IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 July 2024, accepted 17 July 2024, date of publication 23 July 2024, date of current version 1 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3432509

RESEARCH ARTICLE

Feature Selection With Group-Sparse Stochastic Gates

HYERYN PARK AND CHANGHEE LEE, (Member, IEEE)
Department of Artificial Intelligence, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Changhee Lee (changheelee@cau.ac.kr)

This work was supported in part by the Chung-Ang University (CAU) Research under Grant 2021; and in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) through the Korean Government [Ministry of Science and ICT (MSIT)], Artificial Intelligence (AI) Graduate School Program, CAU, under Grant 2021-0-01341.

ABSTRACT Identifying features significantly influencing the target outcome is crucial for understanding complex relationships, reducing computational costs, and improving model generalization in high-dimensional data. While powerful for discovering intricate relationships, deep learning-based feature selection methods often overlook inherent group structures in data, such as gene pathways or categorical variables. Consequently, these methods may fail to select informative features within the relevant groups, potentially leading to the selection of less informative features and ultimately, lower model performance. To address this challenge, we propose a novel deep learning-based feature selection method that achieves both intra-group and inter-group sparsity. By introducing a penalty term that encourages group sparsity, our method effectively selects informative groups of features, thereby improving model performance. We validate our approach through experiments on synthetic and real-world datasets with predefined group structures. Our method achieved a 2.5 $\sim$ 8.2% reduction in prediction RMSE and a 0.3 $\sim$ 1.9% improvement in prediction accuracy compared to existing methods. Furthermore, our approach demonstrated a 50% increase in the selection of biologically relevant features, enhancing model interpretability and alignment with relevant scientific literature. These results confirm the effectiveness of our method in leveraging group structures to improve both performance and interpretability.

INDEX TERMS Deep learning, embedded feature selection, intra-group sparsity, inter-group sparsity.

I. INTRODUCTION

High-dimensional data is prevalent in various fields, including critical domains like medicine and biology. Identifying an informative subset of features that significantly influence the target outcome is crucial. This not only fosters a deeper understanding of the underlying complex relationships between features [1], [2], [3], but also reduces experimental costs and improves model generalization [4], [5]. Feature selection is a well-established area with proposed solutions like wrapper [6] and filter [7], [8] methods. Deep learning (DL) methods have become a powerful tool for *embedded* feature selection, which selects relevant features while concurrently performing model selection [9], [10], [11], [12], [13]. This is particularly advantageous since DL methods can uncover complex relationships between features and the target outcome, enabling the selection of feature subsets that have higher discriminative or predictive power.

Earlier methods have addressed the non-differentiable objective function in feature selection by approximating it with Lasso [14] or elastic net penalization [10], [15]. More recent work tackles this challenge by employing continuous relaxation, which models feature selection as a process involving binary random variables [12], [13]. Such relaxation makes the objective function differentiable with respect to the distribution parameters that govern the binary random variables, allowing for techniques like REINFORCE [16] or reparameterization tricks for discrete random variables – such as using Concrete [17], Hard Concrete (HC) [18], and Gaussian approximation [12] – to be applied.

However, when data has inherent group structures (e.g., gene pathways in gene expression data or one-hot encoded

The associate editor coordinating the review of this manuscript and approving it for publication was Yiqi Liu.

categorical variables), identifying important groups of features alongside individual features becomes crucial for understanding their relationships with the target outcome. While modifications on Lasso-based regularization have addressed this challenge by augmenting a group-level L_2 penalty to promote inter-group sparsity [19], [20], [21], the above DL-based feature selection methods primarily focus on achieving individual feature-level sparsity (using L_0-penalty), neglecting the underlying group structures in the data. Consequently, these methods may fail to select the most informative features within the most relevant groups, potentially leading to degradation in model performance.

Contribution: We propose a novel DL-based feature selection method that achieves both intra- and inter-group sparsity. To achieve this goal, we introduce a novel penalty term that encourages inter-group sparsity, which is a reformulation of the group-sparsity regularization from group Lasso for the DL-based feature selection. This forces the model to discard entire features in each group, thereby only selecting the relevant feature groups, leading to group sparsity. The biggest distinction from the previous DL-based feature selection methods lies in our emphasis on the importance of features at a group level. We incorporate group information during feature selection, prioritizing the group-level feature importance rather than solely relying on penalizing individual features separately. We achieve such inter-group sparsity by proposing a novel penalty term that minimizes the cumulative distribution function (CDF) of a Poisson binomial random variable [22], which is approximately computed by utilizing a Gaussian-based approach.

Throughout experiments, we validate our approach on multiple synthetic and real-world datasets with predefined group structures. Our model discovers relevant features that provide superior prediction performance compared to the sparse-group Lasso [19], [20], [21] and the state-of-the-art benchmarks that only focus on individual-level sparsity. Furthermore, we confirm the selected features by aligning them with relevant scientific literature.

II. RELATED WORK

A. DEEP LEARNING-BASED EMBEDDED FEATURE SELECTION METHOD

Lasso [14] and its variants [23], [24] have been widely applied in various domains as representative embedded feature selection methods, which identify important features by shrinking the coefficients of less important ones through L_1 regularization. While effective, these methods are limited by their reliance on *linear* models to capture feature interactions with the target outcome. Recent advancements in neural networks have emerged as a powerful alternative, capable of capturing the complex, non-linear relationships often present between features and target variables [10], [25], [26]. However, directly applying the L_1 penalty in the input layer generally makes it challenging to induce sparsity and, thus, hinders the interpretability of which features are truly important.

DL methods for embedded feature selection have addressed the challenges of discrete selection (and thus properly achieve sparsity) by introducing stochastic binary gate vectors and an L_0 penalty [12], [13]. Specifically, this is approximately achieved by utilizing continuous relaxation, which models the selection of each feature as a Bernoulli random variable. The main focus here is to overcome the non-differentiability involved in the sampling process via reparameterization tricks such as Concrete distribution [17], [27], Hard Concrete (HC) distribution [18], and Gaussian approximations with a hard sigmoid function (STG) [12].

Recent advancements in DL have explored implicit group structures for feature selection, such as interactions and correlations among input features. For instance, Lee et al. [13] have enhanced the gating process using Gaussian copula to generate correlated gates, while Imrie et al. [28] have introduced composite features capturing predictive feature subsets with interactions. In multi-label feature selection, authors in [29] and [30] have analyzed label correlations using group structures while controlling feature redundancy. Despite leveraging implicit group structures and considering label-feature correlations to identify important features, none of these methods explicitly achieve group sparsity.

B. FEATURE SELECTION WITH GROUP SPARSITY

In many real-world domains (e.g., biology), features often exhibit natural groupings or relationships, which makes the importance of features within the same group possibly correlated. Ignoring these group structures during feature selection can mislead the understanding of the discovered features (e.g., biomarkers) and potentially lead to performance degradation. To address this issue, researchers have proposed methods that leverage pre-defined group information to promote sparsity at the group level, known as *group sparsity* [31]. These methods, explored in various studies [32], [33], [34], incorporate an L_2 penalty at the group level. This penalty encourages all coefficients corresponding to features within a group to approach zero, effectively achieving group sparsity and improved prediction performance. One notable development of group sparsity is *sparse group Lasso* [19], [20], [21], which simultaneously achieves both group-wise and within group sparsity by leveraging both L_1 and L_2 penalties. This can effectively select the most informative features from the most important groups, potentially improving model performance and interpretability.

While successful with linear models, feature selection methods with group sparsity often struggle to capture the complexities of real-world data. In this work, we address this limitation by leveraging the universal approximation ability of deep neural networks to select crucial features with possibly non-linear interactions, while achieving both intra-group and inter-group sparsity. Table 1 compares objectives and regularization terms used by Lasso-based and DL-based feature selection methods to achieve intra-group and inter-group sparsity. We will provide more details in the following sections.

III. PROBLEM FORMULATION

Let $\mathbf{X} = (X_1, \cdots, X_p) \in \mathbb{R}^p$ and $Y \in \mathcal{Y}$ be random variables for the p-dimensional input feature and the target outcome, whose realizations are denoted as $\mathbf{x} = (x_1, \cdots, x_p)$ and y, respectively. Here, $\mathcal{Y}$ is the outcome space, and we will focus our description on C-class classification tasks, i.e., $\mathcal{Y} = \{1, \ldots, C\}$ for ease of notation. We assume that the feature can be divided into pre-specified G *non-overlapping* groups, i.e., $\mathbf{x} = (\mathbf{x}^1, \mathbf{x}^2, \cdots \mathbf{x}^G)$, where each group is a collection of features that share a certain association (e.g., genes with similar molecular functions or biological roles). We denote the g-th group as $\mathbf{x}^g = (x_1^g, x_2^g, \cdots, x_{p_g}^g) \in \mathbb{R}^{p_g}$, comprising p_g features.

Embedded feature selection is a method of selecting a subset of features, i.e., $\mathcal{S} \subset [p]$, that are relevant for predicting the target outcome during model training. Denote $\mathbf{m} = (m_1, \ldots, m_p) \in \{0, 1\}^p$ be a binary gate vector where m_d indicates whether the d-th feature is selected in $\mathcal{S}$ or not, i.e., $m_d = 1$ if $d \in \mathcal{S}$ and $m_d = 0$ otherwise. Then, we can define the selected feature subset, $\tilde{\mathbf{x}} \in (\mathbb{R} \cup \{*\})^p$, as $\tilde{\mathbf{x}} = \mathbf{m} \odot \mathbf{x} + (1 - \mathbf{m}) \odot *$, where $*$ denotes any point not in $\mathbb{R}$. Let $f_\theta : (\mathbb{R} \cup *)^p \to \Delta^{C-1}$ be a function (a neural network parameterized by θ) that takes the feature subset $\tilde{\mathbf{x}}$ as input and outputs a point in the $(C-1)$-simplex for C-class classification.

