Appendix: Federated Causally Invariant Feature Learning

Contents

A Theoretical Analysis

A.1 Proof for Lemma 1

Lemma 1. *If for* $\forall j,\ 0 < P(\mathbf{X}^{c_k}_{\cdot,j} = 1|\xi(\mathbf{X}^{c_k}_{\cdot,-j})) < 1$*, and* $\mathbf{X}^{c_k}$ *is binary, then for* $\forall i,\ 0 < P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x) < 1$*, where* $([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})$ *is a sample of length* $(p+1)$*, formed by concatenating the* i*-th row of the low-dimensional representation space* $[\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}$ *with* $\mathbf{X}^{c_k}_{i,j}$*.*

Proof. Assume that $T = \mathbf{X}^{c_k}_{\cdot,j}$ is the treatment feature. Since for $\forall j,\ 0 < P(\mathbf{X}^{c_k}_{\cdot,j} = 1|\xi(\mathbf{X}^{c_k}_{\cdot,-j})) < 1$, and $\mathbf{X}^{c_k}$ is binary, it also holds that for $\forall j,\ 0 < P(\mathbf{X}^{c_k}_{\cdot,j} = 0|\xi(\mathbf{X}^{c_k}_{\cdot,-j})) < 1$. Therefore, $\exists$ a sample $(x^a_1, x^a_2, \ldots, x^a_p)$ such that $P([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot} = (x^a_1, x^a_2, \ldots, x^a_p)) > 0$ holds. From

$$
\begin{aligned}
&P\left(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = (x^a_1, x^a_2, \ldots, x^a_p, x_{p+1})\right) \\
=&P\left([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot} = (x^a_1, x^a_2, \ldots, x^a_p), \mathbf{X}^{c_k}_{i,j} = x_{p+1}\right) \\
=&P\left([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot} = (x^a_1, x^a_2, \ldots, x^a_p)\right) P\left(\mathbf{X}^{c_k}_{i,j} = x_{p+1}|[\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot} = (x^a_1, x^a_2, \ldots, x^a_p)\right),
\end{aligned} \tag{1}
$$

we have:

$$
0 < P\left(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = (x^a_1, x^a_2, \ldots, x^a_p, x_{p+1})\right) < 1 \tag{2}
$$

for $x_{p+1} = 0$ or $x_{p+1} = 0$ ($\mathbf{X}_{i,j}^{c_k}$ is binary.). Let $q \in \{1, 2, \dots, p\}$, we have

$$\begin{aligned} &P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_p^a, x_{p+1})\right) \\ =&P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1}), [\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = x_q^a\right) \\ =&P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) \cdot \\ &P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = x_q^a | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right), \end{aligned} \tag{3}$$

and according to Equation (2), we can obtain

$$P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) > 0. \tag{4}$$

By substituting Equations (2) and (4) into Equation (3), we have:

$$0 < P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = x_q^a | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) < 1. \tag{5}$$

Similarly, we can obtain:

$$\begin{aligned} 0 < P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = 0 | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) < 1, \\ 0 < P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = \frac{1}{\omega} | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) < 1, \\ 0 < P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = \frac{2}{\omega} | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) < 1, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ 0 < P\left([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} = 1 | ([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,-q}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, \dots, x_{q-1}^a, x_{q+1}^a, \dots, x_p^a, x_{p+1})\right) < 1. \end{aligned} \tag{6}$$

Thus, for $\forall x_q, x_{p+1}$, we can obtain:

$$0 < P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot}, \mathbf{X}_{i,j}^{c_k}) = (x_1^a, x_2^a, \dots, x_{q-1}^a, x_q, x_{q+1}^a, x_p^a, x_{p+1})\right) < 1. \tag{7}$$

Then, we repeat the above for all other low-dimensional representation features one by one, and have:

$$0 < P\left(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot}, \mathbf{X}_{i,j}^{c_k}) = (x_1, x_2, \dots, x_{q-1}, x_q, x_{q+1}, x_p, x_{p+1})\right) < 1. \tag{8}$$

Or equivalently,

$$0 < P(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot}, \mathbf{X}_{i,j}^{c_k}) = x) < 1 \tag{9}$$

for $\forall x$. In conclusion, Lemma 1 is proved. □

A.2 Proof for Theorem 1

Theorem 1. *Under Lemma 1, if the dimension p of the low-dimensional representation space $\xi(\mathbf{X}_{\cdot,-j}^{c_k})$ is finite, then $\exists$ a W^{c_k} such that*

$$P\left(\lim_{n_k \to \infty} \sum_{j=1}^{d} \left\| \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot \mathbf{X}_{\cdot,j}^{c_k})}{(W^{c_k})^T \cdot \mathbf{X}_{\cdot,j}^{c_k}} - \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot (1 - \mathbf{X}_{\cdot,j}^{c_k}))}{(W^{c_k})^T \cdot (1 - \mathbf{X}_{\cdot,j}^{c_k})} \right\|_2^2 = 0\right) = 1. \tag{10}$$

In particular, a W^{c_k} solution that satisfies Equation (10) is $\hat{W}_i^{c_k} = \frac{1}{P(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot}, \mathbf{X}_{i,j}^{c_k}) = x)}$.

Proof. For $\forall j$, we have

$$\left\| \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot \mathbf{X}_{\cdot,j}^{c_k})}{(W^{c_k})^T \cdot \mathbf{X}_{\cdot,j}^{c_k}} - \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot (1 - \mathbf{X}_{\cdot,j}^{c_k}))}{(W^{c_k})^T \cdot (1 - \mathbf{X}_{\cdot,j}^{c_k})} \right\|_2^2 \geq 0. \tag{11}$$

Let $q \in \{1, 2, \dots, p\}$ and $p = |\xi(\mathbf{X}_{\cdot,-j}^{c_k})|$. So, Equation (10) can be simplified to

$$P\left(\lim_{n_k \to \infty} \left(\frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} \neq 0, \mathbf{X}_{i,j}^{c_k} = 1} W_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k} = 1} W_i^{c_k}} - \frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q} \neq 0, \mathbf{X}_{i,j}^{c_k} = 0} W_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k} = 0} W_i^{c_k}} \right) = 0\right) = 1. \tag{12}$$

Based on Lemma 1, we have $0 < P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x) < 1$ for $\forall i, x$. Thus, for $\hat{W}^{c_k}_i = \frac{1}{P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x)}$ and g=0 or 1,

$$\begin{aligned}
&\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:\mathbf{X}^{c_k}_{i,j}=g} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{x:x_{p+1}=g} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \sum\nolimits_{x:x_{p+1}=g} \frac{1}{n_k} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \sum\nolimits_{x:x_{p+1}=g} \frac{1}{n_k} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} \frac{1}{P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x)}.
\end{aligned} \tag{13}$$

According to the law of large numbers, we have

$$\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} = P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x). \tag{14}$$

By substituting Equation (14) into Equation (13), we can obtain

$$\begin{aligned}
&\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:\mathbf{X}^{c_k}_{i,j}=g} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \sum\nolimits_{x:x_{p+1}=g} P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x) \frac{1}{P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x)} \\
=&(\omega+1)^p
\end{aligned} \tag{15}$$

due to $[\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,q} \in \{0, \frac{1}{\omega}, \frac{2}{\omega}, \ldots, 1\}$. Thus, we have

$$\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:\mathbf{X}^{c_k}_{i,j}=0} \hat{W}^{c_k}_i = (\omega+1)^p \tag{16}$$

and

$$\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:\mathbf{X}^{c_k}_{i,j}=1} \hat{W}^{c_k}_i = (\omega+1)^p. \tag{17}$$

