)p(c_i|y)p(x, y)}{p(c_i)p(c_i, x, y)} \tag{4}$$

The equation described in Definition 3 measures the degree of correlation between the specific class C_i and variable X given variable Y. The equation described in Definition 4 measures the information shared by the specific class C_i, variable X, and variable Y.

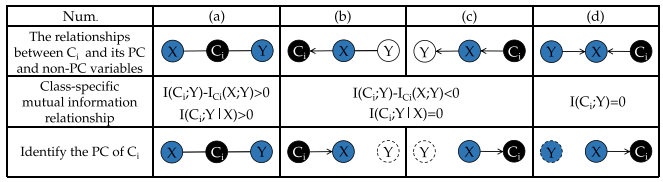


Fig. 1. The possible relationships between C_i (a specific class in the class label C) and its PC and non-PC variables.

IV. Label-Aware Causal Feature Selection Algorithm

In this section, we first use class-specific mutual information to discover the causal features of a specific class in the class label, to propose a label-aware causal feature selection algorithm, and then we analyze the correctness and application conditions of the algorithm.

A. Algorithm Implementation

In the following, we employ a divide-and-conquer strategy and use class-specific mutual information to identify PC and spouses of a specific class in the class variable, respectively.

1) The PC of a specific class in the class label: We use the relationship that the class-specific mutual information between a specific class and its non-PC variables is less than that between its PC variable and its non-PC variables to find the PC. Moreover, the property of the class-specific conditional mutual information between a specific class and its non-PC variables equal to zero, given its PC variable, has been used in the PC discovery. Specifically, the PC of a specific class is determined by the following steps:

- *Step 1:* $\forall X \in \mathbf{U} \setminus C$, if the variable X satisfies $I(C_i; X) gt; 0$, then include the variable X in the set $\mathbf{PC}_i$ and sort it in descending order according to the value of $I(C_i; X)$.
- *Step 2:* Select a variable X in order from the set $\mathbf{PC}_i$, if $\exists Y \in \mathbf{PC}_i \setminus X$ such that $I(C_i; Y) - I_{C_i}(X; Y) lt; 0$ or $I(C_i; Y|X) = 0$ holds, then remove variable Y, until every variable in the set $\mathbf{PC}_i$ has been selected.

Next, we will use existing definitions, theorems, etc., to prove in detail that the PC of a specific class can be found after the above two steps.

In Fig. 1(a), the variables X and Y are either the father or the child of C_i, by Theorem 1: $C_i \not\perp\!\!\!\perp X|\emptyset$ and $C_i \not\perp\!\!\!\perp Y|\emptyset$. Next, we propose Proposition 1 to obtain the class-specific mutual information value size based on the dependencies between C_i and variables. Furthermore, Proposition 1 is a tool for selecting a variable with the class-specific mutual information value greater than 0 with C_i in Step 1.

Proposition 1: The following inequality holds:

(1) $I(C_i; X) \geq 0$, only if $p(c_i, x) = p(c_i)p(x)$, $I(C_i; X) = 0$.

(2) $I(C_i; X|Y) \geq 0$, only if $p(c_i, x|y) = p(c_i|y)p(x|y)$, $I(C_i; X|Y) = 0$.

Proof: (1) According to (1), we can obtain: $I(C_i; X) = \sum_x p(c_i, x) \frac{p(c_i, x)}{p(c_i)p(x)}$. Based on the Log-sum inequality [42], it

further follows that:

$$\sum_x p(c_i,x)\frac{p(c_i,x)}{p(c_i)p(x)} \geq \sum_x p(c_i,x) \times \log \frac{\sum_x p(c_i,x)}{\sum_x p(c_i)p(x)} \tag{5}$$

and the equality sign holds when $p(c_i,x) = p(c_i)p(x)$.

$$\sum_x p(c_i,x) \times \log \frac{\sum_x p(c_i,x)}{\sum_x p(c_i)p(x)}$$

$$= p(c_i) \times \log \frac{p(c_i)}{p(c_i)} = 0 \tag{6}$$

Thus, based on (5) and (6), $I(C_i;X) \geq 0$, only if $p(c_i,x) = p(c_i)p(x)$, $I(C_i;X) = 0$.

(2) According to (3), we can obtain: $I(C_i;X|Y) = \sum_x \sum_y p(c_i,x,y)\frac{p(c_i,x|y)}{p(c_i|y)p(x|y)}$. Based on the Log-sum inequality, it further follows that:

$$\sum_x \sum_y p(c_i,x,y)\frac{p(c_i,x|y)}{p(c_i|y)p(x|y)}$$

$$\geq \sum_x \sum_y p(c_i,x,y) \times \log \frac{\sum_x \sum_y p(c_i,x|y)}{\sum_x \sum_y p(c_i|y)p(x|y)} \tag{7}$$

and the equality sign holds when $p(c_i,x|y) = p(c_i|y)p(x|y)$.

$$\sum_x \sum_y p(c_i,x,y) \times \log \frac{\sum_x \sum_y p(c_i,x|y)}{\sum_x \sum_y p(c_i|y)p(x|y)}$$

$$= p(c_i) \times \log \frac{\sum_y p(c_i|y)}{\sum_y p(c_i|y)} = 0 \tag{8}$$

Thus, in light of (7) and (8), $I(C_i;X|Y) \geq 0$, only if $p(c_i,x|y) = p(c_i|y)p(x|y)$, $I(C_i;X|Y) = 0$. ■

From Proposition 1, we can obtain: $I(C_i;X)gt;0$ and $I(C_i;Y)gt;0$. Due to the inability to determine the size relationship between $I(C_i;X)$ and $I(C_i;Y)$, we cannot know their order in the set $\mathbf{PC}_i$. Thus, after Step 1, variables X and Y will be included in the set $\mathbf{PC}_i$, but the order before and after cannot be known.

Next, we divide the relationship between variables X and Y and C_i into two cases. Case 1: when the variables X and Y are both parents of C_i, $X \perp\!\!\!\perp Y|\emptyset$, i.e., $p(x)p(y) = p(x,y)$, combined with (2), it follows that $I_{C_i}(X;Y) = 0$. Then $I(C_i;Y) - I_{C_i}(X;Y)gt;0$ and $I(C_i;X) - I_{C_i}(X;Y)gt;0$ hold. Moreover, $C_i \not\perp\!\!\!\perp X|Y$, $C_i \not\perp\!\!\!\perp Y|X$, and by Proposition 1, we get: $I(C_i;X|Y)gt;0, I(C_i;Y|X)gt;0$. Thus, in Case 1, the variables X and Y are retained in the set $\mathbf{PC}_i$ after Step 2.

Case 2: when the variables X and Y are not both parents of C_i, we can not directly derive the size of $I(C_i;X) - I_{C_i}(X;Y)$, so we propose Proposition 2 to indirectly determine the size between two class-specific mutual information. More importantly, Proposition 2 is a tool for excluding non-PC variables in Step 2 of PC discovery.

Proposition 2: The following chain rule holds:

(1) $I(C_i;X;Y) = I(C_i;X) - I(C_i;X|Y)$

(2) $I(C_i;X;Y) = I(C_i;Y) - I(C_i;Y|X)$

(3) $I(C_i;X;Y) = I_{C_i}(X;Y) - I(X;Y|C_i)$

Proof: (1):

According to (1) and (3), we can obtain:

$$I(C_i;X) - I(C_i;X|Y)$$

$$= \sum_x p(c_i,x)\log\frac{p(c_i,x)}{p(c_i)p(x)}$$

$$- \sum_x \sum_y p(c_i,x,y)\log\frac{p(c_i,x|y)}{p(c_i|y)p(x|y)}$$

$$= \sum_x \sum_y p(c_i,x,y)\log\frac{p(c_i,x)}{p(c)p(x)}$$

$$- \sum_x \sum_y (c_i,x,y)\log\frac{p(c_i,x|y)}{p(c_i|y)p(x|y)}$$

$$= \sum_x \sum_y p(c_i,x,y)\log\frac{p(c_i|x)p(c|y)p(x,y)}{p(c_i)p(c_i,x,y)} \tag{9}$$

Then, based on (4) and (9), we obtain: $I(C_i;X) - I(C_i;X|Y) = I(C_i;X;Y)$.

The proofs of (2) and (3) follow by the same token. ■

In accordance with Proposition 2: $I(C_i;X) - I_{C_i}(X;Y) = I(C_i;X|Y) - I(X;Y|C_i)$. And as $X \perp\!\!\!\perp Y|C_i$ and $C_i \not\perp\!\!\!\perp X|Y$, $I(X;Y|C_i) = 0$ and $I(C_i;X|Y)gt;0$. It is easy to know: $I(C_i;X) - I_{C_i}(X;Y)gt;0$. Similarly $I(C_i;Y) - I_{C_i}(X;Y)gt;0$ and $I(C_i;Y|X)gt;0$ hold. Thus, in Case 2, the variables X and Y are retained in the set $\mathbf{PC}_i$ after Step 2.

In Fig. 1(b) and (c), the variable X is the father of C_i, and Y is the grandfather, brother, or descendant of C_i. According to Definition 1 and Theorem 1: $C_i \not\perp\!\!\!\perp X|\emptyset$, $C_i \not\perp\!\!\!\perp Y|\emptyset$, $I(C_i;X)gt;0$, $I(C_i;Y)gt;0$. By the chain rule of Proposition 2, we find that $I(C_i;X) - I(C_i;Y) = I(C_i;X|Y) - I(C_i;Y|X)$, and from $I(C_i;X|Y)gt;0$ and $I(C_i;Y|X) = 0$, we know that $I(C_i;X) - I(C_i;Y)gt;0$. Thus, after Step 1, both variables X and Y are included in the set $\mathbf{PC}_i$, and after sorting them in descending order, variable X is ranked before Y. Using the chain rule of Proposition 2, we can convert $I(C_i;Y) - I_{C_i}(X;Y)$ into $I(C_i;Y|X) - I(X;Y|C_i)$, and from $I(C_i;Y|X) = 0$ and $I(X;Y|C_i)gt;0$, it follows that $I(C_i;Y) - I_{C_i}(X;Y)lt;0$. Thus, after Step 2, the variable X is still in the set $\mathbf{PC}_i$, and the variable Y is removed.

