Gradient-based Causal Feature Selection

Zhaolong Ling [1], **Mengxiang Guo** [1], **Xingyu Wu** [2*], **Debo Cheng** [3], **Peng Zhou** [1], **Tianci Li** [1] and **Zhangling Duan**[4]
[1]Anhui University
[2]Hong Kong Polytechnic University
[3]University of South Australia
[4]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
zlling@ahu.edu.cn, ahu_guomengxiang@163.com, xingy.wu@polyu.edu.hk, debo.cheng@unisa.edu.au, doodzhou@ahu.edu.cn, y23301058@stu.ahu.edu.cn, duanzl1024@iai.ustc.edu.cn

Abstract

Causal feature selection leverages causal discovery techniques to identify critical features associated with a target variable using observational data. Traditional methodologies primarily rely on constraint-based or score-based techniques, which are fraught with limitations. For example, conditional independence tests often yield unreliable results in the presence of noise and complex data generation processes, while the computational complexity of learning directed acyclic graphs increases exponentially with the number of variables involved. In light of recent advancements in deep learning, gradient-based methods have shown promise for global causal discovery. However, significant challenges arise when focusing on the identification of local causal features, particularly in defining the local causal constraint space to achieve both minimality and completeness. To address these issues, we introduce a novel gradient-based causal feature selection method (GCFS) that leverages an AutoEncoder to simultaneously model the target variable alongside other variables, thereby capturing of causal associations within a divide-and-conquer framework. Additionally, our approach incorporates a mask pruning strategy that transforms the search process into the minimization of a non-cyclic local reconstruction loss objective function. This function is then effectively optimized using a gradient-based method to accurately identify the causal features related to the target variable. Experimental results substantiate that GCFS surpasses existing methodologies across both synthetic and real datasets.

1 Introduction

Causal feature selection[1], which uncovers causal relationships between variables, has been widely applied in various big data applications, including bioinformatics [Saeys *et al.*, 2007], neuroscience [Bielza and Larrañaga, 2014], and intelligent systems [Khan and Kuru, 2017]. By utilizing the Markov Blanket (MB), it identifies the direct causes, direct effects, and common causes of the target variable, leading to the construction of more accurate predictive models. This approach not only effectively reduces dimensionality and significantly decreases computational load but also improves the model's generalization ability. Under the faithfulness assumption, the MB ensures that the selected feature subset minimizes information redundancy while preserving sufficient information for the target variable [Ling *et al.*, 2024], thereby improving the robustness and interpretability of predictive models, as well as providing a reliable foundation for decision support [Ling *et al.*, 2025]. Existing causal feature selection methods can be roughly divided into two categories: constraint-based methods and score-based methods.

Constraint-based methods follow the Markov and faithfulness assumptions, relying on conditional independence (CI) tests. These methods employ strategies such as simultaneous, divide-and-conquer, or alternating approaches to learn MB [Yu *et al.*, 2020]. These methods improve learning efficiency by selecting relevant features and filtering out irrelevant ones. However, they are prone to generating incorrect MBs in the presence of noise or complex data generation mechanisms, where CI tests may yield inaccurate results [Huang *et al.*, 2023]. Score-based methods combine greedy search and scoring functions (e.g., K2 [Cooper and Herskovits, 1992] and BDeu [Buntine, 1991]) with Bayesian network (BN) structure [Kitson *et al.*, 2023] learning to determine the MB. The core idea of these methods is to learn directed acyclic graphs (DAGs) on the selected and newly added features, extracting MBs at each iteration. However, when the constrained search space is large, the time complexity of DAG learning can become excessively high, limiting its applicability to large-scale datasets [Wu *et al.*, 2020].

With the rapid development of deep learning, the field of causal discovery, which is closely related to causal feature selection, has stepped into the era of deep neural networks [Zeng *et al.*, 2021]. However, causal feature selection remains confined to early constraint-based and score-based methods, failing to effectively integrate modern deep

*Corresponding author
[1]https://github.com/MxGuoz/Appendix

learning techniques. This lag is largely due to the need for causal feature selection to impose strict discrete constraints on the minimal and complete interpretable causal neighborhood (i.e., MB) of the target variable. Existing gradient-based techniques typically focus solely on minimizing prediction errors or loss functions, fundamentally relying on statistical correlations in the data rather than uncovering underlying causal mechanisms [Jiao *et al.*, 2024]. Consequently, they offer limited interpretability. For gradient-based techniques to truly identify causal features, constraints must be properly defined, and exact causalization objectives must be formulated during the optimization process. However, existing differentiable causal discovery methods struggle to accurately eliminate redundancy in the global structure, leading to insufficient guarantees of the minimal and complete properties of causal features. Consequently, achieving the differentiable identification of causal features during the gradient optimization process of neural networks, while ensuring both the minimal and complete properties of causal features within a local causal constraint space, has become a pressing technical challenge.

Thus, we propose a gradient-based causal feature selection method to solve the above problems and to recognize MBs. Under the divide-and-conquer framework, GCFS employs AutoEncoder to simultaneously fit the target and other variables, extracting a weighted adjacency matrix during the message-passing process. To ensure the accurate capture of causality, we introduce DAG constraints to normalize the weighted adjacency matrix. In addition, GCFS integrates a mask pruning strategy to define a local reconstruction loss objective function with acyclic constraints, further facilitating the exploration of local causal relationships. By transforming the discretized search process into the minimization of this objective function, GCFS can leverage a gradient-based optimizer for efficient optimization, ultimately identifying the causal features of the target variable.

To accelerate the feature selection process and reduce the influence of redundant information from variable relationships, GCFS is divided into two phases: a search phase and a retraining phase. This structure allows GCFS to automatically select the optimal MB from all variables while maintaining the completeness of gradient-based training. In the search phase, we apply the $Gumbel\text{-}Max\ Trick$ strategy [Gumbel, 1954] to simulate hard selection during initial pre-training, thereby eliminating suboptimal variables that are not related to the MB of the target variable. The remaining candidate variables then proceed to the retraining phase for relationship reconstruction and further training. To further improve model efficiency, we introduce an ℓ_1 regularization term to promote sparsity in the weighted adjacency matrix and facilitate feature selection. Main contributions are summarized as follows:

- To our knowledge, the proposed GCFS is the first gradient-based algorithm for causal feature selection. GCFS addresses the challenge of identifying differentiable causal features in the optimization process of deep models. It establishes a local mechanism for causal constraint space to facilitate the application of gradient-based paradigms within local causal structures, ensuring both minimality and completeness of causal features.
- Different from existing methods, the proposed GCFS offers at least three practical benefits: 1) GCFS demonstrates excellent robustness in high-dimensional and complex data. 2) As the sample size increases, GCFS exhibits excellent scalability. 3) GCFS features a unified training framework, avoiding the fragmentation of information and inconsistencies from model partitioning.
- Extensive experiments on multiple synthetic and real datasets demonstrate that GCFS outperforms existing causal feature selection methods in accuracy, validating its effectiveness in causal feature selection tasks.