We aim for the selected feature subset to exhibit both inter-group sparsity – i.e., the selection of only a few groups of features – and intra-group sparsity – i.e., the selection of only a few features within each group. To achieve this goal, we rewrite the gate vector as $\mathbf{m} = (\mathbf{m}^1, \mathbf{m}^2, \cdots, \mathbf{m}^G)$, utilizing the group information, and define the embedded feature selection problem as solving the following objective:

$$\begin{aligned} &\underset{\theta, \mathbf{m}}{\text{minimize}}\ \mathbb{E}_{\mathbf{x},y}[\ell_Y(y, f_\theta(\tilde{\mathbf{x}})] \\ &\text{subject to } \|\mathbf{m}\|_0 \le \delta \ \text{ and } \ \sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{1}\big(\|\mathbf{m}_g\|_0 > 0\big) \le \delta_g \quad (1) \end{aligned}$$

where $\ell_Y(y, \hat{y}) = \|y - \hat{y}\|^2$ is the mean squared error (MSE) loss for regression tasks and $\ell_Y(y, \hat{y}) = -\sum_{c=1}^{C} y_c \log \hat{y}_c$ is the cross-entropy loss for classification tasks.[1] Here, δ and δ_g control the size of the selected feature subset and the (scaled) number of groups with nonzero gates, respectively.

IV. METHOD

Unfortunately, the combinatorial problem in (1) becomes intractable for high-dimensional data as the search space increases exponentially with p. Hence, we employ a continuous relaxation by assuming the gate vector is generated from a *Bernoulli distribution* governed by $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_p) \in [0, 1]^p$, i.e., $\mathbf{m} \sim \texttt{Bern}(\boldsymbol{\pi})$, which transforms the combinatorial search into a search over a unit hypercube [12], [13]. Throughout, we will use $\mathbf{M} = (\mathbf{M}^1, \mathbf{M}^2, \ldots, \mathbf{M}^G)$ to represent $\mathbf{m}$ in the form of a random variable.

[1]Here, we slightly abuse the notation and write y_c to present the c-th element of the one-hot encoding of y.

Then, based on the Bernoulli relaxation, we can reformulate our objective in (1) using Lagrangian approximation as:

$$\underset{\theta, \boldsymbol{\pi}}{\text{minimize}}\ \mathbb{E}_{\mathbf{x},y}\mathbb{E}_{\mathbf{m}}\left[\ell_Y(y, f_\theta(\tilde{\mathbf{x}})) + \lambda_1 \mathcal{R}_1 + \lambda_2 \mathcal{R}_2\right] \quad (2)$$

where λ_1 and λ_2 are Lagrangian multipliers that control the trade-offs between the prediction loss and the regularizers. Here, $\mathcal{R}_1 = \sum_{d=1}^{p} \mathbb{1}(m_d = 1)$ corresponds to the L_1 penalty in Lasso, which encourages to close the gate at an individual level achieving intra-group sparsity, and $\mathcal{R}_2 = \sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{1}\Big(\sum_{d=1}^{p_g} m_d^g > 0\Big)$ represents the L_2 penalty in group-Lasso, which achieves inter-group sparsity by forcing none of the gates in each group to be open. Hence, optimizing the expected regularization terms, i.e., $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_1]$ and $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2]$, is crucial to control the intra- and inter-group sparsity of the selected feature subset.

To this goal, we propose a novel **D**oubly **S**parse **F**eature **S**election, which we refer to as **DSFS**, which aims to select feature subsets that maintain both intra- and inter-group sparsity. In this section, we begin by simplifying the expected regularization terms for intra- and inter-group sparsity in (2) as functions of the Bernoulli parameters governing with the gate vectors, and then introduce the reparameterization trick employed to overcome the non-differentiability involved in the selection process.

A. INTRA-GROUP SPARSITY: L_1 PENALTY

For the first regularization term, $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_1]$, we can rewrite the expected number of open gates as follows:

$$\begin{aligned} \mathbb{E}_{\mathbf{m}}[\mathcal{R}_1] &= \mathbb{E}_{\mathbf{m}}\left[\sum_{d=1}^{p} \mathbb{1}(m_d = 1)\right] = \sum_{d=1}^{p} \mathbb{E}_{\mathbf{m}}[\mathbb{1}(m_d = 1)] \\ &= \sum_{d=1}^{p} \mathbb{P}(M_d = 1) = \sum_{d=1}^{p} \pi_d. \end{aligned} \quad (3)$$

This shows that achieving the intra-group sparsity simply boils down to penalizing the sum of probabilities that each gate to be open, i.e., $\sum_{d=1}^{p} \pi_d$.

B. INTER-GROUP SPARSITY: L_2 PENALTY

For the second regularization term, $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2]$, we can derive the expected number of groups with at least one nonzero gate as the following:

$$\begin{aligned} \mathbb{E}_{\mathbf{m}}[\mathcal{R}_2] &= \mathbb{E}_{\mathbf{m}}\left[\sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{1}\left(\sum_{d=1}^{p_g} m_d^g > 0\right)\right] \\ &= \sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{E}_{\mathbf{m}}\left[\mathbb{1}\left(\sum_{d=1}^{p_g} m_d^g > 0\right)\right] \\ &= \sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{P}\big(B^g > 0\big). \end{aligned} \quad (4)$$

Here, $B^g = \sum_{d=1}^{p_g} M_d^g$ is a *Poisson binomial* random variable constructed as the sum of Bernoulli variables that are not

necessarily identically distributed, i.e., $\pi_1^g = \cdots = \pi_{p_g}^g$ is not necessarily true [35]. Thus, the inter-group sparsity can be achieved by reducing the sum of probabilities that the Poisson binomial random variable for each group is larger than zero.

Denote $Q_{B^g}(k) = \mathbb{P}(B^g \leq k)$ for $k \in \{0, 1, \ldots, p_g\}$ be the cumulative distribution function (CDF) for the Poisson binomial random variable, B^g, parameterized by $\boldsymbol{\pi}^g$. Then, we can rewrite $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2]$ in (4) as

$$\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2] = \sum_{g=1}^{G} \sqrt{p_g}\, \mathbb{P}\big(B^g > 0\big) = \sum_{g=1}^{G} \sqrt{p_g}\, (1 - Q_{B^g}(0)) \tag{5}$$

where $Q_{B^g}(k)$ for $k \in \{0, 1, \ldots, p_g\}$ can be formally derived as follows [22]:

$$Q_{B^g}(k) = \sum_{d=0}^{k} \sum_{\mathcal{A} \in \mathcal{F}_d} \prod_{j \in \mathcal{A}} \pi_j \prod_{j \in \mathcal{A}^c} (1 - \pi_j). \tag{6}$$

Here, $\mathcal{F}_d$ is a set containing all subsets of $[d]$ where $d \in 1, 2, 3, \cdots, p_g$, and $\mathcal{A}^c$ is the complement of set $\mathcal{A}$.

Computing the original CDF in (6) requires calculating combinations for all possible subsets, leading to an exponential increase in complexity as the number of features per group grows. This has motivated numerous studies to explore closed-form expressions that efficiently approximate (6), such as employing discrete Fourier transform [36] based on the characteristic function defined in [37] and applying a Gaussian approximation based on the central limit theorem or its refined variant [38]. Please refer to Appendix B for detailed information on these approximations.

In this work, we take the Gaussian approximation in [38], as it offers comparable effectiveness in feature selection while requiring the least computational time compared to other approximation techniques. Applying the central limit theorem with a continuous correction, we can approximate the CDF of the Poisson binomial variable, B^g, as the following:

$$Q_{B^g}(k) \approx \Phi\left(\frac{k + 0.5 - \mu^g}{\sigma^g}\right) \tag{7}$$

where μ^g is the mean and σ^g is the standard deviation of the Poisson binomial B^g given as

$$\mu^g = \mathbb{E}[B^g] = \sum_{d=1}^{P_g} \pi_d, \quad \sigma^g = \mathbb{V}[B^g]^{\frac{1}{2}} = \left(\sum_{d=1}^{P_g} \pi_d(1 - \pi_d)\right)^{\frac{1}{2}}.$$

However, directly employing (7) as the L_2 penalty is not favorable for inter-group sparsity. More specifically, when a group is large (i.e., p_g is large), the corresponding μ^g and σ^g also increase. This causes the gradient of the penalty term to vanish, unintentionally favoring smaller groups based more on their size rather than their actual importance.

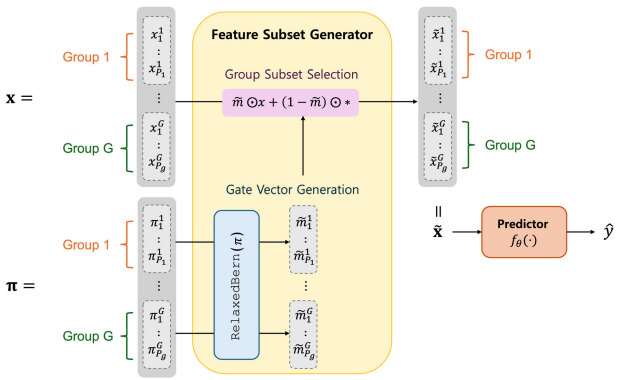

FIGURE 1. An illustration of the feature selection process of DSFS.

Hence, we define the L_2 penalty by placing the normalization term, $\sqrt{p_g}$, inside the Gaussian CDF in (8), as follows:

$$\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2] \approx \sum_{g=1}^{G} \Phi\left(\frac{\mu^g - 0.5}{\sigma^g \sqrt{p_g}}\right). \tag{8}$$

This ensures equitable penalization for different group sizes and allows us to focus on group importance.