In Fig. 1(d), the variable X is the child of C_i, Y is the spouse of C_i with respect to X. By Definition 1: $C_i \perp\!\!\!\perp Y|\emptyset$, so $I(C_i;Y) = 0$. Thus, after Step 1, the variable Y will not enter the set $\mathbf{PC}_i$ and cannot enter Step 2.

In summary, we use two steps to instantiate the unique class-specific mutual information relationships between C_i and its PC to discover the PC of C_i. This process is realized in lines 1 to 14 of Algorithm 1. The reason for using Symmetric Uncertainty (SU) [43] lies in its normalization property, which allows for a fair comparison of the correlation among different features, thus avoiding inconsistencies in the numerical range of Mutual Information (I). Furthermore, Symmetric uncertainty SU preserves the symmetry of mutual information I. SU provides a more balanced evaluation basis by considering the symmetric

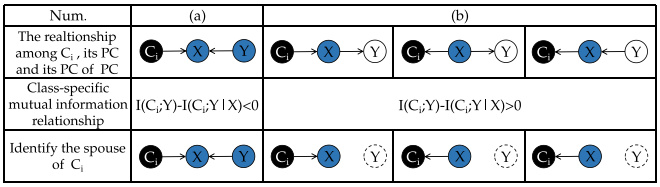


Fig. 2. Four possible cases between C_i (a specific class in the class label C), its PC (variable X), and the PC of its PC (variable Y).

Algorithm 1: DiscoverPC.

Input: $\mathcal{D}$: Data, C_i: Class label C of the i-th class;
Output: $\mathbf{PC}_i$: PC of C_i ;
1 $\mathbf{PC}_i = \emptyset$;
2 **for** each $X \in \mathbf{U} \backslash \{C\}$ **do**
3 **if** $SU(C_i; X) > \delta$ **then**
4 $\mathbf{PC}_i = \mathbf{PC}_i \cup \{X\}$ *in descending order*;
5 **end**
6 **end**
7 **for** each $X \in \mathbf{PC}_i$ **do**
8 **for** each $Y \in \mathbf{PC}_i \backslash \{X\}$ **do**
9 **if** $SU(C_i; X) - SU_{C_i}(X; Y) < 0 \parallel I(C_i; Y|X) < \delta$ **then**
10 $\mathbf{PC}_i = \mathbf{PC}_i \backslash \{Y\}$;
11 **end**
12 **end**
13 **end**
14 Return $\mathbf{PC}_i$;

relationship between features and class label. Thus, the use of the SU method enables a more comprehensive feature selection process and captures more accurately the true impact of features on class labels, ultimately improving classification performance. In line 3, the symbol δ denotes the threshold of the significance level, which is used to control the false positive rate. When the value of symmetric uncertainty is less than δ, it indicates that there is a significant conditional dependence between features and class label.

2) The spouses of a specific class in the class label: We use the relationship that the class-specific conditional mutual information between a specific class and its spouse given the empty set, is less than that between the specific class and its spouse given the corresponding common child variable, to determine the spouse of the specific class. Specifically, the spouses of a specific class are determined by the following step:

- $\forall X \in \mathbf{PC}_i$, if $\exists Y \in \mathbf{PC}_X$ (the parents and children of X) such that $I(C_i; Y) - I(C_i; Y|X) lt; 0$, then the variable Y is included in the spouses $\mathbf{SP}_i$ until every variable in $\mathbf{PC}_i$ has been selected.

Since the spouses of C_i only exist in the PC of each variable in the PC of C_i, we list the possible cases among C_i, the PC of C_i, and the PC of each variable in the PC of C_i in Fig. 2. Next, using the existing definitions and theorems, we show that the spouses of C_i can be found after the above step.

In Fig. 2(a), Y is the spouse of C_i with respect to X, by Theorems 1 and 2: $C_i \perp\!\!\!\perp Y|\emptyset$, $C_i \not\perp\!\!\!\perp Y|X$, and based on Propositions 1 and 2 we have: $I(C_i; Y) = 0$, $I(C_i; Y|X) gt; 0$. It is obvious:

$$I(C_i; Y) - I(C_i; Y|X) < 0 \tag{10}$$

Algorithm 2: LaCFS.

Input: $\mathcal{D}$: Data, C_i: Class label C of the i-th class;
Output: $[\mathbf{PC}_i, \mathbf{SP}_i]$: MB of C_i ;
1 $\mathbf{PC}_i = \text{DiscoverPC}(\mathcal{D}, C_i)$;
2 **for** each $X \in \mathbf{PC}_i$ **do**
3 $\mathbf{PC}_X = \text{FCBF}(\mathcal{D}, X)$;
4 **for** each $Y \in \mathbf{PC}_X$ **do**
5 **if** $I(C_i; Y) - I(C_i; Y|X) < 0$ **then**
6 $\mathbf{SP}_i = \mathbf{SP}_i \cup \{Y\}$;
7 **end**
8 **end**
9 **end**
10 Return $[\mathbf{PC}_i, \mathbf{SP}_i]$;

Thus, following (10), when the variable Y is the spouse of C_i, it will be included in the set $\mathbf{SP}_i$ after the above step.

In Fig. 2(b), Y is not the spouse of C_i with respect to X, in light of Definition 1: $C_i \not\perp\!\!\!\perp Y|\emptyset$, $C_i \perp\!\!\!\perp Y|X$, and then by Propositions 1 and 2 we have: $I(C_i; Y) gt; 0$, $I(C_i; Y|X) = 0$. It is easy to know:

$$I(C_i; Y) - I(C_i; Y|X) > 0 \tag{11}$$

Thus, according to (11), when the variable Y is not the spouse of C_i, it will not be included in the set $\mathbf{SP}_i$ after the above step.

In summary, given a common child, the class-specific conditional mutual information between C_i and its spouse is greater than class-specific mutual information between C_i and its spouse. Thus, after the above step, we can discover the spouse of C_i. In the third line of the Algorithm 2, FCBF [44] (Fast Correlation-Based Filter) is a feature selection method that rapidly identifies features related to class variables based on symmetric uncertainty. Specifically, the algorithm evaluates the relevance of features by calculating the symmetric uncertainty between the features and the target class, incorporating a redundancy elimination mechanism to retain features that are highly correlated with the class label while removing those that are redundant with other features. Through this process, FCBF can quickly filter out the parents and children of variable X (i.e., $\mathbf{PC}_X$), where X is the parent or child variable of the specific class C_i. Yu et al. [45] have theoretically demonstrated that FCBF can be applied to PC discovery. Compared to traditional PC discovery algorithms, the FCBF method evaluates the importance and redundancy of features through pairwise comparisons based on the correlation between features and the target class, rather than enumerating all possible conditional subsets for conditional independence testing as traditional PC algorithms do. This method significantly reduces computational complexity and avoids the combinatorial explosion problem in high-dimensional data. Consequently, FCBF can rapidly identify PC variables that make a significant contribution to feature selection, effectively mitigating the impact of redundant features, thereby markedly enhancing the efficiency and performance of the overall algorithm.

B. Algorithm Analysis

In this section, we analyze the output of LaCFS with maximum relevance and minimum redundancy and what conditions are better for handle a specific class of a class label.

Theorem 3: (Algorithm correctness): The output of LaCFS has maximum relevance and minimum redundancy with respect to a specific class of a class label.

Proof: In the proof, we use $\boldsymbol{MB}_i$ to represent the **MB** of C_i, $\mathbf{MB}(C)$ denotes the **MB** of C.

(1) maximum relevance: if $\forall \mathbf{S} \subseteq \mathbf{F}(\mathbf{F} = \mathbf{U} \setminus C)$, $I(C_i; \mathbf{MB}_i) \geq I(C_i; \mathbf{S})$ with equality if $\mathbf{MB}_i = \mathbf{S}$, the $\mathbf{MB}_i$ has the maximum relevance.

Case 1: $\forall \mathbf{S} \subseteq \mathbf{F} \setminus \mathbf{MB}_i$, depending on (3), we have: $I(C_i; \mathbf{S}|\mathbf{MB}_i) = \sum_{s,mb_i} p(c_i, s, mb_i) \log \frac{p(c_i,s|mb_i)}{p(c_i|s)p(s|mb_i)}$. As $p(c_i, s|mb_i) = p(c_i|s)p(s|mb_i)$, $I(C_i; \mathbf{S}|\mathbf{MB}_i) = 0$. By the chain rule of Proposition 2:

$$\begin{aligned} I(C_i; \mathbf{S}; \mathbf{MB}_i) &= I(C_i; \mathbf{MB}_i) + I(C_i; \mathbf{S}|\mathbf{MB}_i) \\ &= I(C_i; \mathbf{S}) + I(C_i; \mathbf{MB}_i|\mathbf{S}) \end{aligned} \tag{12}$$

In light of (12) and $I(C_i; \mathbf{S}|\mathbf{MB}_i) = 0$, $I(C_i; \mathbf{MB}_i) = I(C_i; \mathbf{S}) + I(C_i; \mathbf{MB}_i|\mathbf{S})$. By Proposition 1, we can get that $\forall \mathbf{S} \subseteq \mathbf{F} \setminus \mathbf{MB}_i$, $I(C_i; \mathbf{MB}_i) gt; I(C_i; \mathbf{S})$.

Case 2: $\forall \mathbf{S} \subseteq \mathbf{MB}_i$, and let $\mathbf{S}' = \mathbf{MB}_i \setminus \mathbf{S}$, based on (13), $I(C_i; \mathbf{MB}_i) \geq I(C_i; \mathbf{S})$ holds with equality if $\mathbf{S}$ equals to $\mathbf{MB}_i$.

$$\begin{aligned} &I(C_i; \mathbf{MB}_i) - I(C_i; \mathbf{S}) = I(C_i; \mathbf{S} \cup \mathbf{S}') - I(C_i; \mathbf{S}) \\ &= I(C_i; \mathbf{S}) + I(C_i; \mathbf{S}'|\mathbf{S}) - I(C_i; \mathbf{S}) = I(C_i; \mathbf{S}'|\mathbf{S}) \end{aligned} \tag{13}$$

Case 3: $\forall \mathbf{S}' \subset \mathbf{MB}_i$, and let $\mathbf{S}'' \subset \mathbf{F} \setminus \mathbf{MB}_i$, and $\mathbf{S} = \mathbf{S}' \cup \mathbf{S}''$, by (14), $I(C_i; \mathbf{S}|\mathbf{MB}_i) = 0$. Then according to (12) and Proposition 1, $I(C_i; \mathbf{MB}_i) gt; I(C_i; \mathbf{S})$.

$$\begin{aligned} &\frac{p(c_i, s|mb_i)}{p(c_i|mb_i)p(s|mb_i)} = \frac{p(c_i, s'', mb_i)}{p(c_i|mb_i)p(s'', mb_i)} \\ &= \frac{p(c_i|s'', mb_i)p(s'', mb_i)}{p(c_i|mb_i)p(s'', mb_i)} = 1 \end{aligned} \tag{14}$$

By Cases 1 to 3, $I(C_i; \mathbf{MB}_i) \geq I(C_i; \mathbf{S})$ with equality if $\mathbf{MB}_i = \mathbf{S}$. Thus the $\mathbf{MB}_i$ has the maximum relevance.

