# Appendices

Appendix organization:

- Appendix A: Proofs for Theorems 1 and 2.
- Appendix B: Detailed Pseudo-code for FedCSL.
- Appendix C: Time Complexity of FedCSL.
- Appendix D: Privacy and Costs Discussion.
- Appendix E: Implementation Details.
- Appendix F: Detailed Experimental Results.

## A Proofs for Theorems 1 and 2

The conditional independence/dependence of two variables are defined as follows.

**Definition A.1** (Conditional Independence)**.** *Two variables $X_i$ and $X_j$ for $\forall i,j \in \{1,2,...,d\}$ are conditionally independent given a variable set $X_{\mathbf{z}}$ for $\forall \mathbf{z} \subseteq \{1,2,...,d\}$ if $P(X_i,X_j|X_{\mathbf{z}}) = P(X_i|X_{\mathbf{z}})P(X_j|X_{\mathbf{z}})$; otherwise, they are conditionally dependent given $X_{\mathbf{z}}$.*

In the following, $X_i \perp\!\!\!\perp X_j|X_{\mathbf{z}}$ and $X_i \not\perp\!\!\!\perp X_j|X_{\mathbf{z}}$ denote that $X_i$ and $X_j$ are conditionally independent and dependent given $X_{\mathbf{z}}$, respectively.

### A.1 Proof for Theorem 1

*Proof.* Given that the null hypothesis for a conditional independence (CI) test of $X_i$ and $X_j$ given an empty set is "$H_0 : X_i \perp\!\!\!\perp X_j|\emptyset$", the $G^2$ statistic is defined as

$$G^2(X_i,X_j|\emptyset) = 2\sum_a^{r_i}\sum_b^{r_j} O_{i,j}^{a,b}\ln(\frac{O_{i,j}^{a,b}}{E_{i,j}^{a,b}}), \tag{1}$$

where $O_{i,j}^{a,b}$ represents the observed number of samples satisfying $X_i = a$ and $X_j = b$, while $E_{i,j}^{a,b}$ represents the expected number of samples satisfying $X_i = a$ and $X_j = b$; further, $r_i$ and $r_j$ are respectively the domain (number of distinct values) of $X_i$ and $X_j$. Due to the null hypothesis of "$H_0 : X_i \perp\!\!\!\perp X_j|\emptyset$", the $G^2$ statistic can be reformulated as follows (**?**).

$$G^2(X_i,X_j|\emptyset) = 2\sum_a^{r_i}\sum_b^{r_j} O_{i,j}^{a,b}\ln(\frac{O_{i,j}^{a,b}n_{c_k}}{O_i^a O_j^b}), \tag{2}$$

where $O_i^a$ and $O_j^b$ denote the number of samples satisfying $X_i = a$ and $X_j = b$, respectively.

According to the large sample theory and central limit theorem, the $G^2$ statistic based on Eq. (2) is asymptotically distributed as $\chi^2$ with appropriate degrees of freedom (**?**), and the number of degrees of freedom (df) used in the test is calculated as

$$df = (r_i - 1)(r_j - 1). \tag{3}$$

Therefore, when $n_{c_k} \to \infty$, we have:

$$\begin{aligned} &\lim_{n_{c_k}\to\infty} G^2(X_i,X_j|\emptyset) = \lim_{n_{c_k}\to\infty} \chi^2(X_i,X_j|\emptyset) \\ &= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(O_{i,j}^{a,b} - E_{i,j}^{a,b})^2}{E_{i,j}^{a,b}} \\ &= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(O_{i,j}^{a,b} - E_{i,j}^{a,b})^2}{P(X_i = a)P(X_j = b)n_{c_k}}. \end{aligned} \tag{4}$$

Assume that $X_i \perp\!\!\!\perp X_j|\emptyset$ holds true in the underlying causal structure behind $\mathcal{D}^{c_k}$. Based on the law of large numbers, when $n_{c_k} \to \infty$, the sample distribution on $\mathcal{D}^{c_k}$ infinitely approaches the causal DAG used to generate $\mathcal{D}^{c_k}$. Thus, we have

$$P\{\lim_{n_{c_k}\to\infty} O_{i,j}^{a,b} - E_{i,j}^{a,b} = 0\} = 1, \tag{5}$$

or equivalently,

$$P\{\lim_{n_{c_k}\to\infty} (O_{i,j}^{a,b} - E_{i,j}^{a,b})^2 = 0\} = 1. \tag{6}$$

We also have:

$$\lim_{n_{c_k}\to\infty} P(X_i = a)P(X_j = b) = t, \tag{7}$$

where $t$ is a constant greater than 0 but less than or equal to 1. Thus, we can obtain:

$$\lim_{n_{c_k}\to\infty} P(X_i = a)P(X_j = b)n_{c_k} = +\infty. \tag{8}$$

Substitute Eq. (6) and Eq. (8) into Eq. (4), thus,

$$\begin{aligned} &\lim_{n_{c_k}\to\infty} G^2(X_i,X_j|\emptyset) = \lim_{n_{c_k}\to\infty} \chi^2(X_i,X_j|\emptyset) \\ &= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(O_{i,j}^{a,b} - E_{i,j}^{a,b})^2}{P(X_i = a)P(X_j = b)n_{c_k}} = 0. \end{aligned} \tag{9}$$

Let $U$ have the $\chi^2$ distribution with $(r_i - 1)(r_j - 1)$ degrees of freedom. Based on Eq. (9), then:

$$\lim_{n_{c_k}\to\infty} P(U > G^2(X_i,X_j|\emptyset)) = 1^+, \tag{10}$$

i.e., $\lim_{n_{c_k}\to\infty} \rho = 1^+$.

