Vertical Federated Feature Screening

Huajun Yin[1,2], **Liyuan Wang**[1,2†], **Yingqiu Zhu**[3], **Liping Zhu**[4], **Danyang Huang**[1,2†]

[1]Center for Applied Statistics and School of Statistics,
Renmin University of China, Beijing, China
[2]Bigdata and Responsible Artificial Intelligence for National Governance,
Renmin University of China, Beijing, China
[3]School of Statistics, University of International Business and Economics, Beijing, China
[4]Institute of Statistics and Big Data, Renmin University of China, Beijing, China

Abstract

With the rapid development of the big data era, Vertical Federated Learning (VFL) has been widely applied to enable data collaboration while ensuring privacy protection. However, the ultrahigh dimensionality of features and the sparse data structures inherent in large-scale datasets introduce significant computational complexity. In this paper, we propose the Vertical Federated Feature Screening (VFS) algorithm, which effectively reduces computational, communication, and encryption costs. VFS is a two-stage feature screening procedure that proceeds from coarse to fine: the first stage quickly filters out irrelevant feature groups, followed by a more refined screening of individual features. It significantly reduces the resource demands of downstream tasks such as secure joint modeling or federated feature selection. This efficiency is particularly beneficial in scenarios with ultrahigh feature dimensionality or severe class imbalance in the response variable. The statistical and computational properties of VFS are rigorously established. Numerical simulations and real-world applications demonstrate its superior performance.

1 Introduction

In the era of big data, the exponential growth of data and concerns over privacy have highlighted the need for Federated Learning (FL). Among its variants, Vertical Federated Learning (VFL) enables collaborative model training using the shared information from overlapping users, without exchanging raw data, thereby ensuring privacy and security [11, 38, 49, 74, 76]. A typical example, as shown in Figure 1, occurs when an e-commerce company and a bank collaborate to train a model predicting a mutual interest, such as individual credit score. To collaboratively train a predictive model, institutions need to utilize secure multi-party computation for data transmission and computation [3, 40, 53, 67]. Following existing literature, the party that possesses the labels (e.g., credit defaults) is referred to as the *active party*, while the other parties are designated as *passive parties* [27, 53, 73].

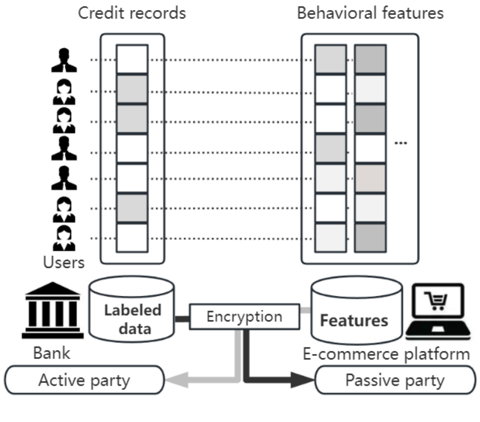


Figure 1: An example of VFL.

[†]Correspondence to: Liyuan Wang <wangly2023@ruc.edu.cn> and Danyang Huang <dyhuang@ruc.edu.cn>.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

However, the increasing feature dimension presents significant challenges for VFL analysis. Numerous real-world cases and theoretical studies indicate that, in ultrahigh-dimensional data, only a small proportion of features are truly useful for prediction [23, 56, 80]. To address this, researchers have proposed several feature selection methods in VFL to reduce the cost of building joint predictive models [7, 37, 45]. However, due to iterative parameter estimation and information communication across multiple parties, methods based on joint modeling using all features still incurs high costs when the feature dimension (p) and sample size (n) are ultrahigh. Therefore, improving the efficiency of these feature selection or joint modeling methods remains a crucial issue in VFL.

Another challenging issue in real-world data is the rare event problem, which represents an extreme case of class imbalance. Following prior literature, observations in the minority class are referred to as *cases*, while those in the majority class are called *controls* [30, 44]. Although controls often dominate the dataset, they typically contribute limited information gain compared to cases [39, 55]. Moreover, the rare event setting implies that cases are extremely scarce, which may invalidate the statistical properties of standard model estimates [9, 17, 33, 64, 70, 71]. Therefore, when improving existing federated learning methods, the rare event problem should also be taken into account.

This study aims to address the challenges of feature selection and joint modeling in VFL, considering both the ultrahigh dimensionality with large p and the occurrence of rare event with large n. We propose a subsampling-based model-free screening statistic, along with a novel two-stage VFL screening algorithm, called the **Vertical Federated Feature Screening** (VFS) algorithm. In the first stage, the ultrahigh-dimensional features are partitioned into multiple groups, irrelevant groups of features are deleted based on VFS statistics. In the second stage, we conduct a more precise screening process for the remaining features. Once fine-grained screening results are obtained, the downstream federated learning techniques can be applied to the condensed dataset with significantly reduced computational complexity. We have established the statistical properties of the VFS algorithm to ensure its reliability. Extensive experiments using simulated and real-world datasets show that our method achieves promising performance with low costs of encryption, decryption, computation, and communication in VFL applications. The main contributions of this paper are threefold.

Unified model-free screening statistic. We propose the VFS statistic which allows for rare events. The VFS statistic is a general framework, compatible with various commonly used statistics but newly designed for rare event scenarios and capable of detecting complex nonlinear relationships.

Coarse-to-fine screening algorithm. We propose a two-stage screening algorithm, which first quickly removes a large number of irrelevant features and then performs a more refined screening. This approach balances the performance and efficiency of the screening process, further reduce the cost of encryption, decryption, computation, and communication in a VFL system.

Theoretical guarantee. We establish the statistical properties of the proposed screening statistics and algorithm, providing a rigorous theoretical foundation in VFL scenarios allowing for rare event and ultrahigh-dimensional features.

2 Related works

Federated feature selection. Existing literature has investigated how to identify important features during the collaborative training of models in a VFL system [31, 47]. For instance, Feng [24] proposed a VFL-based feature selection method that uses deep learning models and complementary information from different parties. Castiglia et al. [7] introduced LESS-VFL, which optimizes communication efficiency by using a short pre-training period and local feature selection techniques. Li et al. [45] presented FedSDG-FS, a secure feature selection method that uses a Gaussian stochastic dual-gate and partially homomorphic encryption. Ji et al. [37] proposed MUSE, a vertical federated feature selection framework based on mutual information. However, when the feature dimension is ultrahigh (i.e., p grows exponentially with n), existing VFL methods incur high communication and encryption costs, highlighting the need for an efficient preprocessing step to reduce dimensionality.

Non-FL feature screening and independence test. Unlike feature selection which precisely identifies true features, the primary goal of feature screening is to address ultrahigh-dimensional data challenges by eliminating irrelevant features through the marginal correlation [15, 22]. In particular, model-free screening methods aim to identify irrelevant variables without relying on specific model assumptions [12, 15, 51, 65, 72, 78]. One effective strategy for model-free screening involves using statistics derived from independence tests, which focus on the asymptotic distribution of test statistics [28, 61, 75, 77, 79]. In the theoretical analysis of such statistics, generalized U-statistics provide a classical and powerful tool [43, 62]. However, there is limited discussion of related methods in VFL settings. In VFL with rare events, direct application of existing methods incurs high computational costs and invalidates asymptotic properties due to the extreme sparsity of cases. This calls for more efficient screening methods and revised theoretical foundations.

Imbalanced data. Undersampling controls [19, 52] and oversampling cases [9, 18] are commonly used approaches for handling imbalanced data. Of the two approaches, undersampling can enhance model performance and reliability without further increasing computational costs [63]. Through undersampling, the statistical properties of methods such as logistic regression [26, 70, 71], decision trees [50], support vector machines [2], and high-dimensional linear discriminant analysis [57] have been re-examined, leading to refined theoretical results. Nevertheless, rare event and its impact on general feature screening statistics are seldom considered in the existing VFL literature.

Cost in VFL. Existing studies have examined various costs in VFL. To ensure data privacy, encryption and decryption procedures introduce substantial resource consumption [10, 54]. Communication cost arises from the transmission of encrypted data and model parameters [7, 8]. Computational cost is dominated by local model training on encrypted data [5, 53]. There is a need for a feature screening approach designed for VFL with minimum costs of encryption, decryption, computation, and communication, which significantly accelerates downstream tasks.

3 Methodology

3.1 Notations and background

In VFL, multiple parties collaboratively build a model using a sample dataset $\mathcal{D} = \{(\mathbf{X}_i, Y_i)\}_{i=1}^n$, where $\mathbf{X}_i = (X_{1,i}, \ldots, X_{p,i})$ is a p-dimensional feature vector. Let $Y = (Y_1, \ldots, Y_n) \in \mathbb{R}^n$ denote the series of labels, and $\mathbf{X} \in \mathbb{R}^{n \times p}$ represents n observations of p features. The observations of the j-th feature are contained in the j-th column of $\mathbf{X}$, denoted as X_j. For simplicity, we assume a two-party VFL setting with a binary response. The extension to multi-party or multi-class scenarios is straightforward and will be discussed later. The active party owns the label column Y, where each Y_i takes a value between 0 and 1, while the passive party owns the p features, represented by the p columns $X_1, \ldots, X_p$. We next present the problem settings, which reflect the two key advantages of our method.

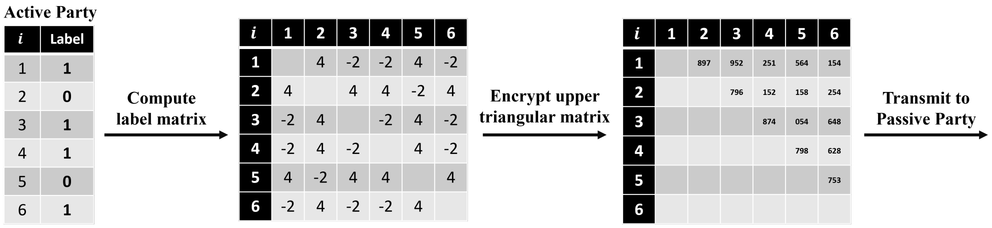
The first one is that we allow for extreme imbalance (i.e., rare event) in the response variable. Following previous literature, we refer to observations with $Y_i = 1$ as *cases* with sample size n_1, those with $Y_i = 0$ as *controls* with sample size n_0, and the total sample size is $n = n_0 + n_1$. Rare events imply that n_1 grows much slower than n, resulting in $n_1/n \to 0$. Such settings frequently arise in collaborative modeling scenarios, including fraud detection [41] and disease prediction [14].

The second advantage is that we allow ultrahigh-dimensional features, where the number of features p can grow at an exponential rate relative to n_1, e.g., $\log p = O(n_1^{\xi})$ with $\xi \in (0, 1)$. For real-world ultrahigh-dimensional datasets, among the p features, only a few, say m, are truly informative [23, 56, 80]. Thus, a screening procedure can dramatically reduce these costs by identifying a feature set $\hat{\mathcal{F}}$ to estimate the true feature set $\mathcal{F}$, thereby reducing the costs of subsequent downstream tasks.

However, in a VFL system, encrypting Y directly would make many model-free statistics infeasible to compute, particularly rank-based statistics. This limitation arises because simple and fast homomorphic encryption schemes, including RSA [59], Paillier [58], and CKKS [13], do not support direct comparisons of magnitude. To this end, we propose a *paired encryption procedure* for feature screening. Specifically, the active party first generates a label matrix $\mathbf{Y} = (\mathbf{Y}_{i_1,i_2})_{n \times n}$ from Y, where $\mathbf{Y}_{i_1,i_2}$ indicates whether the i_1-th and i_2-th observations belong to the same class. The matrix $\mathbf{Y}$ is then encrypted as $[\mathbf{Y}]$ and transmitted to the passive party, where $[c]$ denotes the ciphertext of c. Using $[\mathbf{Y}]$ and its features $\mathbf{X}$, the passive party computes the encrypted screening statistic $[\tilde{T}]$ and returns it to the active party. Finally, the active party decrypts $[\tilde{T}]$ to obtain $\tilde{T}$ for further analysis. For example, in the case of the distance covariance used in our subsequent implementation and CKKS encryption method, we denote

$$[\mathbf{Y}_{i_1,i_2}] = \begin{cases} [4], & Y_{i_1} \neq Y_{i_2} \\ [-2], & Y_{i_1} = Y_{i_2} \end{cases}.$$

It is worth noting that the same plaintext correspond to different ciphertexts, and thus the ciphertexts are no longer binary and cannot be easily decrypted. The encrypted statistic $[\tilde{T}_j]$ for the j-th feature is computed as $[\tilde{T}_j] = (n_0 n_1)^{-1} \sum_{i_1} \sum_{i_2} [\mathbf{Y}_{i_1,i_2}] \|X_{j,i_1} - X_{j,i_2}\|_2$ where X_{j,i_1} denotes the j-th feature of the i_1-th user. An illustration of this process is provided in Figure 2.



i	Label
1	1
2	0
3	1
4	1
5	0
6	1

i	1	2	3	4	5	6
1		4	-2	-2	4	-2
2	4		4	4	-2	4
3	-2	4		-2	4	-2
4	-2	4	-2		4	-2
5	4	-2	4	4		4
6	-2	4	-2	-2	4	

i	1	2	3	4	5	6
1		897	952	251	564	154
2			796	152	158	254
3				874	054	648
4					798	628
5						753
6						

Figure 2: Paired encryption procedure.