2 Related Works

In traditional causal structure learning, researchers narrow the search space by utilizing CI tests followed by scoring searches to identify the optimal network structure. A representative method is MMHC [Tsamardinos *et al.*, 2006], which, in its constraint phase, employs MMPC [Yasin and Leray, 2011] to add candidate variables to the parent-child set of target variable while eliminating redundancy. The maximization phase then derives the optimal DAG by combining TABU search with K2 scoring. Despite reducing the search space complexity, the number of DAGs still grows exponentially, and combinatorial optimization remains challenging. To address this, NOTEARS [Zheng *et al.*, 2018] introduces a continuous differentiable acyclicity constraint based on matrix trace properties, integrating it into the optimization process. This transformation shifts discrete combinatorial optimization into a differentiable problem, enabling DAG learning through gradient-based methods. While gradient-based methods improve DAG solution efficiency by using neural networks with acyclicity constraints, they encounter challenges when directly applied to causal feature selection. Causal feature selection demands strict constraints on the MB of target variables. However, existing methods struggle to globally eliminate redundant features, failing to fully guarantee the minimality and completeness of the MB's properties.

Current causal feature selection methods typically utilize CI tests and score-based search strategies, which are categorized into simultaneous and divide-and-conquer approaches. For example, IAMB [Tsamardinos *et al.*, 2003b] enhances the GS [Margaritis and Thrun, 1999] method by iteratively adding features most correlated with the target to the candidate MB set, thus improving accuracy. Variants such as FBED [Borboudakis and Tsamardinos, 2019] and EAMB [Guo *et al.*, 2022] have since been proposed. Although simultaneous methods are efficient, their data requirements increase exponentially with MB size. To mitigate the sample size requirement, MMMB [Tsamardinos *et al.*, 2003a] adopts a divide-and-conquer strategy, splitting the MB identification process into identifying PCs and spouses [Wu *et al.*, 2022]. Additionally, existing score-based MB learning methods are mainly variants of constraint-based methods that use BN structure learning to identify MBs. For instance, SLL [Niinimaki and Parviainen, 2012] employs BN structure learning to separately identify PCs and spouses, enhancing correctness through computationally intensive symmetric constraint checks. cIn contrast, S^2TMB [Gao and Ji,

2016] removes symmetric constraints to improve efficiency. However, regardless of whether simultaneous or divide-and-conquer strategies are used, these methods still depend on CI tests [Yu *et al.*, 2021]. Although score-based methods can avoid CI tests, the time cost of DAG learning becomes prohibitively high when faced with large search spaces, limiting their application to large-scale datasets [Wu *et al.*, 2023]. We provide more details in Appendix 1.

3 Notations

The structural causal model (SCM) is a mathematical framework for describing and analyzing causal relationships among variables in a set $\mathcal{V} = (X_1, X_2, \ldots, X_d)$, comprising a causal graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and structural equations. The causal graph is a DAG, where each edge in $\mathcal{E}$ represents a direct causal relationship between variables. For example, $X_i \to X_j$ indicates that X_i has a direct effect on X_j, where X_i is the parent variable of X_j. In this case, $X_j \to X_k$ implies that X_i indirectly influences X_k. Given a set of random variables $X = [X_1, X_2, \ldots, X_d]^T$, let $X_{pa(i)}$ denote the parent set of X_i in $\mathcal{G}$. According to the Markov condition of Bayesian networks, the joint distribution $P(X)$ can be expressed as a product of conditional probabilities:

$$P(X) = \prod_{i=1}^{d} P(X_i | X_{pa(i)}) \tag{1}$$

The Additive Noise Model (ANM) [Hoyer *et al.*, 2008] assumes that each variable X_i can be represented as a nonlinear function of its $X_{pa(i)}$ plus an independent noise term as:

$$X_i = f_i\big(X_{pa(i)}\big) + N_i, i = 1, 2, \ldots, d \tag{2}$$

where f_i is a nonlinear function, and N_i represents external noise. Due to the structural properties of ANM, N_i is independent of other variables outside $X_{pa(i)}$. Thus, $P(X)$ can be defined by the functions f_i and noise terms N_i in ANM.

The objective of causal feature selection methods is to identify the MB of a target variable X_T from the observed data x. This involves determining the direct causes (parents **P**), direct effects (children **C**), and common causes (spouses **SP**) associated with X_T. In this study, a divide-and-conquer strategy is adopted to decompose the MB learning problem into two subproblems. First, the **P** and **C** of X_T (PC_T) are learned. Second, the **SP** of X_T (SP_T) are identified. The detailed process is illustrated in Figure 1.

4 Gradient-based Causal Feature Selection

In this section, we provide a detailed description of the proposed method GCFS. We formalize the gradient-based modeling paradigm for MB and its optimization formulation in Section 4.1. Based on this, we detail the entire framework of GCFS, including a optimization module to accelerate the MB learning process in Section 4.2. Finally, we analyze the algorithm's computational flow in Section 4.3.

4.1 Formalization of Local MB Learning

NOTEARS proposes a smooth acyclicity constraint, namely $\operatorname{tr}\left(e^{A \circ A}\right) - d = 0$, where $\operatorname{tr}(\cdot)$ is the trace of a matrix, e^A represents the matrix exponential of A, and $\circ$ denotes the Hadamard product. This formulation transforms the combinatorial optimization problem in traditional causal discovery into a differentiable continuous optimization problem. The specific optimization form is as follows:

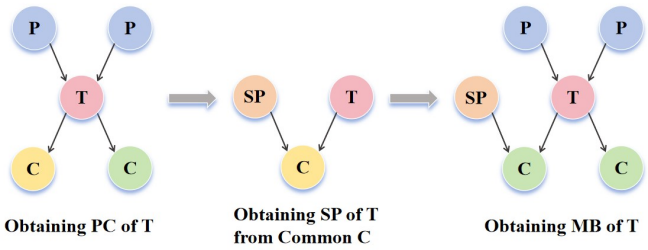

Figure 1: Illustration of the MB Learning Process.

