Zhao and Jia
BMC Medical Research Methodology (2025) 25:154
https://doi.org/10.1186/s12874-025-02582-6

BMC Medical Research Methodology

RESEARCH Open Access

DAGSLAM: causal Bayesian network structure learning of mixed type data and its application in identifying disease risk factors

Yuanyuan Zhao[1] and Jinzhu Jia[1,2*]

Abstract

Background Identifying and understanding disease risk factors is crucial in epidemiology, particularly for chronic and noncommunicable diseases that often have complex interrelationships. Traditional statistical methods struggle to capture these complexities, necessitating more sophisticated analytical frameworks. Bayesian networks and directed acyclic graphs (DAGs) provide powerful tools for exploring the complex relationships between variables. However, existing DAG structure learning algorithms still have limitations in handling mixed-type data (including continuous and discrete variables), which restricts their practical utility. Therefore, developing DAG structure learning methods that can effectively handle mixed data is highly important for obtaining an in-depth understanding of disease risk factors and pathogenic mechanisms.

Methods This study proposes an extension of the NOTEARS algorithm, termed DAGSLAM, which is designed for Bayesian network structure learning with mixed-type data. The algorithm integrates continuous and categorical variables through a tailored loss function, enhancing its applicability to real-world epidemiological datasets.

Results Extensive simulations were conducted across eight distinct scenarios, specifically, variations in the number of nodes, changes in the proportion of categorical nodes, different sample sizes, levels of categorical nodes, variations in edge sparsity, adjustments to the weight scale, different graph types, and diverse noise distributions. These scenarios demonstrate that DAGSLAM consistently outperforms existing methods such as HC, TABU, mDAG, and DAGBagM across key metrics, including precision, recall, F1 score, and structural Hamming distance (SHD). Furthermore, the robustness of DAGSLAM is validated through its application to the National Health and Nutrition Examination Survey (NHANES) dataset, revealing critical causal relationships among risk factors for CHD and diabetes.

Conclusions DAGSLAM provides a powerful and scalable tool for uncovering causal relationships in complex disease networks, with significant implications for risk factor identification and public health research.

Keywords Bayesian networks (BNs), Directed cyclic graphs (DAGs), Causal inference, Risk factor

*Correspondence:
Jinzhu Jia
jzjia@math.pku.edu.cn
[1] Department of Biostatistics, School of Public Health, Peking University, 38 Xueyuan Road, Beijing 100191, China
[2] Center for Statistical Science, Peking University, Beijing 100871, China

Background

In the field of epidemiology, identifying and understanding the risk factors for diseases is a cornerstone of public health research. Many diseases, particularly chronic and noncommunicable conditions, do not occur in isolation but are often interconnected through shared risk factors and complex comorbidities. For example, metabolic disorders, cardiovascular conditions, and other chronic diseases frequently cooccur, forming intricate networks of interactions that challenge traditional approaches to disease prevention and management [1, 2]. These interactions are further complicated by the multifactorial nature of many diseases, where genetic, behavioral, and environmental factors converge to influence disease onset and progression [3]. Understanding these relationships is critical for developing effective prevention strategies and improving population health outcomes.

Traditional statistical methods, such as logistic regression and Cox proportional hazards models, have been widely used to identify associations between individual risk factors and disease outcomes. While these methods are effective for analysing simple, linear relationships, they often fail to capture the complexity of multifactorial diseases and comorbidities. Specifically, they struggle to model the conditional dependencies among multiple variables and disentangle direct and indirect effects in complex networks [4, 5]. Moreover, these methods typically require strong assumptions about the independence of variables and the directionality of relationships, which may not hold in real-world epidemiological data [6]. As a result, there is a growing need for more sophisticated analytical frameworks that can model the intricate causal relationships underlying disease risk.

Bayesian networks (BNs) have emerged as promising tools for addressing these challenges. BNs are probabilistic graphical models that represent variables and their conditional dependencies through Directed cyclic graphs (DAGs) [7]. Unlike traditional statistical methods, BNs can model interactions among multiple variables and provide insights into the causal structure of disease risk factors [6, 8]. These features make BNs particularly well suited for studying complex disease networks and identifying key drivers of comorbidities [9–11]. However, constructing BNs from real-world data is a nontrivial task. Manual construction of causal models on the basis of expert knowledge is not only time-consuming but also inherently prone to bias, as it relies heavily on subjective interpretations and assumptions [12–14]. This process can lead to inconsistencies and inaccuracies in the resulting models, particularly when dealing with complex interrelationships among variables. On the other hand, automated structure learning algorithms have emerged as promising alternatives for discovering causal relationships from observational data [8, 15–18]. However, these algorithms often face significant limitations, especially in regard to handling mixed data types, which include both continuous and categorical variables.

In real-world scenarios, observational data typically comprise a mixture of variable types. For example, clinical outcomes are often represented as binary variables—such as the presence or absence of a disease—while potential biomarkers may be continuous variables, such as the expression levels of specific proteins in the body. While substantial progress has been made in causal structure learning for purely discrete or continuous data, the study of mixed-type data remains relatively underexplored [19–21]. Most existing methods are designed under the assumption that all nodes conform to the same type of distribution, which restricts their applicability to either purely discrete or purely continuous datasets [8, 13, 15, 22–27]. This limitation significantly hampers the ability to jointly model continuous and categorical variables, thereby constraining the practical utility of these methods in complex epidemiological research.

To address these limitations, this study proposes an extension of the NOTEARS (Non-combinatorial Optimization via Trace Exponential and Augmented lagRangian for Structure learning) algorithm, referred to as DAGSLAM (Directed Acyclic Graphs Structure learning via Log-determinant and Augmented lagrangian for Mixed type data), which is specifically designed for BN structure learning with mixed-type data [28]. The proposed method leverages the strengths of BNs in modelling complex causal relationships while overcoming the constraints of existing algorithms. By enabling the integration of both continuous and categorical variables, DAGSLAM provides a more flexible and scalable framework for analysing real-world epidemiological data.

The key contributions of this study are as follows:

- Development of the DAGSLAM algorithm for BN structure learning with mixed-type data addresses a critical gap in existing methodologies.
- Extensive simulation experiments were conducted in various scenarios to validate the robustness of the DAGSLAM algorithm, demonstrating its effectiveness across different conditions and settings.
- The proposed framework was applied to identify risk factors and causal relationships in complex disease

networks, demonstrating its utility in epidemiological research.

The findings of this study will contribute to a better understanding of the complex causal mechanisms underlying disease risk and comorbidities, serving as a foundation for future research and clinical applications in the field of risk factor identification and prevention.

Related works

DAG structure learning has been a long-standing challenge in machine learning and causal inference. Existing algorithms can be broadly categorized into three types: score-based, constraint-based, and hybrid methods [29]. Score-based approaches, such as Hill Climbing (HC) and Tabu Search, aim to optimize a predefined score function over the space of DAGs [15, 23, 30]. Constraint-based methods, like the PC algorithm, rely on conditional independence tests to infer the graph structure [8]. Hybrid methods, such as Max–Min Hill-Climbing (MMHC), combine both score-based and constraint-based strategies to improve robustness [16]. While these methods have been widely used, they often face challenges in scalability and computational efficiency due to the combinatorial nature of the DAG space.

Recently, gradient-based DAG learning has emerged as a promising alternative, leveraging continuous optimization techniques to address the NP-hard problem of DAG discovery. One of the pioneering works in this direction is NOTEARS [28], which reformulates the DAG learning problem as a continuous optimization task by introducing a smooth acyclicity constraint. NOTEARS uses least squares for linear structural equation models (SEMs) and has shown significant improvements in both accuracy and scalability compared to traditional methods.

While NOTEARS is effective for linear SEMs, it is limited in its ability to capture complex, non-linear relationships. To address this, DAG-GNN extends the framework by employing a deep generative model based on variational autoencoders (VAEs) [31]. The key innovation of DAG-GNN lies in its use of graph neural networks (GNNs) to parameterize the encoder and decoder, allowing the model to capture non-linear dependencies in the data. Similarly, GraN-DAG employs neural networks to learn non-linear relationships while maintaining the benefits of continuous optimization [32]. Another notable approach, NOBEARS, introduces polynomial models to infer causal relationships in high-dimensional settings, such as transcriptome networks [33].

Further advancements include NOTEARS-nonlinear, which generalizes NOTEARS to non-parametric DAGs, allowing for more flexible modeling of complex data distributions [34]. More recently, DAGMA proposes a log-determinant acyclicity constraint, which is computationally efficient and exhibits better-behaved gradients compared to the exponential constraint used in NOTEARS [35]. Additionally, GFlowNets introduces a generative flow network framework for DAG learning, offering a novel perspective on sampling and optimization in the DAG space [36].

Despite these advancements, several limitations remain. While many gradient-based algorithms have successfully extended to non-linear SEMs, there is still a lack of robust methods for handling mixed data types (e.g., continuous and categorical variables) in linear settings. This gap is particularly relevant in applications such as biomedical research, where datasets often contain heterogeneous variables. To address this challenge, we propose DAGSLAM, an extension of NOTEARS specifically designed for mixed-type data. By integrating linear regression for continuous variables and logistic regression for binary variables, DAGSLAM provides a unified framework for learning DAGs from diverse data types, filling a critical gap in the current literature.

Methods

Overview and notations of DAG learning problems

Suppose that $X \in \mathbb{R}^{n \times d}$ is a data matrix that contains n independent and identically distributed (i.i.d.) instances of the random vector $\boldsymbol{X} = (X_1, \cdots, X_d)$. In this context, X_j represents a random variable that can be continuous or discrete, and x_{ij} denotes the value of a random variable. The index i is used to distinguish among different instances of the data, whereas j is used to identify different random variables. We define an index set $\boldsymbol{V} := \{1, \ldots, d\}$ to represent the indexes of different nodes (or random variables). Additionally, we define $\boldsymbol{C}$ as an index set of continuous variables and $\boldsymbol{D}$ as an index set of discrete variables. Thus, it follows that $\boldsymbol{V} = \boldsymbol{C} \cup \boldsymbol{D}$.

Let $\mathbb{D}$ denote the discrete space of the directed acyclic graph $G = (\boldsymbol{V}, \boldsymbol{E})$ on d nodes, where $\boldsymbol{V}$ denotes the set of nodes and where $\boldsymbol{E} \subset \boldsymbol{V} \times \boldsymbol{V}$ denotes the set of edges. For $(i,j) \in E$, if X_i is the parent node of X_j, it is represented as $X_i \rightarrow X_j$.

Within the linear SEM utilized by the NOTEARS algorithm, each X_j is formulated as a linear combination of $pa(X_j)$. Each DAG is encoded by a weighted adjacency matrix $W \in \mathbb{R}^{d\times d}$, where $W_{ij} \neq 0$ indicates that X_i acts as a parent node for X_j. If $W_{ij} = 0$ and $W_{ji} = 0$, there is no edge connecting X_i and X_j.

In the original NOTEARS algorithm, the score function $F(W)$ is defined as the least-squares (LS) loss combined with an ℓ_1-regularization term to encourage sparsity in the estimated graph. Specifically, the score function is given by [28]:

$$F(W) = \frac{1}{2n}\|\boldsymbol{X} - \boldsymbol{X}W\|_F^2 + \lambda\|W\|_1, \quad (1)$$

where $\|\boldsymbol{X} - \boldsymbol{X}W\|_F^2$ is the Frobenius norm of the residual matrix, $\|W\|_1$ is the ℓ_1-norm of the weighted adjacency matrix W, and λ is a regularization parameter controlling the sparsity of the graph. The least-squares loss does not assume a specific noise distribution and can be applied to linear SEMs with both Gaussian and non-Gaussian noise. Moreover, the framework is designed to be flexible, enabling the incorporation of other loss functions tailored to different data types and modeling assumptions.

According to the score-based method, we seek to solve the following combinational optimization program for the optimal DAG structure:

$$\min_{W\in\mathbb{R}^{d\times d}} F(W) \quad s.t.\ G(W) \in DAGs, \quad (2)$$

where $F(W)$ is a score function (i.e., a loss function) and where $G(W)$ refers to the graph with d nodes generated by the weighted adjacency matrix W.

To address the combinatorial nature of the problem, the NOTEARS framework introduces a continuous optimization approach. Specifically, it transforms the discrete acyclicity constraint into an equivalent continuous constraint [28]:

$$\min_{W\in\mathbb{R}^{d\times d}} F(W) \quad s.t.\ h(W) = 0, \quad (3)$$

In the original NOTEARS formulation, the acyclicity constraint is defined as $h(W) = \mathrm{tr}\left(e^{W\circ W}\right) - d = 0$, where $\circ$ refers to the Hadamard product, and $e^{W\circ W}$ is the matrix exponential of $W \circ W$. This constraint ensures that the learned graph is acyclic, and its derivative $\nabla h(W) = \left(e^{W\circ W}\right)^T \circ 2W$ is straightforward to compute.

Building on NOTEARS, the DAGMA algorithm [35] proposes a log-determinant constraint as an alternative:

$$h(W) = -\mathrm{logdet}(sI - W \circ W) + d\ \log s = 0, \quad (4)$$

where s is a positive scalar parameter. This constraint leverages the M-matrix property of $sI - W \circ W$ ensuring that the learned graph is acyclic. The log-determinant constraint has a simple derivative form:

$$\nabla h(W) = 2(sI - W \circ W)^{-T} \circ W, \quad (5)$$

which is computationally efficient and well-behaved during optimization. Compared to the matrix exponential-based constraint, the log-determinant constraint offers several advantages, including better gradient behaviour, faster computation, and improved cycle detection, especially for large graphs.