Further, we can obtain

$$\begin{aligned}
&\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{i:[\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,q}\neq 0, \mathbf{X}^{c_k}_{i,j}=1} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \frac{1}{n_k} \sum\nolimits_{x:x_q\neq 0} \sum\nolimits_{x:x_{p+1}=1} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \sum\nolimits_{x:x_q\neq 0} \sum\nolimits_{x:x_{p+1}=1} \frac{1}{n_k} \sum\nolimits_{i:([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j})=x} \hat{W}^{c_k}_i \\
=&\lim_{n_k\to\infty} \sum\nolimits_{x:x_q\neq 0} \sum\nolimits_{x:x_{p+1}=1} P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x) \frac{1}{P(([\xi(\mathbf{X}^{c_k}_{\cdot,-j})]_{i,\cdot}, \mathbf{X}^{c_k}_{i,j}) = x)} \\
=&(\omega+1)^{p-1} \frac{\omega}{\omega+1} \\
=&\omega(\omega+1)^{p-2}.
\end{aligned} \tag{18}$$

Similarly, since $\mathbf{X}_{i,j}^{c_k}$ is binary, we have

$$\begin{aligned}
&\lim_{n_k\to\infty}\frac{1}{n_k}\sum\nolimits_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q}\neq 0,\mathbf{X}_{i,j}^{c_k}=0}\hat{W}_i^{c_k}\\
=&\lim_{n_k\to\infty}\frac{1}{n_k}\sum\nolimits_{x:x_q\neq 0}\sum\nolimits_{x:x_{p+1}=0}\sum\nolimits_{i:([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot},\mathbf{X}_{i,j}^{c_k})=x}\hat{W}_i^{c_k}\\
=&\lim_{n_k\to\infty}\sum\nolimits_{x:x_q\neq 0}\sum\nolimits_{x:x_{p+1}=0}\frac{1}{n_k}\sum\nolimits_{i:([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot},\mathbf{X}_{i,j}^{c_k})=x}\hat{W}_i^{c_k}\\
=&\lim_{n_k\to\infty}\sum\nolimits_{x:x_q\neq 0}\sum\nolimits_{x:x_{p+1}=0}P(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot},\mathbf{X}_{i,j}^{c_k})=x)\frac{1}{P(([\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,\cdot},\mathbf{X}_{i,j}^{c_k})=x)}\\
=&(\omega+1)^{p-1}\frac{\omega}{\omega+1}\\
=&\omega(\omega+1)^{p-2}.
\end{aligned}\tag{19}$$

Substituting Equations (16), (17), (18) and (19) into Equation (12), we obtain:

$$\begin{aligned}
&\lim_{n_k\to\infty}\left(\frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q}\neq 0,\mathbf{X}_{i,j}^{c_k}=1}\hat{W}_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k}=1}\hat{W}_i^{c_k}}-\frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q}\neq 0,\mathbf{X}_{i,j}^{c_k}=0}\hat{W}_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k}=0}\hat{W}_i^{c_k}}\right)\\
=&\frac{\omega(\omega+1)^{p-2}}{(\omega+1)^p}-\frac{\omega(\omega+1)^{p-2}}{(\omega+1)^p}\\
=&0.
\end{aligned}\tag{20}$$

Or equivalently,

$$P\left(\lim_{n_k\to\infty}\left(\frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q}\neq 0,\mathbf{X}_{i,j}^{c_k}=1}\hat{W}_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k}=1}\hat{W}_i^{c_k}}-\frac{\sum_{i:[\xi(\mathbf{X}_{\cdot,-j}^{c_k})]_{i,q}\neq 0,\mathbf{X}_{i,j}^{c_k}=0}\hat{W}_i^{c_k}}{\sum_{i:\mathbf{X}_{i,j}^{c_k}=0}\hat{W}_i^{c_k}}\right)=0\right)=1.\tag{21}$$

In conclusion, Theorem 1 is proved. □

B Related Work

B.1 Federated Feature Selection

With the advent of big data and federated learning (FL) era [1], traditional feature selection methods have shown unacceptable performance in handling data heterogeneity in federated environments. Federated Feature Selection (FFS) has emerged to address this issue. The importance of this research direction is mainly reflected in two aspects: first, it enables multiple participants to collaboratively select high-quality feature subsets without sharing raw data; second, it improves the effect of feature selection while protecting the data privacy of each participant through the FL framework. Our work focuses specifically on horizontal federated feature selection scenarios, where participants share the same feature space but have different sample sets. This section will introduce several representative works in this field.

The pioneering work in this area is Fed-FiS [2], which the authors claim to be the first feature selection algorithm in an FL system. Fed-FiS utilizes information-theoretic measures to estimate feature-feature mutual information and feature-label mutual information, generating local feature subsets on each user device. The central server ranks each feature based on the federated values across features and labels obtained from each device, generating a global dominant feature subset.

Building upon Fed-FiS, a recent work called Fed-MOFS [3] was proposed as an extension and improvement. Fed-MOFS introduces a multi-objective optimization approach to rank features based on their relevance and redundancy. While Fed-FiS uses a scoring function for global feature ranking, Fed-MOFS employs multi-objective optimization to prioritize features with higher relevance and lower redundancy. Another recent contribution to horizontal federated feature selection is the Fed-mRMR algorithm [4]. This method adapts the classic minimum redundancy maximum relevance (mRMR) algorithm to federated learning settings. Fed-mRMR achieves lossless federated feature

selection by extracting certain statistics from the dataset and applying the mRMR algorithm to rank and select relevant features while preserving privacy.

To further enhance the performance of FFS in the Internet of Things scenarios, an unsupervised FFS method named FSHFL [5] was introduced. FSHFL applies a feature relevance outlier detection method combined with an improved one-class support vector machine to remove useless features. Additionally, it proposes a feature relevance hierarchical clustering algorithm FRHC [5] for federated overlapping feature selection.

Building upon the concept of FFS, a multi-participant federated evolutionary feature selection algorithm [6] was proposed to address the situation where some participants have imbalanced data or even miss some classes. This algorithm introduces a trusted third party and adopts a multi-level joint sample-filling strategy to fill imbalanced or empty classes on each participant. It then realizes federated evolutionary feature selection by periodically sharing the optimal feature subsets obtained by participants based on particle swarm optimization. The application of FFS has also been explored in autonomous driving scenarios [7]. In this work, an FFS algorithm was proposed in which vehicles collaborate to filter out less relevant attributes without exchanging raw data. The algorithm consists of two components: a mutual-information-based feature selection algorithm run by vehicles and a novel aggregation function based on Bayes' theorem executed at the edge.

Recently, a comprehensive FFS framework was proposed [8], which introduces a trusted third party to process and integrate optimal feature subsets from multiple participants. Under this framework, a federated evolutionary feature selection algorithm based on particle swarm optimization was developed to effectively solve feature selection problems with multiple participants under privacy protection. Two new operators satisfying the requirement of privacy protection were designed: the feature assembling strategy with multi-participant cooperation and the swarm initialization strategy guided by the assembling solution, to improve the performance of the algorithm.

B.2 Causal Feature Selection

In recent years, causal feature selection has garnered significant attention. Unlike traditional feature selection methods, causal feature selection aims to discover causal relationships between features and the target variable, i.e., primarily identify causal features by discovering the Markov Blanket (MB) of the target variable, thereby improving the interpretability and robustness of models [9]. In theory, the MB of the label variable is the optimal solution to the feature selection problem [10]. Causal feature selection can be applied not only to static environments but also to dynamic environments, such as time series data.

Existing causal feature selection algorithms can be categorized into two main classes: simultaneous MB learning and non-simultaneous MB learning. Simultaneous MB learning algorithms employ a forward-backward strategy to greedily search for parents and children (PC) and spouses (SP) simultaneously without distinguishing the PC of the target variable from its SP. Representative algorithms include IAMB [11], and its variants such as LRH [12], FBED^K[13], and EAMB[14]. These algorithms are time-efficient but require an exponential number of samples, leading to errors in conditional independence tests when samples are insufficient.