In conclusion, Theorem 1 is true. □

### A.2 Proof for Theorem 2

*Proof.* Assume that $X_i \not\perp\!\!\!\perp X_j|\emptyset$ holds true in the underlying causal structure behind $\mathcal{D}^{c_k}$. According the law of large numbers, when $n_{c_k} \to \infty$, the sample distribution on $\mathcal{D}^{c_k}$ infinitely approaches the true causal DAG used to generate $\mathcal{D}^{c_k}$. Let $A$ and $B$ denote $P(X_i = a, X_j = b)$ and $P(X_i = a)P(X_j = b)$, respectively. Thus, for $\forall a, b$, the following equation holds.

$$P\{\lim_{n_{c_k}\to\infty} A - B \neq 0\} = 1, \tag{11}$$

or equivalently,

$$P\{\lim_{n_{c_k}\to\infty} (A - B)^2 \neq 0\} = 1. \tag{12}$$

Further, we can obtain:

$$\lim_{n_{c_k}\to\infty} (A - B)^2 \neq 0. \tag{13}$$

According to Eq. (4), due to the null hypothesis of "$H_0 : X_i \perp\!\!\!\perp X_j|\emptyset$", we have:

$$\begin{aligned}
\lim_{n_{c_k}\to\infty} G^2(X_i, X_j|\emptyset) &= \lim_{n_{c_k}\to\infty} \chi^2(X_i, X_j|\emptyset) \\
&= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(O_{i,j}^{a,b} - E_{i,j}^{a,b})^2}{P(X_i = a)P(X_j = b)n_{c_k}} \\
&= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(An_{c_k} - Bn_{c_k})^2}{Bn_{c_k}} \\
&= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(A - B)^2 n_{c_k}}{B}.
\end{aligned} \tag{14}$$

Substitute Eq. (13) and Eq. (7) into Eq. (14), thus,

$$\begin{aligned}
\lim_{n_{c_k}\to\infty} G^2(X_i, X_j|\emptyset) &= \lim_{n_{c_k}\to\infty} \chi^2(X_i, X_j|\emptyset) \\
&= \lim_{n_{c_k}\to\infty} \sum_a^{r_i}\sum_b^{r_j} \frac{(A - B)^2}{t} n_{c_k} = +\infty.
\end{aligned} \tag{15}$$

Let $U$ have the $\chi^2$ distribution with $(r_i - 1)(r_j - 1)$ degrees of freedom. According to Eq. (15), we can obtain:

$$\lim_{n_{c_k}\to\infty} P(U > G^2(X_i, X_j|\emptyset)) = 0^-, \tag{16}$$

i.e., $\lim_{n_{c_k}\to\infty} \rho = 0^-$.

In conclusion, Theorem 2 is true. □

## B Detailed Pseudo-code for FedCSL

The pseudo-code of the FedCSL algorithm is detailed in Algorithm 1, and FedCSL consists of three steps: *federated causal neighbor learning* (Lines 1-22), *federated global skeleton construction* (Lines 23-36) and *federated skeleton orientation* (Lines 37-50). Specifically, in Step 1, FedCSL first employs a well-established HITON-PC (Aliferis et al. 2010) algorithm, which utilizes CI tests, to independently learn the potential causal neighborhood sets for each variable on every client. For client $c_k$, at the end of this process, we obtain the potential causal neighbor sets of all variables, $CN^{c_k} = \{CN_i^{c_k}\}_{i\in\{1,2,\ldots,d\}} = \{CN_1^{c_k}, CN_2^{c_k}, ..., CN_d^{c_k}\}$ (Line 2). Concurrently, at Line 3, we record all the p-values $\rho_{ij}^{c_k}$ $(i, j \in \{1, 2, ..., d\}$ and $i < j)$ returned by conducting CI tests between every pair of variables under the empty set condition and normalize them to be in the range [0, 1] (Lines 4-8). Subsequently, using the normalized p-values $\hat{\rho}_{ij}^{c_k}$ $(i, j \in \{1, 2, ..., d\}$ and $i < j)$, we calculate the weight for each client (Lines 10-12), which serves as a basis for the subsequent weighted aggregation strategy. However, it is important to note that the potential causal neighbor learned for each variable can differ across different clients. To address this, at Lines 13-19, FedCSL employs a weighted aggregation strategy to determine the optimal number of causal neighbors for each variable, i.e., $|CN_i^*|$ $(i \in \{1, 2, ..., d\})$. To facilitate weighted aggregation, we utilize a mask matrix $\varPsi_i \in \mathbb{R}^{m\times\max_{k=1}^m(|CN_i^{c_k}|)}$ to record the number of causal neighbors for $X_i$ obtained across $m$ clients as follows.

$$\underset{\substack{k=1,2,\ldots,m \\ \xi=1,2,\ldots,\max_{k=1}^m(|CN_i^{c_k}|)}}{\varPsi_i(k,\xi)} = \begin{cases} 1 & if\ \xi = |CN_i^{c_k}| \\ 0 & otherwise \end{cases}, \tag{17}$$

Algorithm 1: FedCSL

**Require:** $\mathcal{D}^{\mathcal{C}} = \{\mathcal{D}^{c_1}, \mathcal{D}^{c_2}, ..., \mathcal{D}^{c_m}\}$: $m$ local datasets held by $m$ clients $\mathcal{C} = \{c_1, c_2, ..., c_m\}$ (each dataset has the same variable space $X = (X_1, X_2, ..., X_d)$)

