Iterative RELIEF for Feature Weighting

Yijun Sun[†,‡] SUN@DSP.UFL.EDU
Jian Li[‡] LI@DSP.UFL.EDU
[†]Interdisciplinary Center for Biotechnology Research
[‡]Department of Electrical & Computer Engineering, University of Florida, Gainesville, FL 32611, USA

Abstract

We propose a series of new feature weighting algorithms, all stemming from a new interpretation of RELIEF as an online algorithm that solves a convex optimization problem with a margin-based objective function. The new interpretation explains the simplicity and effectiveness of RELIEF, and enables us to identify some of its weaknesses. We offer an analytic solution to mitigate these problems. We extend the newly proposed algorithm to handle multiclass problems by using a new multiclass margin definition. To reduce computational costs, an online learning algorithm is also developed. Convergence theorems of the proposed algorithms are presented. Some experiments based on the UCI and microarray datasets are performed to demonstrate the effectiveness of the proposed algorithms.

1. Introduction

Feature selection is one of the fundamental problems in machine learning. The role of feature selection is critical, especially in applications involving many irrelevant features. Yet, compared to classifier design, much rigorous theoretical treatment to feature selection is needed. Most feature selection algorithms rely on heuristic searching and thus cannot provide any guarantee of optimality. This is largely due to the difficulty in defining an objective function that can be easily optimized by some well-established optimization techniques. It is particularly true for the wrapper methods that use nonlinear classifiers to evaluate the goodness of selected feature subsets. This problem can to some extent be alleviated by using feature weighting, which assigns to each feature a real-valued number, instead of a binary one, to indicate its relevance to a learning problem. Among the existing feature weighting algorithms, RELIEF [Kira & Rendell, 1992] is considered one of the most successful ones due to its simplicity and effectiveness [Dietterich, 1997]. We have shown that RELIEF is an online solution to a convex optimization problem, maximizing a margin-based objective function. The margin is defined based on a 1-NN classifier. Therefore, compared with filter methods, RELIEF usually performs better due to the performance feedback of a nonlinear classifier when searching for useful features; compared with wrapper methods, by optimizing a convex problem, RELIEF avoids *any* exhaustive or heuristic combinatorial search and thus can be implemented very efficiently. These two merits make RELIEF particularly suitable for large-scale problems such as DNA microarray.

The new interpretation of RELIEF allows us to identify some weaknesses of the algorithm and to propose some solutions to fix them. One major drawback of RELIEF is that it makes an implicit assumption that the nearest neighbors of a pattern found in the original feature space are the ones in the weighted space, which is highly unlikely in practical applications. Moreover, RELIEF lacks a mechanism to eliminate outlier data. We offer an analytic solution to mitigate theses two issues. In Section 3, we propose a new feature weighting algorithm, referred to as I-RELIEF, by following the principle of the EM algorithm. I-RELIEF treats the nearest neighbors and identity of a pattern as hidden random variables, and iteratively estimates feature weights until convergence. We provide a convergence theorem for I-RELIEF, which shows that under certain conditions, I-RELIEF converges to a unique solution regardless of initial starting points. In Section 4, we extend I-RELIEF to multiclass problems by using a new multiclass margin definition. In order to speed up learning process, in Section 5, we develop an online I-RELIEF algorithm and prove its convergence. Finally, in Section 6, we conduct some experiments based on UCI and microarray datasets to demonstrate the effectiveness of the proposed algorithms.

2. Optimization Approach to RELIEF

We first present a brief review of RELIEF. Let $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{N} \in \mathbb{R}^I \times \{\pm 1\}$ denote a training dataset. The

Appearing in *Proceedings of the 23^{rd} International Conference on Machine Learning*, Pittsburgh, PA, USA, 2006.

key idea of RELIEF is to iteratively estimate the feature weights according to their ability to discriminate between neighboring patterns. In each iteration, a pattern $\mathbf{x}$ is randomly selected and then two nearest neighbors of $\mathbf{x}$ are found, one from the same class (termed the *nearest hit* or NH) and the other from the different class (termed the *nearest miss* or NM). The weight of the i-th feature is then updated as: $w_i = w_i + |\mathbf{x}^{(i)} - \text{NM}^{(i)}(\mathbf{x})| - |\mathbf{x}^{(i)} - \text{NH}^{(i)}(\mathbf{x})|$.