To date, the combinational optimization problem has drastically transformed to a continuous optimization program, which can be conveniently addressed via conventional numerical solution methods such as gradient descent.

Structural equation modelling of mixed-type data

We first construct SEMs for both continuous variables and discrete variables. A continuous variable X_j $(j \in \boldsymbol{C})$ can be modelled as:

$$X_j = w_j^T X + \epsilon_j, \quad (6)$$

where w_j denotes the j-th column of W. It follows that $w_{ij} = 0$ when X_i is not a parent node X_j, and vice versa. ϵ_j is a random noise term with a zero mean and a variance of σ^2. Note that we do not need to assume that ϵ_j follows a normal distribution.

For a binary variable X_k $(k \in \boldsymbol{D})$, the probability of $X_k = 1$ can be given by a logistic regression model:

$$P(X_k = 1) = \frac{\exp\left(w_k^T X\right)}{1+\exp\left(w_k^T X\right)}, \quad (7)$$

where w_k denotes the kth column of W.

Similarly, for a polytomous variable X_k $(k \in \boldsymbol{D})$, a multinomial logistic regression model can be posited:

$$P(X_k = l) = \frac{\exp\left(w_{k(l)}^T X\right)}{\sum_{m=1}^{M} \exp\left(w_{k(m)}^T X\right)}, \quad (8)$$

where $l = 1,2,\cdots,M$, with M being the number of categories for the polytomous variable. For instance, when X_k is a three-class variable, $M = 3$.

Our purpose is to learn the best latent structure of the DAG given the observational data generated from its intrinsic mechanism corresponding to the DAG.

Therefore, it is intuitive to define the loss function as the discrepancy between the true value of every X_j and its fitted value of our models.

Specifically, for a continuous variable $X_j (j \in \boldsymbol{C})$, we define its loss function as:

$$L_j = \|\boldsymbol{X}_\mathbf{i} - w_j^T\boldsymbol{X}\|_2^2, \tag{9}$$

where $w_{jj} = 0$. The loss measures how well the fit is achieved by utilizing all other variables to estimate the continuous variable X_j, which is analogous to the least square loss in the linear regression literature.

For a binary variable X_k $(k \in \boldsymbol{D})$, its loss function can be given by:

$$L_k = -\langle \mathbf{X_k}, w_k^T\boldsymbol{X}\rangle + \log\left(1 + \exp\left(w_k^T\boldsymbol{X}\right)\right), \tag{10}$$

where $w_{kk} = 0$, which is exactly the cross-entropy loss function in the binary classification problem or the negative log-likelihood function in the logistic regression model.

Similarly, for a polytomous variable X_k $(k \in \boldsymbol{D})$, its loss function can be given as follows:

$$L_k = -\sum_{m=1}^{M} \boldsymbol{I}(\mathbf{X_k} = m) \cdot w_{k(m)}^T\boldsymbol{X} + \log\left(\sum_{m=1}^{M} \exp\left(w_{k(m)}^T\boldsymbol{X}\right)\right), \tag{11}$$

where $w_{k(m)}^T$ denotes the coefficient of X in the model of $P(X_k = m)$. This is the same as the cross-entropy loss function in the multiclassification problem or the negative log-likelihood function in the multinomial logistic regression model.

Finally, the total loss function can be defined as the sum of continuous variables and categorical (both binary and polytomous) variables:

$$L(W) = \sum_{j \in \boldsymbol{C}} L_j + \sum_{k \in \boldsymbol{D}} L_k, \tag{12}$$

which is a function of the weighted adjacency matrix W.

The optimization program

After defining the loss function of mixed-type data, we aim to find the optimal W that minimizes the value of the loss function $L(W)$ by searching through the space of all possible weighted adjacency matrices W for DAGs within the 'acyclic' constraint. To this end, we need to solve the following continuous optimization program:

$$\min_{W \in \mathbb{R}^{d \times d}} L(W) \quad s.t.\ h(W) = 0, \tag{13}$$

where $h(W) = -\text{logdet}(sI - W \circ W) + d\log s = 0$.

This equality-constrained program (ECP) can be solved via the augmented Lagrangian method, converting (13) to the following unconstrained program:

$$\min_{W \in \mathbb{R}^{d \times d}} L^{\rho}(W, \alpha), \tag{14}$$

$$\text{where } L^{\rho}(W, \alpha) = L(W) + \frac{\rho}{2}|h(W)|^2 + \alpha h(W), \tag{15}$$

with penalty parameter $\rho > 0$ and Lagrange multiplier α.

When ρ is sufficiently large and α is appropriately chosen, the solution obtained from the unconstrained program (14) closely approximates that of the original program (13) [37].

The Lagrange multiplier α is updated iteratively via the following formula:

$$\alpha \leftarrow \alpha + \rho h\left(W_\alpha^*\right), \tag{16}$$

where W_α^* denotes the optimizer of program (14) at a given α.

To ensure that the learned DAG is a sparse network, an L1 regularization can be added to the loss function, resulting in the following optimization problem:

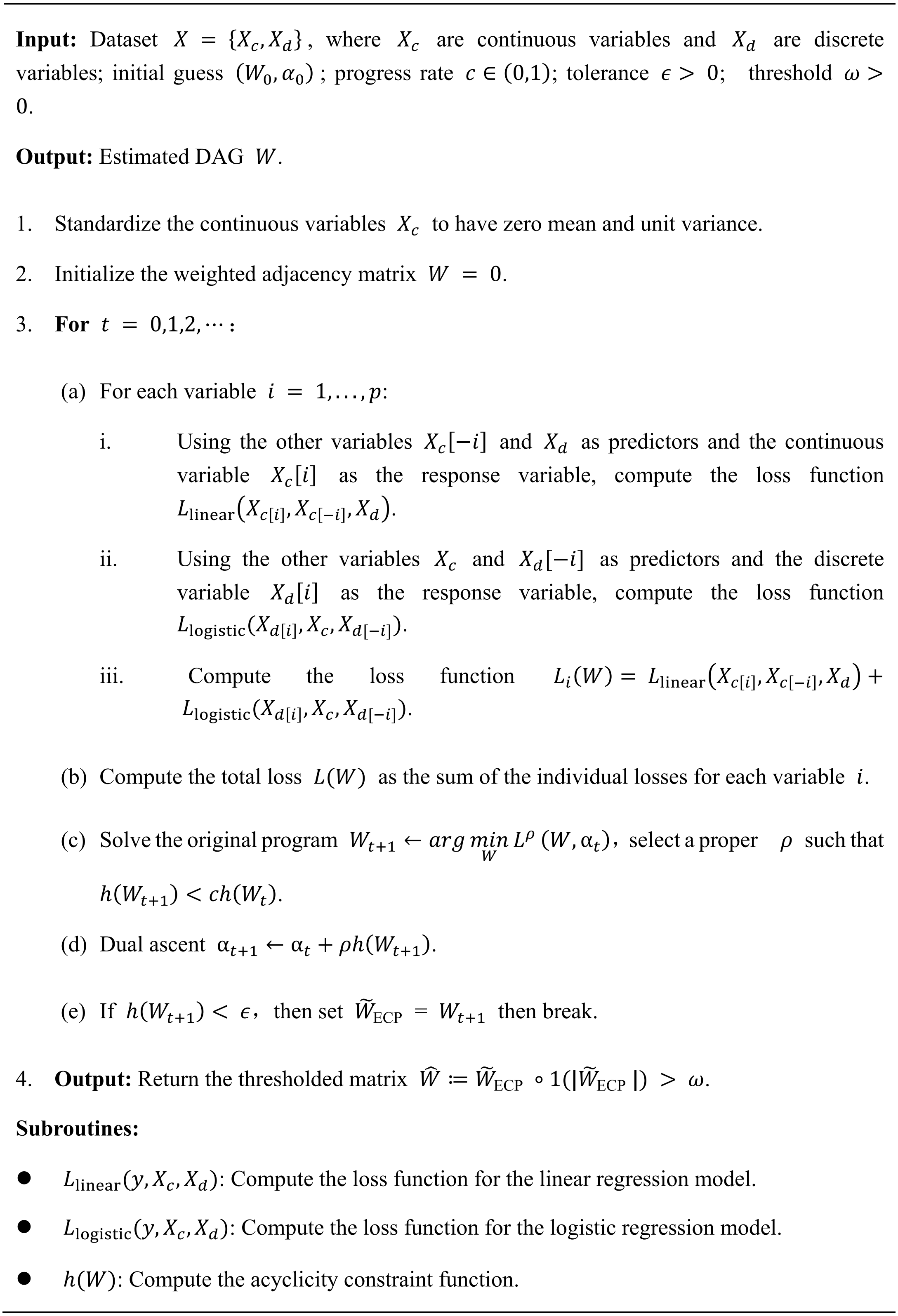
$$\min_{W \in \mathbb{R}^{\mathrm{d} \times \mathrm{d}}} L^{\rho,\lambda}(W, \alpha), \tag{17}$$

$$\text{where } L^{\rho,\lambda}(W, \alpha) = L(W) + \frac{\rho}{2}\left|h(W)\right|^2 + \alpha h(W) + \lambda\|W\|_1, \tag{18}$$

where $\|W\|_1 = \sum_{i \neq j}\left|W_{ij}\right|$.

In the context of regression problems, hard thresholding can effectively diminish the number of false positives. Therefore, we apply the following thresholding to the weights of the edges: after obtaining the optimal solution W^* for problem (17), we set the weights with absolute values less than ω to zero, given a fixed threshold $\omega > 0$. The following pseudocode outlines the DAG-SLAM algorithm, which can be implemented in Python 3 conveniently.

Algorithm 1. DAGSLAM: causal BN structure learning for mixed-type data [28]



```
Input: Dataset $X = \{X_c, X_d\}$, where $X_c$ are continuous variables and $X_d$ are discrete variables; initial guess $(W_0, \alpha_0)$; progress rate $c \in (0,1)$; tolerance $\epsilon > 0$; threshold $\omega > 0$.

Output: Estimated DAG $W$.

1. Standardize the continuous variables $X_c$ to have zero mean and unit variance.
2. Initialize the weighted adjacency matrix $W = 0$.
3. For $t = 0,1,2,\cdots$:
    (a) For each variable $i = 1,\ldots,p$:
        i.   Using the other variables $X_c[-i]$ and $X_d$ as predictors and the continuous variable $X_c[i]$ as the response variable, compute the loss function $L_{\text{linear}}(X_{c[i]}, X_{c[-i]}, X_d)$.
        ii.  Using the other variables $X_c$ and $X_d[-i]$ as predictors and the discrete variable $X_d[i]$ as the response variable, compute the loss function $L_{\text{logistic}}(X_{d[i]}, X_c, X_{d[-i]})$.
        iii. Compute the loss function $L_i(W) = L_{\text{linear}}(X_{c[i]}, X_{c[-i]}, X_d) + L_{\text{logistic}}(X_{d[i]}, X_c, X_{d[-i]})$.
    (b) Compute the total loss $L(W)$ as the sum of the individual losses for each variable $i$.
    (c) Solve the original program $W_{t+1} \leftarrow arg\,\underset{W}{min}\, L^{\rho}\,(W, \alpha_t)$, select a proper $\rho$ such that $h(W_{t+1}) < ch(W_t)$.
    (d) Dual ascent $\alpha_{t+1} \leftarrow \alpha_t + \rho h(W_{t+1})$.
    (e) If $h(W_{t+1}) < \epsilon$, then set $\widetilde{W}_{\text{ECP}} = W_{t+1}$ then break.
4. Output: Return the thresholded matrix $\widehat{W} \coloneqq \widetilde{W}_{\text{ECP}} \circ 1(|\widetilde{W}_{\text{ECP}}|) > \omega$.

Subroutines:
● $L_{\text{linear}}(y, X_c, X_d)$: Compute the loss function for the linear regression model.
● $L_{\text{logistic}}(y, X_c, X_d)$: Compute the loss function for the logistic regression model.
● $h(W)$: Compute the acyclicity constraint function.
```

Results

Simulation studies

We conduct several simulation experiments across different scenarios to evaluate the performance of our proposed DAGSLAM algorithm and compare it against six existing algorithms in the context of DAG structure learning. The algorithms selected for comparison include NOTEARS, which serves as the baseline algorithm for DAGSLAM, and DAGMA, both of which can be implemented in Python 3 (available at https://github.com/xunzheng/notears and https://github.com/kevinsbello/dagma, respectively).

We also include two established conventional score-based algorithms: 'HC' and 'TABU,' both available in the bnlearn package. Additionally, we consider the non-aggregated hill climbing (HC) algorithm from the 'Directed Acyclic Graph Bagging with Mixed Variables (DAGBagM)' package, as well as the 'Mixed Directed Acyclic Graph (mDAG)' algorithm, which is specifically designed for learning DAG structures with mixed node types.

The choice of these algorithms is based on their relevance to our study: NOTEARS and DAGMA are gradient-based algorithms, while HC and TABU represent mature traditional methods within the score-based framework. Furthermore, mDAG and DAGBagM were selected because they, like DAGSLAM, are tailored for learning DAG structures from mixed data types, particularly in biomedical applications.

All parameters for the comparison algorithms are set according to the default values recommended in their respective original publications. It is important to note that the HC, TABU, NOTEARS, and DAGMA methods are designed to handle datasets where all variables are of the same type—either all continuous or all categorical—and therefore cannot process mixed-type data. For simplicity, we treat all nodes as continuous in our simulations.