(2) minimum redundancy: $\forall\ \mathbf{MB}_i$, $\mathbf{MB}_i \subseteq \mathbf{MB}(C)$. Since the **MB** of the class variable is the optimal and minimal feature subset with maximum predictivity for classification [7], the $\mathbf{MB}(C)$ has the minimum redundancy. Thus, the $\mathbf{MB}_i$ has the minimum redundancy. ■

Theorem 4 (Application condition of the algorithm): Assume that the class label C has n class values and rank them in descending order according to their respective sample shares as $C_1, C_2, \ldots, C_n$, where the weight ratio of C_i in the whole sample is denoted by ω_i. At this point, $i \in \{1, 2, \ldots, n\}$ denotes the index of each class in the class label C. The classification accuracy is now required to be at least α under the ideal classifier. The weight ω_k of the specific class C_k must satisfy $\sum_{i=1}^{k-1} \omega_i \leq \alpha \leq \sum_{i=1}^{k} \omega_i$, where k denotes the index of the threshold class determined based on the sample weight ratio ω_i and the classification accuracy requirement α, which satisfies $1 \leq k \leq n$. Then LaCFS will better handle the specific class C_k than algorithms that directly optimize mutual information when $\frac{N\omega_k}{\xi(R_F^2 - R_F)} \geq 1$, where N represents the total number of samples, ξ is the number of samples on each degree of freedom required in the hypothesis testing, and R_F denotes the maximum value domain space of the features.

Proof: Since the G^2-statistic and mutual information have the following relationship [46]:

$$\frac{1}{2N} G^2(C; X|Y) = I(C; X|Y) \tag{15}$$

And for a reliable G^2 conditional independence test between C and X given Y, the minimum number of data samples N is:

$$N \geq (r_C - 1) \times (r_X - 1) \times r_Y \times \xi \tag{16}$$

where r_C, r_X, and r_Y are the numbers of class of C, X, and Y respectively, and thus $I(C; X|Y)$ has to satisfy the same requirement.

$$\begin{aligned} I(C; X|Y) &= \sum_c \sum_x \sum_y p(c, x, y) \log \frac{p(c, x|y)}{p(c|y)p(x|y)} \\ &= \sum_x \sum_y (c_1, x, y) \log \frac{p(c_1, x|y)}{p(c_1|y)p(x|y)} + \cdots \\ &\quad + \sum_x \sum_y (c_n, x, y) \log \frac{p(c_n, x|y)}{p(c_n|y)p(x|y)} \\ &= I(C_1; X|Y) + \cdots + I(C_n; X|Y) = \sum_{i=1}^{n} I(C_i; X|Y) \end{aligned} \tag{17}$$

Based on (17), we can deduce that for $I(C_i; X|Y)$, there is such a sample requirement:

$$N\omega_i \geq (r_{C_i} - 1) \times (r_X - 1) \times r_Y \times \xi \tag{18}$$

where $r_{C_i} = 2$. Further, considering each feature in the dataset, $N\omega_i \geq (r_{C_i} - 1) \times (R_F - 1) \times R_F \times \xi$, i.e., $\frac{N\omega_i}{\xi(R_F^2 - R_F)} \geq 1$.

Thus, when $\frac{N\omega_k}{\xi(R_F^2 - R_F)} \geq 1$, our class-specific method will better handle C_k than algorithms that directly optimize mutual information. ■

V. Experiments

To validate the accuracy and efficiency of the LaCFS algorithm, we compared it with six state-of-the-art algorithms on five benchmark BNs and eight real-world datasets. The six state-of-the-art algorithms are MMMB [31], HITON-MB [32], PCMB [29], BAMB [36], CCMB [35], and CFS [38]. The existing MATLAB package Causal Learner[1] has implemented these six algorithms. Also, we implemented LaCFS using MATLAB and compared LaCFS with the six comparison algorithms in this package [47]. All experiments were conducted on Windows 10, running on a computer with an Intel Core i7-117000 CPU and 32 GB of RAM. The conditional independence test was G^2 at the 0.01 significance level [48].

A. Benchmark BN Datasets

To compare the results of the LaCFS algorithm with existing causal feature algorithms, we used two sets of data generated

[1]The codes of these compared algorithms in MATLAB are available at $http://bigdata.ahu.edu.cn/causal-learner$.

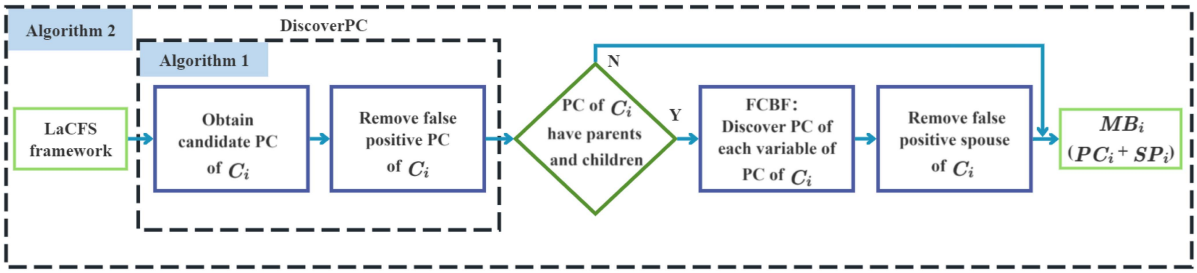


Fig. 3. LaCFS algorithm overall flow chart.

TABLE I
SUMMARY OF NOTATIONS

Symbol	Meaning
U	The entire feature set
S	A subset of **U**
C	Class feature
C_i	The i-th class of the class feature
X,Y,Z	A feature
c,x,y	Possible values that a feature can take
$C \perp\!\!\!\perp X \mid \mathbf{S}$	C and X are independent given **S**
$C \not\perp\!\!\!\perp X \mid \mathbf{S}$	C and X are dependent given **S**
$\mathbf{MB}_C$	Markov blanket of C
$\mathbf{PC}_C$	Parents and children of C
$\mathbf{SP}_C$	Spouses of C
$p(c)$	Probability that C takes the value c
$I(C_i;X)$	Class-specific mutual information between C_i and X
$I(C_i;X \mid \mathbf{S})$	Class-specific mutual information between C_i and X given set **S**

TABLE II
BENCHMARK BAYESIAN NETWORKS

Network	Num. Vars	Num. Edges	Size. Max MB	Max MB Node
Child [49]	20	25	8	2nd
Insurance [50]	27	52	10	4th
Mildew [51]	35	46	9	18th
Munin [52]	189	282	18	95th
HailFinder10 [53]	560	1017	32	3rd

from five benchmark BNs, the details of which are shown in Table I.[2] For each benchmark BN, we used two sets of data. The first set consists of 500 data samples, which we use to represent the small sample dataset, and the second set contains 1000 data samples, which we use to describe the large sample dataset.

We regard the node with the largest MB in each network as the class label and apply 10-fold cross-validation to all dataset-s. The existing causal feature selection methods identify the causal features of the class label as a feature subset and classify all classes by combining these features with the classifier. Then, the classification model is used to predict the labels for the test data without label information and calculate the prediction accuracy. In contrast, LaCFS identifies the causal features of each class of the class label and uses these causal features to train separate classification models in conjunction with a classifier. Then, these classification models are used to predict the test dataset sequentially, obtaining the prediction accuracy of the specific class under the class label. Since the predicted labels of the specific class in the class label are compared with the true labels of all the classes, the prediction accuracies of each class of the class label are summed to obtain the prediction accuracy. Unlike existing causal feature selection methods that select a fixed feature subset for each class of the class label on the training set, LaCFS is able to select a feature subset corresponding to each class of the class label based on its characteristics, thereby more accurately capturing the intrinsic structural features of the data. Furthermore, the experimental framework of LaCFS with its rivals is given in Fig. 3. On the benchmark BN dataset, the following metrics are used to evaluate the algorithms:

- Accuracy. Prediction accuracy is the percentage of correctly classified test samples in all samples. Compactness is the number of features in the output of an algorithm. We report the compactness and prediction accuracy of the NB [54] and SVM [55] classifiers as the accuracy measures of the algorithms under comparison.
- Efficiency. The number of conditional independence tests (CITs) [56] and runtime were used to assess the efficiency of the algorithms. Since LaCFS uses class-specific mutual information that varies from its rivals, we regard performing one judgment of class-specific mutual information value or one pairwise comparison of class-specific mutual information as performing a once conditional independence test.

Table II shows the results in $A \pm B$ format, where A indicates the average NB classifier accuracy, SVM classifier accuracy, compactness, CITs, and runtime, and B indicates the standard deviation. "-" indicates that the method could not generate any output with the corresponding dataset after the memory was exhausted, and the best results are highlighted in bold.

