Weighted General Group Lasso for Gene Selection in Cancer Classification

Yadi Wang, Xiaoping Li, *Senior Member, IEEE,* and Rubén Ruiz

Abstract—Relevant gene selection is crucial for analyzing cancer gene expression data in cancer classification. Intrinsic interactions among selected genes cannot be fully identified by most existing gene selection methods. In this paper, we propose a Weighted General Group Lasso (WGGL) model to select cancer genes in groups. A gene grouping heuristic method is presented based on weighted gene co-expression network analysis. To determine the importance of genes and groups, a method for calculating gene and group weights is presented in terms of joint mutual information. To implement the complex calculation process of WGGL a gene selection algorithm is developed. Experimental results on both random and three cancer gene expression datasets demonstrate that the proposed model achieves better classification performance than two existing state-of-the-art gene selection methods.

Index Terms—Gene selection, Cancer classification, Group Lasso, Heuristic, Joint mutual information.

1 Introduction

In cancer prevention, diagnosis and treatment, gene selection and prediction accuracy for cancer types are essential for cancer classification. Microarray data has been verified as being useful in classifying many cancers. Successfully identifying gene biomarkers is crucial in predicting the correct class type for a given tumor sample and improving the accuracy of a prediction [1]–[4]. The big challenge for gene selection in cancer classification lies in that there are a huge number of genes and a small number of samples in microarray gene expression data. From a biological perspective, only a small subset of genes is strongly indicative of a targeted disease. In other words, most genes are irrelevant to cancer classification which results in noise and a decrease in the accuracy of classification. From a machine learning perspective, having too many genes always leads to overfitting and a negative influence on classification. Therefore, gene selection methods with high prediction accuracy are desirable for effective cancer classification.

Gene grouping is also paramount for gene selection. A complex biological process, e.g., detecting lung cancer or a brain tumor, not only involves detecting single but also interactions between genes within a subset of genes (or components). Each component can be represented by a graph (e.g., gene regulation, protein interaction) in which the relevant genes (components) are connected. These components are potential targets for the administration of medication and are helpful in disclosing biological processes relevant to metastasis. Even though many approaches have been developed for gene selection of microarray data based on groups in recent years, few of them are biologically based. From the viewpoint of biology, an ideal group contains all genes in a gene pathway. However, detecting gene pathways in complex biological processes is difficult. Fortunately, biological pathways can be mapped to network modules [28] in complex biological processes. Functional gene modules can be detected by the gene co-expression network method [23] which has been increasingly used to explore the system-level functionality of genes. Though this method can identify susceptive genes in complex diseases, their biological meanings are always unclear with gene co-expression encoded by binary information. Zhang et al. [24] proposed the weighted gene co-expression network analysis (WGCNA) method to convert gene co-expression similarity measures into network connection strengths. WGCNA finds modules (or clusters) of highly correlated genes using the hierarchical average linkage clustering method [25] and it has been applied to a variety of biological environments [26], [27]. Therefore, it is possible to identify gene pathways according to the identified network modules and further to group genes.

Adaptively identifying important groups and important genes in a specific group is another challenge for gene selection. Some adaptive shrinkage methods have been proposed to achieve adaptively grouped gene selection by constructing weight coefficients for groups and genes based on statistics which depend on the actual gene expression values of cancer microarray data. Some genes irrelevant to biological processes would be selected because the weight coefficients are usually not biologically related. Moreover, some genes irrelevant to cancer classification might be selected which results in reducing classifier performance because the weight coefficients are very much sensitive to noise or outlier of the dataset. The importance of a group depends on both the genes and their interactions. Though the mutual information technique has been applied to feature selection [29]–[33], interactions between features were ignored which led to unstable selected biomarkers.

- *Yadi Wang and Xiaoping Li are with the School of Computer Science and Engineering, Southeast University, Nanjing, China, 211189; and also at the Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, 211189, Nanjing, China. E-mail: yadiwang@seu.edu.cn, xpli@seu.edu.cn.*
- *Rubén Ruiz is with the Grupo de Sistemas de Optimización Aplicada, Instituto Tecnológico de Informática, Ciudad Politécnica de la Innovación, Edifico 8G, Acc. B. Universitat Politècnica de València, Camino de Vera s/n, 46021, València, Spain. E-mail:rruiz@eio.upv.es.*

Joint mutual information [34]–[36] has been employed in gene selection for microarray gene expression data and it performs well in classification accuracy and stability [37]. On the other hand, as the joint mutual information depends only on the probability distribution of a random variable and not on its actual values, it is more effective to assess the importance of genes and groups. It is desirable to use joint mutual information to evaluate groups and genes.

The main contributions of this paper are summarized as follows:

- A new weighted general group Lasso WGGL model is developed for gene selection in cancer classification.
- Based on the weighted gene co-expression network analysis, a gene grouping heuristic GGH is proposed which groups genes according to pathways.
- A gene and group weight calculation GGWC is presented to determine Intra- and Inter-group weights in terms of joint mutual information in and between groups.
- Based on GGH and GGWC, a method is investigated for the developed WGGL model.

The remainder of the paper is organized as follows. Section 2 reviews the previous related work. The problem under study is described in Section 3. Section 4 constructs the weighted general group Lasso model and the corresponding algorithm is presented in Section 5. Section 6 gives the experimental results followed by conclusions in Section 7.

2 Related Work

Traditionally, genes are selected independently by statistical learning methods. Among these methods, type-2 fuzzy logic [4], toward integrating feature selection algorithms [5], a general hybrid adaptive ensemble learning framework [6] and SVM (Support Vector Machine) and its extensions [2], [7], [8] have been widely used for gene selection in cancer classification. By introducing different penalty strategies, new sparse models have been constructed to select genes more effectively. By using the L_1 norm penalty in regression, Lasso [9] and its extensions [10]–[14] have been applied to sparse gene selection. By using a Bayesian regularization term, sparse logistic regression [15] and sparse multinomial logistic regression [16] have been developed. Though these methods have been successfully applied to gene selection in cancer classification, they could not exploit the interaction information among genes.

Biologically speaking, complex diseases, such as cancer and heart disease have many causes, including mutations in gene pathways. The ideal gene selection method should be able to eliminate the unimportant genes and automatically include the highly correlated genes in groups. The group Lasso [17] has been proposed for selecting the highly correlated and relevant variables in groups rather than individual derived variables which allows for more accurate prediction. Meier et al. [18] extended it to logistic group Lasso. Although the group Lasso and its extension give a sparse set of groups, they do not measure sparsity within a group. Later Simon et al. [19] proposed the sparse group Lasso which yields both the groupwise sparsity and the within group sparsity and developed an accelerated generalized gradient descent to fit the model. Fang et al. [21] and Vincent et al. [22] extended it to the adaptive sparse group Lasso and the multinomial sparse group Lasso respectively.

Although group Lasso, sparse group Lasso and their extensions [18], [21], [22] have been successfully applied to classification and gene selection, their effectiveness relies highly on the group division. For microarray gene expression data, it is desirable to divide genes into different groups according to gene pathways. However, it is rather difficult to detect gene pathways in complex biological processes. Although the sparse group Lasso can identify important groups and important genes within selected groups, it applies the same penalty coefficient to all genes without considering their relative importance. Moreover, the importance of a group is merely measured by the number of genes in each group. It is for this reason that if the group sizes are not even, the sparse group Lasso may not work well. The adaptive shrinkage methods [20]–[22] can select genes adaptively by using constructed adaptive weights which seem to have solved these problems. For example, adaptive Lasso [20] was developed to penalize all the coefficients by using the inverse of the initial estimator. Adaptive sparse group Lasso [21] was designed with a group bridge estimator. From a statistical perspective, the constructed weights have statistical meanings, which can also be roundly utilized to evaluate the importance of genes. Although the multinomial sparse group Lasso [22] introduces the weight mechanism which includes group weights and parameter weights (i.e., gene weights), it does not provide biological explanations of these weights. In addition, the above weights rely on the actual values of cancer microarray data, so they are not robust to outliers. Hence, the above-mentioned methods cannot infer the distinct biological meanings and are not very effective when performing gene selection in cancer classification.

Compare to the previous existing works, we apply the weighted gene co-expression network modules corresponding to biological pathways in systematic biology to gene groups in machine learning which has biological meanings and is ease to implementation. Furthermore, the gene and group weights with biological significance can be constructed effectively by joint mutual information. Based on the above two ideas, a Weighted General Group Lasso (WGGL) model is proposed, which is effective to select informative genes in cancer classification.

3 Problem Description

Cancer screening and diagnostic applications often predict the tumor type for a new sample accurately using as few relevant important genes as possible. These genes are closely related to biological processes. In this paper, we focus on a general binary classification problem. Given a data set $(X, \boldsymbol{y}) = \{(\boldsymbol{x}_i, y_i)|i = 1, \cdots, n\}$, where $\boldsymbol{x}_i = (x_{i1}, \cdots, x_{ip})$ is the input vector and $y_i \in \{0, 1\}$ indicates its class label. The classification problem is therefore to obtain a discrimination rule $f : R^p \rightarrow \{0, 1\}$ to judge in which class the features belong. For cancer gene expression data, n and p represent the number of sample tissues and the number of genes respectively. Let $\boldsymbol{y} = (y_1, \cdots, y_n)^T$ be the response vector and $X = (\boldsymbol{x}_1; \cdots; \boldsymbol{x}_n) = (\boldsymbol{x}_{(1)}, \cdots, \boldsymbol{x}_{(p)})$ be the model matrix. Let $\boldsymbol{x}_{(j)} = (x_{1j}, \cdots, x_{nj})^T$ denote the j^{th} predictor. In terms of a linear regression model, the response vector $\boldsymbol{y}$ can be predicted by

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} + \epsilon = \sum_{j=1}^{p} \hat{\beta}_j \boldsymbol{x}_{(j)} + \epsilon \tag{1}$$

where $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \cdots, \hat{\beta}_p)^T$ is the estimated coefficient vector, and $\epsilon = (\epsilon_1, \cdots, \epsilon_n) \sim N(0, \sigma^2 I_n)$ is the error vector. For simplicity, we assume that the predictors are standardized and the response vector is centered which indices that the intercept ϵ can be removed from the model. The number of non-zero estimated coefficients in $\hat{\boldsymbol{\beta}}$ represents the number of selected genes. Assume that predictors are divided into m groups, the input matrix X and $\hat{\boldsymbol{\beta}}$ can be represented as $X = (X^{(1)}, \cdots, X^{(m)})$ and $\hat{\boldsymbol{\beta}} = (\hat{\boldsymbol{\beta}}^{(1)T}, \cdots, \hat{\boldsymbol{\beta}}^{(m)T})^T$ respectively. Then the response vector $\boldsymbol{y}$ is predicted by $\hat{\boldsymbol{y}} = \sum_{l=1}^{m} X^{(l)}\hat{\boldsymbol{\beta}}^{(l)}$. Let $\mathrm{I}(\cdot)$ be the indicator function and $\hat{y}_\tau$ the prediction value for the given sample τ by the discrimination rule. $\mathrm{I}(\hat{y}_\tau > 0.5)$ denotes the classification function $f(\boldsymbol{x}_\tau)$. Therefore, the binary classification problem can be solved by the regression method in [10].