To alleviate the data inefficiency problem, non-simultaneous MB learning methods have been proposed, adopting a divide-and-conquer strategy to learn PC and SP separately. Representative algorithms include MMMB [15], HITON-MB [10], CCMB [16], DCMB [17], and CFS-MI [18]. These algorithms further improve the accuracy of MB discovery by considering true positive features discarded during the MB search process. Recently, CVS [19], a novel causal feature selection algorithm, has been introduced to address the problem of stable prediction across unknown test data. It utilizes conditional independence tests to screen out non-causal features and reduce spurious correlations by leveraging a seed variable, increasing the stability of prediction across unknown test data.

While causal feature selection has made significant strides, the majority of current algorithms fail to address the issue of sample selection bias, which can introduce spurious correlations between features and the target variable. As a solution to this problem, PCFS [20] has been developed to estimate sample weights and mitigate the impact of spurious correlations. Furthermore, causal feature selection has been applied to address the out-of-distribution (OOD) generalization problem. Recently, Wang et al. proposed the CIFD framework that combines causal structure learning and causal effect estimation to select a high-quality causal variable set and achieve better OOD generalization [21]. It

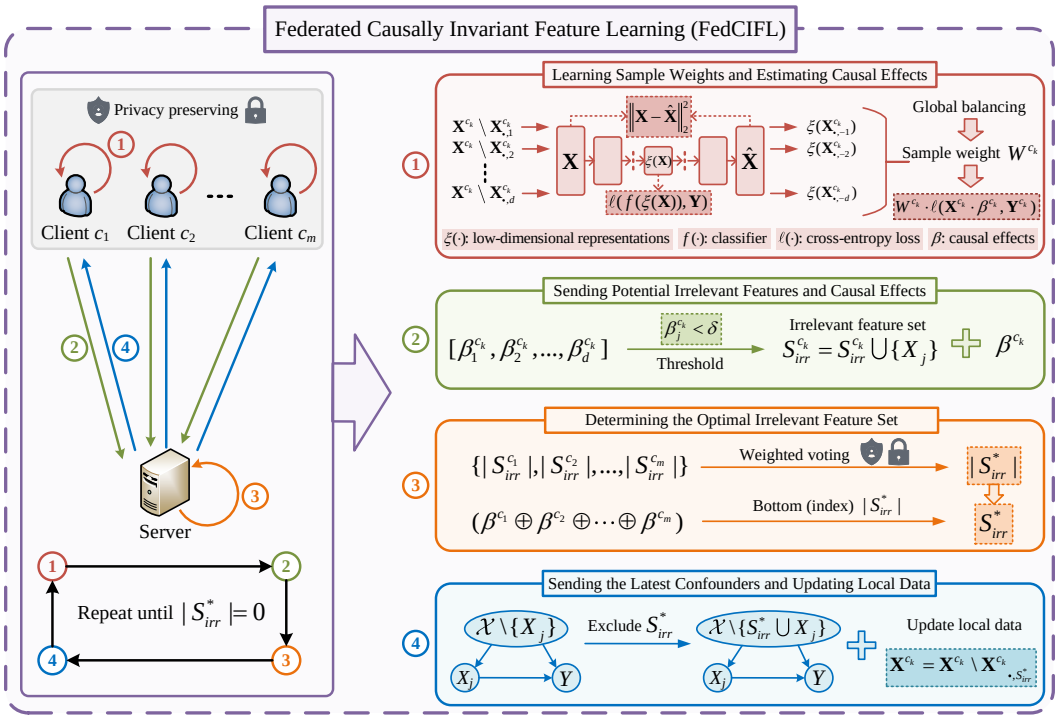

Figure 1: An overview of the proposed FedCIFL method.

is worth noting that PCFS and CIFD differ from previous causal feature selection methods in that they learn invariant causal feature subsets with stronger generalization ability by calculating causal effect values.

Despite the significant progress made by existing causal feature selection algorithms in single-source data scenarios, most of them have not considered the federated setting where data is distributed across multiple data holders. In fact, due to the constraints of data privacy and ownership, causal feature selection in FL scenarios remains an open and largely unexplored problem. Therefore, there is an urgent need to design novel federated causal feature selection methods that can fully utilize the decentralized data while protecting the privacy of each participating party, which is precisely the research motivation behind this paper.

C Detailed Pseudo-code for FedCIFL

The pseudo-code of the FedCIFL algorithm is detailed in Algorithm 1, and FedCIFL comprises the following four iterative steps:

- Step 1: Sample Weight Learning and Causal Effect Estimation **(Lines 3-7)**.
- Step 2: Transmission of Potentially Irrelevant Features and Causal Effects **(Lines 9-16)**.
- Step 3: Optimization of the Irrelevant Feature Set **(Lines 18-20)**.
- Step 4: Latest Confounder Transmission and Local Data Updates **(Lines 22-27)**.

$$\mathcal{L}^{c_k}_{sae} = \frac{1}{n_k}\left\|\mathbf{X}^{c_k} - \hat{\mathbf{X}}^{c_k}\right\|_2^2 + \lambda_1 \sum_{t=1}^{l}\sum_{a=1}^{2}\left(\left\|\mathbf{U}_a^{(t)}\right\|_2^2 + \left\|\mathbf{b}_a^{(t)}\right\|_2^2\right) + \lambda_2 \ell(f(\xi(\mathbf{X}^{c_k})), \mathbf{Y}^{c_k}). \quad (22)$$

Algorithm 1 Federated Causally Invariant Feature Learning (`FedCIFL`)

Require: $\{(\mathbf{X}^{c_k}, \mathbf{Y}^{c_k})\}_{k=1}^{m}$: m private labeled datasets held by m clients; δ: threshold.
Ensure: S_{ci}: the causally invariant feature subset.

repeat
  {*Step 1: Learning Sample Weights and Estimating Causal Effects*}
  for $k = 1$ to m **do**
    Learn the low-dimensional representation $\xi(\mathbf{X}^{c_k})$ and record $\mathcal{L}_{sae}^{c_k}$ based on Equation (22)
    Optimize the sample weight set $W^{c_k} = [W_1^{c_k}, W_2^{c_k}, \ldots, W_{n_k}^{c_k}]^T$ based on Equation (23)
    Estimate the causal effect vector $\beta^{c_k} = [\beta_1^{c_k}, \beta_2^{c_k}, \ldots, \beta_d^{c_k}]$ based on Equation (24)
  end for
  {*Step 2: Sending Potential Irrelevant Features and Causal Effects*}
  for $k = 1$ to m **do**
    $S_{irr}^{c_k} = \emptyset$
    for $j = 1$ to d **do**
      if $|\beta_j^{c_k}| < \delta$ **then**
        $S_{irr}^{c_k} = S_{irr}^{c_k} \cup \{X_j\}$
      end if
    end for
  end for
  {*Step 3: Determining the Optimal Irrelevant Feature Set*}
  $\Delta = \mathring{Rank}([\mathcal{L}_{sae}^{c_1}, \mathcal{L}_{sae}^{c_2}, \ldots, \mathcal{L}_{sae}^{c_m}])$, see Equation (25)
  $|S_{irr}^*| = \arg\min_{M_h \in \{M_1, M_2, \ldots\}} (\sum_{k=1}^{m} \Delta(k) \ \ subject \ \ to \ \ |S_{irr}^{c_k}| = M_h)$, see Equation (26)
  $S_{irr}^* = \mathring{Bottom}_{|S_{irr}^*|}(\beta^{c_1} \oplus \beta^{c_2} \oplus \cdots \oplus \beta^{c_m})$, see Equation (27)
  {*Step 4: Sending the Latest Confounders and Updating Local Data*}
  for $j = 1$ to d **do**
    $\mathcal{X} \setminus \{X_j\} \longrightarrow \mathcal{X} \setminus \{S_{irr}^* \cup X_j\}$ // Removing irrelevant features from the confounder set
  end for
  for $k = 1$ to m **do**
    $\mathbf{X}^{c_k} = \mathbf{X}^{c_k} \setminus \mathbf{X}_{\cdot, S_{irr}^*}^{c_k}$ // Updating each client's local data
  end for
until $|S_{irr}^*| = 0$
Let L be the number of iterations, and $S_{irr}^*(\psi)$ be the irrelevant feature set of the ψ-th iteration
$S_{ci} = \mathcal{X} \setminus \{S_{irr}^*(1) \cup S_{irr}^*(2) \cup \cdots \cup S_{irr}^*(L)\}$
return S_{ci}