C. REPARAMETERIZATION TRICK

The optimization problem in (2) includes a sampling process, i.e., $\mathbf{m} \sim \texttt{Bern}(\boldsymbol{\pi})$, whose non-deterministic aspect renders classical gradient-based optimization method inapplicable. To circumvent this challenge, we apply continuous relaxation on the gate vector, $\mathbf{m}$, via a HC [18], a widely employed reparameterization trick for Bernoulli variables. Formally, given the uniform random variables $\mathbf{u} = (u_1, \ldots, u_p)$ where $u_d \sim \texttt{Uniform}(0, 1)$, we define the relaxed gate vector, $\tilde{\mathbf{m}} = (\tilde{m}_1, \tilde{m}_2, \ldots, \tilde{m}_p) \in [0, 1]^p$, as the following:

$$\begin{aligned} s_d &= \sigma\left(\frac{1}{\beta}\left(\log\frac{\pi_d}{1 - \pi_d} + \log\frac{u_d}{1 - u_d}\right)\right)(\varsigma - \gamma) + \gamma \\ \tilde{m}_d &= \min(0, \max(1, s_d)) \end{aligned} \tag{9}$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is a sigmoid function and β is the temperature that controls the degree of approximation. Here, we stretch the concrete distribution onto an interval (γ, ς) with $\gamma < 0$ and $\varsigma > 1$, followed by a hard sigmoid. Please find the schematic illustration of the relaxed sampling procedure employed for our feature selection method in Fig 1.

Overall, under the reparameterization trick, we can rewrite our objective as follows:

$$\begin{aligned} \underset{\theta, \boldsymbol{\pi}}{\text{minimize}}\, \mathbb{E}_{\mathbf{x},y}\Big[& \mathbb{E}_{\mathbf{u}}[\ell_Y(y, f_\theta(\tilde{\mathbf{m}} \odot \mathbf{x} + (1 - \tilde{\mathbf{m}}) \odot *))] \\ & + \lambda_1 \sum_{d=1}^{p} \pi_d + \lambda_2 \sum_{g=1}^{G} \Phi\left(\frac{\mu^g - 0.5}{\sigma^g \sqrt{p_g}}\right)\Big]. \end{aligned} \tag{10}$$

V. EXPERIMENT

In this section, we evaluate our proposed method for embedded feature selection with intra- and inter-group sparse

using various synthetic and real-world datasets specifically chosen to reflect group information.

Benchmarks: We compare DSFS with various feature selection methods ranging from machine learning to state-of-the-art DL methods: linear models with L_1 penalty (**Lasso**) [31] and with both intra- and inter-group sparsity (**Group Lasso**) [20], tree-based ensemble models that can provide feature importance (**RForest**) [39] and (**XGBoost**) [40], DL-based feature selection method using stochastic gates with individual-level sparsity[2] (**STG**) [12], using correlated gates with Gaussian copula[3] (**SEFS**) [13] and selecting predictive subsets with interactions in composite features[4] (**CompFS**) [28] and extension of sparse-group Lasso [20] into a DL framework that applies sparsity regularization at both the feature level and the group level (**DeepGL**).

Performance Metrics: For synthetic experiments where the ground-truth important features are available, we propose a metric to measure the group sparsity given as the following:

$$GS(\tilde{\mathbf{m}}) = \frac{\sum_{g=1}^{G} \|\tilde{\mathbf{m}}^g\|_2}{\|\tilde{\mathbf{m}}\|_2}. \tag{11}$$

This implies the L_2 norm of the relaxed mask vector of each group to that of the entire features, allowing us to compare the group sparsity of different methods. For instance, when the same number of features are selected, the one selecting features from fewer groups will have a lower value compared to those selecting features from a larger number of groups. (Hence, when all the features are selected from a single group, the group sparsity will have 1, which is its lowest value.)

However, in real-world experiments, the ground-truth important features are typically unknown. We therefore indirectly assess the performance of the selected features by training a separate model (here, we use RForest to capture non-linearity) based on the feature subsets selected by the feature selection methods. Here, we derive $\mathbf{m}$ from the relaxed Bernoulli distribution as $m_d = \mathbb{1}(\pi_d \geq \delta)$ where δ is a threshold that determines the number of features to be selected. (Note that a similar process is also applied for the benchmarks.) We then evaluate the prediction performance given different feature subsets using metrics appropriate for the task type. For classification, we use accuracy (ACC), the area under the receiver operating characteristic curve (AUROC), the area under the precision-recall curve (AUPRC), and F1 score. For regression, we utilize root mean squared error (RMSE), mean absolute error (MAE), and R^2 score.

Implementation Details: We implement DSFS using a multi-layer perceptron (MLP) network with the rectified linear unit (ReLU) as the non-linear activation function. To isolate the effect of network architecture and ensure a fair comparison of the regularization terms, we implement both STG and DSFS using a 3-layer multi-layer perceptron (MLP) with 100 hidden nodes and the rectified linear unit (ReLU) as non-linear activation functions. This provides sufficient expressivity to distinguish them from their linear counterparts (i.e., Lasso and Group Lasso, respectively) while maintaining focus on the regularization terms. Please refer to Appendix C-D for more details. SEFS is modified to incorporate group information into the correlation structure for generating correlated gate vectors. To ensure a fair comparison with our model, we excluded the self-supervision phase and utilized only the supervision phase. As CompFS does not directly support the integration of group information, we have adhered to the original settings from the referenced paper. DeepGL extends the sparse-group Lasso by incorporating sparsity regularization at both the feature and group levels. Feature-level sparsity is achieved through L_2 regularization, which applies penalties to the squared weights associated with individual features. This reduces the contribution of less important features by shrinking their weights towards zero, indicating their reduced importance in predicting the target. Group-level sparsity involves applying L_2 regularization to the combined weights of feature groups. This encourages sparsity at the group level, potentially shrinking the entire set of weights associated with less important groups of features towards zero. Unlike advanced feature selection methods, DeepGL does not directly perform feature selection as it focus on shrinking the weights not removing, it selects features based on their importance scores, similar to pruning [41], [42]. We choose λ_1 and λ_2 for DeepGL and DSFS (λ_1 only for STG, SEFS and CompFS) via a grid search from possible candidates $\{0, 0.5, 1.0 \cdots 10.0\}$ and $\{0, 5, 10, \cdots 100\}$, based on the validation performance that balances both intra- and inter-group sparsity with the predictive power of the model.

[2] https://github.com/runopti/stg

[3] https://github.com/chl8856/SEFS

[4] https://github.com/a-norcliffe/Composite-Feature-Selection

A. SYNTHETIC EXPERIMENTS

1) DATASET DESCRIPTION

We consider synthetic experiments in which the ground truth for both feature and group importance is available. This involves constructing 5 different synthetic scenarios, which serve as variations of the data generation process initially introduced for evaluating group Lasso [43]. More specifically, we first generate p-dimensional input features each of which is generated from a standard normal distribution, i.e., $\mathbf{x} \sim \mathcal{N}(0, \mathbf{I})$. Then, we split the input features into G pre-specified groups with possibly different sizes. To consider cases where the target outcome is influenced by only a subset of feature groups, we set features from this subset, denoted as $\mathbf{x}_{\mathcal{S}} = (\mathbf{x}_i)_{i \in \mathcal{S}}$ where $\mathcal{S} \subset [G]$, as relevant features. Then, the target outcome is generated as a linear combination of these relevant features, as $y = \sum_{g \in \mathcal{S}} a_g \mathbf{1}^\top \mathbf{x}_g + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 0.5^2)$ is a random noise and $a_g \in \mathcal{A}$ is a coefficient randomly chosen from a set of possible values $\mathcal{A}$. For different synthetic scenarios, we generate 10,000 samples with varying input dimensions, pre-specified group

TABLE 1. Comparison of intra- and inter-group sparsity between Lasso-based linear models and DL-based models.

	Lasso-based Feature Selection		DL-based Feature Selection	
Objective	$\underset{\beta}{\text{minimize}}\ \mathbb{E}_{\mathbf{x},y}\left[\ell_Y(y,\beta^T\mathbf{x})+\lambda_1\mathcal{R}_1+\lambda_2\mathcal{R}_2\right]$		$\underset{\theta,\mathbf{m}}{\text{minimize}}\ \mathbb{E}_{\mathbf{x},\mathbf{y},\mathbf{m}}\left[\ell_Y(y,f_\theta(\hat{\mathbf{x}}))+\lambda_1\mathcal{R}_1+\lambda_2\mathcal{R}_2\right]$	
Intra-group	$\mathcal{R}_1=\|\beta\|_1$	$\mathcal{R}_2=0$	$\mathcal{R}_1=\|m\|_0$	$\mathcal{R}_2=0$
Inter-group	$\mathcal{R}_1=0$	$\mathcal{R}_2=\sum_{g=1}^{G}\sqrt{p_g}\|\beta^g\|_2$	$\mathcal{R}_1=0$	$\mathcal{R}_2=\sum_{g=1}^{G}\sqrt{p_g}\,\mathbb{1}(\sum_{d=1}^{p_g}m_d^g>0)$
Intra- and inter-group	$\mathcal{R}_1=\|\beta\|_1$	$\mathcal{R}_2=\sum_{g=1}^{G}\sqrt{p_g}\|\beta^g\|_2$	$\mathcal{R}_1=\|m\|_0$	$\mathcal{R}_2=\sum_{g=1}^{G}\sqrt{p_g}\,\mathbb{1}(\sum_{d=1}^{p_g}m_d^g>0)$

TABLE 2. Data generation process for synthetic experiments.

Scenario	p	p_g	G	$\mathcal{S}$	$\mathcal{A}$	Noise
Synthetic A	735	$10\sim20$	50	random	$\{1\}$	
Synthetic B	735	$10\sim20$	50	random	$\{1,2,3\}$	✓
Synthetic C	120	20	6	$\{3,4,5\}$	$\{0.2,1\}$	
Synthetic D	120	20	6	$\{3,4,5\}$	$\{0.2,0.5,1\}$	
Synthetic E	100	10	10	$\{2,5,7,9\}$	$\{1\}$	✓

TABLE 3. Performance comparison for Synthetic A & B.