4 Weighted General Group Lasso

In this paper, an effective grouping method will be introduced to group the given data into m groups. We use joint mutual information to determine the weights of the divided m groups and those of the genes in each group. Based on the two types of weights, we propose a weighted general group Lasso (WGGL for short) statistical learning model which is formulated as follows:

$$\begin{aligned}\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \Big\{ &\frac{1}{2}\|\boldsymbol{y} - \sum_{l=1}^{m} X^{(l)}\boldsymbol{\beta}^{(l)}\|_2^2 \\ + (1-\alpha)\lambda \sum_{l=1}^{m} \eta_l \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_2 &+ \alpha\lambda \sum_{l=1}^{m} \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_1 \Big\}\end{aligned} \tag{2}$$

In fact, Equ. (2) is a generalization of existing Lasso models and follows the same grouped gene selection framework $\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \Big\{ L(\boldsymbol{\beta}) + R(\boldsymbol{\beta}) \Big\}$ in which $L(\boldsymbol{\beta})$ is the loss function and $R(\boldsymbol{\beta})$ is the penalty term. In the proposed WGGL model, the loss function is the squared error loss which is identical to that of existing Lasso models, i.e.,

$$L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{y} - \sum_{l=1}^{m} X^{(l)}\boldsymbol{\beta}^{(l)}\|_2^2. \tag{3}$$

However, the penalty term of the WGGL model is different from that of existing Lasso models and contains both the weights of groups and those of genes, i.e.,

$$R(\boldsymbol{\beta}) = (1-\alpha)\lambda \sum_{l=1}^{m} \eta_l \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_2 + \alpha\lambda \sum_{l=1}^{m} \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_1 \tag{4}$$

where $\alpha \in [0, 1]$ and $\lambda \in [0, \infty)$ are regularization parameters. η_l and $\boldsymbol{w}^{(l)}$ are the group weight and the gene weight matrices respectively. They can be calculated by joint mutual information (to be discussed in Section 5.2). The first item of Equ. (4) is called the adaptive group Lasso penalty which encourages sparsity of genes in groups. The second item is called the adaptive Lasso penalty which encourages sparsity of genes within each group. The weighted l_1/l_2-norm $\sum_{l=1}^{m} \eta_l \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_2$ (or the adaptive group Lasso penalty) penalizes the coefficients of significant groups, i.e., smaller group weight coefficients mean more important groups are selected first. The weighted l_1-norm $\sum_{l=1}^{m} \|\boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}\|_1$ (or the adaptive Lasso penalty) penalizes each gene in the selected groups so that the coefficients of irrelevant genes are shrunken to zero, i.e., bigger gene weight coefficients imply that those genes are less important and therefore there is a lower possibility of them being selected. In other words, the penalty term (4) leads to sparsity in both inter- and intra-group genes. WGGL improves the accuracy of gene selection and reduces estimation bias by adopting lower penalties for larger coefficients. In addition, WGGL becomes the sparse group Lasso model developed in [19] if the weight matrix $\boldsymbol{w}^{(l)}$ is an identity matrix. Therefore, WGGL is a generalization which includes the sparse group lasso of [19] in particular cases.

It is difficult to calculate $\hat{\boldsymbol{\beta}}$ in the current form of Equ. (2). Fortunately, the gene weight matrix $\boldsymbol{w}^{(l)}$ is a positive invertible matrix (to be proved in Section 5.2.1). We denote $\boldsymbol{\theta}^{(l)} = \boldsymbol{w}^{(l)}\boldsymbol{\beta}^{(l)}$ $(l = 1, \ldots, m)$ and Equ. (2) is simplified as:

$$\begin{aligned}\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \Big\{ &\frac{1}{2}\|\boldsymbol{y} - \sum_{l=1}^{m} \tilde{X}^{(l)}\boldsymbol{\theta}^{(l)}\|_2^2 \\ &+ (1-\alpha)\lambda \sum_{l=1}^{m} \eta_l \|\boldsymbol{\theta}^{(l)}\|_2 + \alpha\lambda \sum_{l=1}^{m} \|\boldsymbol{\theta}^{(l)}\|_1 \Big\}\end{aligned} \tag{5}$$

where $\tilde{X}^{(l)} = X^{(l)}(\boldsymbol{w}^{(l)})^{-1}$. Therefore, to obtain the optimal estimated coefficient vector $\hat{\boldsymbol{\beta}}$ of WGGL we need to obtain the optimal $\hat{\boldsymbol{\theta}}$. It is important to note that Equ. (5) is convex, i.e., the optimal solution $\hat{\boldsymbol{\theta}}$ can be obtained by subgradient equations. For the g^{th} group $(g = 1, \ldots, m)$, the solution $\hat{\boldsymbol{\theta}}^{(g)}$ satisfies

$$X^{(g)T}(\boldsymbol{y} - \sum_{l=1}^{m} \tilde{X}^{(l)}\hat{\boldsymbol{\theta}}^{(l)}) = \eta_g(1-\alpha)\lambda \boldsymbol{u}_g + \alpha\lambda \boldsymbol{v}_g, \tag{6}$$

where $\boldsymbol{u}_g$ and $\boldsymbol{v}_g$ are subgradients of $\|\hat{\boldsymbol{\theta}}^{(g)}\|_2$ and $\|\hat{\boldsymbol{\theta}}^{(g)}\|_1$ respectively. According to [19], $\boldsymbol{u}_g = \hat{\boldsymbol{\theta}}^{(g)}/\|\hat{\boldsymbol{\theta}}^{(g)}\|_2$ if $\hat{\boldsymbol{\theta}}^{(g)} \neq \mathbf{0}$, otherwise $\|\boldsymbol{u}_g\|_2 \leq 1$. $v_{gj} = sign(\hat{\theta}_j^{(g)})$ if $\hat{\theta}_j^{(g)} \neq 0$, otherwise $v_{gj} \in [-1, 1]$.

Following the analysis in [19], $\hat{\boldsymbol{\theta}}^{(g)} = \mathbf{0}$ is satisfied in Equ. (6) if $\|S(\tilde{X}^{(g)T}\boldsymbol{r}_{(-g)}, \alpha\lambda)\|_2 \leq \eta_g(1-\alpha)\lambda$ where $\boldsymbol{r}_{(-g)} = \boldsymbol{y} - \sum_{l \neq g} X^{(l)}\hat{\boldsymbol{\theta}}^{(l)}$ is the partial residual of $\boldsymbol{y}$. S is the coordinate-wise soft threshold operator which is defined as $S(\boldsymbol{z}, \alpha\lambda))_j = sign(z_j)(|z_j| - \alpha\lambda)_+$. If $\hat{\boldsymbol{\theta}}^{(g)} \neq \mathbf{0}$, then the subgradient condition for $\hat{\theta}_k^{(g)}$ becomes

$$\tilde{X}_k^{(g)T}(\boldsymbol{y} - \sum_{l=1}^{m} \tilde{X}^{(l)}\hat{\boldsymbol{\theta}}^{(l)}) = (1-\alpha)\lambda\eta_g u_{gk} + \alpha\lambda v_{gk}. \tag{7}$$

This is satisfied for $\hat{\theta}_k^{(g)} = 0$ (or $\hat{\beta}_k^{(g)} = 0$) if $|\tilde{X}_k^{(g)T}\boldsymbol{r}_{(-g,k)}| \leq \alpha\lambda$ where $\boldsymbol{r}_{(-g,k)} = \boldsymbol{r}_{(-g)} - \sum_{j \neq k} \tilde{X}_j^{(g)T}\hat{\boldsymbol{\theta}}^{(g)}$ is the partial residual of $\boldsymbol{y}$. When $\hat{\theta}_k^{(g)} \neq 0$, $\hat{\theta}_k^{(g)}$ is obtained by

$$\hat{\theta}_k^{(g)} = \arg\min_{\hat{\theta}_k^{(g)}} \Big\{ \frac{1}{2}\|\boldsymbol{y} - \sum_{l=1}^{m} \tilde{X}^{(l)}\boldsymbol{\theta}^{(l)}\|_2^2 + (1-\alpha)\lambda\eta_g\|\boldsymbol{\theta}^{(g)}\|_2 + \alpha\lambda\|\boldsymbol{\theta}^{(g)}\|_1 \Big\}. \tag{8}$$

Equ. (8) is a one-dimensional optimization problem in respect to $\hat{\theta}_k^{(g)}$ which can be solved by classical existing optimization algorithms, e.g., gradient descent algorithms [19] to get $\hat{\beta}_k^{(g)} = \hat{\theta}_k^{(g)}/w_k^{(g)}$. Similarly, the optimal solution $\hat{\boldsymbol{\beta}}$ of WGGL can be obtained by $\hat{\boldsymbol{\beta}}^{(g)} = (\boldsymbol{w}^{(g)})^{-1}\hat{\boldsymbol{\theta}}^{(g)}$.

5 Proposed Algorithm

For given microarray gene expression data, two aspects are crucial in the performance of gene selection in cancer classification: appropriately dividing the genes into groups and determining the weights of groups and genes with biological meanings. For the problem under study, we propose a new gene selection algorithm (GSA for short) based on the weighted general group Lasso model. GSA contains three components: gene grouping heuristic (GGH), gene and group weight calculation (GGWC) and solution construction procedure (SCP). For a given X, $\boldsymbol{y}$, α and λ, GSA outputs $\hat{\boldsymbol{\beta}}$. The flowchart of the proposed gene selection framework as in Fig. 1. Based on the flowchart, the detailed framework of GSA is depicted by Algorithm 1.

Algorithm 1: Gene Selection Algorithm (GSA) framework

```
Input: $X$, $\boldsymbol{y}$, $\alpha$, $\lambda$
Output: $\hat{\boldsymbol{\beta}}$
Divide genes in $X$ into $m$ groups $(X^{(1)}, \cdots, X^{(m)})$ by GGH;
for $l = 1$ to $m$ do
    Call GGWC to calculate $\boldsymbol{w}^{(l)}$, $\eta_l$ and $\tilde{X}^{(l)}$ in terms of $X^{(l)}$;
Call SCP to calculate $\hat{\boldsymbol{\beta}}$;
return $\hat{\boldsymbol{\beta}}$.
```

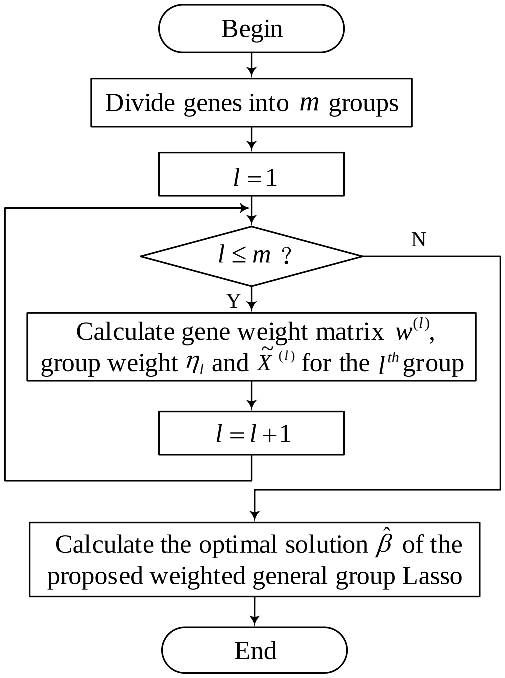

Begin
Divide genes into m groups
$l = 1$
$l \leq m$?
N
Y
Calculate gene weight matrix $w^{(l)}$,
group weight η_l and $\tilde{X}^{(l)}$ for the l^{th} group
$l = l + 1$
Calculate the optimal solution $\hat{\beta}$ of the
proposed weighted general group Lasso
End

Fig. 1: Flowchart of the proposed gene selection framework.