$$\begin{aligned}\mathcal{L}_{sw2}^{c_k} = &\sum_{j=1}^{d} \left\| \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot \mathbf{X}_{\cdot,j}^{c_k})}{(W^{c_k})^T \cdot \mathbf{X}_{\cdot,j}^{c_k}} - \frac{\xi(\mathbf{X}_{\cdot,-j}^{c_k})^T \cdot (W^{c_k} \odot (1 - \mathbf{X}_{\cdot,j}^{c_k}))}{(W^{c_k})^T \cdot (1 - \mathbf{X}_{\cdot,j}^{c_k})} \right\|_2^2 \\ &+ \lambda_3 \left(\sum_{i=1}^{n_k} W_i^{c_k} - n_k \right)^2 + \lambda_4 \sum_{i=1}^{n_k} (W_i^{c_k} - 1)^2 .\end{aligned} \tag{23}$$

$$\begin{aligned}\mathcal{L}_{wce}^{c_k} = &- \sum_{i=1}^{n_k} W_i^{c_k} \cdot (y_i^{c_k} \cdot \log \frac{1}{1 + exp(-\mathbf{x}_i^{c_k} \cdot \beta^{c_k})} + \\ &(1 - y_i^{c_k}) \cdot \log(1 - \frac{1}{1 + exp(-\mathbf{x}_i^{c_k} \cdot \beta^{c_k})}) + \lambda_5 \|\beta^{c_k}\|_1,\end{aligned} \tag{24}$$

$$\Delta = \mathring{Rank}\left([\mathcal{L}_{sae}^{c_1}, \mathcal{L}_{sae}^{c_2}, \ldots, \mathcal{L}_{sae}^{c_m}]\right). \tag{25}$$

$$|S_{irr}^*| = \arg\min_{M_h \in \{M_1, M_2, \ldots\}} \left(\sum_{k=1}^{m} \Delta(k) \ \ subject \ \ to \ \ |S_{irr}^{c_k}| = M_h \right). \tag{26}$$

$$S_{irr}^* = \mathring{Bottom}_{|S_{irr}^*|} \left(\beta^{c_1} \oplus \beta^{c_2} \oplus \cdots \oplus \beta^{c_m}\right). \tag{27}$$

D Privacy and Cost Analysis

D.1 Privacy Preservation Capability of FedCIFL

As illustrated in Figure 1, in Step 1 of FedCIFL, each client independently trains a low-dimensional representation $\xi(\mathbf{X}^{c_k})$ using a supervised autoencoder, learns sample weights W^{c_k}, and evaluates the causal effect of each feature on the label using their local data. No raw data, whether feature data $\mathbf{X}^{c_k}$ or label data $\mathbf{Y}^{c_k}$, needs to be transmitted from the clients. In Step 2, the FL clients only send the minimum loss function values $[\mathcal{L}_{sae}^{c_1}, \mathcal{L}_{sae}^{c_2}, \ldots, \mathcal{L}_{sae}^{c_m}]$ obtained during the supervised autoencoder training and the causal effect vector $[\beta^{c_1}, \beta^{c_2}, \ldots, \beta^{c_m}]$ of each feature on the label to the server for aggregation. This process ensures that the server does not have access to any raw data, preserving the privacy of the client's sensitive information.

Steps 3 and 4 of FedCIFL also do not involve the exchange of any raw data between the clients and the server. In Step 3, the server determines the optimal irrelevant feature set based on the aggregated information received from the clients, without requiring access to the original data. Step 4 involves the server sending the latest confounder set to the clients and the clients updating their local data accordingly, without exposing any raw data to the server. Moreover, FedCIFL does not require the server to have knowledge of the sample size of each client, further enhancing privacy preservation. The weighted voting strategy employed in Step 3 allows the server to determine the optimal irrelevant feature set without needing to know the specific sample sizes of individual clients, reducing the risk of inferring sensitive information about the clients' data. By carefully controlling the information exchanged between the clients and the server and avoiding the transmission of raw data, FedCIFL provides a robust privacy-preserving solution for federated feature selection tasks.

To further mitigate the risk of inferring sensitive information about the client's local data during the federated learning process, FedCIFL can be combined with the following two techniques: (1) Additive homomorphic encryption (Paillier's scheme [22]) can be applied in the aggregation process of Equation (26) and Equation (27) to prevent leakage of sensitive information. By encrypting the individual values before aggregation, the server can perform the necessary computations on the encrypted data without gaining access to the actual values. This ensures that the server cannot infer any information about the client's local data, even if it has access to the aggregated results. (2) To prevent the leakage of semantic information about each feature during communication between FL clients and the server, we incorporate a feature masking strategy inspired by [23] into FedCIFL. Instead of directly sending feature names or semantic information, each client assigns unique identifiers to the features based on a predefined ordering scheme. The server instructs the clients to sort the features alphabetically and assign identifiers (e.g., "1", "2", "3", etc.) accordingly. In the case of ties, the clients consider the subsequent letters until a unique ordering is achieved. The clients then send only these assigned identifiers to the server for aggregation, effectively masking the semantic information of the original features.

D.2 Communication Cost of FedCIFL

Communication efficiency is a crucial consideration in FL as it can significantly impact the performance and practicality of FL systems. FedCIFL addresses this challenge by introducing a relatively low communication overhead compared to other FL methods. As shown in Figure 1, only Steps 2 and 4 of FedCIFL require communication with the server. For Step 2, each client only needs to send a one-dimensional vector $\beta^{c_k} = [\beta_1^{c_k}, \beta_2^{c_k}, \ldots, \beta_d^{c_k}]$ of length d to the server. In Step 4, the server only needs to send an irrelevant feature set S_{irr}^* with a maximum size of d to each client. Therefore, each iteration incurs a communication cost of $O(dm + dm)$.

Let L be the number of iterations in FedCIFL, and $S_{irr}^*(\psi)$ be the irrelevant feature set generated in Step 4 of the ψ-th iteration. Since $\sum_{\psi=1}^{L} |S_{irr}^*(\psi)| < d$ always holds, Step 2 generates a total communication cost of $O(mdL)$ over L iterations, while Step 4 incurs a total communication cost of $O(m * (|S_{irr}^*(1)| + |S_{irr}^*(2)| + \cdots + |S_{irr}^*(L)|))$=$O(m * \sum_{\psi=1}^{L} |S_{irr}^*(\psi)|)$=$O(md)$. Consequently, the overall communication cost of FedCIFL across L iterations is $O(mdL + md)$.

Compared to other FL methods that require the exchange of high-dimensional model parameters or gradients in each iteration, FedCIFL significantly reduces the communication overhead. By only transmitting the causal effect vector β^{c_k} and the irrelevant feature set S_{irr}^*, FedCIFL minimizes

the amount of data that needs to be communicated between the clients and the server. This makes FedCIFL particularly suitable for scenarios where communication bandwidth is limited or expensive.