Remark 1 (Effect to Privacy-Preserving Mechanisms). VFS is compatible with any suitable homomorphic encryption (HE) scheme. While our focus is on the statistical and computational aspects of feature screening in VFL, for reference, the privacy guarantees of HE are well established by Gentry [29]. Although certain inference attacks may exist, more advanced HE variants can provide stronger protection [4, 21], and combining HE with differential privacy (DP) offers an additional layer of privacy assurance [46, 68].

3.2 Vertical Federated Feature Screening algorithm

To address the challenges posed by the ultrahigh-dimensional and extremely imbalanced data, we propose the Vertical Federated Feature Screening (VFS) and shown in Algorithm 1. Guided by the sparsity of true features, we adopt a *coarse-to-fine* strategy to identify useful features through a two-stage process. The overall procedure is illustrated in Figure 3.

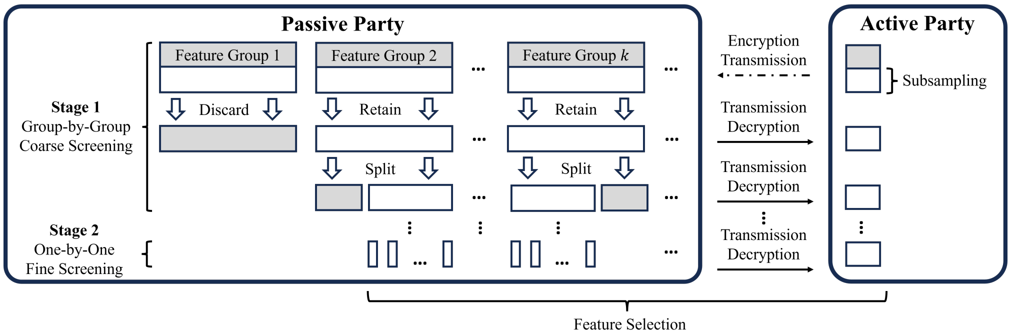


Figure 3: Framework of Vertical Federated Feature Screening (VFS).

Preliminary Step (Subsampling): Generate binary indicator variable η_i ($1 \leqslant i \leqslant n_0$) from Bernoulli($sn_1/n_0$), indicating whether the i-th control is selected, where s is a predefined constant representing the subsample ratio.

Stage 1 (Group-by-Group Coarse Screening): Features are partitioned into groups, and a coarse screening process rapidly eliminates irrelevant groups, significantly reducing dimensionality. Specifically, for each group $\mathbf{X}_k$, the screening statistic $\tilde{\mathbf{T}}_k$ is computed and hypothesis testing is performed. If the null hypothesis is rejected, the feature group $\mathbf{X}_k$ is considered to contain true features and is retained; otherwise, it is discarded. The coarse screening is repeatedly performed until the number of remaining features becomes computationally acceptable.

Algorithm 1 Vertical Federated Feature Screening (VFS)

```
Input: Feature matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$, label series $Y \in \mathbb{R}^n$, subsample ratio $s$, initial group size $s_0$, decay coefficient $\rho$, # features of stage switching $p'$, screening threshold $\delta$.
Output: Retained feature matrix $\mathbf{X}' \in \mathbb{R}^{n\times d}$.
for $i = 1$ to $n_0$ do
    Sample the $i$-th control with probability $sn_1/n_0$.
Active party: Generate $\mathbf{Y}$, encrypt $\mathbf{Y}$ into $[\mathbf{Y}]$, send $[\mathbf{Y}]$ to passive party.
$p_0 \leftarrow p$, $r \leftarrow 1$
while $p_{r-1} > p'$ do
    Passive party: Divide $p_{r-1}$ features into $g_r = \lceil p_{r-1}/s_{r-1} \rceil$ groups.
    for $k = 1$ to $g_r$ do
        Passive party: Compute $[\tilde{\mathbf{T}}_k]$ for $k$-th group of features, send $[\tilde{\mathbf{T}}_k]$ to active party.
        Active party: Decrypt $[\tilde{\mathbf{T}}_k]$ into $\tilde{\mathbf{T}}_k$, perform hypothesis test to determine whether to discard $k$-th group of features.
    $p_r \leftarrow$ # retained features, $s_r \leftarrow \rho s_{r-1}$, $r \leftarrow r + 1$.
$p' \leftarrow p_{r-1}$
for $j = 1$ to $p'$ do
    Passive party: Compute $[\tilde{T}_j]$ for $j$-th feature, send $[\tilde{T}_j]$ to active party.
    Active party: Decrypt $[\tilde{T}_j]$ into $\tilde{T}_j$, use threshold $\delta$ to determine whether to discard $j$-th feature.
```

Stage 2 (One-by-One Fine Screening): The remaining features are subjected to a refined screening process, ensuring both efficiency and precision. The screening statistic $\tilde{T}_j$ is computed for each feature X_j. If $\tilde{T}_j$ exceeds the threshold δ, the feature X_j is retained; otherwise, it is discarded. This stage enables the accurate identification of true features. For convenience, let d denote the number of features retained.

Compared to classical methods that rely solely on fine screening, coarse screening substantially lowers the time required for encryption, decryption, computation, and communication, further enhancing the overall efficiency of the VFS algorithm. The comparison of the time costs is shown in Table 1, with detailed derivation provided in Appendix A. In particular, the VFS statistics only depend on $(s+1)n_1$ observations with $n_1 \ll n$, enabling feature screening with minimal costs, and facilitating downstream tasks. Moreover, to enhance the generalizability of our method, it is essential to utilize a model-free statistic for the screening process. We propose a VFS screening statistic whose definition and theoretical properties are provided in Section 4.

Table 1: Comparison of costs between classical screening procedure with VFS based on group partition. The costs for encrypting, sending, and decrypting a number are ℓ_e, ℓ_s, and ℓ_d, respectively, and the cost of multiplying a plaintext number with a ciphertext number is ℓ_c. Without loss of generality, assume that the time complexity of the screening statistic is $O(n^2)$.

Cost	Classical Screening	VFS
Encryption	$O(n^2\ell_e)$	$O((s+1)^2n_1^2\ell_e)$
Computation	$O(n^2p\ell_c)$	$O\left((s+1)^2n_1^2(\frac{p}{s_0}+s_0)\ell_c\right)$
Communication	$O(p\ell_s)$	$O\left((\frac{p}{s_0}+s_0)\ell_s\right)$
Decryption	$O(p\ell_d)$	$O\left((\frac{p}{s_0}+s_0)\ell_d\right)$

4 Theoretical analysis of VFS

In this section, we provide theoretical support for the effectiveness of VFS by illustrating its asymptotic properties. We establish the screening consistency of the model-free VFS statistic and demonstrate its suitability for group screening in VFL with ultrahigh-dimensional features.

4.1 VFS statistics

Before introducing the proposed VFS statistic, we briefly review generalized U-statistics[36]. Consider K independent collections of observations $\{X_1^{(1)}, \cdots, X_{n_1}^{(1)}\}, \cdots, \{X_1^{(K)}, \cdots, X_{n_K}^{(K)}\}$, the *generalized U-statistic* could be defined as

$$U = \prod_{k=1}^{K} \binom{n_k}{m_k}^{-1} \sum_c h\left(X^{(1)}_{i_{1,1}}, \cdots, X^{(1)}_{i_{1,m_1}}; \cdots; X^{(K)}_{i_{K,1}}, \cdots, X^{(K)}_{i_{K,m_K}}\right),$$

where n_k denotes the number of observations in the k-th collection, m_k denotes the number of observations used in kernel function from the k-th collection, $\sum_c$ denotes the summation over all possible combinations of distinct elements $\{i_{k,1}, \ldots, i_{k,m_k}\}$ from $\{1, \ldots, n_k\}$, and $h(X^{(1)}_{i_{1,1}}, \cdots, X^{(1)}_{i_{1,m_1}}; \cdots; X^{(K)}_{i_{k,1}}, \cdots, X^{(K)}_{i_{K,m_K}})$ is symmetric within each block of arguments. This formulation generalizes the classical U-statistic by allowing the kernel to depend on observations from different populations.

In our setting, we adopt the *generalized U-statistic* framework instead of the conventional one because we are particularly interested in scenarios where the class-0 and class-1 observations are imbalanced. By explicitly separating the two classes within the kernel function, this formulation provides greater flexibility in modeling the asymmetric contributions of each class. Inspired by the generalized U-statistics, we consider a sampled statistic to deal with the rare event scenario. Specifically, let $\tilde{n}_0 = \sum_{i=1}^{n_0} \eta_i$, where η_i denotes whether i-th control is selected. We define the model-free VFS statistic for j-th feature X_j as

$$\tilde{T}_j = \binom{\tilde{n}_0}{m_0}^{-1} \binom{n_1}{m_1}^{-1} \sum_c \left(\prod_{l=1}^{m_0} \eta_{i_{0,l}}\right) h\left(X^{(0)}_{j,i_{0,1}}, \ldots, X^{(0)}_{j,i_{0,m_0}}; X^{(1)}_{j,i_{1,1}}, \ldots, X^{(1)}_{j,i_{1,m_1}}\right), \quad (1)$$

where $X^{(0)}_{j,i_0}$ and $X^{(1)}_{j,i_1}$ denote observations from controls and cases, respectively. The kernel function $h(\cdot)$ is symmetric within each set $\{X^{(l)}_{k,i_{l,1}}, \ldots, X^{(l)}_{k,i_{l,m_l}}\}$ for $l = 0, 1$. Denote $\tilde{\theta}_j = \mathrm{E}\{h(X^{(0)}_{j,i_{0,1}}, \ldots, X^{(0)}_{j,i_{0,m_0}}; X^{(1)}_{j,i_{1,1}}, \ldots, X^{(1)}_{j,i_{1,m_1}})\}$, so $\tilde{T}_j$ serves as an unbiased estimator of $\tilde{\theta}_j$. For the k-th group of features $\mathbf{X}_k$, the VFS group statistic is similarly defined as $\tilde{\mathbf{T}}_k$, except that X_j in (1) is replaced by $\mathbf{X}_k$. Note that (1) is not a classical generalized U-statistic because we additionally introduce the sampling indicators η_i.

4.2 Asymptotic properties of VFS statistics

To investigate the theoretical properties of the VFS statistic, we consider the following conditions.

(C1) **(Feature Distribution)** There exist constants $\tilde{c}_1, \tilde{c}_2 > 0$ such that for all $t > 0$,

$$\sup_p \max_{1 \leqslant j \leqslant p} \Pr\{|X_j| \geqslant t\} < \tilde{c}_1 \exp\{-\tilde{c}_2 t\}.$$

(C2) **(Feature Dimension)** There exists $0 < \gamma < 1/2$ such that $\log p = o(n_1^{1-2\gamma})$.

(C3) **(Imbalance Restriction)** Case sample size n_1 is allowed to be $n_1^{\gamma} = \Omega(\log n + \log p)$.

(C4) **(Signal Separation)** There is a signal gap between true feature set $\mathcal{F}$ and irrelevant feature set $\mathcal{F}^c$. Formally, there exists $0 < \kappa < 1/2 - \gamma$ such that $\min_{j \in \mathcal{F}} |\tilde{\theta}_j| - \max_{j \in \mathcal{F}^c} |\tilde{\theta}_j| \geqslant 2n_1^{-\kappa}$.

