Statistics Innovation

ARTICLE

Open Access

https://doi.org/10.48130/stati-0025-0001

Statistics Innovation **2025**, 2: e001

SRD: Sparse ramp discrimination for classification and variable selection on high-dimensional biological data

Xin Zhou[1*] and Zuoheng Wang[1,2]

[1] *Department of Biostatistics, Yale University School of Public Health, New Haven, CT 06510, USA*
[2] *Department of Biomedical Informatics & Data Science, Yale University School of Medicine, New Haven, CT 06510, USA*
* Corresponding author, E-mail: xin.zhou@yale.edu

Abstract

A massive amount of biological data generates a high volume of information. However, high-dimensional and noisy data also present significant challenges for data analysis, particularly in pattern classification. Many large-margin classification methods follow a regularization framework with the 'loss + penalty' structure. By incorporating LASSO or elastic-net penalties, we may perform classification and variable selection simultaneously. Most large margin methods rely on convex loss functions, which are computationally advantageous but provide poor approximations of the 0-1 loss due to their unbounded nature. In contrast, non-convex loss functions offer better approximations and greater robustness to outliers. We propose the sparse ramp discrimination (SRD) method, which integrates the non-convex smoothed ramp loss function with the elastic-net penalty. We apply the difference of the convex (d.c.) algorithm to efficiently solve the non-convex optimization problem through a sequence of convex subproblems. The robustness of SRD makes it well-suited for noisy, high-dimensional biological data. The performance of the proposed SRD is illustrated through simulation studies and on a colon cancer involving high throughput biological datasets.

Citation: Zhou X, Wang Z. 2025. SRD: Sparse ramp discrimination for classification and variable selection on high-dimensional biological data. *Statistics Innovation* 2: e001 https://doi.org/10.48130/stati-0025-0001

Introduction

The rapid advancement of high throughput technologies has enabled the collection of diverse omics data generated through whole genome tiling arrays, mass spectrometry, and more recently next-generation sequencing platforms. While these massive data provide us with vast information, they also pose significant challenges for data analysis. One major issue is the high dimensionality of the data—for instance, microarray gene expression data can measure tens of thousands of genes simultaneously—coupled with the typically small sample sizes. This 'curse of dimensionality' makes it difficult to accurately estimate model parameters. Additionally, the inherent noise in biological data and the uncertainty in the target response often lead to overfitting, further complicating analysis.

Classification plays a pivotal role in biomedical and bioinformatics research[1,2]. It involves learning a function from training data to predict the class label of any valid input covariates. The support vector machine (SVM) is one of the commonly used classification methods[3–5]. However, SVMs cannot perform variable selection simultaneously, which is essential for analyzing high-dimensional data. Zhu et al.[6] proposed the ℓ_1-norm SVM, which incorporates the LASSO penalty[7] for automatic variable selection. Wang et al.[8] applied the elastic-net penalty[9] to SVMs for variable selection on high dimensional data. The elastic-net penalty exhibits a 'grouping effect', which tends to select or remove highly correlated covariates together.

Many large margin classification algorithms, including SVMs, are in the regularization framework, typically following the 'loss + penalty' format. In addition to the LASSO and elastic-net penalties, other penalties, such as $\ell_{1/2}$[10] and SCAD[11], have also been applied to pattern classification in high-throughput biological data analysis.

In this work, we focus on the loss function. For binary classification problems, the most straightforward loss function is the 0-1 loss. However, it is well known that minimizing the 0-1 loss is NP-hard. As a result, surrogate loss functions are commonly used instead. Generally, loss functions can be categorized into two types: convex and non-convex.

Convex loss functions, such as the hinge loss used in SVMs, are computationally advantageous due to the convexity. However, the unbounded property of the convex function often leads to poor approximations to the 0-1 loss function. To address this limitation, various non-convex loss functions have been proposed, including the Ψ loss[12], the ramp loss[13,14], and the sigmoid loss[15,16]. Non-convex loss functions, such as the ramp loss, are particularly robust to outliers[14]. In contrast, convex loss functions can be sensitive to outliers, making the corresponding classifiers prone to being dominated by outliers.

The robustness of non-convex loss functions makes them well-suited for handling noisy, high dimensional biological data. However, non-convex optimization poses significant challenges. In this work, we use a smoothed version of the ramp loss function, and propose applying the difference of convex (d.c.) algorithm to efficiently solve the non-convex optimization problem through a sequence of convex subproblems. We refer to the proposed method as Sparse Ramp Discrimination (SRD).

Methods

Large margin classification

Consider n training data pairs: $\{(\mathbf{x}_i, y_i), i = 1, \ldots, n\}$, where x_i is a d-dimensional covariate vector representing the i' th sample, and $y_i \in \{1, -1\}$ is the corresponding class label. The linear decision function is described as:

$$f(\mathbf{x}) = \mathbf{x}^T\boldsymbol{\beta} + \beta_0$$

where, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_d)_T$, and β_0 is a scalar. The decision function divides the covariate space into two regions based on the sign of $f(\mathbf{x})$.