Table II shows the experimental results for LaCFS and the other six algorithms. From the experimental results, we have the following analysis:

In terms of accuracy: on these five BNs with sample sizes of 500 and 1000 (10 datasets in total), LaCFS performs best on ten datasets compared to its rivals. In particular, on the NB classifier, LaCFS improves by more than 10%, 8%, and 10% on Insurance, Mildew, and Munin datasets with 500 samples, respectively, compared to the other six algorithms. Furthermore, LaCFS improves by 5% to 20% more using the SVM classifier than MMMB, HITON-MB, PCMB, BAMB, CCMB, and CFS on Mildew and HailFinder10 datasets with 500 samples. Owing

[2]The detailed information of the benchmark BNs can be obtained from $https://www.bnlearn.com/bnrepository/$

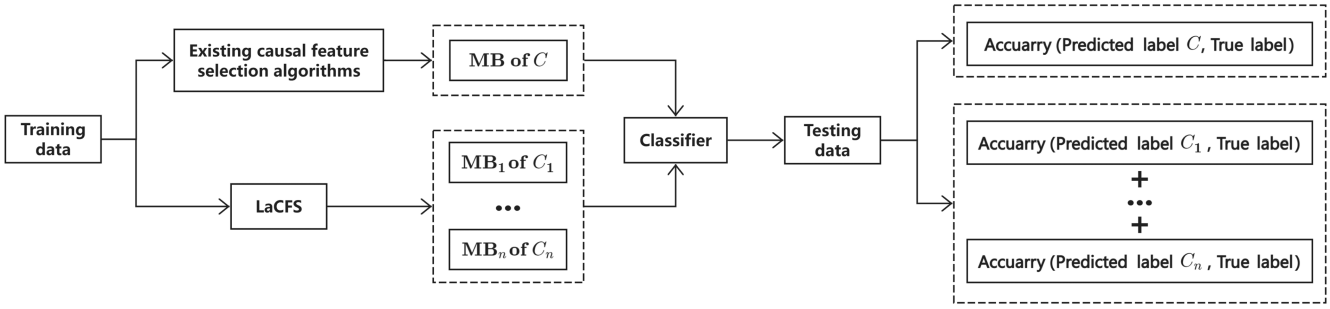


Fig. 4. The experimental framework of LaCFS and its rivals, where C denotes the class label, C_i denotes the i-th class of the class label, MB_i to represent the MB of C_i, and Accuracy (Predicted label i, True label) represents the number of i that is correctly classified in the True label with reference to the Predicted label i. LaCFS select the different specific MBs for each class of a class label, and accurately classify each class, rather than the same MB for different classes, as in existing causal feature selection algorithms.

to LaCFS selecting the different MBs for each class, classifying each class accurately, rather than the same MB for different classes as in existing algorithms; LaCFS achieves higher classification accuracy on NB and KNN classifiers. For compactness, PCMB has the smallest number of features selected on most of the ten datasets. However, its classification accuracy differs significantly from that of LaCFS, especially on the Mildew and HailFinder10 datasets, where it is only less than half the accuracy of LaCFS. On the Munin dataset, PCMB selected the least number of features, but the number is 0, resulting in its classification accuracy being 0. In addition, LaCFS did not outperform most existing causal feature selection methods in terms of the Compactness metric. This is primarily due to the strategy of LaCFS selecting causal features individually for each class of the class label. Compared to existing methods that treat the MBs of class variables as the set of features for each class of the class label, LaCFS needs to integrate the causal features for each class of the class label to ensure that all features that may affect the class variables are covered when classifying. Nonetheless, LaCFS performs optimally in terms of Compactness in the HailFinder10 network. This may be due to the extremely high correlation of the selected features with the target variables, which reduces redundant information and results in lower Compactness. Concerning CITs, LaCFS has fierce competition with other algorithms, with LaCFS performing the least CITs in 4 out of 10 datasets. LaCFS fails to optimize on the Mildew and Munin datasets, which is attributed to the fact that these two datasets have a high number of node state values. Since LaCFS requires causal feature selection for each node state value, the number of conditional independence tests required increases as the number of node state values increases. Regarding runtime: LaCFS is almost the fastest on these ten datasets. Specifically, on 8 out of 10 datasets, it requires the least run time, just below BAMB on the Mildew dataset with 500 samples, and only BAMB and CFS are faster than LaCFS on Munin dataset with 1000 samples. In particular, on HailFinder10 networks, the time consumption of LaCFS is generally more than ten times lower than that of the MMMB, HITOM-MB, BAMB, CCMB, and CFS. Since existing divide-and-conquer algorithms use enumerating conditioning sets in MB discovery, requiring extensive conditional independence tests, LaCFS avoids this problem by using class-specific mutual information leading to a significant reduction in the running time.

For a more intuitive comparison of accuracy, the accuracies of LaCFS, MMMB, HITON-MB, PCMB, BAMB, CCMB, and CFS are shown in Figs. 4 and 5. Regardless of the number of data samples and the classifier, Figs. 4 and 5(a)–5(b) show that LaCFS is superior to the other six algorithms. Especially on the Insurance and Mildew datasets, LaCFS has a significant improvement compared to the other six algorithms. To further compare the efficiency (CITs) of LaCFS with its rivals, the Friedman test was performed at the 5% significance level [57] and the Nemenyi test [58], [59]. The Friedman test is a non-parametric statistical test used to detect whether there are significant differences between multiple related samples [60]. Unlike parametric testing methods, the Friedman test does not require the data to meet the assumptions of normality or homogeneity of variance. Considering that experimental data cannot satisfy these assumptions [61], the Friedman test was chosen for the experiment. In the experiment, the null hypothesis of the Friedman test is that there is no significant difference in CITs between different algorithms. The test results reject this hypothesis, indicating that there are significant differences in efficiency among the algorithms. The average ranks for MMMB, HITON-MB, PCMB, BAMB, CCMB, CFS, and LaCFS are 3.25, 4.05, 4.30, 4.50, 1.50, 4.70, 5.70, respectively (the higher the rank, the better the performance in efficiency). Since the Friedman test can only determine whether there are differences overall, the experiment further uses the Nemenyi test for post hoc testing to determine which algorithms specifically have differences. The Nemenyi test is also a non-parametric statistical test [60], which is suitable for post hoc testing, especially when the data does not meet the normal distribution assumption. The core of the Nemenyi test is to calculate the average ranking of each group and determine whether there is a significant performance difference between the two algorithms based on a predefined critical difference. When the difference between the average rankings of two algorithms exceeds the critical difference, it can be concluded that there is a significant difference in efficiency between them. The critical difference of CITs is up to 2.85. Thus, it can be observed that LaCFS is significantly more efficient than others.

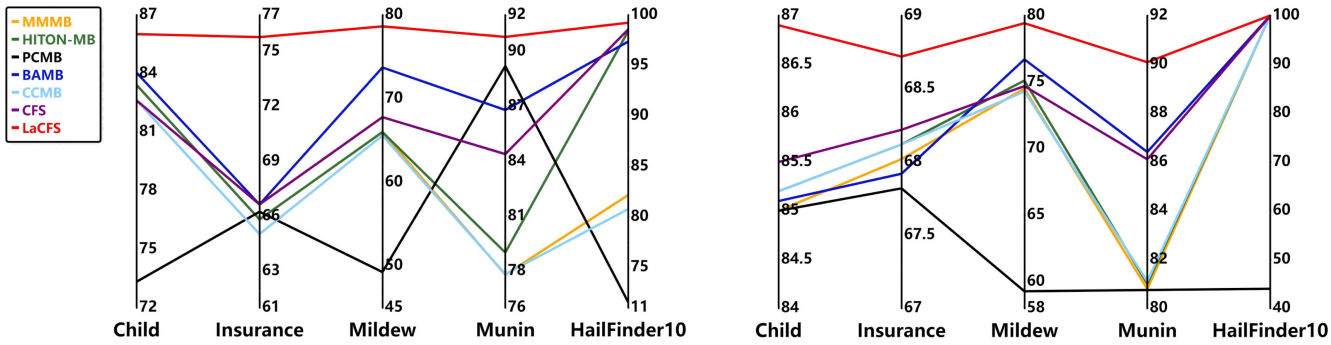


Fig. 5. The classification accuracies (%) of LaCFS and its competitors using the NB classifier on five benchmark BNs with different data sizes. (Left: Size=500; Right: Size=1000).

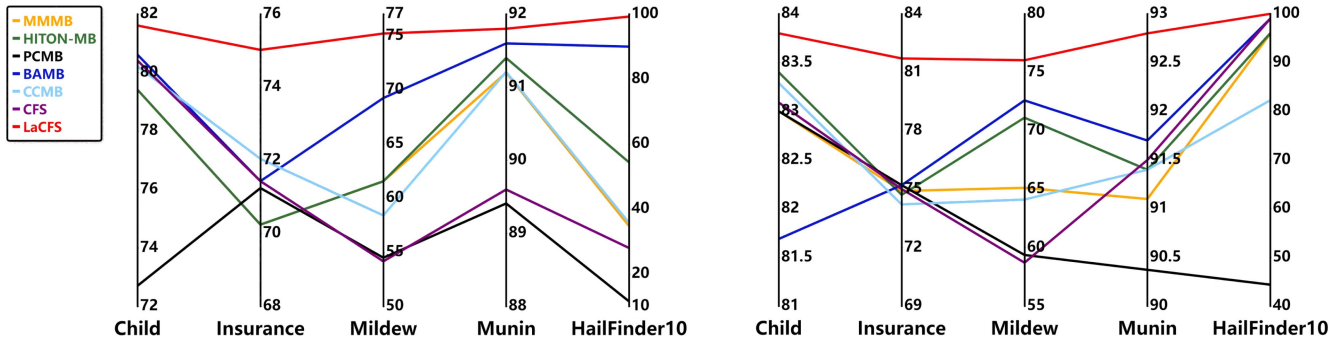


Fig. 6. The classification accuracies (%) of LaCFS and its competitors using the SVM classifier on five benchmark BNs with different data sizes. (Left: Size=500; Right: Size=1000).

