computers

MDPI

Article

GFLASSO-LR: Logistic Regression with Generalized Fused LASSO for Gene Selection in High-Dimensional Cancer Classification

Ahmed Bir-Jmel [1,2,*], Sidi Mohamed Douiri [3], Souad El Bernoussi [3], Ayyad Maafiri [4], Yassine Himeur [2,*], Shadi Atalla [2], Wathiq Mansoor [2] and Hussain Al-Ahmad [2]

1 Research Center STIS, M2CS, Department of Applied Mathematics and Informatics, ENSAM, Mohammed V University in Rabat, Rabat 10000, Morocco
2 College of Engineering and Information Technology, University of Dubai, Dubai P.O. Box 14143, United Arab Emirates; satalla@ud.ac.ae (S.A.); wmansoor@ud.ac.ae (W.M.); halahmad@ud.ac.ae (H.A.-A.)
3 Laboratory of Mathematics, Computer Science & Applications-Security of Information, Department of Mathematics, Faculty of Sciences, Mohammed V University in Rabat, Rabat 10000, Morocco; douirisidimohamed@gmail.com (S.M.D.); souad.elbernoussi@gmail.com (S.E.B.)
4 Engineering Science Laboratory, National School of Applied Sciences (ENSA), Ibn Tofail University, Kenitra 14000, Morocco; maafiri.ayyad@gmail.com
* Correspondence: a.birjmel@um5r.ac.ma (A.B.-J.); yhimeur@ud.ac.ae (Y.H.)

Abstract: Advancements in genomic technologies have paved the way for significant breakthroughs in cancer diagnostics, with DNA microarray technology standing at the forefront of identifying genetic expressions associated with various cancer types. Despite its potential, the vast dimensionality of microarray data presents a formidable challenge, necessitating efficient dimension reduction and gene selection methods to accurately identify cancerous tumors. In response to this challenge, this study introduces an innovative strategy for microarray data dimension reduction and crucial gene set selection, aiming to enhance the accuracy of cancerous tumor identification. Leveraging DNA microarray technology, our method focuses on pinpointing significant genes implicated in tumor development, aiding the development of sophisticated computerized diagnostic tools. Our technique synergizes gene selection with classifier training within a logistic regression framework, utilizing a generalized Fused LASSO (GFLASSO-LR) regularizer. This regularization incorporates two penalties: one for selecting pertinent genes and another for emphasizing adjacent genes of importance to the target class, thus achieving an optimal trade-off between gene relevance and redundancy. The optimization challenge posed by our approach is tackled using a sub-gradient algorithm, designed to meet specific convergence prerequisites. We establish that our algorithm's objective function is convex, Lipschitz continuous, and possesses a global minimum, ensuring reliability in the gene selection process. A numerical evaluation of the method's parameters further substantiates its effectiveness. Experimental outcomes affirm the GFLASSO-LR methodology's high efficiency in processing high-dimensional microarray data for cancer classification. It effectively identifies compact gene subsets, significantly enhancing classification performance and demonstrating its potential as a powerful tool in cancer research and diagnostics.

Keywords: gene selection; cancer classification; DNA microarray; penalized logistic regression; Generalized Fused LASSO; sub-gradient method

Citation: Bir-Jmel, A.; Douiri, S.M.; Bernoussi, S.E.; Maafiri, A.; Himeur, Y.; Atalla, S.; Mansoor, W.; Al-Ahmad, H. GFLASSO-LR: Logistic Regression with Generalized Fused LASSO for Gene Selection in High-Dimensional Cancer Classification. *Computers* **2024**, *13*, 93. https://doi.org/10.3390/computers13040093

Academic Editor: Hersh Sagreiya Sagreiya

Received: 26 February 2024
Revised: 26 March 2024
Accepted: 1 April 2024
Published: 6 April 2024

1. Introduction

Currently, advancements in technology have made massive datasets, like DNA microarray data, fundamental to statistical analysis [1,2]. High-dimensional data pose challenges in classification tasks and gene subset identification [3]. DNA microarrays are vital in medicine and biology for various research purposes, including gene co-regulation, clinical diagnosis, and differential gene expression [4,5]. Microarray technology, crucial in

cancer research, identifies genes critical to tumor growth and aids in developing diagnostic systems [6]. However, microarray data analysis is complex due to computational and biological challenges. The high cost of microarray experiments limits sample size, creating datasets with many genes but few samples [7,8], leading to computational instability and the curse of dimensionality [9,10]. Consequently, gene selection becomes a crucial yet difficult task.

Scientists aim to identify markers for lab testing to create effective disease treatments. Efficient gene selection improves understanding of gene–disease relationships and classifier quality, addressing the curse of dimensionality [11–13]. Gene selection, a significant challenge due to the gene-sample number disparity, requires advanced optimization methods [14,15]. Despite numerous gene selection methods aiming to enhance classification accuracy, further research is needed, particularly for diseases like cancer [16,17]. High classification accuracy is essential for personalized medicine, enabling better physician decision-making and potentially saving lives. Gene [18,19] selection algorithms include filter, wrapper, and embedded methods [20].

Filter methods evaluate genes independently, assigning scores to select high-scoring genes. Though easy to implement, they overlook gene interactions. Examples include the Fischer score, t-test, signal-to-noise ratio, information gain, and ReliefF [21–25]. Wrapper methods use classifiers to select gene subsets, testing candidate subsets for performance. Despite achieving high performance, they are computationally intensive and risk overfitting [20]. Embedded methods integrate selection into the learning algorithm, offering efficiency and performance without extensive classifier execution [26]. These methods consider gene dependencies but are algorithm-specific and computationally complex [27–29].

Filter methods evaluate genes independently, assigning scores to select high-scoring genes. Though easy to implement and computationally efficient, they have significant limitations: primarily, they overlook gene interactions, which can lead to suboptimal gene selection in complex biological processes where gene interactions play a crucial role. Examples include the Fischer score, t-test, signal-to-noise ratio, information gain, and ReliefF. Their simplicity and neglect of gene–gene interactions can lead to a lack of robustness in capturing the multifaceted nature of genetic influences on diseases [21–25]. Wrapper methods, on the other hand, use classifiers to select gene subsets, evaluating the performance of candidate subsets. While these methods can achieve high performance by considering gene interactions and utilizing the predictive power of classifiers, they come with their own set of limitations. They are computationally intensive, making them impractical for very large datasets, and they carry a significant risk of overfitting, especially when the number of genes vastly exceeds the number of samples. This can lead to models that perform well on training data but poorly on unseen data [20]. Embedded methods offer a middle ground by integrating gene selection directly into the learning algorithm, thus offering efficiency and potentially high performance without necessitating the execution of separate classifiers for feature selection. These methods can account for gene dependencies through their integration with the learning model, offering a more nuanced approach to gene selection. However, their algorithm-specific nature means that each method is tailored to a particular model or set of assumptions about the data, which can limit their applicability. Moreover, they can still be computationally complex, posing challenges for their use in large-scale genomic studies [27–29]. These limitations highlight the ongoing need for innovative solutions that balance accuracy, computational efficiency, and the ability to handle high-dimensional genomic data.

Regularized methods, particularly penalized logistic regression (PLR) with LASSO, are gaining traction for cancer classification [30–38]. While LASSO is effective for feature selection and model simplification, it has limitations such as potential bias in coefficient estimation, difficulty in handling highly correlated predictors, and a tendency to select only one variable from a group of highly correlated variables. These limitations prompted the development of the Elastic-Net (EN) penalty by Zou and Hastie [39], which combines the L_1 and L_2 penalties to better handle correlated predictors, and the adaptive LASSO

(ALASSO) by Zou et al. [40], which introduces weights for penalization within the L_1 norm to improve variable selection accuracy.

To overcome the deficiencies in achieving the oracle property, Zou introduced ALASSO [40], which theoretically possesses the oracle property under certain conditions. However, the performance of ALASSO can be sensitive to the choice of weights, which may require additional tuning. Structured penalties, like Fused LASSO [41,42], further enhance LASSO by considering gene relationships, aiding in feature reduction and meaningful gene selection [43]. Despite its advantages, Fused LASSO can be computationally intensive for large datasets and may not perform well if the true underlying model does not adhere to its assumption of spatial or sequential coherence. The Fused LASSO and its theoretical foundations [44,45] have led to the Generalized Fused LASSO (GFLASSO) within linear regression [46,47], which penalizes coefficient differences based on a gene graph to encourage sparse and connected solution vectors [47]. GFLASSO further extends the applicability to more complex data structures but faces challenges in scalability and the need for a well-defined gene graph, which may not always be available or accurately represent biological relationships.

The L_1 norm penalties' absolute values complicate minimization due to non-differentiability at zero, challenging analytical solution derivation. Initial solutions for linear regression included FLSA [48] and other algorithms [49–51]. For logistic regression, fewer algorithms exist, with Hofling et al. [46] applying quadratic approximation iteratively to the log-likelihood function. Table 1 presents a comparison of Contributions in DNA Microarray Data Analysis.

In the wake of comprehensive literature analysis, it becomes evident that the frontier of gene selection for cancer research is ripe for innovation. Current methodologies, while robust, exhibit limitations in computational efficiency, model overfitting, and the nuanced understanding of gene interdependencies. This literature underscores the essential balance between gene selection's computational demands and the need for precise, interpretable models capable of navigating the vast, complex landscape of genomic data. Notably, regularized regression methods like LASSO and its derivatives represent a significant stride towards addressing these challenges. However, their effectiveness is contingent upon the integration of gene relational structures and overcoming inherent mathematical complexities.

This study introduces an innovative algorithm, termed GFLASSO, which leverages sub-gradient methods and unfolds in two distinct phases: initially, a preprocessing step where a preliminary subset of genes is selected via a univariate filter method that utilizes Pearson's correlation coefficient; followed by the application of an improved Generalized Fused Lasso technique, which also employs Pearson's correlation coefficient to assess inter-gene relationships. The primary objective of this research is to showcase the effectiveness of our gene selection methodology in enhancing the classification of high-dimensional data, particularly within the context of cancer research. Overall, the main contributions of this study are summarized as follows:

- Introduced an improved approach for dimension reduction and important gene selection in cancer research using DNA microarray technology.
- Integrated gene selection with classifier training into a single process, enhancing the efficiency of identifying cancerous tumors.
- Formulated gene selection as a logistic regression problem with a generalized Fused LASSO (GFLASSO) regularizer, incorporating dual penalties to balance gene relevance and redundancy.
- Utilized a sub-gradient algorithm for optimization, demonstrating the algorithm's objective function is convex, Lipschitzian, and possesses a global minimum, meeting necessary convergence conditions.
- Showed that the GFLASSO-LR method significantly improves cancer classification from high-dimensional microarray data, yielding compact and highly performant gene subsets.

The remainder of this paper is structured as follows: Section 2 provides a detailed presentation of the proposed method, beginning with preliminary concepts before delving into the sub-gradient method and its application for estimating $\widehat{\beta}_{\text{GFLASSO}}$. Subsequently, Section 3 evaluates the performance of our proposed approach using four well-known gene expression classification datasets, detailing both the parameter settings and the numerical results obtained. Finally, Section 4 concludes the paper and outlines directions for future work.

Table 1. Comparison of Contributions in DNA Microarray Data Analysis.