© The Author(s)

Many large margin classification methods, such as SVMs, fall in the regularization framework, where the objective function is constructed in the form of 'loss + penalty' as follows:

$$\frac{1}{n}\sum_{i=1}^{n} L(y_i f(\mathbf{x}_i)) + \sum_{j=1}^{d} p_\lambda(\beta_j) \tag{1}$$

where, $L(\cdot)$ is a loss function and $p_\lambda(\cdot)$ is a penalty function with regularization parameters λ. For binary classification problems, the most straightforward loss function is the 0-1 loss $L(r) = \mathbb{I}(r < 0)$, where $\mathbb{I}(\cdot)$ is the indicator function. The well-known SVMs use the hinge loss, $L(r) = \max(1-r, 0)$, and the ridge penalty, $p_\lambda(\beta) = \frac{\lambda}{2}\beta^2$. Although SVMs are widely applied in bioinformatics for high-dimensional data analysis, they do not support automatic variable selection. Traditional variable selection methods, such as recursive feature elimination (SVM-RFE)[5,17], are computationally intensive and often lack stability[18].

Zhu et al.[6] proposed the ℓ_1-norm SVM, which uses the LASSO penalty, $p_\lambda(\beta) = \lambda|\beta|$[7] for automatic variable selection. However, the LASSO penalty has two notable limitations: (1) the number of selected variables is constrained by the sample size; (2) the LASSO penalty tends to select only one or a few highly correlated variables. To address these issues, Wang et al.[8] applied the elastic-net penalty[9], defined as:

$$p_{\lambda_1,\lambda_2}(\beta) = \lambda_1|\beta| + \frac{\lambda_2}{2}\beta^2 \tag{2}$$

where, λ_1, $\lambda_2 \geq 0$ are regularization parameters. The elastic-net penalty represents an important generalization of the LASSO penalty. As a hybrid of the LASSO and ridge penalties, the elastic-net penalty retains the variable selection feature of LASSO while introducing a grouping effect, a characteristic of the ridge penalty. This grouping effect ensures that highly correlated variables are likely to be selected or removed together, resulting in similar estimated coefficients for such variables.

Sparse ramp discrimination

The loss functions in Eqn (1) can be categorized into two types: convex and non-convex. An example of a convex loss function is the hinge loss used in SVMs. It serves as a convex upper bound of the 0-1 loss function. The convexity makes the optimization algorithm computationally efficient. However, the unbounded nature of convex loss functions often results in a poor approximation of the 0-1 loss.

On the other side, non-convex loss functions, such as the ramp loss[14], provide better approximations to the 0-1 loss. It is well known that by truncating the unbounded hinge loss, ramp loss-related methods have been shown to be robust to outliers in training data for classification problems[14]. The robustness makes non-convex loss functions suitable for handling noisy high dimensional biological data. However, the non-convex optimization is challenging, which often limits their practice application. Here we consider a novel non-convex loss function:

$$T(r) = \begin{cases} 0 & \text{if } r \geq 1, \\ (1-r)^2 & \text{if } 0 \leq r < 1, \\ 2-(1+r)^2 & \text{if } -1 \leq r < 0, \\ 2 & \text{if } r < -1. \end{cases}$$

We refer to this as the smoothed ramp loss in this article. Figure 1 illustrates the hinge loss, ramp loss, and smoothed ramp loss functions. Unlike the ramp loss, the smoothed ramp loss is differentiable everywhere, providing computational advantages for optimization. Moreover, similar to the ramp loss, the smoothed ramp loss is also robust to outliers.

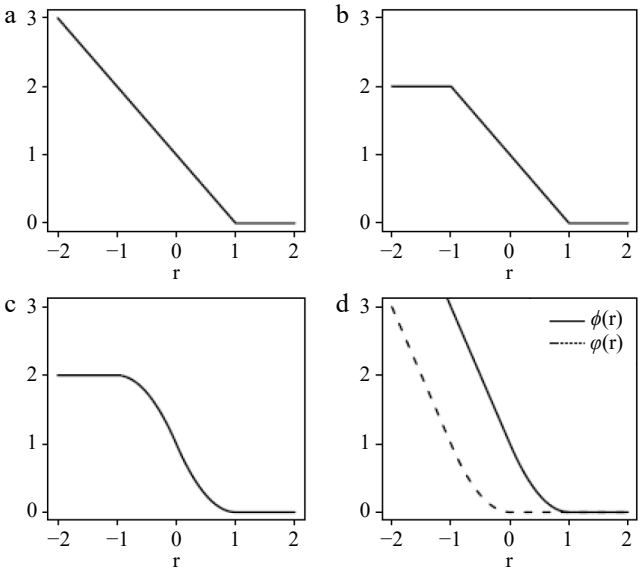

Fig. 1 (a) Hinge loss, (b) ramp loss, and (c) smoothed ramp loss functions. (d) Shows the difference of convex decomposition of the smoothed ramp loss, $T(r) = \phi(r) - \varphi(r)$.