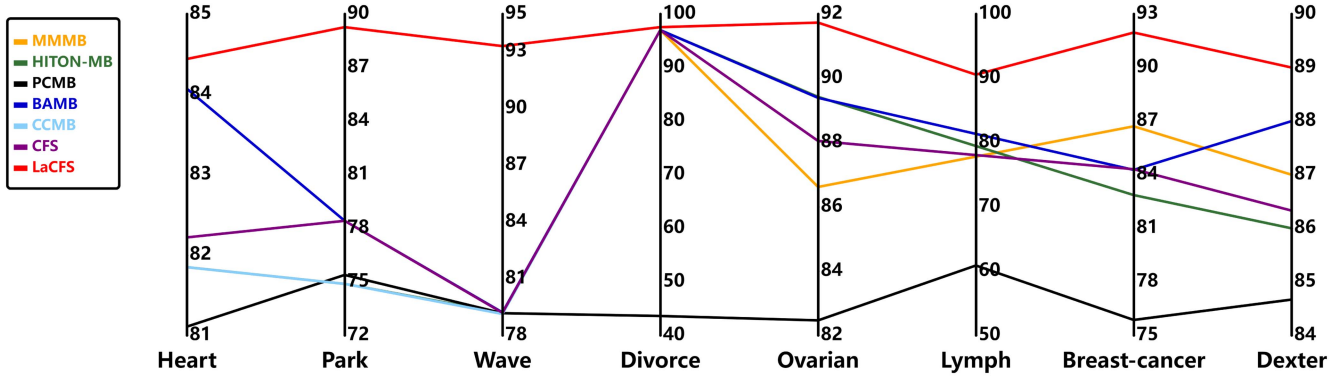


Fig. 7. The experimental results of the classification accuracies (%) of LaCFS and its competitors using the NB classifier on eight real-world datasets.

In summary, the results on these five BN datasets show that the LaCFS algorithm is competitively efficient. However, the accuracy of LaCFS is higher than all the above algorithms. This is because, based on the class-specific mutual information, LaCFS selects the causal features for each class of the class label, classifying each class accurately.

B. Real-World Datasets

In addition to the benchmark BN, the performance on real-world datasets is also essential. In this section, we test the algorithms on the eight real-world datasets in Table III, ranging from low to high dimensionality. The Heart, Wave, Divorce, Lymph, Breast-cancer, and Dexter are UCI machine learning databases [62], and the Park [63] and Ovarian [64] are biological datasets. We apply 10-fold cross-validation to all datasets and compare the mean of each iteration with the other six algorithms. For these existing causal feature selection methods, the causal features of the class label are used as a feature subset to classify all classes; and for our class-specific causal feature selection algorithm, the causal features of each class in the class label are used as a feature subset to classify the corresponding class.

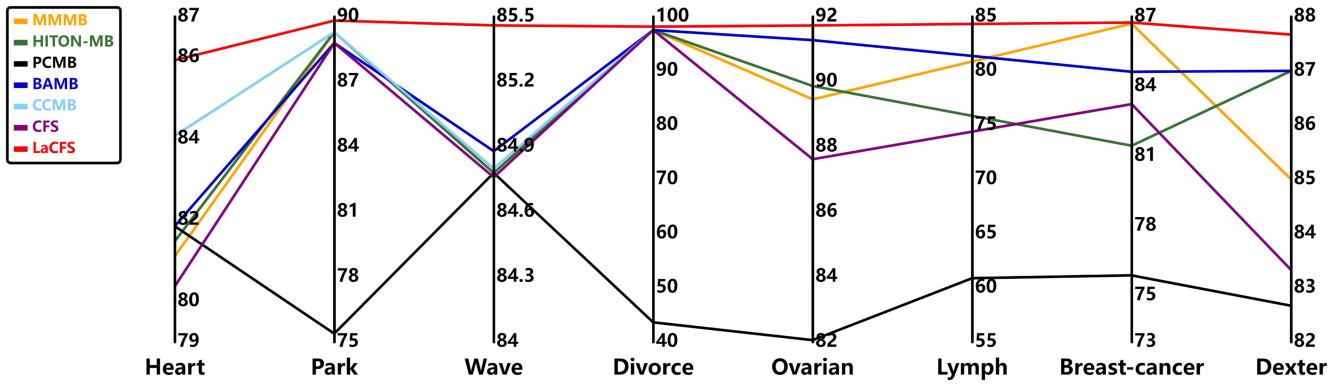


Fig. 8. The experimental results of the classification accuracies (%) of LaCFS and its competitors using the SVM classifier on eight real-world datasets.

TABLE III
COMPARISON OF LACFS, MMMB, HITON-MB, PCMB, BAMB, CCMB AND CFS ON FIVE BENCHMARK BNS WITH 500 SAMPLES

Network	Algorithm	NB	SVM	Compactness	CITs	Runtime
Child	MMMB	83.41±6.03	79.41±3.78	11±0	$(1.24\pm0.27)\times10^3$	0.12±0.02
	HITON-MB	83.41±6.03	79.41±3.78	11±1	$(1.16\pm0.23)\times10^3$	0.08±0.01
	PCMB	73.34±13.35	72.71±10.76	**7±2**	$(2.84\pm1.19)\times10^3$	0.43±0.16
	BAMB	84.04±6.93	80.61±3.31	11±1	$(1.91\pm0.33)\times10^3$	0.16±0.04
	CCMB	82.63±5.92	80.21±4.66	11±1	$(2.92\pm0.50)\times10^3$	0.20±0.03
	CFS	82.62±4.33	80.41±2.83	9±1	$(1.57\pm0.45)\times10^3$	0.09±0.02
	LaCFS	**86.02±5.41**	**81.61±4.12**	15±1	$(\mathbf{1.14\pm0.06})\times10^3$	**0.05±0.00**
Insurance	MMMB	65.83±5.67	70.24±5.69	9±1	$(5.29\pm0.70)\times10^2$	0.07±0.01
	HITON-MB	65.83±5.67	70.24±5.69	9±1	$(5.52\pm0.73)\times10^2$	0.03±0.01
	PCMB	66.25±5.62	71.24±4.04	9±1	$(2.59\pm0.39)\times10^3$	0.44±0.07
	BAMB	66.65±6.26	71.43±5.13	**8±0**	$(7.05\pm1.62)\times10^2$	0.08±0.02
	CCMB	65.03±7.32	72.04±5.09	**8±0**	$(4.22\pm0.22)\times10^3$	0.37±0.02
	CFS	66.65±6.26	71.43±5.13	**8±0**	$(\mathbf{5.01\pm0.82})\times10^2$	0.05±0.01
	LaCFS	**75.79±6.36**	**75.02±8.33**	12±2	$(6.19\pm0.41)\times10^2$	**0.02±0.01**
Mildew	MMMB	66.01±7.40	61.56±7.31	34±0	$(1.50\pm0.11)\times10^6$	124.80±8.63
	HITON-MB	66.01±7.40	61.56±7.31	34±0	$(1.50\pm0.11)\times10^6$	124.80±8.63
	PCMB	49.28±6.59	54.50±6.51	**1±0**	$(1.67\pm0.03)\times10^3$	0.38±0.01
	BAMB	73.73±8.07	69.25±4.67	10±1	$(\mathbf{1.26\pm0.35})\times10^3$	**0.09±0.03**
	CCMB	65.60±7.74	58.41±4.88	27±1	$(8.73\pm1.13)\times10^4$	3.26±0.46
	CFS	67.79±6.75	54.17±5.45	19±1	$(3.39\pm0.52)\times10^4$	2.01±0.31
	LaCFS	**78.64±4.22**	**75.21±5.58**	19±7	$(2.21\pm0.81)\times10^3$	0.12±0.04
Munin	MMMB	77.81±5.91	91.20±1.39	118±5	$(1.67\pm0.15)\times10^8$	21843.59±2263.61
	HITON-MB	79.02±5.25	91.40±1.34	99±3	$(3.34\pm0.36)\times10^7$	4733.75±652.82
	PCMB	89.21±2.48	89.41±1.83	**1±0**	$(1.65\pm0.68)\times10^4$	1.52±0.64
	BAMB	86.81±3.22	91.60±1.57	13±3	$(3.79\pm1.82)\times10^3$	0.19±0.13
	CCMB	77.84±6.67	91.21±2.65	107±5	$(3.45\pm0.15)\times10^8$	40932.21±1371.58
	CFS	84.40±3.73	89.60±0.79	19±2	$(1.78\pm0.33)\times10^4$	0.73±0.14
	LaCFS	**90.80±1.36**	**91.80±1.49**	19±3	$(\mathbf{3.63\pm0.64})\times10^3$	**0.11±0.01**
HailFinder10	MMMB	82.17±3.29	34.78±5.60	256±4	$(1.33\pm0.08)\times10^7$	934.32±60.39
	HITON-MB	98.39±1.28	54.37±10.59	66±7	$(1.24\pm0.15)\times10^5$	4.74±0.70
	PCMB	11.56±3.08	11.56±3.08	**1±0**	$(\mathbf{1.48\pm0.28})\times10^4$	1.22±0.22
	BAMB	97.37±2.89	90.00±3.11	24±1	$(7.84\pm1.55)\times10^4$	6.12±1.57
	CCMB	80.78±3.85	35.78±5.50	271±8	$(7.99\pm0.74)\times10^6$	264.10±18.88
	CFS	98.57±1.94	28.01±3.53	43±1	$(1.22\pm0.15)\times10^6$	72.77±10.05
	LaCFS	**99.23±2.43**	**99.23±2.43**	3±0	$(1.55\pm0.04)\times10^4$	**0.53±0.01**

Like the BN dataset, we also focus on time efficiency and accuracy.

- Accuracy. Prediction accuracy is the percentage of correctly classified test samples in all samples. Compactness is the number of features in the output of an algorithm. We report the compactness and prediction accuracy of the NB and SVM classifiers as the accuracy measures of the algorithms under comparison.
- Efficiency. We report both the number of CITs and runtime as the efficiency measures.

Table IV shows the results in $A \pm B$ format, where A indicates the average NB classifier accuracy, SVM classifier accuracy, compactness, CITs, and runtime, and B indicates the standard deviation. "-" indicates that the method could not generate any output with the corresponding dataset after the memory was exhausted or running for three days. The best results are highlighted in bold.