**Ensure:** $\mathcal{G}^*$: the final causal structure

```
    /* Step 1: federated causal neighbor learning */
 1: for $k = 1, 2, ..., m; i = 1, 2, ..., d$ do
 2:     $CN_i^{c_k}$=HITON-PC($\mathcal{D}^{c_k}, X_i$) // use HITON-PC (Aliferis et al. 2010) to learn the potential causal neighbor set of variable $X_i$ at client $c_k$.
 3:     Record the p-value $\rho_{ij}^{c_k}$ ($j \in \{1, 2, ..., d\}$ and $i < j$) returned by conducting CI tests between $X_i$ and $X_j$ under the empty set condition at client $c_k$.
 4:     if $\rho_{ij}^{c_k} \in [0, \alpha]$ then
 5:         $\hat{\rho}_{ij}^{c_k} = \frac{\alpha - \rho_{ij}^{c_k}}{\alpha}$ // normalize the p-value $\rho_{ij}^{c_k}$
 6:     else if $\rho_{ij}^{c_k} \in (\alpha, 1]$ then
 7:         $\hat{\rho}_{ij}^{c_k} = \frac{\rho_{ij}^{c_k} - \alpha}{1 - \alpha}$ // normalize the p-value $\rho_{ij}^{c_k}$
 8:     end if
 9: end for
10: for $k = 1, 2, ..., m$ do
11:     $w_{c_k} = \frac{2}{d(d-1)} \sum_{i=1}^d \sum_{j=i+1}^d \hat{\rho}_{ij}^{c_k}$ // calculate the weight value of client $c_k$
12: end for
13: for $i = 1, 2, ..., d$ do
14:     if $\| \varPsi_i \|_1 \neq 0$ then
15:         $|CN_i^*| = \overset{\circ}{M}ax([w_{c_1}, w_{c_2}, ..., w_{c_m}]\varPsi_i)$ // calculating the optimal number of causal neighbors for variable $X_i$
16:     else
17:         $|CN_i^*| = 0$
18:     end if
19: end for
20: for $i = 1, 2, ..., d$ do
21:     $CN_i^* = \overset{\circ}{T}op_{|CN_i^*|}([w_{c_1}, w_{c_2}, ..., w_{c_m}]B_i)$ // determining the optimal causal neighbors of variable $X_i$
22: end for
    /* Step 2: federated global skeleton construction */
23: for $i = 1, 2, ..., m; j = 1, 2, ..., (i - 1)$ do
24:     if $X_i \in CN_j^*$ and $X_j \in CN_i^*$ then
25:         There is an undirected edge connecting $X_i$ and $X_j$.
26:     else if $X_i \notin CN_j^*$ and $X_j \notin CN_i^*$ then
27:         There is no edge connection between $X_i$ and $X_j$.
28:     else {There is an asymmetric edge between $X_i$ and $X_j$.}
29:         if $\sum_{k=1}^m AES(\gamma, k) > 0$ then
30:             There is an undirected edge connecting $X_i$ and $X_j$.
31:         else
32:             There is no edge connection between $X_i$ and $X_j$.
33:         end if
34:     end if
35: end for
36: Obtain the final global skeleton $S^*$.
    /* Step 3: federated skeleton orientation */
37: for $k = 1, 2, ..., m$ do
38:     $A_k \xleftarrow[\mathcal{D}^{c_k}]{greedy\ search\ and\ scoring} S^*$
39: end for
40: $A^* = (A_1 * w_{c_1}) \oplus (A_2 * w_{c_2}) \oplus \cdots \oplus (A_m * w_{c_m})$
41: for $i = 1, 2, ..., m; j = 1, 2, ..., (i - 1)$ do
42:     if $A^*(i, j) > A^*(j, i)$ then
43:         $\mathcal{G}^* \Leftarrow \{X_i \to X_j\}$
44:     else if $A^*(i, j) \leq A^*(j, i)$ and $A^*(j, i) \neq 0$ then
45:         $\mathcal{G}^* \Leftarrow \{X_i \leftarrow X_j\}$
46:     else {There is no edge connection between $X_i$ and $X_j$.}
47:         $\mathcal{G}^* \Leftarrow \{X_i \nLeftrightarrow X_j\}$
48:     end if
49: end for
50: return $\mathcal{G}^*$
```

where $\max_{k=1}^{m}(|CN_i^{c_k}|)$ denotes the maximum number of causal neighbors for $X_i$ learned across all clients, and $\xi$ represents an indicator variable for the number of causal neighbors ($\xi \in \{1, 2, ..., \max_{k=1}^{m}(|CN_i^{c_k}|)\}$). Eq. (17) denotes that if the number of causal neighbors for $X_i$ learned at client $c_k$ is $\xi$, $\mathit{\Psi}_i(k, \xi) = 1$; otherwise, $\mathit{\Psi}_i(k, \xi) = 0$. Finally, FedCSL utilizes the weighted aggregation strategy again to determine the optimal causal neighbor set for each variable, i.e., $CN_i^*$ ($i \in \{1, 2, ..., d\}$) (Lines 20-22). Here, for $X_i$, its potential causal neighbor sets learned at all clients are recorded in a mask matrix $B_i \in \mathbb{R}^{m \times d}$ as follows.

$$\underset{k=1,2,...,m;j=1,2,...,d}{B_i(k,j)} = \begin{cases} 1 & if\ X_j \in CN_i^{c_k} \\ 0 & otherwise \end{cases}, \quad (18)$$

where if $X_j$ is a causal neighbor of $X_i$ at client $c_k$, then $B_i(k, j) = 1$; otherwise, $B_i(k, j) = 0$.