5.1 Gene Grouping Heuristic

Cancer-diagnosis is a complex and well-orchestrated biological process with a synergistic work of a large number of genes. For the sparse group Lasso model [19], genes are grouped into “genesets” using cytogenetic position data. Not all genes in the dataset are grouped and the involved genes are grouped coarsely in an intuitive way without considering a gene enrichment function. In addition, the grouping method used by this model groups specific datasets which cannot be applied to general cases. In fact, all related genes, no matter how closely or loosely related biologically they are, are clustered into one group which results in inaccurate predictions. However, interactions among genes can be represented by networks which leads to the possible use of the weighted gene co-expression network analysis (WGCNA) [24] in order to group genes. WGCNA is a network-based systems biology approach in which highly correlated across sample genes are clustered into the same group (or module).

In WGCNA, module identification essentially depends on the weighted gene co-expression networks. Genes consist of the nodes in each network. The edges of the network are constructed using correlations between genes in the expression data which are measured by the similarities between genes. Two nodes are only connected by an edge if their similarity is not less than threshold σ. Fig. 2 shows an example.

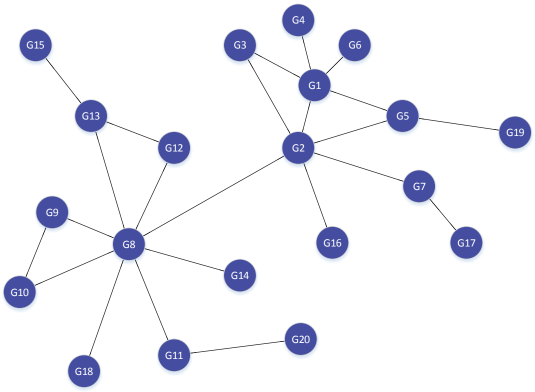


Fig. 2: Gene Network.

Inspired by this idea, we propose a gene grouping heuristic (GGH) which applies the weighted gene co-expression network modules in systematic biology to gene grouping in machine learning. Since there would be two or more types of tumor in a cancer, GGH groups each of them by WGCNA. In this paper, two tumor types are considered for the cancer gene expression data, which is represented by $X = [X_{T_1}, X_{T_2}]$. $X_{T_1} = (\boldsymbol{x}_1; \cdots; \boldsymbol{x}_{n'})$ and $X_{T_2} = (\boldsymbol{x}_{n'+1}; \cdots; \boldsymbol{x}_n)$ denote type1 data and type2 data respec-

```
Algorithm 2: Gene Grouping Heuristic (GGH)
  Input: Matrix $X = [X_{T_1}, X_{T_2}]$
  Output: Groups of $X$
1 Divide genes of type1 data $X_{T_1}$ into groups
  $G_1 = \{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_{k_1}\}$ by IdentifyModule(NeT$_1$);
2 Divide genes of type2 data $X_{T_2}$ into groups
  $G_2 = \{\hat{g}_{k_1+1}, \hat{g}_{k_1+2}, \ldots, \hat{g}_m\}$ by
  IdentifyModule(NeT$_2$);
3 $G \leftarrow G_1 \bigcup G_2$;
4 return $G$.
```

```
Algorithm 3: IdentifyModule (NeT)
   Input: Matrix $X = (\boldsymbol{x}_{(1)}, \cdots, \boldsymbol{x}_{(p)})$.
   Output: Identified modules or groups.
1  for $h = 1$ to $p$ do
2      for $j = 1$ to $p$ do
3          Calculate the gene co-expression similarity $s_{hj}$;
4  Determine the threshold parameter $\sigma$ in terms of $X$
   using the approximate scale-free topology criterion;
5  for $h = 1$ to $p$ do
6      for $j = 1$ to $p$ do
7          Calculate $a_{hj}$ by $a_{hj} = s_{hj}^{\sigma}$;
8  Construct NeT using adjacency matrix $A = (a_{hj})_{p \times p}$;
9  for $h = 1$ to $p$ do
10     for $j = 1$ to $p$ do
11         $l_{hj} \leftarrow 0$, $\hat{k}_h \leftarrow 0$, $\hat{k}_j \leftarrow 0$;
12         for $u = 1$ to $p$ do
13             $l_{hj} \leftarrow l_{hj} + a_{hu}a_{uj}$;
14             $\hat{k}_h \leftarrow \hat{k}_h + a_{hu}$;
15             $\hat{k}_j \leftarrow \hat{k}_j + a_{hu}$;
16         $\omega_{hj} \leftarrow \frac{l_{hj}+a_{hj}}{\min\{\hat{k}_h,\hat{k}_j\}+1-a_{hj}}$;
17         $d_{hj}^{\omega} \leftarrow 1 - \omega_{hj}$;
18 Construct the hierarchical clustering dendrogram
   according to matrix $D = (d_{hj}^{\omega})_{p \times p}$;
19 Call the dynamic tree cut algorithm to identify the
   modules of NeT $V = \{v_1, v_2, \ldots, v_k\}$;
20 return Groups $V$.
```

tively. X_{T_t} can be represented by $X_{T_t} = (\boldsymbol{x}_{(1)}^{T_t}, \cdots, \boldsymbol{x}_{(p)}^{T_t})$ $(t = \{1, 2\})$. $\boldsymbol{x}_{(i)}^{T_t}$ $(i = 1, \ldots, p)$ denotes a gene (or a network node). Row $\boldsymbol{x}_j$ $(j = 1, \cdots, n'$ or $j = n'+1, \cdots, n)$ corresponds to a sample measurement.

According to [24], the gene co-expression similarity $s_{hj} = |cor(\boldsymbol{x}_{(h)}^{T_t}, \boldsymbol{x}_{(j)}^{T_t})|$ measures the similarity between columns $\boldsymbol{x}_{(h)}^{T_t}$ and $\boldsymbol{x}_{(j)}^{T_t}$. Using the power adjacency function $a_{hj} = s_{hj}^{\sigma}$ $(\sigma \geq 1)$, an adjacency matrix $A = (a_{hj})_{p \times p}$ is obtained by transforming the similarity matrix $S = (s_{hj})_{p \times p}$ in which $a_{hj} \in [0, 1]$ denotes the network connection strength between nodes h and j. By applying the approximate scale-free topology criterion [24], we can get the optimal threshold parameter σ for X_{T_1} and X_{T_2}. The weighted gene co-expression network of X_{T_1} or X_{T_2} (NeT$_1$ or NeT$_2$) can be constructed by its symmetric adjacency matrix $A_{p \times p}$. The total connection strength between nodes h and j is measured by the topological overlap similarity ω_{hj} (TOM for short) [26]. $\omega_{hj} = \frac{l_{hj}+a_{hj}}{\min\{\hat{k}_h,\hat{k}_j\}+1-a_{hj}}$ where $l_{hj} = \sum_u a_{hu}a_{uj}$, $\hat{k}_h = \sum_u a_{hu}$, $u = 1, \cdots, p$. The dissimilarity between nodes h and j is calculated by $d_{hj}^{\omega} = 1 - \omega_{hj}$. By applying the dynamic tree cut algorithm presented in [27] to both NeT$_1$ and NeT$_2$, modules (or groups) are identified. The GGH grouping nodes in NeT$_1$ and NeT$_2$ are formally depicted in Algorithm 2 and the routine IdentifyModule (NeT) is given in Algorithm 3.

The time complexity of steps 1-3 is $O(p^2)$, that of step 4 is $O(p^2)$, that of steps 5-7 is $O(p^2)$ and that of step 8 is also $O(p^2)$. The time complexity of steps 9-17 is $O(p^3)$ and that of step 18-19 is $O(p^2 \log p)$. Therefore, the time complexity of GGH is $O(p^3)$. Algorithm 2 divides each gene twice and the p genes are extended to $\hat{p} = 2p$ genes in the obtained m groups. Therefore, the input matrix X can be represented by $X = (X^{(1)}, \cdots, X^{(m)}) = (\boldsymbol{x}_{(1)}, \cdots, \boldsymbol{x}_{(\hat{p})})$.

5.2 Gene and Group Weight Calculation

In the existing literature, both gene weights and group weights are vocally calculated either by using statistical methods or only by the number of genes. Little research considers the biological relations among genes which leads to a lack of precision in cancer diagnosis. In fact, genes have interactions with each other which can be measured by their joint mutual information. Mutual information between genes $\boldsymbol{x}_{(h)}$ and $\boldsymbol{x}_{(j)}$ in cancer gene expression data generally describes the degree of mutual dependence between the two vectors. In this paper, we use joint mutual information to evaluate the gene weight of $\boldsymbol{x}_{(k)}$ which depends on the degree of correlation between $\boldsymbol{x}_{(k)}$ and each pair of the other genes $\left(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}\right)$ $(h \neq j \neq k)$. Each group weight is determined by the gene weights among the group. There are two processes involved: gene weight and group weight computation.

5.2.1 Computing Gene Weights

In the group $\hat{g}_l$ $(l = 1, 2, \ldots, m)$, gene $\boldsymbol{x}_{(k)}$ not only correlates with, but has an impact on both pairs of genes $\left(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}\right)$ $(h \neq j \neq k)$. In other words, the weight of gene $\boldsymbol{x}_{(k)}$ depends on both the independent importance s_k^l and the dependent importance t_k^l in its group.

Let $\hat{X} = (\hat{x}_1, \cdots, \hat{x}_n)^T$, $Y = (y_1, \cdots, y_n)^T$ and $Z = (z_1, \cdots, z_n)^T$. According to [38], mutual information is introduced to measure the amount of information shared by $\hat{X}$ and Y which is used to describe the degree of correlation between the two variables and is defined as follows:

$$I(\hat{X}; Y) = \sum_{\hat{x} \in \hat{X}} \sum_{y \in Y} p(\hat{x}, y) \log \frac{p(\hat{x}, y)}{p(\hat{x})p(y)}, \tag{9}$$

According to [34], the joint mutual information is defined as:

$$I(\hat{X}, Y; Z) = \sum_{\hat{x} \in \hat{X}} \sum_{y \in Y} \sum_{z \in Z} p(\hat{x}, y, z) \log \frac{p(\hat{x}, y, z)}{p(\hat{x}, y)p(z)}, \tag{10}$$

where $p(\hat{x}, y, z)$ is the joint probability of $\hat{x}, y$ and z, $p(\hat{x}, y)$ is the joint probability of $\hat{x}$ and y, and $p(z)$ is the probability of z. Here, the probability is a Gaussian kernel probability density estimator which is the same as in [39]. Based on

Equ. (10), we define the independent importance s_k^l of $\boldsymbol{x}_{(k)}$ on $\big(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}\big)$ in group $\hat{g}_l$ $(l = 1, 2, \ldots, m)$ as:

$$s_k^l = \frac{1}{A_{\hat{p}_l-1}^2} \sum_{h=1}^{\hat{p}_l} \sum_{j=1}^{\hat{p}_l} I\big(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)}\big), \quad \{h \neq j \neq k; k = 1, \cdots, \hat{p}_l\} \tag{11}$$

in which $A_{\hat{p}_l-1}^2 = \frac{(\hat{p}_l-1)!}{(\hat{p}_l-1-2)!} = (\hat{p}_l - 1)(\hat{p}_l - 2)$ is the number of permutations for all gene pairs except $\boldsymbol{x}_{(k)}$ in group $\hat{g}_l$. s_k^l measures the average amount of information shared by all the other gene pairs with respect to $\boldsymbol{x}_{(k)}$ in group $\hat{g}_l$. In terms of Equations (10) and (11), a greater s_k^l implies more information is shared by all pairs of the remaining genes with the gene $\boldsymbol{x}_{(k)}$. In other words, s_k^l quantitatively measures the significant degree of $\boldsymbol{x}_{(k)}$ in the remaining genes in $\hat{g}_l$. A greater s_k^l implies more significant $\boldsymbol{x}_{(k)}$ in $\hat{g}_l$.