Table IV shows the experimental results for LaCFS and the other six algorithms. From the experimental results, we have the following analysis:

We can see that LaCFS outperforms the other algorithms for all datasets regarding classification accuracy, regardless of using the NB or SVM classifiers. For the Park and Wave datasets, LaCFS achieves 10% to 15% or higher classification accuracy than the other algorithms when using the NB classifier. In addition, using the SVM classifier, LaCFS is 4% higher

TABLE IV
COMPARISON OF LACFS, MMMB, HITON-MB, PCMB, BAMB, CCMB AND CFS ON FIVE BENCHMARK BNS WITH 1000 SAMPLES

Network	Algorithm	NB	SVM	Compactness	CITs	Runtime
Child	MMMB	85.00±1.69	83.00±2.57	11±1	(1.32±0.24)×10^3	0.14±0.02
	HITON-MB	85.20±2.31	83.40±2.78	**9±1**	(1.24±0.21)×10^3	0.10±0.01
	PCMB	85.00±1.69	83.00±2.57	11±1	(9.39±2.15)×10^3	1.39±0.30
	BAMB	85.10±3.20	81.69±2.94	11±1	(1.91±0.19)×10^3	0.21±0.02
	CCMB	85.20±2.57	83.29±3.37	10±1	(3.47±0.44)×10^3	0.28±0.03
	CFS	85.50±2.75	83.09±3.41	**9±1**	(1.72±0.42)×10^3	0.12±0.03
	LaCFS	**86.90±2.93**	**83.80±3.03**	12±1	(**1.07±0.04**)×10^3	**0.05±0.00**
Insurance	MMMB	68.02±5.34	74.91±3.51	9±1	(**5.78±0.12**)×10^2	0.09±0.00
	HITON-MB	68.12±5.41	74.71±3.47	**8±0**	(6.41±0.14)×10^2	0.05±0.01
	PCMB	67.82±5.47	75.21±3.53	9±0	(3.16±0.19)×10^3	0.58±0.04
	BAMB	67.92±5.55	75.21±3.52	**8±0**	(7.18±0.26)×10^2	0.09±0.01
	CCMB	68.12±5.41	74.22±3.69	**8±0**	(5.08±0.17)×10^3	0.44±0.03
	CFS	68.22±5.07	75.01±3.52	**8±0**	(5.87±0.04)×10^2	0.06±0.01
	LaCFS	**68.72±5.12**	**81.71±3.40**	12±2	(6.55±0.48)×10^2	**0.03±0.00**
Mildew	MMMB	74.51±4.19	65.12±1.90	34±0	(1.14±0.04)×10^6	76.22±3.13
	HITON-MB	75.12±5.03	71.12±6.36	24±1	(2.42±0.21)×10^4	0.98±0.09
	PCMB	59.29±2.98	59.38±2.90	**1±0**	(**1.68±0.01**)×10^3	0.27±0.01
	BAMB	76.71±3.61	72.60±4.59	13±0	(6.42±0.63)×10^3	0.41±0.06
	CCMB	74.30±4.38	64.12±2.70	28±1	(8.71±0.18)×10^4	3.08±0.13
	CFS	74.70±4.85	58.72±3.98	18±0	(3.37±0.12)×10^4	1.64±0.04
	LaCFS	**79.43±5.00**	**76.03±4.47**	22±4	(2.13±0.44)×10^3	**0.14±0.03**
Munin	MMMB	80.80±2.19	91.10±0.83	73±8	(1.33±0.77)×10^7	1763.60±1132.31
	HITON-MB	81.01±2.30	91.40±0.66	72±8	(6.67±0.42)×10^6	698.02±462.94
	PCMB	0.00±0.00	0.00±0.00	**0±0**	(4.28±0.06)×10^3	0.45±0.06
	BAMB	86.41±2.04	91.70±1.04	13±2	(2.06±0.91)×10^3	0.17±0.09
	CCMB	81.11±2.28	91.40±0.66	74±8	(7.55±0.46)×10^7	9112.02±560.21
	CFS	86.11±2.13	91.50±1.55	12±2	(**2.00±0.07**)×10^3	**0.09±0.04**
	LaCFS	**90.09±3.14**	**92.80±1.67**	25±6	(5.97±0.22)×10^3	0.24±0.08
HailFinder10	MMMB	**100.00±0.00**	95.92±3.69	32±2	(1.40±0.02)×10^5	7.53±0.27
	HITON-MB	**100.00±0.00**	96.02±3.74	32±2	(1.15±0.10)×10^5	5.45±0.06
	PCMB	44.02±13.21	44.42±12.65	4±1	(4.31±0.55)×10^4	5.10±0.71
	BAMB	**100.00±0.00**	99.00±1.25	30±1	(1.09±0.17)×10^5	10.16±0.42
	CCMB	99.90±0.32	82.26±13.60	49±11	(4.44±0.53)×10^6	151.54±16.21
	CFS	99.90±0.32	82.26±13.60	49±11	(4.44±0.53)×10^6	151.54±16.21
	LaCFS	**100.00±0.00**	**100.00±0.00**	**3±0**	(**1.35±0.00**)×10^4	**0.51±0.00**

TABLE V
REAL-WORLD DATASETS

Dataset	Number of features	Number of samples
Heart	13	270
Park	22	195
Wave	40	5000
Divorce	54	170
Ovarian	2190	216
Lymph	4026	96
Breast-cancer	17816	280
Dexter	20000	300

than MMMB, HITON-MB, PCMB, BAMB, and CFS on the Heart dataset. On the Ovarian and Breast-cancer data, CCMB cannot obtain results in an efficient time, mainly because the CCMB introduces cross-checking and complement processes to find more true variables, which is a time-consuming step. Meanwhile, other algorithms have selected many features on the high-dimensional dataset of Lymph, making it difficult to obtain valid results. Only PCMB and LaCFS yield results, but the former selected only one feature resulting in a lower classification accuracy than the latter, reaching around 20% to 30%. Regarding compactness, the number of features PCMB selects is almost minimal on the eight datasets. However, there is a significant difference between its classification accuracy and that of LaCFS, especially on the Divorce dataset, where the accuracy is only half that of LaCFS. LaCFS did not outperform most existing causal feature selection methods in terms of the Compactness metric. This is primarily due to the strategy of LaCFS selecting causal features individually for each class of the class label. To ensure that all features that may affect the class variables are covered when classifying, LaCFS needs to integrate the causal features for each class of the class label. Specifically, LaCFS has the best Compactness performance on the Park and Divorce datasets, excluding the case where PCMB sacrifices accuracy to extract only one feature. This may be due to the relatively small number of features and moderate sample size of Park and Divorce datasets, which enable LaCFS to perform feature selection and optimization more effectively. Regarding CITs and runtime: LaCFS does exhibit some variations when handling datasets of different scales. On low-dimensional datasets, such as Heart, Park, Wave, and Divorce, the time consumption of LaCFS is the least. Specifically, in these datasets, LaCFS takes just under 0.1 times the runtime required by the other algorithms. At the same time, LaCFS performs fewer CITs. On the Wave dataset, the number of CITs required for others is more than ten times that of LaCFS. This indicates that LaCFS incurs minimal computational overhead on low-dimensional datasets, allowing for rapid results. However, on high-dimensional datasets such as Ovarian, Breast-cancer, and Dexter, the efficiency of LaCFS is indeed affected. This is because LaCFS needs to discover spouses from the PC of each variable in the PC of each class, which significantly increases the computational complexity when the number of features is vast. Thus, on these high-dimensional datasets, the computation time of LaCFS is longer, and there are some challenges in terms of scalability. In contrast, CFS discovers spouses from the PC of children of multiple parents of the target variable to improve efficiency, thus requiring the fewest CITs and least runtime on high-dimensional datasets