	Synthetic A				Synthetic B			
No.	No. Group		*GS*		No. Group		*GS*	
Feature	STG	DSFS	STG	DSFS	STG	DSFS	STG	DSFS
735	50		7.04		50		7.04	
400	47	**25**	6.38	**4.98**	38	**26**	6.08	**5.09**
300	34	**18**	5.41	**4.24**	33	**9**	5.43	**3.00**
150	33	**9**	5.47	**2.30**	21	**8**	4.40	**2.80**
30	19	**2**	4.20	**1.41**	17	**2**	4.03	**1.41**
10	10	**2**	3.16	**1.41**	9	**1**	3.00	**1.00**

TABLE 4. Performance comparison for Synthetic C & D.

	Synthetic C				Synthetic D			
No.	No. Group		*GS*		No. Group		*GS*	
Feature	STG	DSFS	STG	DSFS	STG	DSFS	STG	DSFS
120	6		2.450		6		2.450	
60	6	**3**	2.20	**1.73**	6	**3**	2.28	**1.73**
40	3	**2**	1.71	**1.41**	4	**3**	1.97	**1.73**
20	3	**1**	1.64	**1.00**	4	**1**	1.89	**1.00**

TABLE 5. Performance comparison for Synthetic E.

	Synthetic E			
No.	No. Group		*GS*	
Feature	STG	DSFS	STG	DSFS
100	10		3.162	
15	5	**4**	2.19	**1.99**
10	4	**2**	1.97	**1.38**
5	3	**1**	1.71	**1.00**

information, and the subset of important feature groups, as specified in Table 2.

- **Synthetic A**: The relevant feature groups, $\mathcal{S}$, are randomly chosen through coin flipping, and the coefficients for these relevant features are set to 1.
- **Synthetic B**: This further complicates Synthetic A by introducing multiple levels of importance represented by $\mathcal{A}=\{1,2,3\}$, with additive noise from $\mathcal{N}(0,0.5^2)$.
- **Synthetic C**: This scenario has two levels of group importance, denoted as $\mathcal{A}=\{0.2,1\}$, to verify whether a model can select important feature groups even for groups with small coefficients.
- **Synthetic D**: This further complicates Synthetic D by adding another level of group importance, denoted as $\mathcal{A}=\{0.2,0.5,1\}$, with additive noise $\mathcal{N}(0,0.5^2)$.
- **Synthetic E**: This scenario satisfies both intra- and inter-group sparsity, by randomly setting the coefficient of 5 to 9 features in each group to zero.

2) QUANTITATIVE AND QUALITATIVE RESULTS

Tables 3, 4, and 5 compare DSFS with STG based on the number of groups, to which features selected by each method belong, and *GS* score, which measures the group sparsity, both given that the same number of features are selected for both methods. To emphasize the importance of inter-group sparsity, we focused on comparing our method with STG, which only considers intra-group sparsity. For DSFS, the number of selected groups decreases with the decreasing number of selected features (achieved by increasing the regularization coefficients), indicating its focus on capturing group-level signals. In contrast, STG selects features scattered across different groups, resulting in a larger number of selected groups and higher GS scores, as it promotes sparsity only at the feature level. For example, for Synthetic A and B (in Table 3), where STG and DSFS select 30 important features, the number of groups selected is significantly different, and DSFS achieves much lower *GS* score than STG. For Synthetic C and D (in Table 4), when the coefficients of the relevant features are different, DSFS selects features in groups with small signals, whereas STG fails to select any feature with small coefficients with large penalties. For Synthetic E (in Table 5), which contains both intra- and inter-group sparsity, STG selects features across multiple groups, while DSFS focuses on specific groups with strong signals, achieving the inter-group sparsity. The detailed results of the selected features are provided in Appendix C-B.

B. REAL-WORLD EXPERIMENTS: GAS SENSOR DATASET

1) DATASET DESCRIPTION

We use a specific subset (denoted as 'batch10') from the Gas Sensor array dataset [44] to focus on the scenario with the same group size. The selected 'batch10' contains

TABLE 6. Performance comparison of the selected features by different methods.

Methods	GAS (RMSE | MAE | R^2 SCORE)						PBMC (AUROC | AUPRC | ACC | F1 SCORE)							
	$\|\|S\|\|$=20			$\|\|S\|\|$=80			$\|\|S\|\|$=50				$\|\|S\|\|$=100			
Lasso	41.0	23.9	0.942	34.7	19.4	0.972	0.947	0.900	0.968	0.922	0.939	0.881	**0.969**	0.925
RForest	41.6	24.9	0.948	34.5	18.4	0.973	0.939	0.877	0.962	0.918	0.936	0.876	0.959	0.909
XGBoost	40.7	23.2	0.957	34.6	18.8	0.972	0.911	0.838	0.957	0.910	0.906	0.839	0.947	0.904
Group Lasso	40.8	23.2	0.960	33.8	17.8	0.979	0.949	0.895	0.970	0.925	0.940	0.882	**0.969**	0.929
STG	39.9	22.1	0.966	32.9	16.9	0.982	0.954	0.901	0.971	0.935	**0.943**	0.891	0.968	0.928
SEFS	39.2	21.7	0.967	33.3	17.2	0.976	0.952	0.900	0.972	0.940	**0.943**	0.894	**0.969**	**0.932**
CompFS	40.3	22.9	0.962	32.7	16.7	**0.985**	0.952	0.897	0.971	0.939	0.942	0.877	**0.969**	0.925
DeepGL	41.6	24.8	0.947	34.2	18.5	0.969	0.948	0.892	0.968	0.929	**0.943**	0.894	**0.969**	0.931
DSFS	**38.2**	**21.0**	**0.969**	**32.4**	**16.6**	**0.985**	**0.956**	**0.912**	**0.975**	**0.947**	**0.943**	**0.895**	**0.969**	0.931

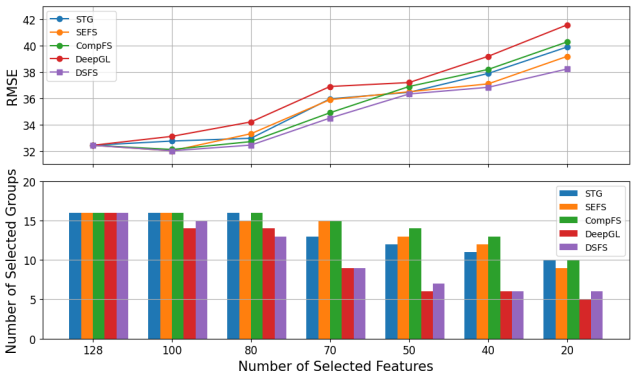

FIGURE 2. Comparison of the number of selected groups and RMSE between DSFS and other benchmarks for the Gas Sensor dataset.

TABLE 7. Comparison between STG and DSFS with varying the number of selected features for the PBMC dataset.

No. Feature	No. Group		ACC		AUROC		AUPRC	
	STG	DSFS	STG	DSFS	STG	DSFS	STG	DSFS
3346	24		0.967		0.946		0.887	
2000	24	24	0.967	0.967	0.935	0.937	0.886	0.890
1500	24	20	0.962	0.964	0.926	0.929	0.870	0.879
1000	24	19	0.964	0.966	0.932	0.936	0.875	0.885
800	24	19	0.960	0.967	0.928	0.936	0.868	0.887
500	24	17	0.964	0.969	0.929	0.940	0.878	0.897
300	24	16	0.963	0.964	0.935	0.929	0.874	0.880
200	24	16	0.964	0.967	0.931	0.938	0.880	0.891
100	21	13	0.968	0.969	0.943	0.943	0.891	0.895
50	19	9	0.971	0.975	0.954	0.956	0.901	0.912
30	15	7	0.972	0.973	0.955	0.955	0.902	0.910

3,600 instances with 128 features extracted from 16 chemical sensors exposed to 6 different gases at various concentration levels (600 instances for each class). We treat a set of features from the same sensor as a group, resulting in a total of $G = 16$ groups, each with a group size of 8, i.e., $p^g = 8$. Here, we focus on a regression task of predicting the gas concentration level as in [45].

2) QUANTITATIVE RESULTS

In Fig 2, we compare the tendency of the number of groups and features selected by DSFS and other DL models as we increase the trade-off coefficients, i.e., both L_1 and L_2 penalties for DeepGL and DSFS and the L_1 penalty for others. Since the proposed method consists of two trade-off coefficients that affect both intra- and inter-group sparsity, we explore various combinations to identify the configuration that best balances intra- and inter-group sparsity. More specifically, we first perform a grid search over possible candidate values to identify the best λ_1 and λ_2, individually, by setting the other coefficient to 0. Then, we conduct a grid search on an array of 10 linearly interpolated values from $(\lambda_1, 0)$ and $(0, \lambda_2)$.

To ensure a fair comparison, we evaluate all the methods when they select the same number of features. Our proposed method tends to select the fewest groups given the same number of selected features, achieving inter-group sparsity. Additionally, the RMSE performance improvement over other comparison targets becomes more significant when selecting a smaller number of features with a strong penalty. This suggests that employing inter-group sparsity, which in turn leads to selecting fewer sensor types, appears to be beneficial for predicting overall gas concentrations.

Furthermore, we compare the performance of feature subsets selected by different benchmarks in Table 6. We observe that the RMSE, MAE, and R^2 score of the RForest trained on the features selected by DL models outperform those trained on features selected by the statistical and ensemble-based methods. This highlights the potential advantage of DL models in capturing complex relationships within the data. Interestingly, both Lasso and STG underperform compared to their group-sparsity counterparts, Group Lasso and DSFS, respectively. This suggests that incorporating group-level sparsity as an inductive bias can be beneficial for feature selection. Specifically, direct integration of a group sparsity penalty proves advantageous when datasets exhibit clear group structures, as it contrasts with indirect methods of managing feature correlations. Additionally, while DeepGL significantly contributes to group-level sparsity, its distinct L_2-based regularization mechanism, which differs from the DL-based feature selection methods, appears to have led to performance degradation.