In this work, we integrate the smoothed ramp loss with the elastic-net penalty into the regularization framework in Eqn (1), and propose a novel method called sparse ramp discrimination (SRD):

$$\min_{\beta,\beta_0} \frac{1}{n}\sum_{i=1}^{n} T\left(y_i\left(\mathbf{x}_i^T\beta + \beta_0\right)\right) + \sum_{j=1}^{d}\left(\lambda_1|\beta_j| + \frac{\lambda_2}{2}\beta_j^2\right) \tag{3}$$

Here we only present a case where all covariates are penalized using the same parameters λ_1 and λ_2. In practice, different values of λ_1's or λ_2's can be assigned to individual covariates.

Note that $T(\cdot)$ is Lipschitz continuous with a Lipschitz constant of 2. According to **Theorem 1** in Wang et al.[8], SRD has the grouping effect. For the sake of completeness, we restate the theorem for SRD in this article.

Theorem 1. *Let $\hat{\beta}$ and $\hat{\beta}_0$ denote the solution for (3). For any pair (j, j'), we have*

$$|\hat{\beta}_j - \hat{\beta}_{j'}| \leq \frac{2}{\lambda_2}\sum_{i=1}^{n}|x_{ij} - x_{ij'}|$$

Furthermore, if the input covariates $\mathbf{x}_{\cdot j} = (x_{1j}, ..., x_{nj})^T$ and $\mathbf{x}_{\cdot j'} = (x_{1j'}, ..., x_{nj'})^T$ are centered and normalized, then

$$|\hat{\beta}_j - \hat{\beta}_{j'}| \leq \frac{2\sqrt{n}}{\lambda_2}\sqrt{2(1-\rho)}$$

Theorem 1 demonstrates that highly correlated covariates tend to have similar estimated coefficients. As a result, they are more likely to be selected or removed together, particularly when λ_2 is large.

Algorithm

The smoothed ramp loss is a non-convex loss function, which makes the optimization problem in Eqn (3) a non-convex minimization task. Similar to the robust truncated hinge loss SVM[14], we solve the non-convex minimization problem using the d.c. algorithm[19], also known as the Concave-Convex Procedure (CCCP) in the machine learning community[20]. This approach assumes that the objective function can be expressed as the sum of a convex component, $Q_{vex}(\Theta)$, and a concave component, $Q_{cav}(\Theta)$. As outlined in Algorithm 1, the d.c. algorithm solves the non-convex optimization problem by minimizing a sequence of convex subproblems.

Algorithm 1. The d.c. algorithm for minimizing $Q(\Theta) = Q_{vex}(\Theta) + Q_{cav}(\Theta)$.

Initialize $\Theta^{(0)}$
repeat

$$\Theta^{(t+1)} = argmin_{\Theta} Q_{vex}(\Theta) + \left\langle Q_{cav}^{'}\left(\Theta^{(t)}\right), \Theta \right\rangle$$

until *convergence of* $\Theta^{(t)}$

Let

$$\phi(r) = \begin{cases} 0 & \text{if } r \geq 1, \\ (1-r)^2 & \text{if } 0 \leq r < 1, \\ 1-2r & \text{if } r < 0; \end{cases} \quad \text{and} \quad \varphi(r) = \begin{cases} 0 & \text{if } r \geq 0, \\ r^2 & \text{if } -1 \leq r < 0, \\ -1-2r & \text{if } r < -1. \end{cases}$$

Note that both $\varphi(\cdot)$ and $\phi(\cdot)$ are smooth and convex. We have a difference-of-convex decomposition of the smoothed ramp loss,

$$T(r) = \phi(r) - \varphi(r) \tag{4}$$

as illustrated in Fig. 1d. Denote Θ as (β, β_0). Applying Eqn (4), the objective function in Eqn (3) can be decomposed as:

$$Q^s(\Theta) = \underbrace{\frac{1}{n}\sum_{i=1}^{n}\phi(r_i) + \sum_{j=1}^{d}\left(\lambda_1|\beta_j| + \frac{\lambda_2}{2}\beta_j^2\right)}_{Q_{\text{vex}}(\Theta)} \underbrace{- \frac{1}{n}\sum_{i=1}^{n}\varphi(r_i)}_{Q_{\text{cav}}(\Theta)}$$

where, the margin $r_i = y_i\left(\mathbf{x}_i^T\beta + \beta_0\right)$. To simplify, we introduce the notation,

$$\kappa_i = \frac{\partial Q_{cav}}{\partial r_i} = -\frac{1}{n}\frac{d\varphi(r_i)}{dr_i} \tag{5}$$

for $i = 1, ..., n$. Thus, the convex subproblem at the $(t + 1)$'th iteration of the d.c. algorithm is:

$$\min_{\beta,\beta_0} \frac{1}{n}\sum_{i=1}^{n}\phi(r_i) + \sum_{i=1}^{n}\kappa_i^{(t)} r_i + \sum_{j=1}^{d}\left(\lambda_1|\beta_j| + \frac{\lambda_2}{2}\beta_j^2\right) \tag{6}$$