Similarly to s_k^l, based on Equs. (9) and (10), we define the dependent importance t_k^l as:

$$t_k^l = \frac{1}{A_{\hat{p}_l-1}^2} \sum_{h=1}^{\hat{p}_l} \sum_{j=1}^{\hat{p}_l} \Big[I\big(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)}\big) - I\big(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(k)}\big) - I\big(\boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)}\big)\Big]^+, \ \{h \neq j \neq k; \ k = 1, \cdots, \hat{p}_l\} \tag{12}$$

where $[\varsigma]^+ = \max(\varsigma, 0)$. t_k^l illustrates the average increment of the shared information between all pairs of the remaining genes with respect to $\boldsymbol{x}_{(k)}$ in $\hat{g}_l$. $t_k^l > 0$ indicates that more information can be obtained from the joint mutual information $I(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)})$ than the sum of mutual information of $I(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(k)})$ and $I(\boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)})$. In addition, joint mutual information can be expressed by $I(\boldsymbol{x}_{(h)}, \boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)}) = I(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(k)}) + I(\boldsymbol{x}_{(j)}; \boldsymbol{x}_{(k)}) - I(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)}) + I(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)} | \boldsymbol{x}_{(k)})$ according to [37]. $t_k^l > 0$ implies that the correlation of all the other pair of genes increases when the gene $\boldsymbol{x}_{(k)}$ is introduced to $(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)})$ so that $I\big(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)} | \boldsymbol{x}_{(k)}\big) > I\big(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)}\big)$, i.e., the conditional mutual information of $(\boldsymbol{x}_{(h)}; \boldsymbol{x}_{(j)})$ given $\boldsymbol{x}_{(k)}$ is greater than the mutual information of $\boldsymbol{x}_{(h)}$ and $\boldsymbol{x}_{(j)}$. On the contrary, $\boldsymbol{x}_{(k)}$ has no influence on any pair of the remaining genes when $t_k^l = 0$.

To evaluate the amount of s_k^l and t_k^l, we apply the information entropy to $\boldsymbol{s}^l = (s_1^l, \ldots, s_{\hat{p}_l}^l)^T$ and $\boldsymbol{t}^l = (t_1^l, \ldots, t_{\hat{p}_l}^l)^T$. For variable or vector $\hat{X}$, $H(\hat{X}) = -\sum_{\hat{x} \in \hat{X}} p(\hat{x}) \log(\hat{x})$, where $p(\hat{x}) = \hat{x} / \sum_{i=1}^{n} \hat{x}_i$ represents the probability distribution of each $\hat{x} \in \hat{X}$ which is different from those of Equs. (9) and (10). The entropy $H(\hat{X})$ is an average uncertainty measure of $\hat{X}$. Less the information entropy of a variable means greater the amount of information the variable provides and implies more important the variable. In this paper, we only denote the weights of $\boldsymbol{s}^l$ and $\boldsymbol{t}^l$ by $\mu_1 = \frac{e^{-H(\boldsymbol{s}^l)}}{e^{-H(\boldsymbol{s}^l)} + e^{-H(\boldsymbol{t}^l)}}$ and $\mu_2 = \frac{e^{-H(\boldsymbol{t}^l)}}{e^{-H(\boldsymbol{s}^l)} + e^{-H(\boldsymbol{t}^l)}}$ respectively. By integrating s_k^l and t_k^l, the comprehensive importance of $\boldsymbol{x}_{(k)}$ in group $\hat{g}_l$ is determined by $\bar{s}_k^l = \mu_1 s_k^l + \mu_2 t_k^l$ which can be simplified as:

$$\bar{s}_k^l = \frac{e^{-H(\boldsymbol{s}^l)} s_k^l + e^{-H(\boldsymbol{t}^l)} t_k^l}{e^{-H(\boldsymbol{s}^l)} + e^{-H(\boldsymbol{t}^l)}}. \tag{13}$$

Since $s_k^l \geq 0$ and $t_k^l \geq 0$, it is natural that $\bar{s}_k^l \geq 0$. The gene $\boldsymbol{x}_{(k)}$ has distinct biological meanings when $\bar{s}_k^l > 0$ whereas it is meaningless when $\bar{s}_k^l = 0$.

In terms of Equ. (2), a greater weight means the gene coefficient is less important. Therefore, we define the weight of $\boldsymbol{x}_{(k)}$ in group $\hat{g}_l$ as:

$$w_k^{(l)} = \begin{cases} \frac{e^{-H(\boldsymbol{s}^l)} + e^{-H(\boldsymbol{t}^l)}}{e^{-H(\boldsymbol{s}^l)} s_k^l + e^{-H(\boldsymbol{t}^l)} t_k^l}, & \text{if } \bar{s}_k^l > 0 \\ 1/\varepsilon, & \text{otherwise} \end{cases} \tag{14}$$

in which $0 < \varepsilon \ll 1$ is a threshold given in advance. In other words, the genes with $\bar{s}_k^l = 0$ are penalized by a very big weight. According to Equ. (14), we construct the weight matrix of genes in group $\hat{g}_l$ $(l = 1, \cdots, m)$ for Equ. (2) as:

$$\boldsymbol{w}^{(l)} = \text{diag}(w_1^{(l)}, \ldots, w_{\hat{p}_l}^{(l)}), \tag{15}$$

Since $w_k^{(l)} > 0$, the determinant of the weight matrix $\boldsymbol{w}^{(l)} \neq 0$, i.e., $\det(\boldsymbol{w}^{(l)}) = w_1^{(l)} \times \cdots \times w_{\hat{p}_l}^{(l)} \neq 0$. Therefore, matrix $\boldsymbol{w}^{(l)}$ is invertible.

5.2.2 *Computing Group Weights*

The importance of group $\hat{g}_l$ $(l = 1, \ldots, m)$ depends on the importance of its genes which is defined by:

$$\xi_l = \sum_{k=1}^{\hat{p}_l} \bar{s}_k^l. \tag{16}$$

ξ_l demonstrates the importance of $\hat{g}_l$ using the sum of the importance of its genes. A greater ξ_l value denotes what the importance of $\hat{g}_l$ is. $\xi_l = 0$ implies that $\hat{g}_l$ is not important.

Similarly to gene weight computing, we construct the weight coefficients of $\hat{g}_l$ using:

$$\eta_l = \left(\xi_l + \frac{1}{\sqrt{\hat{p}_l}}\right)^{-1} \tag{17}$$

where $\hat{p}_l$ is the number of genes in $\hat{g}_l$. The group weight vector is defined as:

$$\boldsymbol{\eta} = (\eta_1, \ldots, \eta_m)^T \tag{18}$$

which are the weights of the group Lasso in Equ. (2).

According to the above analysis, the gene and group weights calculation (GGWC) procedure is formally described in Algorithm 4.

In terms of Equations (9)-(12), the time complexity of GGWC is $O(\hat{p}_l^3)$. The weights constructed by GGWC explain biologically the importance of both genes and groups.

5.3 Solution Construction Procedure

Once we obtain the gene weights and group weights for the grouped genes, we can construct a solution in terms of WGGL in Equ. (2). The idea follows that given in [18], [19] which separates the general optimization procedure for the objective into two sequential steps, i.e., groupwise sparsity selection and within group sparsity selection. They are repeated until a convergence condition is met. In this paper, the absolute difference between two consecutive estimates smaller than 0.001 is set as the termination criterion. The procedure is formally described in Algorithm 5. Although the time complexity in each iteration is $O(m\hat{p}_g)$, the number of iterations cannot be determined. Therefore, it is hard to estimate the time complexity of SCP in a closed form.

Algorithm 4: Gene and Group Weight Calculation (GGWC)

```
Input: $X^{(l)}$
Output: $\boldsymbol{w}^{(l)}$, $\eta_l$, $\tilde{X}^{(l)}$
for $k = 1$ to $\hat{p}_l$ do
    Calculate $s_k^l$ by Equ. (11);
    Calculate $t_k^l$ by Equ. (12);
$H(\boldsymbol{s}^l) \leftarrow 0$, $H(\boldsymbol{t}^l) \leftarrow 0$;
for $k = 1$ to $\hat{p}_l$ do
    $H(\boldsymbol{s}^l) \leftarrow H(\boldsymbol{s}^l) - p(s_k^l)\log(s_k^l)$;
    $H(\boldsymbol{t}^l) \leftarrow H(\boldsymbol{t}^l) - p(t_k^l)\log(t_k^l)$;
$\xi_l \leftarrow 0$;
for $k = 1$ to $\hat{p}_l$ do
    Calculate $\bar{s}_k^l$ using Equ. (13);
    $\xi_l \leftarrow \xi_l + \bar{s}_k^l$;
    if $\bar{s}_k^l > 0$ then
        $w_k^{(l)} \leftarrow 1/\bar{s}_k$;
    else
        $w_k^{(l)} \leftarrow 1/\varepsilon$;
$\boldsymbol{w}^{(l)} \leftarrow \mathrm{diag}(w_1^{(l)}, \ldots, w_{\hat{p}_l}^{(l)})$;
Calculate $\eta_l$ by Equation (17);
$\tilde{X}^{(l)} \leftarrow X^{(l)}(\boldsymbol{w}^{(l)})^{-1}$;
return $\boldsymbol{w}^{(l)}$, $\eta_l$, $\tilde{X}^{(l)}$.
```

Algorithm 5: Solution Construction Procedure (SCP)

```
Input: $\boldsymbol{w}$, $\boldsymbol{\eta}$, $\tilde{X}$
Output: $\hat{\boldsymbol{\beta}}$
$\boldsymbol{\beta}_0 \leftarrow \boldsymbol{0}$;
repeat
    $flag \leftarrow$ True;
    for $g = 1$ to $m$ do
        if ($\|S(\tilde{X}^{(g)T}\boldsymbol{r}_{(-g)}, \alpha\lambda)\|_2 \leq \eta_g(1-\alpha)\lambda$) then
            $\hat{\boldsymbol{\theta}}^{(g)} \leftarrow \boldsymbol{0}$;
            $\hat{\boldsymbol{\beta}}^{(g)} \leftarrow \boldsymbol{0}$;
        for $k = 1$ to $\hat{p}_g$ do
            if $|\tilde{X}_k^{(g)T}\boldsymbol{r}_{(-g,k)}| \leq \alpha\lambda$ then
                $\hat{\theta}_k^{(g)} \leftarrow 0$;
                $\hat{\beta}_k^{(g)} \leftarrow 0$;
            else
                Calculate $\hat{\theta}_k^{(g)}$ by Euq. (8);
                $\hat{\beta}_k^{(g)} \leftarrow \hat{\theta}_k^{(g)}/w_k^{(g)}$;
    $\boldsymbol{\beta}' \leftarrow \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0$;
    for $g = 1$ to $m$ do
        for $k = 1$ to $\hat{p}_g$ do
            if $\boldsymbol{\beta}'_{g,k} > 0.001$ then
                $flag \leftarrow$ False;
    if $flag = False$ then
        $\boldsymbol{\beta}_0 \leftarrow \hat{\boldsymbol{\beta}}$;
until ($flag = True$);
return $\hat{\boldsymbol{\beta}}$.
```