In Step 2, FedCSL utilizes the learned optimal causal neighbor sets for all variables, i.e. $CN^* = \{CN_1^*, CN_2^*, ..., CN_d^*\}$, to construct a global skeleton. Given any two variables $X_i$ and $X_j$, if there is an edge connecting $X_i$ and $X_j$ in the true casual structure, then $X_i$ and $X_j$ are necessarily each other's causal neighbors; otherwise, they are not causal neighbors of each other. Therefore, we set that if $X_i \in CN_j^*$ and $X_j \in CN_i^*$, we connect $X_i$ and $X_j$ with an undirected edge (Lines 24-25); if $X_i \notin CN_j^*$ and $X_j \notin CN_i^*$, we consider that there is no edge between $X_i$ and $X_j$ (Lines 26-27). However, we may also encounter cases where $X_i \in CN_j^*$ but $X_j \notin CN_i^*$ (or $X_i \notin CN_j^*$ and $X_j \in CN_i^*$). In this case (Line 28), there is an asymmetric edge between $X_i$ and $X_j$. To correct asymmetric edges, at Lines 29-33, FedCSL design a weighted scoring strategy to determine whether each asymmetric edge should be preserved as an undirected edge in the initial global skeleton (Lines 29-30) or removed from it (Lines 31-32). Here, the score of the $\gamma$-th asymmetric edge on the $k$-th client is denoted as $AES(\gamma, k)$, and we have:

$$AES(\gamma, k) = \begin{cases} (1+1) * w_{c_k} & if\ X_i \in CN_j^{c_k} \wedge X_j \in CN_i^{c_k} \\ (-1-1) * w_{c_k} & if\ X_i \notin CN_j^{c_k} \wedge X_j \notin CN_i^{c_k} \\ (-1+1) * w_{c_k} & if\ X_i \in CN_j^{c_k} \wedge X_j \notin CN_i^{c_k} \\ (1-1) * w_{c_k} & if\ X_i \notin CN_j^{c_k} \wedge X_j \in CN_i^{c_k}. \end{cases} \quad (19)$$

Finally, FedCSL corrects all asymmetric edges in the initial global skeleton and construct the final global skeleton $S^*$.

In Step 3, the cloud server first sends $S^*$ to each client, and then FedCSL uses a Bayesian score criteria, BDeu (Scutari 2016), and a search procedure, hill-climbing (Gámez, Mateo, and Puerta 2011) to greedily orient the undirected edges in $S^*$ at each client (Lines 37-39). Here, the BDeu score for the causal structure $\mathcal{G}_k$ learned on dataset $\mathcal{D}^{c_k}$ is defined as follows.

$$BDeu(\mathcal{G}_k, \mathcal{D}^{c_k}) = \log P(\mathcal{G}_k) + \sum_{i=1}^{d} \sum_{l=1}^{q_i} \Big[ \log \frac{\mathit{\Gamma}(\frac{H'}{q_i})}{\mathit{\Gamma}(H_{il} + \frac{H'}{q_i})} + \sum_{u=1}^{r_i} \log \frac{\mathit{\Gamma}(H_{ilu} + \frac{H'}{r_i q_i})}{\mathit{\Gamma}(\frac{H'}{r_i q_i})} \Big], \quad (20)$$

where $\mathit{\Gamma}$ is the Gamma function, $i$ is the index over the $d$ variables, $l$ is the index over the $q_i$ combinations of values of the parents of variable $X_i$, and $u$ is the index of the $r_i$ possible values (states) of $X_i$; further, $H_{ilu}$ is the number of instances in $\mathcal{D}^{c_k}$ where $X_i$ has the $u$-th value, and its parents have the $l$-th combination of values, and $H_{il} = \sum_{u=1}^{r_i} H_{ilu}$; $H'$ is the equivalent sample size (ESS, also sometimes known as the imaginary sample size, ISS) representing the confidence level in the prior parameters; $P(\mathcal{G}_k)$ is the prior probability of a particular graph structure which is generally assumed to be the same for all graphs and so can be ignored. By alternately performing the search procedure and the scoring criteria, FedCSL gets a global causal structure with the highest scoring at each client. Let $A_k$ denote the adjacency matrix corresponding to the learned causal structure $\mathcal{G}_k$ at client $c_k$, and "$A_k(i, j) = 1$" denotes that there is an edge from $X_i$ to $X_j$ in $\mathcal{G}_k$. Subsequently, at Line 40, FedCSL sends all adjacency matrices (i.e., $A_1$,$A_2$,...,$A_m$) back to the server to compute the aggregated adjacency matrix $A^*$. Here, the symbol $\oplus$ represents the element-wise addition of matrices. Finally, at Lines 41-49, FedCSL compares the elements at corresponding positions on the diagonal of matrix $A^*$ for obtaining the final causal structure (marked as $\mathcal{G}^*$). Here, we set a condition that if $A^*(i, j) > A^*(j, i)$, then there exists a directed edge from $X_i$ to $X_j$ (Lines 42-43); if $A^*(i, j) \leq A^*(j, i)$ and $A^*(j, i) \neq 0$, then there exists a directed edge from $X_j$ to $X_i$ (Lines 44-45); otherwise, there is no edge connected between $X_i$ and $X_j$ (Lines 46-47).

Finally, we obtain the final causal structure $\mathcal{G}^*$ (Line 50).