It is popular to use coordinate descent methods to solve such optimization problems[21,22]. However, under some circumstances, coordinate descent methods may suffer from slow convergence. In this work, we apply the General Iterative Shrinkage and Thresholding (GIST) algorithm[23]. GIST updates all coordinates simultaneously through gradient descent and a thresholding rule, and it is easy to implement. We do not delve into the details of GIST in this article. Interested readers may refer to Gong et al.[23]. The SRD procedure is summarized in the following algorithm:

The algorithm is not limited to the smoothed ramp loss. It is applicable to a broader class of smooth non-convex loss functions that can be similarly expressed using a difference-of-convex decomposition, as shown in Eqn (4). For instance, the logistic sigmoid loss function[16], $s(r) = 1/\{1+ \exp(r)\}$, can also be used in this framework. It is easy to verify its difference-of-convex decomposition as $s(r) = \phi'(r) - \varphi'(r)$, where:

$$\phi'(r) = \begin{cases} \dfrac{1}{1+\exp(r)} & \text{if } r \geq 0, \\ \dfrac{1}{2} - \dfrac{1}{4}r & \text{if } r < 0; \end{cases} \quad \text{and} \quad \varphi'(r) = \begin{cases} 0 & \text{if } r \geq 0, \\ \dfrac{1}{2} - \dfrac{1}{4}r - \dfrac{1}{1+\exp(r)} & \text{if } r < 0. \end{cases}$$

Implementation

We have implemented Algorithm 2 in the R package SRDnet. SRD involves two tuning parameters λ_1 and λ_2. λ_1 plays a more significant role in variable selection, making the performance highly sensitive to its value.

We first generate a pre-specified set of λ_2 values by ignoring λ_1 and using a similar procedure as in SVMs. Then for each pre-specified λ_2, we compute the solution path using a fine grid of λ_1 values to achieve better tuning for λ_1. We start with finding $\lambda_{\max}$,

Algorithm 2. The d.c. algorithm for SRD.

Set $\in$ to a small quantity, say, 10^{-5}.
Initialize (β, β_0).

Initialize $\kappa_i^{(0)}$ by (5), for i =1, ..., n.
repeat

Compute $(\hat{\beta}, \hat{\beta}_0)$ by using GIST to solve (6).

Update $r_i = y_i\left(\mathbf{x}_i^T\hat{\beta} + \hat{\beta}_0\right)$, for i =1, ..., n.

Update $\kappa_i^{(t+1)}$ by (5), for i =1, ..., n.

until $\|\kappa^{(t+1)} - \kappa^{(t)}\|_{\infty} < \in$

which is the smallest λ_1 that sets $\boldsymbol{\beta} = 0$. According to the Karush-Kuhn-Tucker (KKT) conditions[24], $\lambda_{\max}$ satisfies at least one of the following inequalities,

$$\lambda_{\max} \leq \tilde{\lambda}_{\max,j} := \frac{2}{n}\max\left(\sum_{i=1}^{n}\max\left(y_i x_{ij}, 0\right), \sum_{i=1}^{n}\max\left(-y_i x_{ij}, 0\right)\right)$$

for j =1, ..., d. Let $\tilde{\lambda}_{\max} = \max_j \tilde{\lambda}_{\max,j}$. We first set $\lambda_1 = \tilde{\lambda}_{\max}$ and gradually decrease λ_1 in Algorithm 2 to locate $\lambda_{\max}$. We set $\lambda_{\min} = \tau\lambda_{\max}$, where τ is a user-defined quantity. The default value of τ is $\tau = 0.01$ for $n < d$ data and $\tau = 0.0001$ for $n \geq d$ data. Between $\lambda_{\min}$ and $\lambda_{\max}$, we place K points uniformly in the log-scale. The default value for K is 98 such that the solution path has 100 λ_1 values. A warm-start trick, as in GCDnet[22], is also implemented. Specifically, for computing the solution of the $(k + 1)$'th λ_1, the solution at the k'th λ_1 is used as the initial value in Algorithm 2.

Simulations

In this section, we conducted simulation studies to evaluate the performance of the proposed SRD method. The training data consisted of 100 subjects (50 cases and 50 controls), where each input $\mathbf{x}$ was a d = 500-dimensional vector. Independent tuning and testing data were generated in the same way as the training data. The sample sizes for the tuning and testing data were 100 and 10,000, respectively. The tuning data were used to select the optimal tuning parameters via a grid search, and the testing data were used to evaluate the accuracy of various classification methods.