Reference	Methodological Approach	Application Area	Challenges Addressed	Notable Contributions
Li et al. [1]	High-dimensional data analysis	DNA microarray data	Statistical analysis difficulty in classification tasks	Developed approaches for gene subset identification
Feng et al. [2]	High-dimensional data analysis	DNA microarray data	High-dimensional data challenges in gene selection	Improved understanding of gene–disease relationships
Mehrabi et al. [3]	Efficient gene selection methods	DNA microarray data	Curse of dimensionality in microarray data analysis	Enhanced classification accuracy for diseases like cancer
Syu et al. [4]	Microarray technology analysis	Medicine and Biology	Clinical diagnosis and gene co-regulation	Identified critical genes for tumor growth and diagnostic systems development
Caraffi et al. [5]	Microarray data analysis	Cancer research	Computational and biological challenges	Developed diagnostic systems to aid in cancer treatment
Ghavidel et al. [7]	Machine learning approaches	Microarray data analysis	Computational instability and sample size limitation	Addressed the curse of dimensionality with computational methods
Birjmel et al. [12]	Optimization methods for gene selection	DNA microarray data	Gene-sample number disparity	Advanced optimization methods to improve classifier quality
Yaqoob et al. [16]	Gene selection for personalized medicine	Cancer research	Need for high classification accuracy	Optimized gene selection methods for better physician decision-making
Alharthi et al. [30]	Regularized methods, PLR with LASSO	Cancer classification	Limitations of LASSO	Introduced Elastic-Net (EN) and adaptive LASSO (ALASSO) for cancer classification
Zou et al. [40]	Adaptive LASSO	Statistical analysis	Oracle property deficiencies in LASSO	Introduced weights for penalization within the L_1 norm
Hofling et al. [46]	Generalized Fused LASSO (GFLASSO)	Linear regression	Challenges in L_1 norm penalties minimization	Penalized coefficient differences based on a gene graph for sparse solutions

2. Methods

2.1. Preliminaries

In Microarray data, each value of x_{ij} is the measure of the expression level of the j^{th} gene in the i^{th} sample, this array contains real values, and there are no missing values. Where N is the number of genes and M is the number of samples.

DNA microarray analysis for cancer classification is formulated as a supervised classification problem. The limited number of samples that can be collected due to the high cost of preparing DNA microarrays presents a challenge for statistical learning techniques. Linear models, specifically logistic regression, are commonly used for discriminant analysis

in supervised learning. In binary logistic regression, it is usual to consider the values of the variable to be explained as belonging to the set $\{0,1\}$; this is a binary classification.

Let $\Pi_i = \mathbb{P}(y_i = 1/x_i = x_{i1}, \ldots, x_{iN})$ denote the probability that the class of x_i is 1, for any sample x_i, $i = 1, \ldots, M$, of the dataset matrix.

It can then be deduced that $\mathbb{P}(y_i = 0/x_i = x_{i1}, \ldots, x_{iN}) = 1 - \Pi_i$. The logistic regression model is written [52] as follows:

$$\Pi_i = \frac{e^{\beta_0 + \sum_{j=1}^{N} \beta_j x_{ij}}}{1 + e^{\beta_0 + \sum_{j=1}^{N} \beta_j x_{ij}}}$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_N)^T$ denotes the vector of regression coefficients including a constant β_0. This vector will be estimated from the training data. Without loss of generality, it was assumed that the genes are centered:

$\sum_{i=1}^{M} x_{ij} = 0$ and $\frac{1}{M} \sum_{i=1}^{M} x^2{}_{ij} = 1, \quad \forall j \in \{1, 2, \ldots, N\}$.

For clarity and simplicity, let us adopt the following notation. Let $i = 1, 2, \ldots, M$ denote

- $x_i' = (1, x_i) = (1, x_{i1}, \ldots, x_{iN})$, or simply for any vector $\mathbf{x} = (x_1, \ldots, x_N)$ of $\mathbb{R}^N$, it was noted that $x' = (1, x) = (1, x_1, \ldots, x_N)$
- $x_i'\beta = \beta_0 + \sum_{j=1}^{N} \beta_j x_{ij}$ is the usual scalar product.

Moving on, for $i = 1, \ldots, M$,

$$\Pi_i = \frac{e^{x_i'\beta}}{1 + e^{x_i'\beta}} \qquad \text{et} \qquad \ln\left(\frac{\Pi_i}{1 - \Pi_i}\right) = x_i'\beta$$

The probability of observing the response y_i for a sample x_i is written in a more compact way:

$$\mathbb{P}(y_i/x_i) = \Pi_i^{y_i}(1 - \Pi_i)^{1-y_i}$$

Let f be the "sigmoid" function used in the logistic regression defined by $f(z) = \frac{1}{1+e^{-z}}$ for any real z; the latter has a value in $[0,1]$. It can be noticed then that $\Pi_i = f(x_i'\beta)$.

The classification rule for a new example x via logistic regression is therefore defined by

$$Class(x) = \begin{cases} 0 & \text{if } f(x'\beta) < 0.5 \\ 1 & \text{if } f(x'\beta) \geqslant 0.5 \end{cases}$$

The methods for estimating the β coefficients include the statistical method called "maximum likelihood estimation", which allows us to determine the values of the β parameter of the model, which renders the following maximum value

$$L(\beta) = \prod_{i=1}^{M} \Pi_i^{y_i}(1 - \Pi_i)^{1-y_i}$$

This is equivalent to finding β in $\mathbb{R}^{N+1}$, which maximizes the log-likelihood function.

$$\ln(L(\beta)) = \sum_{i=1}^{M} y_i \ln(\Pi_i) + (1 - y_i) \ln(1 - \Pi_i)$$

Maximizing $\ln(L(\beta))$ is the same as minimizing $-\ln(L(\beta))$. Let us then note $g(\beta) = -\ln(L(\beta))$ for all β of $\mathbb{R}^{N+1}$. After some elementary calculations, we arrive at

$$g(\beta) = -\sum_{i=1}^{M} \left(y_i x_i'\beta - \ln\left(1 + e^{x_i'\beta}\right) \right)$$

In cancer classification, logistic regression is used to model the relationship between a set of independent variables (genes) and a binary dependent variable (tumor or normal class) by minimizing the function $g(\beta)$ in Equation (1).

$$\hat{\beta}_{LR} = \arg\min_{\beta \in \mathbb{R}^{N+1}} g(\beta) = \arg\min_{\beta \in \mathbb{R}^{N+1}} -\sum_{i=1}^{M} (y_i x_i' \beta - \ln\left(1 + e^{x_i' \beta}\right)) \quad (1)$$

However, logistic regression may be limited in classifying high-dimensional data as the minimization problem may not have a unique solution, as shown by Albert et al. [53]. They found that when the dataset is completely or nearly completely separated, the estimators of (1) become infinite and non-unique, while in the presence of overlap in the data, the minimum of (1) exists and is finite.

In microarray data, the high number of genes can lead to overfitting and multicollinearity in the estimators (β) since only a few genes are actually associated with the response variable. To enhance classification accuracy, it is important to develop gene selection methods that eliminate irrelevant genes. One such method is penalized logistic regression (PLR), which adds a positive penalty to the likelihood function g, causing some coefficients to approach zero, a technique known as regularization. The penalized log-likelihood can be expressed as shown in Equation (2):

$$PLR = g(\beta) + \lambda P(\beta) \quad (2)$$

where $P(\beta)$ represents the penalty term and λ is the regularization parameter that determines the extent of the penalty. For $\lambda = 0$, this can lead to the solution of (1) if it exists. On the other hand, for high values of λ, the regularization term has a greater impact on the coefficient estimates. The selection of the tuning parameter is critical in fitting the model, and if it is chosen via cross-validation, the classifier can achieve satisfactory classification accuracy. These penalized methods are commonly used in gene selection and classification of high-dimensional data, as reported in [54].

In the following, we denote $\widehat{\beta}_\lambda$ as the vector of size $N + 1$ that represents a solution to the minimization problem (2), i.e.,

$$\widehat{\beta}_\lambda = \arg\min_{\beta \in \mathbb{R}^{N+1}} (g(\beta) + \lambda P(\beta)) \quad (3)$$

Several penalty terms have been explained and applied in the literature, namely, Ridge (L_2) [55], LASSO (L_1) [56], Elastic net [39], Adaptive LASSO [40], Fused LASSO and Generalized Fused LASSO [47], Relaxed LASSO [57], Group LASSO [58], Random LASSO [59], and many others. These penalties, along with logistic regression, can be used successfully to obtain high classification rates. We are specifically interested in the Generalized Fused LASSO penalty. This focus sets the stage for introducing the sub-gradient method.

2.2. Sub-Gradient Method

The sub-gradient method is a technique for minimizing non-differentiable convex functions, similar to the ordinary gradient method for differentiable functions. However, it has some key differences, such as its ability to handle non-differentiable functions directly and the lack of guarantee for decreasing the function value.

These methods were introduced in the 1960s by N.Z. Shor [60] for the unconstrained minimization of convex functions in the field of network transport. The first convergence results are attributed to Polyak [61] and Nemirovski [62], and they have been further developed by various researchers since then.

Consider a function $f : \Omega \to \mathbb{R}$, where $\Omega \subset \mathbb{R}^n$ is a convex set, and $\bar{x} \in \Omega$. We define a **subgradient** of f at $\bar{x}$ as an element $v \in \mathbb{R}^n$ satisfying the following inequality:

$$\langle v, x - \bar{x} \rangle \leq f(x) - f(\bar{x}) \text{ for all } x \in \Omega$$

The set of all subgradients is termed the **subdifferential** of f at $\bar{x}$, denoted $\partial f(\bar{x})$.

For a convex function $f : \Omega \to \mathbb{R}$ defined on a convex set $\Omega \subset \mathbb{R}^n$, several properties of its subdifferential have been well-established [63]:

1. Let $\bar{x} \in \text{int}(\Omega)$, then $\partial f(\bar{x})$ is a non-empty, convex, bounded set.
2. Assuming differentiability of f at $\bar{x} \in \text{int}(\Omega)$, we have $\partial f(\bar{x}) = \nabla f(\bar{x})$ and

$$\langle \nabla f(\bar{x}), x - \bar{x} \rangle \leq f(x) - f(\bar{x}) \text{ for all } x \in \Omega$$

3. Suppose Ω is a convex set in $\mathbb{R}^n$, and let $f_i : \Omega \to \mathbb{R}$ be convex functions with $\lambda_i > 0$ for $i = 1, \ldots, m$. If $\bar{x} \in \Omega$, then the subdifferential sum rule is expressed as follows:

$$\partial \left(\sum_{i=1}^{m} \lambda_i f_i \right) (\bar{x}) = \sum_{i=1}^{m} \lambda_i \partial f_i(\bar{x})$$

Let us now present some important examples of sub-differential calculation:

Example 1. The norm function (L_1):
Let $f : x \in \mathbb{R}^n \mapsto \|x\|_1 = |x_1| + \ldots + |x_n|$.
The sub-differential of f *is the Cartesian product of the following intervals:*

$$\partial f(x) = I_1 \times \cdots \times I_n, \quad I_k = \begin{cases} [-1,1] & x_k = 0 \\ \{1\} & x_k > 0 \\ \{-1\} & x_k < 0 \end{cases}$$

Example 2. *Let* n *be an integer greater than or equal to* 2, *and* $(k,l) \in 1, n^2$ *with* $k < l$.
Let $f_{k,l}$ *be the function defined on* $\mathbb{R}^n$ *by* $f_{k,l} : x \in \mathbb{R}^n \mapsto |x_k - x_l|$
The subdifferential of $f_{k,l}$ *is the Cartesian product of the following intervals:*

$$\partial f_{k,l}(x) = I_1 \times \cdots \times I_n.$$

With:

$$I_i = \{0\} \text{ for all } i \in 1, n \setminus \{k,l\}, I_k = \begin{cases} [-1,1] & x_k - x_l = 0 \\ \{1\} & x_k - x_l > 0 \\ \{-1\} & x_k - x_l < 0 \end{cases}, I_l = \begin{cases} [-1,1] & x_k - x_l = 0 \\ \{1\} & x_k - x_l < 0 \\ \{-1\} & x_k - x_l > 0 \end{cases}$$

2.2.1. Subgradient Algorithm

The classical subgradient method is designed to solve convex optimization problems without type constraints:

$$\text{minimize } f(x) \text{ subject to } x \in \mathbb{R}^n \tag{4}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ represents a convex function. Let $\{\alpha_k\}$, where $k \in \mathbb{N}$, be a sequence of positive numbers. The subgradient algorithm generated by $\{\alpha_k\}$ is defined as follows: given an initial point $x_1 \in \mathbb{R}^n$, the iterative procedure is as follows:

$$x_{k+1} := x_k - \alpha_k v_k \quad \text{with} \quad v_k \in \partial f(x_k), \quad k \in \mathbb{N} \tag{5}$$

We presuppose that problem (4) possesses an optimal solution and that f is convex and Lipschitz on $\mathbb{R}^n$.