6 Experimental results

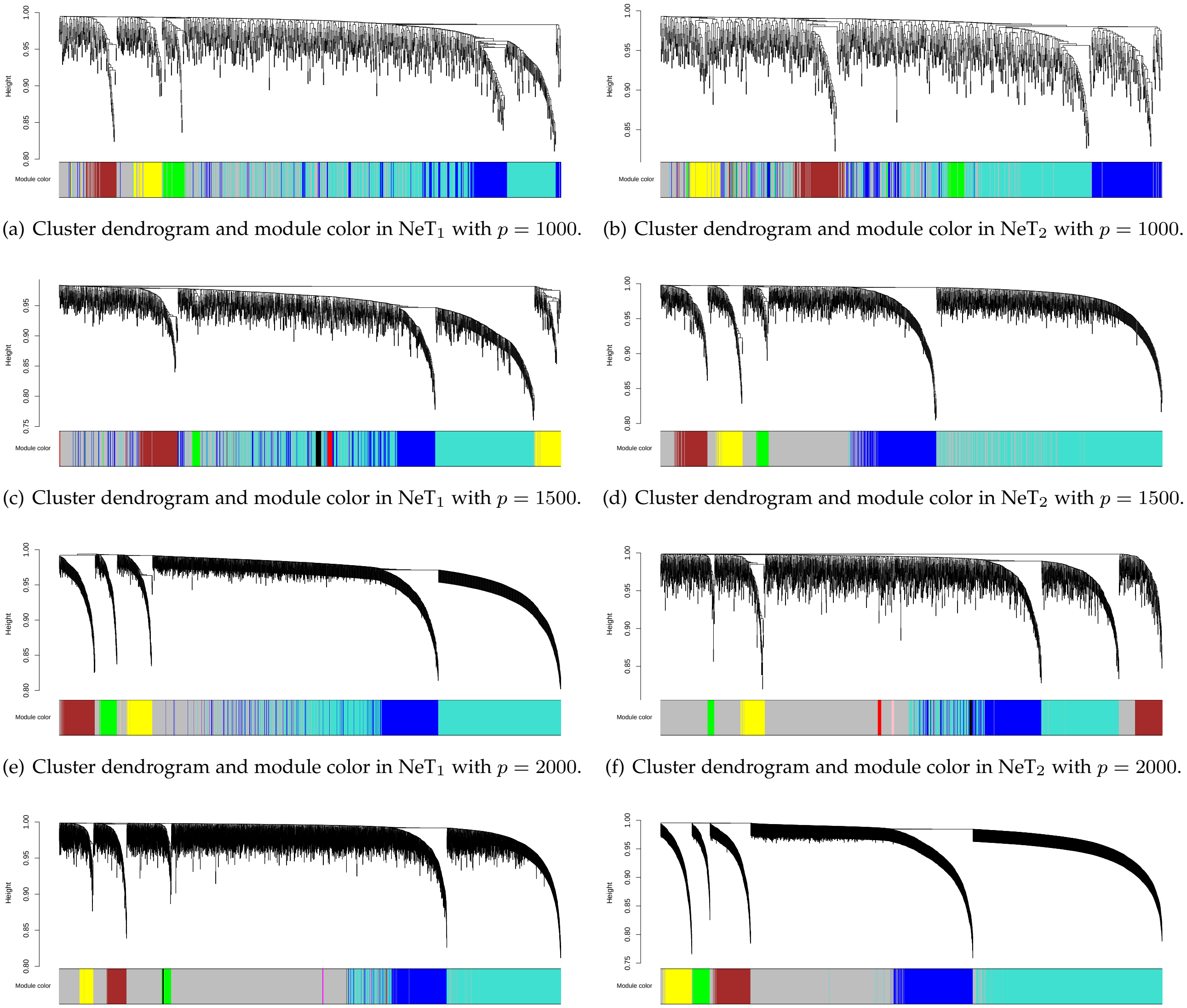
The related parameters and components of the proposed GSA framework are calibrated on random calibration instances. The proposed GSA with the WGGL model is compared with GSA methods with the sparse group Lasso and the group Lasso models over benchmark gene selection instances. All methods are implemented in R-3.3.2 for windows and tested on an Intel(R) Core(TM) i5-2400 CPU @ 3.10 GHz computer with 8.00 GB RAM with Windows Server 2007 standard.

The commonly used indexes misclassification error and the number of selected genes are adopted to evaluate the obtained response vector $\boldsymbol{y}$ of the proposals. Misclassification error is the error on the test data which is a set of examples used only to assess the performance (generalization) of a full specified classifier, which is defined by $E = \frac{1}{n}\sum_{i=1}^{n} \mathrm{I}(f(\boldsymbol{x}_i) \neq y_i)$. The number of selected genes is an index which reflects the gene selection performance of an algorithm.

6.1 Parameters and Components Calibration

Simon et. al [19] considered the SGL model using GG (Given Groups) for gene grouping and group and CE (Constant Estimator) for gene weight determination. Since SGL is similar to the WGGL model constructed in this paper, GG and CE are adopted for component calibration. There are two variants (GGH and GG) for the gene grouping component and two variants (GGWC and CE) for the weight construction component. In addition, there are two parameters λ and α which might have an effect on the performance of the proposed GSA. In this paper, λ takes values from $\{0.005, 0.01, 0.1, 0.15, 0.2, 0.5\}$ and $\alpha \in \{0.05, 0.1, 0.2, 0.4, 0.6, 0.8, 0.9, 0.95, 0.97, 0.99\}$. Therefore, there are $2 \times 2 \times 6 \times 10 = 240$ combinations. To calibrate the most appropriate components and parameters, each combination is performed on four groups of random calibration instances and each group is replicated 8 times. $240 \times 4 \times 8 = 7680$ tests are conducted in total. The performance of GSA is evaluated by RPD (relative percentage deviation). Let $E_k(H)$ be the misclassification error for instance k obtained by algorithm H and E_k^* be the lowest misclassification error for instance k obtained in all tests. RPD is defined by $RPD = \frac{E_k(H) - E_k^*}{E_k^*} \times 100\%$.

The calibration instances are randomly generated in a way so that the input matrix X and the response vector $\boldsymbol{y} = (y_1, \cdots, y_n)^T$ ($i = 1, \cdots, n$ and $y_i \in \{0, 1\}$) follow the distributions given in [27]. Four random datasets $(n, p) = \{(50, 1000), (75, 1500), (100, 2000), (120, 4000)\}$ are generated. GGH divides each gene twice and the p genes are extended to $\hat{p} = 2p$ genes. The four groups with $\hat{p} = 2000, 3000, 4000, 8000$ genes are divided into $m = 12, 14, 15, 16$ groups, respectively, by GGH. Details are given in Table 1. The number of rows for NeT$_1$ and NeT$_2$ represent the number of two randomly selected types of tumor. Fig. 3 demonstrates the cluster dendrograms in NeT$_1$ and NeT$_2$ for four random datasets respectively. The gene dendrograms are obtained by average linkage hierarchical clustering. The color row underneath each dendrogram



(a) Cluster dendrogram and module color in NeT$_1$ with $p = 1000$.

(b) Cluster dendrogram and module color in NeT$_2$ with $p = 1000$.

(c) Cluster dendrogram and module color in NeT$_1$ with $p = 1500$.

(d) Cluster dendrogram and module color in NeT$_2$ with $p = 1500$.

(e) Cluster dendrogram and module color in NeT$_1$ with $p = 2000$.

(f) Cluster dendrogram and module color in NeT$_2$ with $p = 2000$.

(g) Cluster dendrogram and module color in NeT$_1$ with $p = 4000$.

(h) Cluster dendrogram and module color in NeT$_2$ with $p = 4000$.

Fig. 3: Identification of network modules in NeT$_1$ and NeT$_2$.

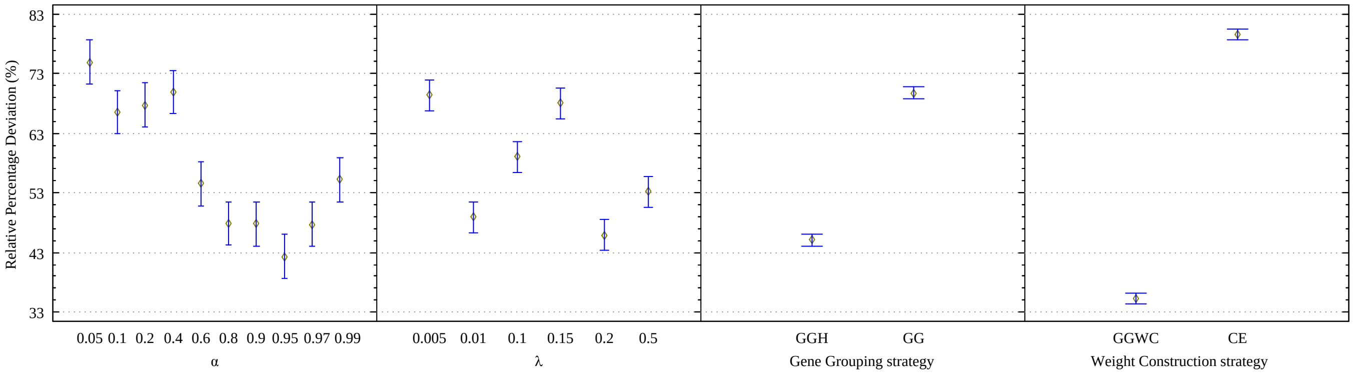


Fig. 4: Means plots and 95% confidence level Tukey HSD intervals for the parameters and components on the random calibration instances.

shows the module assignment determined by the dynamic tree cut algorithm. Genes in the same branch can be assigned to different modules.

The RPDs of each combination over the random instances are analyzed by the multi-factor analysis of variance (ANOVA) technique. Hypotheses should be ideally met by the experimental data among which three main hypotheses (independence of the residuals, homoscedasticity or homo-

TABLE 1: Results of identified network modules in NeT_1 and NeT_2 from the four groups of random calibration datasets.

Module	$n = 50, p = 1000$		$n = 75, p = 1500$		$n = 100, p = 2000$		$n = 120, p = 4000$	
	$[\text{NeT}_1]_{28\times1000}$	$[\text{NeT}_2]_{22\times1000}$	$[\text{NeT}_1]_{27\times1500}$	$[\text{NeT}_2]_{48\times1500}$	$[\text{NeT}_1]_{66\times2000}$	$[\text{NeT}_2]_{34\times2000}$	$[\text{NeT}_1]_{36\times4000}$	$[\text{NeT}_2]_{84\times4000}$
black	—	—	17	—	—	12	9	—
blue	154	212	247	222	311	293	478	600
brown	51	94	112	90	133	107	156	285
green	41	45	26	36	67	26	64	137
grey	328	282	386	540	530	996	2077	1309
magenta	—	—	—	—	—	—	8	—
pink	—	—	—	—	—	10	8	—
red	—	—	21	—	—	14	11	—
turquoise	375	314	617	539	859	450	1082	1451
yellow	51	53	74	73	100	92	107	218

geneity of the factor's levels variance and normality in the residuals of the model) are checked and accepted. Means plots and 95% confidence level Tukey HSD intervals for α, λ and the two algorithm components are depicted in Fig. 4. Recall that overlapping confidence intervals indicate statistical insignificance among the overlapped means.

Fig. 4 implies that RPD of the proposal has a mostly non-increasing tendency when $\alpha < 0.95$ while it increases when $\alpha \geq 0.95$. Most of the differences are statistically significant when $\alpha < 0.95$. GSA gets the least RPD when $\alpha = 0.95$. Therefore, we set $\alpha = 0.95$ for GSA in the following experiments. RPD of GSA fluctuates with an increase in λ. GSA gets two minimal values when $\lambda = 0.01$ and $\lambda = 0.2$. The RPD when $\lambda = 0.2$ is even less than that when $\lambda = 0.01$. The differences are statistically significant for the other λ values. Therefore, $\lambda = 0.2$ is adopted in the following experiments.