$$\begin{aligned} \min_{A,\psi} \quad & \frac{1}{2n} \sum_{j=1}^{n} \left\| X^{(j)} - f(X^{(j)}, A) \right\|_F^2 + \lambda \|A\|_1 \\ \text{subject to} \quad & \operatorname{tr}\left(e^{A \circ A}\right) - d = 0 \end{aligned} \tag{3}$$

where n denotes the number of samples, $X^{(j)}$ represents the j-th observed sample, and $f(X^{(j)}, A)$ denotes the data generation model, with ψ being the parameters associated with f. Since NOTEARS utilizes a linear Structural Equation Model (SEM), $f(X^{(j)}, A)$ is defined as $A^T X^{(j)}$ in Eq. (3). When adopting a linear model for $f(X^{(j)}, A)$, the optimization in Eq. (3) is insufficient to capture the complex nonlinear relationships involving causal features. Moreover, the global structure obtained has limitations in effectively eliminating redundant features, failing to ensure the minimal completeness of the causal features. Hence, $f(X^{(j)}, A)$ can be extended by using a graph autoencoder, as shown in Eq. (4).

$$\begin{aligned} f(X^{(j)}, A) &= g_2\left(A^T g_1(X^{(j)})\right), \\ H^{(j)} &= g_1(X^{(j)}), H^{(j)\prime} = A^T H^{(j)}, \\ f(X^{(j)}, A) &= g_2(H^{(j)\prime}) \end{aligned} \tag{4}$$

where g_1: $\mathbb{R}^l \to \mathbb{R}^{l'}$ and g_2: $\mathbb{R}^{l'} \to \mathbb{R}^l$ are respectively the encoder and decoder (i.e., the AutoEncoder in Figure 2), both of which are implemented using multilayer perceptrons (MLPs). The dimensions l and l' represent the input feature dimension and the latent representation dimension, respectively. The encoder first transforms the input sample $X^{(j)}$ into a latent representation $H^{(j)}$. Then, $H^{(j)}$ undergoes a linear transformation by the matrix A^T to perform the message-passing process, yielding $H^{(j)\prime}$. Finally, the decoder reconstructs $H^{(j)\prime}$ into $\hat{X}$. Throughout the process, the message-passing mechanism is similar to a graph convolutional layer [Ng *et al.*, 2019]. The adjacency matrix aggregates information from neighboring variables into the latent representation of the target variable, enabling information propagation and representation learning on the graph. To identify the MBs associated with X_T more efficiently, it is necessary to define a suitable local search region and recover the local structure precisely by minimizing the local reconstruction error. To achieve this, we adopt a local graph construction method. This method

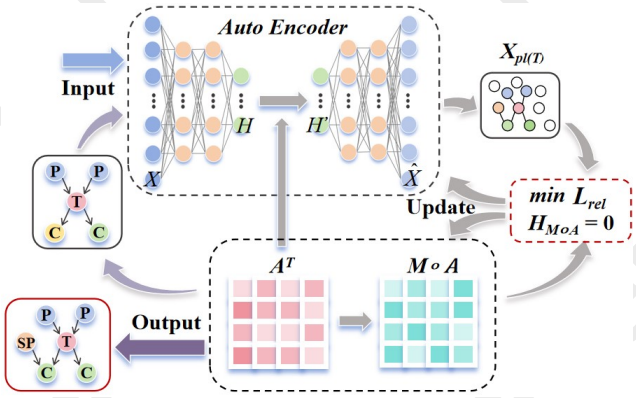

Figure 2: GCFS: Flowchart of the MB Model Framework.

identifies the potential relational neighborhood of the X_T (i.e., $X_{\mathrm{pl}(T)}$ in Figure 2) by gradually expanding its neighborhood based on the connectivity relationships among variables. Inspired by [Liang *et al.*, 2023], in this process, $\mathcal{Q}$ is used as the metric for $X_{\mathrm{pl}(T)}$. As $\mathcal{Q}$ increases, $X_{\mathrm{pl}(T)}$ gradually expands to include variables that are more broadly related to X_T, making the set of variables in $X_{\mathrm{pl}(T)}$ conditionally independent of other variables under the given conditions. The final optimization problem is formulated by minimizing the local reconstruction error, while adding an ℓ_1 regularization [Wang *et al.*, 2018] term to encourage the sparsity of matrix A. This problem can be formalized as:

$$\min_{A,\Phi_1,\Phi_2} \frac{1}{2n}\sum_{j=1}^{n}\left\|X^{(j)}_{\mathrm{pl}(T)} - \hat{X}^{(j)}_{\mathrm{pl}(T)}\right\|_F^2 + \lambda\|A\|_1 \tag{5}$$

where $\hat{X}^{(j)}_{\mathrm{pl}(T)}$ denotes the reconstructed output of $X^{(j)}_{\mathrm{pl}(T)}$. Additionally, Φ_1 and Φ_2 represent the weights of the encoder g_1 and the decoder g_2, respectively. The optimization Eq. (5) can identify the **PC** of each variable. However, this method only captures the correlation between the variables and fails to distinguish between **P** and **C** in the causal relationship. Although distinguishing between **P** and **C** is not necessary when searching for MB, it significantly narrows down the search space of **SP** when employing a divide-and-conquer strategy. By learning the parent set (i.e., **SP**) via the common subvariable, the efficiency of MB learning can be improved. Therefore, when optimizing the process of minimizing the local reconstruction error, it is still necessary to introduce the directed acyclic constraint in NOTEARS. By restricting the search space, it ensures that the generated graph structure remains acyclic, helping to determine the causal direction between variables. Since our focus is on learning the MB of X_T, we concentrate on the local graph of X_T rather than the entire graph. Thus, it is necessary to introduce a mask matrix [Ng *et al.*, 2022] to prune A, filtering out parts unrelated to X_T and focusing on optimizing the relevant parts to progressively uncover the structure of $X_{\mathrm{pl}(T)}$. The mask matrix M can be formalized as:

$$M_{jk} = \begin{cases} 0, & \text{if } X_j \notin \mathbf{X_{pl(T)}} \text{ and } X_k \notin \mathbf{X_{pl(T)}} \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