Below we present a new interpretation of RELIEF from the optimization point of view. We first define the margin for pattern $\mathbf{x}_n$ as $\rho_n = d(\mathbf{x}_n - \text{NM}(\mathbf{x}_n)) - d(\mathbf{x}_n - \text{NH}(\mathbf{x}_n))$ [Gilad-Bachrach et al., 2004], where $d(\cdot)$ is a distance function defined as $d(\mathbf{x}) = \sum_i |x_i|$. Note that $\rho_n > 0$ if only if $\mathbf{x}_n$ is correctly classified by 1-NN. One natural idea is to scale each feature such that the averaged margin in a weighted feature space is maximized:

$$\begin{array}{ll} \max_{\mathbf{w}} & \sum_{n=1}^{N} \left(\sum_{i=1}^{I} w_i |\mathbf{x}_n^{(i)} - \text{NM}^{(i)}(\mathbf{x}_n)| \right. \\ & \left. - \sum_{i=1}^{I} w_i |\mathbf{x}_n^{(i)} - \text{NH}^{(i)}(\mathbf{x}_n)| \right) , \\ \text{s.t.} & \|\mathbf{w}\|_2^2 = 1, \mathbf{w} \geqslant 0 , \end{array} \tag{1}$$

where the constraint $\|\mathbf{w}\|_2^2 = 1$ prevents the maximization from increasing without bound, and $\mathbf{w} \geqslant 0$ ensures that the weight vector is a distance metric. By defining $\mathbf{z} = \sum_{n=1}^{N} |\mathbf{x}_n - \text{NM}(\mathbf{x}_n)| - |\mathbf{x}_n - \text{NH}(\mathbf{x}_n)|$, where $|\cdot|$ is the point-wise absolute operator, Eq. (1) can be simplified as: $\max_{\mathbf{w}} \mathbf{w}^{\mathrm{T}}\mathbf{z}$, s.t. $\|\mathbf{w}\|_2^2 = 1, \mathbf{w} \geqslant 0$. By using the Lagrangian technique, the solution can be expressed as $\mathbf{w} = \frac{1}{2\lambda}(\mathbf{z} + \boldsymbol{\theta})$, where λ and $\boldsymbol{\theta} \geqslant 0$ are the Lagrangian multipliers, satisfying $\boldsymbol{\theta}^T\mathbf{w} = 0$. With the Karush-Kuhn-Tucker condition, it is easy to verify the following three cases: (1) $z_i = 0 \Rightarrow \theta_i = 0 \Rightarrow w_i = 0$; (2) $z_i > 0 \Rightarrow z_i + \theta_i > 0 \Rightarrow w_i > 0 \Rightarrow \theta_i = 0$; and (3) $z_i < 0 \Rightarrow \theta_i > 0 \Rightarrow w_i = 0 \Rightarrow z_i = -\theta_i$. It follows that the optimum solution can be calculated in a closed form as $\mathbf{w} = (\mathbf{z})^+ / \|(\mathbf{z})^+\|_2$, where $(z_i)^+ = \max(z_i, 0)$.

By comparing the expression of $\mathbf{w}$ with the update rule of RELIEF, we conclude that RELIEF is an online solution to the optimization scheme Eq. (1). This is true except when $w_i = 0$ for $z_i \leq 0$, which usually corresponds to irrelevant features. From the above analysis, we find that RELIEF may be the only algorithm that utilizes the performance of a highly nonlinear classifier yet results in a simple convex problem with a closed-form solution. This clearly explains the simplicity and effectiveness of RELIEF.

Other distance functions can be also used. If the Euclidean distance is used, the resulting algorithm is Simba [Gilad-Bachrach et al., 2004]. However, Simba returns many local maxima, for which the mitigation offered in Simba is to restart the algorithm from several starting points. Hence the acquisition of the global minimum is not guaranteed through its invocation.