From Fig. 4, it can be observed that the difference between GGH and GG is statistically significant. Since the RPD of GSA with GGH is much less than that of GG for gene grouping, we adopt GGH to group genes in the following experiments. In addition, we can observe that the difference between GGWC and CE is statistically significant. The RPD of GSA with GGWC is much less than that of CE for gene and group weight determination. Therefore, GSA uses GGWC to determine the gene and group weights in the following experiments.

6.2 Comparison Results

To evaluate the advantages of the introduced WGGL model compared with the GL [17] and SGL [19] models, we compare GSA methods with the three models on three frequently studied public gene expression datasets: leukemia, brain cancer and ovarian cancer. Before the comparisons, the involved data is standardized by preprocessing.

6.2.1 Data Preprocessing

Leukemia dataset [1], [2] includes the expression profiles of 7129 genes in 47 cases of acute lymphoblastic leukemia (ALL) and 25 cases of acute myeloid leukemia (AML). The original data is available at: http://portals.broadinstitute.org/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=43. According to [1], [2], we preprocess this dataset using a threshold of 100 and a ceiling of 16000. If gene expression levels are less than 100, they are assigned a value of 100. Similarly, gene expression levels are assigned a value of 16000 if they are greater than 16000. Variation filters are applied to filter those genes which violate $\max(\hat{g})/\min(\hat{g}) > 5$ and $\max(\hat{g}) - \min(\hat{g}) > 500$. $\max(\hat{g})$ and $\min(\hat{g})$ are the maximum and minimum values of gene expressions of gene $\hat{g}$ among different samples. After preprocessing, the dataset contains 3571 genes. We set the label of 47 ALL samples to be 0 and 25 AML samples to be 1. To comprehensively illustrate the performance of the proposed GSA with the WGGL model, three subsets (Leukemia_1, Leukemia_2 and Leukemia_3) are constructed by randomly selecting 1000, 1500 and 2000 genes from the preprocessed leukemia dataset. Each of the four datasets is randomly split into 43 groups of training data and 29 groups of test data for the two types of acute leukemia.

Brain cancer dataset [40] contains expression levels of 12625 genes of 50 gliomas samples: 28 glioblastomas and 22 anaplastic oligodendrogliomas. The initial dataset is available at: http://portals.broadinstitute.org/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=82. After preprocessing, only 4139 genes are left. 28 glioblastomas samples are labelled to 0 and 22 anaplastic oligodendrogliomas samples are labelled to 1. Similarly to the leukemia data, three subsets (Brain_1, Brain_2 and Brain_3) are constructed by randomly selecting 1000, 1500 and 2000 genes from the preprocessed brain cancer dataset. Each of the four datasets is randomly split into 33 samples for training and 17 samples for testing the performance of the diagnostic rule.

Ovarian cancer dataset [41] includes the expression profiles of 54675 genes based on the 12 samples from the resistant cohort and 16 samples from the sensitive cohort. The gene expression raw data files have been deposited to NCBI Gene Expression Omnibus (GEO accession GSE51373 at http://www.ncbi.nlm.nih.gov/projects/geo/). Only 3228 important genes are selected after the complete dataset preprocess. 16 sensitive cohort cancer samples are set as the 0 class and 12 resistant cohort samples as the 1 class. Similarly to the above two cases, three subsets (Ovarian_1, Ovarian_2 and Ovarian_3) are constructed by randomly selecting 1000, 1500 and 2000 genes from the preprocessed ovarian cancer dataset. Each of the four datasets is randomly split into 17 samples for training and 11 test samples.

Weighted gene co-expression networks are constructed for acute lymphoblastic leukemia data (ALL) and acute myeloid leukemia data (AML) in the leukemia dataset. This is also done for glioblastomas (GLI) and anaplastic oligodendrogliomas (OLI) in the brain cancer dataset and for sensitive cohort (SEN) and resistant cohort (RES) in the ovarian cancer dataset. All the datasets are grouped into different groups by GGH as shown in Table 2.

TABLE 2: Number of groups obtained by GGH on different datasets.

Dataset	Leukemia_1		Leukemia_2		Leukemia_3		Leukemia	
	ALL	AML	ALL	AML	ALL	AML	ALL	AML
Groups	7	10	9	11	9	12	10	17
Dataset	Brain_1		Brain_2		Brain_3		Brain	
	GLI	OLI	GLI	OLI	GLI	OLI	GLI	OLI
Groups	7	7	9	10	10	12	16	24
Dataset	Ovarian_1		Ovarian_2		Ovarian_3		Ovarian	
	SEN	RES	SEN	RES	SEN	RES	SEN	RES
Groups	10	10	11	10	11	13	16	16

6.2.2 Comparative Performance Analysis

GSA is an implementation of WGGL. Based on the calibrated parameters and components, we compare the proposed WGGL against GL and SGL on the above datasets. Since the samples are randomly selected, 10 replicates are performed on each dataset. We adopt two commonly used cancer classification performance evaluation indexes, the average misclassification error (AME) and the average number of genes selected (ANGS) on the 10 replicates for evaluation. In addition, average computation times are compared on each dataset. The experimental results are shown in Table 3.

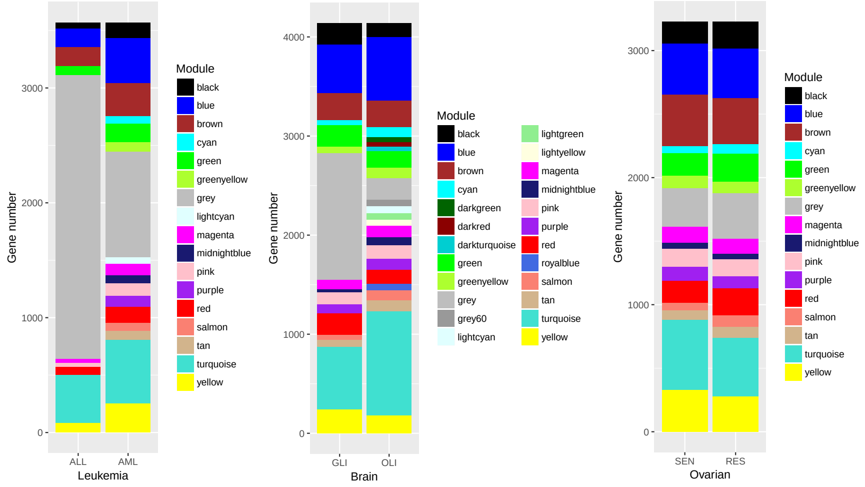
From Table 3, it can be observed that WGGL achieves obviously lower AMEs and significantly smaller ANGSs than SGL and GL on all datasets. For example, the average AME of WGGL is 0.110 which is much smaller than those of SGL and GL with 0.128 and 0.176, respectively, on the Leukemia subsets. The lower average AME of WGGL indicates the WGGL obtains the best classification performance on the four leukaemia datasets among the three models. In addition, WGGL obtains the least ANGSs on the four datasets among the three models. Even for the same AME on the Leukemia_2 subset, WGGL uses only 26.4 ANGS while SGL needs 35.9. In other words, WGGL performs better in classification and gene selection than SGL and GL on the invloved datasets. As for efficiency, Table 3 shows that WGGL always needs more computation time than SGL and GL. For example, WGGL needs about 278s while SGL and GL take 157s and 171s, respectively, on the leukemia dataset. The reason lies in that: (i) WGGL (by the GSA) needs more time for gene grouping, and (ii) group and gene weight determination by GGH and GGWC is much time-consuming. In any case, the computation time is perfectly acceptable.

It is well known that there is an imbalance in the positive and negative data sets in gene classification problems. Thus accuracy measurement at each class is important to provide further insight into the performance of each model. According to [47], we adopt Matthews correlation coefficient (MCC) which is used in machine learning as a measure of the quality of binary classifications. Mathematically, the formula of MCC is defined as: $MCC = \frac{TP\times TN - FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$, where TP, FP, TN, FN are the numbers of true positives, false positives, true negatives and false negatives, respectively. To avoid the one-time occasionality and ensure the validity of the test, we compute the average value of five-fold cross validation MCC across 10 trails. Table 4 reports the results of five-fold cross validation MCC on different datasets. It is shown that the WGGL achieves the highest MCC than other two models, which shows that WGGL is more accurate at each class in all cases than SGL and GL.

TABLE 3: Experimental results of the three models on different datasets.

Index	Dataset	GL	SGL	WGGL
AME	Leukemia_1	0.155	0.117	**0.097**
	Leukemia_2	0.168	**0.114**	**0.114**
	Leukemia_3	0.183	0.133	**0.108**
	Leukemia	0.196	0.146	**0.121**
	Average	0.176	0.128	**0.110**
	Brain_1	0.266	0.239	**0.161**
	Brain_2	0.205	0.191	**0.103**
	Brain_3	0.232	0.224	**0.153**
	Brain	0.302	0.276	**0.198**
	Average	0.251	0.233	**0.154**
	Ovarian_1	0.183	0.202	**0.129**
	Ovarian_2	0.193	0.187	**0.138**
	Ovarian_3	0.230	0.212	**0.147**
	Ovarian	0.246	0.226	**0.180**
	Average	0.213	0.207	**0.149**
ANGS	Leukemia_1	34.5	26.7	**19.8**
	Leukemia_2	39.7	35.9	**26.4**
	Leukemia_3	48.3	46.5	**32.9**
	Leukemia	63.6	52.8	**42.6**
	Brain_1	23.4	30.7	**16.4**
	Brain_2	43.4	32.8	**24.2**
	Brain_3	49.2	39.4	**33.8**
	Brain	67.3	54.2	**35.7**
	Ovarian_1	38.9	25.6	**22.6**
	Ovarian_2	44.3	34.8	**25.7**
	Ovarian_3	48.9	37.3	**35.2**
	Ovarian	58.8	49.5	**39.6**
Time (s)	Leukemia_1	45.08	37.13	48.69
	Leukemia_2	66.08	61.39	79.81
	Leukemia_3	66.35	62.24	86.98
	Leukemia	170.73	156.78	277.53
	Brain_1	41.46	36.51	49.75
	Brain_2	53.88	46.29	68.76
	Brain_3	64.61	53.47	89.60
	Brain	239.72	193.81	332.79
	Ovarian_1	39.96	32.94	42.43
	Ovarian_2	48.74	40.38	56.76
	Ovarian_3	60.21	51.53	76.12
	Ovarian	155.19	132.64	226.38

Details of the top 10 informative genes found by GL, SGL and WGGL for the leukaemia dataset are shown in Table 5. The biologically significant genes obtained by each classifier are indicated in bold. The biological experimental results proved some genes included in the frequently selected gene sets are mostly and functionally related to carcinogenesis or tumor histogenesis. For example, the most frequently selected gene set of WGGL, which include cystatin C (CST3) and myeloperoxidase (MPO) genes are proved experimentally to be correlated with ALL or AML leukemia. The gene CST3 is located at the extracellular region of the cell and has the role of invading human glioblastoma cells. The decrease in CST3 in CSF might contribute to the process of metastasis



(a) (b) (c)

Fig. 5: Results of network modules identified by GGH from the three cancer datasets.

TABLE 4: The results of five-fold cross validation MCC on different datasets.