# C Time Complexity of FedCSL

Step 2 of FedCSL, i.e., *federated global skeleton construction*, is to design a weighted scoring strategy to correct each asymmetrical edge. During this computation, we can directly access the learning results of causal neighbor sets for each variable on every client obtained in Step 1, as well as the weight values for each client. Therefore, there is no need for any additional significant time overhead, and the time complexity of Step 2 can be considered negligible. Additionally, Step 3 of FedCSL performs the score-and-search strategy on the given final global skeleton $S^*$ rather than on an empty graph. It means that during the search process, FedCSL does not need to perform adding edges and removing edges operations, but only needs to perform reversing edges operation to achieve the highest score, that is, the entire search space is very small.

Therefore, the time complexity of FedCSL mainly lies in Step 1, and the computational cost of this step is measured via the number of CI (conditional independence) tests. Let $|\cdot|$ denote the size of a variable set and $p$ denote the largest size of the causal neighbor set of any variable in a dataset. For Step 1-1 in FedCSL, on a single dataset (a local dataset held by a client), the time complexity of the causal neighbor learning process of any variable is $O(2^p|X|) = O(2^p d)$ (Aliferis et al. 2010), and thus the time complexity of learning the causal neighbors for all $d$ variables on all $m$ local datasets is $O(2^p d^2 m)$. In the subsequent Step 1-2, Step 1-3 and Step 1-4, the FedCSL algorithm does not conduct any additional CI tests. Overall, the computational complexity of FedCSL is $O(2^p d^2 m)$ CI tests.

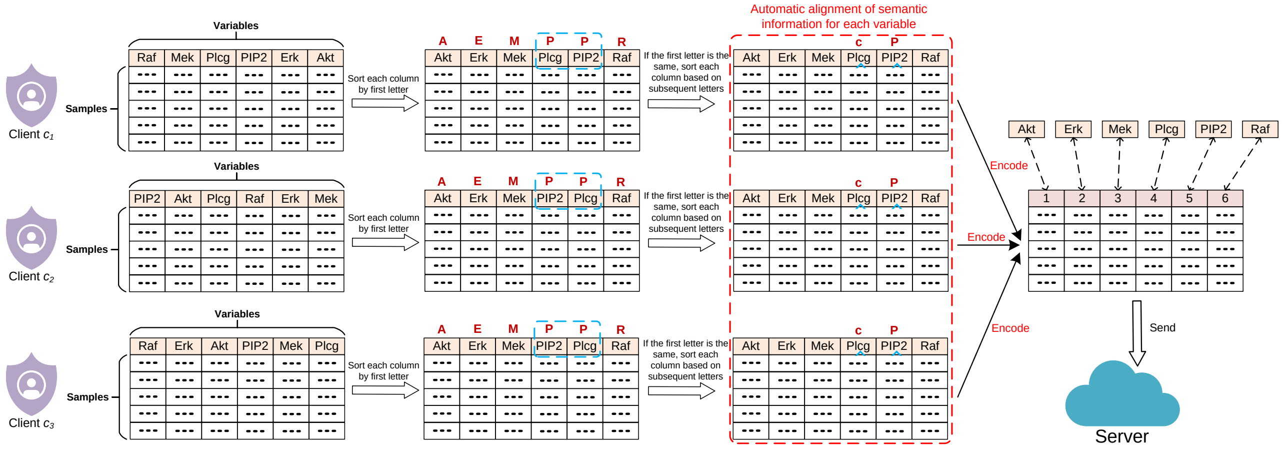

Variables
Raf Mek Plcg PIP2 Erk Akt
Samples
Client $c_1$
Variables
PIP2 Akt Plcg Raf Erk Mek
Samples
Client $c_2$
Variables
Raf Erk Akt PIP2 Mek Plcg
Samples
Client $c_3$
Sort each column by first letter
A E M P P R
Akt Erk Mek Plcg PIP2 Raf
Akt Erk Mek PIP2 Plcg Raf
If the first letter is the same, sort each column based on subsequent letters
Automatic alignment of semantic information for each variable
C P
Akt Erk Mek Plcg PIP2 Raf
Encode
Akt Erk Mek Plcg PIP2 Raf
1 2 3 4 5 6
Send
Server

Figure 1: An example illustrating the encryption of semantic information for variables.

# D Privacy and Costs Discussion

## D.1 Privacy Issues of FedCSL

The proposed FedCSL algorithm only exchanges structural information and weight parameter information throughout the entire federated learning process, without leaking the raw data stored at each client. Moreover, to prevent the inference of original local data from the learned structural information and weight parameters, we apply the following two techniques. (1) We apply additive homomorphic encryption technique (Paillier's scheme (Paillier 1999)) in the weighted aggregation strategy of Step 1 and Step 3 to prevent potential leakage of the relative sample sizes held by each client through their weight values. (2) To protect the semantic information of variables and avoid direct communication between clients, we apply an easily implementable privacy protection strategy (Huang et al. 2023) to the FedCSL algorithm. Specifically, the remote server instructs each client to sort and assign unique identifiers (e.g., "1," "2," "3," and so on) to the semantic information of all variables, following their alphabetical order. In case variables share the same first letter, they are further sorted based on the second letter of their semantic information, and this process continues. Subsequently, each client sends only the assigned identifiers to the remote server for aggregation, ensuring the protection of variable semantics. As shown in Fig. 1, we provide an example demonstrating how the semantic information of variables is encrypted.