Each element M_{jk} of M indicates whether the edge between variables X_j and X_k is retained. If both X_j and X_k are not in $\mathbf{X_{pl(T)}}$, then $M_{jk} = 0$, indicating that the edge is removed; otherwise, $M_{jk} = 1$, meaning the edge is retained. After adjusting the constraint conditions, the final optimization objective is given by Eq. (7), where L_{rel} denotes the function combining the prediction error and a sparsity constraint. The MB model framework is illustrated in Figure 2.

$$\begin{aligned} &\min_{A,\Phi_1,\Phi_2} \frac{1}{2n}\sum_{j=1}^{n}\left\|X^{(j)}_{\mathrm{pl}(T)} - \hat{X}^{(j)}_{\mathrm{pl}(T)}\right\|_F^2 + \lambda\|M \circ A\|_1 \\ &\text{subject to} \quad h(M \circ A) = 0 \end{aligned} \tag{7}$$

The augmented Lagrangian method transforms the constrained optimization problem in Eq. (7) into an unconstrained optimization problem by introducing Lagrange multipliers and penalty terms [Bertsekas, 1997], as follows:

$$\begin{aligned} L_\rho(A,\Phi_{1,2},\mu) = &\frac{1}{2n}\sum_{j=1}^{n}\left\|X^{(j)}_{\mathrm{pl}(T)} - \hat{X}^{(j)}_{\mathrm{pl}(T)}\right\|_F^2 + \lambda\|M \circ A\|_1 \\ &+ \mu h(M \circ A) + \frac{\rho}{2}|h(M \circ A)|^2 \end{aligned} \tag{8}$$

where μ is the Lagrange multiplier that adjusts the influence of the constraint, and $\rho > 0$ is the penalty parameter that controls the strength of the penalty for the constraint term. By iteratively adjusting these parameters, the error is minimized while gradually satisfying the constraint conditions. The update rules for the iterative optimization are as follows:

$$A^{k+1}, \Phi_{1,2}^{k+1} \leftarrow \arg\min_{A,\Phi_1,\Phi_2} L_{\rho^k}(A,\Phi_{1,2},\mu^k), \tag{9}$$

$$\mu^{k+1} \leftarrow \mu^k + \rho^k h(M \circ A^{k+1}), \tag{10}$$

$$\rho^{k+1} \leftarrow \begin{cases} \beta\rho^k, & \text{if } |h(M \circ A^{k+1})| \geq \gamma |h(M \circ A^k)| \\ \rho^k, & \text{otherwise} \end{cases} \tag{11}$$

where $\beta > 1$ and $\gamma < 1$ are tuning hyperparameters used to control the growth rate of the penalty parameter and the threshold for error changes, respectively. Since Eq. (9) is a first-order differentiable optimization problem, it can be solved using the gradient descent method. In practice, the parameters can be updated efficiently by using the Autograd feature of deep learning frameworks such as TensorFlow with the Adam optimizer [Kinga *et al.*, 2015].

4.2 Framework of GCFS

In the model proposed in the previous section, we design the structure based on a local scope by treating all variables within the scope as inputs. By defining a local reconstruction loss function with acyclicity constraints and employing gradient descent, we obtain the MB of X_T. This approach is capable of fully capturing information within the local scope. Nevertheless, there may still be some redundant information, which increases the complexity of model training. To further optimize the model, we introduce an improved scheme that can automatically select the optimal MB from all variables, thereby reducing the influence of redundant information within the local scope. The overall improved framework is illustrated in Figure 3.

The method is divided into two phases: the search phase and the retraining phase. In the search phase, the framework's

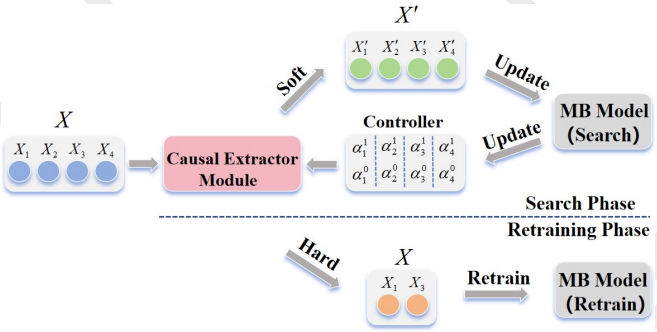

Figure 3: Overall Framework of GCFS.

parameters are initialized at first. Then, all variables X are input into the causal extractor module. This module applies weighted processing to the input variables using the controller module's parameters (α_n^1, α_n^0) and progressively optimizes these parameters through gradient descent to minimize the loss function. Ultimately, the importance of each variable to X_T can be incrementally determined in this way. This procedure can be framed as an expectation sampling problem, as follows:

$$\mathbf{X}'_n = (\alpha_n^1 \cdot 1 + \alpha_n^0 \cdot 0) \cdot X_n = \alpha_n^1 X_n \tag{12}$$
$$\mathbf{X}' = [\mathbf{X}'_1, \mathbf{X}'_2, \cdots, \mathbf{X}'_n] \tag{13}$$

where $(\alpha_n^1$ and $\alpha_n^0)$ represent the probabilities of the n-th variable being selected or discarded, respectively. Through this soft selection mechanism, the causal extractor module can dynamically adjust the weights of each variable during the search phase based on the current parameters of the controller. However, soft selection cannot completely eliminate the influence of suboptimal variables on the final MB, leading to discrepancies between the search and retraining phases. Hence, $Gumbel\text{-}Max\ Trick$ [Gumbel, 1954] is utilized to simulate the hard selection process based on controller parameters. Specifically, the hard selection process can be implemented as follows in Eq. (14).

$$\begin{aligned} &z_n = \text{one\_hot}\left(\text{argmax}\left[\log \alpha_n^0 + g_0, \log \alpha_n^1 + g_1\right]\right) \\ &\text{where } g_j = -\log\left(-\log\left(u_j\right)\right) \\ &u_j \sim \text{Uniform}(0,1) \quad \forall j \in [0,1] \end{aligned} \tag{14}$$

where g_0 and g_1 are independently and identically distributed Gumbel noises, and u_j is sampled from the uniform distribution $\text{Uniform}(0,1)$. Since z_n in Eq. (14) is obtained through the non-differentiable $argmax$ and one_hot operations [Kelley *et al.*, 2016], it is approximated by the $Softmax$ operation to apply the gradient optimization strategy, as follow:

$$p_n^j = \frac{\exp\left(\left(\log \alpha_n^j + g_j\right)/\tau\right)}{\exp\left(\left(\log \alpha_n^1 + g_1\right)/\tau\right) + \exp\left(\left(\log \alpha_n^0 + g_0\right)/\tau\right)} \tag{15}$$

where p_n^j denotes the probability of the n-th variable being selected $(j = 1)$ or discarded $(j = 0)$ and τ is the temperature parameter used to control the smoothness of the $Softmax$. By using p_n^j to simulate hard selection, the aim is to bridge the gap between the search phase and the retraining phase. Finally, this selection process can be written as:

$$\mathbf{X}'_n = (p_n^1 \cdot 1 + p_n^0 \cdot 0) \cdot X_n = p_n^1 X_n \tag{16}$$