Simulation setup

To ensure the robustness and effectiveness of our proposed algorithm across various conditions, we conducted eight sets of simulation experiments in a total of 20 distinct scenarios, each featuring different combinations of the number of nodes, the proportion of categorical nodes, sample size, the levels of categorical nodes, edge sparsity, weight scale, graph type, and noise distribution. A summary of the simulation settings is presented in Table 1.

In each experimental set, a random graph G was produced on the basis of either the Erdős-Rényi (ER) model or the scale-free (SF) model, and the corresponding adjacency matrix $B \in \{0,1\}^{d \times d}$ was obtained. We configured various numbers of true edges $S_0 \in \{0.5d, d, 2d\}$. For the edge $i \rightarrow j$ that truly exists between nodes i and j, its weight w_{ij} was assigned independently from a uniform distribution over the interval $\alpha \cdot [0.5,2] \cup -\alpha \cdot [0.5,2]$, where α is the weight scale factor. This results in a weighted adjacency matrix $W = [w_{ij}] \in \mathbb{R}^{d \times d}$. Given W, we sample X through SEM according to each node's type. For continuous nodes X_j, data are generated via the linear model $X_j = w_j^T X + \epsilon_j$, where the random disturbance term ϵ_j follows one of three noise models: Gaussian distribution, exponential distribution or Gumbel distribution. For binary nodes X_k, data are produced via the logistic regression model $P(X_k = 1) = \frac{\exp(w_k^T X)}{1+\exp(w_k^T X)}$. For categorical nodes X_k with m levels ($m > 2$), data are generated via a multinomial logistic regression model represented as $P(X_k = l) = \frac{\exp\left(w_{k(l)}^T X\right)}{\sum_{m=1}^{M} \exp\left(w_{k(m)}^T X\right)}$, where $l = 1,2,\cdots,m$. On the basis of the aforementioned data generation mechanism, a random dataset $X \in \mathbb{R}^{n \times d}$ is produced, with each row being independent and identically distributed.

For each simulation scenario, we generated $n \in \{100,500,1000,5000,10000\}$ samples for graphs with $d \in \{10,20,40,100\}$ nodes. A specific number or proportion of nodes (10%, 20%, or 50%) was randomly selected to be categorical nodes, while the remaining nodes were designated continuous nodes.

On the basis of the recommended hyperparameter values provided in the results of Zheng et al., a weight threshold of $\omega = 0.3$ and a regularization coefficient of $\lambda = 0.1$ were established for all the aforementioned simulation experiments [37].

Evaluation metrics

For each method, the performance of the learned structures was assessed via five common metrics: the false discovery rate (FDR), true positive rate (TPR), false positive rate (FPR), structural Hamming distance (SHD), and F1 score. These metrics were evaluated for both the estimated directed structure (i.e., edges with direction) and the estimated skeleton structure (i.e., edges without direction). Given that we are learning causal BN structures, our primary focus is on the accuracy of the learned edge directions. Therefore, we emphasize the directed metrics in our analysis. The skeleton metrics are also reported but are considered supplementary references. In subsequent discussions, unless otherwise specified, all mentioned metrics will refer to the directed metrics.

We define positive (P) edges as those present in the estimated graph, true (T) edges as those found in the ground truth graph, and false (F) edges as the nonedges

Table 1 Simulation settings for each scenario

Scenario	d	k	m	n	s0/d	α	Graph type	Noise distribution
1	10	1	2	1000	1	1	ER	Gauss
2	20	1	2	1000	1	1	ER	Gauss
3	40	1	2	1000	1	1	ER	Gauss
4	100	1	2	1000	1	1	ER	Gauss
5	20	10%	2	1000	1	1	ER	Gauss
6	20	20%	2	1000	1	1	ER	Gauss
7	20	50%	2	1000	1	1	ER	Gauss
8	20	1	2	100	1	1	ER	Gauss
9	20	1	2	500	1	1	ER	Gauss
10	20	1	2	5000	1	1	ER	Gauss
11	20	1	2	10,000	1	1	ER	Gauss
12	20	1	3	1000	1	1	ER	Gauss
13	20	1	4	1000	1	1	ER	Gauss
14	20	1	2	1000	0.5	1	ER	Gauss
15	20	1	2	1000	2	1	ER	Gauss
16	20	1	2	1000	1	0.5	ER	Gauss
17	20	1	2	1000	1	2	ER	Gauss
18	20	1	2	1000	1	1	SF	Gauss
19	20	1	2	1000	1	1	ER	Exp
20	20	1	2	1000	1	1	ER	Gumbel

d Number of nodes, *k* Proportion or number of categorical nodes, *m* Levels of categorical nodes, *n* Sample size, *s0/d* Edge sparsity, *α* Weight scale

in the ground truth graph. True positive (TP) edges are defined as the estimated edges that have the correct direction in the ground truth graph, reversed (R) edges are the estimated edges with the opposite direction, and false positive (FP) edges are the estimated edges that do not exist in the ground truth skeleton. Additionally, let E be the extra edges in the estimated graph compared with the ground truth skeleton, and let M be the missing edges from the ground truth skeleton.

The five metrics of the estimated directed structure are given by:

$$FDR_{directed} = \frac{R + FP}{P}$$

$$TPR_{directed} = \frac{TP}{T}$$

$$FPR_{directed} = \frac{R + FP}{F}$$

$$SHD_{directed} = E + M + R$$

$$F1_{directed} = \frac{2 \times (1 - FDR_{directed}) \times TPR_{directed}}{1 - FDR_{directed} + TPR_{directed}}$$

The five metrics of the estimated skeleton structure are defined as follows:

$$FDR_{skeleton} = \frac{FP}{P}$$

$$TPR_{skeleton} = \frac{R + TP}{T}$$

$$FPR_{skeleton} = \frac{FP}{F}$$

$$SHD_{skeleton} = E + M$$

$$F1_{directed} = \frac{2 \times (1 - FDR_{skeleton}) \times TPR_{skeleton}}{1 - FDR_{skeleton} + TPR_{skeleton}}$$

In each simulation scenario, the performance metrics are calculated as the average of 10 independent replicates.

Simulation results

The details in terms of the FDR, TPR, FPR, SHD, and F1 score of each method are summarized in Table 2. To intuitively illustrate the ability of the DAGSLAM method to recover true DAG structures, we present visualizations of the true DAG structures, heatmaps of the adjacency matrix of the true graph, and the weighted adjacency

Table 2 Simulation results for each scenario (averaged over 10 replicates)

Scenario	Method	Directed					Skeleton				
		FDR	TPR	FPR	SHD	F1	FDR	TPR	FPR	SHD	F1
1	DAGSLAM	0.00	0.90	0.00	1.0	0.95	0.00	0.90	0.00	1.0	0.95
	NOTEARS	0.00	0.90	0.00	1.0	0.95	0.00	0.90	0.00	1.0	0.95
	DAGMA	0.10	0.90	0.03	1.0	0.90	0.00	1.00	0.00	0.0	1.00
	HC	0.26	0.85	0.09	3.1	0.79	0.14	0.99	0.05	1.7	0.92
	TABU	0.23	0.82	0.07	2.5	0.79	0.06	1.00	0.02	0.7	0.97
	mDAG	0.42	0.50	0.11	8.8	0.50	0.11	0.78	0.03	3.2	0.83
	DAGBagM	0.36	0.76	0.13	4.5	0.69	0.17	1.00	0.06	2.1	0.91
2	DAGSLAM	0.00	0.99	0.00	0.3	0.99	0.00	0.99	0.00	0.3	0.99
	NOTEARS	0.15	0.85	0.02	5.1	0.85	0.10	0.90	0.01	4.1	0.90
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.43	0.65	0.06	9.9	0.61	0.12	1.00	0.02	2.9	0.93
	TABU	0.42	0.66	0.06	9.5	0.62	0.12	1.00	0.02	2.7	0.94
	mDAG	0.28	0.73	0.03	20.1	0.72	0.01	0.99	0.00	0.5	0.99
	DAGBagM	0.35	0.75	0.05	8.7	0.69	0.14	1.00	0.02	3.7	0.92
3	DAGSLAM	0.00	0.97	0.00	1.0	0.99	0.00	0.97	0.00	1.0	0.99
	NOTEARS	0.00	0.98	0.00	0.7	0.99	0.00	0.98	0.00	0.7	0.99
	DAGMA	0.00	1.00	0.00	0.0	1.00	0.00	1.00	0.00	0.0	1.00
	HC	0.50	0.67	0.04	27.6	0.57	0.26	1.00	0.02	14.4	0.85
	TABU	0.42	0.77	0.03	22.4	0.66	0.25	1.00	0.02	13.3	0.86
	mDAG	0.30	0.66	0.02	37.4	0.68	0.02	0.92	0.00	4.0	0.95
	DAGBagM	0.46	0.71	0.03	25.2	0.60	0.25	1.00	0.02	13.7	0.85
4	DAGSLAM	0.00	0.98	0.00	1.9	0.99	0.00	0.98	0.00	1.7	0.99
	NOTEARS	0.02	0.96	0.00	4.6	0.97	0.01	0.98	0.00	3.0	0.98
	DAGMA	0.03	0.99	0.00	2.7	0.98	0.01	1.00	0.00	1.4	0.99
	HC	0.57	0.74	0.02	97.8	0.55	0.42	1.00	0.01	72.3	0.73
	TABU	0.55	0.77	0.02	95.4	0.57	0.42	1.00	0.01	72.6	0.73
	mDAG	0.16	0.79	0.00	94.9	0.81	0.01	0.94	0.00	6.7	0.97
	DAGBagM	0.58	0.74	0.02	104.9	0.54	0.44	0.99	0.02	80.7	0.71
5	DAGSLAM	0.00	0.95	0.00	1.0	0.97	0.00	0.95	0.00	1.0	0.97
	NOTEARS	0.18	0.79	0.02	6.2	0.81	0.11	0.87	0.01	4.7	0.88
	DAGMA	0.30	0.80	0.04	7.0	0.74	0.13	1.00	0.02	3.0	0.93
	HC	0.42	0.65	0.06	9.6	0.61	0.11	1.00	0.02	2.6	0.94
	TABU	0.39	0.68	0.05	8.9	0.64	0.11	1.00	0.01	2.5	0.94
	mDAG	0.28	0.70	0.03	6.5	0.71	0.02	0.95	0.00	1.4	0.96
	DAGBagM	0.38	0.74	0.06	10.0	0.66	0.17	1.00	0.03	4.7	0.90
6	DAGSLAM	0.16	0.80	0.02	5.0	0.82	0.05	0.90	0.01	3.0	0.92
	NOTEARS	0.27	0.70	0.03	8.3	0.71	0.12	0.85	0.01	5.3	0.86
	DAGMA	0.47	0.70	0.07	12.3	0.60	0.24	1.00	0.04	6.3	0.86
	HC	0.32	0.75	0.04	7.1	0.71	0.09	1.00	0.01	2.2	0.95
	TABU	0.31	0.78	0.04	7.2	0.73	0.11	1.00	0.02	2.8	0.94
	mDAG	0.29	0.70	0.03	6.6	0.71	0.03	0.95	0.00	1.6	0.96
	DAGBagM	0.31	0.81	0.04	7.4	0.75	0.15	1.00	0.02	3.7	0.92

Table 2 (continued)