C. REAL-WORLD EXPERIMENTS: PBMC DATASET

1) DATASET DESCRIPTION

Peripheral Blood Mononuclear Cells (PBMCs) [46], [47] are a type of blood cell with a round nucleus, and they

TABLE 8. The top 10 genes with high relevance scores among the selected 30 features by DSFS and STG, respectively.

Ensembel Gene ID	Chromosome Map	STG	DSFS	Relevance Score
ENSG00000188822	1		✓	0.89
ENSG00000000938	1		✓	6.35
ENSG00000117281	1	✓	✓	2.76
ENSG00000026751	1		✓	1.11
ENSG00000188404	1		✓	1.78
ENSG00000153563	2	✓	✓	**7.78**
ENSG00000113088	5	✓	✓	1.12
ENSG00000145649	5	✓	✓	0.81
ENSG00000150787	11	✓		1.03
ENSG00000270647	17	✓	✓	0.89

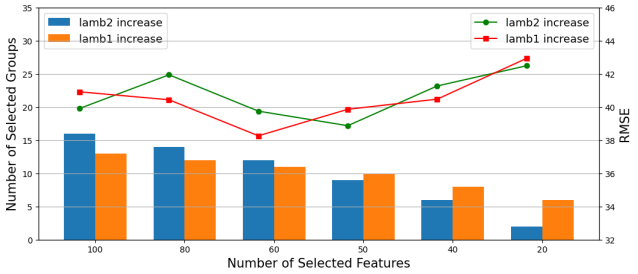

FIGURE 3. Comparison of the number of selected groups and RMSE based on the regularizer coefficients for the Gas Sensor dataset.

include lymphocytes (T cells, B cells, and natural killer cells) and monocytes. PBMCs can be used in various research applications to study immune responses, investigate diseases, and develop therapeutic strategies. We focus on classifying two types of T lymphocytes (i.e., $\mathcal{Y} = \{0, 1\}$), namely the CD4 and CD8 T-cells which are white blood cells that play crucial roles in the immune system whose relative proportions are pivotal for medical disease classification and immune status assessment [48]. For instance, in viral infections like HIV/AIDS, a decline in CD4 T-cell count indicates disease progression [49], while the balance of CD4 and CD8 T-cells in allergic and immune-related diseases serves as indicators of health status and treatment efficacy. The dataset consists of 11, 990 samples described by $p = 3346$ genes. We map ensemble gene IDs to chromosomal locations to group genes from the same chromosome together,[5] resulting in a total of $G = 24$ groups (22 autosomes and 2 sex chromosomes), each with a group size p_g that varies widely, ranging from 3 to 360.

2) QUANTITATIVE RESULTS

In Table 7, we compare STG and DSFS based on the classification performance – i.e., accuracy, AUROC, and AUPRC – averaged over the 5-fold cross-validation, while varying the number of selected features. Similar to the previous results, we observe that the proposed method achieves higher group-level sparsity compared to STG and provides better classification performance, especially in AUPRC, in most cases. This highlights the importance of inducing the group-level structure for feature selection. Furthermore, Table 6 shows a similar tendency when we compare the classification performance of RForest trained on the feature subsets selected by DSFS with those selected by the benchmarks. Particularly, the proposed method outperforms the benchmarks in most cases or achieves similar performance, highlighting the effectiveness of utilizing both DL models and inducing inter-group sparsity.

3) QUALITATIVE RESULTS

We further provide supporting evidence for the importance of the selected features based on the GeneCards database,[6] which provides comprehensive, user-friendly information on the annotated and predicted human genes. In Table 8, we present the top 10 genes based on relevance scores among the 30 features selected by STG and DSFS. Notably, our method identifies 9 out of the top 10 genes with the highest relevance scores as important features, while STG selects only 6. This quantitative superiority underscores our method's ability to prioritize biologically meaningful features, crucial for interpreting population-level complex relationships in a given dataset.

For example, our method identifies genes like Lck and ENSG00000000938 (FGR) with high relevance scores, highlighting their significant roles in cellular processes such as CD8 T cell activation and chromosome group interactions [49], [50]. The association of Lck with CD8 T cells, in particular, emphasizes its relevance in distinguishing cell types, a critical aspect often overlooked by conventional methods. Furthermore, our analysis consistently demonstrates that genes selected by our proposed method exhibit higher relevance scores than those selected by STG. This not only validates our approach but also highlights its capacity to uncover key biological insights that might be obscured by traditional feature selection methods. In conclusion, our method not only improves model performance but also provides a deeper understanding of the underlying biological mechanisms through interpretable feature selection. By prioritizing groups of features with high relevance scores, we ensure that our model effectively captures essential biological signals, contributing to more meaningful and actionable insights. Please refer to Appendix C for more details.

D. SENSITIVITY ANALYSIS

In this section, we show how the prediction performance changes by varying λ_1 and λ_2 that trade-off the impact of the two regularization terms, $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_1]$ and $\mathbb{E}_{\mathbf{m}}[\mathcal{R}_2]$, respectively. Fig 3 compares the RMSE performance and the number of selected groups with the same number of selected features, achieved by increasing λ_2 while fixing λ_1, and vice versa. Notably, we can observe that the number of selected groups significantly decreases by increasing λ_2 achieving the inter-group sparsity, while increasing λ_1 has less impact on the number of selected feature groups.

[5] https://biodbnet-abcc.ncifcrf.gov/db/db2db.php
[6] https://www.genecards.org

VI. LIMITATIONS AND FUTURE WORK

While we propose a novel DL-based feature selection method that can achieve intra- and inter-group sparsity, simultaneously, DSFS requires pre-specified group information, limiting its applicability to datasets where such information is readily available. (Please see Appendix C-C for experiments with incorrect group information.) Additionally, like all methods for selecting features or assessing feature importance from observational data, DSFS relies on the assumption that the identified feature subsets are sufficient for achieving good predictive power. Hence, all identified features should undergo additional evaluation or verification by domain experts before deployment in practical applications.

There are multiple promising directions to explore in group-sparse regularization as future work: Firstly, incorporating group sparsity into DL models shows promise for enhancing prediction performance, particularly in fields like biology. For instance, gene selection analysis of cancer data using priors of overlapping groups, e.g., biologically meaningful gene sets, can be a promising field of research as promoting group sparsity can help uncover important genes and gene sets. Secondly, expanding pruning techniques based on our group-sparsity regularization can offer opportunities to enhance network pruning performance. By acknowledging the group-level impact of weights within a network, group-sparse pruning techniques can be developed to improve efficiency.

VII. CONCLUSION

In this paper, we propose a DL-based feature selection method that leverages group-sparse stochastic gates. This method achieves both intra-group and inter-group sparsity by reformulating the corresponding constraints as learnable penalty terms. We demonstrate the effectiveness of our approach by evaluating its performance on synthetic data, particularly when datasets exhibit significant group patterns. Our experiments on two real-world datasets with group structures validate that DSFS identifies features with superior discriminative/predictive power, which are further corroborated by supporting scientific literature.

APPENDIX A
DEFINITIONS OF THE ABBREVIATIONS

The abbreviations used in this work are listed in Table 9.

TABLE 9. Definitions of the abbreviations.

	Definitions of the abbreviations
DSFS	Doubly Sparse Feature Selection, which aims to select feature subsets that maintain both intra- and inter-group sparsity
CDF	Cumulative Distribution Function
HC	Hard Concrete
STG	Gaussian approximations with a hard sigmoid function model (Stochastic Gate)
ACC	accuracy
AUROC	area under the receiver operating characteristic curve
AUPRC	area under the precision-recall curve
RMSE	root mean squared error
MAE	mean absolute error
GS	proposed metric to measure the group sparsity

APPENDIX B
COMPUTING THE CDF FOR THE POISSON BINOMIAL DISTRIBUTION

A. METHODS FOR COMPUTING THE CDF

In this section, we introduce other approaches to computing the exact or approximate CDF of the Poisson binomial distribution. The first approach detailed in [36] and [37] derives a closed-form expression of the CDF by applying the discrete Fourier transform (DFT) as the following:

$$\begin{aligned} Q_{B^g}(k) &= \frac{1}{p_g+1}\sum_{d=0}^{p_g}\sum_{m=0}^{k}\exp(-iwdm)x_d \\ &= \frac{1}{p_g+1}\sum_{d=0}^{p_g}\frac{\big(1-\exp(-iwd(k+1))\big)x_d}{1-\exp(-iwd)} \end{aligned} \tag{12}$$

where $i=\sqrt{-1}$, $w=2\pi/(p_g+1)$, and $\exp(-iwdm)$ with $m=0,1,\cdots,k$ is a geometric sequence.

Next, we introduce two well-known approaches to approximate the CDF of the Poisson binomial distribution leveraging the first (μ^g), second (σ^g), and third (γ^g) moments defined as the following:

$$\begin{aligned} \mu^g &= \mathbb{E}[B^g] = \sum_{d=1}^{P_g}\pi_d \\ \sigma^g &= \mathbb{V}[B^g]^{\frac{1}{2}} = \left(\sum_{d=1}^{P_g}\pi_d(1-\pi_d)\right)^{\frac{1}{2}} \\ \gamma^g &= \mathbb{E}\left[\left(\frac{B^g-\mu^g}{\sigma^g}\right)^3\right] = \frac{1}{(\sigma^g)^3}\sum_{d=1}^{p_g}\pi_d(1-\pi_d)(1-2\pi_d). \end{aligned} \tag{13}$$

The Poisson approximation (PA) [38] employs the Poisson binomial distribution to approximate the CDF of B^g, as shown in the following equation:

$$Q_{B^g}(k) \approx \sum_{m=0}^{k}\frac{(\mu^g)^k\exp(-\mu^g)}{k!}. \tag{14}$$

However, this approximation can become increasingly inaccurate as μ^g becomes large.