It is also worth noting that the communication cost of FedCIFL is independent of the number of samples on each client. This is because the clients and the server only exchange the causal effect vector β^{c_k} and the irrelevant feature set S^*_{irr}, whose dimensions are determined by the number of features rather than the sample size. This property makes FedCIFL scalable to scenarios where each client holds a large number of samples.

E Implementation Details

All experiments were conducted on a computer equipped with an Intel Core i9-10900 3.70-GHz CPU, NVIDIA GeForce RTX 3060 GPU, and 64 GB memory. For all datasets, the values of λ_1, λ_2, λ_3, λ_4, λ_5, and ω are set to 0.0001, 10, 0.0001, 0.001, 0.01, and 6, respectively. The value of δ is set to 0.01 for the synthetic dataset and 0.001 for the Amazon Review dataset. The number of stacked layers is consistently set to 2 across all datasets. The significance level for conditional independence tests performed in EAMB-V3, EAMB-V5, CVS-V3, and CVS-V5 is set to 0.01.

EAMB-V3, EAMB-V5, CVS-V3, and CVS-V5 are implemented in MATLAB for selecting causal features, while PCFS-V3, PCFS-V5, Fed-FiS, FPSO-FS, and our FedCIFL are implemented in PYTHON. It is worth noting that for a fair evaluation of the selected features across all algorithms, we consistently train logistic regression (LR) and multilayer perceptron (MLP) classifiers in a federated setting using PYTHON code.

F Experimental Results Using a Logistic Regression (LR) Classifier

F.1 Experimental Results Using a LR Classifier on Synthetic Data

Based on the experimental results presented in Figure 2 and Figure 3, which depict the performance of various methods using the logistic regression (LR) classifier on synthetic datasets, we can make the following observations:

- In Figure 2, which represents the scenario where data is IID across clients but OOD for the test set, FedCIFL consistently achieves the best performance in terms of accuracy and F1 score across different numbers of clients and dataset dimensions ($d = 40$ and $d = 60$). Moreover, FedCIFL exhibits a stable performance as the number of clients increases, indicating its robustness to varying client numbers. In terms of RMSE, FedCIFL is among the top-performing methods, with its performance being comparable to or slightly better than the best-performing baselines.
- Existing federated feature selection methods, such as Fed-FiS and FPSO-FS, show suboptimal performance and higher fluctuations in the IID+OOD scenario. This can be attributed to their focus on capturing correlations between features and labels, which may not be sufficient to handle the distribution shift between the training and test sets. On the other hand, the causal feature selection methods (EAMB-V3, EAMB-V5, CVS-V3, CVS-V5, PCFS-V3, and PCFS-V5) demonstrate better performance than Fed-FiS and FPSO-FS, highlighting the importance of capturing causal relationships. However, their performance is still inferior to FedCIFL, possibly due to the lack of effective federated aggregation strategies, which may lead to the loss of some causally invariant features or the inclusion of irrelevant features.
- Moving on to Figure 3, which represents the more challenging Non-IID+OOD scenario, FedCIFL continues to outperform all baselines across different numbers of clients and dataset dimensions. The performance gap between FedCIFL and the baselines is more pronounced in this scenario compared to the IID+OOD setting. This observation suggests that FedCIFL's ability to capture causally invariant features is particularly beneficial in the presence of both data heterogeneity across clients and distribution shift between the training and test sets.
- The existing federated feature selection methods (Fed-FiS and FPSO-FS) and causal feature selection methods (EAMB-V3, EAMB-V5, CVS-V3, CVS-V5, PCFS-V3, and PCFS-V5)

exhibit larger performance fluctuations and a wider gap compared to FedCIFL in the Non-IID+OOD scenario. This further highlights the limitations of these methods in handling the combined challenges of data heterogeneity and distribution shift.

- It is worth noting that the performance of all other algorithms generally decreases to varying degrees as the number of clients increases. This is because, in our experimental design, the total number of samples remains fixed. As the number of clients increases, the sample size allocated to each client decreases, leading to more severe sample selection bias and less reliable results from statistical tests. However, FedCIFL consistently maintains its superiority and exhibits a more stable performance across different client numbers, demonstrating its robustness and effectiveness in learning causally invariant features.

In summary, the experimental results using the LR classifier on synthetic datasets demonstrate the superiority of FedCIFL in both IID+OOD and Non-IID+OOD scenarios. FedCIFL's ability to capture causally invariant features enables it to achieve better performance and exhibit greater stability compared to existing federated feature selection methods and causal feature selection methods, particularly in the presence of data heterogeneity and distribution shift.

F.2 Experimental Results Using a LR Classifier on Real-World Data

The experimental results on the real-world Amazon Review dataset using the logistic regression (LR) classifier, as presented in Table 1, provide further insights into the performance of FedCIFL and the baselines in learning causally invariant features for cross-domain generalization.

Across all four cross-domain tasks (DEK→B, BEK→D, BDK→E, and BDE→K), FedCIFL consistently achieves the highest accuracy metric, outperforming all other methods. This demonstrates the effectiveness of FedCIFL in capturing the underlying causal relationships between features and labels, which enables it to generalize well to different target domains. The superior performance of FedCIFL can be attributed to its sample reweighting strategy and iterative refinement of the confounder set, which help mitigate the impact of data heterogeneity and distribution shift.

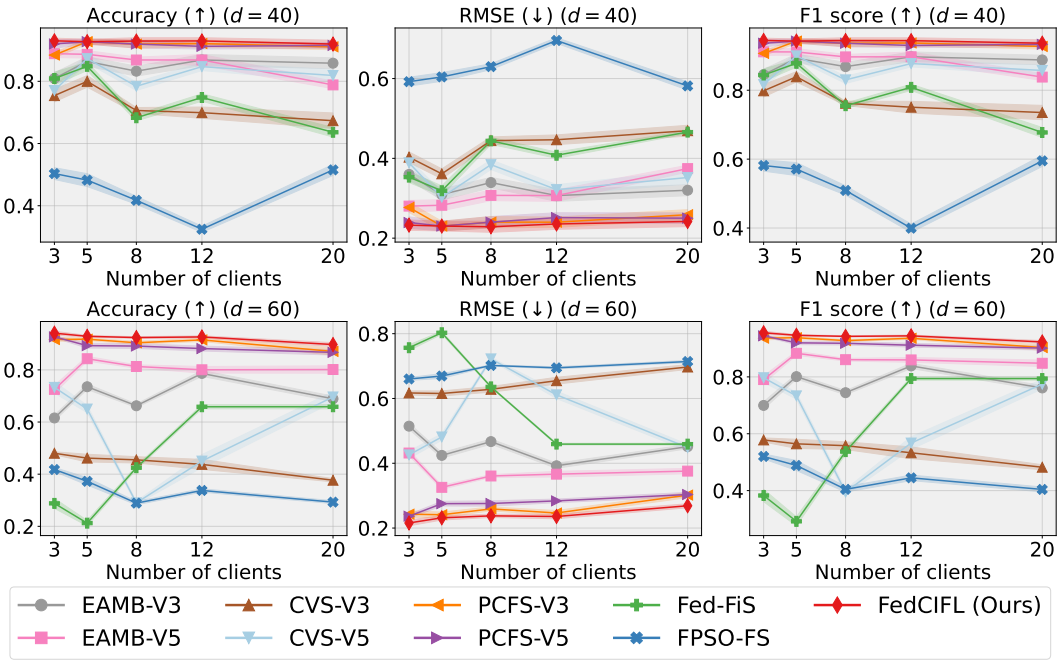

Figure 2: Experimental results on synthetic datasets where data is IID across clients but OOD for the test set. A total of 6,000 samples are *unevenly* distributed among {3, 5, 8, 12, 20} clients. The figure shows the performance of all methods in three metrics (Accuracy, RMSE, and F1 score from left to right) under two different dataset dimensions, $d = \{40, 60\}$, from top to bottom.