Algorithm 1 GCFS

Require: x: data, X_T: target variable, θ_1, θ_2: hyperparameters $\Phi_1, \Phi_2, \mathcal{Q}, \lambda, \mu, \rho$: initial parameters
Ensure: [**P**, **C**, **SP**]: **MB** of X_T

1: $\mathbf{CMB_T} \leftarrow$ `SearchPhase`$(x, X_T, \Phi_1, \Phi_2, \mathcal{Q}, \lambda, \mu, \rho)$
2: Initialize Φ_1, Φ_2 of AutoEncoder for each $X_k \in \mathbf{CMB_T}$
3: **for** $t = 1, 2, \ldots$ **do**
4: $\quad X_{pl(T)}, \mathcal{M} \leftarrow$ `getMBGraph`$(A, \mathcal{Q})$
5: $\quad$ Optimize Eq. (9) using Adam algorithm
6: $\quad$ Update parameters μ and ρ by Eq. (10) and Eq. (11);
7: $\quad$ **if** $h(M \circ A) \leq \theta_1$ **or** $\rho \geq \theta_2$ **then**
8: $\quad\quad$ **break**
9: $\quad$ **end if**
10: **end for**
11: [**P**, **C**] $\leftarrow$ adjacency matrix A
12: **for** each $c \in$ **C** **do**
13: $\quad$ **SP** $\leftarrow$ Update adjacency matrix A'
14: $\quad$ **SP** $=$ **SP** $\setminus \{X_T\}$
15: **end for**
16: **return** [**P**, **C**, **SP**]

After causal extractor module, $\mathbf{X}'$ replaces the original variable set $\mathbf{X}$ as the input to enter the retraining phase. In the search phase, the model selects the top K features with the highest predictive ability (i.e., $\mathbf{X}'$). In the retraining phase, the model adjusts the input feature dimensions in the MB model of Section 4.1 by reducing the original dimension (containing all variables) to K dimensions, and then retrains based on these K features. During the training, the model once again minimizes the loss function via the back propagation algorithm and updates the parameters of each layer, ultimately identifying the MB of X_T.

4.3 Algorithm Analysis

Algorithm 1 summarizes the process of learning the MB of X_T using the GCFS algorithm. In line 1, the initial selection in the search phase aims to obtain the most predictive causal feature set $\mathbf{CMB_T}$ related to X_T. In line 2, each variable X_k from $\mathbf{CMB_T}$ is input and the parameters Φ_1 and Φ_2 of the AutoEncoder are re-initialized to reconstruct the relationships among the variables. Lines 3-10 perform multiple iterations to optimize the objective function. Specifically, in line 4, the algorithm calls the function $getMBGraph$ to identify the $X_{pl(T)}$ and M of X_T based on the current A and $\mathcal{Q}$. In line 5, the minimization of the objective function in Eq. (9) is performed by employing automatic differentiation in conjunction with the Adam optimizer. In line 6, μ and ρ are updated according to Eq. (10) and Eq. (11) to control the regularization strength and the DAG constraint. Lines 7-9 are used to check whether the iterative optimization process should be stopped early. If the stopping conditions are met, indicating convergence or numerical instability, the optimization process is terminated and the final A is output. In line 11, **P** and **C** of X_T are extracted based on the optimized A. Lines 12-15 are used to obtain **SP** of X_T. Specifically, in line 13, each sub-variable c in **C** is taken as the target, and lines 2-10 are executed again to get A'. Subsequently, **SP** of X_T is obtained by removing X_T from the parent set of c in line 14. Finally, **P**, **C** and **SP** together form **MB** of X_T.

Nodes	Algorithm	Size=1000 F1	Precision	Recall	CITs	Runtime	Size=5000 F1	Precision	Recall	CITs	Runtime
50	EEMB	0.44±0.04	**0.75**	0.37	215	0.03	0.54±0.03	0.71	0.53	411	0.03
	EAMB	0.44±0.04	0.71	0.38	104	**0.01**	0.54±0.03	0.68	0.54	117	**0.01**
	CFS-MI	0.28±0.03	0.68	0.21	98	**0.01**	0.22±0.02	0.54	0.17	84	**0.01**
	S^2TMB	0.12±0.01	0.14	0.10	**0**	41.33	0.13±0.00	0.19	0.12	**0**	42.89
	GCFS	**0.46±0.02**	0.70	**0.54**	**0**	85.34	**0.60±0.01**	**0.73**	**0.61**	**0**	143.92
100	EEMB	0.36±0.02	**0.68**	0.30	400	0.02	0.48±0.01	0.67	0.47	747	0.09
	EAMB	0.36±0.03	0.63	0.31	200	**0.01**	0.47±0.01	0.61	**0.50**	218	**0.02**
	CFS-MI	0.18±0.02	0.51	0.14	169	**0.01**	0.18±0.02	0.45	0.14	151	**0.02**
	S^2TMB	0.08±0.01	0.11	0.07	**0**	108.98	0.09±0.01	0.15	0.8	**0**	544.72
	GCFS	**0.41±0.03**	0.65	**0.39**	**0**	114.67	**0.53±0.00**	**0.77**	0.47	**0**	187.53
200	EEMB	0.35±0.01	**0.62**	0.30	907	0.04	0.45±0.01	0.62	0.42	1433	0.55
	EAMB	0.33±0.01	0.52	0.31	409	0.02	0.43±0.01	0.54	0.43	440	0.07
	CFS-MI	0.20±0.01	0.56	0.15	358	**0.01**	0.14±0.02	0.43	0.10	308	**0.06**
	S^2TMB	0.05±0.01	0.10	0.04	**0**	403.88	0.06±0.00	0.12	0.05	**0**	1968.34
	GCFS	**0.40±0.02**	0.59	**0.39**	**0**	141.82	**0.57±0.01**	**0.73**	**0.57**	**0**	265.17
500	EEMB	0.30±0.01	0.45	0.29	2888	**0.10**	0.40±0.01	0.49	0.43	4137	1.74
	EAMB	0.27±0.01	0.32	0.30	1041	0.25	0.36±0.01	0.37	0.44	1135	**0.08**
	CFS-MI	0.20±0.01	0.56	0.15	916	0.04	0.15±0.01	0.45	0.11	797	0.14
	S^2TMB	-	-	-	-	-	-	-	-	-	-
	GCFS	**0.44±0.04**	**0.68**	**0.49**	**0**	219.33	**0.55±0.01**	**0.68**	**0.58**	**0**	375.65
800	EEMB	0.29±0.01	0.38	0.31	5449	0.18	0.38±0.01	0.42	0.45	7553	0.63
	EAMB	0.24±0.00	0.24	0.32	1732	0.28	0.31±0.00	0.29	0.46	1871	**0.16**
	CFS-MI	0.23±0.01	0.59	0.17	1460	**0.05**	0.18±0.01	0.48	0.13	1287	0.22
	S^2TMB	-	-	-	-	-	-	-	-	-	-
	GCFS	**0.40±0.02**	**0.61**	**0.38**	**0**	265.49	**0.59±0.00**	**0.74**	**0.58**	**0**	453.71
1500	EEMB	0.25±0.01	0.29	0.31	13349	0.53	0.33±0.00	0.32	0.45	19848	1.46
	EAMB	0.18±0.00	0.15	0.32	4239	0.23	0.24±0.00	0.19	0.46	4303	1.12
	CFS-MI	0.24±0.01	0.60	0.18	2803	**0.13**	0.18±0.01	0.48	0.13	2411	**0.44**
	S^2TMB	-	-	-	-	-	-	-	-	-	-
	GCFS	**0.42±0.05**	**0.68**	**0.40**	**0**	351.27	**0.59±0.00**	**0.75**	**0.59**	**0**	595.86