The last approximation is refined normal approximation (RNA), which addresses the skewness in the distribution of B^g by incorporating a correction on the Gaussian approximation used in (7), as the following equation:

$$Q_{B^g}(k) \approx G^g\left(\frac{k+0.5-\mu^g}{\sigma^g}\right). \tag{15}$$

TABLE 10. Performance comparison between Poisson binomial distribution methods.

Method	Exact	DFT	PA	GA	RNA
Sec/Epoch	2.595	7.019	1.970	**1.968**	2.787
No. Feature	72	70	71	69	69
No. Group	11	11	11	11	11
RMSE	40.4	40.0	**39.4**	39.7	39.7

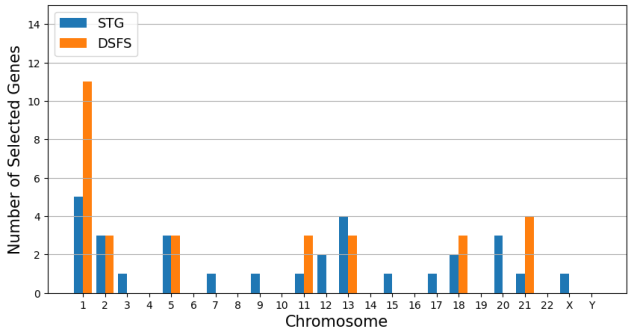

FIGURE 4. The number of selected genes in each chromosome for ours and STG, respectively.

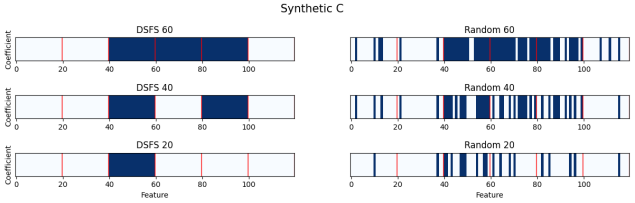

FIGURE 5. Comparison of the features selected with and without group-level information, for Synthetic C.

Here, $G^g(x) = \Phi(x) + \gamma^g(1 - x^2)\,\phi(x)/6$, where $\phi(x)$ is the probability density function of the standard normal distribution, and γ^g is defined in (13).

B. PERFORMANCE COMPARISON

Now, we compare the above methods for computing the exact or approximate CDF of the Poisson binomial distributions to evaluate the best computing method for achieving inter-group sparsity. We have carefully designed this regularization term with computational efficiency in mind. Crucially, its complexity scales linearly with the number of gates (or features) in the dataset. This linear scaling effectively avoids the exponential increase in computation that would have resulted from directly calculating the gradient of the CDF for the Poisson binomial distribution, which we employ to model the group-wise behavior of the gates.

In Table 10, we show the computation time (seconds per epoch), the number of selected features, the number of groups that the selected features belong to, and the RMSE performance on the Gas Sensor dataset, obtained by setting $\lambda_1 = 2$ and $\lambda_2 = 30$ for a fair comparison. To emphasize the computational efficiency of our approach, we directly compares the actual and approximate computation times

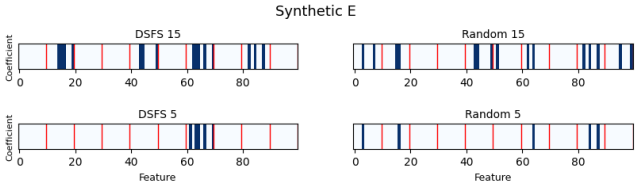

FIGURE 6. Comparison of the features selected with and without group-level information, for Synthetic E.

TABLE 11. The top 30 genes selected by DSFS and STG, respectively.

Ensembel Gene ID	Chromosome Map	STG	DSFS	Relevance Score
ENSG00000142634	1		✓	0.26
ENSG00000142676	1	✓	✓	0.18
ENSG00000188822	1		✓	0.89
ENSG00000000938	1		✓	6.35
ENSG00000142937	1	✓	✓	0.18
ENSG00000155366	1	✓	✓	0.49
ENSG00000117281	1	✓	✓	2.76
ENSG00000177954	1		✓	0.32
ENSG00000026751	1		✓	1.11
ENSG00000188404	1		✓	1.78
ENSG00000116667	1	✓	✓	-
ENSG00000034510	2	✓	✓	-
ENSG00000153563	2	✓	✓	**7.78**
ENSG00000158050	2	✓	✓	0.35
ENSG00000144713	3	✓		0.53
ENSG00000113088	5	✓	✓	1.12
ENSG00000145649	5	✓	✓	0.81
ENSG00000186468	5	✓	✓	0.53
ENSG00000136213	7	✓		-
ENSG00000186106	8	✓		-
ENSG00000180817	10	✓		-
ENSG00000172732	11	✓	✓	-
ENSG00000150787	11	✓		1.03
ENSG00000150687	11	✓	✓	-
ENSG00000110367	11	✓	✓	-
ENSG00000089693	12	✓	✓	0.15
ENSG00000166523	12	✓	✓	0.18
ENSG00000139187	12	✓	✓	0.26
ENSG00000135441	12	✓		-
ENSG00000136305	14	✓		-
ENSG00000072864	16	✓		0.26
ENSG00000270647	17	✓	✓	0.89
ENSG00000198933	17	✓		0.18
ENSG00000002834	17		✓	0.40
ENSG00000141753	17		✓	-
ENSG00000099804	19		✓	0.67
ENSG00000071626	19	✓	✓	-
ENSG00000186111	19	✓	✓	0.41
ENSG00000105319	19	✓	✓	-
ENSG00000101439	20	✓		0.18
ENSG00000128309	22	✓		-

associated with the Poisson binomial distribution. For instance, Poisson-based or Gaussian-based approximations (denoted as PA and GA, respectively) significantly reduce computation time compared to the exact computations using the original definition in (6) and the DFT (12). Interestingly, the exact computation in (6) is faster than the DFT-based and RNA-based methods as the number of features in each group is relatively small. The number of selected features and groups, and the RMSE performance are similar across all the evaluated computation methods, all achieving desired inter-group sparsity. Consequently, we adopt the

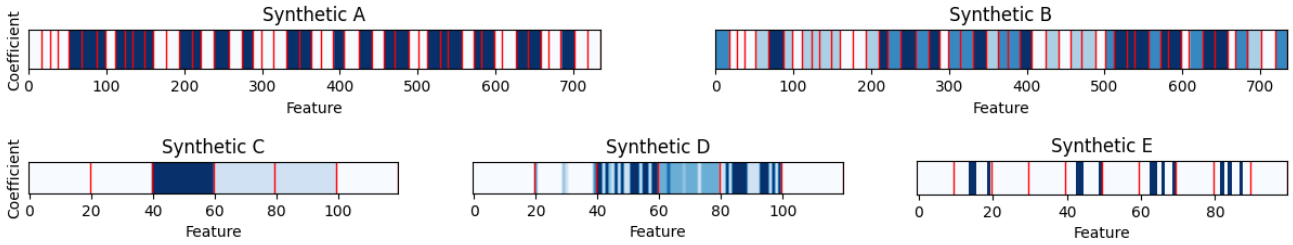

FIGURE 7. Visualization of Synthetic datasets' coefficient.

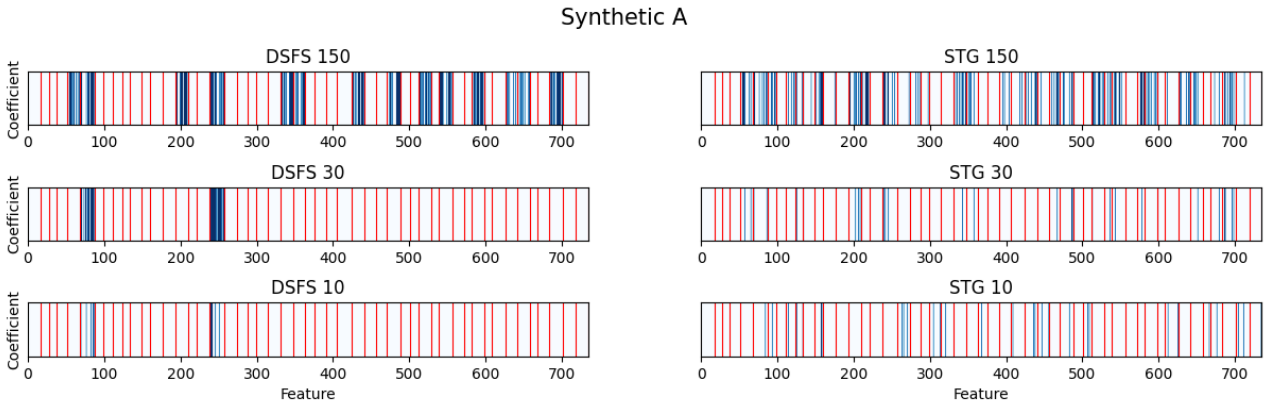

FIGURE 8. Comparison of the features selected by DSFS and STG, respectively, for Synthetic A.