TABLE VI
COMPARISON OF LACFS, MMMB, HITON-MB, PCMB, BAMB, CCMB AND CFS ON REAL-WORLD DATASTES

Dataset	Algorithm	NB	SVM	Compactness	CITs	Runtime
Heart	MMMB	81.85±8.63	81.11±8.63	7±0	(2.71±0.81)×10^2	0.05±0.01
	HITON-MB	81.85±8.63	81.48±8.90	7±0	(2.61±0.55)×10^2	**0.01±0.00**
	PCMB	81.11±8.63	81.85±7.50	**5±1**	(8.39±1.87)×10^2	0.16±0.04
	BAMB	84.07±8.56	81.85±7.08	7±1	(3.69±1.08)×10^2	0.03±0.02
	CCMB	81.85±9.63	84.07±8.20	8±1	(7.11±1.17)×10^2	0.09±0.02
	CFS	82.22±9.04	80.37±7.82	**5±0**	**(2.23±0.68)**×10^2	0.02±0.01
	LaCFS	**84.44±7.77**	**85.93±6.72**	9±1	(3.05±0.08)×10^2	**0.01±0.00**
Park	MMMB	74.89±10.09	89.29±5.60	22±0	(2.94±0.19)×10^5	14.92±1.00
	HITON-MB	74.89±10.09	89.29±5.60	22±0	(2.42±0.16)×10^5	9.89±0.69
	PCMB	75.39±1.97	75.39±1.97	**1±0**	(5.77±0.77)×10^2	0.12±0.02
	BAMB	78.42±7.22	88.79±6.21	17±1	(1.19±0.16)×10^4	0.90±0.14
	CCMB	74.89±10.09	89.29±5.60	22±0	(4.06±0.35)×10^5	19.73±1.18
	CFS	78.42±7.22	88.79±6.21	17±1	(2.58±0.35)×10^4	0.92±0.13
	LaCFS	**89.26±5.46**	**89.82±7.60**	8±1	**(4.13±0.02)**×10^2	**0.01±0.00**
Wave	MMMB	79.18±1.82	84.78±1.75	**17±0**	(5.41±0.31)×10^4	5.30±0.25
	HITON-MB	79.18±1.82	84.78±1.75	**17±0**	(4.51±0.25)×10^4	3.24±0.18
	PCMB	79.18±1.82	84.78±1.75	**17±0**	(6.54±0.45)×10^5	126.58±9.26
	BAMB	79.22±1.67	84.88±2.02	18±1	(6.75±0.39)×10^4	15.57±0.81
	CCMB	79.14±1.95	84.80±1.72	**17±0**	(8.11±0.42)×10^4	7.71±0.60
	CFS	79.20±1.81	84.76±1.77	**17±0**	(2.61±0.18)×10^4	1.62±0.09
	LaCFS	**93.30±0.96**	**85.46±1.10**	**17±0**	**(1.49±0.08)**×10^3	**0.16±0.01**
Divorce	MMMB	96.98±4.34	97.54±4.34	54±0	(9.06±0.55)×10^7	7239.21±547.85
	HITON-MB	96.98±4.34	97.54±4.34	54±0	(7.32±0.45)×10^7	6344.16±428.92
	PCMB	43.64±39.13	43.64±39.13	**1±1**	**(4.79±1.56)**×10^3	0.59±0.16
	BAMB	96.98±4.34	97.54±4.34	54±0	(1.34±0.00)×10^6	245.53±9.51
	CCMB	96.98±4.34	97.54±4.34	54±0	(8.69±0.45)×10^7	8104.59±460.57
	CFS	96.98±4.34	97.54±4.34	54±0	(1.34±0.00)×10^6	242.71±9.46
	LaCFS	**97.60±4.08**	**97.60±4.08**	54±0	(4.83±0.23)×10^5	**1.72±0.05**
Ovarian	MMMB	86.62±4.40	89.46±8.27	9±2	(2.74±0.20)×10^4	2.62±0.28
	HITON-MB	89.42±6.42	89.87±6.30	7±1	(3.65±0.43)×10^4	1.87±0.32
	PCMB	82.47±8.48	82.06±11.18	**3±1**	(3.28±6.48)×10^6	2112.92±4535.97
	BAMB	89.39±6.78	91.28±6.87	10±1	(4.63±1.99)×10^4	3.28±1.34
	CCMB	-	-	-	-	-
	CFS	88.05±10.26	87.62±8.23	6±1	**(1.86±0.10)**×10^4	**1.39±0.07**
	LaCFS	**91.73±5.93**	**91.73±7.02**	45±4	(9.25±0.76)×10^4	2.13±0.20
Lymph	MMMB	-	-	-	-	-
	HITON-MB	-	-	-	-	-
	PCMB	60.89±22.39	60.89±22.39	**1±0**	(2.05±0.48)×10^6	176.27±44.36
	BAMB	-	-	-	-	-
	CCMB	-	-	-	-	-
	CFS	-	-	-	-	-
	LaCFS	**90.56±5.79**	**84.33±10.40**	182±42	**(1.40±0.32)**×10^6	**29.30±6.65**
Breast_cancer	MMMB	86.71±5.71	86.70±5.25	19±7	(2.65±0.62)×10^5	31.04±7.12
	HITON-MB	82.86±9.55	81.45±8.91	11±2	(3.09±0.29)×10^5	13.50±1.32
	PCMB	75.87±3.85	75.87±3.85	**2±1**	(1.88±3.87)×10^6	758.52±2104.79
	BAMB	84.27±5.31	84.63±4.66	14±2	(1.52±0.94)×10^6	1147.26±728.87
	CCMB	-	-	-	-	-
	CFS	84.31±6.72	83.24±7.28	8±2	**(1.67±0.18)**×10^5	**12.23±1.01**
	LaCFS	**91.96±2.91**	**86.75±6.20**	217±14	(4.76±0.17)×10^6	167.17±25.21
Dexter	MMMB	87.00±9.09	85.00±10.33	11±3	(1.91±0.18)×10^5	31.83±3.78
	HITON-MB	86.00±9.00	87.00±8.67	10±2	(1.99±0.26)×10^5	**7.72±1.12**
	PCMB	84.67±8.20	82.67±9.79	**8±2**	(8.70±1.34)×10^5	122.27±21.72
	BAMB	88.00±8.34	87.00±8.67	13±1	(3.47±0.56)×10^5	17.91±3.59
	CCMB	-	-	-	-	-
	CFS	86.33±8.23	83.33±10.89	9±1	**(1.91±0.18)**×10^5	11.81±1.38
	LaCFS	**89.00±9.03**	**87.67±8.47**	45±5	(5.44±0.41)×10^5	12.82±1.02

such as Ovarian, Breast-cancer, and Dexter. In addition, the contradiction between high feature counts and low sample sizes in high-dimensional datasets, leading to the curse of dimensionality and the risk of overfitting, makes LaCFS require more CITs to explore these relationships. It is worth noting that the Lymph dataset is difficult for most methods to learn effectively due to its low sample size. However, LaCFS is able to identify the most relevant features of each class of the class label, thereby enabling effective learning despite the lack of samples.

To provide a more visual representation of the accuracy of LaCFS against its competitors, we show the performance of the seven algorithms on the NB and SVM classifiers in Figs. 6 and 7. The LaCFS algorithm has the highest classification accuracy on eight datasets regardless of whether it is the NB or SVM classifier. In particular, the LaCFS algorithm significantly improves classification accuracy on the NB classifier compared to the other algorithms. To further evaluate the efficiency (CITs) of the proposed methods against other methods, the Friedman test is conducted at a 5% significance level under the null hypothesis. The null hypothesis of CITs is rejected; the average ranks for MMMB, HITON-MB, PCMB, BAMB, CCMB, CFS, and LaCFS are 4.13, 4.38, 3.50, 3.88, 1.63, 5.63, 4.88, respectively (the higher the rank, the better the performance in efficiency). Then, with the Nemenyi test, the critical difference of CITs is up to 3.18; thus, it can be observed that LaCFS is significantly more efficient than MMMB, HITON-MB, PCMB, BAMB, and CCMB.

In summary, on eight real-world datasets, the LaCFS algorithm has the highest classification accuracy on NB and SVM classifiers, which is still comparable in efficiency.

VI. CONCLUSION

In this paper, we analyze the unique relationships between each class in the class label and its causal features in class-specific mutual information. Then we propose a label-aware causal feature selection algorithm (LaCFS) based on class-specific mutual information to identify the causal features of each class in the class label. Through extensive experiments

on five benchmark BN and eight real-world datasets, the results show that LaCFS has a significant advantage in terms of accuracy while its time consumption is comparable to other methods. Specifically, LaCFS outperforms all compared algorithms in terms of accuracy, especially when dealing with high-dimensional datasets. However, the computational efficiency on high-dimensional datasets still needs to be improved. To further improve the scalability and computational efficiency of LaCFS on larger datasets, future work could consider optimizing the feature selection and spouse discovery processes. By improving the feature selection strategy and reducing unnecessary computational steps by combining deep learning approaches, it is possible to mitigate the computational complexity while maintaining high accuracy, thereby increasing the efficiency of LaCFS and application potential on larger datasets. In addition, the class-specific mutual information computation used in LaCFS can only handle discrete values, and continuous values must be discretized in advance. In this process, the discretization method also affects the performance of the algorithm. Thus, future research could focus on exploring the direct application of class-specific mutual information computation methods to continuous and mixed data scenarios to further enhance the applicability and performance of the algorithm.

REFERENCES

[1] H. Ni et al., "Feature incremental learning with causality," *Pattern Recognit.*, vol. 146, 2024, Art. no. 110033.

[2] F. Kamalov, F. Thabtah, and H. H. Leung, "Feature selection in imbalanced data," *Ann. Data Sci.*, vol. 10, pp. 1527–1541, 2023.

[3] X. Huang et al., "Decorrelated spectral regression: An unsupervised dimension reduction method under data selection bias," *Neurocomputing*, vol. 549, 2023, Art. no. 126406.

[4] D. Koller and M. Sahami, "Toward optimal feature selection," in *Proc. 13th Int. Conf. Int. Conf. Mach. Learn.*, pp. 284–292, 1996.

[5] I. Guyon et al., "Causal feature selection," in *Computational Methods of Feature Selection*. London, U.K.: Chapman and Hall/CRC, 2007, pp. 79–102.

[6] R. Jiao, B. H. Nguyen, B. Xue, and M. Zhang, "A survey on evolutionary multiobjective feature selection in classification: Approaches, applications, and challenges," *IEEE Trans. Evol. Comput.*, vol. 28, no. 4, pp. 1156–1176, Aug. 2024.

[7] Y. Hu, X. Zhang, E. Ngai, R. Cai, and M. Liu, "Software project risk analysis using Bayesian networks with causality constraints," *Decis. Support Syst.*, vol. 56, pp. 439–449, 2013.

[8] F. Xie, R. Cai, B. Huang, C. Glymour, Z. Hao, and K. Zhang, "Generalized independent noise condition for estimating latent variable causal graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 14,891–14,902.

[9] K. Yu, L. Liu, J. Li, W. Ding, and T. D. Le, "Multi-source causal feature selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 9, pp. 2240–2256, Sep. 2020.

[10] X. Yuan, X. Hu, and P. Li, "Multi-source multi-label feature selection," in *Proc. 2023 Int. Joint Conf. Neural Netw.*, 2023, pp. 1–8.

[11] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, "Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: Algorithms and empirical evaluation," *J. Mach. Learn. Res.*, vol. 11, no. 1, pp. 171–234, 2010.

[12] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, "Local causal and Markov blanket induction for causal discovery and feature selection for classification Part II: Analysis and extensions," *J. Mach. Learn. Res.*, vol. 11, no. 1, pp. 235–284, 2010.

[13] J. G. Richens, C. M. Lee, and S. Johri, "Improving the accuracy of medical diagnosis with causal machine learning," *Nat. Commun.*, vol. 11, no. 1, 2020, Art. no. 3923.

[14] A. Moors, Y. Boddez, and J. D. Houwer, "The power of goal-directed processes in the causation of emotional and other actions," *Emotion Rev.*, vol. 9, no. 4, pp. 310–318, 2017.

[15] E. K. Buabang, Y. Boddez, O. T. Wolf, and A. Moors, "The role of goal-directed and habitual processes in food consumption under stress after outcome devaluation with taste aversion," *Behav. Neurosci.*, vol. 137, no. 1, pp. 1–14, 2023.

[16] A. Yadu, P. Suhas, and N. Sinha, "Class specific interpretability in CNN using causal analysis," in *Proc. 2021 IEEE Int. Conf. Image Process.*, 2021, pp. 3702–3706.

[17] O. Ahmad, N. Béreux, L. Baret, V. Hashemi, and F. Lecue, "Causal analysis for robust interpretability of neural networks," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2024, pp. 4685–4694.

[18] K. Yu et al., "Causality-based feature selection: Methods and evaluations," *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–36, 2020.

[19] X. Zhang, H. Yao, Z. Lv, and D. Miao, "Class-specific information measures and attribute reducts for hierarchy and systematicness," *Inf. Sci.*, vol. 563, pp. 196–225, 2021.

[20] L. Luo et al., "Tri-level attribute reduction based on neighborhood rough sets," *Appl. Intell.*, vol. 54, no. 5, pp. 3786–3807, 2024.

[21] W. Gao, L. Hu, and P. Zhang, "Class-specific mutual information variation for feature selection," *Pattern Recognit.*, vol. 79, pp. 328–339, 2018.