We considered two scenarios similar to those in Wang et al.[25]. In the first scenario, all covariates are independent. The '+' class follows a normal distribution with a mean of:

$$\mu_+ = \left(\underbrace{0.5, \cdots, 0.5}_{10}, \underbrace{0, \cdots, 0}_{490}\right)^T$$

and a covariance matrix $\boldsymbol{\Sigma} = \mathbf{I}_{d\times d}$. The '−' class follows a similar distribution except that the mean is:

$$\mu_- = \left(\underbrace{-0.5, \cdots, -0.5}_{10}, \underbrace{0, \cdots, 0}_{490}\right)^T$$

The Bayes optimal classification rule depends solely on the first 10 covariates, with a Bayes error of 5.69%. The second scenario represents an example where the relevant covariates are correlated. The '+' class follows a normal distribution with a mean of:

$$\mu_+ = \left(\underbrace{1, \cdots, 1}_{10}, \underbrace{0, \cdots, 0}_{490}\right)^T$$

and a covariance matrix:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}^*_{10\times 10} & 0_{10\times 490} \\ 0_{490\times 10} & \mathbf{I}_{490\times 490} \end{pmatrix}$$

where, the diagonal elements of $\mathbf{\Sigma}^*$ are 1 and all off-diagonal elements are 0.8. The '−' class has a similar distribution except that the mean is:

$$\mu_- = \left(\underbrace{-1,\cdots,-1}_{10},\underbrace{0,\cdots,0}_{490}\right)^T$$

The Bayes optimal classification rule depends only on the first 10 highly correlated covariates, and its Bayes error is 13.47%.

We compared the standard SVM, hybrid huberized SVM (HHSVM)[25], and the proposed SRD method. HHSVM uses a smoothed hinge loss, called huberized hinge loss, combined with an elastic-net penalty. SVM was implemented using the e1071 R package, and the R package GCDnet[22] was used for HHSVM.

We used the tuning dataset to select the tuning parameters for each method. For SVM, we searched over a pre-specified set of tuning parameter values for C, and selected the one that minimized the prediction error on the tuning data. For HHSVM and SRD, we chose a pre-specified set of λ_2 values, and adaptively determined a fine grid of λ_1 values using the procedure described in the methods, with $\tau = 0.01$ and $K = 98$. The best pair (λ_1, λ_2) was selected based on the smallest prediction error on the tuning data. The performance of the selected models was then evaluated on independent testing data. Each experiment was repeated 1,000 times.

The mean prediction errors and corresponding standard deviations (in parentheses) are summarized in Table 1 (first two rows). As shown in Table 1, the prediction errors of SRD and HHSVM are similar, and outperform those of SVM in both scenarios. This improvement is attributed to the variable selection mechanism employed by SRD and HHSVM. Unlike SVM, which uses all covariates and is thereby affected by noise covariates, SRD and HHSVM are more robust due to their ability to select relevant features. The similar performance of SRD and HHSVM in scenario 2 highlights that SRD effectively identifies all correlated relevant covariates, a result of the 'grouping effect' provided by the elastic-net penalty, as in HHSVM.

The computation of SRD is higher than that of HHSVM due to the nature of non-convex optimization. At our computational facility, HHSVM took approximately 26.6 min for scenario I with 1,000 repeats. In contrast, with the same number of pre-specified (λ_1, λ_2) pairs for tuning, SRD took approximately 145.2 min for 1,000 repeats.

We perturbed the training data by randomly selecting *perc*(10% or 20%) of the samples and flipping their class labels to the opposite. We repeated the above testing procedure on the perturbed training data. The results are also presented in Table 1. The performance of all methods deteriorated due to the perturbation. However, SRD was the least affected, as expected. This supports our claim that the non-convex smoothed ramp loss enhances the robustness of the model compared to convex loss functions, leading to more accurate classifiers in the presence of outliers in the training data.

We also compared the covariates selected by HHSVM and SRD (SVM retains all covariates). We consider q_{signal}, the number of selected relevant covariates, and q_{noise}, the number of selected noise covariates. The results are presented in Table 2. When there is no perturbation in the training data, both HHSVM and SRD perform similarly. However, in the presence of perturbation, SRD tends to select more relevant covariates. Despite this, both methods also select a significant number of noise covariates. One way to further remove the noise covariates is to use tuning methods that favor more parsimonious models. For example, applying the 'one-standard error' rule[26] during cross-validation for parameter tuning has proven effective in screening out noise.

Table 1. Mean (std) of prediction errors evaluated on independent test data for two simulation scenarios.

		SVM	HHSVM	SRD
perc = 0%	Scenario I	$0.230_{(0.014)}$	$0.084_{(0.015)}$	$0.083_{(0.015)}$
	Scenario II	$0.171_{(0.013)}$	$0.146_{(0.010)}$	$0.144_{(0.009)}$
perc = 10%	Scenario I	$0.285_{(0.021)}$	$0.131_{(0.032)}$	$0.112_{(0.023)}$
	Scenario II	$0.204_{(0.025)}$	$0.154_{(0.014)}$	$0.149_{(0.012)}$
perc = 20%	Scenario I	$0.340_{(0.025)}$	$0.216_{(0.056)}$	$0.172_{(0.040)}$
	Scenario II	$0.253_{(0.039)}$	$0.174_{(0.033)}$	$0.153_{(0.015)}$

The number of training samples is 100, the number of total covariates is 500, and the number of relevant covariates is 10. The reported results are the average test errors over 1,000 repetitions on a test set of 10,000 samples, with the values in parentheses indicating the corresponding standard deviations. In scenario I, the covariates are independent, while in scenario II, the relevant covariates are highly correlated. perc: describes the percent of training samples randomly selected to have their class labels flipped. Specifically, perc = 0% indicates no perturbation in the training data, perc = 10% corresponds to a 10% perturbation, and perc = 20% corresponds to a 20% perturbation.