Scenario	Method	Directed					Skeleton				
		FDR	**TPR**	**FPR**	**SHD**	**F1**	**FDR**	**TPR**	**FPR**	**SHD**	**F1**
7	DAGSLAM	0.09	0.74	0.01	6.1	0.81	0.05	0.77	0.00	5.4	0.85
	NOTEARS	0.25	0.50	0.02	11.0	0.60	0.07	0.62	0.01	8.6	0.74
	DAGMA	0.53	0.50	0.07	15.2	0.47	0.24	0.81	0.03	9.0	0.78
	HC	0.36	0.79	0.05	8.9	0.71	0.19	1.00	0.03	4.7	0.90
	TABU	0.37	0.74	0.05	8.9	0.68	0.15	1.00	0.02	3.8	0.91
	mDAG	0.31	0.70	0.04	7.3	0.69	0.06	0.96	0.01	2.1	0.95
	DAGBagM	0.36	0.76	0.05	8.7	0.69	0.16	1.00	0.02	4.0	0.91
8	DAGSLAM	0.03	0.91	0.00	2.3	0.94	0.02	0.92	0.00	2.1	0.95
	NOTEARS	0.16	0.82	0.02	5.4	0.83	0.09	0.89	0.01	4.1	0.90
	DAGMA	0.27	0.89	0.04	6.9	0.80	0.19	0.99	0.03	4.9	0.89
	HC	0.49	0.63	0.07	13.9	0.56	0.25	0.92	0.04	8.1	0.82
	TABU	0.49	0.64	0.08	14.4	0.56	0.27	0.92	0.04	8.8	0.81
	mDAG	0.32	0.56	0.03	9.2	0.61	0.02	0.80	0.00	4.3	0.88
	DAGBagM	0.43	0.73	0.07	13.0	0.63	0.27	0.94	0.04	8.7	0.82
9	DAGSLAM	0.00	0.98	0.00	0.5	0.99	0.00	0.98	0.00	0.5	0.99
	NOTEARS	0.14	0.84	0.02	5.2	0.85	0.10	0.88	0.01	4.5	0.89
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.44	0.65	0.06	10.5	0.60	0.15	0.99	0.02	3.7	0.92
	TABU	0.44	0.67	0.06	10.7	0.61	0.16	0.99	0.02	4.2	0.91
	mDAG	0.34	0.62	0.04	7.8	0.64	0.01	0.93	0.00	1.5	0.96
	DAGBagM	0.43	0.72	0.07	11.4	0.62	0.21	0.99	0.03	5.9	0.87
10	DAGSLAM	0.00	0.98	0.00	0.5	0.99	0.00	0.98	0.00	0.5	0.99
	NOTEARS	0.15	0.85	0.02	5.0	0.85	0.10	0.90	0.01	4.1	0.90
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.41	0.66	0.05	9.2	0.62	0.10	1.00	0.01	2.3	0.95
	TABU	0.41	0.66	0.05	9.1	0.62	0.10	1.00	0.01	2.3	0.95
	mDAG	0.31	0.70	0.04	6.4	0.69	0.01	1.00	0.00	20.3	0.99
	DAGBagM	0.26	0.82	0.03	5.9	0.78	0.10	1.00	0.01	2.3	0.95
11	DAGSLAM	0.00	0.98	0.00	0.4	0.99	0.00	0.98	0.00	0.4	0.99
	NOTEARS	0.15	0.85	0.02	4.9	0.85	0.10	0.91	0.01	3.9	0.90
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.40	0.65	0.05	8.5	0.62	0.06	1.00	0.01	1.4	0.97
	TABU	0.39	0.66	0.05	8.4	0.63	0.07	1.00	0.01	1.5	0.96
	mDAG	0.29	0.71	0.03	5.9	0.71	0.00	1.00	0.00	20.1	1.00
	DAGBagM	0.22	0.83	0.03	4.8	0.80	0.06	1.00	0.01	1.4	0.97
12	DAGSLAM	0.13	0.99	0.02	3.2	0.93	0.13	0.99	0.02	3.2	0.93
	NOTEARS	0.04	0.84	0.00	3.3	0.90	0.01	0.87	0.00	2.7	0.93
	DAGMA	0.12	0.89	0.01	3.8	0.88	0.07	0.93	0.01	2.9	0.93
	HC	0.58	0.51	0.08	15.8	0.42	0.24	0.92	0.04	7.7	0.83
	TABU	0.47	0.60	0.06	12.2	0.54	0.18	0.94	0.02	5.5	0.87
	mDAG	0.56	0.35	0.05	13.8	0.32	0.05	0.76	0.00	5.7	0.84
	DAGBagM	0.49	0.58	0.06	12.5	0.53	0.18	0.93	0.02	5.5	0.87

Table 2 (continued)

Scenario	Method	Directed					Skeleton				
		FDR	TPR	FPR	SHD	F1	FDR	TPR	FPR	SHD	F1
13	DAGSLAM	0.00	0.90	0.00	2.0	0.95	0.00	0.90	0.00	2.0	0.95
	NOTEARS	0.00	0.90	0.00	2.0	0.95	0.00	0.90	0.00	2.0	0.95
	DAGMA	0.00	0.90	0.00	2.0	0.95	0.00	0.90	0.00	2.0	0.95
	HC	0.43	0.65	0.06	10.1	0.60	0.13	0.99	0.02	3.2	0.93
	TABU	0.40	0.66	0.05	9.6	0.63	0.13	0.96	0.02	3.6	0.91
	mDAG	0.43	0.51	0.04	10.2	0.52	0.02	0.87	0.00	2.9	0.92
	DAGBagM	0.39	0.73	0.05	10.3	0.66	0.20	0.95	0.03	5.9	0.87
14	DAGSLAM	0.00	1.00	0.00	0.0	1.00	0.00	1.00	0.00	0.0	1.00
	NOTEARS	0.00	1.00	0.00	0.0	1.00	0.00	1.00	0.00	0.0	1.00
	DAGMA	0.00	1.00	0.00	0.0	1.00	0.00	1.00	0.00	0.0	1.00
	HC	0.57	0.59	0.04	7.8	0.49	0.26	1.00	0.02	3.7	0.85
	TABU	0.53	0.59	0.04	6.9	0.51	0.20	1.00	0.02	2.8	0.88
	mDAG	0.37	0.64	0.02	3.7	0.64	0.01	1.00	0.00	0.1	1.00
	DAGBagM	0.42	0.70	0.03	5.2	0.63	0.17	1.00	0.01	2.2	0.90
15	DAGSLAM	0.09	0.89	0.02	5.2	0.90	0.02	0.96	0.00	2.3	0.97
	NOTEARS	0.02	0.95	0.01	1.9	0.97	0.00	0.97	0.00	1.1	0.99
	DAGMA	0.00	1.00	0.00	0.0	1.00	0.00	1.00	0.00	0.0	1.00
	HC	0.56	0.71	0.25	39.1	0.54	0.41	0.96	0.18	29.0	0.73
	TABU	0.47	0.78	0.20	31.5	0.63	0.35	0.96	0.15	24.2	0.77
	mDAG	0.30	0.55	0.06	19.8	0.62	0.06	0.74	0.01	12.3	0.83
	DAGBagM	0.66	0.59	0.36	58.6	0.37	0.51	0.90	0.28	46.2	0.63
16	DAGSLAM	0.00	0.86	0.00	2.8	0.92	0.00	0.86	0.00	2.8	0.92
	NOTEARS	0.06	0.76	0.01	4.9	0.84	0.00	0.80	0.00	3.9	0.89
	DAGMA	0.08	0.79	0.01	4.1	0.85	0.00	0.86	0.00	2.8	0.92
	HC	0.56	0.49	0.07	13.8	0.42	0.16	0.95	0.02	4.7	0.89
	TABU	0.55	0.51	0.07	13.8	0.43	0.17	0.95	0.02	5.0	0.88
	mDAG	0.35	0.62	0.04	7.8	0.63	0.01	0.95	0.00	1.3	0.97
	DAGBagM	0.44	0.59	0.05	10.4	0.57	0.10	0.95	0.01	3.3	0.92
17	DAGSLAM	0.00	0.90	0.00	2.1	0.94	0.00	0.90	0.00	2.1	0.94
	NOTEARS	0.11	0.88	0.01	4.6	0.88	0.11	0.88	0.01	4.6	0.88
	DAGMA	0.13	0.90	0.02	4.0	0.89	0.10	0.93	0.01	3.4	0.92
	HC	0.53	0.56	0.07	13.5	0.50	0.19	0.96	0.03	5.5	0.88
	TABU	0.43	0.62	0.06	10.5	0.59	0.12	0.96	0.02	3.7	0.91
	mDAG	0.48	0.40	0.04	12.8	0.40	0.05	0.73	0.00	6.1	0.83
	DAGBagM	0.45	0.60	0.06	11.0	0.57	0.13	0.95	0.02	4.0	0.91
18	DAGSLAM	0.00	0.95	0.00	1.0	0.97	0.00	0.95	0.00	1.0	0.97
	NOTEARS	0.11	0.89	0.01	3.0	0.89	0.05	0.95	0.01	2.0	0.95
	DAGMA	0.10	0.94	0.01	2.1	0.92	0.05	0.99	0.01	1.1	0.97
	HC	0.20	0.89	0.02	4.2	0.85	0.10	1.00	0.01	2.2	0.95
	TABU	0.19	0.89	0.02	4.0	0.85	0.09	1.00	0.01	1.9	0.95
	mDAG	0.25	0.77	0.03	5.6	0.76	0.06	0.97	0.01	1.8	0.95
	DAGBagM	0.33	0.78	0.05	8.1	0.71	0.16	0.99	0.02	4.0	0.91

Table 2 (continued)

Scenario	Method	Directed					Skeleton				
		FDR	TPR	FPR	SHD	F1	FDR	TPR	FPR	SHD	F1
19	DAGSLAM	0.24	0.80	0.03	7.3	0.78	0.16	0.89	0.02	5.6	0.86
	NOTEARS	0.11	0.83	0.01	5.0	0.86	0.09	0.85	0.01	4.6	0.88
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.45	0.65	0.06	10.6	0.60	0.15	1.00	0.02	3.6	0.92
	TABU	0.42	0.67	0.06	10.1	0.62	0.15	1.00	0.02	3.6	0.92
	mDAG	0.30	0.62	0.03	7.9	0.66	0.02	0.88	0.00	2.8	0.93
	DAGBagM	0.39	0.75	0.06	10.5	0.67	0.20	0.99	0.03	5.6	0.88
20	DAGSLAM	0.22	0.94	0.03	5.7	0.85	0.18	0.99	0.03	4.7	0.89
	NOTEARS	0.19	0.88	0.02	4.4	0.84	0.09	0.99	0.01	2.3	0.94
	DAGMA	0.18	0.90	0.02	4.0	0.86	0.09	1.00	0.01	2.0	0.95
	HC	0.40	0.66	0.05	8.9	0.63	0.09	1.00	0.01	2.1	0.95
	TABU	0.38	0.68	0.05	8.4	0.64	0.09	1.00	0.01	1.9	0.96
	mDAG	0.26	0.73	0.03	5.8	0.73	0.02	0.97	0.00	1.0	0.97
	DAGBagM	0.38	0.73	0.06	9.6	0.66	0.16	1.00	0.02	4.2	0.91

matrices of both the true graph and those estimated by the DAGSLAM algorithm for each scenario in Figs. S1—S20 (Additional file 1).

The number of nodes

Simulation (i) comprises scenarios 1, 2, 3 and 4, with a focus on the impact of varying the number of nodes d, with values set at $\{10, 20, 40, 100\}$. Overall, DAGSLAM consistently outperforms other methods across different node counts. At $d = 10$, DAGSLAM achieves an impressive directed F1 score of 0.95 and a SHD of 1.0, demonstrating its ability to accurately capture the true structure in smaller networks (Table 2). As the number of nodes increases to $d = 20$ and $d = 40$, DAGSLAM's performance continues to improve, with F1 scores remaining close to 1 and SHD values near 0, indicating its stability and effectiveness in handling more complex networks. Notably, DAGSLAM consistently appears in the upper-right corner of the precision-recall (PR) plots (Fig. 1A and B), achieving both high precision and high recall across all node counts, with values approaching 1.00.

In contrast, other algorithms exhibit varying degrees of performance degradation as d increases. For instance, HC, mDAG and DAGBagM show a noticeable decline in F1 scores at $d = 40$ (Fig. 1C and D, Table 2), with HC dropping to 0.57, mDAG to 0.68 and TABU to 0.66. At $d = 100$, performance of HC, TABU, and DAGBagM further deteriorates, with HC and TABU achieving F1 scores of 0.55 and 0.57, respectively, and DAGBagM dropping to 0.54. This decline is accompanied by a sharp increase in SHD values (Fig. 1E and F), reflecting their challenges in accurately modeling high-dimensional data.

From the PR plots (Fig. 1A and B), DAGSLAM consistently outperforms other methods across all node counts, maintaining high precision and recall. While NOTEARS and DAGMA also perform well in high-dimensional settings ($d = 40$ and $d = 100$), their precision and recall values at $d = 20$ are slightly lower than those of DAGSLAM, indicating a relative weakness in mid-range scenarios.

F1 scores and SHD values also reflect similar facts. NOTEARS and DAGMA perform comparably to DAGSLAM in high-dimensional settings ($d = 40$ and $d = 100$), achieving high F1 scores and low SHD values. However, at $d = 20$, both algorithms exhibit a slight drop in performance compared to DAGSLAM, with DAGMA and NOTEARS achieving F1 scores of 0.86 and 0.85, respectively, and slightly higher SHD values (4.0 and 5.1, respectively) (Fig. 1C, D, E and F, Table 2). This suggests

(See figure on next page.)

Fig. 1 Results for Simulation (i): Comparison of the performance of DAGSLAM, NOTEARS, DAGMA, HC, TABU, mDAG, and DAGBagM with varying numbers of nodes d= 10, 20, 40, 100. **A** Precision-Recall plot for detecting directed structures. **B** Precision-Recall plot for detecting skeleton structures. **C** F1 score for detecting directed structures. **D** F1 score for detecting skeleton structures. **E** Structural Hamming Distance (SHD) for detecting directed structures. **F** SHD for detecting skeleton structures. The results are averaged over 10 replicates for each scenario. The error bars in C), D), E) and F) indicate the standard deviation of each bar