Condition (C1) holds naturally if $(X_1, X_2, \ldots, X_p)$ is uniformly bounded or follows a multivariate normal distribution. This condition prevents features from taking extreme values with high probability and is widely used in ultrahigh-dimensional data analysis [22, 69]. Condition (C2) is same as a classical assumption in screening methods. However, the effectiveness depends on assumptions related to n_1 instead of n due to sample imbalance. In the balanced case, where n_1 is proportional to n, it reduces to the classical screening assumption. Condition (C3) specifies the growth rate of n_1, preventing it from growing too slowly, thereby ensuring the convergence of $\tilde{T}_j$ [60]. Condition (C4) requires sufficiently strong signals for true features to distinguish them from numerous irrelevant ones [22, 48]. Under Conditions (C1)-(C4), we can prove the following theorem.

Theorem 1 (**Asymptotic Properties of VFS Statistics**). *Assume the screening threshold in Stage 2 of VFS is δ. Then we have the following conclusions.*

(1) Convergence Rate:

$$\Pr\left\{|\tilde{T}_j - \tilde{\theta}_j| \geqslant n_1^{-\kappa}\right\} \leqslant O\left(\exp\left\{-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\right\} + n \exp\left\{-\tilde{C}_2 n_1^{\gamma}\right\}\right),$$

where $0 < \gamma < 1/2 - \kappa$, $\tilde{C}_1$ and $\tilde{C}_2$ are constants.

(2) Screening Consistency:

$$\Pr\left\{\mathcal{F} \not\subset \hat{\mathcal{F}}\right\} \leqslant O\left(m\left[\exp\left\{-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\right\} + n\exp\left\{-\tilde{C}_2 n_1^{\gamma}\right\}\right]\right).$$

(3) Model Size Bound:

$$\Pr\left\{|\hat{\mathcal{F}}| \leqslant \frac{2}{\delta}\sum_j |\tilde{\theta}_j|\right\} \geqslant 1 - O\left(p\left[\exp\left\{-\tilde{C}_3\delta^2 n_1^{1-2\gamma}\right\} + n\exp\left\{-\tilde{C}_4 n_1^{\gamma}\right\}\right]\right),$$

where $\tilde{C}_3$ and $\tilde{C}_4$ are constants, $|\cdot|$ denotes the number of elements in the set.

Theorem 1(1) demonstrates that $\tilde{T}_j$ converges to its expected value $\tilde{\theta}_j$ and specifies the convergence rate. Theorem 1(2) shows that the probability of failing to screen true features is negligibly small, thereby confirming the screening consistency of $\tilde{T}_j$. Theorem 1(3) states that, for any threshold δ, we can derive an upper bound, which constrains the number of screened features with high probability. In summary, subsampling in VFS significantly reduces computational costs while maintaining its effectiveness for feature screening, even for rare event. This guarantees the statistical property of Stage 2 in VFS algorithm.

For Stage 1 of VFS, noting that we employ hypothesis testing, rather than threshold screening, to determine whether each group of features should be retained. Its validity is ensured by Theorem 2, with the definitions of $h_{a,b}$ and $\xi_{a,b}$ provided in Appendix B.4. Define the null hypothesis as $\{\mathbf{H}_0^k : \mathbf{X}_k \perp\!\!\!\perp Y\}$, which means all features within the k-th group are independent of the corresponding response variable Y.

Theorem 2 (**Asymptotic Normality of Group-Based VFS Statistics**). *Under the null hypothesis and $n_1/n \to 0$, if $\xi_{0,1} > 0$ or $\xi_{1,0} > 0$, and $\mathrm{E}\{h_{1,0}^4(\mathbf{X}_k^{(0)})\} < \infty$, then the group-based VFS statistics $\tilde{\mathbf{T}}_k$ satisfies $n_1^{1/2}\tilde{\mathbf{T}}_k \xrightarrow{d} N(0, m_1^2\xi_{0,1} + m_0^2\xi_{1,0}/s)$.*

When $\xi_{0,1} = \xi_{1,0} = 0$, $\xi_{0,2} > 0$ and $s \to \infty$, further denote

$$G(x,y) = \mathrm{E}\{h_{0,2}(\mathbf{X}_k^{(1)}, x)h_{0,2}(\mathbf{X}_k^{(1)}, y)\}.$$

If $h_{0,2}(\cdot,\cdot)$ depending on n_1 satisfying

$$\frac{\mathrm{E}\left\{G^2(\mathbf{X}_{k,1}^{(1)}\mathbf{X}_{k,2}^{(1)})\right\} + n_1^{-1}\mathrm{E}\left\{h_{0,2}^4(\mathbf{X}_{k,1}^{(1)}, \mathbf{X}_{k,2}^{(1)})\right\}}{\mathrm{E}\left\{h_{0,2}^2(\mathbf{X}_{k,1}^{(1)}, \mathbf{X}_{k,2}^{(1)})\right\}^2} \to 0, \tag{2}$$

where $\mathbf{X}_{k,1}^{(1)}$ and $\mathbf{X}_{k,2}^{(1)}$ are indepnedent copy of $\mathbf{X}_k^{(1)}$, then $n_1\tilde{\mathbf{T}}_k/\xi_{0,2}^{1/2} \xrightarrow{d} N(0, m_1^2(m_1-1)^2/2)$ as $n_1 \to \infty$.

The proof of Theorem 2 can be found in [35]. Theorem 2 motivates us to group all features and conduct hypothesis testing on each group. Rejecting the null hypothesis implies that the features in the group are not independent of Y, suggesting they may contain true feature(s). It is worth noting that Theorem 2 does not require a specific grouping strategy, which implies that the grouping strategy does not have a significant impact on the screening results. This guarantees the statistical property of Stage 1 in VFS algorithm.

In summary, the VFS statistic effectively addresses data imbalance challenges while maintaining robust performance under balanced scenarios. By selecting appropriate kernel functions, it generalizes existing test statistics–such as distance correlation [28, 61] and projection correlation [77, 79]–to rare event scenarios. It captures complex nonlinear dependencies often overlooked by linear methods. Developed within a general framework and supported by solid theoretical guarantees, it offers a reliable foundation for a wide range of applications. These strengths make the VFS statistic a adaptable and robust tool for feature screening.

Remark 2 (Generalization to Multi-Party and Multi-Class Scenarios). The properties of VFS discussed in this section can be naturally extended to both multi-party and multi-class settings. In the multi-party setting, each passive party only needs to compute the VFS statistics for its own features, while the active party handles decryption; no interaction between features held by different parties is required. In the multi-class setting, the kernel function h in the VFS statistics only needs to be modified to accommodate differences among multiple classes. The properties and derivations of this extension are analogous to the theorems presented in this section, with further details provided in Appendix C.

Notably, the VFS framework differs from classical multiple testing in that it does not require controlling the type I error; moreover, without introducing any additional mechanisms, it can still guarantee that the cumulative type II error across iterations converges to zero under certain assumptions. In addition, Stage 2 plays an essential role in refining feature screening and should not be skipped. Detailed discussions are also provided in Appendix C.

5 Simulation studies

In this section, we adopt numerical results on simulated datasets to demonstrate the effectiveness of VFS. In the following parts, we use the kernel function similar to the distance covariance but designed for rare events, whose definition and properties are presented in the Appendix D.

5.1 Simulation settings

Data generation. The simulated dataset is created with the following steps:

Step 1. Covariance matrix. We randomly generate $\mathbf{X} \in \mathbb{R}^{n\times p}$ from a multivariate normal distribution with zero-mean and covariance matrix $\Sigma = (\sigma_{j_1,j_2})_{p\times p}$, where $\sigma_{j_1,j_2} = 0.5^{|j_1-j_2|}$.

Step 2. True features. In real-world scenarios, true features often tend to appear in groups. Therefore, we select m true features in groups of 5 with the coefficients for each group predefined as $(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) = (3, -4, 2, -2, -3)$.

Step 3. Response. For $1 \leqslant i \leqslant n$, the binary label variable Y_i is drawn from a Bernoulli distribution with success probability Y_i^*. We generate Y^* from the following models:

Model 1: $Y_i^* = \mathrm{L}\left(\sum_{j=1}^{m} \beta_j X_{j,i} - \gamma\right)$

Model 2: $Y_i^* = \mathrm{L}\left(\exp\left\{\sum_{j=1}^{m} \beta_j X_{j,i}\right\} - \gamma\right)$

Model 3: $Y_i^* = \mathrm{L}\left(\sum_{j=1}^{\frac{m}{2}} \beta_j X_{j,i} + \sum_{j=\frac{m}{2}+1}^{m} \beta_j \exp\{X_{j,i}\} - \gamma\right)$

Here, $\mathrm{L}(t)$ denotes the Logistic function, γ is a positive constant ensuring that n_1/n is small enough.

Encryption settings. To conduct feature screening in a VFL system, different encryption methods are required depending on the selected statistic. In these experiments, we consider both the Paillier [58] and CKKS [13] homomorphic encryption schemes, following the paired encryption procedure described in Section 3.1. Since the results are almost the same, we only present the result for Paillier.

Performance measurement. We repeated each experiment $M = 200$ times on a machine equipped with an Intel(R) Xeon(R) Gold 5320 CPU @ 2.20GHz and 72 GB of RAM, and evaluated the screening performance from two perspectives: (1) statistical effectiveness, measured by the Positive Selection Rate (PSR = $|\hat{\mathcal{F}} \cap \mathcal{F}|/|\mathcal{F}|$) and the False Discovery Rate (FDR = $|\hat{\mathcal{F}} - \mathcal{F}|/|\hat{\mathcal{F}}|$), and (2) computational efficiency, evaluated by the average total computation time denoted as Time.

Hyperparameter tuning. To demonstrate the robustness of VFS, we design comparative experiments of hyperparameter selection. Specifically, we investigate the impact of hyperparameter on the performance of VFS, including subsampling ratios $s = 1, 2, 3, 4, 5$, initial group size $s_0 = 10, 20, 50, 100, 200$, and number of retained features after screening $d = 50, 100, 150, 200, 250$.

Comparison with other methods. It is remarkable that most existing literature focuses on feature selection and few screening methods have been developed for VFL. Since VFS includes distance covariance and improved projection covariance designed for rare events as special cases, we compare the screening results with these corresponding classical statistics in simulations. Additionally, we also include comparisons with mutual information [75] and Chi-squared statistics [34], which are not encompassed within the VFS framework.

5.2 Simulation results

A subset of the experimental results is presented in Figure 4, while the complete simulation results can be found in Appendix E. Compared to classical screening methods that evaluate each feature individually, VFS significantly reduces computational time while achieving comparable or even superior performance. This advantage becomes greater as the feature dimension p increases, because VFS can quickly filter out a large number of irrelevant features. This is consistent with the theoretical results and strongly demonstrates the role of VFS in reducing feature dimensionality.

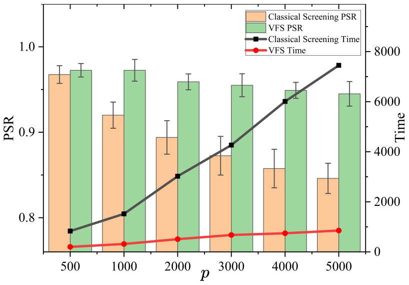
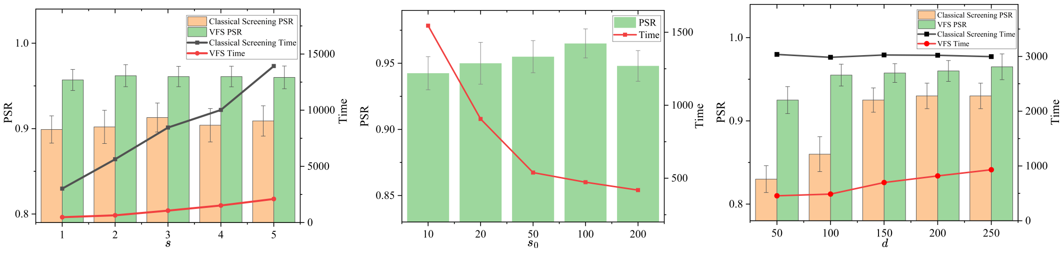


Figure 4: Performance of VFS and classical screening.