## D.2 Communication Cost

Communication is a critical bottleneck in federated networks. Therefore, developing communication-efficient methods during the training process in federated learning is essential. Here, we argue that FedCSL introduces relatively low communication pressure. In Step 1, each client needs the server to transmit the learned causal neighbors for each variable and the weight values specific to each client, requiring a total of $O(md|CN_i^{c_k}| + m) = O(md^2 + m) = O(md^2)$ information cost. In Step 2, the entire federated global skeleton construction process is executed on the server, incurring no additional communication expenses. In Step 3, first, the final global skeleton needs to be sent from the server to each client. Then, the adjacency matrices learned at each client along with the weight values specific to each client are sent back to the server for obtaining the final causal structure through a weighted aggregation strategy, resulting in a total information cost of $O(m|S^*| + m|A_k| + m) = O(md^2 + md^2 + m) = O(md^2)$.

# E Implementation Details

All experiments were conducted on a computer with Intel Core i9-10900 3.70-GHz CPU, NVIDIA GeForce RTX 3060 GPU and 64-GB memory. The significance level for CI tests is set to 0.01. For the ADL[1], NOTEARS-ADMM[2], NOTEARS-MLP-ADMM[3], GS-FedDAG[4], AS-FedDAG[5] and FedPC[6] algorithms, we used the source codes provided by their authors. NOTEARS-ADMM and NOTEARS-MLP-ADMM use 0.3 as the threshold to prune edges in a causal structure, and GS-FedDAG and AS-FedDAG use 0.5 as the threshold, those are the same as the original paper. FedPC, ADL-AllData, ADL-Avg, ADL-Best, ADL-Voting and our method are implemented in MATLAB, and NOTEARS-ADMM, NOTEARS-MLP-ADMM, GS-FedDAG and AS-FedDAG are implemented in PYTHON.

# F Detailed Experimental Results

## F.1 More Evaluation Metrics.

Let $TP$ be the number of true positives (edges in both the true structure and learned structure); $FP$ the number of false positives (edges in the learned structure but not in the true causal structure; $TN$ the number of true negatives (edges

[1] https://github.com/Xianjie-Guo/ADL.

[2] https://github.com/ignavierng/notears-admm.

[3] https://github.com/ignavierng/notears-admm.

[4] https://github.com/ErdunGAO/FedDAG.

[5] https://github.com/ErdunGAO/FedDAG.

[6] https://github.com/Xianjie-Guo/FedPC.

not in either the true or learned structure); and $FN$ the number of false negatives (edges in the true structure but missing from the learned structure). To further evaluate the performance of FedCSL in comparison to its rivals, we employ five new metrics, False Discovery Rate (FDR), True Positive Rate (TPR), Reverse, Miss and Extra, as follows.

- *False Discovery Rate (FDR)*. FDR is the ratio of false edges in the learned causal structure to the edges in the learned causal structure. That is, $FDR = \frac{FP}{TP+FP}$.
- *True Positive Rate (TPR)*. TPR is the ratio of correct edges in the learned causal structure to total edges in the true causal structure. That is, $TPR = \frac{TP}{TP+FN}$.
- *Reverse*. The number of edges with wrong directions according to the true causal structure.
- *Miss*. The number of missing edges in the causal structure learned by the algorithm against the true causal structure.
- *Extra*. The number of extra edges in the learned causal structure.

In all figures, (↑) means the higher the better, and (↓) means the lower the better.

### F.2 Experiment Results on Benchmark Bayesian Network Data

In this section, we report the experimental results of FedCSL and its baselines on benchmark BN datasets in terms of FDR, TPR, Reverse, Miss and Extra metrics.

From Fig. 2, we observe that in most cases, our method achieves higher TPR (True Positive Rate) and lower FDR (False Discovery Rate), Reverse, Miss, and Extra values compared to the baseline algorithms, which validates the superiority of our method. As the number of clients increases, all algorithms experience a certain degree of performance degradation. However, our method demonstrates excellent stability, particularly on the Pigs and Gene BN datasets. In comparison to the best baseline FedPC, when the number of clients is greater than 5, FedPC exhibits a significant decline in performance, whereas our method remains remarkably stable.

### F.3 Experiment Results on High-dimensional Synthetic Data

Since existing federated CSL methods cannot be scaled to such high-dimensional datasets (with 5,000 variables). Thus, we develop the following four new algorithms (i.e., ADL-AllData, ADL-Avg, ADL-Best and ADL-Voting) using an efficient and effective CSL method, ADL (Guo et al. 2023), and compare them with our method on the high-dimensional synthetic datasets.

- ADL-AllData. We centralize all clients' data to a single dataset and run the ADL algorithm on it.
- ADL-Avg. We first run the ADL algorithm at each client independently for obtaining $m$ causal structures, and then calculate the average value of the metrics corresponding to all learned causal structures as the final result.
- ADL-Best. We first run ADL at each client independently to get $m$ causal structures, and then select the causal structure with the highest F1 score as the final output.
- ADL-Voting. We apply a voting method (Na and Yang 2010) to the ADL algorithm.

In this section, we present the experimental results of FedCSL and four new baselines on the high-dimensional synthetic datasets in terms of FDR, TPR, Reverse, Miss and Extra metrics. As shown in Fig. 3, our method outperforms all other baseline algorithms significantly in terms of TPR and Reverse metrics. Regarding the FDR metric, our method is only slightly worse than ADL-Voting when the number of clients is 2. Additionally, for the Miss and Extra metrics, our method performs slightly worse than ADL-AllData and ADL-Voting, respectively.

These experimental results further demonstrate the substantial superiority of our method not only on the benchmark Bayesian network datasets but also on the high-dimensional synthetic datasets, highlighting the outstanding performance of the FedCSL algorithm in causal structure learning tasks under privacy-preserving scenarios.