Data analysis

In this section, we investigated the performance of SVM, HHSVM, and our proposed SRD method using microarray gene expression data from a colon cancer study[2]. This dataset includes 62 samples, consisting of 40 colon cancer tumors and 22 normal tissues. Each sample contains expression measurements for 2,000 genes obtained using an Affymetrix gene chip. The data are publicly available.

We pre-processed the data following the procedure described by Dudoit et al.[27]. Microarray gene expression levels below the minimum threshold of 100 were set to 100, and the maximum threshold was capped at 16,000. We excluded genes with max/min ≤ 5 and (max − min) ≤ 500, where max and min denote the maximum and minimum expression levels of a gene across samples. The expression levels were then logarithmically transformed. After thresholding, filtering, and logarithmic transformation, the data were standardized to have a zero mean and unit standard deviation for each gene. The pre-processed data (p = 1,224) were used for all subsequent analysis. It has been reported that five samples (three tumors and two normal tissues) were contaminated[28], making this dataset particularly suitable for assessing the robustness of the proposed SRD method.

We compared the performance of SVM, HHSVM, and SRD. All methods involve at least one tuning parameter. We used 10-fold cross-validation to select the optimal parameters. Since the sample size is small, the cross-validation error is known to have high variance. To mitigate this, we repeated the cross-validation procedure ten times and averaged the cross-validation errors to reduce the variance. We selected the parameter with the smallest average

Table 2. Comparison of variable selection.

		HHSVM		SRD	
		q_{signal}	q_{noise}	q_{signal}	q_{noise}
perc = 0%	Scenario I	$9.8_{(0.6)}$	$28.0_{(38.0)}$	$9.7_{(0.6)}$	$17.5_{(30.9)}$
	Scenario II	$6.0_{(3.4)}$	$50.8_{(93.7)}$	$6.5_{(3.4)}$	$53.7_{(94.6)}$
perc = 10%	Scenario I	$8.8_{(1.3)}$	$22.0_{(29.1)}$	$9.2_{(1.0)}$	$17.0_{(21.0)}$
	Scenario II	$6.9_{(3.1)}$	$41.9_{(86.6)}$	$7.5_{(2.9)}$	$45.5_{(82.9)}$
perc = 20%	Scenario I	$7.0_{(2.0)}$	$36.9_{(56.3)}$	$7.7_{(1.8)}$	$28.5_{(36.7)}$
	Scenario II	$6.3_{(3.2)}$	$41.2_{(83.1)}$	$7.5_{(3.0)}$	$31.4_{(71.3)}$

The setups are the same as those described in Table 1. q_{signal} is the number of selected relevant covariates, and q_{noise} is the number of selected noise covariates. The results are averages over 1000 repetitions, and the numbers in parentheses are the standard deviations.

cross-validation error. This tuning strategy is referred to as the optimal rule in this article. For HHSVM and SRD, we aimed to choose a parsimonious model if its prediction performance was not compromised too much. A common approach, the 'one-standard error' rule[26] is used in cross-validation to select the most parsimonious model whose prediction error is no more than one standard error above the optimal error. In this work, we used a slightly different way to compute the standard error to incorporate into the repeated cross-validation procedure. That is, the standard error was computed through the individual prediction error rates for each of the ten repetitions. Here, we applied both tuning strategies: the optimal rule and the 'one-standard error' rule.

To evaluate classification performance, we conducted a nested 10-fold cross-validation procedure[29]. The data were randomly partitioned into ten roughly equal-sized folds. In each iteration, nine folds were used as training data, where an internal cross-validation procedure was performed to tune the model parameters. The classification model was then trained on the full training set using the selected parameters and applied to predict class labels for the left-out fold. This process was repeated nine more times to obtain predictions for all folds, yielding the overall prediction error. For a reliable evaluation of prediction accuracy, we repeated the outer cross-validation procedure 100 times with different fold partitions. The average and standard deviation of these 100 prediction errors are reported in Table 3. During the cross-validation procedure with 100 repetitions, we trained a total of 1,000 classification models for each method. For HHSVM and SRD, the average number of selected genes is also reported in Table 3.

We can see that SRD seems to have a better classification accuracy than HHSVM. It confirms again the robustness of SRD in classification and variable selection when there exist outliers in the training data. Interestingly, SVM performed similarly with SRD although SVM used all covariates. The reason perhaps is that the gene expression levels are highly correlated for many genes. The correlation enhances the signal. In the simulation studies, both scenarios have ten relevant covariates, and the only difference is that in scenario II the relevant covariates are highly correlated. SVM performed much better in scenario II than in scenario I.