Several methodologies have been proposed in the literature, as documented in [61,63–67], for selecting the sequence of steps α_k to ensure convergence of the subgradient method. Note that

$$\bar{V} := \min\{f(x) \mid x \in \mathbb{R}^n\} \quad \text{and} \quad V_k := \min\{f(x_1), \ldots, f(x_k)\}$$

There are three main results that have been found:

1. Suppose $\alpha_k = \epsilon$ for all $k \in \mathbb{N}$. Then there exists a positive constant C such that

$$0 \leq V_k - \bar{V} < \ell^2 \epsilon \quad \text{when} \quad k > \frac{C}{\epsilon^2}$$

2. Suppose that

$$\sum_{k=1}^{\infty} \alpha_k = \infty \quad \text{and} \quad \lim_{k\to\infty} \alpha_k = 0 \tag{6}$$

Then, we have convergence $V_k \to \bar{V}$ as $k \to \infty$, and

$$\liminf_{k\to\infty} f(x_k) = \bar{V}$$

3. Suppose the generated sequence $\{\alpha_k\}$ in (5) satisfies

$$\sum_{k=1}^{\infty} \alpha_k = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty \tag{7}$$

Consequently, the sequence $\{V_k\}$ converges to the optimal value $\bar{V}$, and the iterative sequence $\{x_k\}$ generated by (5) converges to an optimal solution $\bar{x}$ of (4).

2.2.2. Generalized Fused LASSO

Proposed by Tibshirani and Al [42], the Fused LASSO is an extension of LASSO aiming to improve its properties when the features are ordered. It adds another penalty to L_1 to encourage zero differences between successive coefficients. The idea of the Fused LASSO has been generalized in the context of linear regression models (and generalized linear models) [46,47]. This is accomplished by constructing a graph $\mathcal{G} = (V, E)$ with $V = \{1, \ldots, N\}$ vertices (representing genes in DNA microarray data), each corresponding to a coefficient, and a set of edges E. In the generalized Fused LASSO, differences $|\beta_k - \beta_l|$ for $(k, l) \in E$ are penalized, with associated weights representing the degree of association between genes. The objective function is

$$\begin{aligned} \widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2} = \arg \min_{\beta\in\mathbb{R}^{N+1}} g(\beta) + \lambda_1 \sum_{k=0}^{N} |\beta_k| \\ + \lambda_2 \sum_{(k,l)\in E, k<l} w_{k,l} |\beta_k - \beta_l| \end{aligned} \tag{8}$$

The penalties $w_{k,l}$ in the Fused LASSO regularizer (8) are typically determined from the initial data. Here, $\lambda_{k,l} = \lambda_2 \times w_{k,l}$ represents the penalty applied to the difference between genes g_k and g_l. As $w_{k,l}$ increases, the penalty on $|\beta_k - \beta_l|$ also increases, making β_k and β_l more likely to be equal. The first penalty ensures many β_j are zero, promoting sparsity. Therefore, $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ tends to have sparse and equal coefficients for connected genes in the graph. The choice of λ_1 and λ_2 is determined using appropriate cross-validation methods [68].

Due to the penalty (8), some coefficients of $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ are reduced to zero, indicating unused genes in the model. The parameters λ_1 and λ_1 control the degree of penalization, affecting the number of non-zero estimated coefficients. As λ_1 increases, the number of null components in $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ increases. The set of selected genes is defined as

$$\mathcal{S}(\lambda_1, \lambda_2) = \left\{ g_j : \widehat{\beta}_{GFLASSO,\lambda_1,\lambda_2,j} \neq 0; j = 1, \ldots, N \right\}$$

where $\widehat{\beta}_{GFLASSO,\lambda_1,\lambda_2,j}$ is the penalized logistic regression coefficient (8) associated with the g_j gene for the penalty parameters λ_1 and λ_2.

We formulate the problem of gene selection as a logistic regression problem (maximum log-likelihood) with a generalized Fused LASSO regularizer, a penalty that is rarely used in this type of problem. The selection is performed by minimizing (8).

To construct the graph $\mathcal{G} = (V, E)$ and assign weights $w_{k,l}$ to each edge (k, l), we consider $V = \{1, \ldots, N\}$ as the set of vertices (genes) associated with our DNA microarray data. The interaction between genes is measured using the Pearson correlation coefficient. An edge is formed between vertices k and l if and only if the absolute correlation between genes g_k and g_l is greater than or equal to a threshold r_0. The weight of edge (k, l) is defined as the absolute value of the correlation between g_k and g_l: $w_{k,l} = |\rho_{g_k,g_l}|$. This way, the graph $\mathcal{G} = (V, E)$ and the weights $w_{k,l}$ are established. The correlation coefficient between g_k and g_l is defined as

$$\rho_{g_k,g_l} = \frac{\text{cov}(g_k, g_l)}{\sigma_{g_k}\sigma_{g_l}} \tag{9}$$

where $\text{cov}(g_k, g_l)$ is the covariance between g_k and g_l, σ_{g_k} is the standard deviation of g_k, and σ_{g_l} is the standard deviation of g_l.

The use of Fused LASSO is motivated by several factors [69]. One of the main advantages is its ability to handle high-dimensional data where the number of genes exceeds the number of samples. Traditional LASSO may struggle in such cases, while Elastic Net, though potentially outperforming LASSO, is relatively less efficient than Fused LASSO, especially when there are relationships among genes. Additionally, Fused LASSO explicitly regularizes the differences between neighboring coefficients using an L_1 norm regularizer, exploiting relationships between genes. This results in sparse coefficients, with non-zero components equal for some connected genes in the graph, balancing individual gene relevance and redundancy.

2.2.3. Structure of the Proposed Approach

The proposed approach consists of a two-step sequential process (as depicted in Figure 1). Firstly, a preprocessing phase eliminates irrelevant genes, selecting the top genes ranked by relevance determined by their correlation with the target class. Then, an embedded method is applied to the selected genes, involving the construction of a logistic regression classifier. The model is trained to find $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ by solving the minimization problem outlined in Equation (8) using the subgradient algorithm.

With respect to the estimation of $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ in (8), since the function in question is not differentiable due to the presence of the absolute value, we use the subgradient algorithm to solve the minimization problem (8). This algorithm has not been previously used to solve the logistic regression problem with the Generalized Fused LASSO penalty. In the following, we demonstrate the convergence of this algorithm for (8).

2.3. Sub-Gradient for Estimating $\widehat{\beta}_{GFLASSO,\lambda_1,\lambda_2}$

It is known that the sub-gradient algorithm can converge to the optimal solution of a convex minimization problem when an appropriate step sequence (α_k) is chosen, as long as the function in question meets the following conditions:

- It is convex.
- It admits a global minimum.
- It is Lipschitz.

To apply this algorithm to our problem (8), we must first ensure that the function meets these three conditions.

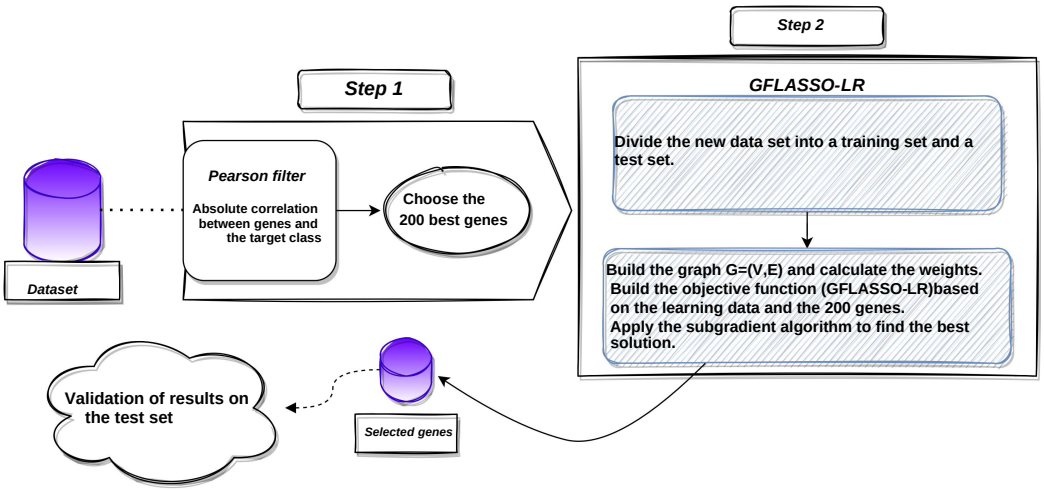

Figure 1. Flowchart of the proposed approach ($GFLASSO-LR$) for gene selection in microarray data.

2.3.1. Characteristics and Convexity Analysis of the GFLASSO Penalty Function

It is important to note that

$$h_{\lambda_1,\lambda_2}(\beta) = g(\beta) + \lambda_1 \sum_{j=0}^{N} |\beta_j| + \lambda_2 \sum_{(k,l)\in E, k<l} w_{k,l}|\beta_k - \beta_l| \quad \forall \beta \in \mathbb{R}^{N+1} \tag{10}$$

1. **The convexity of h_{λ_1,λ_2}:**

   First, there is

$$\begin{aligned} g(\beta) &= -\sum_{i=1}^{M} \left(y_i x_i' \beta - \ln\left(1 + e^{x_i'\beta}\right)\right) \\ &= -\sum_{i=1}^{M} y_i x_i' \beta + \sum_{i=1}^{M} \ln\left(1 + e^{x_i'\beta}\right), \ \forall \beta \in \mathbb{R}^{N+1} \end{aligned}$$

   Let us show that g is convex on $\mathbb{R}^{N+1}$:

   - The function $\beta \mapsto -\sum_{i=1}^{M} y_i x_i' \beta$ is convex; it is a linear function on $\mathbb{R}^{N+1}$.
   - The function $\beta \mapsto x_i'\beta$ is convex for all i in $\{1, \ldots, M\}$, so $\beta \mapsto \ln\left(1 + e^{x_i'\beta}\right)$ is convex for all i in $\{1, \ldots, M\}$. (It is the composition of an increasing function convex on $\mathbb{R}$ and a convex function on $\mathbb{R}^{N+1}$).

   The function g is therefore **convex** on $\mathbb{R}^{N+1}$, being the sum of $M+1$ convex functions. On the other hand, the function $\beta \mapsto \lambda_1 \sum_{j=0}^{N} |\beta_j|$ is convex as the sum of $N+1$ convex functions.
   Next, we show that the function $\beta \mapsto \lambda_2 \sum_{(k,l)\in E, k<l} w_{k,l}|\beta_k - \beta_l|$ is convex.
   Indeed, let $(k,l) \in \{1, \ldots, N\}^2$ with $k < l$. The function $\beta \mapsto \lambda_2 w_{k,l}|\beta_k - \beta_l| = f_{k,l}(\beta)$ is convex. This is because $f_{k,l}$, defined in Example (2), is composed of an affine function and another convex one, with $\lambda_2 w_{k,l} \geq 0$. Then the function $\beta \mapsto \lambda_2 \sum_{(k,l)\in E, k<l} w_{k,l}|\beta_k - \beta_l|$ is convex as it is a sum of convex functions.
   Finally, we can deduce that the function h_{λ_1,λ_2} is convex as it is a sum of convex functions.

2. **The existence of a global minimum of h_{λ_1,λ_2}:**
   We recall that a real-valued function $f : \mathbb{R}^n \to \mathbb{R}$ is coercive if $\lim_{|\mathbf{x}|\to\infty} f(\mathbf{x}) = \infty$.
   We have

$$\forall \beta \in \mathbb{R}^{N+1}, \quad L(\beta) = \prod_{i=1}^{M} \Pi_i^{y_i} (1 - \Pi_i)^{1-y_i}$$

   and

$$g(\beta) = -\ln(L(\beta))$$

   with $\Pi_i = \frac{e^{x_i'\beta}}{1+e^{x_i'\beta}}$ for all i in $\{1, \ldots, M\}$.