Table 1: Performance of Different Algorithms on 1000 and 5000 Samples Generated by Nonlinear ANM with GPs (**Appendix 2.1** for details).

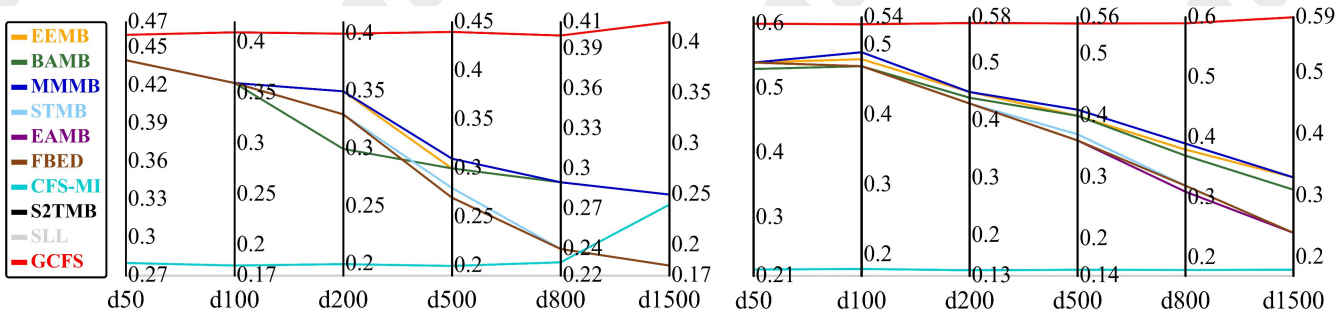

Figure 4: Comparison of F1 Scores for Various Algorithms under ANM with GPs. (Left: Size = 1000; Right: Size = 5000)

5 Experiments

To validate the effectiveness and accuracy of GCFS, we conducted experiments on various synthetic datasets generated by two distinct mechanisms as well as on real datasets. GCFS was compared against nine causal feature selection algorithms, including divide-and-conquer methods (MMMB, STMB [Gao and Ji, 2017]), simultaneous methods (FBED, EAMB), alternating PC-Spouse methods (BAMB [Ling *et al.*, 2019], EEMB [Wang *et al.*, 2020]), score-based methods (SLL, S^2TMB), and mutual information-based method (CFS-MI [Ling *et al.*, 2022a]). We used standard evaluation metrics: Precision measures the proportion of true positives (TP) among all outputs. Recall is the ratio of TP to the total number of actual positives. The F1 Score is the harmonic mean of Precision and Recall, where F1 = 1 is the best case and F1 = 0 is the worst case. [Xie *et al.*, 2024] CITs denote the number of conditional independence tests performed. Runtime refers to the algorithm's execution time. If an algorithm requires ≥ 1 hour for one run, its runtime is denoted as "−". The best results are highlighted in bold.

5.1 Synthetic Datasets

The DAGs of the synthetic datasets are generated using the ER model, with the number of nodes set as $d \in \{50, 100, 200, 500, 800, 1500\}$ and the number of edges set

to $2d$. For each DAG, data samples were generated with sizes $n \in \{1000, 5000\}$. Each sample group was subjected to 10 independent experiments, and the mean and standard deviation were recorded. To ensure reliability, each sample group underwent 10 independent experiments, recording the mean and standard deviation. [Ling *et al.*, 2022b] Data was generated using two mechanisms: ANM with GPs, which models causal relationships with Gaussian processes and adds noise, and Additive Model with GPs, which sums multiple independent Gaussian processes for each input. The noise term N_i was sampled from the $N(0, 1)$, introducing randomness into the data generation process [Xie *et al.*, 2022].

In the ANM with GPs data generation experiments, Table 1 and 2 present the performance of GCFS compared to other algorithms, while Figures 4 illustrates F1 score variations across different node scales and sample sizes. GCFS consistently outperforms existing methods in F1 scores across all node sizes and sample scales. For example, with 50 or 100 nodes, EEMB and EAMB achieve high precision but suffer from low recall, limiting their F1 scores. Conversely, GCFS maintains high precision and significantly boosts recall, demonstrating superior F1 performance. This advantage is even more evident with larger node scales (500 or 1500). Traditional CI test-based approaches falter due to diminished CI test reliability in high-dimensional and complex data, leading to lower F1 scores. In contrast, GCFS maintains high F1 scores in these challenging scenarios by leveraging gradient-based techniques and defining the local constraint spaces.