[22] P. Wu et al., "Dynamic feature selection combining standard deviation and interaction information," *Int. J. Mach. Learn. Cybern.*, vol. 14, pp. 1407–1426, 2023.

[23] K. Kuang et al., "Causal inference," *Engineering*, vol. 6, no. 3, pp. 253–263, 2020.

[24] Z. Shen, P. Cui, K. Kuang, B. Li, and P. Chen, "Causally regularized learning with agnostic data selection bias," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 411–419.

[25] J. Runge et al., "Causal inference for time series," *Nat. Rev. Earth Environ.*, vol. 4, no. 7, pp. 487–505, 2023.

[26] D. Margaritis and S. Thrun, "Bayesian network induction via local neighborhoods," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 505–511.

[27] I. Tsamardinos, C. F. Aliferis, A. R. Statnikov, and E. Statnikov, "Algorithms for large scale Markov blanket discovery," in *Proc. FLAIRS Conf.*, 2003, pp. 376–380.

[28] G. Borboudakis and I. Tsamardinos, "Forward-backward selection with early dropping," *J. Mach. Learn. Res.*, vol. 20, no. 1, pp. 276–314, 2019.

[29] J. M. Pena, R. Nilsson, J. Björkegren, and J. Tegnér, "Towards scalable and data efficient learning of Markov boundaries," *Int. J. Approx. Reasoning*, vol. 45, no. 1, pp. 97–122, 2007.

[30] A. Srivastava, S. P. Chockalingam, and S. Aluru, "A parallel framework for constraint-based Bayesian network learning via Markov blanket discovery," *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 6, pp. 1699–1715, Jun. 2023.

[31] I. Tsamardinos, C. F. Aliferis, and A. Statnikov, "Time and sample efficient discovery of Markov blankets and direct causal relations," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2003, pp. 673–678.

[32] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, "Hiton: A novel Markov blanket algorithm for optimal variable selection," in *Proc. AMIA Annu. Symp.*, 2003, pp. 21–25.

[33] S. Fu and M. C. Desmarais, "Fast Markov blanket discovery algorithm via local learning within single pass," in *Proc. Conf. Can. Soc. Comput. Stud. Intell.*, Springer, 2008, pp. 96–107.

[34] T. Gao and Q. Ji, "Efficient Markov blanket discovery and its application," *IEEE Trans. Cybern.*, vol. 47, no. 5, pp. 1169–1179, May 2017.

[35] X. Wu, B. Jiang, K. Yu, H. Chen, and C. Miao, "Accurate Markov boundary discovery for causal feature selection," *IEEE Trans. Cybern.*, vol. 50, no. 12, pp. 4983–4996, Dec. 2020.

[36] Z. Ling, K. Yu, H. Wang, L. Liu, W. Ding, and X. Wu, "Bamb: A balanced Markov blanket discovery approach to feature selection," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 5, pp. 1–25, 2019.

[37] H. Wang, Z. Ling, K. Yu, and X. Wu, "Towards efficient and effective discovery of Markov blankets for feature selection," *Inf. Sci.*, vol. 509, pp. 227–242, 2020.

[38] Z. Ling, B. Li, Y. Zhang, Q. Wang, K. Yu, and X. Wu, "Causal feature selection with efficient spouses discovery," *IEEE Trans. Big Data*, vol. 9, no. 2, pp. 555–568, Apr. 2023.

[39] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[40] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, Prediction, and Search*. Cambridge, MA, USA: MIT Press, 2000.

[41] X. Zhang, J. Yang, and L. Tang, "Three-way class-specific attribute reducts from the information viewpoint," *Inf. Sci.*, vol. 507, pp. 840–872, 2020.

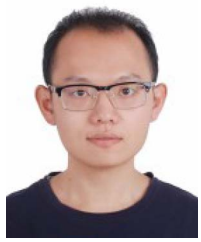
[42] T. M. Cover and J. A. Thomas, *Information Theory and Statistics*, vol. 1. New York, NY, USA: Dover, 1991.

[43] X. F. Song, Y. Zhang, D. W. Gong, and X. Y. Sun, "Feature selection using bare-bones particle swarm optimization with mutual information," *Pattern Recognit.*, vol. 112, 2021, Art. no. 107804.
[44] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *J. Mach. Learn. Res.*, vol. 5, pp. 1205–1224, 2004.
[45] K. Yu, Z. Ling, L. Liu, P. Li, H. Wang, and J. Li, "Feature selection for efficient local-to-global Bayesian network structure learning," *ACM Trans. Knowl. Discov. Data*, vol. 18, no. 2, pp. 1–27, 2023.
[46] K. Yu, L. Liu, and J. Li, "A unified view of causal and non-causal feature selection," *ACM Trans. Knowl. Discov. Data*, vol. 15, no. 4, pp. 1–46, 2021.
[47] Z. Ling, K. Yu, Y. Zhang, L. Liu, and J. Li, "Causal learner: A toolbox for causal structure and Markov blanket learning," *Pattern Recognit. Lett.*, vol. 163, pp. 92–95, 2022.
[48] R. E. Neapolitan et al., *Learning Bayesian Networks*, vol. 38, Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2004.
[49] D. J. Spiegelhalter and R. G. Cowell, "Learning in probabilistic expert systems," in *Bayesian Statistics 4*, J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, Eds. Oxford, U.K.: Clarendon Press, 1992, pp. 447–466.
[50] J. Binder, D. Koller, S. Russell, and K. Kanazawa, "Adaptive probabilistic networks with hidden variables," *Mach. Learn.*, vol. 29, no. 2–3, pp. 213–244, 1997.
[51] A. L. Jensen and F. V. Jensen, "MIDAS - An influence diagram for management of mildew in winter wheat," in *Proc. 12th Conf. Uncertainty Artif. Intell.*, North-Holland, 1996, pp. 349–356.
[52] S. Andreassen et al., "MUNIN - An expert EMG assistant," in *Computer-Aided Electromyography and Expert Systems*. Amsterdam, The Netherlands: Elsevier, 1989.
[53] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. L. Winkler, "Hailfinder: A Bayesian system for forecasting severe weather," *Int. J. Forecasting*, vol. 12, no. 1, pp. 57–71, 1996.
[54] I. Rish et al., "An empirical study of the naive Bayes classifier," in *Proc. 2001 Workshop Empirical Methods Artif. Intell.*, 2001, pp. 41–46.
[55] G. Fung, O. Mangasarian, and J. Shavlik, "Knowledge-based support vector machine classifiers," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 537–544.
[56] R. G. Cowell, P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter, *Probabilistic Networks and Expert Systems: Exact Computational Methods for Bayesian Networks*. Berlin, Germany: Springer Science & Business Media, 2007.
[57] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Statist. Assoc.*, vol. 32, no. 200, pp. 675–701, 1937.
[58] P. B. Nemenyi, *Distribution-Free Multiple Comparisons*. Princeton, NJ, USA: Princeton Univ., 1963.
[59] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.
[60] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. London, U.K.: Chapman and Hall/CRC, 2003.
[61] J. A. Rice, *Mathematical Statistics and Data Analysis*. Boston, MA, USA: Cengage Learning, 2006.
[62] D. Dheeru and E. K. Taniskidou, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml
[63] M. Little, P. McSharry, E. Hunter, J. Spielman, and L. Ramig, "Suitability of dysphonia measurements for telemonitoring of Parkinson's disease," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1015–1022, 2009.
[64] B. Hitt and P. Levine, "Multiple high-resolution serum proteomic features for ovarian cancer detection," U.S. Patent App. 11/093,018, Mar. 23 2006.

Zhaolong Ling received the PhD degree from the School of of Computer and Information, the Hefei University of Technology, China, in 2020. He is an assistant professor with the School of Computer Science and Technology, Anhui University, China. He has published more than ten papers in highly regarded journals, including *IEEE Transactions on Knowledge and Data Engineering*, *ACM Transactions on Intelligent Systems and Technology*, *ACM Transactions on Knowledge Discovery from Data*, *IEEE Transactions on Big Data*, etc. His research interests include feature selection, casual discover, and data mining.

Jingxuan Wu received the BS degree from Anqing Normal University, China, in 2021. Currently, he is working toward the MSc degree with the School of Computer Science and Technology, Anhui University, China. His research interests include feature selection, casual discovery, and data mining.

Yiwen Zhang received the PhD degree in management science and engineering from the Hefei University of Technology, Anhui, China, in 2013. He is a professor with the School of Computer Science and Technology, Anhui University. His current research interests include service computing, cloud computing, and Big Data.



Peng Zhou (Senior Member, IEEE) received the BE degree in computer science and technology from the University of Science and Technology of China in 2011, and the PhD degree in computer science from the Institute of Software, Chinese Academy of Sciences in 2017. He is currently an associate professor with the School of Computer Science and Technology, Anhui University. He has published more than 20 papers in highly regarded conferences and journals, including *IEEE Transactions on Neural Networks and Learning Systems*, *IEEE Transactions on Cybernetics*, *Pattern Recognition*, IJCAI, AAAI, SDM, ICDM, etc. His research interests include machine learning, data mining and artificial intelligence.

Xindong Wu (Fellow, IEEE) received the bachelor's and master's degrees in computer science from the Hefei University of Technology, China, and the PhD degree in artificial intelligence from the University of Edinburgh, Britain, in 1993. He is a Foreign member of the Russian Academy of Engineering, and the AAAS (American Association for the Advancement of Science). He is director and professor of the Key Laboratory of Knowledge Engineering with Big Data (the Ministry of Education of China), Hefei University of Technology, China. His research interests include Big Data analytics, data mining and knowledge engineering.

Kui Yu (Member, IEEE) received the PhD degree in computer science from the Hefei University of Technology, China, in 2013. From 2013 to 2015, he was a postdoctoral fellow with the School of Computing Science of Simon Fraser University, Canada. From 2015 to 2018, he was a research fellow with the University of South Australia, Australia. He is a professor with the School of Computer and Information, Hefei University of Technology. His main research interests include causal discovery and machine learning.

Xingyu Wu received the BS degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2018, and the PhD degree from University of Science and Technology of China (USTC), Hefei, China, in 2023. He is currently a postdoctoral fellow with the Department of Computing, The Hong Kong Polytechnic University. His research interests include causality-based machine learning, automatic machine learning, and large-scale pretraining model.