Dataset	GL	SGL	WGGL
Leukemia_1	0.817	0.819	**0.916**
Leukemia_2	0.810	0.829	**0.884**
Leukemia_3	0.796	0.814	**0.854**
Leukemia	0.765	0.809	**0.863**
Brain_1	0.549	0.637	**0.724**
Brain_2	0.560	0.701	**0.796**
Brain_3	0.540	0.616	**0.705**
Brain	0.494	0.551	**0.651**
Ovarian_1	0.683	0.592	**0.720**
Ovarian_2	0.608	0.615	**0.714**
Ovarian_3	0.530	0.562	**0.664**
Ovarian	0.500	0.543	**0.629**

and the spread of cancer cells in the leptomeningeal tissues [43]. Matsuo et al. [44] believed that the percentage of MPO-positive blast cells was the most simple and useful factor in predicting a prognosis for AML patients in this category. Genes CST3, MPO, PTX3 and IGL are selected in the grey module in AMLNet. Genes CST3, MPO, IGL, MEF2C and KIT are selected in the blue module in ALLNet. Genes DF, IGB, TCL1 and PYGL are selected in the different groups respectively. In particular, note that gene groups CST3, MPO and IGL are highly correlated with the occurrence of leukaemia. Compared with the other two models, more important genes are frequently selected by WGGL.

Table 6 depicts details of the top 10 informative genes found by the three models for the brain cancer dataset. Similar to those of the leukaemia dataset, we observe that the most frequently selected genes by WGGL are the g-lypican (GPC1) and protein tyrosine phosphatase, receptor-type, zeta polypeptide (PTPRZ) genes. These genes are the top two significant informative genes ranked by the proposed WGGL model and they are highly related to brain cancer. For example, multiple proteoglycan core proteins and related enzymes have been differentially expressed in glioblastoma tumors relative to normal brains. These genes (including genes GPC1 and PTPRZ [45]) promote tumor cell invasion or tumor development. The first ranked gene selected by WGGL is GPC1, of which the gene function is in accordance with the result given by Whipple et al. [46]. It was reported that the increase expression of GPC1 on tumor cells or on tumor-associated endothelial cells is associated with alterations in RTK signaling and promoting tumorigenesis in brain, breast, and pancreatic cancer. In particular, note that GPC1, PTPRZ and PTCH genes are highly correlated with brain cancer.

Table 7 shows the top 10 ranked informative genes found by the three models for the ovarian cancer dataset. It can be observed that the most frequently selected genes by WGGL involve the insulin-like growth factor 1 (IGF1), the insulin-like growth factor 2 (IGF2) and the insulin receptor (INSR). The first ranked gene selected by WGGL is the insulin-like growth factor 1 (IGF1) gene. Koti et al. [41] showed that IGF1 potentially acts as one of the key signalling pathways which is involved in the development of intrinsic chemotherapy resistance in ovarian cancer. Some genes are frequently selected by WGGL while they are not discovered by the other two models. WGGL always selects genes which are more relevant to ovarian cancer classification than SGL and GL. For example, the CDKN2C and NGFRAP1 genes are less important than the IGF1 gene as shown in [41]. However, the IGF1 and IGFBP3 genes selected by WGGL are highly correlated with ovarian cancer for the same group.

The results of network modules identified by GGH from the three cancer datasets are given in column stacking diagram Fig. 5. The total number of genes in the ordinate is stacked by the number of genes in different modules. From this figure we observe that the group sizes are not even for the leukemia and brain two datasets. Especially, the size of biggest group (grey module) is 2472 is far larger than the smallest group (magenta module) whose size is 34 for leukemia dataset. The size of biggest group (grey module) is 1274 is far larger than the smallest group (midnightblue module) whose size is 31 for brain dataset. The results of WGGL is based on the GGH and SGL and GL is based on the GG in Table 5, 6 and 7. WGGL can still work better than the other two models on the unevenly sized groups. To illustrate this, we report the top 10 ranked informative

TABLE 5: Top 10 ranked informative genes selected by the three models from the leukaemia dataset.

Rank	Gene description		
	WGGL	SGL	GL
1	**CST3 cystatin C (amyloid angiopathy and cerebral hemorrhage)**	ADA adenosine deaminase	RNS2 ribonuclease 2 (eosinophil-derived neurotoxin; EDN)
2	**MPO myeloperoxidase**	TTF mRNA for small G protein	MB-1 gene
3	DF D component of complement (adipsin)	**MPO myeloperoxidase**	**IGL immunoglobulin lambda light chain**
4	**IGB immunoglobulin-associated beta (B29)**	IGHM immunoglobulin mu	KIT V-kit Hardy-Zuckerman 4 feline sarcoma viral oncogene homolog
5	PTX3 pentaxin-related gene, rapidly induced by IL-1 beta	LYZ lysozyme	CFD complement factor D (adipsin)
6	**IGL immunoglobulin lambda light chain**	PR264 gene	CD19 gene
7	**TCL1 gene (T cell leukemia) extracted from H.sapiens mRNA for Tcell leukemia/lymphoma 1**	**MEF2C MADS box transcription enhancer factor 2, polypeptide C (myocyte enhancer factor 2C)**	**IGB immunoglobulin-associated beta (B29)**
8	PYGL glycogen phosphorylase L (liver form)	RNS2 ribonuclease 2 (eosinophil-derived neurotoxin; EDN)	IGHM immunoglobulin mu
9	**MEF2C MADS box transcription enhancer factor 2, polypeptide C (myocyte enhancer factor 2C)**	**IGB immunoglobulin-associated beta (B29)**	**MEF2C MADS box transcription enhancer factor 2, polypeptide C (myocyte enhancer factor 2C)**
10	KIT V-kit Hardy-Zuckerman 4 feline sarcoma viral oncogene homolog	**IGL immunoglobulin lambda light chain**	MANB mannosidase alpha-B (lysosomal)

TABLE 6: Top 10 ranked informative genes selected by the three models from the brain cancer dataset.

Rank	Gene description		
	WGGL	SGL	GL
1	**GPC1 human mRNA for heparan sulfate proteaglycan (glypican)**	PAGA H.sapiens mRNA for tetranectin (plasminogen-binding protein)	SH3 domain-containing protein SH3P17 mRNA
2	**PTPRZ protein tyrosine phosphatase, receptor-type, zeta polypeptide**	**PTCH patched (Drosophila) homolog**	**PKD2 autosomal dominant polycystic kidney disease type II**
3	**N-MYC oncogene protein mRNA**	**HMG2 high-mobility group (nonhistone chromosomal) protein 2**	CENPB centromere protein B (80kD)
4	GCSH glycine cleavage system protein H (aminomethyl carrier)	CSNK1D human casein kinase I delta mRNA	**HMG2 high-mobility group (nonhistone chromosomal) protein 2**
5	**PTCH patched (Drosophila) homolog**	GUSB human beta-glucuronidase mRNA	ORF mRNA
6	**PKD2 autosomal dominant polycystic kidney disease type II**	**PTPRZ protein tyrosine phosphatase, receptor-type, zeta polypeptide**	PAGA H.sapiens mRNA for tetranectin (plasminogen-binding protein)
7	RBP1 cellular retinol-binding protein mRNA	MHC class I region proline rich protein mRNA	KIAA0115 gene
8	APXL apical protein (Xenopus laevis-like)	KIAA0045 gene	**PTCH patched (Drosophila) homolog**
9	**HMG2 high-mobility group (nonhistone chromosomal) protein 2**	**GPC1 human mRNA for heparan sulfate proteaglycan (glypican)**	MHC class I region proline rich protein mRNA
10	ADM homo sapiens mRNA for adrenomedullin precursor	DLX7 distal-less homeobox 7	**GPC1 human mRNA for heparan sulfate proteaglycan (glypican)**

TABLE 7: Top 10 ranked informative genes selected by the three models from the ovarian cancer dataset.

Rank	Gene description		
	WGGL	SGL	GL
1	**IGF1 insulin-like growth factor 1 (somatomedin C)**	**CDKN2C cyclin-dependent kinase inhibitor 2C (p18, inhibits CDK4)**	IGLC1 immunoglobulin lambda constant 1 (Mcg marker)
2	**MYC v-myc avian myelocytomatosis viral oncogene homolog**	IGLV1-44 immunoglobulin lambda variable 1-44	**NGFRAP1 nerve growth factor receptor (TNFRSF16) associated protein 1**
3	**IGF2 insulin-like growth factor 2 (somatomedin A)**	GBP1 guanylate binding protein 1, interferon-inducible	IGKC immunoglobulin kappa constant
4	ZFP36 ring finger protein	**IGF2 insulin-like growth factor 2 (somatomedin A)**	MRI1 methylthioribose-1-phosphate isomerase 1
5	NFKBIA nuclear factor of kappa light polypeptide gene enhancer in B-cells inhibitor, alpha	IGJ immunoglobulin J polypeptide, linker protein for immunoglobulin alpha and mu polypeptides	**IGFBP3 insulin-like growth factor binding protein 3**
6	**INSR insulin receptor**	IGKC immunoglobulin kappa constant	GJB2 gap junction protein, beta 2, 26kDa
7	PLP2 proteolipid protein 2 (colonic epithelium-enriched)	**NGFRAP1 nerve growth factor receptor (TNFRSF16) associated protein 1**	**IGF2 insulin-like growth factor 2 (somatomedin A)**
8	**IGFBP3 insulin-like growth factor binding protein 3**	GUSBP11 glucuronidase, beta pseudogene 11	**INSR insulin receptor**
9	PSMB9 proteasome (prosome, macropain) subunit, beta type, 9	**IGFBP3 insulin-like growth factor binding protein 3**	APLP2 amyloid beta (A4) precursor-like protein 2
10	**OAT ornithine aminotransferase**	**OAT ornithine aminotransferase**	INHBA inhibin, beta A

genes selected by SGL and GL models based on the GGH on the three cancer datasets in Table 8.

After adopting the GGH, SGL selects 49.6 ANGS and GL selects 59.1 ANGS on the leukaemia dataset. The number of biologically significant genes frequently selected by the SGL and GL models based on the GGH in the first two columns of Table 8 are still less than those by WGGL in Table 5. More specially, gene TCL1 is selected by WGGL model which is in the smallest magenta module in the networks which are constructed for ALL, but the other two models cannot select this gene. Similarly, SGL selects 48.7 ANGS and GL selects 64.9 ANGS based on GGH on the brain dataset. We observe that the frequently selected genes N-MYC by WGGL in Table 6 which is not selected by the other two models based on GGH in middle two columns of Table 8. This gene is contained in the smallest midnightblue module. Finally, SGL selects 41.3 ANGS and GL selects 50.8 ANGS based on GGH on the ovarian dataset. Genes IGF1 and MYC are frequently selected by WGGL in Table 7 while they are not discovered by the other two models based on GGH in the last two columns of Table 8. Gene IGF1 in magenta module and gene MYC in blue module, which are small sized groups. Therefore, the proposed WGGL model can achieve better gene selection performance on the unevenly sized groups as compared with GL and SGL models.

TABLE 8: Top 10 ranked informative genes selected by the SGL and GL models based on the GGH on the three cancer datasets.

Rank	Leukaemia		Brain		Ovarian	
	SGL	GL	SGL	GL	SGL	GL
1	CD36	**IGB**	**PKD2**	PAX8A	**IGF2**	**OAT**
2	**CST3**	AQP3	CENPB	**PTCH**	**IGFBP3**	IGKC
3	**MPO**	IGHM	**PTPRZ**	**HMG2**	ITPR3	**IGFBP3**
4	ZYX	ANX1	**GPC1**	STPKC2K	TMC6	RNF139
5	**IGB**	CD36	DRAP1	**GPC1**	**OAT**	**IGF2**
6	**IGL**	**IGL**	PSP31	CENPB	**INSR**	ZFP36
7	CD24	**MEF2C**	**HMG2**	PRB2	CPVL	**INSR**
8	Epb72	ZYX	NRF1	PSP31	SCAMP1	TMC6
9	**MEF2C**	GB	BMP-2A	**PKD2**	**CDKN2C**	**NGFRAP1**
10	IGHM	**MPO**	**PTCH**	NRF1	RNF139	CPVL

7 Conclusion

In this paper, the weighted general group Lasso for gene selection in cancer classification has been proposed and a new gene selection algorithm has also been developed. The network-based system biology approach is introduced to the constructed sparse group Lasso. Weighted gene co-expression network analysis is applied to identify important network modules for cancer datasets and group genes. Biologically significant gene and group weights are calculated by joint mutual information. Experimental results on benchmark instances show that the proposed model and algorithm are more suitable for classification and gene selection than existing models.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No. 61572127), the Key Research & Development program in Jiangsu Province (No. BE2015728) and Collaborative Innovation Center of Wireless Communications Technology. Rubén Ruiz is partially supported by the Spanish Ministry of Economy and Competitiveness, under the project "SCHEYARD – Optimization of Scheduling Problems in Container Yards" (No. DPI2015-65895-R) financed by FEDER funds.