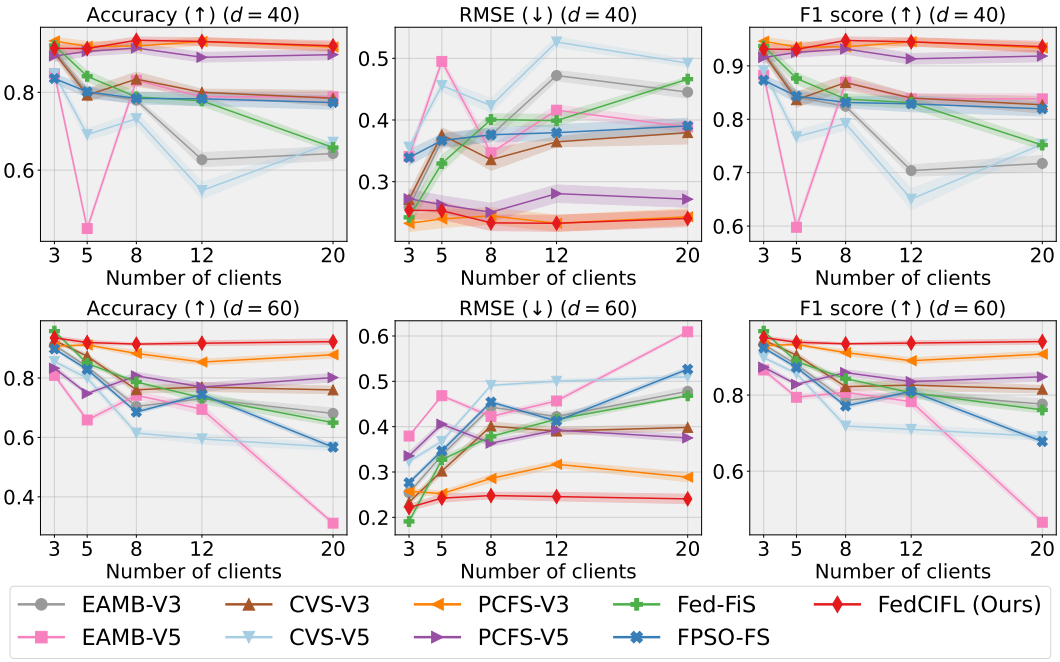

Figure 3: Experimental results on synthetic datasets where data is Non-IID across clients and OOD for the test set. A total of 6,000 samples are *unevenly* distributed among {3, 5, 8, 12, 20} clients. The figure shows the performance of all methods in three metrics (Accuracy, RMSE, and F1 score from left to right) under two different dataset dimensions, $d = \{40, 60\}$, from top to bottom.

In terms of RMSE metric, `FedCIFL` exhibits the lowest values on most cross-domain tasks, indicating its ability to make more accurate predictions compared to the baselines. The lower RMSE values of `FedCIFL` suggest that it is better able to capture the true causal relationships between features and labels, leading to more precise predictions in the target domains.

The performance of EAMB-V3 is generally better than that of the federated feature selection methods (i.e., Fed-FiS and FPSO-FS). This highlights the importance of considering causal relationships in feature selection, especially in the presence of domain shift. However, the EAMB-V3 method still lags behind `FedCIFL`, possibly due to their lack of effective federated aggregation strategies and the inability to fully mitigate the impact of data heterogeneity and distribution shift.

It is worth noting that the performance of all methods varies across different cross-domain tasks, indicating the varying levels of difficulty in adapting to different target domains. For example, the BDE→K task appears to be relatively easier, with most methods achieving higher accuracy and F1 scores compared to other tasks. On the other hand, the DEK→B task seems to be more challenging,

Table 1: Accuracy (%), RMSE, and F1 score (%) of the 4 cross-domain tasks on the Amazon Review dataset based on the logistic regression (LR) classifier.

Metrics	Tasks	EAMB-V3	EAMB-V5	CVS-V3	CVS-V5	PCFS-V3	PCFS-V5	Fed-FiS	FPSO-FS	FedCIFL (Ours)
Accuracy (↑)	DEK→B	71.80±1.10	67.40±1.97	66.20±1.61	62.60±2.00	69.10±2.85	65.50±1.94	73.80±2.47	71.55±1.84	**74.85±1.22**
	BEK→D	76.14±0.66	69.97±1.48	70.83±1.90	61.10±1.04	71.68±1.09	66.62±1.52	75.94±2.02	78.50±1.55	**81.10±3.01**
	BDK→E	80.70±2.71	72.48±1.98	77.09±1.84	58.80±2.51	74.54±3.71	68.42±2.78	79.75±2.05	78.55±2.64	**83.21±2.80**
	BDE→K	80.85±1.57	72.38±2.03	79.00±1.84	59.90±1.87	78.05±2.18	68.37±1.99	81.25±2.95	83.01±2.72	**84.11±1.78**
RMSE (↓)	DEK→B	**0.427±0.00**	0.458±0.01	0.466±0.01	0.482±0.00	0.446±0.02	0.462±0.01	0.436±0.01	0.454±0.01	0.435±0.01
	BEK→D	0.405±0.00	0.442±0.01	0.443±0.01	0.485±0.01	0.435±0.01	0.461±0.01	0.419±0.02	0.391±0.01	**0.377±0.02**
	BDK→E	0.369±0.02	0.425±0.01	0.393±0.02	0.489±0.00	0.417±0.02	0.457±0.01	0.374±0.01	0.379±0.02	**0.347±0.02**
	BDE→K	0.365±0.01	0.426±0.00	0.382±0.01	0.476±0.00	0.389±0.01	0.453±0.01	0.362±0.03	0.349±0.02	**0.335±0.01**
F1 (↑)	DEK→B	**73.39±1.22**	69.49±1.24	64.67±1.50	56.86±3.27	69.52±2.75	61.63±2.84	70.96±2.79	68.12±2.76	72.00±2.15
	BEK→D	75.95±0.54	70.19±1.25	69.74±2.78	53.85±3.27	69.98±1.04	64.17±1.96	76.85±1.94	78.61±1.59	**81.05±2.81**
	BDK→E	80.74±2.90	73.30±2.46	76.49±1.95	48.85±3.45	73.63±4.26	64.60±3.68	78.42±2.44	78.08±2.80	**82.56±3.05**
	BDE→K	81.57±1.68	73.74±1.92	80.44±1.21	62.07±1.82	78.47±1.64	67.58±1.55	80.50±3.44	83.62±2.26	**84.71±1.50**

with lower overall performance across all methods. Despite these variations, FedCIFL consistently maintains its superiority, demonstrating its robustness and adaptability to different domain adaptation scenarios.

In summary, the experimental results on the Amazon Review dataset using the LR classifier provide further evidence of the superiority of FedCIFL in learning causally invariant features for improved cross-domain generalization. FedCIFL's ability to effectively capture causal relationships and mitigate the impact of data heterogeneity and distribution shift enables it to outperform state-of-the-art baselines on most cross-domain tasks. The results also highlight the importance of considering causal relationships in the FFS problem and the challenges associated with adapting to different target domains in real-world applications.