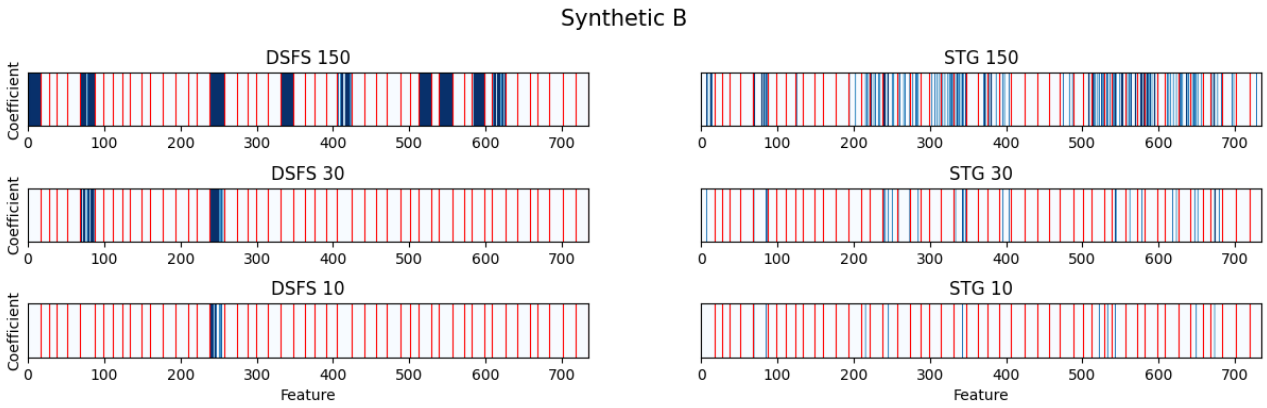

FIGURE 9. Comparison of the features selected by DSFS and STG, respectively, for Synthetic B.

Gaussian-based approximation for modeling the group-wise sum of corresponding gate vectors, enabling our method to effectively handle datasets with larger numbers of features.

APPENDIX C
ADDITIONAL EXPERIMENTS

A. REAL-WORLD EXPERIMENTS: PBMC DATASET

In this subsection, we further compare the selected features (i.e., genes) from DSFS and those from STG based on the supporting bioinformatic references. Here, we use GeneCards database[7] that provides comprehensive, user-friendly information on all annotated and predicted human genes.

In Table 11, we list the top 30 genes selected by each method along with their corresponding relevance score from GeneCards, which quantifies the functional relevance of each gene. Among these genes, 19 are selected in common (indicated by ✓) and the remaining genes differ between the two methods, which potentially explains the observed

[7] https://www.genecards.org

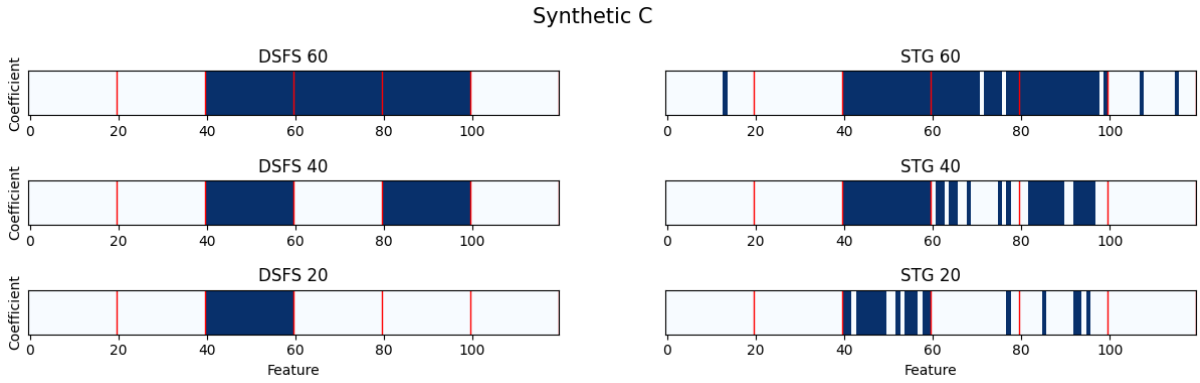

FIGURE 10. Comparison of the features selected by DSFS and STG, respectively, for Synthetic C.

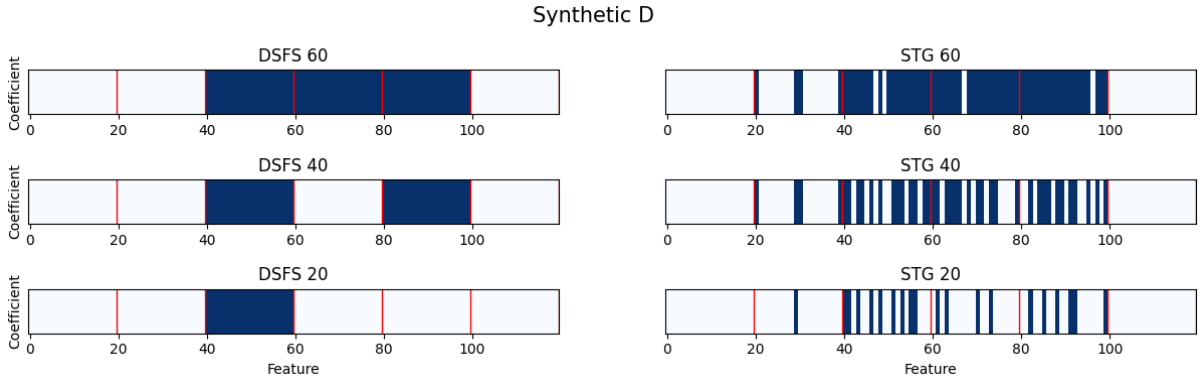

FIGURE 11. Comparison of the features selected by DSFS and STG, respectively, for Synthetic D.

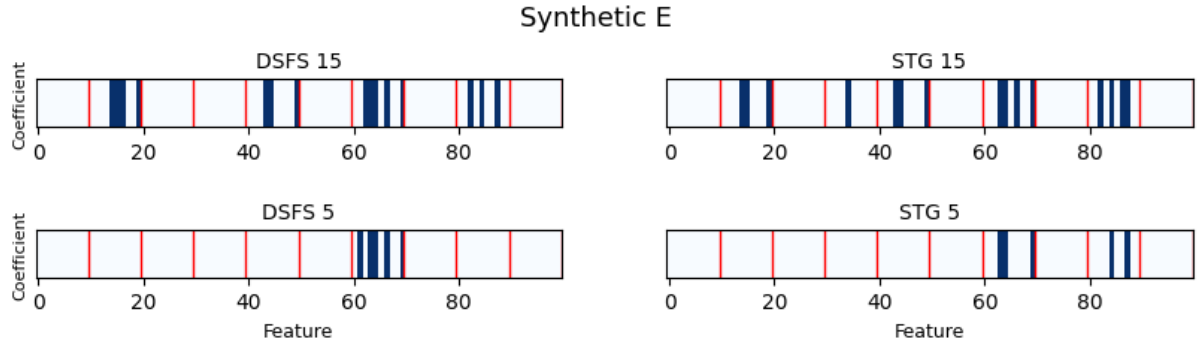

FIGURE 12. Comparison of the features selected by DSFS and STG, respectively, for Synthetic E.

performance gain of DSFS. Particularly, Fig 4 shows that DSFS achieves higher group sparsity by selecting genes from only 7 groups (here, chromosomes), whereas STG selects genes from 15 groups. Overall, genes selected by our method exhibit higher average relevance scores compared to STG, highlighting our model's ability to focus on both intra- and inter-group sparsity.

B. SYNTHETIC EXPERIMENTS

In this subsection, we visually compare the same number of features selected by DSFS and STG on the five different synthetic scenarios in Figures 8 – 12. The heatmap shows $\boldsymbol{\pi}$ for DSFS (left) and truncated $\boldsymbol{\mu}$ for STG (right), respectively. Fig 7 shows the coefficient values (higher the darker) of different synthetic scenarios, where red boxes indicate the groups of features.

In Synthetic A and B, our proposed method exhibits group-based feature selection: With a relatively small penalty, it prioritizes inter-group sparsity, effectively selecting most of the relevant groups. With a relatively stronger penalty, our method promotes intra-group sparsity, allowing DSFS to identify the most important features within those groups.

TABLE 12. Performance comparison of the selected features by different layer and hidden nodes.

	2, 50	2, 100	2, 200	2, 500	3, 50	3, 100	3, 200	3, 500	4, 50	4, 100	4, 200	4, 500
RMSE	36	29	30.1	30.1	35.9	28.8	29.7	31.7	39.1	34.0	34.0	34.0
MAE	21.1	16.5	16.7	16.7	19.5	15.3	16.8	17.0	21.6	20.6	20.6	20.6
R^2 score	0.971	0.981	0.979	0.979	0.971	0.982	0.980	0.979	0.967	0.968	0.968	0.968

Synthetic C and D showcase a focus on a specific group of highly important features. Notably, in Synthetic D with individual noise randomization, STG selects features outside the main group. This highlights the importance of group sparsity in achieving robustness in feature selection. Our method, by considering group structure, is less susceptible to noise outside the relevant groups and tends to identify features closer to the true coefficients.

C. IMPACT OF INCORRECT GROUP INFORMATION

To see the impact of incorrectly provided group information on the selected features, we compare the results of DSFS when trained with correct group information versus that trained with randomly assigned group information on the synthetic dataset (i.e., Synthetic C and E). In Fig 5 and Fig 6, random grouping introduces unexpected sparsity between groups, leading to the selection of incorrect features unrelated to genuinely important ones. This opens the future scope of our work on integrating intra- and inter-group sparsity while jointly learning the group structure from the data, particularly when group information is absent.

D. HYPERPARAMETER TUNING FOR THE MLP ARCHITECTURE

We designed our experiments to assess the impact of our proposed group sparsity regularization on feature selection. Hence, to isolate the effect of network architecture and ensure a fair comparison, we implemented deep learning models using 3-layer MLPs with 100 hidden nodes and ReLU activation layers. This provides sufficient expressivity to distinguish them from their linear counterparts (i.e., Lasso and Group Lasso, respectively) while maintaining focus on the regularization terms. We have conducted hyperparameter tuning for the MLP architecture through a grid search where the potential set of candidates for the number of layers is {2, 3, 4} and for the number of hidden nodes is {50, 100, 200, 500} based on the validation performance. In Table 12, we show the performance comparison of the selected features by different layer and hidden nodes of MLP architecture. We have fixed λ_1, λ_2 and other parameters except for the number of layers and the number of hidden nodes. As seen Table 12, MLPs show mostly similar performance across varying numbers of hidden layers and nodes, particularly when sufficient expressivity is ensured. Hence, we have decided to use a 3-layer MLP with 100 nodes as our baseline network architecture.