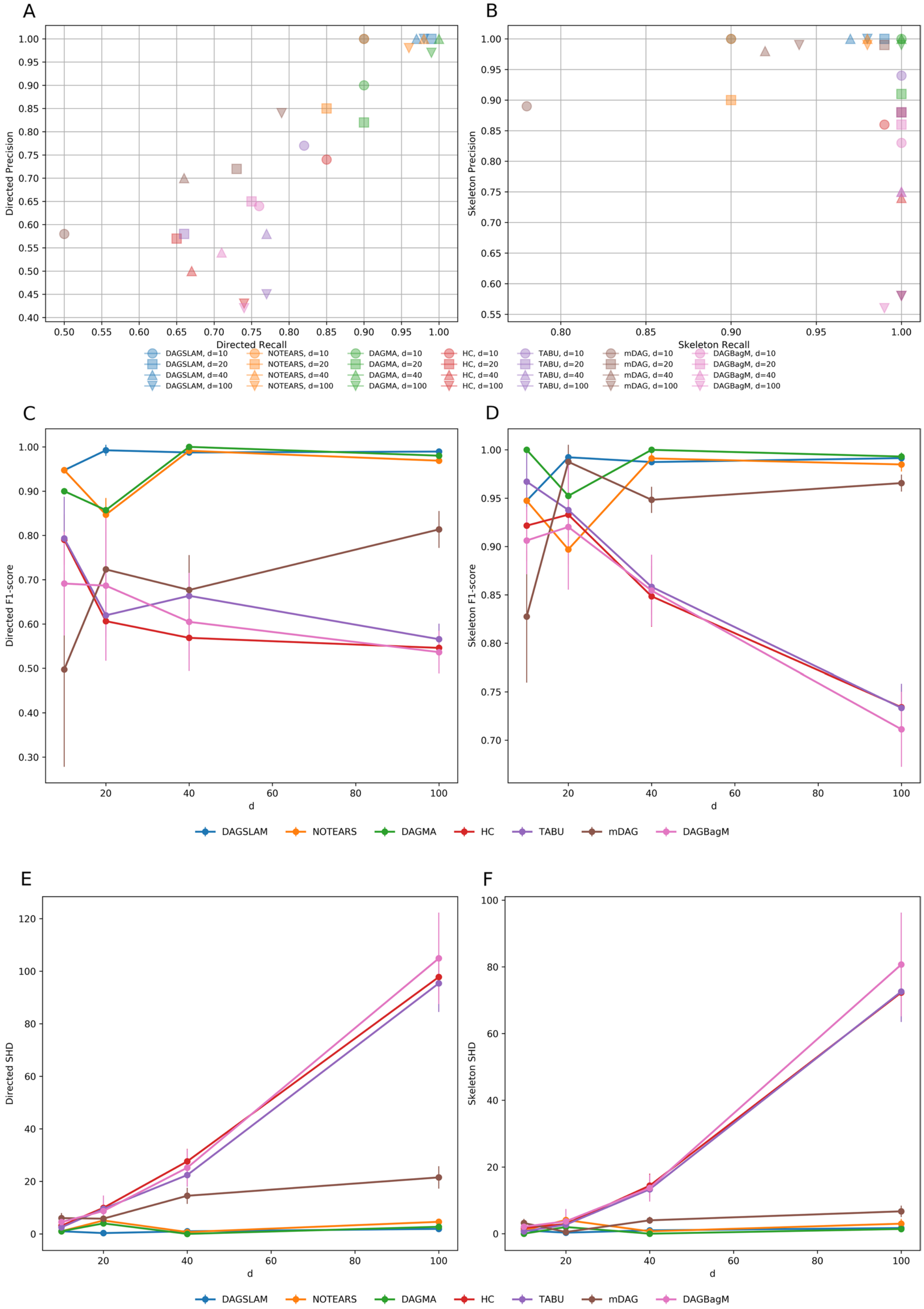


Fig. 1 (See legend on previous page.)

that DAGSLAM is more robust in mid-range node counts.

Proportion of categorical nodes

Simulation (ii) includes scenarios 5, 6, and 7, with a focus on the impact of varying the proportion of categorical nodes, with values set at $\{10\%, 20\%, 50\%\}$. The true DAG graph and adjacency matrix remain the same as those in scenario 3. We fixed the number of nodes at $d = 20$, resulting in the corresponding number of categorical nodes $k = 2,4, 10$.

The performance of the algorithms is significantly affected by the proportion of categorical nodes. As the proportion k increases, the performance of the gradient-based algorithms—DAGSLAM, NOTEARS, and DAGMA—declines to varying degrees. For instance, at $k = 10$ (50% categorical nodes), DAGSLAM's F1 score drops from 0.97 to 0.82, while NOTEARS and DAGMA show F1 scores around 0.71 and 0.60, respectively (Fig. 2C, Table 2). This decline may be attributed to the inherent challenges that arise when integrating categorical data into the structure learning process, complicating the optimization landscape.

Despite this, DAGSLAM maintains the best performance across all scenarios ($k = 2,4,10$). In Fig. 2A, DAGSLAM is consistently closer to the upper-right corner compared to other algorithms, indicating its advantages in both directed precision and directed recall. In terms of F1 scores (Fig. 2C), DAGSLAM outperforms other algorithms, while its SHD values **(**Fig. 2E) remain lower than those of the other methods, demonstrating its effectiveness in capturing the true structure even with relatively large proportion of categorical nodes.

In contrast, the performance of HC, TABU, mDAG, and DAGBagM shows little change with increasing proportions of categorical nodes, and in some cases, their performance even improves slightly. However, despite being more robust in handling categorical data, these algorithms consistently underperform compared to DAGSLAM, with F1 scores only around 0.70 (Fig. 2C, Table 2).

Sample size

Simulation (iii) comprises scenarios 8, 9, 2, 10, and 11, focusing on the impact of varying the number of sample sizes n, with values set at $\{100, 500, 1000, 5000, 10000\}$, under a fixed DAG topology structure. In general, DAGSLAM consistently outperforms other methods across all sample sizes, followed by NOTEARS and DAGMA, while DAGBagM and mDAG rank lower, with HC and TABU performing the worst.

From the PR plots (Fig. 3A), DAGSLAM consistently appears in the upper-right corner across all sample sizes, achieving both high precision and high recall. At $n = 100$, DAGSLAM achieves a directed F1 score of 0.94 and an SHD value of 2.3 (Fig. 3C and E, Table 2). As n increases, its performance improves further, reaching an F1 score close to 1 and an SHD value near 0 at $n = 5000$ and $n = 10000$. Notably, DAGSLAM's performance stabilizes when $n \geq 1000$, showing no significant improvement with further increases in sample size.

NOTEARS and DAGMA also perform well, particularly in large-sample scenarios ($n = 5000$ and $n = 10000$), where their F1 scores approach 0.85 and SHD values remain below 5.0 (Fig. 3C and E, Table 2). However, at smaller sample sizes ($n = 100$ and $n = 500$), However, at smaller sample sizes ($n = 100$ and $n = 500$), their performance shows a slight decline, and across all sample sizes, they do not perform as well as DAGSLAM.

In contrast, DAGBagM and mDAG exhibit poor performance at smaller sample sizes, with F1 scores below 0.65 and high SHD values (DAGBagM has an F1 score of 0.63 and SHD of 13.0 at $n = 100$, while mDAG has an F1 score of 0.61 and SHD of 9.2) (Fig. 3C and E, Table 2). Although their performance improves as n increases, even at $n = 10000$, their F1 scores and SHD values remain significantly worse than those of DAGSLAM, NOTEARS, and DAGMA. Notably, DAGBagM shows a unique trend where its performance continues to improve slightly even when $n \geq 1000$.

HC and TABU struggle the most in low-sample scenarios, with F1 scores below 0.6 and high SHD values at $n = 100$ (HC has an F1 score of 0.56 and SHD of 13.9, while TABU has an F1 score of 0.56 and SHD of 14.4) (Fig. 3C and E, Table 2). While their performance improves with larger sample sizes, they still lag behind other algorithms at $n = 10000$.

Levels of categorical nodes

Simulation (iv) includes scenarios 2, 12, and 13, which focus on the impact of varying the levels of categorical nodes m, with values set at $\{2,3,4\}$, while maintaining a

(See figure on next page.)

Fig. 2 Results for Simulation (ii): Comparison of the performance of DAGSLAM, NOTEARS, DAGMA, HC, TABU, mDAG, and DAGBagM with varying number (proportion) of categorical nodes k= 2 (10%), 4 (20%), 10 (50%). **A** Precision-Recall plot for detecting directed structures. **B** Precision-Recall plot for detecting skeleton structures. **C** F1 score for detecting directed structures. **D** F1 score for detecting skeleton structures. **E** Structural Hamming Distance (SHD) for detecting directed structures. **F** SHD for detecting skeleton structures. The results are averaged over 10 replicates for each scenario. The error bars in C), D), E) and F) indicate the standard deviation of each bar

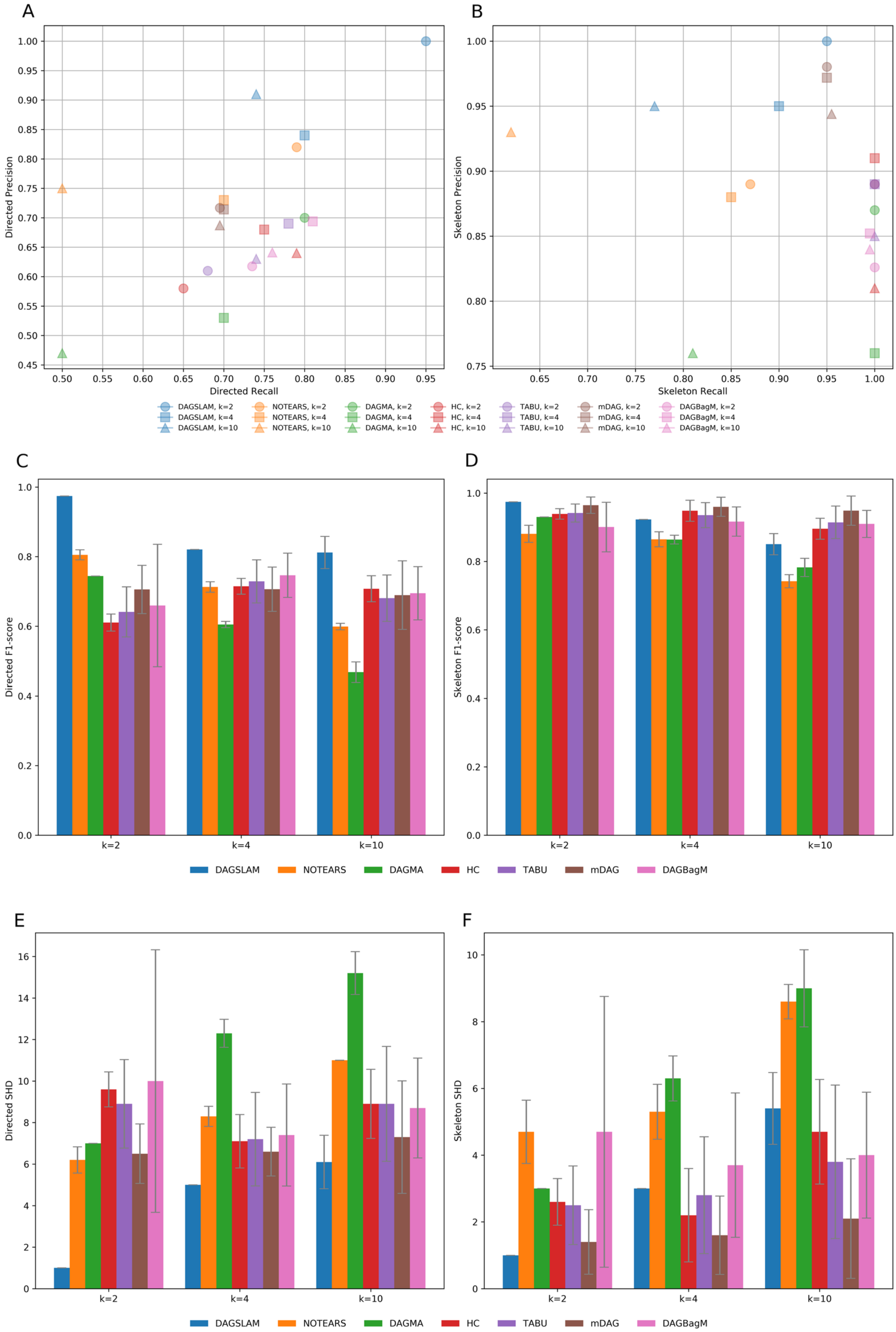


Fig. 2 (See legend on previous page.)

fixed DAG topology structure. It is important to note that mDAG and DAGBagM cannot handle categorical nodes with more than 2 levels ($m > 2$), and therefore, these algorithms treat multi-category variables as continuous variables.

The performance of the algorithms varies with the levels of categorical nodes. At $m = 2$ and $m = 3$, DAGSLAM leads the other algorithms, achieving higher F1 scores, TPR, and lower SHD. For instance, at $m = 3$, DAGSLAM achieves an F1 score of 0.93 and a TPR of 0.99, which are higher than all other algorithms (Fig. 4C, Table 2).

Interestingly, as the number of levels m increases to 4, the performance of NOTEARS and DAGMA slightly improves, and at this level, NOTEARS and DAGMA's performance matches that of DAGSLAM exactly, both achieving an F1 score of 0.95 and a SHD of 2.0 (Fig. 4C and E, Table 2). However, DAGSLAM still significantly outperforms the other four algorithms, which continue to show lower F1 scores below 0.70 and higher SHD values around 10.0 (Fig. 4C and E,Table 2).

This observation suggests that when $m \geq 4$, multi-category variables can be treated as continuous variables without a significant loss in performance for the gradient-based algorithms. However, other algorithm's inability to handle categorical variables more than 2 levels limits their applicability in scenarios with higher categorical levels.

Owing to space limitations in the main text, the detailed results of the simulation experiments examining the effects of edge sparsity, weight scale, graph type, and noise distribution are provided in Additional file 2.

Computational time analysis

To better understand the computational limits of our proposed algorithm, we investigated the impact of the number of nodes d and sample size n on computational time. In this analysis, we compare DAGSLAM with two other gradient-based algorithms, NOTEARS and DAGMA. Note that we do not compare these with the other four algorithms implemented in R, as our algorithm is implemented in Python. The simulations were conducted on a machine with the following specifications: AMD Ryzen 5 5500U @ 2.10GHz with 6 cores and 16.0GB of RAM.

The computational time of the algorithms was analysed as a function of the number of nodes d and sample size n, as shown in Figs. 5and 6. The specific computational time results are detailed in Table S1 and Table S2 (Additional file 1).