As the sample size increases, GCFS's F1 score improves markedly. For instance, with 800 nodes, increasing the sample size from 1000 to 5000 raises the F1 score by 47.5% (from 0.40 to 0.59), whereas other methods show minimal gains or even declines, such as CFS-MI. This suggests that larger sample sizes may amplify noise, adversely affecting mutual information metrics. While CFS-MI may occasionally achieve higher precision, it generally incurs more false positives or negatives, resulting in lower overall F1 scores. Traditional score-based methods, like S^2TMB, exhibit lower F1 scores and struggle with large node scales. In contrast, GCFS demonstrates excellent scalability on large-scale datasets and can effectively utilize more sample information to enhance the accuracy of causal feature identification.

Regarding CITs and runtime, GCFS performs zero CITs, indicating its differentiable causal feature selection avoids redundant conditional independence tests and mitigates MB errors in high-dimensional nonlinear settings, ensuring stable and precise causal identification. While GCFS demonstrates superior performance on high-dimensional nodes compared to score-based methods, its runtime is relatively longer. For example, with 50 nodes and 1000 samples, it takes 85.34 seconds compared to 0.01–0.1 seconds for CFS-MI and EAMB. This is due to GCFS's reliance on neural networks and multiple optimization steps to satisfy causal constraints. In the Additive Model with GPs experiments, GCFS also demonstrates better accuracy. More details are provided in Appendix 2.1.

5.2 Real-world Datasets

We conducted experiments on real biological datasets, specifically using the Sachs [Sachs *et al.*, 2005] dataset. The dataset records the expression levels of proteins and phospholipids in human cells using multi-parameter single-cell technology, and its ground truth network consists of 11 nodes and 17 edges. As a commonly used benchmark in graphical models, the Sachs dataset has a known consensus network (based on experimentally annotated gold standard networks), making it widely accepted in the biological community. Figure 5 and Table 3 visualize the learning results of GCFS on the Sachs dataset. The regions highlighted in red represent the correctly identified outcomes. The experimental results show that GCFS outperforms the comparison algorithms in precision (0.95), recall (0.56), and F1 score (0.67), demonstrating its advantage in accurately identifying causal features. Moreover, despite GCFS having a higher runtime (25.72 seconds) than constraint-based methods, it still demonstrates faster processing speed compared to the SLL algorithm.

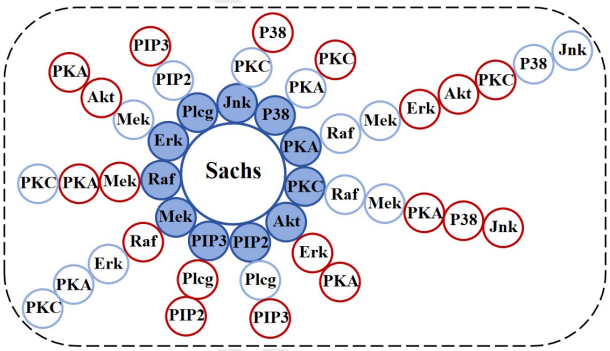

Figure 5: Real Results of GCFS for Learning MB Among Different Molecules on the Sachs Dataset.

Algorithm	F1	Precision	Recall	CITs	Runtime
EEMB	0.64	0.91	0.54	24	0.07
EAMB	0.64	0.91	0.54	20	**0.01**
CFS-MI	0.41	0.70	0.33	22	0.18
SLL	0.14	0.18	0.12	**0**	47.29
GCFS	**0.67**	**0.95**	**0.56**	**0**	25.72

Table 2: Results on the Sachs' Protein Signaling Network (**Appendix 2.2** for details).

6 Conclusion

In this paper, we propose a novel causal feature selection method, GCFS, which utilizes gradient descent methods and neural networks to explore the MB of the target variable. To effectively delineate the local causal constraint space for achieving minimality and completeness, we incorporate acyclicity constraints and mask matrices. Additionally, to accelerate the MB learning process, we apply the $Gumbel\text{-}Max\ Trick$ strategy to simulate hard selection during the search phase. Through experiments conducted on both synthetic and real datasets, GCFS has demonstrated strong competitiveness in causal feature selection tasks, particularly achieving significant improvements in precision. Future work includes extending GCFS to support real-time incremental causal feature selection and enhancing its robustness to missing data via causal inference-based imputation.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (under grant 62306002, 62376001, and 62120106008), and the Natural Science Foundation of Anhui Province of China under Grant 2108085QF270.

References

[Bertsekas, 1997] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

[Bielza and Larrañaga, 2014] Concha Bielza and Pedro Larrañaga. Bayesian networks in neuroscience: a survey. *Frontiers in computational neuroscience*, 8:131, 2014.

[Borboudakis and Tsamardinos, 2019] Giorgos Borboudakis and Ioannis Tsamardinos. Forward-backward selection with early dropping. *Journal of Machine Learning Research*, 20(8):1–39, 2019.

[Buntine, 1991] Wray Buntine. Theory refinement on bayesian networks. In *Uncertainty proceedings 1991*, pages 52–60. Elsevier, 1991.

[Cooper and Herskovits, 1992] Gregory F Cooper and Edward Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9:309–347, 1992.

[Gao and Ji, 2016] Tian Gao and Qiang Ji. Efficient markov blanket discovery and its application. *IEEE transactions on Cybernetics*, 47(5):1169–1179, 2016.

[Gao and Ji, 2017] Tian Gao and Qiang Ji. Efficient score-based markov blanket discovery. *International Journal of Approximate Reasoning*, 80:277–293, 2017.

[Gumbel, 1954] Emil Julius Gumbel. *Statistical theory of extreme values and some practical applications: a series of lectures*, volume 33. US Government Printing Office, 1954.

[Guo *et al.*, 2022] Xianjie Guo, Kui Yu, Fuyuan Cao, Peipei Li, and Hao Wang. Error-aware markov blanket learning for causal feature selection. *Information Sciences*, 589:849–877, 2022.

[Hoyer *et al.*, 2008] Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

[Huang *et al.*, 2023] Shanshan Huang, Qingsong Li, Lei Wang, Yuanhao Wang, and Li Liu. Score-based causal feature selection for cancer risk prediction. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pages 198–203, 2023.

[Jiao *et al.*, 2024] Licheng Jiao, Yuhan Wang, Xu Liu, Lingling Li, Fang Liu, Wenping Ma, Yuwei Guo, Puhua Chen, Shuyuan Yang, and Biao Hou. Causal inference meets deep learning: A comprehensive survey. *Research*, 7:0467, 2024.

[Kelley *et al.*, 2016] David R Kelley, Jasper Snoek, and John L Rinn. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome research*, 26(7):990–999, 2016.