Moreover, as shown in Figure 5, VFS is robust to hyperparameter settings. Increasing the subsampling ratio s raises computational costs but provides little performance improvements. Choosing $s = 1$ is sufficient for handling our simulated data. A larger initial group size s_0 leads to greater time savings, but excessively large s_0 may hinder the identification of true features. The number of retained features after screening d involves a trade-off between PSR and computational time.



(a) Subsampling ratio s.

(b) Initial group size s_0.

(c) Number of retained features d.

Figure 5: PSR and computational time of different hyperparameter.

Finally, we compare VFS with other feature screening methods. The results presented in Table 2 demonstrate that VFS exhibits significant advantages.

Table 2: PSR ± standard deviation of different feature screening methods.

Screening Method	Model 1	Model 2	Model 3
VFS	0.965 ± 0.012	0.985 ± 0.013	0.856 ± 0.013
Mutual Information	0.789 ± 0.017	0.847 ± 0.020	0.666 ± 0.019
Chi-squared	0.904 ± 0.016	0.949 ± 0.014	0.712 ± 0.017

6 Real data analysis

To illustrate the performance of VFS in real data applications, we first consider four publicly available real-world datasets: Activity [1], Gina [32], p53 Mutants [42], and RNA-Seq [25]. We adopt LESS-VFL [7] and MUSE [37] as feature selection methods. The following two approaches are considered: (1) **VFS+Selection** (VFS+): Feature screening is first performed using VFS algorithm, followed by further feature selection using LESS-VFL or MUSE; (2) **Selection**: Directly applying LESS-VFL or MUSE for selection on all features.

Since the true feature set $\mathcal{F}$ is unknown in real-world datasets, PSR and FDR cannot be directly computed. Therefore, we compare model performance using Accuracy and AUC of predictive models, and also compare the computational cost (i.e., Time) of the two methods. As shown in Table 3, VFS+Selection achieves comparable accuracy while significantly reducing computational costs, with this advantage becoming more pronounced as the feature dimension p increases. For instance, in RNA-Seq dataset, VFS+Selection requires less than 5% of the computation time compared to using either LESS-VFL or MUSE alone. Table 11 in Appendix E provides the complete results for all datasets, which also demonstrates the robustness of VFS in real-world applications.

Table 3: Feature selection performance. Accuracy and AUC are evaluated on the test set, Time includes the time for feature screening (if performed) and feature selection.

Dataset	Method	LESS-VFL			MUSE		
		Accuracy	AUC	Time	Accuracy	AUC	Time
Activity	VFS+	1.000	1.000	678.5	1.000	1.000	23.0
p =560	Selection	0.998	1.000	953.4	1.000	1.000	44.8
Gina	VFS+	0.836	0.915	201.8	0.842	0.842	21.8
p=968	Selection	0.843	0.924	467.0	0.843	0.842	61.5
p53	VFS+	0.835	0.830	248.8	0.839	0.845	53.0
$p = 5408$	Selection	0.843	0.859	1493.3	0.849	0.862	279.3
RNA-Seq	VFS+	0.994	1.000	74.9	1.000	1.000	30.9
$p = 20531$	Selection	0.944	1.000	1791.1	1.000	1.000	1062.3

To further demonstrate the robustness of VFS in actual VFL pipelines, we conduct an application-based evaluation motivated by a real-world credit modeling task in collaboration with a leading third-party payment platform in China. Our goal is to build a model using merchant risk labels as the response and a large number of transaction-based features as inputs. The dataset contains a sample size of $n = 10{,}000$ merchants and a feature dimension of $p = 189{,}236$, covering key features in different groups and their interactions. Those feature groups are, respectively, merchant attributes, transaction activity, revenue scale, business growth, customer structure, payment channel distribution, transaction stability, and temporal transaction patterns. The model outputs are further utilized to inform financial product recommendation strategies for merchants.

After removing redundant features with the VFS algorithm, we validate its efficiency gains in downstream VFL tasks using SplitNN [66], a neural network-based VFL method, as an example. The results in Table 4 show that VFS achieves a substantial reduction in computational cost, with the runtime reduced to less than 10% of that required by the original method. Furthermore, VFS achieves improved out-of-sample AUC while using significantly fewer features. This application-based evaluation further demonstrates the practical relevance, scalability, and robustness of VFS, especially in scenarios involving ultrahigh-dimensional features.

Table 4: Performance of VFS on the real-world credit modeling dataset using SplitNN.

# Features	AUC	Screening Time	Modeling Time	Total Time
100	0.937	29.1	100.3	129.4
500	0.941	30.9	125.3	156.2
1000	0.936	31.7	151.9	183.6
All	0.901	-	1911.8	1911.8

7 Conclusion

In this paper, we introduce Vertical Federated Feature Screening (VFS), a two-stage feature screening algorithm based on model-free statistics and designed for accelerating feature selection process in VFL, especially when handling with ultrahigh-dimensional and rare event datasets. Theoretical analysis demonstrates that by combining subsampling and group screening techniques, VFS effectively eliminates irrelevant features in Stage 1, while providing more precise screening in Stage 2. This approach reduces the overall costs of existing feature selection methods. Experiments on both synthetic and real-world datasets show that VFS not only lowers computational costs but also retains strong screening performance, underscoring its potential for real-world VFL applications.

While VFS is effective in practice, it also has certain limitations and offers several avenues for future improvement. First, although VFS is a flexible framework that accommodates a wide range of screening statistics, its computational efficiency may depend on the specific statistic used. Second, as VFS conducts marginal screening, the retained feature set may still contain redundancies. Finally, an interesting direction for future research lies in integrating homomorphic encryption with differential privacy to strengthen resilience against potential attacks targeting homomorphic encryption.

Acknowledgments and Disclosure of Funding

This research is supported by the National Natural Science Foundation of China (NSFC 72471230, 12225113 and 12171477), the National Key R&D Program of China (2023YFA1008702 and 2023YFC3304701), the Beijing Social Science Fund(23GLA008), the MOE Project of Key Research Institute of Humanities and Social Sciences (grant 22JJD110001) and Public Computing Cloud, Renmin University of China.

References

[1] D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In Esann, volume 3, pages 3–4, 2013.

[2] L. Bao, C. Juan, J. Li, and Y. Zhang. Boosted near-miss under-sampling on svm ensembles for concept detection in large-scale imbalanced datasets. Neurocomputing, 172:198–206, 2016.

[3] A. Ben-David, N. Nisan, and B. Pinkas. Fairplaymp: a system for secure multi-party computation. In Proceedings of the 15th ACM Conference on Computer and Communications Security, CCS '08, page 257–266, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781595938107.

[4] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser. Machine learning classification over encrypted data. Cryptology ePrint Archive, 2014.

[5] D. Cai, T. Fan, Y. Kang, L. Fan, M. Xu, S. Wang, and Q. Yang. Accelerating Vertical Federated Learning . IEEE Transactions on Big Data, 10(06):752–760, Dec. 2024. ISSN 2332-7790.

[6] Z. Cai, R. Li, and Y. Zhang. A distribution free conditional independence test with applications to causal discovery. Journal of Machine Learning Research, 23(85):1–41, 2022.

[7] T. Castiglia, Y. Zhou, S. Wang, S. Kadhe, N. Baracaldo, and S. Patterson. LESS-VFL: Communication-efficient feature selection for vertical federated learning. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, Proceedings of the 40th International Conference on Machine Learning, volume 202 of Proceedings of Machine Learning Research, pages 3757–3781. PMLR, 23–29 Jul 2023.

[8] T. J. Castiglia, A. Das, S. Wang, and S. Patterson. Compressed-VFL: Communication-efficient learning with vertically partitioned data. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, Proceedings of the 39th International Conference on Machine Learning, volume 162 of Proceedings of Machine Learning Research, pages 2738–2766. PMLR, 17–23 Jul 2022.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16:321–357, 2002.

[10] Z. Chen, Z. Gu, Y. Lu, X. Ren, R. Zhong, W.-J. Lu, J. Zhang, Y. Zhang, H. Wu, X. Zheng, H. Liu, T. Chu, C. Hong, C. Wei, D. Niu, and Y. Xie. Safe: A scalable homomorphic encryption accelerator for vertical federated learning. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 44(5):1662–1675, 2025.

[11] K. Cheng, T. Fan, Y. Jin, Y. Liu, T. Chen, D. Papadopoulos, and Q. Yang. Secureboost: A lossless federated learning framework. IEEE Intelligent Systems, 36(6):87–98, 2021.

[12] X. Cheng and H. Wang. A generic model-free feature screening procedure for ultra-high dimensional data with categorical response. Computer Methods and Programs in Biomedicine, 229:107269, 2023.

[13] J. H. Cheon, A. Kim, M. Kim, and Y. Song. Homomorphic encryption for arithmetic of approximate numbers. In T. Takagi and T. Peyrin, editors, Advances in Cryptology – ASIACRYPT 2017, pages 409–437, Cham, 2017. Springer International Publishing.

[14] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 29. Curran Associates, Inc., 2016.

[15] H. Cui, R. Li, and W. Zhong. Model-free feature screening for ultrahigh dimensional discriminant analysis. Journal of the American Statistical Association, 110(510):630–641, 2015.

[16] C. Dai, B. Lin, X. Xing, and J. S. Liu. False discovery rate control via data splitting. Journal of the American Statistical Association, 118(544):2503–2520, 2023.

[17] J. de Haan-Ward, S. J. Bonner, and D. Woolford. On the prediction of rare events when sampling from large data. Communications in Statistics-Simulation and Computation, pages 1–21, 2024.

[18] G. Douzas, F. Bacao, and F. Last. Improving imbalanced learning through a heuristic oversampling method based on k-means and smote. Information Sciences, 465:1–20, 2018.

[19] C. Drummond, R. C. Holte, et al. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In Workshop on Learning from Imbalanced Datasets II, volume 11, 2003.

[20] O. J. Dunn. Multiple comparisons among means. Journal of the American statistical association, 56(293):52–64, 1961.

[21] A. Falcetta and M. Roveri. Privacy-preserving deep learning with homomorphic encryption: An introduction. IEEE Computational Intelligence Magazine, 17(3):14–25, 2022.

[22] J. Fan and J. Lv. Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society Series B: Statistical Methodology, 70(5):849–911, 2008.

[23] J. Fan and J. Lv. A selective overview of variable selection in high dimensional feature space. Statistica Sinica, 20(1):101—148, January 2010. ISSN 1017-0405.

[24] S. Feng. Vertical federated learning-based feature selection with non-overlapping sample utilization. Expert Systems with Applications, 208:118097, 2022.

[25] S. Fiorini. gene expression cancer RNA-Seq. UCI Machine Learning Repository, 2016.

[26] W. Fithian and T. Hastie. Local case-control sampling: Efficient subsampling in imbalanced data sets. Annals of statistics, 42(5):1693, 2014.

[27] R. Fu, Y. Wu, Q. Xu, and M. Zhang. Feast: A communication-efficient federated feature selection framework for relational data. Proceedings of the ACM on Management of Data, 1 (1):1–28, 2023.

[28] L. Gao, Y. Fan, J. Lv, and Q.-M. Shao. Asymptotic distributions of high-dimensional distance correlation inference. Annals of Statistics, 49(4):1999, 2021.

[29] C. Gentry. Fully homomorphic encryption using ideal lattices. In Proceedings of the forty-first annual ACM symposium on Theory of computing, pages 169–178, 2009.

[30] S. Greeland and D. C. Thomas. On the need for the rare disease assumption in case-control studies. American Journal of Epidemiology, 116(3):547–553, 09 1982. ISSN 0002-9262.

[31] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. Journal of Machine Learning Research, 3(Mar):1157–1182, 2003.

[32] I. Guyon, A. Saffari, G. Dror, and G. Cawley. Agnostic learning vs. prior knowledge challenge. In 2007 International Joint Conference on Neural Networks, pages 829–834. IEEE, 2007.

[33] H. He and E. A. Garcia. Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9):1263–1284, 2009.