Figure 5 illustrates the computational time for each algorithm with a fixed sample size of $n = 1000$ while varying the number of nodes d from 20 to 100. As d increases, the computational time for all three algorithms shows a significant upward trend. Specifically, the computational time for both DAGSLAM and NOTEARS increases by approximately two orders of magnitude, reaching around 10^3 s when $d = 100$. In contrast, DAGMA's computational time increases by only one order of magnitude, indicating a more moderate growth compared to the other two algorithms.

Figure 6 presents the computational time with a fixed number of nodes $d = 20$ while varying the sample size n from 1000 to 10,000. In this scenario, both DAGSLAM and NOTEARS exhibit an increase in computational time by about one order of magnitude. At $n = 10000$, the computational time for DAGSLAM is approximately 10^2 s, while DAGMA's computational time remains relatively stable, showing minimal change despite the increase in sample size.

These results highlight that while DAGSLAM's computational time is higher than that of NOTEARS and DAGMA, this is largely due to its design for handling mixed-type data, which requires more complex computations. In contrast, NOTEARS and DAGMA benefit from treating all variables as continuous, allowing for more efficient matrix operations. This analysis underscores the importance of considering both computational efficiency and the specific data characteristics when selecting an appropriate algorithm for causal inference tasks.

Real-world dataset application

We apply our proposed method to the National Health and Nutrition Examination Survey (NHANES) dataset, which is a continuous program that employs sophisticated multistage probability sampling to gather a representative sample of the American population and evaluate the health and nutritional status of people across the United States. The NHANES survey includes demographic, dietary, examination, laboratory and health-related survey data. The NHANES study protocol received approval from the Ethics Review Committee of the National Center for Health Statistics, and all

(See figure on next page.)

Fig. 3 Results for Simulation (iii): Comparison of the performance of DAGSLAM, NOTEARS, DAGMA, HC, TABU, mDAG, and DAGBagM with varying sample size n= 100, 500, 1000, 5000, 10,000. **A** Precision-Recall plot for detecting directed structures. **B** Precision-Recall plot for detecting skeleton structures. **C** F1 score for detecting directed structures. **D** F1 score for detecting skeleton structures. **E** Structural Hamming Distance (SHD) for detecting directed structures. **F** SHD for detecting skeleton structures. The results are averaged over 10 replicates for each scenario. The error bars in C), D), E) and F) indicate the standard deviation of each bar

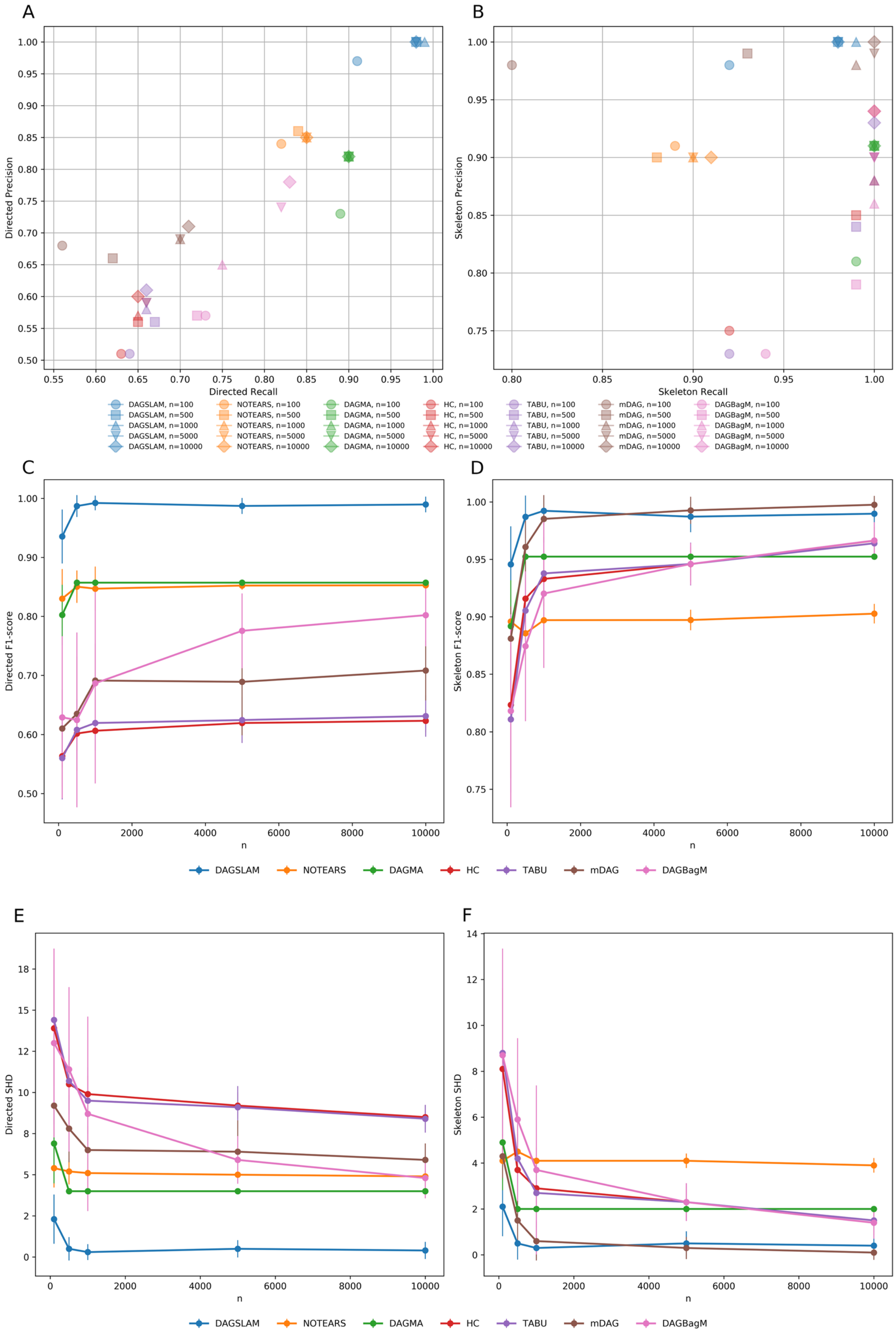


Fig. 3 (See legend on previous page.)

participants provided written informed consent. For further information, please refer to the NHANES—NCHS Research Ethics Review Board Approval page on the website.

We integrated data from 2005–2020, focusing on elderly women aged 65–79 years. The variables of interest and their types are listed in Table 3, comprising a total of 13 variables: 8 continuous variables and 5 categorical variables. After excluding cases with missing data, we obtained a complete dataset of 813 samples. Our objective was to investigate the risk factors for coronary heart disease (CHD) and diabetes, aiming to establish a causal network of the interactions among these risk factors.

The learned DAG structure is illustrated in Fig. 7, with the diseases of interest highlighted in orange. The arrows indicate positive relationships, whereas the diamonds represent negative relationships. The estimated weighted adjacency matrix W is detailed in Table S3 (Additional file 1). The results show that fasting glucose (GLU) serves as a critical starting node, impacting both waist circumference (WAIST) and glycohemoglobin (GHB). Elevated glucose levels are associated with increased waist circumference, reflecting the relationship between insulin resistance and central obesity, which is a known risk factor for metabolic syndrome [38]. Higher GHB levels indicate poorer long-term glucose control and a greater risk of diabetes [39].

Elevated levels of GHB promote the development of diabetes, further complicating the metabolic landscape. This relationship highlights the cascading effects of glucose metabolism on diabetes risk. Additionally, alcohol consumption (AL) has been shown to inhibit diabetes, indicating that higher alcohol intake may be associated with a lower risk of developing diabetes in this population. This relationship is complex; moderate alcohol consumption, particularly red wine, has been linked to improved insulin sensitivity and better glucose metabolism, potentially lowering the risk of type 2 diabetes, especially in females [40–42]. However, these benefits may not apply to everyone, especially those with a family history of diabetes or other health issues [43].

Waist circumference acts as a central node in the DAG, influencing several other factors. It promotes body mass index (BMI), reinforcing the connection between waist size and overall body fat. However, WAIST also inhibits high-density lipoprotein cholesterol (HDL), which is associated with low HDL levels and is a significant risk factor for cardiovascular disease [44]. This interplay suggests that as waist circumference increases, the risk of cardiovascular issues may rise due to lower HDL levels.

HDL not only promotes total cholesterol (TCHOL) but also inhibits triglycerides (TGs). This dual role highlights the complexity of lipid metabolism, where HDL is beneficial in managing triglyceride levels while also being part of the broader lipid profile.

TCHOL is positively associated with both low-density lipoprotein (LDL) and TG. Elevated total cholesterol levels typically correlate with increased LDL levels, which is a well-established risk factor for cardiovascular disease [44]. The promotion of triglycerides by total cholesterol further underscores the interconnected nature of lipid profiles.

Both alcohol consumption and hypertension (HTN) negatively influence CHD incidence. This finding is intriguing, as it suggests that in this population, higher alcohol intake and hypertension may be associated with a reduced risk of CHD. Moderate intake of alcoholic beverages has been associated with lower coronary heart disease risk due to beneficial components such as polyphenols and resveratrol, which may improve cardiovascular health [45–47]. However, the definition of moderate drinking varies, typically considered to be up to two drinks per day for men and one drink per day for women. Excessive drinking can lead to numerous health risks, including hypertension and heart disease [48].

Hypertension also inhibits CHD, which may seem counterintuitive given the established link between hypertension and increased cardiovascular risk. This relationship may be explained by the possibility that individuals diagnosed with hypertension become more vigilant in terms of their cardiovascular health. Upon learning of their condition, they may adopt healthier lifestyle choices, such as improved diet, increased physical activity, and adherence to medication regimens, which collectively contribute to a reduced incidence of coronary heart disease. This proactive approach to managing hypertension could mitigate some of the risks typically associated with high blood pressure.

(See figure on next page.)

Fig. 4 Results of Simulation (iv)**:** Comparison of the performance of DAGSLAM, NOTEARS, DAGMA, HC, TABU, mDAG, and DAGBagM with varying levels of categorical nodes $m = 2, 3, 4$. **A** Precision-Recall plot for detecting directed structures. **B** Precision-Recall plot for detecting skeleton structures. **C** F1 score for detecting directed structures. **D** F1 score for detecting skeleton structures. **E** Structural Hamming Distance (SHD) for detecting directed structures. **F** SHD for detecting skeleton structures. The results are averaged over 10 replicates for each scenario. The error bars in C), D), E) and F) indicate the standard deviation of each bar

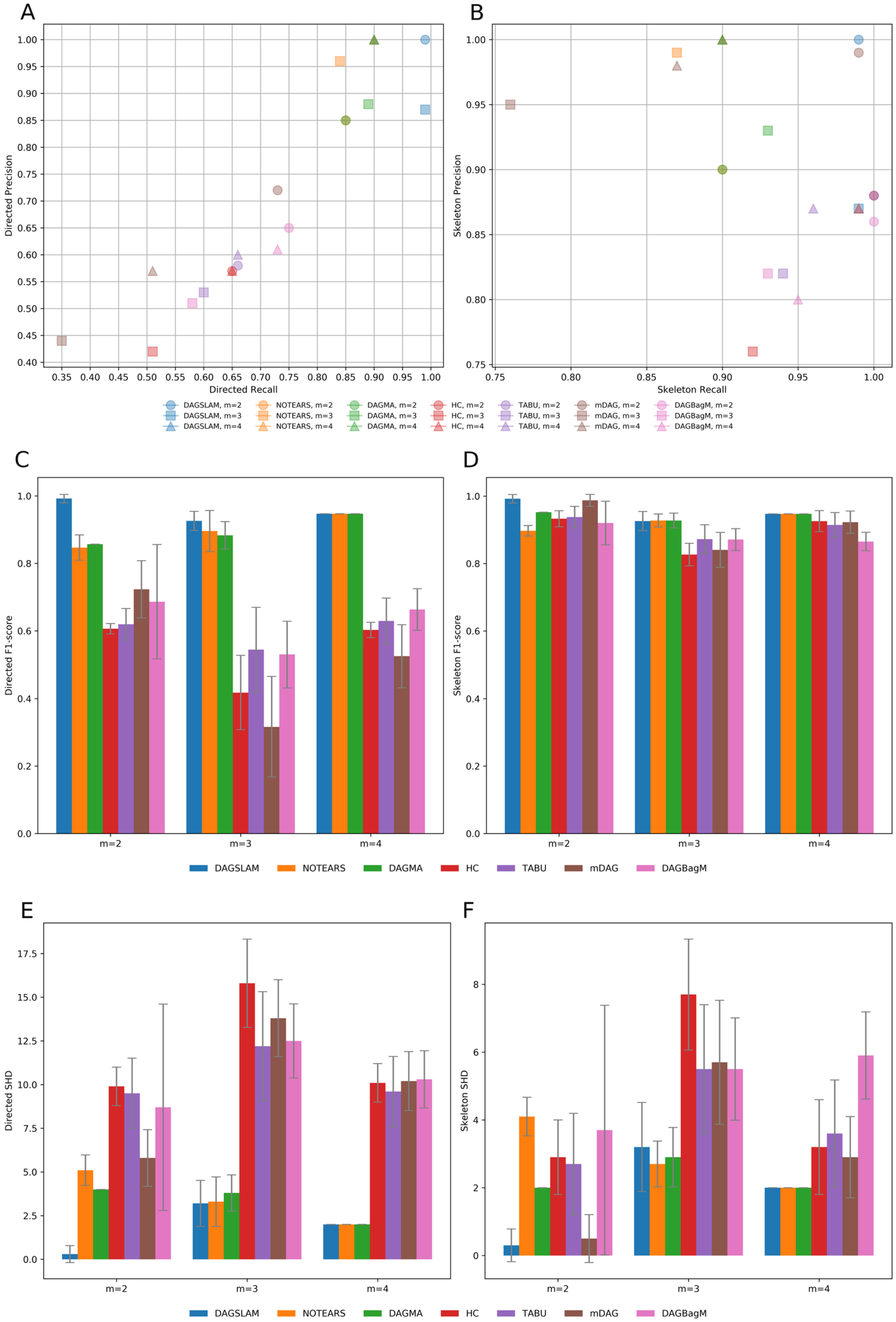


Fig. 4 (See legend on previous page.)