Additionally, we have
$\forall i \in [|1, M|], 0 < \Pi_i < 1$ *and* $y_i \in \{0, 1\}$. So $0 < L(\beta) < 1 \qquad \forall \beta \in \mathbb{R}^{N+1}$
Hence, $g(\beta) \in \mathbb{R}^{+*} \qquad \forall \beta \in \mathbb{R}^{N+1}$
Let us examine the expression for h_{λ_1,λ_2}. Given that the function g is positive on $\mathbb{R}^{N+1}$ and λ_1 in (10) is strictly positive, it follows that

$$h_{\lambda_1,\lambda_2}(\beta) \geqslant \lambda_1 \|\beta\|_1 \qquad \forall \beta \in \mathbb{R}^{N+1}$$

Based on the latest inequality, we can deduce that h_{λ_1,λ_2} is a coercive function. Additionally, h_{λ_1,λ_2} is continuous, so it attains a **a global minimum** at least once (see appendix in reference [70]).

3. **The function h_{λ_1,λ_2} is ℓ-Lipschitz on $\mathbb{R}^{N+1}$:**
We only have one condition left to establish to show that the sub-gradient algorithm converges to a global minimum of h_{λ_1,λ_2}; it is to prove that h_{λ_1,λ_2} is ℓ-Lipschitzian on $\mathbb{R}^{N+1}$, i.e., find a positive ℓ such that

$$\forall \beta, \gamma \in \mathbb{R}^{N+1}, \qquad \left|h_{\lambda_1,\lambda_2}(\beta) - h_{\lambda_1,\lambda_2}(\gamma)\right| \leqslant \ell \|\beta - \gamma\|$$

Note in passing that the Lipschitz constant ℓ depends on the choice of the norm on $\mathbb{R}^{N+1}$. Since all norms are equivalent on $\mathbb{R}^{N+1}$, whether a function is Lipschitz or not does not depend on the chosen norm.
Let us start by showing that the function g is Lipschitz. We first have

$$g(\beta) = -\sum_{i=1}^{M}\left(y_i x_i'\beta - \ln\left(1 + e^{x_i'\beta}\right)\right) \quad \forall \beta \in \mathbb{R}^{N+1}$$

The function g is of class C^1 on $\mathbb{R}^{N+1}$. Find the partial derivatives of g:
Let $\beta \in \mathbb{R}^{N+1}$. We have

$$\forall j \in [|0, N|], \qquad \frac{\partial g(\beta)}{\partial \beta_j} = -\sum_{i=1}^{M}\left(y_i x_{ij}' - \Pi_i x_{ij}'\right)$$

Therefore,
$\|\nabla g(\beta)\|_1 \leq \sum_{j=0}^{N}\left(\sum_{i=1}^{M}\left(\left|y_i x_{ij}'\right| + \left|x_{ij}'\right|\right)\right)$
Let $M_1 = \sum_{j=0}^{N}\left(\sum_{i=1}^{M}\left(\left|y_i x_{ij}'\right| + \left|x_{ij}'\right|\right)\right)$. Then we have $\|\nabla g(\beta)\|_1 \leq M_1$
As we are working on the finite dimensional normed vector space $\mathbb{R}^{N+1}$, there then exists some positive M_2 such that $\|\nabla g(\beta)\| \leq M_2$
We can notice that the real M_2 does not depend on β; more precisely it only depends on the training set. So we get

$$\forall \beta \in \mathbb{R}^{N+1}, \qquad \|\nabla g(\beta)\| \leq M_2 \tag{11}$$

So g is a M_2-Lipschitz function on $\mathbb{R}^{N+1}$ by the mean value inequality.
In the following, we will prove that the function h_{λ_1,λ_2} is Lipschitz.
Let $\beta, \gamma \in \mathbb{R}^{N+1}$. We have

$$\begin{array}{c} |\lambda_1 \sum_{j=0}^{N}|\beta_j| - \lambda_1 \sum_{j=1}^{N}|\gamma_j|| = \lambda_1 |\sum_{j=0}^{N}(|\beta_j| \\ -|\gamma_j|)| \leq \lambda_1 \sum_{j=0}^{N}||\beta_j| - |\gamma_j|| \leq \lambda_1 \|\beta - \gamma\|_1 \end{array}$$

Similarly, since the weights $w_{k,l} \in [0, 1]$,

$$\begin{aligned}
&\left|\lambda_2 \sum_{(k,l)\in E,k<l} w_{k,l}|\beta_k-\beta_l| - \lambda_2 \sum_{(k,l)\in E,k<l} w_{k,l}|\gamma_k-\gamma_l|\right| \\
&= \lambda_2 \left| \sum_{(k,l)\in E,k<l} w_{k,l}(|\beta_k-\beta_l| - |\gamma_k-\gamma_l|)\right| \\
&\leq \lambda_2 \sum_{(k,l)\in E,k<l} |(\beta_k-\beta_l)-(\gamma_k-\gamma_l)| \\
&\leq \lambda_2 \frac{N(N-1)}{2}\|\beta-\gamma\|_1
\end{aligned}$$

Thereby, $\forall \beta,\gamma \in \mathbb{R}^{N+1}$, $\quad |\lambda_1 \sum_{j=0}^{N}|\beta_j| - \lambda_1 \sum_{j=0}^{N}|\gamma_j|| \leq \lambda_1\|\beta-\gamma\|_1$ and

$$\begin{aligned}
&\forall \beta,\gamma \in \mathbb{R}^{N+1}, \\
&\quad |\lambda_2 \textstyle\sum_{(k,l)\in E,k<l} w_{k,l}|\beta_k-\beta_l| \\
&-\lambda_2 \textstyle\sum_{(k,l)\in E,k<l} w_{k,l}|\gamma_k-\gamma_l|| \leq \lambda_2 \frac{N(N-1)}{2}\|\beta-\gamma\|_1
\end{aligned}$$

By norm equivalence, we can deduce that there are some positive M_3 and M_4 such that

$$\forall \beta,\gamma \in \mathbb{R}^{N+1}, \ |\lambda_1 \sum_{j=0}^{N}|\beta_j| - \lambda_1 \sum_{j=0}^{N}|\gamma_j|| \leq M_3\|\beta-\gamma\| \tag{12}$$

$$\begin{aligned}
&\forall \beta,\gamma \in \mathbb{R}^{N+1}, \quad |\lambda_2 \textstyle\sum_{(k,l)\in E,k<l} w_{k,l}|\beta_k-\beta_l| \\
&-\lambda_2 \textstyle\sum_{(k,l)\in E,k<l} w_{k,l}|\gamma_k-\gamma_l|| \leq M_4\ |\beta-\gamma\|
\end{aligned} \tag{13}$$

It follows from (11), (12) and (13) that

$$\begin{aligned}
&\forall \beta,\gamma \in \mathbb{R}^{N+1}, \ |h_{\lambda_1,\lambda_2}(\beta) \\
&-h_{\lambda_1,\lambda_2}(\gamma)| \leqslant (M_2+M_3+M_4)\|\beta-\gamma\|
\end{aligned}$$

Finally, it suffices to take $\ell := M_2+M_3+M_4$ to deduce that h_{λ_1,λ_2} **is ℓ-lipschitzian on** $\mathbb{R}^{N+1}$**.**

2.3.2. Sub-Gradient Algorithm

After confirming the convergence of our sub-gradient algorithm for estimating β, let us now detail the method for the objective function h_{λ_1,λ_2}. The sub-differential sum rule states that $\partial h_{\lambda_1,\lambda_2}(\beta) = \partial g(\beta) + \lambda_1 \partial||\beta|1| + \lambda 2 \sum_{(k,l)\in E,k<l} w_{k,l}\partial f_{k,l}(\beta)$. In this equation, $f_{k,l}$ is a function defined in the Example (2), and g is differentiable on $\mathbb{R}^{N+1}$; so $\partial g(\beta) = \nabla g(\beta)$. The subdifferentials of the norm L_1, and $f_{k,l}$ are provided in the Examples 1 and 2, respectively, which leads to the final expression

$$\begin{aligned}
&\forall \beta \in \mathbb{R}^{N+1}, \partial h_{\lambda_1,\lambda_2}(\beta) = \nabla g(\beta) + \lambda_1 \partial|\|\beta\|_1| \\
&+\lambda_2 \textstyle\sum_{(k,l)\in E,k<l} w_{k,l}\partial f_{k,l}(\beta)
\end{aligned}$$

Thus, for a well-chosen sequence (α_k), the search algorithm for $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$ for problem (8) is as follows:

The use of the generalized Fused LASSO penalty in our classification model enables an embedded gene selection process. Genes associated with null coefficients in β^* have no impact on the classification of new examples, as the classification rule in logistic regression is determined by the value of $\frac{1}{1+e^{-x^t\beta^*}}$. If this value is less than 0.5, the example is classified as 0; otherwise, it is classified as 1. This approach fosters a more efficient and effective classification model by eliminating irrelevant genes.

3. Experiments and Results

This section presents the performance of our proposed approach on four well-known gene expression classification datasets (see Table 2). The parameter settings and numerical results will be described in the following subsections. The implementation of our proposed approach was conducted using MATLAB R2018a, and the datasets were divided into training and test sets to assess the performance of gene subset selection.

Table 2. Description of datasets (DNA microarray) used in GFLASSO-LR.

Dataset	# Genes	# Samples	# Classes	Source
Colon	2000	62	2	[71]
DLBCL	5469	77	2	[72]
Prostate_Tumor	10,509	102	2	[72]
Prostate	12,600	102	2	[73]

3.1. Datasets

To evaluate our approach, we selected DNA microarray datasets related to cancer recognition, which are publicly available. These datasets have been utilized in multiple supervised classification studies (as shown in Table 2). They exhibit a variety of characteristics: some have a low number of samples, while others have a higher number. Additionally, all datasets have binary classes. Since our proposed method is designed for large-scale microarrays, all datasets consist of a high number of genes, ranging from 2000 to 12,600.

3.2. Settings

In our approach, we begin with a pre-processing step, which involves selecting the top 200 highest-ranked genes using the absolute correlation coefficient. Subsequently, we randomly split our dataset into a training set and a test set. Using the training set, we then construct our objective function (8). To minimize this function, we employ the sub-gradient algorithm (Algorithm 1). It is important to note that the performance of this algorithm is highly dependent on the choice of step size (α_k). A well-chosen (α_k) can significantly impact both the convergence rate and the quality of the estimate obtained.

Algorithm 1: Subgradient algorithm to minimize h_{λ_1,λ_2}.

Function: $GFLASSO - LR(X, it_{max}, \lambda_1, \lambda_2, r_0, (\alpha_k))$ **Inputs:** X: Training set, it_{max}: the maximum number of iterations, λ_1 and λ_2: the coefficients that control the penalties, r_0: the threshold for building an edge in the graph $\mathcal{G} = (V, E)$, (α_k): the sequence that controls the steps in the subgradient algorithm.
Output: β^*: an approximate value of $\widehat{\beta}_{\text{GFLASSO},\lambda_1,\lambda_2}$.

Construct the graph $\mathcal{G} = (V, E)$ and calculate the weights $w_{k,l}$
k = 1;
Choose β^1 randomly from $\mathbb{R}^{N+1}$
$\beta^* = \beta^1$, $h^*_{\lambda_1,\lambda_2} = h_{\lambda_1,\lambda_2}(\beta^1)$
$k = 2$ **while** $k < it_{max}$ **do**
    $\beta^k = \beta^{k-1} - \alpha_k * v_{k-1}$ with v_{k-1} a randomly chosen vector of $\partial h_{\lambda_1,\lambda_2}(\beta^{k-1})$ **if** $h_{\lambda_1,\lambda_2}(\beta^k) \leq h^*_{\lambda_1,\lambda_2}$ **then**
$\beta^* = \beta^k$, $h^*_{\lambda_1,\lambda_2} = h_{\lambda_1,\lambda_2}(\beta^k)$.
$k = k + 1$.
Return β^*.