### F.4 Statistical Tests

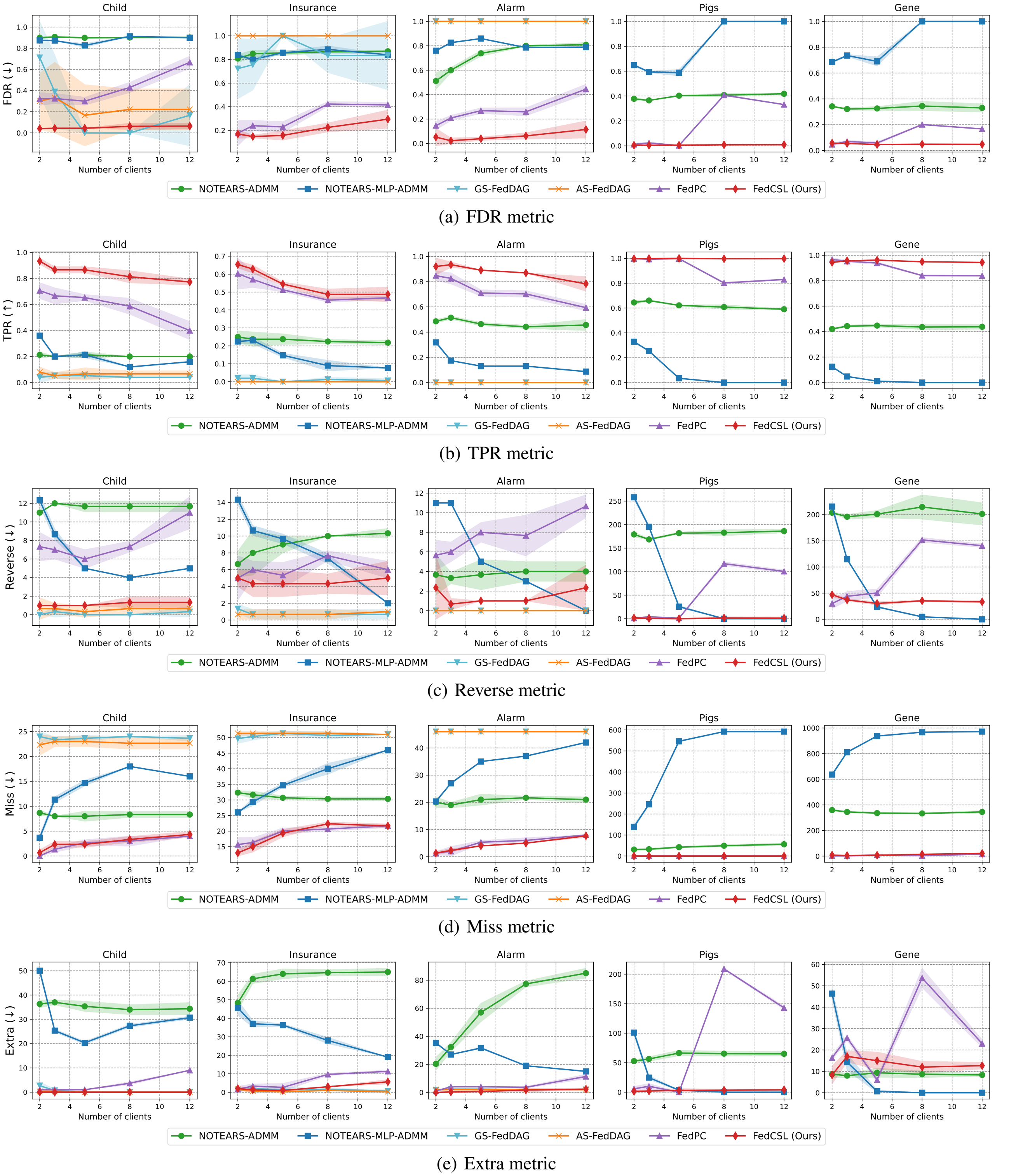
In this section, we adopt the Friedman test and Nemenyi test (Demšar 2006) to verify whether FedCSL is significantly better than other methods.

We first perform the Friedman test at the 0.05 significance level under the null-hypothesis which states that the performance of all algorithms is the same on all datasets (i.e., the average ranks of all algorithms are equivalent). The average ranks of FedCSL and the baselines when using different metrics are summarized in Table 1. As GS-FedDAG and AS-FedDAG do not yield any output on the Pigs and Gene datasets, we only utilize experimental results from the Child, Insurance, and Alarm datasets across different numbers of clients for conducting statistical tests. From Table 1, we can see that the null hypothesis is rejected on these two metrics (i.e. SHD and F1 score). We also note that FedCSL performs better than the baselines (the lower rank value is better).

Table 1: The average ranks of FedCSL and the baselines using SHD and F1 metrics. (Since GS-FedDAG and AS-FedDAG fail to produce any output on the Pigs and Gene datasets, we only employ experimental results on the Child, Insurance and Alarm datasets across various numbers of clients to conduct statistical tests.)

| Algorithm | Avg rank | |
|---|---|---|
| | SHD | F1 |
| NOTEARS-ADMM | 5.67 | 3.33 |
| NOTEARS-MLP-ADMM | 5.2 | 3.8 |
| GS-FedDAG | 3.77 | 5.47 |
| AS-FedDAG | 3.27 | 5.4 |
| FedPC | 2.1 | 2 |
| FedCSL (Ours) | **1** | **1** |

To further analyze the significant difference between FedCSL and the baselines, we perform the Nemenyi test, which



(a) FDR metric

(b) TPR metric

(c) Reverse metric

(d) Miss metric

(e) Extra metric

Figure 2: Structure learning results on the benchmark BN datasets. There are 5,000 samples in total, distributed unevenly across $\{2, 3, 5, 8, 12\}$ clients. We show the performance of all methods in five metrics (FDR, TPR, Reverse, Miss and Extra from top to bottom). Note that due to insufficient memory, GS-FedDAG and AS-FedDAG are unable to produce results on the Pigs and Gene networks.