[Khan and Kuru, 2017] Wasiq Khan and Kaya Kuru. An intelligent system for spoken term detection that uses belief combination. *IEEE Intelligent Systems*, 32(1):70–79, 2017.

[Kinga *et al.*, 2015] D Kinga, Jimmy Ba Adam, et al. A method for stochastic optimization. In *International conference on learning representations (ICLR)*, volume 5, page 6. San Diego, California;, 2015.

[Kitson *et al.*, 2023] Neville Kenneth Kitson, Anthony C Constantinou, Zhigao Guo, Yang Liu, and Kiattikun Chobtham. A survey of bayesian network structure learning. *Artificial Intelligence Review*, 56(8):8721–8814, 2023.

[Liang *et al.*, 2023] Jiaxuan Liang, Jun Wang, Guoxian Yu, Carlotta Domeniconi, Xiangliang Zhang, and Maozu Guo. Gradient-based local causal structure learning. *IEEE Transactions on Cybernetics*, 54(1):486–495, 2023.

[Ling *et al.*, 2019] Zhaolong Ling, Kui Yu, Hao Wang, Lin Liu, Wei Ding, and Xindong Wu. Bamb: A balanced markov blanket discovery approach to feature selection. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(5):1–25, 2019.

[Ling *et al.*, 2022a] Zhaolong Ling, Ying Li, Yiwen Zhang, Kui Yu, Peng Zhou, Bo Li, and Xindong Wu. A light causal feature selection approach to high-dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7639–7650, 2022.

[Ling *et al.*, 2022b] Zhaolong Ling, Kui Yu, Yiwen Zhang, Lin Liu, and Jiuyong Li. Causal learner: A toolbox for causal structure and markov blanket learning. *Pattern Recognition Letters*, 163:92–95, 2022.

[Ling *et al.*, 2024] Zhaolong Ling, Jingxuan Wu, Yiwen Zhang, Peng Zhou, Xingyu Wu, Kui Yu, and Xindong Wu. Label-aware causal feature selection. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[Ling *et al.*, 2025] Zhaolong Ling, Jiale Yu, Yiwen Zhang, Debo Cheng, Peng Zhou, Xingyu Wu, Bingbing Jiang, and Kui Yu. Local causal discovery without causal sufficiency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 18737–18745, 2025.

[Margaritis and Thrun, 1999] Dimitris Margaritis and Sebastian Thrun. Bayesian network induction via local neighborhoods. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, pages 505–511, 1999.

[Ng *et al.*, 2019] Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*, 2019.

[Ng *et al.*, 2022] Ignavier Ng, Shengyu Zhu, Zhuangyan Fang, Haoyang Li, Zhitang Chen, and Jun Wang. Masked

gradient-based causal structure learning. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pages 424–432. SIAM, 2022.

[Niinimaki and Parviainen, 2012] Teppo Niinimaki and Pekka Parviainen. Local structure discovery in bayesian networks. *arXiv preprint arXiv:1210.4888*, 2012.

[Sachs *et al.*, 2005] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

[Saeys *et al.*, 2007] Yvan Saeys, Iñaki Inza, and Pedro Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 08 2007.

[Tsamardinos *et al.*, 2003a] Ioannis Tsamardinos, Constantin F Aliferis, and Alexander Statnikov. Time and sample efficient discovery of markov blankets and direct causal relations. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 673–678. ACM, 2003.

[Tsamardinos *et al.*, 2003b] Ioannis Tsamardinos, Constantin F Aliferis, Alexander R Statnikov, and Er Statnikov. Algorithms for large scale markov blanket discovery. In *FLAIRS conference*, volume 2, pages 376–380, 2003.

[Tsamardinos *et al.*, 2006] Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. The max-min hill-climbing bayesian network structure learning algorithm. *Machine learning*, 65:31–78, 2006.

[Wang *et al.*, 2018] Yao Wang, Deyu Meng, and Ming Yuan. Sparse recovery: from vectors to tensors. *National Science Review*, 5(5):756–767, 2018.

[Wang *et al.*, 2020] Hao Wang, Zhaolong Ling, Kui Yu, and Xindong Wu. Towards efficient and effective discovery of markov blankets for feature selection. *Information Sciences*, 509:227–242, 2020.

[Wu *et al.*, 2020] Xingyu Wu, Bingbing Jiang, Kui Yu, Huanhuan Chen, et al. Accurate markov boundary discovery for causal feature selection. *IEEE transactions on cybernetics*, 50(12):4983–4996, 2020.

[Wu *et al.*, 2022] Xingyu Wu, Bingbing Jiang, Yan Zhong, and Huanhuan Chen. Multi-target markov boundary discovery: Theory, algorithm, and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4964–4980, 2022.

[Wu *et al.*, 2023] Xingyu Wu, Bingbing Jiang, Xiangyu Wang, Taiyu Ban, and Huanhuan Chen. Feature selection in the data stream based on incremental markov boundary learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10):6740–6754, 2023.

[Xie *et al.*, 2022] Feng Xie, Biwei Huang, Zhengming Chen, Yangbo He, Zhi Geng, and Kun Zhang. Identification of linear non-gaussian latent hierarchical structure. In *International Conference on Machine Learning*, pages 24370–24387. PMLR, 2022.

[Xie *et al.*, 2024] Feng Xie, Zheng Li, Peng Wu, Yan Zeng, Chunchen Liu, and Zhi Geng. Local causal structure learning in the presence of latent variables. *arXiv preprint arXiv:2405.16225*, 2024.

[Yasin and Leray, 2011] Amanullah Yasin and Philippe Leray. immpc: A local search approach for incremental bayesian network structure learning. In *Advances in Intelligent Data Analysis X*, pages 401–412, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[Yu *et al.*, 2020] Kui Yu, Xianjie Guo, Lin Liu, Jiuyong Li, Hao Wang, Zhaolong Ling, and Xindong Wu. Causality-based feature selection: Methods and evaluations. *ACM Computing Surveys (CSUR)*, 53(5):1–36, 2020.

[Yu *et al.*, 2021] Kui Yu, Lin Liu, and Jiuyong Li. A unified view of causal and non-causal feature selection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(4):1–46, 2021.

[Zeng *et al.*, 2021] Yan Zeng, Shohei Shimizu, Ruichu Cai, Feng Xie, Michio Yamamoto, and Zhifeng Hao. Causal discovery with multi-domain lingam for latent factors. In *Causal Analysis Workshop Series*, pages 1–4. PMLR, 2021.

[Zheng *et al.*, 2018] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.