References

[1] T. Gloub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield, and E. Lander, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring". *Science*, vol, 286, no. 5439, pp. 531-537, 1999.

[2] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machine," *Machine Learning*, vol. 46, no. 1, pp. 389-422, 2002.

[3] Z. Y. Algamal and M. H. Lee, "Penalized logistic regression with the adaptive LASSO for gene selection in high-dimensional cancer classification," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9326-9332, 2015.

[4] T. Nguyen and S. Nahavandi, "Modified AHP for gene selection and cancer classification using type-2 fuzzy logic," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 2, pp. 273-287, 2016.

[5] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491-502, 2005.

[6] Z. Yu, L. Li, J. Liu, and G. Han, "Hybrid adaptive classifier ensemble," *IEEE Transactions on Cybernetics*, vol. 45, no. 2, pp. 177-190, 2015.

[7] L. Wang and J. H. Zhu, "Hybrid huberized support vector machines for microarray classification and gene selection," *Bioinformatics*, vol. 24, no. 3, pp. 412-419, 2008.

[8] Y. Tian, Z. Qi, X. Ju, Y. Shi, and X. Liu, "Nonparallel support vector machines for pattern classification," *IEEE Transactions on Cybernetics*, vol. 44, no, 7, pp. 1067-1079, 2014.

[9] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society*, vol. 58, no. 1, pp. 267-288, 1996.

[10] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society*, vol. 67, no. 2, pp. 301-320, 2005.

[11] O. Arslan, "Weighted LAD-LASSO method for robust parameter estimation and variable selection in regression," *Computational Statistics and Data Analysis*, vol. 56, no. 6, pp. 1952-1965, 2012.

[12] J. Wang, C. Lu, M. Wang, P. Li, S. Yan, and X. Hu, "Robust face recognition via adaptive sparse representation," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2368-2378, 2014.

[13] W. Yang, Y. Gao, Y. Shi, and L. Cao, "MRM-Lasso: a sparse multiview feature selection method via low-rank analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2801-2815, 2015.

[14] S. Zheng and W. Liu, "An experimental comparison of gene selection by Lasso and Dantzig selector for cancer classification," *Computers in Biology and Medicine*, vol. 41, no. 11, pp. 1033-1040, 2011.

[15] G. C. Cawley and N. L. C. Talbot, "Gene selection in cancer classification using sparse logistic regression with Bayesian regularisation," *Bioinformatics*, vol. 22, no. 19, pp. 2348-2355, 2006.

[16] B. Krishnapuram, L. Carin, M. A. Figueiredo, and A. J. Hartemink, "Sparse multinomial logistic regression: fast algorithms and generalization bounds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 957-968, 2005.

[17] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society, Series B*, vol. 68, no. 1, pp. 49-67, 2006.

[18] L. Meier, S. van de Geer, and P. Buhlmann, "The group lasso for logistic regression," *Journal of the Royal Statistical Society Series B*, vol. 70, no. 1, pp. 53-71, 2008.

[19] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231-245, 2013.

[20] H. Zou, "The adaptive Lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101 no. 476, pp. 1418-1429, 2006.

[21] K. Fang, X. Wang, S. Zhang, and J. Z. S. Ma, "Bi-level variable selection via adaptive sparse group lasso," *Journal of Statistical Computation and Simulation*, vol. 85, no. 13, pp. 1-11, 2014.

[22] M. Vincent and N. R. Hansen, "Sparse group lasso and high dimensional multinomial classification," *Computational Statistics and Data Analysis*, vol. 71, no. 1, pp. 771-786, 2014.

[23] J. M. Stuart, E. Segal, D. Koller, and S. K. Kim, "A gene-coexpression network for global discovery of conserved genetic modules," *Science*, vol. 302, no. 5643, pp. 249-255, 2003.

[24] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, pp. 1-45, 2005.

[25] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A.-L. Barabsi, "Hierarchical organization of modularity in metabolic networks," *Science*, vol. 297, no. 5586, pp. 1551-1555, 2002.

[26] A. M. Yip and S. Horvath, "Gene network interconnectedness and the generalized topological overlap measure," *BMC Bioinformatics*, 8:22, 2007.

[27] P. Langfelder, B. Zhang, and S. Horvath, "Defining clusters from a hierarchical cluster tree: the Dynamic Tree Cut package for R," *Bioinformatics*, vol. 24, no. 5, pp. 719-720, 2008.

[28] N. K. MacLennan and J. J. Dong, "Weighted gene co-expression network analysis identifies biomarkers in glycerol kinase deficient mice," *Molecular Genetics and Metabolism*, vol. 98, no. 1-2, pp. 203-214, 2009.

[29] B. Guo and M. S. Nixon, "Gait feature subset selection by mutual information," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 39, no. 1, pp. 36-46, 2008.

[30] P. Maji, "Mutual information-based supervised attribute clustering for microarray sample classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 1, pp. 127-140, 2010.

[31] Q. Hu, W. Pan, L. Zhang, D. Zhang, Y. Song, M. Guo, and D. Yu, "Feature selection for monotonic classification," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 1, pp. 69-81, 2011.

[32] J. Yang and C. Ong, "An effective feature selection method via mutual information estimation" *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 6, pp. 1550-1559, 2012.

[33] Q. Qiu, V. M. Patel, and R. Chellappa, "Information-theoretic dictionary learning for image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2173-2184, 2014.

[34] H. Yang and J. Moody, "Feature selection based on joint mutual information," *Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis, Rochester, New York*, pp.22-25, 1999.

[35] M. Sehhati, A. Mehridehnavi, H. Rabbani, and M. Pourhossein, "Stable gene signature selection for prediction of breast cancer recurrence using joint mutual information," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 12, no. 6, pp. 1440-1448, 2015.

[36] M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using joint mutual information maximisation," *Expert Systems With Applications*, vol. 42, no. 22, pp. 8520-8532, 2015.

[37] G. Brown, A. Pocock, M. J. Zhao, and M. Luján, "Conditional likelihood maximisation: a unifying framework for information theoretic feature selection," *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 27-66, 2012.

[38] T. M. Cover and J. A. Thomas, Elements of Information Theory. NewYork: Wiley, 1991.

[39] K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano, "Reverse engineering of regulatory networks in human B cells," *Nature Genetics*, vol. 37, no. 4, pp. 382-390, 2005.

[40] C. L. Nutt, D. R. Mani, R. A. Betensky, P. Tamayo, J. G. Cairncross, et al., "Gene expression-based classification of malignant gliomas correlates better with survival than histological classification," *Cancer Research*, vol. 63, no. 7, pp. 1602-1607, 2003.

[41] M. Koti, R. J. Gooding, P. Nuin, A. Haslehurst, C. Crane, et al., "Identification of the IGF1/PI3K/NFκB/ERK gene signalling networks associated with chemotherapy resistance and treatment response in high-grade serous epithelial ovarian cancer," *BMC Cancer*, vol. 13, no. 1, pp. 1-11, 2013.

[42] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation", *Advances in Information Retrieval*, vol. 3408, pp. 345-359, 2005.

[43] A. Nagai, M. Terashima, T. Harada, K. Shimode, H. Takeuchi, et al, "Cathepsin B and H activities and cystatin C concentrations in cerebrospinal fluid from patients with leptomeningeal metastasis". *Clinica Chimica Acta*, vol. 329, nos. 1-2, pp. 53-60, 2003.

[44] T. Matsuo, K. Kuriyama, Y. Miyazaki, S. Yoshida, M. Tomonaga, et al, "The percentage of myeloperoxidase-positive blast cells is a strong independent prognostic factor in acute myeloid leukemia, even in the patients with normal karyotype". *Leukemia*, vol. 17, no. 8, pp. 1538-1543, 2003.

[45] A. Wade, A. E. Robinson, J. R. Engler, C. Petritsch, C. D. James, et al, "Proteoglycans and their roles in brain cancer". *Febs Journal*, vol. 280, no. 10, pp. 2399-2417, 2013.

[46] C. A. Whipple, A. L. Young, and M. Korc, "A Kras(G12D)-driven genetic mouse model of pancreatic cancer requires glypican-1 for efficient proliferation and angiogenesis". *Oncogene*, vol. 31, pp. 2535-44, 2011.

[47] B. W. Matthews. "Comparison of the predicted and observed secondary structure of T4 phage lysozyme". *Biochimica Et Biophysica Acta*, vol. 405, no. 2, pp. 442-451, 1975.

Yadi Wang is a Phd candidate in the School of Computer Science and Engineering at Southeast University. She received B.S. degree in the School of Mathematics and Information Sciences from Henan Polytechnic University and M.S. degree in the School of Mathematics and Information Sciences at Henan Normal University. Her current research interests include machine learning and data mining.

Xiaoping Li (M'09-SM'12) received his B.Sc. and M.Sc. degrees in Applied Computer Science from the Harbin University of Science and Technology in 1993 and 1999, respectively, and the Ph.D. degree in Applied Computer Science from the Harbin Institute of Technology in 2002. He is a full professor at the School of Computer Science and Engineering, Southeast University, Nanjing, China. He is the author or co-author over more than 100 academic papers, some of which have been published in international journals such as *IEEE Transactions on Services Computing*, *IEEE Transactions on Automation Science and Engineering*, *IEEE Transactions on Cybernetics*, *IEEE Transactions on Systems, Man and Cybernetics: Systems*, *Information Sciences*, *Omega*, *European Journal of Operational Research*, *International Journal of Production Research*, *Expert Systems with Applications* and *Journal of Network and Computer Applications*. His research interests focus on Scheduling in Cloud Computing, Scheduling in Cloud Manufacturing, Machine Scheduling, Project Scheduling, Terminal Container Scheduling.

Rubén Ruiz received the B.Sc. and M.Sc. degrees in computer science engineering from the Universitat Politècnica de València in 1998 and 2000, respectively, and the Ph.D. degree in statistics and operations research from the same university in 2003. He is a full professor of Statistics and Operations Research at the Polytechnic University of Valencia, Spain. He is co-author of more than 60 papers in International Journals and has participated in presentations of more than a hundred and fifty papers in national and international conferences. He is editor of the Elsevier's journal *Operations Research Perspectives (ORP)* and co-editor of the JCR-listed journal *European Journal of Industrial Engineering (EJIE)*. He is also associate editor of other important journals like *TOP* or *Applied Mathematics and Computation* as well as member of the editorial boards of several journals most notably *European Journal of Operational Research* and *Computers and Operations Research*. He is the director of the Applied Optimization Systems Group (SOA, http://soa.iti.es) at the Instituto Tecnológico de Informática (ITI, http://www.iti.es) where he has been principal investigator of several public research projects as well as privately funded projects with industrial companies. His research interests include scheduling and routing in real life scenarios.