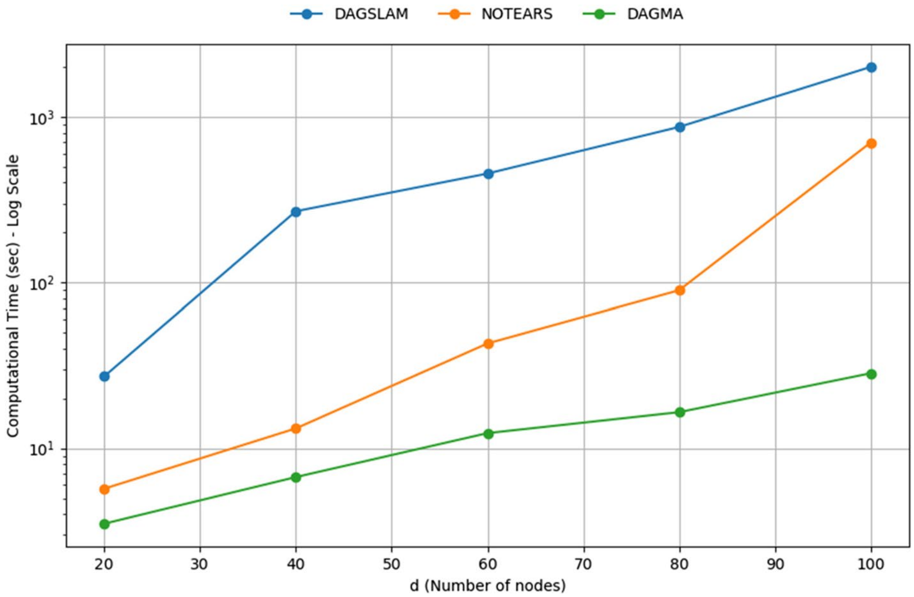


Fig. 5 Computational time (in seconds) as a function of the number of nodes (*d*) with the sample size (*n*) fixed at *n* = 1000. The graph displays the performance of three algorithms: DAGSLAM (blue), NOTEARS (orange), and DAGMA (green) on a logarithmic scale. The computational results are averaged over 10 replicates

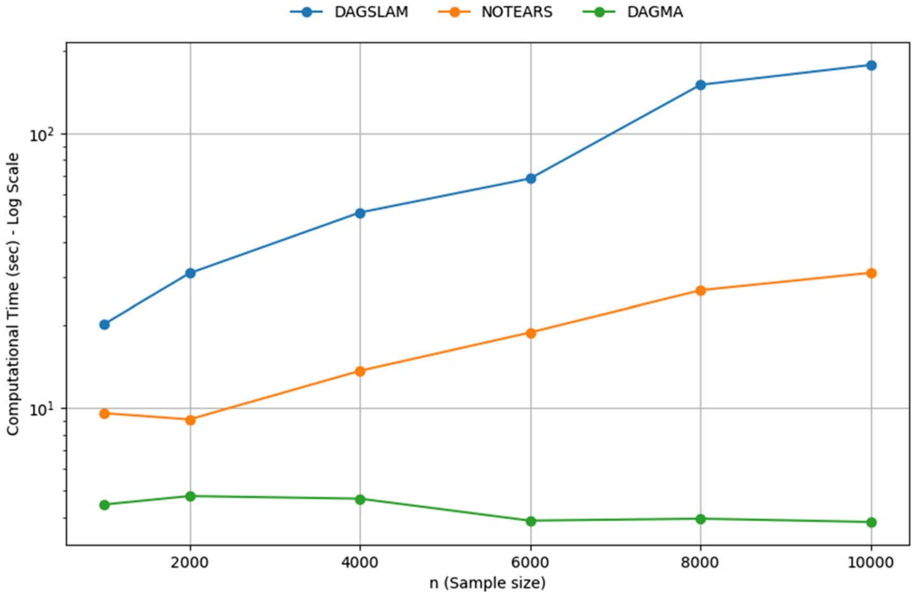


Fig. 6 Computational time (in seconds) as a function of sample size (*n*) with the number of nodes (*d*) fixed at d = 20. This graph depicts the performance of DAGSLAM (blue), NOTEARS (orange), and DAGMA (green) with the time displayed on a logarithmic scale. The computational results are averaged over 10 replicates

Discussion

DAGs have become an essential tool in epidemiological research for modelling causal relationships among disease risk factors. They allow researchers to disentangle direct and indirect effects, providing a clearer understanding of the underlying causal mechanisms [5]. However, existing DAG learning methods often struggle with real-world biomedical datasets, which frequently involve mixed data types (e.g., continuous biomarkers

Table 3 Descriptions of the variables in the real-world dataset analysis

Abbreviation	Type	Description
AL	Categorical	Had at least 12 alcohol drinks in lifetime 1: YES, 0: NO
BMI	Continuous	Body Mass Index (kg/m^2)
CHD	Categorical	Doctor ever told you had coronary heart disease 1: YES, 0: NO
DI	Categorical	Doctor told you have diabetes 1: YES, 0: NO
GLU	Continuous	Fasting Glucose (mg/dL)
GHB	Continuous	Glycohemoglobin (%)
HDL	Continuous	Direct High-Density Lipoprotein Cholesterol (mg/dL)
HTN	Categorical	Doctor ever told you had high blood pressure 1: YES, 0: NO
LDL	Continuous	Low-Density Lipoproteins Cholesterol, Friedewald equation (mg/dL)
SM	Categorical	Smoked at least 100 cigarettes in life 1: YES, 0: NO
TCHOL	Continuous	Total Cholesterol (mg/dL)
TG	Continuous	Triglycerides, refrigerated serum (mg/dL)
WAIST	Continuous	Waist Circumference (cm)

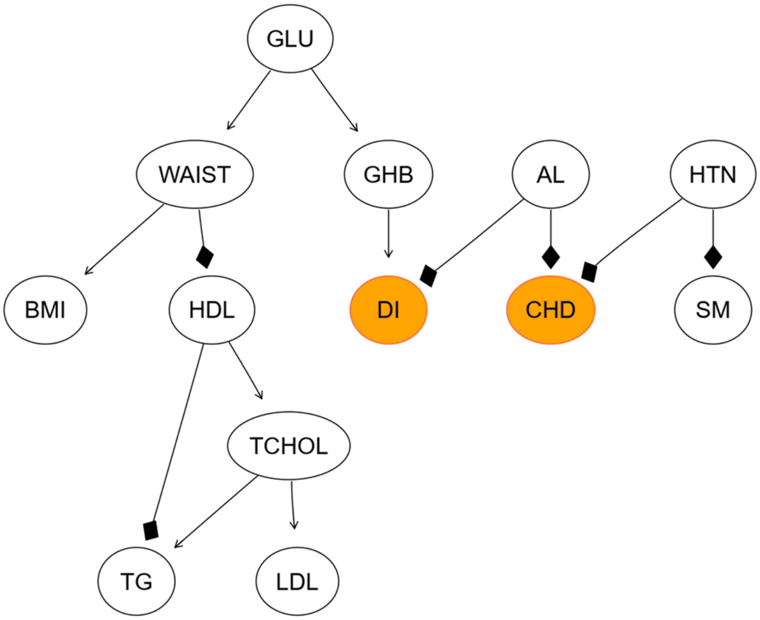


Fig. 7 Learned DAG via the DAGSLAM algorithm for the NHANES dataset. The diseases of interest are highlighted in orange. The arrows indicate positive relationships, whereas the diamonds represent negative relationships. AL: alcohol consumption; BMI: body mass index; CHD: coronary heart disease; DI: diabetes; GLU: fasting glucose; GHB: glycohemoglobin; HDL: high-density lipoprotein cholesterol; HTN: hypertension; LDL: low-density lipoprotein cholesterol; SM: smoking; TCHOL: total cholesterol; TG: triglycerides; WAIST: waist circumference

and categorical clinical outcomes) and complex, high-dimensional structures [16, 28]. These limitations have hindered their broader application in disease risk factor modelling.

To address these challenges, we propose an extension of the NOTEARS algorithm, referred to as DAGSLAM, which is specifically designed to handle mixed-type data. This novel algorithm integrates continuous and categorical variables into a unified framework by introducing a tailored loss function that accommodates both data types. By building on the conceptual framework of the original NOTEARS, DAGSLAM achieves robust performance across a variety of scenarios, including networks with varying node counts, sample sizes, proportions and levels of categorical variables, variations in edge density, adjustments to the weight scale, different graph types, and diverse noise distributions. Furthermore, the algorithm demonstrates scalability to larger datasets, making it suitable for real-world applications.

In addition to its methodological advancements, DAGSLAM is the first adaptation of the gradient-based algorithm to mixed-type data and its first application in the medical domain for identifying disease risk factors. Through extensive simulations, the algorithm consistently outperforms existing methods, such as HC, TABU, mDAG, and DAGBagM, across key metrics, including precision, recall, F1 score, and SHD. These results highlight its ability to accurately infer both directed and undirected structures, even in challenging scenarios with limited sample sizes or high proportions of categorical variables. By bridging the gap between theoretical advancements in DAG learning and practical applications in epidemiology, DAGSLAM provides a powerful and scalable tool for uncovering causal relationships in complex disease networks.

Performance analysis with mixed-type data

Our simulation results demonstrate that DAGSLAM excels in scenarios with a significant proportion of categorical variables. In scenarios 5, 6, and 7, where the proportion of categorical nodes is set at 10%, 20%, and 50%, respectively, DAGSLAM shows a clear advantage, with its relative performance over other algorithms becoming more pronounced as the proportion of categorical variables increases. This underscores its ability to effectively handle mixed-type data, which is a common feature of real-world biomedical datasets. However, in scenarios with minimal categorical variables (scenarios 3 and 4, with proportions of 2.5% and 1.0%, respectively), the performance of DAGSLAM is comparable to that of NOTEARS and DAGMA. This suggests that when the proportion of categorical variables is below 2.5%, treating all variables as continuous may be a viable alternative. Therefore, we recommend using DAGSLAM when the proportion of categorical variables is $\geq 2.5\%$, as it provides superior performance in these cases. However, users should be aware that DAGSLAM's computational time is longer compared to NOTEARS and DAGMA, and thus, a trade-off between computational efficiency and performance must be considered based on the specific characteristics of the dataset.

Furthermore, when examining the impact of the number of levels of categorical variables (m), DAGSLAM demonstrates significant advantages when $m = 2$ or 3. However, when $m \geq 4$, its performance aligns with that of NOTEARS and DAGMA. From a computational efficiency perspective, it may be more practical to treat multi-category variables as continuous and use NOTEARS or DAGMA in such cases. This flexibility allows users to optimize their choice of algorithm based on the specific structure of their data.

Instance analysis and real-world applications

The application of DAGSLAM to real-world datasets further underscores its utility. The DAG constructed from the NHANES dataset provides a comprehensive overview of the interrelationships among various health factors in elderly women. The findings highlight the importance of considering multiple risk factors when assessing health outcomes according to this demographic.

The positive relationships between fasting glucose, waist circumference, and glycohemoglobin underscore the critical role of metabolic health. The negative influence of waist circumference on HDL levels is particularly concerning, as low HDL is a significant risk factor for cardiovascular disease.

The observed negative relationships between alcohol consumption and CHD and diabetes are noteworthy and suggest that moderate alcohol intake may confer some protective effects. However, our findings should not be seen as an endorsement of alcohol consumption at any level. While there may be potential benefits of alcohol for insulin-sensitive cardiovascular health, these benefits must be carefully balanced against the associated risks [45–47]. The negative effects of alcohol are directly related to the amount consumed, and it is important to highlight that there is no completely safe level of alcohol intake [49, 50]. Consulting healthcare professionals before making dietary changes related to alcohol consumption is advisable.

Overall, these causal relationships provide valuable insights into the risk factors affecting the health of elderly women and emphasize the need for targeted interventions to address these interconnected issues.

Limitations and future directions

While DAGSLAM represents a significant advancement in DAG learning for mixed-type data, several limitations remain. One key limitation of DAGSLAM is its computational efficiency. The algorithm's design for handling mixed-type data requires individual vector operations for each column (or node), which inherently increases computational complexity. In contrast, NOTEARS and DAGMA treat all variables as continuous, allowing for efficient matrix operations that significantly reduce computation time. Future research should focus on exploring more efficient optimization methods to improve the computational performance of DAGSLAM. It is worth noting, however, that in our target application scenarios—such as disease risk factor networks—the node count (up to $d = 40$) and sample size (around $n = 1000$) are typically moderate, and the computational time of DAGSLAM remains within an acceptable range.

Additionally, the current implementation of DAGSLAM assumes linear relationships among variables, which may not fully capture the complexity of certain real-world datasets. Future work could explore the incorporation of nonlinear interactions, potentially through kernel-based methods or neural network architectures, to better model relationships in mixed-type data. This would further enhance the algorithm's ability to handle the intricate dependencies often found in biomedical data.