[34] D. Huang, R. Li, and H. Wang. Feature screening for ultrahigh dimensional categorical data with applications. Journal of Business & Economic Statistics, 32(2):237–244, 2014.

[35] D. Huang, L. Wang, and L. Zhu. Do more observations bring more information in rare events?, 2025. URL `https://arxiv.org/abs/2506.13671`.

[36] S. Janson and K. Nowicki. The asymptotic distributions of generalized U-statistics with applications to random graphs. Probability Theory and Related Fields, 90(3):341–375, Sept. 1991. ISSN 1432-2064.

[37] X. Ji, C. Wang, O. Gadyatskaya, F. Zhao, Z. Mao, and W. Xi. Muse: A trustworthy vertical federated feature selection framework. IEEE Transactions on Computational Social Systems, 2024.

[38] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1–2):1–210, 2021.

[39] F. Kamalov, F. Thabtah, and H. H. Leung. Feature selection in imbalanced data. Annals of Data Science, 10(6):1527–1541, 2023.

[40] B. Knott, S. Venkataraman, A. Hannun, S. Sengupta, M. Ibrahim, and L. van der Maaten. Crypten: Secure multi-party computation meets machine learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, Advances in Neural Information Processing Systems, volume 34, pages 4961–4973. Curran Associates, Inc., 2021.

[41] Y. Kou, C. Lu, S. Sirwongwattana, and Y. Huang. Survey of fraud detection techniques. In 2004 IEEE International Conference on Networking, Sensing and Control, pages 749–754, 2004.

[42] R. Lathrop. p53 Mutants. UCI Machine Learning Repository, 2009.

[43] A. J. Lee. U-statistics: Theory and Practice. Routledge, 2019.

[44] S. Lewallen and P. Courtright. Epidemiology in practice: case-control studies. Community eye health, 11(28):57—58, 1998. ISSN 0953-6833.

[45] A. Li, H. Peng, L. Zhang, J. Huang, Q. Guo, H. Yu, and Y. Liu. Fedsdg-fs: Efficient and secure feature selection for vertical federated learning. In IEEE INFOCOM 2023-IEEE Conference on Computer Communications, pages 1–10. IEEE, 2023.

[46] B. Li, D. Micciancio, M. Schultz-Wu, and J. Sorrell. Securing approximate homomorphic encryption using differential privacy. In Annual International Cryptology Conference, pages 560–589. Springer, 2022.

[47] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. ACM computing surveys (CSUR), 50(6):1–45, 2017.

[48] R. Li, W. Zhong, and L. Zhu. Feature screening via distance correlation learning. Journal of the American Statistical Association, 107(499):1129–1139, 2012.

[49] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. IEEE signal processing magazine, 37(3):50–60, 2020.

[50] W.-C. Lin, C.-F. Tsai, Y.-H. Hu, and J.-S. Jhang. Clustering-based undersampling in class-imbalanced data. Information Sciences, 409:17–26, 2017.

[51] W. Liu, Y. Ke, J. Liu, and R. Li. Model-free feature screening and fdr control with knockoff features. Journal of the American Statistical Association, 117(537):428–443, 2022.

[52] X.-Y. Liu, J. Wu, and Z.-H. Zhou. Exploratory undersampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2):539–550, 2008.

[53] Y. Liu, Y. Kang, T. Zou, Y. Pu, Y. He, X. Ye, Y. Ouyang, Y.-Q. Zhang, and Q. Yang. Vertical federated learning: Concepts, advances, and challenges. IEEE Transactions on Knowledge and Data Engineering, 36(7):3615–3634, 2024.

[54] F. Luo, S. Al-Kuwari, and Y. Ding. Svfl: Efficient secure aggregation and verification for cross-silo federated learning. IEEE Transactions on Mobile Computing, 23(1):850–864, 2024.

[55] S. Maldonado, R. Weber, and F. Famili. Feature selection for high-dimensional class-imbalanced data sets using support vector machines. Information Sciences, 286:228–246, 2014. ISSN 0020-0255.

[56] N. Meinshausen and P. Bühlmann. Stability selection. Journal of the Royal Statistical Society Series B: Statistical Methodology, 72(4):417–473, 08 2010.

[57] A. Mojiri, A. Khalili, and A. Zeinal Hamadani. New hard-thresholding rules based on data splitting in high-dimensional imbalanced classification. Electronic Journal of Statistics, 16(1): 814–861, 2022.

[58] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In J. Stern, editor, Advances in Cryptology — EUROCRYPT '99, pages 223–238, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[59] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2):120–126, 1978.

[60] Y. Sang and X. Dang. Grouped feature screening for ultrahigh-dimensional classification via gini distance correlation. Journal of Multivariate Analysis, 204:105360, 2024.

[61] D. Sejdinovic, B. Sriperumbudur, A. Gretton, and K. Fukumizu. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. The Annals of Statistics, pages 2263–2291, 2013.

[62] R. J. Serfling. Approximation theorems of mathematical statistics. John Wiley & Sons, 2009.

[63] V. S. Spelmen and R. Porkodi. A review on handling imbalanced data. In 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), pages 1–11. IEEE, 2018.

[64] M. Tomz, G. King, and L. Zeng. Relogit: Rare events logistic regression. Journal of Statistical Software, 8:1–27, 2003.

[65] Z. Tong, Z. Cai, S. Yang, and R. Li. Model-free conditional feature screening with fdr control. Journal of the American Statistical Association, 118(544):2575–2587, 2023.

[66] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar. Split learning for health: Distributed deep learning without sharing raw patient data. CoRR, abs/1812.00564, 2018. URL http://arxiv.org/abs/1812.00564.

[67] N. Volgushev, M. Schwarzkopf, B. Getchell, M. Varia, A. Lapets, and A. Bestavros. Conclave: secure multi-party computation on big data. In Proceedings of the Fourteenth EuroSys Conference 2019, EuroSys '19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362818.

[68] B. Wang, Y. Chen, H. Jiang, and Z. Zhao. Ppefl: Privacy-preserving edge federated learning with local differential privacy. IEEE Internet of Things Journal, 10(17):15488–15500, 2023.

[69] H. Wang. Forward regression for ultra-high dimensional variable screening. Journal of the American Statistical Association, 104(488):1512–1524, 2009.

[70] H. Wang. Logistic regression for massive data with rare events. In International Conference on Machine Learning, pages 9829–9836. PMLR, 2020.

[71] H. Wang, A. Zhang, and C. Wang. Nonuniform negative sampling and log odds correction with rare events data. Advances in Neural Information Processing Systems, 34:19847–19859, 2021.

[72] J. Wu and H. Cui. Model-free feature screening based on hellinger distance for ultrahigh dimensional data. Statistical Papers, 65(9):5903–5930, 2024.

[73] Y. Wu, N. Xing, G. Chen, T. T. A. Dinh, Z. Luo, B. C. Ooi, X. Xiao, and M. Zhang. Falcon: A privacy-preserving and interpretable vertical federated learning system. Proceedings of the VLDB Endowment, 16(10):2471–2484, 2023.

[74] Z. Wu, J. Hou, and B. He. Vertibench: Advancing feature distribution diversity in vertical federated learning benchmarks. In The Twelfth International Conference on Learning Representations, 2024.

[75] Y. Xu and Q. Qian. i-sisso: Mutual information-based improved sure independent screening and sparsifying operator algorithm. Engineering Applications of Artificial Intelligence, 116: 105442, 2022.

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

[77] Y. Zhang and L. Zhu. Projective independence tests in high dimensions: the curses and the cures. Biometrika, 111(3):1013–1027, 2024.

[78] L. Zhu, L. Li, R. Li, and L. Zhu. Model-free feature screening for ultrahigh-dimensional data. Journal of the American Statistical Association, 106(496):1464–1475, 2011.

[79] L. Zhu, K. Xu, R. Li, and W. Zhong. Projection correlation between two random vectors. Biometrika, 104(4):829–843, 2017.

[80] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society. Series B (Statistical Methodology), 67(2):301–320, 2005. ISSN 13697412, 14679868.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and introduction, we explicitly articulate the contributions of this paper to the existing research on vertical federated learning (VFL). Specifically, we propose a model-free VFS algorithm, which effectively reduces computational, communication, and encryption costs in a VFL system.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Section 2, we analyze the limitations of existing research from three key perspectives: federated feature selection, feature screening and independence testing, and the challenge of imbalanced data. This analysis further highlights the significance and necessity of the present study.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: In Section 4, we begin by presenting four assumptions (C1)–(C4) and providing justifications for their validity. Building upon these assumptions, we establish Theorems 1 and 2, with detailed proofs included in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: This paper introduces a novel VFS algorithm, whose implementation details are provided in Algorithm 1. The simulation settings are specified in Section 5, and the datasets along with the methodologies used for real data applications are described in Section 6. To facilitate reproducibility, the definition of the VFS statistic employed in the experiments is included in Appendix D.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All datasets used in this study are publicly available and properly cited. To facilitate the reproducibility of our findings, we will provide the essential code for the VFS algorithm framework as supplemental material.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The main text outlines the basic settings of the simulation studies and real data experiments, while Appendix E provides more detailed information on the parameter settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Figure 4 and Figure 5 present a visualization of the numerical simulation results, along with information on the statistical significance of the experiments. Furthermore, Appendix E contains more comprehensive experimental results to further strengthen the credibility of our findings.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In Section 5, we provide details of the equipment used for running the experiments. Since the focus of our experiments is on comparing the runtime of different methods rather than the absolute execution time, there are no specific requirements for the equipment configuration to ensure reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have thoroughly reviewed the NeurIPS Code of Ethics and affirm that this paper fully conforms to its guidelines.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This paper's proposed Vertical Federated Feature Screening (VFS) algorithm has significant applications in fields like healthcare and finance, enabling institutions to collaborate on sensitive data while preserving privacy, and enhancing the efficiency of collaborative modeling in privacy-preserving frameworks.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: After thorough consideration, we conclude that the VFS algorithm presented in this paper does not present any apparent risk of misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The experiments in this paper utilize publicly available datasets, and the original papers corresponding to these datasets are duly cited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: To support the reproducibility of the results, we provide a concise code framework for the VFS algorithm, which is included in the zip file of the Supplementary Material.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing or research involving human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing or research involving human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The primary approach developed in this research does not incorporate LLMs as essential, novel, or non-traditional elements.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Appendix

The appendix consists of the following parts. Appendix A presents the derivation of the computational costs reported in Table 1. Appendix B provides the proofs of the main theorems. Appendix C discusses several technical details and extensions of the VFS framework. Appendix D defines the screening statistic used in the experiments. Finally, Appendix E reports additional experimental results.

A Computational complexity analysis of algorithm

First, we discuss the number of encrypted computations of VFS statistics required for feature screening. For classical screening, we need to compute $\tilde{T}_j$ for each of the p features X_j, so the number of computations can be written as $O(p)$. For VFS, the number of computations depends on the distribution of m true features. We consider two extreme cases.

Best Case. All m true features are divided in one group. After one round of group screening, only this group is retained. Assuming $s_0 < p'$, these s_0 features need to be screened individually. Thus, the total number of computations is $O\left(\frac{p}{s_0} + s_0\right)$.

Worst Case. In each round of group screening, all m true features are placed in different groups. Assuming the group screening process involves r rounds, it is obvious that the first round will require $\frac{p}{s_0}$ computations, and the subsequent $r-1$ rounds will each require $\frac{m}{\rho}$ computations, where r satisfies $m\rho^{r-1}s_0 < p'$. Noting that p' features need to be screened individually, the total number of computations is $O\left(\frac{p}{s_0} + \frac{m}{\rho}\log_\rho\left(\frac{p'}{ms_0}\right) + p'\right)$.

To summarize, we combine the results discussed above in Table 5.

Table 5: Comparison of encrypted computation counts for $\tilde{\mathbf{T}}_k$ across different methods.

Method	Classical Screening	VFS (Best Case)	VFS (Worst Case)
Computation Counts	$O(p)$	$O\left(\frac{p}{s_0} + s_0\right)$	$O\left(\frac{p}{s_0} + \frac{m}{\rho}\log_\rho\left(\frac{p'}{ms_0}\right) + p'\right)$

Next, we will outline the cost for each part of feature screening in VFL. Recall that the costs for encrypting, sending, and decrypting a number are denoted as ℓ_e, ℓ_s, and ℓ_d, respectively, while the cost of multiplying a plaintext number with a ciphertext number is ℓ_c. In general, let P represent the number of encrypted computations of VFS statistics required for feature screening, and let N denote the sample size.