REFERENCES

[1] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper," in *Proc. 12th Int. FLAIRS Conf.*, 1999, pp. 235–239.
[2] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, Feb. 1997.
[3] M. Jackson, L. Marks, G. H. May, and J. B. Wilson, "The genetic basis of disease," *Essays biochemistry*, vol. 62, no. 5, pp. 643–723, 2018.
[4] F. Min, Q. Hu, and W. Zhu, "Feature selection with test cost constraint," *Int. J. Approx. Reasoning*, vol. 55, no. 1, pp. 167–179, Jan. 2014.
[5] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?': Explaining the predictions of any classifier," in *Proc. 22th ACM SICKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 1–29.
[6] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, Dec. 1997.
[7] H. Liu and R. Setiono, "A probabilistic approach to feature selection-a filter solution," in *Proc. 13th Int. Conf. Mach. Learn.*, 1996, pp. 1–26.
[8] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Machine Learning Proceedings*. Amsterdam, The Netherlands: Elsevier, 1992, pp. 249–256.
[9] D. Roy, K. S. R. Murty, and C. K. Mohan, "Feature selection using deep neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–6.
[10] Y. Li, C.-Y. Chen, and W. W. Wasserman, "Deep feature selection: Theory and application to identify enhancers and promoters," *J. Comput. Biol.*, vol. 23, no. 5, pp. 322–336, May 2016.
[11] A. Mirzaei, V. Pourahmadi, M. Soltani, and H. Sheikhzadeh, "Deep feature selection using a teacher–student network," *Neurocomputing*, vol. 383, pp. 396–408, Mar. 2020.
[12] Y. Yamada, O. Lindenbaum, S. Negahban, and Y. Kluger, "Feature selection using stochastic gates," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 1–27.
[13] C. Lee, F. Imrie, and M. van der Schaar, "Self-supervision enhanced feature selection with correlated gates," in *Proc. 10th Int. Conf. Learn. Represent.*, 2022, pp. 1–11.
[14] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[15] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 67, no. 2, pp. 301–320, Apr. 2005.
[16] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, May 1992.
[17] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *Proc. 5th Int. Conf. Learn. Represent.*, 2017, pp. 1–14.
[18] C.-T. Huang, J.-C. Chen, and J.-L. Wu, "Learning sparse neural networks through mixture-distributed regularization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1–10.
[19] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group lasso and a sparse group lasso," 2010, *arXiv:1001.0736*.
[20] S. Noah, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. Comput. Graph. Stat.*, vol. 22, no. 2, pp. 231–245, 2013.
[21] J. Li, M. Chang, Q. Gao, X. Song, and Z. Gao, "Lung cancer classification and gene selection by combining affinity propagation clustering and sparse group lasso," *Current Bioinf.*, vol. 15, no. 7, pp. 703–712, Dec. 2020.
[22] S. M. Samuels, "On the number of successes in independent trials," *Ann. Math. Statist.*, vol. 36, no. 4, pp. 1272–1278, Aug. 1965.
[23] P. Zhao and B. Yu, "On model selection consistency of lasso," *J. Mach. Learn. Res.*, vol. 7, pp. 2541–2563, Nov. 2006.
[24] N. Meinshausen and P. Buhlmann, "High-dimensional graphs and variable selection with the lasso," *Ann. Statist.*, vol. 34, no. 3, pp. 1–29, Jun. 2006.

[25] S. Scardapane, D. Comminiello, A. Hussain, and A. Uncini, "Group sparse regularization for deep neural networks," *Neurocomputing*, vol. 241, pp. 81–89, Jun. 2017.
[26] R. Ma, J. Miao, L. Niu, and P. Zhang, "Transformed $\ell 1$ regularization for learning sparse deep neural networks," *Neural Netw.*, vol. 119, pp. 286–298, Nov. 2019.
[27] E. Jang, S. Gu, and B. Poole, "Categorical reparametrization with gumble-softmax," in *Proc. 5th Int. Conf. Learn. Represent.*, 2017, pp. 1–49.
[28] F. Imrie, A. Norcliffe, P. Liò, and M. van der Schaar, "Composite feature selection using deep ensembles," in *Proc. 36th Conf. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 36142–36160.
[29] Y. Fan, J. Liu, J. Tang, P. Liu, Y. Lin, and Y. Du, "Learning correlation information for multi-label feature selection," *Pattern Recognit.*, vol. 145, Jan. 2024, Art. no. 109899.
[30] T. Yin, H. Chen, J. Wan, P. Zhang, S.-J. Horng, and T. Li, "Exploiting feature multi-correlations for multilabel feature selection in robust multi-neighborhood fuzzy β covering space," *Inf. Fusion*, vol. 104, Apr. 2024, Art. no. 102150.
[31] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 68, no. 1, pp. 49–67, Feb. 2006.
[32] Y. Nardi and A. Rinaldo, "On the asymptotic properties of the group lasso estimator for linear models," *Electron. J. Statist.*, vol. 2, no. 1, pp. 1–36, Jan. 2008.
[33] F. R. Bach, "Consistency of the group lasso and multiple kernel learning," *J. Mach. Learn. Res.*, vol. 9, no. 6, pp. 1–39, 2008.
[34] J. Huang and T. Zhang, "The benefit of group sparsity," *Ann. Statist.*, vol. 38, no. 4, pp. 1978–2004, Aug. 2010.
[35] W. Hoeffding, "On the distribution of the number of successes in independent trials," *Ann. Math. Statist.*, vol. 27, no. 3, pp. 713–721, Sep. 1956.
[36] M. Fernandez and S. Williams, "Closed-form expression for the Poisson-binomial probability density function," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 2, pp. 803–817, Apr. 2010.
[37] K. B. Athreya and S. N. Lahiri, *Measure Theory and Probability Theory*, vol. 19. Springer, 2006, pp. 317–337.
[38] Y. Hong, "On computing the distribution function for the Poisson binomial distribution," *Comput. Statist. Data Anal.*, vol. 59, pp. 41–51, Mar. 2013.
[39] T. K. Ho, "Random decision forests," in *Proc. 3rd Int. Conf. Document Anal. Recognit.*, 1995, pp. 278–282.
[40] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1–30.
[41] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the value of network pruning," in *Proc. 7th Int. Conf. Learn. Represent.*, 2019, pp. 1–38.
[42] E. Diao, G. Wang, J. Zhan, Y. Yang, J. Ding, and V. Tarokh, "Pruning deep neural networks from a sparsity perspective," in *Proc. 11th Int. Conf. Learn. Represent.*, 2023, pp. 1–22.
[43] S. Xiang, X. Tong, and J. Ye, "Efficient sparse group feature selection via nonconvex optimization," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 1–7.
[44] A. Vergara, "Gas sensor array drift dataset at different concentrations," UCI Mach. Learn. Repository, 2013, doi: 10.24432/C5MK6M.
[45] C. Du, C. Du, S. Zhe, A. Luo, Q. He, and G. Long, "Bayesian group feature selection for support vector learning machines," in *Proc. 20th Conf. Adv. Knowl. Discovery Data Mining*, 2016, pp. 1–45.
[46] G. X. Zheng et al., "Massively parallel digital transcriptional profiling of single cells," *Nature Commun.*, vol. 8, no. 1, p. 14049, 2017.
[47] A. Gayoso et al., "A Python library for probabilistic analysis of single-cell omics data," *Nature Biotechnol.*, vol. 40, no. 2, pp. 163–166, Feb. 2022.
[48] H. Streeck, J. S. Jolin, Y. Qi, B. Yassine-Diab, R. C. Johnson, D. S. Kwon, M. M. Addo, C. Brumme, J.-P. Routy, S. Little, H. K. Jessen, A. D. Kelleher, F. M. Hecht, R.-P. Sekaly, E. S. Rosenberg, B. D. Walker, M. Carrington, and M. Altfeld, "Human immunodeficiency virus type 1-Specific CD8+T-Cell responses during primary infection are major determinants of the viral set point and loss of CD4+T cells," *J. Virology*, vol. 83, no. 15, pp. 7641–7648, Aug. 2009.
[49] M. Krogsgaard, D. Moogk, S. Zhong, W. Rittase, V. Fang, J. Dougherty, A. Perez-Garcia, I. Osman, C. Zhu, N. Varadarajan, N. P. Restifo, and A. B. Frey, "Constitutive LcK activity drives sensitivity differences between CD8+ memory T cell subsets," *J. Immunology*, vol. 196, no. 1, p. 13332, May 2016.
[50] V. Horkova, A. Drobek, D. Paprckova, V. Niederlova, A. Prasai, V. Uleri, D. Glatzova, M. Kraller, M. Cesnekova, S. Janusova, E. Salyova, O. Tsyklauri, T. A. Kadlecek, K. Krizova, R. Platzer, K. Schober, D. H. Busch, A. Weiss, J. B. Huppa, and O. Stepanek, "Unique roles of co-receptor-bound LCK in helper and cytotoxic T cells," *Nature Immunology*, vol. 24, no. 1, pp. 174–185, Jan. 2023.

HYERYN PARK received the B.S. degree in IT media engineering from Duksung Women's University, Seoul, South Korea, in 2023. She is currently pursuing the M.S. degree in artificial intelligence with Chung-Ang University, Seoul. Her research interests include deep learning-based feature selection, explainable AI for time series, domain adaptation, and survival analysis.

CHANGHEE LEE (Member, IEEE) received the B.S. and M.S. degrees from Korea University, Seoul, South Korea, in 2007 and 2011, respectively, and the Ph.D. degree from the University of California, Los Angeles, USA, in 2021. He has been an Assistant Professor with the Department of Artificial Intelligence, Chung-Ang University, Seoul, since 2021. His research interests include integrating multiple modalities, building advanced time-series models and causal inference models, discovering scientific knowledge from data, and interpreting "black-box" machine learning methods.