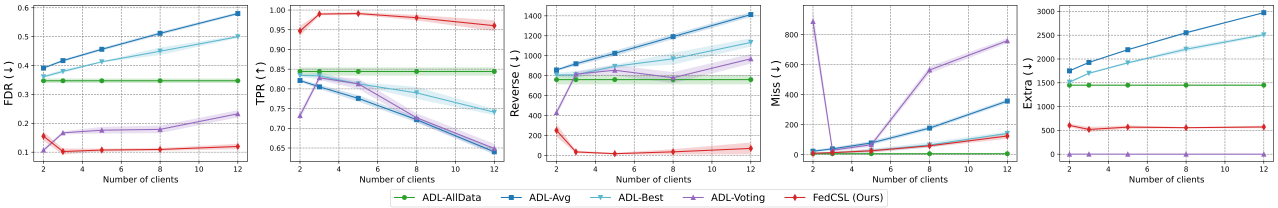
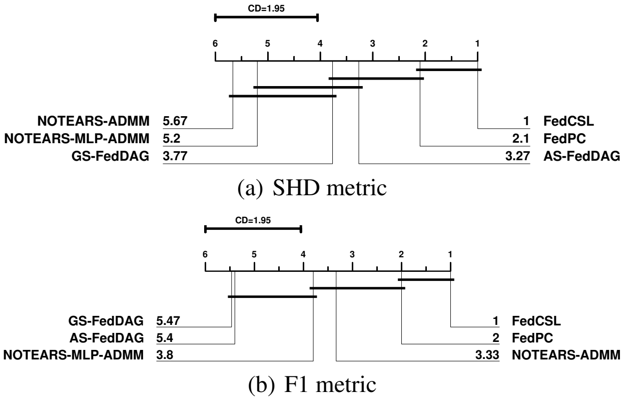


Figure 3: Structure learning results on the high-dimensional synthetic datasets with 5,000 variables. There are 5,000 samples in total, distributed unevenly across $\{2, 3, 5, 8, 12\}$ clients. We show the performance of all methods in five metrics (FDR, TPR, Reverse, Miss and Extra from left to right).

states that the performance levels of two algorithms are significantly different if the corresponding average ranks differ by at least one critical difference (CD). The CD for the Nemenyi test is calculated as follows (i.e., Eq. (21)).

$$\mathrm{CD} = q_{\alpha,\theta}\sqrt{\frac{\theta(\theta+1)}{6\eta}}, \tag{21}$$

where $\alpha$ is the significance level, $\theta$ is the number of comparison algorithms, and $\eta$ denotes the number of datasets with different numbers of clients. In our experiments, $\theta = 6$, $q_{\alpha=0.05,\theta=6} = 2.85$ at significance level $\alpha = 0.05$. Whether using SHD or F1 metrics, $\eta = 3 * 5 = 15$ (three benchmark BN datasets across $\{2, 3, 5, 8, 12\}$ clients), and thus $\mathrm{CD} = 1.95$.



(a) SHD metric

(b) F1 metric

Figure 4: Crucial difference diagram of the Nemenyi test for SHD and F1 metrics on the benchmark BN datasets.

Figs. 4(a) and 4(b) provide the CD diagrams, where the average rank of each algorithm is marked along the axis (lower ranks to the right). Whether using SHD or F1 metrics, we observe that FedCSL significantly outperforms NOTEARS-ADMM, NOTEARS-MLP-ADMM, GS-FedDAG and AS-FedDAG, and FedCSL achieves a comparable performance against FedPC. Additionally, FedCSL is the only algorithm that achieves the lowest rank value whether using SHD or F1 metrics.

## References

Aliferis, C. F.; Statnikov, A.; Tsamardinos, I.; Mani, S.; and Koutsoukos, X. D. 2010. Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: algorithms and empirical evaluation. *Journal of Machine Learning Research*, 11(1).

Demšar, J. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7: 1–30.

Gámez, J. A.; Mateo, J. L.; and Puerta, J. M. 2011. Learning Bayesian networks by hill climbing: efficient methods based on progressive restriction of the neighborhood. *Data Mining and Knowledge Discovery*, 22(1): 106–148.

Guo, X.; Yu, K.; Liu, L.; Li, P.; and Li, J. 2023. Adaptive Skeleton Construction for Accurate DAG Learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(10): 10526–10539.

Huang, J.; Guo, X.; Yu, K.; Cao, F.; and Liang, J. 2023. Towards Privacy-Aware Causal Structure Learning in Federated Setting. *IEEE Transactions on Big Data*, 9(6): 1525–1535.

Na, Y.; and Yang, J. 2010. Distributed Bayesian network structure learning. In *2010 IEEE International Symposium on Industrial Electronics*, 1607–1611. IEEE.

Paillier, P. 1999. Public-key cryptosystems based on composite degree residuosity classes. In *International Conference on The Theory and Applications of Cryptographic Techniques*, 223–238. Springer.

Scutari, M. 2016. An empirical-Bayes score for discrete Bayesian networks. In *Conference on Probabilistic Graphical Models*, 438–448. PMLR.