Encryption. The label matrix $\mathbf{Y} \in \mathbb{R}^{N\times N}$ needs to be encrypted, so the encryption cost is $O(N^2\ell_e)$.

Computation. Since P encrypted computations of VFS statistics are required, and each computation has a time complexity proportional to the square of the sample size, i.e. N^2, the computation cost can be written as $O(N^2P\ell_c)$.

Communication. After each encrypted VFS statistic is computed, the result needs to be sent, so the communication cost is $O(P\ell_s)$.

Decryption. After the active party receives encrypted VFS statistics, decryption is required for each, so the decryption cost is $O(P\ell_d)$.

Clearly, for classical screening, $N = n$, and for VFS, $N = (s+1)n_1$. By substituting the computation counts P from Table 5 into these expressions, we obtain the results presented in the following table.

B Proof of theorem

B.1 Useful lemmas

Before presenting the proof details of the theorem, we introduce two useful lemmas.

Lemma 1 (Hoeffding's Inequality for Two Sample U-Statistics)**.** *Let $X_1, \ldots, X_m, Y_1, \ldots, Y_n$ be $(m+n)$ independent random variables. For $m \geqslant r$ and $n \geqslant s$, consider a random variable of the*

Table 6: Comparison of costs for feature screening across different methods in an appropriate group partition. Without loss of generality, assume that the time complexity of the chosen statistic is $O(n^2)$.

Cost	Classical Screening	VFS (Best Case)	VFS (Worst Case)
Encryption	$O(n^2\ell_e)$	$O((s+1)^2n_1^2\ell_e)$	$O((s+1)^2n_1^2\ell_e)$
Computation	$O(n^2p\ell_c)$	$O\left((s+1)^2n_1^2(\frac{p}{s_0}+s_0)\ell_c\right)$	$O\left((s+1)^2n_1^2\left(\frac{p}{s_0}+\frac{m}{\rho}\log_\rho\left(\frac{p'}{ms_0}\right)+p'\right)\ell_c\right)$
Communication	$O(p\ell_s)$	$O\left((\frac{p}{s_0}+s_0)\ell_s\right)$	$O\left(\left(\frac{p}{s_0}+\frac{m}{\rho}\log_\rho\left(\frac{p'}{ms_0}\right)+p'\right)\ell_s\right)$
Decryption	$O(p\ell_d)$	$O\left((\frac{p}{s_0}+s_0)\ell_d\right)$	$O\left(\left(\frac{p}{s_0}+\frac{m}{\rho}\log_\rho\left(\frac{p'}{ms_0}\right)+p'\right)\ell_d\right)$

form

$$U=\binom{m}{r}^{-1}\binom{n}{s}^{-1}\sum_c h(X_{i_1},\dots,X_{i_r},Y_{j_1},\dots,Y_{j_s}),$$

where the summation $\sum_c$ is taken over all r-tuples $(i_1,\dots,i_r)$ of distinct positive integers $\leqslant m$ and all s-tuples $(j_1,\dots,j_s)$ of distinct positive integers $\leqslant n$.

If $a\leqslant h(X_{i_1},\dots,X_{i_r},Y_{j_1},\dots,Y_{j_s})\leqslant b$, then for any $t>0$, and $r\leqslant m$, $s\leqslant n$, we have

$$\Pr\{|U-E(U)|\geqslant t\}\leqslant 2\exp\left\{-\frac{2\lambda t^2}{(b-a)^2}\right\},$$

where $\lambda=\min\{[m/r],[n/s]\}$.

Lemma 2 (Multiplicative Chernoff Bound)**.** *Let $X_1,X_2,\dots,X_n$ be independent Bernoulli random variables, and let $S_n=\sum_{i=1}^n X_i$ denote their sum. Let $\mu=E[S_n]$ be the expected value of S_n. Then, for any $0<\lambda<1$, the following bound holds:*

$$\Pr\{S_n\leqslant(1-\lambda)\mu\}\leqslant\exp\left\{-\frac{\lambda^2\mu}{2}\right\}.$$

B.2 Asymptotic properties of generalized U-statistics

We first focus on the scenario where subsampling is not considered, i.e., when using the full sample. To perform feature screening, it is sufficient to compute the marginal correlation between each X_j and Y, such as Kendall's τ, Spearman's ρ rank correlation, distance correlation, and improved projection correlation [6]. All of these correlation measures can be unified within the framework of two sample U-statistics. Specifically, if Y takes values between 0 and 1, we can define a generalized U-statistic for the j-th feature X_j and Y as follows, using the same notation as in the main text:

$$T_j=\binom{n_0}{m_0}^{-1}\binom{n_1}{m_1}^{-1}\sum_c h(X^{(0)}_{j,i_{0,1}},\dots,X^{(0)}_{j,i_{0,m_0}};X^{(1)}_{j,i_{1,1}},\dots,X^{(1)}_{j,i_{1,m_1}}).$$

Similarly, we propose the following condition in order to study the theoretical properties of T_j:

(C1') **(Feature Distribution)** There exist constants $c_1,c_2>0$ such that for all $t>0$,

$$\sup_p\max_{1\leqslant j\leqslant p}\Pr\{|X_j|\geqslant t\}<c_1\exp\{-c_2t\}.$$

(C2') **(Feature Dimension)** There exists $0<\gamma<1/2$ such that $\log p=o(n_1^{1-2\gamma})$.

(C3') **(Class Imbalance)** Case sample size n_1 satisfing $n_1^\gamma=\Omega(\log n+\log p)$.

(C4') **(Signal Strength)** For all true features X_j with $j\in\mathcal{F}$, there exists $0<\kappa<1/2$ such that

$$\min_{j\in\mathcal{F}}|\theta_j|\geqslant 2n_1^{-\kappa}.$$

Under Conditions (C1')-(C4'), we can obtain a result similar to Theorem 1.

Theorem 3 (Asymptotic Properties of Generalized U-Statistics in VFS)**.** *Assume the screening threshold in Stage 2 is δ. Then we have the following conclusions.*

(1) Convergence Rate.

$$\Pr\left\{|T_j - \theta_j| > n_1^{-\kappa}\right\} \leqslant O\left(\exp\left\{-C_1 n_1^{1-2\gamma-2\kappa}\right\} + n\exp\left\{-C_2 n_1^{\gamma}\right\}\right),$$

where $0 < \kappa < 1/2$*,* $0 < \gamma < 1/2 - \kappa$*,* C_1 *and* C_2 *are constants.*

(2) Screening Consistency.

$$\Pr\left\{\mathcal{F} \not\subset \hat{\mathcal{F}}\right\} \leqslant O\left(m\left[\exp\left\{-C_1 n_1^{1-2\gamma-2\kappa}\right\} + n\exp\left\{-C_2 n_1^{\gamma}\right\}\right]\right).$$

(3) Model Size Bound.

$$\Pr\left\{|\hat{\mathcal{F}}| \leqslant \frac{2}{\delta}\sum_j |\theta_j|\right\} \geqslant 1 - O\left(p\left[\exp\left\{-C_3\delta^2 n_1^{1-2\gamma}\right\} + n\exp\left\{-C_4 n_1^{\gamma}\right\}\right]\right),$$

where C_3, C_4 *are constants.*

Proof. To simplify, we denote $h(X^{(0)}_{j,i_{0,1}}, \ldots, X^{(0)}_{j,i_{0,m_0}}; X^{(1)}_{j,i_{1,1}}, \ldots, X^{(1)}_{j,i_{1,m_1}})$ as h_j. For the needs of the subsequent analysis, we truncate h_j at M, then the kernel function can be written as

$$h_j = h_j \cdot \mathbf{1}\{|h_j| \leqslant M\} + h_j \cdot \mathbf{1}\{|h_j| > M\} \overset{def}{=} h_{j,1} + h_{j,2},$$

and denote $\theta_{j,l} = E[h_{j,l}], T_{j,l} = \binom{n_0}{m_0}^{-1}\binom{n_1}{m_1}^{-1}\sum_c h_{j,l}, l = 1, 2.$

For the first part $T_{j,1}$, since $-M < h_{j,1} < M$, according to Lemma 1, for all $\epsilon > 0$, we have

$$\Pr\left\{|T_{j,1} - \theta_{j,1}| \geqslant \epsilon\right\} \leqslant 2\exp\left\{-\frac{\lambda\epsilon^2}{2M^2}\right\},$$

where $\lambda = \min\{[n_0/m_0], [n_1/m_1]\}$.

Suppose that n_1 is exactly an integer multiple of m_1 and that n_0 is sufficiently large. In this case, we have $\lambda = n_1/m_1$ and

$$\Pr\{|T_{j,1} - \theta_{j,1}| \geqslant \epsilon\} \leqslant 2\exp\left\{-\frac{n_1\epsilon^2}{2m_1M^2}\right\}.$$

It's noteworthy to point out that we only consider the case where n_1 is divisible by m_1; otherwise, we can discard redundant positive observations: this approach loses some observations, but makes our analysis simpler.

For the second part $T_{j,2}$, let $M = n_1^{\gamma}, 0 < \gamma < \frac{1}{2} - \kappa$, then $\theta_{j,2} \to 0$ when $n \to \infty$(in that case, both n_1 and M tend to infinity, and $|h_j| < M$ holds with high probability), thus

$$\Pr\{|T_{j,2} - \theta_{j,2}| > \epsilon\} \leqslant \Pr\{|T_{j,2}| > \epsilon/2\}.$$

If we assume that $|h_j| \to \infty$ only if there exists $X^{(l)}_{j,i_{l,r}} \to \infty, l \in \{0, 1\}, r \in \{1, \ldots, m_l\}$, we can find a constant $\alpha > 0$ such that $|h_j| < M$ if $X_{j,i} < \alpha M$ for each observation $1 \leqslant i \leqslant n$. In this case, $h_{j,2} = 0$ and $T_{j,2} = 0$, so we have the inclusion relationship of events:

$$\{|T_{j,2}| > \epsilon/2\} \subseteq \{\exists i \in \{1, \ldots, n\} \text{ s.t. } |X_{j,i}| > \alpha M\}.$$

Under Condition (C1'), we have

$$\begin{aligned}\Pr\{|T_{j,2} - \theta_{j,2}| > \epsilon\} &\leqslant \Pr\{|T_{j,2}| > \epsilon/2\}\\ &\leqslant n \max_{1\leqslant j\leqslant p} \Pr\{|X_j| > \alpha M\}\\ &\leqslant nc_1\exp\{-c_2\alpha M\},\end{aligned}$$

where $c_1, c_2 > 0$ are constants.

In summary, T_j converges to its parameter function θ_j and we have

$$\begin{aligned}\Pr\{|T_j-\theta_j|>2\epsilon\} &\leqslant \Pr\{|T_{1,j}-\theta_1|>\epsilon\}+\Pr\{|T_{2,j}-\theta_2|>\epsilon\}\\ &\leqslant O(\exp\left\{-\frac{n_1\epsilon^2}{2m_1M^2}\right\}+n\exp\{-c_2\alpha M\}).\end{aligned} \tag{3}$$

At last, we let $\epsilon=n_1^{-\kappa}/2$ and $M=n_1^{\gamma}$, denote $C_1,C_2>0$ as constants, and get the following inequality

$$\Pr\{|T_j-\theta_j|>n_1^{-\kappa}\}\leqslant O(\exp\{-C_1n_1^{1-2\gamma-2\kappa}\}+n\exp\{-C_2n_1^{\gamma}\}).$$

Condition (C3') ensures that $n\exp\{-C_2n_1^{\gamma}\}$ tends to 0 as n and n_1 approach infinity, thereby guaranteeing the convergence of T_j to θ_j.

The proof of Theorem 3(1) is complete.