We used two tuning strategies. From Table 3, the 'one-standard error' rule is effective to select a parsimonious model without sacrificing too much on the prediction accuracy. We investigated the genes selected in the 1,000 SRD models determined by the 'one-standard error' rule during the cross-validation procedure. The top 20 most frequently selected genes are listed in Table 4.

The molecular carcinogenesis of colorectal cancer is complex and poorly understood. Some comments about the top frequently selected genes are worthy of mention. MYH9 and its regulatory MYL9 are components of the cytoskeletal network involved in actomyosin-based contractility and implicated in the invasive behavior of tumor cells[30]. hnRNP A1, a major member of the hnRNP family, binds to G-rich repetitive sequences and quadruplex structures in DNA. The overexpression of hnRNP A1 in colorectal cancer cells leads to evasion of cancer cell apoptosis[31]. Uroguanylin, encoded by GUCA2B, is markedly down-regulated in adenocarcinomas of the colon, and the treatment with uroguanylin leads to induction of apoptosis in human colon carcinoma cells *in vitro*[32]. S100P overexpressed in colorectal carcinomatous tissues, and the overexpression is significantly correlated with tumor metastasis, advanced clinical stage, and recurrence[33].

Table 3. Results on the colon cancer dataset.

	Optimal rule		'One-standard error' rule	
	Test error	No. of genes	Test error	No. of genes
SVM	$12.3\%_{(1.7\%)}$	1,224		
HHSVM	$13.6\%_{(2.1\%)}$	$86.2_{(100.1)}$	$13.7\%_{(2.0\%)}$	$60.5_{(78.5)}$
SRD	$11.9\%_{(1.9\%)}$	$219.0_{(136.4)}$	$12.3\%_{(2.2\%)}$	$140.4_{(100.8)}$

Test errors are averages of 100 cross validation repetitions. The numbers of genes are averages of selected genes in 1,000 classification models. The numbers in parentheses are the corresponding standard deviations.

Table 4. The 20 most frequently selected genes by SRD with the 'one-standard error' rule from the colon cancer dataset.

Accession	Gene name	Freq.
R87126	Myosin, heavy chain 9, non-muscle (MYH9)	990
M76378	Cysteine and glycine-rich protein 1 (CSRP1)	989
J02854	Myosin, light chain 9, regulatory (MYL9)	988
M63391	Desmin (DES)	983
X12671	Heterogeneous nuclear ribonucleoprotein A1 (hnRNP A1)	983
Z50753	Guanylate cyclase activator 2B (GUCA2B)	981
R36977	General transcription factor IIIA	973
T47377	S100P Protein	972
J03040	Secreted protein, acidic, cysteine-rich (SPARC)	970
X14958	High mobility group AT-hook 1 (HMGA1)	967
T51571	P24480 Calgizzarin	967
H43887	Complement factor D precursor	966
L07648	MAX interactor 1, dimerization protein (MXI1)	957
M22382	Heat shock 60kDa protein 1 (HSPD1)	955
H64489	Tetraspanin 1 (TSPAN1)	945
H20709	Myosin, light chain 6, alkali (MYL6)	942
H40095	Macrophage migration inhibitory factor	935
R84411	Small nuclear ribonucleoprotein associated proteins B	929
T51023	Heat shock protein 90kDa alpha (cytosolic)	922
T71025	Metallothionein 1G (MT1G)	916

Conclusions

In this article, we have proposed the Sparse Ramp Discrimination (SRD) method, which leverages the smoothed ramp loss to assess the 'badness-of-fit' and incorporates the elastic-net penalty for automatic variable selection. We apply the difference of the convex (d.c.) algorithm to efficiently solve the non-convex optimization problem through a sequence of convex subproblems. From the simulation studies and real data analysis, SRD is robust to outliers, making it particularly well-suited for handling noisy, high dimensional biological data.

Author contributions

The authors confirm contribution to the paper as follows: study conception and design: Zhou X; data collection: Zhou X; analysis and interpretation of results: Zhou X, Wang Z; manuscript preparation: Zhou X, Wang Z. All authors reviewed the results and approved the final version of the manuscript.

Data availability

R codes are available on GitHub (https://github.com/xinzhoubiostat/SRDnet.git).

Acknowledgments

We thank the editor and two referees for their thoughtful and constructive comments, and Dr. Gang Xu for assistance in editing this manuscript. Zhou X was partially supported by NIH grant

R03CA252808, and Wang Z was partially supported by NIH grant R01LM014087.

Conflict of interest

The authors declare that they have no conflict of interest.