F.3 Experimental Results Using a LR Classifier for Ablation Study

Based on the ablation study results presented in Figure 4, which depicts the performance of FedCIFL and its three variant algorithms using the logistic regression (LR) classifier, we can make the following observations:

- Across different numbers of clients and data dimensions ($d = 40$ and $d = 60$), FedCIFL consistently outperforms the three variant algorithms ("FedCIFL w/o iter", "FedCIFL w/o SAE", and "FedCIFL w/o weighting") in terms of accuracy and F1 score. This finding reinforces the effectiveness and necessity of each key module in our proposed FedCIFL method for the FFS task, even when using a simpler classifier like LR.
- The performance gap between FedCIFL and "FedCIFL w/o iter" highlights the importance of iteratively refining the confounder set. By optimizing the confounder set through multiple iterations, FedCIFL can better identify true confounders and mitigate the impact of limited local data on the accuracy of federated causal effect estimation, leading to improved performance.
- FedCIFL's superiority over "FedCIFL w/o SAE" demonstrates the benefit of employing a supervised autoencoder to learn a low-dimensional representation space. By capturing nonlinear relationships among features and enhancing the robustness of the learned sample weights, the supervised autoencoder contributes to FedCIFL's better performance in balancing the sample distribution between the treatment and control groups.

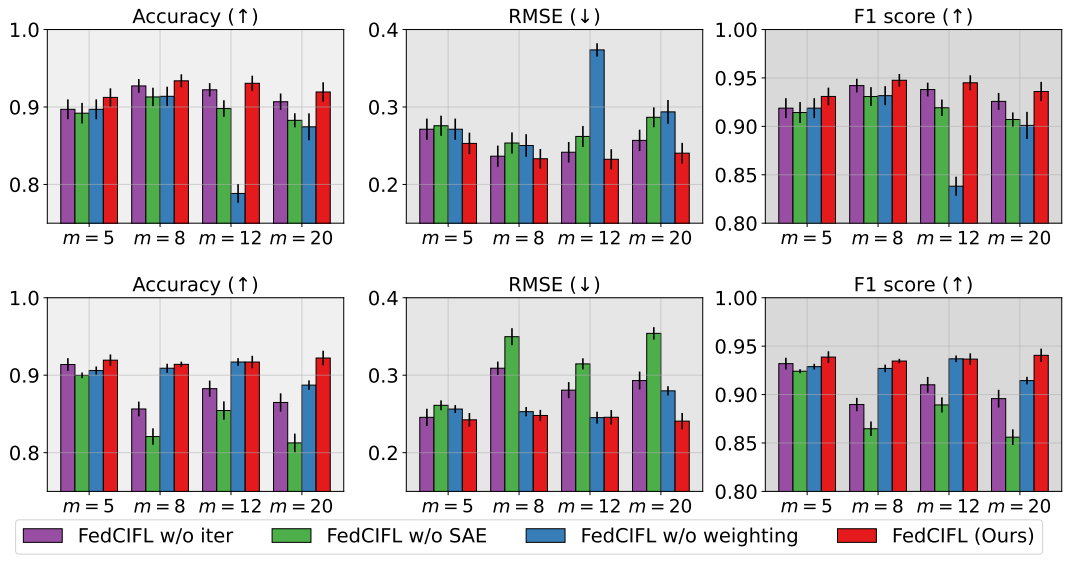

Figure 4: Results of ablation experiments using a logistic regression (LR) classifier. The figure shows the performance of all methods in three metrics (Accuracy, RMSE, and F1 score from left to right) under two different dataset dimensions, $d = \{40, 60\}$, from top to bottom.

- The advantage of FedCIFL over "FedCIFL w/o weighting" emphasizes the significance of the highly privacy-preserving weighted voting strategy. By effectively resolving conflicts arising from the presence of multiple modes and accurately determining the optimal irrelevant feature set size without requiring knowledge of individual clients' sample sizes, the weighted voting strategy enables FedCIFL to achieve better performance.
- It is worth noting that as the number of clients varies, the performance of all other methods exhibits significant fluctuations. In contrast, our FedCIFL maintains a consistently stable performance across different client numbers, demonstrating its robustness and effectiveness in learning causally invariant features, even when the local sample size on each client is limited.
- Comparing the results of Figure 4 (using the LR classifier) with those of Figure 5 in the main text (using the MLP classifier), we can observe that the performance of all methods is generally lower when using the LR classifier. This suggests that the MLP classifier, with its ability to capture more complex and nonlinear relationships, is better suited for the Non-IID+OOD FFS setting. However, the relative performance of FedCIFL compared to the variant algorithms remains consistent across both classifiers, highlighting the effectiveness of each key module in our proposed method.

In summary, the ablation study results using the LR classifier, as presented in Figure 4, further validate the effectiveness and necessity of each key module in our proposed FedCIFL method for the FFS task. The iterative optimization of the confounder set, the use of a supervised autoencoder for learning a low-dimensional representation space, and the highly privacy-preserving weighted voting strategy all contribute to FedCIFL's superior performance in capturing causally invariant features and achieving improved generalization ability, even when using a simpler classifier like LR.

G Statistical Tests

In this section, we adopt the Friedman test and Nemenyi test [24] to verify whether FedCIFL is significantly better than other methods on the real-world Amazon Review dataset.

We first perform the Friedman test [24] 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 rankings of all algorithms are equivalent). For real-world datasets, the average rankings of FedCIFL and the baselines when using different metrics are summarized in Table 2. Table 2 shows that no matter which classifier is used, the null hypothesis is rejected on these three metrics (i.e. Accuracy, RMSE and F1 score). We also note that FedCIFL always performs better than the baselines on all metrics. (In Table 2, the lower ranking value is better.)

To further analyze the significant difference between FedCIFL and the baselines, we perform the Nemenyi test [24], which states that the performance levels of two algorithms are significantly different if the corresponding average rankings differ by at least one critical difference (CD). The CD for the Nemenyi test is calculated as follows (i.e., Eq. (28)).

$$\mathrm{CD} = q_{\alpha,\theta}\sqrt{\frac{\theta(\theta+1)}{6\eta}}, \tag{28}$$

where α is the significance level, θ is the number of comparison algorithms, and η denotes the number of datasets with different numbers of clients. In our experiments, $\theta = 9$, $q_{\alpha=0.05,\theta=9} = 3.102$ at significance level $\alpha = 0.05$. When using the real-world Amazon Review dataset, $\eta = 4 * 2 = 8$ (four cross-domain tasks with two classifiers), and thus $\mathrm{CD} = 4.25$.

Table 2: The average rankings of FedCIFL and the baselines on the real-world Amazon Review dataset using Accuracy, RMSE and F1 score metrics.

Algorithm		EAMB-V3	EAMB-V5	CVS-V3	CVS-V5	PCFS-V3	PCFS-V5	Fed-FiS	FPSO-FS	FedCIFL (Ours)
Avg rank	Accuracy	3	6.25	6.13	9	5.5	7.88	2.88	3.25	**1.13**
	RMSE	2.25	5.38	7	8.5	5.38	6.88	4	4.13	**1.5**
	F1 score	2.63	5.5	6.25	8.88	5.5	7.88	3.25	3.88	**1.25**

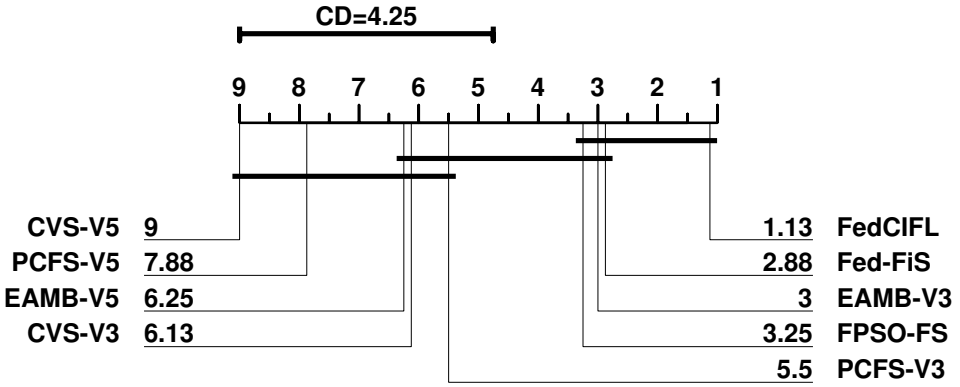

Figure 5: Crucial difference diagram of the Nemenyi test for Accuracy metric on the real-world Amazon Review dataset.