3.2.1. Choice of (α_k)

The choice of an appropriate (α_k) step for the sub-gradient algorithm is an essential step in our study. To begin, we set $\lambda_1 = 1$, $\lambda_2 = 0.2$, and $r_0 = 0.7$. Then, the "Prostate_Tumor" dataset is utilized to conduct a parametric study.

We start by selecting the 200 best-ranked genes and split the resulting dataset into two disjoint sets: 80% for training and 20% for testing. Our objective function is constructed based on the training set. The goal of this subsection is to choose a step size (α_k) that accelerates the convergence of our algorithm to the global minimum. At this stage, we are not concerned with classifier testing, as our focus is solely on observing the behavior of the sub-gradient algorithm in finding the minimum of h_{λ_1,λ_2}.

We initially compared the classic step $\alpha_k = \frac{1}{k}$ to the step proposed by Alber et al. in [74], $\alpha'_k = \frac{1}{k} * \frac{1}{\max(1,|v_{k-1}|)}$, where v_{k-1} is a randomly chosen vector of $\partial h_{\lambda_1,\lambda_2}\left(\beta^{k-1}\right)$ (as outlined in Algorithm 1). We set a maximum number of iterations at 10,000. It is evident that these steps ensure the convergence of our algorithm from the sub-gradient to the global minimum.

From the results depicted in Figure 2, it is evident that using α'_k accelerates the convergence of the sub-gradient algorithm. This acceleration is attributed to the inclusion of the term v_{k-1} in α'_k.

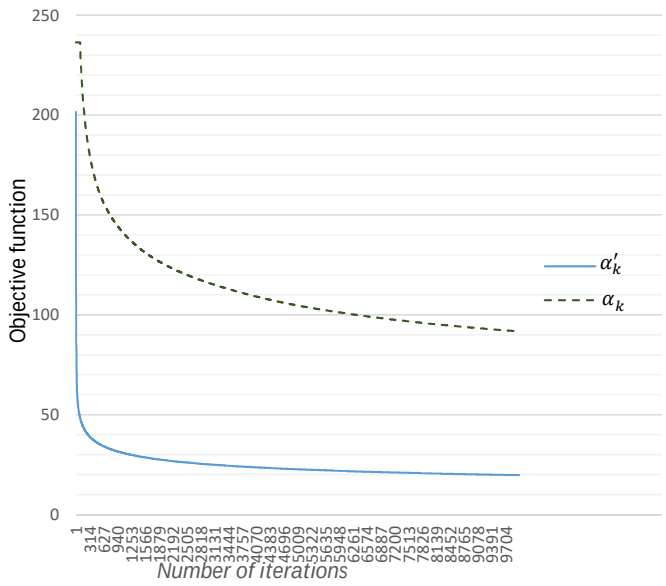

Figure 2. Comparison between the steps α_k and α'_k to minimize the function $h_{\lambda 1,\lambda_2}$.

To determine the optimal step size, we drew inspiration from the work of Alber et al. in [74] and utilized steps of the form $\alpha'_k = \alpha_k * \frac{1}{\max(1,\ |v_{k-1}|)}$, where v_{k-1} represents a randomly chosen vector of $\partial h_{\lambda_1,\lambda_2}\left(\beta^{k-1}\right)$ and (α_k) denotes a sequence of positive terms satisfying

$$\sum_{k=1}^{\infty} \alpha_k = \infty \text{ and } \lim_{k\to\infty} \alpha_k = 0$$

The final steps ensure the convergence of the subgradient algorithm. In the following, we compare different forms of the step (α_k), specifically those of the form $\frac{1}{k^\gamma}$ with $\gamma \in [0.05, 1]$. Figure 3 illustrates how the objective function changes with the number of iterations for different values of γ. To enhance the interpretability of the data, we have plotted it on a logarithmic scale. Additionally, Table 3 presents the optimal values of the objective function h_{λ_1,λ_2} obtained using the subgradient algorithm with different steps α'_k for various values of γ.

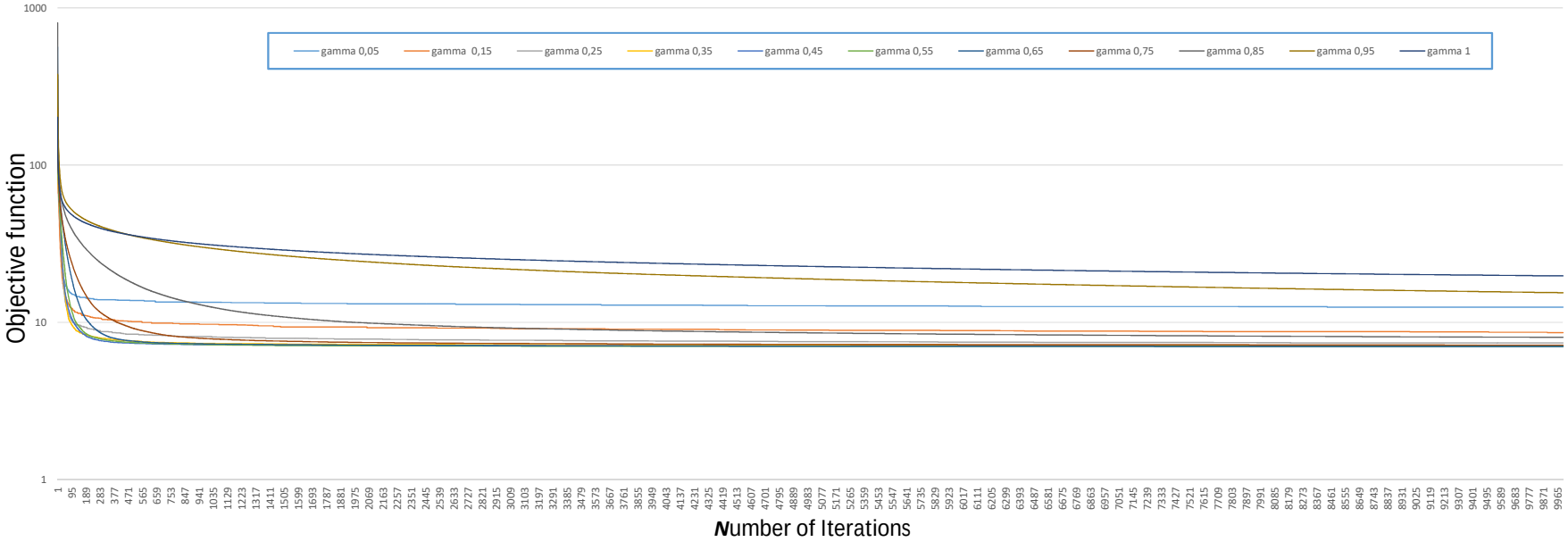

Figure 3. Comparison between α'_k steps with different γ for the minimization of h_{λ_1,λ_2}.

Table 3. Comparison between α'_k steps with different γ for the minimization of h_{λ_1,λ_2}.

γ	0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95	1
The optimal value $h^*_{\lambda_1,\lambda_2}$	12.49	8.61	7.38	7.13	7.04	7.01	**7.01**	7.15	8.02	15.42	19.72

In this regard, Figure 3 and Table 3 demonstrate that the step size $\alpha'_k = \frac{1}{k^{0.65}} \times \frac{1}{\max(1,\|v_{k-1}\|)}$ leads to faster convergence compared to other step sizes, yielding superior results.

3.2.2. Choice of λ_1 and λ_2

After determining the optimal step size (α'_k), we proceed to select the best values for the parameters λ_1 and λ_2. We compare different combinations of these parameters by analyzing the gene set selected by minimizing h_{λ_1,λ_2}, aiming to find values of λ_1 and λ_2 that achieve the highest classification rate on the test set while minimizing the number of genes used in the classification.

The experiments were conducted using the "Prostate_tumor" dataset, and the accuracy rates were determined using a 5-fold cross-validation method. Initially setting $\lambda_2 = 0.2$, and by varying λ_1, the results are shown in Table 4 for different values of λ_1 (with $\lambda_2 = 0.2$).

Table 4. Comparison between the λ_1 for the "Prostate_Tumor" dataset (aggregated values across the 5-fold Cross-validation with $\lambda_2 = 0.2$).

λ_1	Accuracy (Test)	Number of Genes
0.2	96.14	55.6
0.4	96.1	36.8
0.6	96.14	32.6
0.8	96.14	29
1	96.14	24
1.2	97.1	28
1.4	96.14	26.2
1.6	**97.1**	**25.4**
1.8	96.14	25.6
2	96.14	22.6
2.2	96.14	24.6
2.4	96.14	23.8
2.6	95.14	22.2
2.8	96.14	20.4
3	95.14	20.8
3.2	94.19	20.4
3.4	94.19	20.4
3.6	94.19	20.4
3.8	95.14	19.6
4	94.19	19.2

Typically, we can observe that when $\lambda_1 = 1.6$, the results are better. Specifically, this value of λ_1 results in high accuracy on test data (97.1%), while only utilizing 25 genes. However, as we increase λ_1, the number of selected genes decreases, which is a result of the L_1 penalty. After determining that $\lambda_1 = 1.6$ is optimal, we proceed to determine the best value for λ_2. Table 5 shows the results obtained for different values of λ_2, with λ_1 fixed at 1.6. Specifically, it is evident that when $\lambda_2 = 0$, the method employed corresponds to the classic LASSO approach. However, setting $\lambda_2 = 0.1$ yields improved performance, achieving a high accuracy of 97.1% on the test data while utilizing only 22 genes. Consequently, based on these findings, the optimal values of $\lambda_1 = 1.6$ and $\lambda_2 = 0.1$ were selected for the final model.

Table 5. Comparison between the λ_2 for the "Prostate_Tumor" dataset (5 fold Cross-validation with $\lambda_1 = 1.6$).

λ_2	Accuracy (Test)	Number of Genes
0	94.19	21.6
0.025	96.14	20
0.05	96.14	21.2
0.075	96.14	22.4
0.1	**97.1**	**22**
0.125	97.1	23.8
0.15	97.1	24
0.175	96.14	25
0.2	97.1	25.4
0.225	95.14	25.6
0.25	97.1	26.6
0.275	95.14	28.2
0.3	95.19	27.4
0.325	95.14	28.4
0.35	95.14	28.6
0.375	95.19	27.2
0.4	96.14	27.2
0.425	96.14	28.4
0.45	96.14	25.8
0.475	96.14	28.6
0.5	95.19	28.2
0.525	95.1	29.6
0.55	97.1	27
0.575	95.14	30.8
0.6	96.19	29.6
0.625	95.19	28.2
0.65	95.19	32.2
0.675	96.1	31.2
0.7	94.14	26.8
0.725	96.14	29.8
0.75	95.19	29.2

3.3. Results and Comparisons

Firstly, to limit the search space and accelerate the convergence speed of our proposed approach, we selected the initial subset of genes based on the Pearson correlation filter. Subsequently, the GFLASSO-LR algorithm was applied to determine the optimal gene subset. The quality of this subset was assessed based on the accuracy of the test set and the number of genes selected. Due to the non-deterministic nature of our proposed method, each dataset was randomly divided into a training set comprising 70% of the original dataset and a test set comprising 30%. This process was repeated in 50 independent runs for each dataset to ensure reliable results.

Table 6 summarizes the outcomes of our study across four microarray datasets: Colon, DLBCL, Prostate_Tumor, and Prostate. These results stem from experiments employing a range of gene selection techniques to assess their impact on classification accuracy, thereby validating and demonstrating the effectiveness of our proposed method. Specifically, the table compares performance metrics for two classifiers, KNN and SVM, both without selection and with the application of the Pearson filter. Additionally, it includes outcomes using logistic regression enhanced by LASSO and GFLASSO penalties, offering a comprehensive view of the method's efficacy in gene selection for cancer classification.

Our embedded methods (LASSO-LR and GFLASSO-LR) were implemented based on the following parameters:

- $\lambda_1 = 1.6, \lambda_2 = 0$, which corresponds to a LASSO-LR type penalty.
- $\lambda_1 = 1.6, \lambda_2 = 0.1$, which corresponds to a generalized Fused LASSO (GFLASSO-LR) type penalty.

Table 6. The results obtained over 50 executions, without and with selection via the LASSO and GFLASSO penalties.