Dates

Received 31 October 2024; Revised 12 February 2025; Accepted 18 February 2025; Published online 26 February 2025

References

1. Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, et al. 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286:531–37
2. Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, et al. 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the United States of America* 96:6745–50
3. Brown MP, Grundy WN, Lin D, Cristianini N, Sugnet CW, et al. 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences of the United States of America* 97:262–67
4. Furey TS, Cristianini N, Duffy N, Bednarski DW, Schummer M, et al. 2000. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics* 16:906–14
5. Guyon I, Weston J, Barnhill S, Vapnik V. 2002. Gene selection for cancer classification using support vector machines. *Machine learning* 46:389–422
6. Zhu J, Rosset S, Hastie T, Tibshirani R, Zhu J, et al. 2003. 1-norm support vector machines. *Proceedings of the 17th International Conference on Neural Information Processing Systems, 9–11 December 2003, Whistler, British Columbia, Canada*. Cambridge, MA, United States: MIT Press. pp. 49–56. https://dl.acm.org/doi/10.5555/2981345.2981352
7. Tibshirani R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58:267–88
8. Wang L, Zhu J, Zou H. 2006. The doubly regularized support vector machine. *Statistica Sinica* 16:589–615
9. Zou H, Hastie T. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 67:301–20
10. Liang Y, Liu C, Luan XZ, Leung KS, Chan TM, et al. 2013. Sparse logistic regression with a $L_{1/2}$ penalty for gene selection in cancer classification. *BMC Bioinformatics* 14:198
11. Zhang HH, Ahn J, Lin X, Park C. 2006. Gene selection using support vector machines with non-convex penalty. *Bioinformatics* 22:88–95
12. Shen X, Tseng GC, Zhang X, Wong WH. 2003. On ψ-learning. *Journal of the American Statistical Association* 98:724–34
13. Collobert R, Sinz F, Weston J, Bottou L. 2006. Trading convexity for scalability. *Proceedings of the 23rd international conference on Machine learning, 25–29 June 2006, Pittsburgh, Pennsylvania, USA*. New York, USA: ACM. pp. 201–8. doi: 10.1145/1143844.1143870
14. Wu Y, Liu Y. 2007. Robust truncated hinge loss support vector machines. *Journal of the American Statistical Association* 102:974–83
15. Mason L, Baxter J, Bartlett P, Frean M. 1999. Boosting algorithms as gradient descent. *Advances in neural information processing systems*. Cambridge, MA, United States: MIT Press. pp. 512–18. https://dl.acm.org/doi/10.5555/3009657.3009730
16. Bartlett PL, Jordan MI, McAuliffe JD. 2006. Convexity, classification, and risk bounds. *Journal of the American Statistical Association* 101:138–56
17. Zhou X, Tuck DP. 2007. MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data. *Bioinformatics* 23:1106–14
18. Fan J, Li R. 2001. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association* 96:1348–60
19. Thi Hoai An L, Dinh Tao P. 1997. Solving a class of linearly constrained indefinite quadratic problems by DC algorithms. *Journal of global optimization* 11:253–85
20. Yuille AL, Rangarajan A. 2003. The concave-convex procedure. *Neural computation* 15:915–36
21. Friedman J, Hastie T, Tibshirani R. 2010. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33:1–22
22. Yang Y, Zou H. 2013. An efficient algorithm for computing the HHSVM and its generalizations. *Journal of Computational and Graphical Statistics* 22:396–415
23. Gong P, Zhang C, Lu Z, Huang J, Ye J. 2013. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. *In international conference on machine learning* 28:37–45
24. Boyd S, Vandenberghe L. 2004. *Convex optimization*. Cambridge, UK: Cambridge University Press. doi: 10.1017/cbo9780511804441
25. Wang L, Zhu J, Zou H. 2008. Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics* 24:412–19
26. Hastie T, Tibshirani R, Friedman J. 2009. The elements of statistical learning: data mining, inference, and prediction. *New York, NY: Springer. doi: 10.1007/978-0-387-84858-7*
27. Dudoit S, Fridlyand J, Speed TP. 2002. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association* 97:77–87
28. Li L, Weinberg CR, Darden TA, Pedersen LG. 2001. Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method. *Bioinformatics* 17:1131–42
29. Ambroise C, McLachlan GJ. 2002. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences of the United States of America* 99:6562–66
30. Betapudi V, Licate LS, Egelhoff TT. 2006. Distinct roles of nonmuscle myosin II isoforms in the regulation of MDA-MB-231 breast cancer cell spreading and migration. *Cancer Research* 66:4725–33
31. Fujiya M, Konishi H, Mohamed Kamel MK, Ueno N, Inaba Y, et al. 2014. microRNA-18a induces apoptosis in colon cancer cells via the autophagolysosomal degradation of oncogenic heterogeneous nuclear ribonucleoprotein A1. *Oncogene* 33:4847–56
32. Shailubhai K, Yu HH, Karunanandaa K, Wang JY, Eber SL, et al. 2000. Uroguanylin treatment suppresses polyp formation in the Apc(Min/+) mouse and induces apoptosis in human colon adenocarcinoma cells via cyclic GMP. *Cancer Research* 60:5151–57
33. Dong L, Wang F, Yin X, Chen L, Li G, et al. 2014. Overexpression of S100P promotes colorectal cancer metastasis and decreases chemosensitivity to 5-FU in vitro. *Molecular and cellular biochemistry* 389:257–64

Copyright: © 2025 by the author(s). Published by Maximum Academic Press, Fayetteville, GA. This article is an open access article distributed under Creative Commons Attribution License (CC BY 4.0), visit https://creativecommons.org/licenses/by/4.0/.