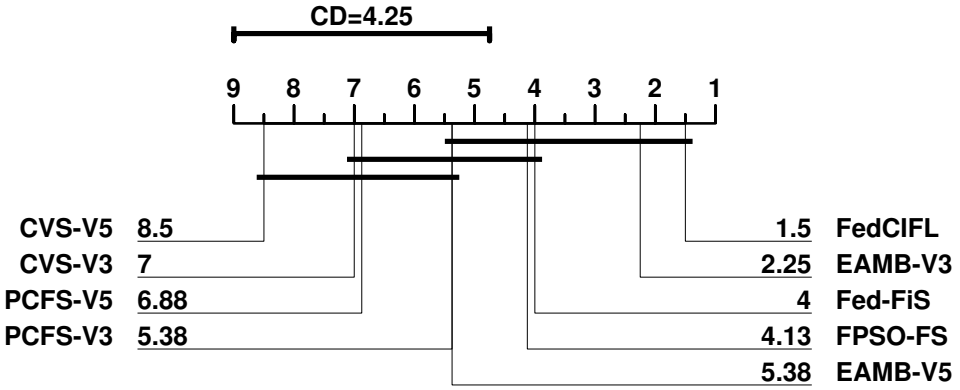

Figure 6: Crucial difference diagram of the Nemenyi test for RMSE metric on the real-world Amazon Review dataset.

Figs. 5-7 provide the CD diagrams on three different metrics, respectively. In each CD diagram, the average ranking of each algorithm is marked along the axis (lower rankings to the right). When using the Accuracy metric, we observe that `FedCIFL` significantly outperforms PCFS-V3, CVS-V3, EAMB-V5, PCFS-V5 and CVS-V5, and `FedCIFL` achieves a comparable performance against Fed-FiS, EAMB-V3 and FPSO-FS. When using the RMSE metric, we observe that `FedCIFL` achieves a comparable performance against EAMB-V3, Fed-FiS, FPSO-FS, EAMB-V5 and PCFS-V3, and `FedCIFL` significantly outperforms the other baselines. When using the F1 score metric, we observe that `FedCIFL` achieves a comparable performance against EAMB-V3, Fed-FiS, FPSO-FS, and `FedCIFL` significantly outperforms the other baselines. Additionally, on all metrics, `FedCIFL` is the only algorithm that achieves the lowest ranking value.

References

[1] Yang, Q., Y. Liu, T. Chen, et al. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[2] Banerjee, S., E. Elmroth, M. Bhuyan. Fed-fis: A novel information-theoretic federated feature selection for learning stability. In *International Conference on Neural Information Processing*, pages 480–487. Springer, 2021.

[3] Banerjee, S., D. Bhuyan, E. Elmroth, et al. Cost-efficient feature selection for horizontal federated learning. *IEEE Transactions on Artificial Intelligence*, 2024.

[4] Hermo, J., V. Bolón-Canedo, S. Ladra. Fed-mrmr: A lossless federated feature selection method. *Information Sciences*, 669:120609, 2024.

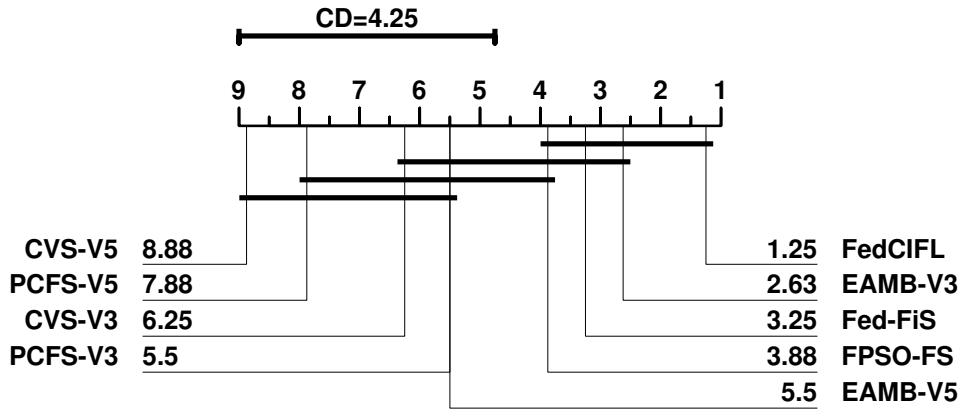

Figure 7: Crucial difference diagram of the Nemenyi test for F1 score metric on the real-world Amazon Review dataset.

[5] Zhang, X., A. Mavromatis, A. Vafeas, et al. Federated feature selection for horizontal federated learning in IoT networks. *IEEE Internet of Things Journal*, 10(11):10095–10112, 2023.

[6] Hu, Y., Y. Zhang, D. Gong, et al. Multi-participant federated feature selection algorithm with particle swarm optimization for imbalanced data under privacy protection. *IEEE Transactions on Artificial Intelligence*, 2022.

[7] Cassará, P., A. Gotta, L. Valerio. Federated feature selection for cyber-physical systems of systems. *IEEE Transactions on Vehicular Technology*, 71(9):9937–9950, 2022.

[8] Hu, Y., Y. Zhang, X. Gao, et al. A federated feature selection algorithm based on particle swarm optimization under privacy protection. *Knowledge-Based Systems*, 260:110122, 2023.

[9] Yu, K., X. Guo, L. Liu, et al. Causality-based feature selection: Methods and evaluations. *ACM Computing Surveys (CSUR)*, 53(5):1–36, 2020.

[10] Aliferis, C. F., A. Statnikov, I. Tsamardinos, et al. 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), 2010.

[11] Tsamardinos, I., C. F. Aliferis, A. R. Statnikov, et al. Algorithms for large scale Markov blanket discovery. In *FLAIRS*, vol. 2, pages 376–81. 2003.

[12] Liu, X., X. Liu. Swamping and masking in Markov boundary discovery. *Machine Learning*, 104(1):25–54, 2016.

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

[14] Guo, X., K. Yu, F. Cao, et al. Error-aware Markov blanket learning for causal feature selection. *Information Sciences*, 589:849–877, 2022.

[15] Tsamardinos, I., C. F. Aliferis, A. Statnikov. Time and sample efficient discovery of Markov blankets and direct causal relations. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 673–678. 2003.

[16] Wu, X., B. Jiang, K. Yu, et al. Accurate Markov boundary discovery for causal feature selection. *IEEE Transactions on Cybernetics*, 50(12):4983–4996, 2019.

[17] Guo, X., K. Yu, L. Liu, et al. Causal feature selection with dual correction. *IEEE Transactions on Neural Networks and Learning Systems*, 35(1):938–951, 2022.

[18] Ling, Z., Y. Li, Y. Zhang, et al. A light causal feature selection approach to high-dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7639–7650, 2023.

[19] Kuang, K., H. Wang, Y. Liu, et al. Stable prediction with leveraging seed variable. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):6392–6404, 2023.

[20] Yang, S., X. Guo, K. Yu, et al. Causal feature selection in the presence of sample selection bias. *ACM Transactions on Intelligent Systems and Technology*, 14(5):1–18, 2023.

[21] Wang, Y., K. Yu, G. Xiang, et al. Discovering causally invariant features for out-of-distribution generalization. *Pattern Recognition*, page 110338, 2024.

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

[23] Huang, J., X. Guo, K. Yu, et al. Towards privacy-aware causal structure learning in federated setting. *IEEE Transactions on Big Data*, 9(6):1525–1535, 2023.

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