Dataset	Performance	Without Selection		Pearson Filter		LASSO-LR	GFLASSO-LR
		1NN	SVM	1NN	SVM		
Colon	Accuracy (%)	72.33	80.56	80.66	82.55	83.56	**83.67**
	Number of genes	2000		200		15.4	19.9
DLBCL	Accuracy (%)	81.04	95.65	95.47	**97.91**	90.78	90.87
	Number of genes	5469		200		21.24	20.4
Prostate_Tumor	Accuracy (%)	80	91.27	90.86	**94.6**	93.73	**94.6**
	Number of genes	10,509		200		20.66	21.34
Prostate	Accuracy (%)	80	91.27	91.73	94.26	93.8	**95.27**
	Number of genes	12,600		200		20.64	20.94

Accuracy: The average classification accuracy (70% training and 30% testing). **Genes**: The number of genes used in the classification. **SVM**: The support vector machine classifier using a linear kernel. **LR**: The logistic regression classifier with LASSO and GFLASSO regularization, respectively. **1NN**: the $1-$ nearest neighbor classifier.

The first column of the table displays the datasets used. The second column shows the performance measures, including accuracy (in percent) and the number of selected genes. The third and fourth columns show the results obtained using the 1NN and SVM classifiers, respectively, without gene selection. The fifth and sixth columns present the results obtained using the SVM and 1NN classifiers after having selected 200 genes via the Pearson correlation filter. The seventh and eighth columns present the results of experiments using our LASSO-LR and GFLASSO-LR approaches, respectively.

Figures 4–6 show the average classification accuracy obtained from 50 experiments for different methods. According to these figures, the GFLASSO-LR approach shows good performance on the majority of datasets. More precisely, the highest average accuracy rates are obtained for the Colon, Prostate Tumor, and Prostate datasets.

From Table 6 and Figure 4, it is noteworthy that the GFLASSO-LR method outperforms the traditional LASSO-LR method in terms of accuracy for all datasets. This improvement is particularly significant for the datasets Prostate_Tumor and Prostate, which consist of a small number of genes ranging from 19 to 21. Additionally, both GFLASSO-LR and LASSO-LR methods achieve high classification accuracy (greater than or equal to 83.57%) using a small number of genes compared to the original number.

The effectiveness of our selection methods was compared to classifiers (1NN and SVM) both with and without selection, employing the Pearson filter. Figure 5 and Table 6 demonstrate that our GFLASSO-LR and LASSO-LR approaches outperform the 1NN classifier in both scenarios. Specifically, our methods succeed in enhancing accuracy for the majority of datasets. For instance, in the Colon dataset, an accuracy greater than or equal to 83.56% was achieved using less than 20 genes, indicating a significant improvement in classification accuracy (ranging from 3 to 11% difference) with fewer genes. The most notable improvement was observed for the Prostate dataset, where an accuracy of 94.6% was attained (with a 4–14% difference) using less than 21 genes.

Similarly, Figure 6 and Table 6 demonstrate the effectiveness of our gene selection approaches against the SVM classifier. Indeed, our methods show significant improvements in accuracy for the colon and prostate datasets, while using just fewer than 22 genes.

Afterward, we conducted a Kruskal–Wallis test to analyze the significance of the differences in accuracy obtained by our proposed GFLASSO-LR classifier compared to SVM and 1NN classifiers (without selection). The test revealed a statistically significant difference in performance among the classifiers (H-statistic $=$ 6.55, p-value $=$ 0.038), emphasizing the impact of classifier choice on classification accuracy.

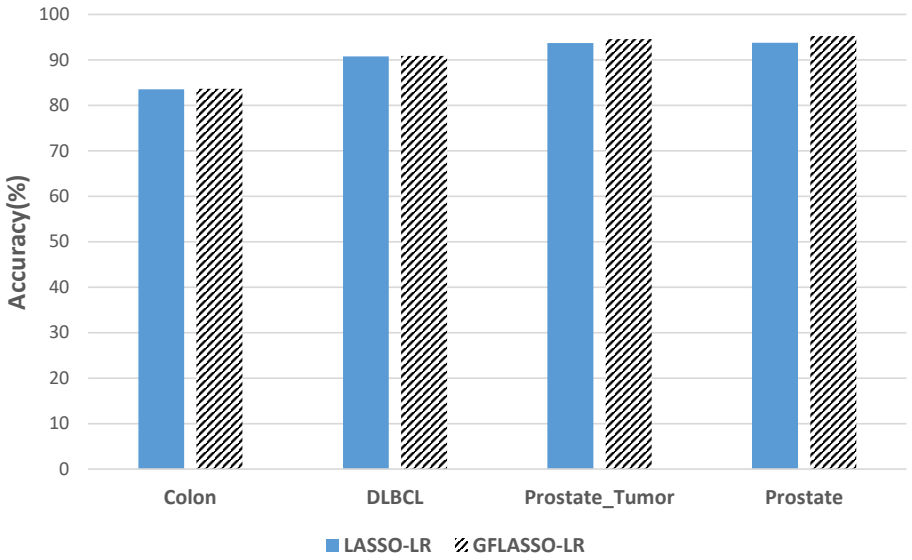

Figure 4. Comparison of classification accuracy (LASSO-LR and GFLASSO-LR).

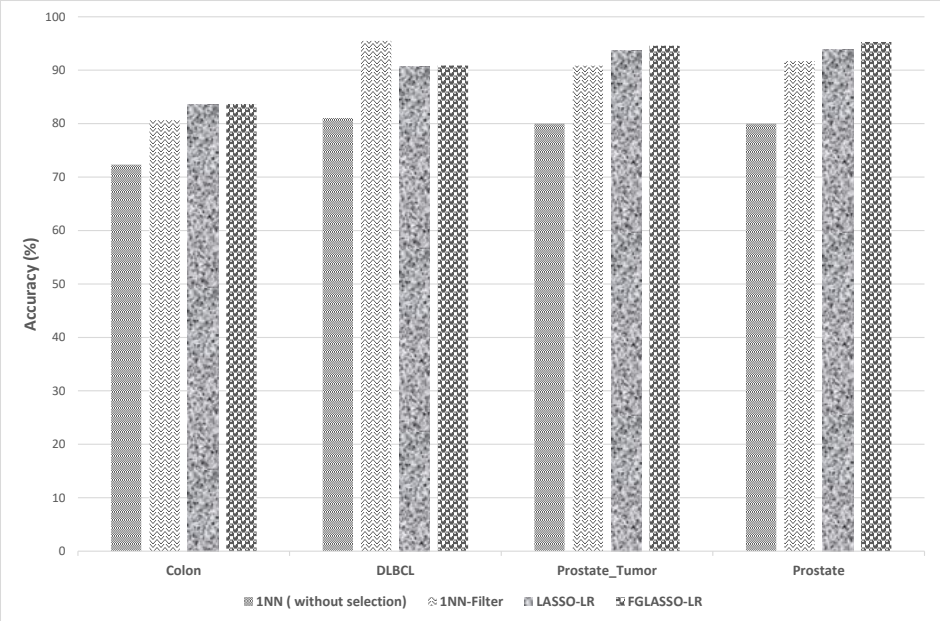

Figure 5. Comparison of classification accuracy between 1NN, LASSO-LR and GFLASSO-LR.

The boxplot (Figure 7) illustrates the performance distribution of GFLASSO-LR compared to the other classifiers across various datasets. GFLASSO-LR consistently achieves higher classification accuracies, particularly noticeable in the Colon and Prostate_Tumor datasets. This highlights the effectiveness of incorporating GFLASSO regularization techniques in logistic regression, enhancing classification performance in specific contexts. These findings emphasize the importance of selecting the appropriate classifier to achieve optimal classification outcomes for individual datasets.

Overall, the results presented in the table show that the GFLASSO-LR algorithm can improve classification accuracy and decrease the number of selected genes. The proposed methods have important applications in the field of genomics and can advance the understanding of gene function and disease diagnosis. Based on the experiments we conducted, we can conclude that our gene selection approaches are well founded. Our methods achieved high classification accuracy in the datasets used in the study.

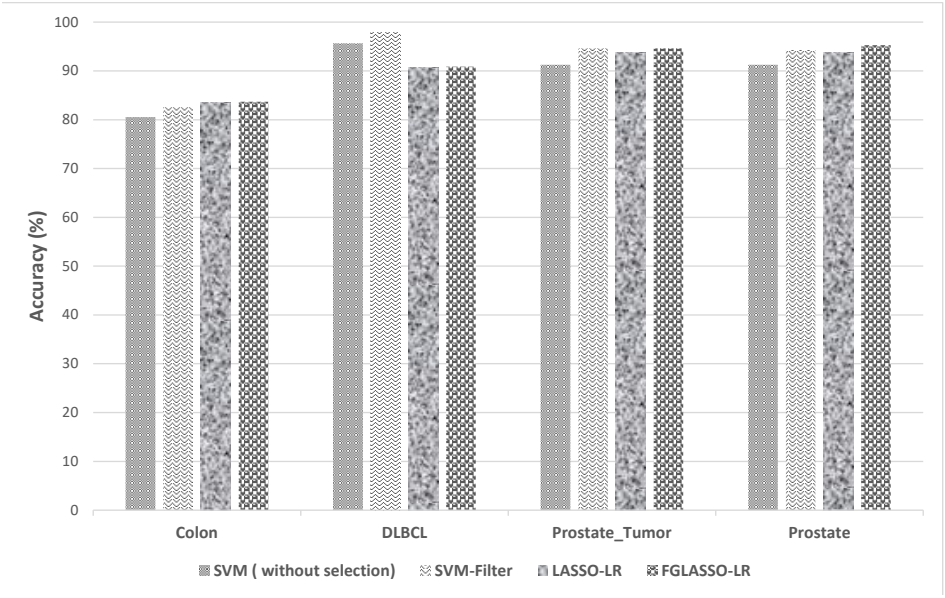

Figure 6. Comparison of classification accuracy between SVM, LASSO-LR, and GFLASSO-LR.

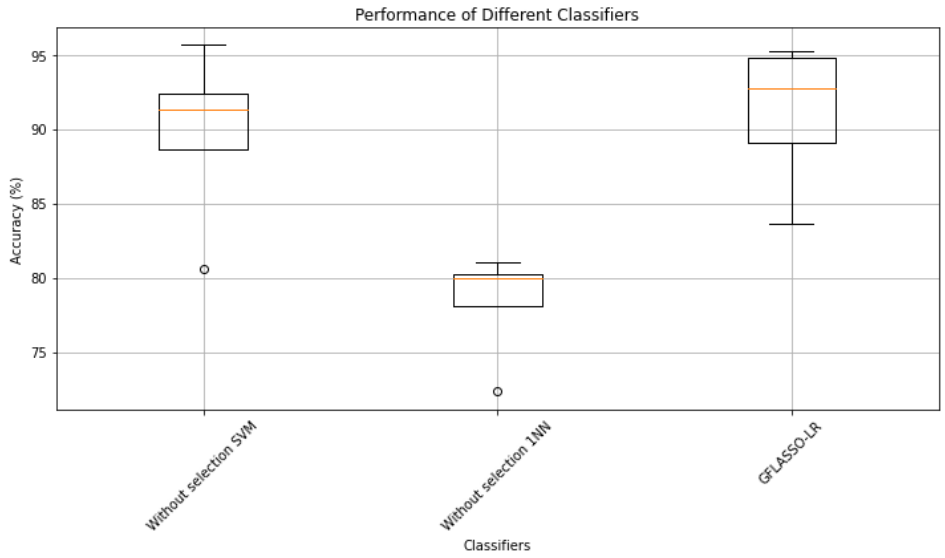

Figure 7. Results of the Kruskal–Wallis Test: GFLASSO-LR vs. SVM and 1NN (Without Selection).

Comparison with Other Approaches

In this subsection, we compare our GFLASSO-LR method with algorithms referenced in the literature: [31,32,40,56,75,76]. All of these papers propose integrated methods based on different types of penalties and apply them to the gene selection problem. To make the comparison meaningful, experiments are performed under the same conditions for each algorithm. Our approach is run 50 times on each dataset, with the dataset randomly divided into two parts (70% for training and 30% for testing) in each run. The average accuracy on the test set and the average number of genes used in classification are chosen. It is noted that the authors of [31] performed 100 runs.