Moreover, the current algorithm does not support the inclusion of prior knowledge, such as blacklists or whitelists of edges, which could improve performance in scenarios where domain expertise is available. The incorporation of such constraints into the structure learning process is a promising direction for future research.

Finally, while DAGSLAM has shown robustness across various simulation scenarios, its performance in highly dense networks (where $\frac{s_0}{d} \geq 2$) and with exponentially distributed data requires further improvement. Future studies should focus on enhancing the algorithm's effectiveness in these challenging contexts, ensuring that it can be reliably applied to a wider range of epidemiological datasets.

Conclusions

In this study, we present DAGSLAM, a novel extension of the NOTEARS algorithm for learning DAGs from mixed-type data. Through extensive simulations and real-world applications, we demonstrate that DAGSLAM outperforms existing methods in terms of accuracy, scalability, and robustness. By enabling the integration of continuous and categorical variables, DAGSLAM provides a powerful framework for modelling complex disease networks, with significant implications for risk factor identification and public health research. Future work will focus on addressing the algorithm's limitations and further enhancing its capabilities for real-world applications.

Abbreviations

BNs	Bayesian Networks
DAGs	Directed Acyclic Graphs
CHD	Coronary Heart Disease
NHANES	National Health and Nutrition Examination Survey
SHD	Structural Hamming Distance
FDR	False discovery rate
TPR	True positive rate
FPR	False positive rate
ER	Erdős-Rényi
SF	Scale-free
NOTEARS	Non-combinatorial Optimization via Trace Exponential and Augmented lagRangian for Structure learning
DAGSLAM	Directed Acyclic Graphs Structure learning via Log-determinant and Augmented lagrangian for Mixed type data
HC	Hill-Climbing
MMHC	Max-Min Hill-Climbing
mDAG	Mixed Directed Acyclic Graph
DAGBagM	Directed Acyclic Graph Bagging with Mixed Variables
GLU	Fasting glucose
GHB	Glycohemoglobin
HDL	High-density lipoprotein cholesterol
HTN	Hypertension
LDL	Low-density lipoprotein cholesterol
SM	Smoking
TCHOL	Total Cholesterol
TG	Triglycerides
WAIST	Waist Circumference
AL	Alcohol consumption
BMI	Body mass index
DI	Diabetes

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12874-025-02582-6.

Additional file 1: Supplementary tables and figures

Additional file 2: Supplementary results

Acknowledgements

The authors would like to express their sincere gratitude to Xun Zheng et al. for their pioneering work on the NOTEARS algorithm, which provided valuable inspiration for this study. Additionally, the authors extend their thanks to the creators of the DAGMA algorithm for their innovative contributions, particularly the log-determinant acyclicity constraint, which has significantly influenced the development of our proposed method. The authors also wish to thank the National Health and Nutrition Examination Survey (NHANES) for making their public dataset available, which enabled the case study analysis presented in this work.

Authors' contributions

Yuanyuan Zhao: Conceptualization, Data analysis and Visualization, Writing – original draft, Writing – review & editing. Jinzhu jia: Conceptualization, Supervision, Writing – review & editing. All the authors have read and approved the final manuscript.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability

The datasets generated and used for the simulation experiments in this study, as well as the DAGSLAM algorithm implementation code, simulation experiment code, and detailed result files, are all available in the GitHub repository at https://github.com/yuanyuan-zhao-pku/DAGSLAM. The data used for the case study analysis in this work can be freely downloaded from the official website of the National Health and Nutrition Examination Survey (NHANES) at NHANES - National Health and Nutrition Examination Survey Homepage.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare no competing interests.

Received: 14 December 2024 Accepted: 29 April 2025
Published online: 06 June 2025

References

1. GBD 2019 Diseases and Injuries Collaborators. Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019. Lancet (London, England). 2020;396(10258):1204–22.
2. Tsao CW, Aday AW, Almarzooq ZI, Anderson CAM, Arora P, Avery CL, Baker-Smith CM, Beaton AZ, Boehme AK, Buxton AE, et al. Heart disease and stroke statistics-2023 update: a report from the American Heart Association. Circulation. 2023;147(8):e93–621.
3. Esser N, Legrand-Poels S, Piette J, Scheen AJ, Paquot N. Inflammation as a link between obesity, metabolic syndrome and type 2 diabetes. Diabetes Res Clin Pract. 2014;105(2):141–50.
4. Jr DWH, Lemeshow S, Sturdivant RX. Applied logistic regression, 3rd Edition. 2013.
5. Greenland S, Pearl J, Robins JM. Causal diagrams for epidemiologic research. Epidemiology. 1999;10(1):37–48.
6. Neuberg LG. Causality: models, reasoning, and inference. Econ Theory. 2003;19(4):675–85.
7. Lauritzen SL. Graphical models. Oxford: Oxford University Press; 1996.
8. Spirtes P, Glymour C, Scheines R. Causation, prediction, and search. 1993.
9. Kong D, Chen R, Chen Y, Zhao L, Huang R, Luo L, Lai F, Yang Z, Wang S, Zhang J, et al. Bayesian network analysis of factors influencing type 2 diabetes, coronary heart disease, and their comorbidities. BMC Public Health. 2024;24(1):1267.
10. Galindez G, Sadegh S, Baumbach J, Kacprowski T, List M. Network-based approaches for modeling disease regulation and progression. Comput Struct Biotechnol J. 2023;21:780–95.
11. Polotskaya K, Muñoz-Valencia CS, Rabasa A, Quesada-Rico JA, Orozco-Beltrán D, Barber X. Bayesian networks for the diagnosis and prognosis of diseases: a scoping review. Mach Learn Knowl Extr. 2024;6(2):1243–62.
12. Koller D, Friedman N. Probabilistic graphical models: principles and techniques - adaptive computation and machine learning. Cambridge: The MIT Press; 2009.
13. Heckerman D. A tutorial on learning with Bayesian networks. In: Innovations in Bayesian networks: theory and applications. edn. Berlin, Heidelberg: Springer Berlin Heidelberg. 2008:33–82.
14. Fenton N, Neil M. Risk assessment and decision analysis with Bayesian networks. Boca Raton: CRC Press; 2012.
15. Chickering MD. Optimal structure identification with greedy search. J Mach Learn Res. 2003;3:507–54.
16. Tsamardinos I, Brown LE, Aliferis CF. The max-min hill-climbing Bayesian network structure learning algorithm. Mach Learn. 2006;65:31–78.
17. Scutari M. Learning Bayesian networks with the bnlearn R package. J Stat Softw. 2009;35:1–22.
18. Magliacane S, Claassen T, Mooij JM. Ancestral causal inference. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona, Spain: Curran Associates Inc. 2016:4473–4481.
19. Bubnova AV, Deeva I, Kalyuzhnaya AV. MIxBN: library for learning Bayesian networks from mixed data. Procedia Comput Sci. 2021;193:494–503.
20. Chowdhury S, Wang R, Yu Q, Huntoon CJ, Karnitz LM, Kaufmann SH, Gygi SP, Birrer MJ, Paulovich AG, Peng J, et al. DAGBagM: learning directed acyclic graphs of mixed variables with an application to identify prognostic protein biomarkers in ovarian cancer. BMC Bioinformatics. 2020;23:321.
21. Zhong W, Dong L, Poston TB, Darville T, Spracklen CN, Wu D, Mohlke KL, Li Y, Li Q, Zheng X. Inferring regulatory networks from mixed observational data using directed acyclic graphs. Front Genet. 2020;11:8.
22. Nagarajan R, Scutari M, Lèbre S. Bayesian networks in R. 2013.
23. Buntine WL. Theory refinement on Bayesian networks. In: Conference on Uncertainty in artificial intelligence. 1991:1991.
24. Verma T, Pearl J. Causal networks: semantics and expressiveness. In: Conference on uncertainty in Artificial Intelligence. 2013:2013.
25. Colombo D, Maathuis MH. Order-independent constraint-based causal structure learning. Proc 30th Int Conf Sci Stat Database Manag. 2012;15:3741–3782.
26. Sedgewick AJ, Shi IW, Donovan RM, Benos PV. Learning mixed graphical models with separate sparsity parameters and stability-based model selection. BMC Bioinformatics. 2016;17:307–18.
27. Sedgewick AJ, Ramsey J, Spirtes P, Glymour C, Benos PV. Mixed graphical models for causal analysis of multi-modal variables. arXiv [Preprint] 2017. Available from: https://arxiv.org/abs/1704.02621.
28. Zheng X, Aragam B, Ravikumar P, Xing EP. DAGs with NO TEARS: continuous optimization for structure learning. In: Neural information processing systems. 2018:2018.
29. He C, Liu W, Ren J. Bayesian network structure learning: a review. In: 2022 6th Asian Conference on Artificial Intelligence Technology (ACAIT): 9–11 Dec. 2022. 2022;2022:1–7.
30. Schwarz G. Estimating the dimension of a model. Ann Stat. 1978;6:461–4.
31. Yu Y, Chen J, Gao T, Yu M. DAG-GNN: DAG structure learning with graph neural networks. In: International conference on machine learning. 2019:2019.
32. Lachapelle S, Brouillard P, Deleu T, Lacoste-Julien SJA. Gradient-based neural DAG learning. arXiv [Preprint] 2019. Available from: https://arxiv.org/abs/1906.02226.
33. Lee HC, Danieletto M, Miotto R, Cherng S, Dudley JT. Scaling structural learning with NO-BEARS to infer causal transcriptome networks. Pac Symp Biocomput Pac Symp Biocomput. 2019;25:391–402.
34. Zheng X, Dan C, Aragam B, Ravikumar P, Xing EP. Learning sparse nonparametric DAGs. In: International conference on artificial intelligence and statistics. 2019:2019.
35. Bello K, Aragam B, Ravikumar P. DAGMA: learning DAGs via M-matrices and a log-determinant acyclicity characterization. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. New Orleans, LA, USA: Curran Associates Inc.; 2022:Article 598.
36. Bengio Y, Lahlou S, Deleu T, Hu EJ, Tiwari M, Bengio E. GFlowNet foundations. J Mach Learn Res. 2023;24:Article 210.
37. Nocedal J, Wright SJ. Penalty and augmented Lagrangian methods. In: Nocedal J, Wright SJ, editors. Numerical optimization. 2nd ed. New York: Springer; 2006. p.497–528.
38. American Diabetes Association. Diagnosis and classification of diabetes mellitus. Diabetes Care. 2014;37(Suppl 1):S81–90.
39. Kahn SE, Cooper ME, Del Prato S. Pathophysiology and treatment of type 2 diabetes: perspectives on the past, present, and future. Lancet (London, England). 2014;383(9922):1068–83.
40. Koppes LL, Dekker JM, Hendriks HF, Bouter LM, Heine RJ. Moderate alcohol consumption lowers the risk of type 2 diabetes: a meta-analysis of prospective observational studies. Diabetes Care. 2005;28(3):719–25.
41. Davies MJ, Baer DJ, Judd JT, Brown ED, Campbell WS, Taylor PR. Effects of moderate alcohol intake on fasting insulin and glucose concentrations and insulin sensitivity in postmenopausal women: a randomized controlled trial. JAMA. 2002;287(19):2559–62.
42. Llamosas-Falcón L, Rehm J, Bright S, Buckley C, Carr T, Kilian C, Lasserre AM, Lemp JM, Zhu Y, Probst C. The relationship between alcohol consumption, BMI, and type 2 diabetes: a systematic review and dose-response meta-analysis. Diabetes Care. 2023;46(11):2076–83.
43. Knott C, Bell S, Britton A. Alcohol consumption and the risk of type 2 diabetes: a systematic review and dose-response meta-analysis of more than 1.9 million individuals from 38 observational studies. Diabetes Care. 2015;38(9):1804–1812.
44. Grundy SM, Cleeman JI, Daniels SR, Donato KA, Eckel RH, Franklin BA, Gordon DJ, Krauss RM, Savage PJ, Smith SC Jr, et al. Diagnosis and management of the metabolic syndrome: an American Heart Association/National Heart, Lung, and Blood Institute Scientific Statement. Circulation. 2005;112(17):2735–52.
45. Panagiotakos DB, Kouli GM, Magriplis E, Kyrou I, Georgousopoulou EN, Chrysohoou C, Tsigos C, Tousoulis D, Pitsavos C. Beer, wine consumption, and 10-year CVD incidence: the ATTICA study. Eur J Clin Nutr. 2019;73(7):1015–23.
46. Haseeb S, Alexander B, Baranchuk A. Wine and cardiovascular health: a comprehensive review. Circulation. 2017;136(15):1434–48.
47. Arranz S, Chiva-Blanch G, Valderas-Martínez P, Medina-Remón A, Lamuela-Raventós RM, Estruch R. Wine, beer, alcohol and polyphenols on cardiovascular disease and cancer. Nutrients. 2012;4(7):759–81.
48. Cecchini M, Filippini T, Whelton PK, Iamandii I, Di Federico S, Boriani G, Vinceti M. Alcohol intake and risk of hypertension: a systematic review

and dose-response meta-analysis of nonexperimental cohort studies. Hypertension (Dallas, Tex : 1979). 2024;81(8):1701–1715.

49. Burton R, Sheron N. No level of alcohol consumption improves health. Lancet (London, England). 2018;392(10152):987–8.
50. Rehm J, Gmel GE Sr, Gmel G, Hasan OSM, Imtiaz S, Popova S, Probst C, Roerecke M, Room R, Samokhvalov AV, et al. The relationship between different dimensions of alcohol use and the burden of disease-an update. Addiction (Abingdon, England). 2017;112(6):968–1001.

Publisher's Note