Noting that $\hat{\mathcal{F}}=\{j:|T_j|>\delta\}$. From condition (C4') and the Theorem 3(1) , we know that for any $j\notin\mathcal{F}$, $\theta_j=0$ and $|T_j|\leqslant n_1^{-\kappa}$ with high probability; for any $j\in\mathcal{F}$, $|\theta_j|\geqslant 2n_1^{-\kappa}$ and $|T_j|\geqslant n_1^{-\kappa}$ with high probability. Specifically, if we let $\delta=n_1^{-\kappa}$, we have

$$\Pr\{j\notin\hat{\mathcal{F}}|j\in\mathcal{F}\}\leqslant O(\exp\{-C_1n_1^{1-2\gamma-2\kappa}\}+n\exp\{-C_2n_1^{\gamma}\}).$$

Since we have m true features, the probability that there exists an important feature that is not screened can be expressed as

$$\Pr\{\mathcal{F}\not\subset\hat{\mathcal{F}}\}\leqslant O(p[\exp\{-C_1n_1^{1-2\gamma-2\kappa}\}+n\exp\{-C_2n_1^{\gamma}\}]).$$

The proof of Theorem 3(2) is complete.

Let $\epsilon=\delta/4$ and $M=n_1^{\gamma}$, (3) can be written as

$$\Pr\left\{|T_j-\theta_j|>\frac{\delta}{2}\right\}\leqslant O(\exp\left\{-C_3\delta^2n_1^{1-2\gamma}\right\}+n\exp\{-C_4n_1^{\gamma}\}), \tag{4}$$

where C_3,C_4 are constants.

We define event $\mathcal{A}=\{\max_{1\leqslant j\leqslant p}|T_j-\theta_j|\leqslant\delta/2\}$. When $\mathcal{A}$ holds, the number of $\{j:|T_j|\geqslant\delta\}$ (i.e. $|\hat{\mathcal{F}}|$) cannot exceed the number of $\{j:|\theta_j|\geqslant\delta/2\}$, which is bounded by $(\delta/2)^{-1}\sum_j|\theta_j|$. Together with (4), we find that the model size (i.e. number of selected feature by our method) is bounded.

$$\Pr\left\{|\hat{\mathcal{F}}|\leqslant\frac{2}{\delta}\sum_j|\theta_j|\right\}\geqslant\Pr\{\mathcal{A}\}\geqslant 1-O(p[(\exp\left\{-C_3\delta^2n_1^{1-2\gamma}\right\}+n\exp\{-C_4n_1^{\gamma}\}]).$$

The proof of Theorem 3(3) is complete. □

B.3 Proof of Theorem 1

Proof. Recall that the VFS statistic is obtained by considering subsampling based on generalized U-statistics. Therefore, the proof strategy for Theorem 1 is essentially similar to that of Theorem 3, as showed below.

First of all, we need to define the event $\mathcal{B}=\{\tilde{n}_0>n_1\}$. We want $\mathcal{B}$ to hold, otherwise we cannot use all positive observations.

Since $\tilde{n}_0$ is a random variable with a mean of sn_1, from Lemma 2, for any $0<\lambda<1$ we have

$$\Pr\{\tilde{n}_0\leqslant(1-\lambda)sn_1\}\leqslant\exp\left\{-\frac{\lambda^2}{2}sn_1\right\}.$$

Let $\lambda=1-1/s$, the probability of $\{\tilde{n}_0\leqslant n_1\}$(i.e., $\mathcal{B}$ does not hold) can be obtained as

$$\Pr\{\tilde{n}_0\leqslant n_1\}\leqslant\exp\left\{-\frac{(s-1)^2n_1}{2s}\right\}\leqslant\exp\left\{-\frac{(s-1)^2n_1^{\gamma}}{2s}\right\}. \tag{5}$$

It is clear that $n_1 \to \infty$ as $n \to \infty$. Thus, the probability of $\tilde{n}_0 \leqslant n_1$ tends to 0, i.e., $\mathcal{B}$ holds with high probability.

Similar to the method used in the previous section, we truncate the inner part of summation at M. Thus, we have

$$\tilde{T}_j = \binom{\tilde{n}_0}{m_0}^{-1}\binom{n_1}{m_1}^{-1}\sum_{c_S} h_{j,1} + \binom{\tilde{n}_0}{m_0}^{-1}\binom{n_1}{m_1}^{-1}\sum_{c_S} h_{j,2} \overset{def}{=} \tilde{T}_{j,1} + \tilde{T}_{j,2}\,, \tag{6}$$

where c_S denotes all possible combinations satisfying $\eta_{i_l} = 1$ for $l = 1, \ldots, m_0$. The number of such combinations is $\binom{\tilde{n}_0}{m_0}\binom{n_1}{m_1}$, so $\tilde{T}_{j,1}$ in (6) remains a generalized U-statistic.

For the first part $\tilde{T}_{j,1}$, as when using full data, we have

$$\Pr\{|\tilde{T}_{j,1} - \tilde{\theta}_{j,1}| \geqslant \epsilon\} \leqslant 2\exp\left\{-\frac{n_1\epsilon^2}{2m_1M^2}\right\}. \tag{7}$$

Since $\mathcal{B}$ holds with high probability, we can be assured that this operation is feasible.

For the second part $\tilde{T}_{j,2}$, we can also follow a similar approach to show that it satisfies

$$\begin{aligned}\Pr\{|\tilde{T}_{j,2} - \tilde{\theta}_{j,2}| \geqslant \epsilon\} &\leqslant \Pr\{|\tilde{T}_{j,2}| > \epsilon/2\} \\ &\leqslant n \max_{1\leqslant j\leqslant p} \Pr\{|X_j| > \alpha M\} \\ &\leqslant n\tilde{c_1}\exp\{-\tilde{c_2}\alpha M\}.\end{aligned} \tag{8}$$

It's important to note that, since we do not know the sampling method δ_i in advance, we should ensure that all observations (instead of only the observations drawn from subsampling) satisfy $|X_j| < \alpha M$ to guarantee the correctness of our proof.

Combining inequality (5), (7) and (8), we have

$$\Pr\{|\tilde{T}_j - \tilde{\theta}_j| \geqslant 2\epsilon\} \leqslant O(\exp\left\{-\frac{n_1\epsilon^2}{2m_1M^2}\right\} + n\exp\{-\tilde{c_2}\alpha M\} + \exp\left\{-\frac{(s-1)^2n_1^\gamma}{2s}\right\}).$$

Let $M = n_1^\gamma, 0 < \gamma < \frac{1}{2} - \kappa$, the above inequality can be written as

$$\Pr\{|\tilde{T}_j - \tilde{\theta}_j| \geqslant 2\epsilon\} \leqslant O(\exp\left\{-\frac{\epsilon^2 n_1^{1-2\gamma}}{2m_1}\right\} + n\exp\{-\tilde{c_2}\alpha n_1^\gamma\} + \exp\left\{-\frac{(s-1)^2n_1^\gamma}{2s}\right\}). \tag{9}$$

At last, we let $\epsilon = n_1^{-\kappa}/2$, denote $\tilde{C}_1, \tilde{C}_2 > 0$ as constants, and get the following inequality.

$$\Pr\{|\tilde{T}_j - \tilde{\theta}_j| \geqslant n_1^{-\kappa}\} \leqslant O(\exp\{-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\} + n\exp\{-\tilde{C}_2 n_1^\gamma\}).$$

The proof of Theorem 1(1) is complete.

Noting that $\hat{\mathcal{F}} = \{j : |T_j| > \delta\}$. From condition (C4) and the Theorem 1(1) , we know that for any $j \notin \mathcal{F}$, $\tilde{\theta_j} = 0$ and $|\tilde{T_j}| \leqslant n_1^{-\kappa}$ with high probability; for any $j \in \mathcal{F}$, $|\tilde{\theta}_j| \geqslant 2n_1^{-\kappa}$ and $|\tilde{T}_j| \geqslant n_1^{-\kappa}$ with high probability. Specifically, if we let $\delta = n_1^{-\kappa}$, we have

$$\Pr\{j \notin \hat{\mathcal{F}} | j \in \mathcal{F}\} \leqslant O(\exp\{-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\} + n\exp\{-\tilde{C}_2 n_1^\gamma\}).$$

Since we have m true features, the probability that there exists an important feature that is not screened can be expressed as

$$\Pr\{\mathcal{F} \not\subset \hat{\mathcal{F}}\} \leqslant O(p[\exp\{-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\} + n\exp\{-\tilde{C}_2 n_1^\gamma\}]).$$

The proof of Theorem 1(2) is complete.

Furthermore, if we let $\epsilon = \delta/4$, inequality (9) can be written as

$$\Pr\left\{|T_j - \theta_j| > \frac{\delta}{2}\right\} \leqslant O(\exp\left\{-\tilde{C}_3\delta^2 n_1^{1-2\gamma}\right\} + n\exp\{-\tilde{C}_4 n_1^\gamma\}). \tag{10}$$

We define event $\mathcal{A} = \{\max_{1 \leqslant j \leqslant p} |\tilde{T}_j - \tilde{\theta}_j| \leqslant \delta/2\}$. When $\mathcal{A}$ holds, the number of $\{j : |\tilde{T}_j| \geqslant \delta\}$ (i.e. $|\hat{\mathcal{F}}|$) cannot exceed the number of $\{j : |\theta_j| \geqslant \delta/2\}$, which is bounded by $(\delta/2)^{-1} \sum_j |\tilde{\theta}_j|$. Together with inequality (10), we find that the model size (i.e. number of selected feature by our method) is bounded.

$$\Pr\left\{ |\hat{\mathcal{F}}| \leqslant \frac{2}{\delta} \sum_j |\tilde{\theta}_j| \right\} \geqslant \Pr\{\mathcal{A}\} \geqslant 1 - O(p[\exp\left\{-\tilde{C}_3 \delta^2 n_1^{1-2\gamma}\right\} + n \exp\{-\tilde{C}_4 n_1^{\gamma}\}]).$$

The proof of Theorem 1(3) is complete. □

B.4 Preliminaries for Theorem 2

To facilitate the discussion of Theorem 2, for $0 \leqslant a < m_0$ and $0 \leqslant b < m_1$, we define (a, b)-observation projection of h as

$$\begin{aligned} & h_{a,b}\left(\mathbf{X}_{k,1}^{(0)}, \ldots, \mathbf{X}_{k,a}^{(0)}; \mathbf{X}_{k,1}^{(1)}, \ldots, \mathbf{X}_{k,b}^{(1)}\right) \\ = & \mathbb{E}\left\{ h\left(\mathbf{X}_{k,1}^{(0)}, \ldots, \mathbf{X}_{k,m_0}^{(0)}; \mathbf{X}_{k,1}^{(1)}, \ldots, \mathbf{X}_{k,m_1}^{(1)}\right) \,\middle|\, \mathbf{X}_{k,1}^{(0)}, \ldots, \mathbf{X}_{k,a}^{(0)}; \mathbf{X}_{k,1}^{(1)}, \ldots, \mathbf{X}_{k,b}^{(1)} \right\}. \end{aligned}$$

Based on this, we define $\xi_{a,b}$ as the variance of $h_{a,b}$, that is,

$$\xi_{a,b} = \mathrm{Var}\left\{ h_{a,b}\left(\mathbf{X}_{k,1}^{(0)}, \ldots, \mathbf{X}_{k,a}^{(0)}; \mathbf{X}_{k,1}^{(1)}, \ldots, \mathbf{X}_{k,b}^{(1)}\right) \right\}.$$

C Additional discussions on VFS framework

This section provides supplementary discussions extending the main text. We elaborate on three aspects of the VFS framework: (i) its extension to multi-party and multi-class settings; (ii) its conceptual distinction from classical multiple testing and the associated cascading error analysis; and (iii) the necessity of Stage 2 for refined feature selection.

Extension to multi-party and multi-class settings. In the main text, we only considered the simplest case involving two parties and two classes. In fact, VFS can be naturally extended to multi-party and multi-class settings.

First, it is important to note that VFS is performed marginally on individual features and does not involve interactions between features. In the multi-party setting, the active party only needs to encrypt the pairwise label matrix and send the result to each passive party. Ciphertext computations are carried out locally by each passive party, while the active party is only required to perform decryption. This process is identical to that in the two-party scenario.