Table 7 summarizes the classification accuracy and the number of selected genes (from the original papers) for each approach. A dash (-) indicates that the result is not reported in the corresponding work. The results obtained by our approach are competitive with previous works.

Table 7. Comparison between our method (GFLASSO-LR) and state-of-the-art methods.

Dataset	Performences	GFLASSO-LR (Our Method)	BLASSO [75]	LASSO [56]	SCAD [76]	ALASSO [40]	Nouvel ALASSO [31]	CBPLR [32]
Colon	**Acccuracy (%)**	83.67 < 2 >	93	79.53	79.51	78.4	82.91	89.5
	Number of genes	19.9	11	14	14	14	12	10
DLBCL	**Acccuracy (%)**	90.87 < 3 >	84	88.33	74.02	84.41	91.32	91.7
	Number of genes	20.4	17	24	24	24	22	17
Prostate	**Acccuracy (%)**	**95.27 < 1 >**	-	91.14	60.13	82.11	93.53	93.6
	Number of genes	20.94	-	29	28	28	24	16

< · >: Indicates the rank of our method among other algorithms in terms of average accuracy for each dataset. **BLASSO** = Bayesian Lasso quantile regression. **LASSO** = The classic LASSO method. **SCAD** = smoothly clipped absolute deviation. **ALASSO** = The classical Adaptive LASSO method. **New ALASSO** = A new Adaptive LASSO method based on the weighted Mahalanobis distance. **CBPLR** = A new method of Adaptive LASSO based on the correlation between genes.

For the Colon dataset, we achieve an accuracy rate of 83.67% using only 19 genes. Our method ranks second, outperforming the other five approaches, with the best performance achieved by the BLASSO approach (93%). For the DLBLC dataset, we obtain the third-best performance (90.87%) after the "New ALASSO" and "CBPLR" approaches, with a difference of less than 1%. For the "Prostate" dataset, we achieved the best performance (accuracy of 95.27%) followed by the "CBPLR" method with an accuracy of 93.6%. The number of genes selected by GFLASSO-LR is relatively similar to that of other methods, except for the Prostate dataset, which uses slightly more genes.

The results of this comparative analysis with previous embedded gene selection methods in cancer classification indicate that our improved generalized fused LASSO penalization approach is effective in the gene selection problem.

3.4. Time and Memory Complexity

3.4.1. Pre-Processing Step (Selecting the Top 200 Highest-Ranked Genes)

- Time Complexity: Computing correlation coefficients for each gene across all samples and selecting the top 200 genes: $O(N \times M)$ (as provided in the information).
- Memory Complexity: Storing microarray data and correlation coefficients: $O(N \times M)$.

3.4.2. Splitting-GFLASSO-LR Algorithm

- Time Complexity: $O(M)$ for randomly splitting the dataset into training and test sets.
- Memory Complexity: $O(M)$ for storing the training and test sets.

3.4.3. Building the Objective Function Based on the Training Data and the 200 Genes

Regarding the time complexity, in the worst case, where each gene is correlated with every other gene, the number of edges in the graph would be $\frac{200\times(200-1)}{2}$. Additionally, computing the absolute correlation values between each pair of genes is estimated using $O\left(\frac{200\times(200-1)}{2} \times M\right)$.

3.4.4. Sub-Gradient Algorithm

Time Complexity: The dominant term in the time complexity is represented as follows.

$$O\left(\frac{200 \times (200-1)}{2} \times M + it_{max} \times \frac{200^3}{2}\right) \quad (14)$$

Typically, the components and computational complexity of an optimization process can be summarized as follows:

- Constructing the graph and calculating weights: negligible compared to the overall complexity.
- Combining the subdifferentials: Once the subdifferentials of each component function are computed, they are combined according to the given expression. This involves a combination of nearly $\frac{200^2}{2}$ subdifferentials, resulting in a time complexity of $O(\frac{200^3}{2})$.
- Iterating over the sub-gradient algorithm: $O(it_{max})$ iterations.
- Each iteration involves updating the coefficients using a sub-gradient: $O(\frac{200^3}{2})$ operations.

Memory Complexity: Storing the graph structure and weights: $O(\frac{200\times(200-1)}{2})$ in the worst case. Storing intermediate results and coefficients: $O(200)$ (since only one set of coefficients needs to be stored).

Overall, the analysis outlines a computationally intensive process with significant emphasis on handling gene interaction data efficiently. While time complexity primarily concerns the combinatorial interactions among genes and the iterative optimization process, memory complexity focuses on efficiently storing gene data, correlations, and algorithmic intermediates. The optimization aims to identify a subset of genes critical for further analysis, with considerations for computational and storage efficiencies.

3.5. Discussion

The discussion on the performance of the GFLASSO-LR approach compared to traditional and recent methods in gene selection for cancer classification underscores its practical utility and effectiveness. Our method's superior or competitive performance across different datasets, particularly in achieving high classification accuracy with a relatively small number of genes, highlights its potential for advancing personalized medicine and genomic research. The ability to identify the most relevant genes from massive datasets with precision is crucial for understanding complex diseases like cancer. Our GFLASSO-LR approach not only simplifies the gene selection process but also ensures that the selected genes are highly indicative of the condition being studied, thus facilitating more accurate diagnoses and the development of targeted therapies.

Moreover, the consistency of our method's performance across various datasets demonstrates its robustness and adaptability to different types of cancer, which is a significant advantage in the rapidly evolving field of genomics. The use of logistic regression with the GFLASSO penalty enhances the model's ability to deal with the high dimensionality and multicollinearity inherent in microarray data, thereby overcoming some of the common challenges faced in genomic data analysis. However, it is worth noting that while our approach shows promise, further research is necessary to explore its full potential and limitations. Future studies could focus on refining the algorithm to improve efficiency and accuracy, as well as testing it on a wider range of datasets. Additionally, integrating our method with other data types, such as clinical and phenotypic information, could provide a more comprehensive understanding of the gene–disease relationships and further enhance its application in personalized medicine.

4. Conclusions

High-dimensional classification problems in microarray dataset analysis are a crucial area of research in cancer classification. In this paper, we propose and apply an improved method, GFLASSO-LR, for simultaneous gene coefficient estimation and selection to improve classification performance. We also demonstrate the convergence of the sub-gradient algorithm to solve the associated non-differential convex optimization problem.

The proposed method was evaluated based on the number of selected genes and classification accuracy on four sets of high-dimensional cancer classification data. The results consistently showed that GFLASSO-LR can significantly reduce the number of relevant genes and has superior accuracy compared to the classical LASSO method. Overall, the results demonstrate that GFLASSO is a promising method for accurately analyzing

high-dimensional microarray data in cancer classification. The method can be applied to other types of high-dimensional classification data related to the medical domain. Future research could extend the present work to cover high-dimensional multiclass cancer data and focus on very high-dimensional microarray data for cancer classification.

Author Contributions: Conceptualization, A.B.-J.; Methodology, A.B.-J.; Software, A.B.-J.; Validation, A.B.-J. and S.M.D.; Formal analysis, S.E.B., A.M., Y.H., W.M. and H.A.-A.; Investigation, S.M.D., S.E.B., A.M., Y.H. and S.A.; Resources, S.M.D. and Y.H.; Writing—original draft, A.B.-J.; Writing—review & editing, S.M.D., S.E.B., A.M., Y.H., S.A., W.M. and H.A.-A.; Visualization, S.E.B., A.M. and S.A.; Supervision, S.M.D., S.E.B., A.M. and S.A.; Project administration, Y.H., W.M. and H.A.-A.; Funding acquisition, Y.H., W.M. and H.A.-A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Higher Education, Scientific Research and Innovation, the Digital Development Agency (DDA), and the National Center for Scientific and Technical Research (CNRST) under Project ALKHAWARIZMI/2020/19.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author. The data are not publicly available due to data sensitivity and privacy concerns.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, M.; Ke, L.; Wang, L.; Deng, S.; Yu, X. A novel hybrid gene selection for tumor identification by combining multifilter integration and a recursive flower pollination search algorithm. *Knowl.-Based Syst.* **2023**, *262*, 110250. [CrossRef]
2. Feng, S.; Jiang, X.; Huang, Z.; Li, F.; Wang, R.; Yuan, X.; Sun, Z.; Tan, H.; Zhong, L.; Li, S.; et al. DNA methylation remodeled amino acids biosynthesis regulates flower senescence in carnation (*Dianthus caryophyllus*). *New Phytol.* **2024**, *241*, 1605–1620. [CrossRef] [PubMed]
3. Mehrabi, N.; Haeri Boroujeni, S.P.; Pashaei, E. An efficient high-dimensional gene selection approach based on the Binary Horse Herd Optimization Algorithm for biologicaldata classification. *Iran J. Comput. Sci.* **2024**, 1–31. [CrossRef]
4. Syu, G.D.; Dunn, J.; Zhu, H. Developments and applications of functional protein microarrays. *Mol. Cell. Proteom.* **2020**, *19*, 916–927. [CrossRef] [PubMed]
5. Caraffi, S.G.; van der Laan, L.; Rooney, K.; Trajkova, S.; Zuntini, R.; Relator, R.; Haghshenas, S.; Levy, M.A.; Baldo, C.; Mandrile, G.; et al. Identification of the DNA methylation signature of Mowat-Wilson syndrome. *Eur. J. Hum. Genet.* **2024**, 1–11. [CrossRef] [PubMed]
6. Srivastava, S.; Jayaswal, N.; Kumar, S.; Sharma, P.K.; Behl, T.; Khalid, A.; Mohan, S.; Najmi, A.; Zoghebi, K.; Alhazmi, H.A. Unveiling the potential of proteomic and genetic signatures for precision therapeutics in lung cancer management. *Cell. Signal.* **2024**, *113*, 110932. [CrossRef]
7. Ghavidel, A.; Pazos, P. Machine learning (ML) techniques to predict breast cancer in imbalanced datasets: A systematic review. *J. Cancer Surviv.* **2023**, 1–25. [CrossRef]
8. Bir-Jmel, A.; Douiri, S.M.; Elbernoussi, S. Gene selection via a new hybrid ant colony optimization algorithm for cancer classification in high-dimensional data. *Comput. Math. Methods Med.* **2019**, *2019*, 7828590. [CrossRef] [PubMed]
9. Bir-Jmel, A.; Douiri, S.M.; Elbernoussi, S. Gene selection via BPSO and Backward generation for cancer classification. *RAIRO-Oper. Res.* **2019**, *53*, 269–288. [CrossRef]
10. Sethi, B.K.; Singh, D.; Rout, S.K.; Panda, S.K. Long Short-Term Memory-Deep Belief Network based Gene Expression Data Analysis for Prostate Cancer Detection and Classification. *IEEE Access* **2023**, *12*, 1508–1524. [CrossRef]
11. Maafiri, A.; Bir-Jmel, A.; Elharrouss, O.; Khelifi, F.; Chougdali, K. LWKPCA: A New Robust Method for Face Recognition Under Adverse Conditions. *IEEE Access* **2022**, *10*, 64819–64831. [CrossRef]
12. Bir-Jmel, A.; Douiri, S.M.; Elbernoussi, S. Minimum redundancy maximum relevance and VNS based gene selection for cancer classification in high-dimensional data. *Int. J. Comput. Sci. Eng.* **2024**, *27*, 78–89.
13. Maafiri, A.; Chougdali, K. Robust face recognition based on a new Kernel-PCA using RRQR factorization. *Intell. Data Anal.* **2021**, *25*, 1233–1245. [CrossRef]
14. Amaldi, E.; Kann, V. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theor. Comput. Sci.* **1998**, *209*, 237–260. [CrossRef]
15. Blum, A.L.; Rivest, R.L. Training a 3-node neural network is NP-complete. *Neural Netw.* **1992**, *5*, 117–127. [CrossRef]
16. Yaqoob, A.; Verma, N.K.; Aziz, R.M. Optimizing gene selection and cancer classification with hybrid sine cosine and cuckoo search algorithm. *J. Med. Syst.* **2024**, *48*, 10. [CrossRef] [PubMed]
17. Bechar, A.; Elmir, Y.; Medjoudj, R.; Himeur, Y.; Amira, A. Harnessing transformers: A leap forward in lung cancer image detection. In Proceedings of the 2023 6th International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, 8–9 November 2023; pp. 218–223.

18. Hamza, A.; Lekouaghet, B.; Himeur, Y. Hybrid whale-mud-ring optimization for precise color skin cancer image segmentation. In Proceedings of the 2023 6th International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, 8–9 November 2023; pp. 87–92.
19. Habchi, Y.; Himeur, Y.; Kheddar, H.; Boukabou, A.; Atalla, S.; Chouchane, A.; Ouamane, A.; Mansoor, W. Ai in thyroid cancer diagnosis: Techniques, trends, and future directions. *Systems* **2023**, *11*, 519. [CrossRef]
20. Kohavi, R.; John, G.H. Wrappers for feature subset selection. *Artif. Intell.* **1997**, *97*, 273–324. [CrossRef]
21. Gu, Q.; Li, Z.; Han, J. Generalized fisher score for feature selection. *arXiv* **2012**, arXiv:1202.3725.
22. Jafari, P.; Azuaje, F. An assessment of recently published gene expression data analyses: Reporting experimental design and statistical factors. *BMC Med. Inform. Decis. Mak.* **2006**, *6*, 27. [CrossRef]
23. Mishra, D.; Sahu, B. Feature selection for cancer classification: A signal-to-noise ratio approach. *Int. J. Sci. Eng. Res.* **2011**, *2*, 1–7.
24. Wang, Z. *Neuro-Fuzzy Modeling for Microarray Cancer Gene Expression Data*; First year transfer report; University of Oxford: Oxford, UK, 2005.
25. Kononenko, I. Estimating attributes: Analysis and extensions of RELIEF. In *European Conference on Machine Learning*; Springer: Berlin/Heidelberg, Germany, 1994; pp. 171–182.
26. Kishore, A.; Venkataramana, L.; Prasad, D.V.V.; Mohan, A.; Jha, B. Enhancing the prediction of IDC breast cancer staging from gene expression profiles using hybrid feature selection methods and deep learning architecture. *Med. Biol. Eng. Comput.* **2023**, *61*, 2895–2919. [CrossRef] [PubMed]
27. Du, J.; Zhang, Z.; Sun, Z. Variable selection for partially linear varying coefficient quantile regression model. *Int. J. Biomath.* **2013**, *6*, 1350015. [CrossRef]
28. Li, C.J.; Zhao, H.M.; Dong, X.G. Bayesian empirical likelihood and variable selection for censored linear model with applications to acute myelogenous leukemia data. *Int. J. Biomath.* **2019**, *12*, 1950050. [CrossRef]
29. Li, L.; Liu, Z.P. Biomarker discovery from high-throughput data by connected network-constrained support vector machine. *Expert Syst. Appl.* **2023**, *226*, 120179. [CrossRef]
30. Alharthi, A.M.; Lee, M.H.; Algamal, Z.Y. Gene selection and classification of microarray gene expression data based on a new adaptive L1-norm elastic net penalty. *Inform. Med. Unlocked* **2021**, *24*, 100622. [CrossRef]
31. Alharthi, A.M.; Lee, M.H.; Algamal, Z.Y. Weighted L1-norm logistic regression for gene selection of microarray gene expression classification. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2020**, *4*, 2088–5334. [CrossRef]
32. Algamal, Z.Y.; Lee, M.H. Penalized logistic regression with the adaptive LASSO for gene selection in high-dimensional cancer classification. *Expert Syst. Appl.* **2015**, *42*, 9326–9332. [CrossRef]
33. Algamal, Z.Y.; Lee, M.H. A two-stage sparse logistic regression for optimal gene selection in high-dimensional microarray data classification. *Adv. Data Anal. Classif.* **2019**, *13*, 753–771. [CrossRef]
34. Li, L.; Liu, Z.P. A connected network-regularized logistic regression model for feature selection. *Appl. Intell.* **2022**, *52*, 11672–11702. [CrossRef]
35. Yang, Z.Y.; Liang, Y.; Zhang, H.; Chai, H.; Zhang, B.; Peng, C. Robust Sparse Logistic Regression with the $L_q(0 < q < 1)$ Regularization for Feature Selection Using Gene Expression Data. *IEEE Access* **2018**, *6*, 68586–68595.
36. Ijaz, M.; Asghar, Z.; Gul, A. Ensemble of penalized logistic models for classification of high-dimensional data. *Commun.-Stat.-Simul. Comput.* **2019**, *50*, 2072–2088. [CrossRef]
37. Kastrin, A.; Peterlin, B. Rasch-based high-dimensionality data reduction and class prediction with applications to microarray gene expression data. *Expert Syst. Appl.* **2010**, *37*, 5178–5185. [CrossRef]
38. Wang, Y.; Zhang, W.; Fan, M.; Ge, Q.; Qiao, B.; Zuo, X.; Jiang, B. Regression with Adaptive Lasso and Correlation based Penalty. *Appl. Math. Model.* **2021**, *105*, 179–196. [CrossRef]
39. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2005**, *67*, 301–320. [CrossRef]
40. Zou, H. The adaptive lasso and its oracle properties. *J. Am. Stat. Assoc.* **2006**, *101*, 1418–1429. [CrossRef]
41. Bach, F.; Jenatton, R.; Mairal, J.; Obozinski, G. Structured sparsity through convex optimization. *Stat. Sci.* **2012**, *27*, 450–468. [CrossRef]
42. Tibshirani, R.; Saunders, M.; Rosset, S.; Zhu, J.; Knight, K. Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2005**, *67*, 91–108. [CrossRef]
43. Jang, W.; Lim, J.; Lazar, N.A.; Loh, J.M.; Yu, D. Some properties of generalized fused lasso and its applications to high dimensional data. *J. Korean Stat. Soc.* **2015**, *44*, 352–365. [CrossRef]
44. Rinaldo, A. Properties and refinements of the fused lasso. *Ann. Stat.* **2009**, *37*, 2922–2952. [CrossRef]
45. Qian, J.; Jia, J. On stepwise pattern recovery of the fused lasso. *Comput. Stat. Data Anal.* **2016**, *94*, 221–237. [CrossRef]
46. Höfling, H.; Binder, H.; Schumacher, M. A coordinate-wise optimization algorithm for the Fused Lasso. *arXiv* **2010**, arXiv:1011.6409.
47. Viallon, V.; Lambert-Lacroix, S.; Hoefling, H.; Picard, F. On the robustness of the generalized fused lasso to prior specifications. *Stat. Comput.* **2016**, *26*, 285–301. [CrossRef]
48. Hoefling, H. A path algorithm for the fused lasso signal approximator. *J. Comput. Graph. Stat.* **2010**, *19*, 984–1006. [CrossRef]
49. Liu, J.; Yuan, L.; Ye, J. An efficient algorithm for a class of fused lasso problems. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; pp. 323–332.

50. Tibshirani, R.J.; Taylor, J. The solution path of the generalized lasso. *Ann. Stat.* **2011**, *39*, 1335–1371. [CrossRef]
51. Johnson, N.A. A dynamic programming algorithm for the fused lasso and l 0-segmentation. *J. Comput. Graph. Stat.* **2013**, *22*, 246–260. [CrossRef]
52. Fisher, R.A.; Yates, F. *Statistical Tables for Biological, Agricultural and Medical Research*; Oliver and Boyd: Edinburgh, UK, 1938.
53. Albert, A.; Anderson, J.A. On the existence of maximum likelihood estimates in logistic regression models. *Biometrika* **1984**, *71*, 1–10. [CrossRef]
54. Liang, Y.; Liu, C.; Luan, X.Z.; Leung, K.S.; Chan, T.M.; Xu, Z.B.; Zhang, H. Sparse logistic regression with a $L_{1/2}$ penalty for gene selection in cancer classification. *BMC Bioinform.* **2013**, *14*, 198. [CrossRef] [PubMed]
55. Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, *12*, 55–67. [CrossRef]
56. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Methodol.* **1996**, *58*, 267–288. [CrossRef]
57. Meinshausen, N. Relaxed lasso. *Comput. Stat. Data Anal.* **2007**, *52*, 374–393. [CrossRef]
58. Meier, L.; Van De Geer, S.; Bühlmann, P. The group lasso for logistic regression. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2008**, *70*, 53–71. [CrossRef]
59. Wang, S.; Nan, B.; Rosset, S.; Zhu, J. Random lasso. *Ann. Appl. Stat.* **2011**, *5*, 468. [CrossRef]
60. Shor, N.Z. Application of the gradient-descent method to solution of the network transport problem. *Cybern. Syst. Anal.* **1967**, *3*, 43–45. [CrossRef]
61. Polyak, B.T. Minimization of unsmooth functionals. *USSR Comput. Math. Math. Phys.* **1969**, *9*, 14–29. [CrossRef]
62. Nemirovski, A.S.; Yudin, D.B. Cesari convergence of the gradient method of approximating saddle points of convex-concave functions. *Dokl. Akad. Nauk. SSSR* **1978**, *239*, 1056–1059.
63. Rockafellar, R.T. *Convex Analysis*; Princeton University Press: Princeton, NJ, USA, 1970.
64. Anstreicher, K.M.; Wolsey, L.A. Two "well-known" properties of subgradient optimization. *Math. Program.* **2009**, *120*, 213–220. [CrossRef]
65. Polyak, B.T. A general method for solving extremal problems. *Dokl. Akad. Nauk. SSSR* **1967**, . *174*, 33–36.
66. Shor, N.Z. *Minimization Methods for Non-Differentiable Functions*; Springer Science & Business Media: Berlin, Germany, 2012; Volume 3.
67. Mordukhovich, B.S.; Nam, N.M. *An Easy Path to Convex Analysis and Applications*; Synthesis Lectures on Mathematics and Statistics; Springer: Cham, Switzerland, 2013; Volume 6.
68. Huang, J.; Ma, S.; Zhang, C.H. *The Iterated Lasso for High-Dimensional Logistic Regression*; Technical report; The University of Iowa, Department of Statistics and Actuarial Sciences: Iowa City, IA, USA, 2008.
69. Cui, L.; Bai, L.; Wang, Y.; Philip, S.Y.; Hancock, E.R. Fused lasso for feature selection using structural information. *Pattern Recognit.* **2021**, *119*, 108058. [CrossRef]
70. Bertsekas, D.P. Nonlinear programming. *J. Oper. Res. Soc.* **1997**, *48*, 334. [CrossRef]
71. Alon, U.; Barkai, N.; Notterman, D.A.; Gish, K.; Ybarra, S.; Mack, D.; Levine, A.J. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA* **1999**, *96*, 6745–6750. [CrossRef] [PubMed]
72. Statnikov, A.; Aliferis, C.F.; Tsamardinos, I.; Hardin, D.; Levy, S. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics* **2005**, *21*, 631–643. [CrossRef] [PubMed]
73. Singh, D.; Febbo, P.G.; Ross, K.; Jackson, D.G.; Manola, J.; Ladd, C.; Tamayo, P.; Renshaw, A.A.; D'Amico, A.V.; Richie, J.P.; et al. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell* **2002**, *1*, 203–209. [CrossRef] [PubMed]
74. Alber, Y.I.; Iusem, A.N.; Solodov, M.V. On the projected subgradient method for nonsmooth convex optimization in a Hilbert space. *Math. Program.* **1998**, *81*, 23–35. [CrossRef]
75. Algamal, Z.Y.; Alhamzawi, R.; Ali, H.T.M. Gene selection for microarray gene expression classification using Bayesian Lasso quantile regression. *Comput. Biol. Med.* **2018**, *97*, 145–152. [CrossRef]
76. Fan, J.; Li, R. Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Stat. Assoc.* **2001**, *96*, 1348–1360. [CrossRef]