The extension to multi-class settings is also feasible. Define the screening statistic as

$$T_K = \binom{\tilde{n}_0}{m_0}^{-1} \prod_{j=1}^{K} \binom{n_j}{m_j}^{-1} \sum_c \left(\prod_{l=1}^{m_0} \eta_{i_{0,l}} \right) h\left(X_{i_1^0}^{(0)}, \ldots, X_{i_{m_0}^0}^{(0)}; \ldots; X_{i_1^K}^{(K)}, \ldots, X_{i_{m_K}^K}^{(K)} \right).$$

The asymptotic distribution of T_K can be referred to Corollary 2 in [35]. The cut-off value can be selected accordingly, and then theoretical guarantees in Theorems 1 and 2 hold in the multi-class case. This establishes the theoretical foundation for extending VFS to multi-class settings.

Difference with multiple testing and cascading error analysis. In VFS framework, our focus differs from classical multiple testing, where the goal is typically to control the Type I error. For instance, the Bonferroni correction reduces false positives by using α/m, but at the cost of increased Type II error [16, 20]. In contrast, we fix α regardless of the number of groups to maintain statistical power, thereby focusing on reducing Type II error rather than controlling Type I error. Theoretical guarantees on the Type II error are established in Theorem 1(2). Furthermore, the potential accumulation of cascading errors in Stage 1 can be explicitly quantified. Suppose that each iteration involves

l groups and that a total of r iterations are conducted with $lr \lesssim p$. Under similar conditions as in Theorem 1(2), the probability that any relevant feature is mistakenly removed after Stage 1 satisfies

$$\mathrm{P}\{\mathcal{F} \not\subset \hat{\mathcal{F}}\} \leqslant O\left[lr\left\{\exp\left(-\tilde{C}_1 n_1^{1-2\gamma-2\kappa}\right) + n\exp\left(-\tilde{C}_2 n_1^{\gamma}\right)\right\}\right] \to 0\,,$$

ensuring that errors do not accumulate uncontrollably across iterations.

Necessity of Stage 2 in VFS. Theorem 2 shows that the screening statistic is asymptotically normal when condition (2) is satisfied, which requires that each feature group in Stage 1 contains a sufficiently large number of features (empirically more than five [77]). Consequently, Stage 1 alone is not sufficiently fine-grained, as the retained groups may still contain a considerable number of irrelevant features. This highlights the necessity of Stage 2 for further refinement.

D Definition and asymptotic properties of the VFS statistic adjusted from distance covariance

For simplicity, we omit the subscript j of the VFS statistic $\tilde{T}_j$ here. The VFS statistic used in Section 5 and 6 is defined as

$$\tilde{T}_{\mathrm{dcov}} = \binom{\tilde{n}_0}{2}^{-1}\binom{n_1}{2}^{-1}\sum_{i=1}^{\tilde{n}_0}\sum_{j>i}\sum_{k=1}^{n_1}\sum_{l>k} h_{\mathrm{dcorr}}\left(X_i^{(0)}, X_j^{(0)}; X_k^{(1)}, X_l^{(1)}\right) \tag{11}$$

with kernel function

$$\begin{aligned} h_{\mathrm{dcov}}(X_1^{(0)}, X_2^{(0)}; X_1^{(1)}, X_2^{(1)}) = {} & \left\|X_1^{(0)} - X_1^{(1)}\right\|_2 + \left\|X_1^{(0)} - X_2^{(1)}\right\|_2 + \left\|X_2^{(0)} - X_1^{(1)}\right\|_2 \\ & + \left\|X_2^{(0)} - X_2^{(1)}\right\|_2 - 2\left\|X_1^{(0)} - X_2^{(0)}\right\|_2 - 2\left\|X_1^{(1)} - X_2^{(1)}\right\|_2 . \end{aligned} \tag{12}$$

Furthermore, as a corollary of Theorem 2, we can obtain the asymptotic distribution of $\tilde{T}_{\mathrm{dcov}}$.

Corollary 1 (Asymptotic Normality of Group-based $\tilde{T}_{\mathrm{dcov}}$)**.** *Under the null hypothesis, $n_1/n \to 0$ and condition in* (2)*, the VFS statistic defined in (11) satisfies*

$$n_1\tilde{T}_{\mathrm{dcov}}/\xi_{0,2}^{1/2} \xrightarrow{d} N(0,2)$$

E Additional experimental results

In numerical simulations, the default simulation setting is as follows: sample size $n = 10000$, number of features $p = 2000$, number of true features $m = 20$, initial group size $s_0 = 100$, decay coefficient $\rho = 0.5$ and number of selected features after screening $d = 100$. While keeping other settings fixed, Tables 7 – 10 investigate the effects of total number of features p, number of true features m, initial group size s_0, and number of retained features after screening d, respectively. In these tables, Time reports the computational time for encrypted feature screening in seconds, and Time Ratio compares the average computational time of VFS with that of classical screening. Results from the ablation studies suggest that our method is robust and not sensitive to hyperparameter choices.

For real-world datasets, Table 11 reports the results under different numbers of retained features after screening (i.e., d), which further demonstrates the robustness of VFS in real-world applications.

Table 7: Performance of VFS and classical screening under different p.

p	VFS						Classical Screening						Time
	PSR		FDR		Time		PSR		FDR		Time		Time
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Ratio
Model 1													
1000	0.962	0.950	0.808	0.810	316.2	318.7	0.927	0.950	0.815	0.810	1519.0	1472.6	0.208
2000	0.959	0.950	0.808	0.810	507.5	502.1	0.894	0.900	0.821	0.820	3023.1	3029.5	0.168
5000	0.945	0.950	0.811	0.810	852.3	866.7	0.846	0.850	0.831	0.830	7634.5	7463.1	0.112
Model 2													
1000	0.985	1.000	0.803	0.800	382.6	382.2	0.964	0.950	0.807	0.810	1490.8	1522.5	0.257
2000	0.983	1.000	0.804	0.800	506.8	513.9	0.943	0.950	0.811	0.810	2988.0	3047.6	0.170
5000	0.978	1.000	0.804	0.800	1085.2	1127.2	0.910	0.900	0.818	0.820	7385.2	7534.5	0.147
Model 3													
1000	0.860	0.850	0.828	0.830	470.5	474.8	0.786	0.800	0.843	0.840	1526.4	1491.2	0.308
2000	0.869	0.850	0.826	0.830	530.0	529.1	0.743	0.750	0.852	0.850	3056.6	2997.0	0.173
5000	0.836	0.850	0.833	0.830	825.7	836.5	0.684	0.700	0.863	0.860	7432.0	7356.0	0.111

Table 8: Performance of VFS and classical screening under different m.

m	VFS						Classical Screening						Time
	PSR		FDR		Time		PSR		FDR		Time		Time
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Ratio
Model 1													
10	0.992	1.000	0.901	0.900	363.7	368.3	0.971	1.000	0.903	0.900	3004.6	3032.4	0.121
20	0.957	0.950	0.809	0.810	429.6	436.8	0.896	0.900	0.821	0.820	3019.2	2950.1	0.142
50	0.836	0.840	0.582	0.580	658.6	659.3	0.733	0.720	0.633	0.640	2996.8	2992.1	0.220
Model 2													
10	0.999	1.000	0.900	0.900	500.0	490.1	0.991	1.000	0.901	0.900	3007.0	2999.9	0.166
20	0.981	1.000	0.804	0.800	455.8	459.9	0.940	0.950	0.812	0.810	2970.6	2976.1	0.153
50	0.864	0.860	0.568	0.570	748.9	744.1	0.761	0.760	0.619	0.620	2999.9	2972.9	0.250
Model 3													
10	0.956	1.000	0.904	0.900	406.2	398.7	0.886	0.900	0.911	0.910	2952.8	3039.8	0.138
20	0.858	0.850	0.828	0.830	546.5	543.5	0.738	0.750	0.853	0.850	3058.4	3007.6	0.179
50	0.487	0.480	0.756	0.760	592.9	611.0	0.451	0.460	0.775	0.770	2940.8	2952.0	0.202

Table 9: Performance of VFS and classical screening under different s_0.

s_0	VFS						Classical Screening						Time
	PSR		FDR		Time		PSR		FDR		Time		Time
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Ratio
Model 1													
20	0.946	0.950	0.811	0.810	831.2	841.9	0.891	0.900	0.822	0.820	2978.8	2979.0	0.279
50	0.958	0.950	0.808	0.810	528.8	528.6							0.178
100	0.958	0.950	0.808	0.810	473.7	474.9							0.159
Model 2													
20	0.978	1.000	0.805	0.800	633.3	640.4	0.943	0.950	0.811	0.810	3042.8	2956.6	0.208
50	0.981	1.000	0.804	0.800	623.6	623.2							0.205
100	0.984	1.000	0.803	0.800	522.0	512.1							0.172
Model 3													
20	0.843	0.850	0.832	0.830	665.8	685.6	0.744	0.750	0.851	0.850	3021.2	2982.6	0.220
50	0.852	0.850	0.830	0.830	565.9	562.3							0.187
100	0.858	0.850	0.828	0.830	383.2	389.4							0.127

Table 10: Performance of VFS and classical screening under different d.

d	VFS						Classical Screening						Time
	PSR		FDR		Time		PSR		FDR		Time		Time
	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Ratio
Model 1													
50	0.927	0.950	0.629	0.620	455.6	446.6	0.841	0.850	0.664	0.660	3036.8	2955.6	0.150
100	0.957	0.950	0.809	0.810	486.9	500.7	0.894	0.900	0.821	0.820	2984.3	2984.0	0.163
200	0.968	1.000	0.903	0.900	818.9	832.0	0.933	0.950	0.907	0.905	3020.8	3026.7	0.271
Model 2													
50	0.966	0.950	0.614	0.620	403.1	404.7	0.901	0.900	0.640	0.640	3031.4	3007.3	0.133
100	0.984	1.000	0.803	0.800	583.0	590.3	0.944	0.950	0.811	0.810	3000.9	2982.1	0.194
200	0.985	1.000	0.901	0.900	752.2	736.4	0.965	0.950	0.904	0.905	2978.9	2997.6	0.253
Model 3													
50	0.808	0.800	0.677	0.680	378.3	368.7	0.683	0.700	0.727	0.720	2978.8	3051.6	0.127
100	0.860	0.850	0.828	0.830	424.8	409.0	0.741	0.750	0.852	0.850	3029.6	2979.8	0.140
200	0.877	0.900	0.912	0.910	864.7	873.3	0.799	0.800	0.920	0.920	3040.9	3028.1	0.284

Table 11: Performance of Selection v.s. VFS+Selection. 80% of the data is randomly split for training and 20% for testing. Accuracy and AUC are both evaluated on the test set.

				LESS-VFL				MUSE			
Dataset	Method		Screening Time	Accuracy	AUC	Selection Time	Total Time	Accuracy	AUC	Selection Time	Total Time
Gina	VFS+	100	6.1	0.810	0.885	138.0	144.1	0.823	0.823	5.1	11.2
		200	9.9	0.836	0.915	191.9	201.8	0.842	0.842	11.8	21.8
		500	21.3	0.859	0.933	295.8	317.0	0.843	0.842	31.4	52.6
	Selection	All	-	0.843	0.924	467.0	467.0	0.843	0.842	61.5	61.5
Activity	VFS+	50	3.4	0.999	1.000	325.7	329.1	1.000	1.000	2.2	5.6
		100	4.9	0.999	1.000	469.3	474.2	1.000	1.000	6.4	11.3
		200	8.5	1.000	1.000	670.0	678.5	1.000	1.000	14.6	23.0
	Selection	All	-	0.998	1.000	953.4	953.4	1.000	1.000	44.8	44.8
RNA-Seq	VFS+	200	21.6	0.994	1.000	53.3	74.9	1.000	1.000	9.3	30.9
		1000	17.4	1.000	1.000	123.3	140.7	1.000	1.000	51.1	68.5
		2000	18.0	1.000	1.000	207.1	225.1	1.000	1.000	103.9	121.9
	Selection	All	-	0.944	1.000	1791.1	1791.1	1.000	1.000	1062.3	1062.3
p53	VFS+	200	2.8	0.833	0.775	74.8	77.7	0.783	0.806	8.8	11.6
		500	2.7	0.835	0.802	137.8	140.5	0.808	0.813	24.0	26.7
		1000	3.5	0.835	0.830	245.3	248.8	0.839	0.845	49.6	53.0
	Selection	All	-	0.843	0.859	1493.3	1493.3	0.849	0.862	279.3	279.3