3. Iterative RELIEF Algorithm

Two major drawbacks of RELIEF become clear from the objective function in Eq. (1): first, the nearest neighbors are defined in the original feature space, which are highly unlikely to be the ones in the weighted space; second, the objective function optimized by RELIEF is actually the average margin. In the presence of outliers, some margins can take very negative values. In a highly noisy data case with a large amount of irrelevant features or mislabelling, the aforementioned two issues can become so severe that the performance of RELIEF may be greatly deteriorated. A heuristic algorithm, called RELIEF-F [Kononenko, 1994], has been proposed to address the first problem. RELIEF-F averages K, instead of just one, nearest neighbors in computing the sample margins. Empirical studies have shown that RELIEF-F can achieve significant performance improvement over the original RELIEF. As for the second problem, to our knowledge, no such algorithm exists. In this section, we propose an analytic solution capable of handling these two issues simultaneously.

We first define two sets: $\mathcal{M}_n = \{i : 1 \leq i \leq N, y_i \neq y_n\}$ and $\mathcal{H}_n = \{i : 1 \leq i \leq N, y_i = y_n, i \neq n\}$, associated with each pattern $\mathbf{x}_n$. Suppose now that we have known, for each pattern $\mathbf{x}_n$, its nearest hit and miss, the indices of which are saved in the set $\mathcal{S}_n = \{(s_{n1}, s_{n2})\}$, where $s_{n1} \in \mathcal{M}_n$ and $s_{n2} \in \mathcal{H}_n$. For example, $s_{n1} = 1$ and $s_{n2} = 2$ mean that the nearest miss and hit of $\mathbf{x}_n$ are $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. We also denote $\mathbf{o} = [o_1, \cdots, o_N]^{\mathrm{T}}$ as a set of binary parameters, such that $o_n = 0$ if $\mathbf{x}_n$ is an outlier, or $o_n = 1$ otherwise. Then the objective function we want to optimize is $C(\mathbf{w}) = \sum_{\{n=1, o_n=1\}}^{N} \left(\|\mathbf{x}_n - \mathbf{x}_{s_{n1}}\|_{\mathbf{w}} - \|\mathbf{x}_n - \mathbf{x}_{s_{n2}}\|_{\mathbf{w}} \right)$, which can be easily optimized by using the conclusion drawn in Section 2. Of course, we do not know the set $\mathcal{S} = \{\mathcal{S}_n\}_{n=1}^{N}$ and the vector $\mathbf{o}$. However, if we assume the elements of $\{\mathcal{S}_n\}_{n=1}^{N}$ and $\mathbf{o}$ are random variables, we can proceed by deriving the probability distributions of the unobserved data. We first make a guess on the weight vector $\mathbf{w}$. By using the pairwise distances that have been computed when searching for the nearest hits and misses, the probability of the i-th data point being the nearest miss of $\mathbf{x}_n$ can be naturally defined as $P_m(i|\mathbf{x}_n, \mathbf{w}) = \frac{f(\|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}})}{\sum_{j \in \mathcal{M}_n} f(\|\mathbf{x}_n - \mathbf{x}_j\|_{\mathbf{w}})}$. Similarly, the probability of the i-th data point being the nearest hit of $\mathbf{x}_n$ is $P_h(i|\mathbf{x}_n, \mathbf{w}) = \frac{f(\|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}})}{\sum_{j \in \mathcal{H}_n} f(\|\mathbf{x}_n - \mathbf{x}_j\|_{\mathbf{w}})}$, where $f(\cdot)$ is a kernel function. One commonly used example is $f(d) = \exp(-d/\sigma)$, where the kernel width σ is a user defined parameter. Likewise, the probability of $\mathbf{x}_n$ being an outlier can be readily defined as $P_o(o_n = 0|\mathcal{D}, \mathbf{w}) = \frac{\sum_{i \in \mathcal{M}_n} f(\|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}})}{\sum_{\mathbf{x}_i \in \mathcal{D} \setminus \mathbf{x}_n} f(\|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}})}$.

Now we are ready to derive the following iterative algorithm. Although we adopt the idea of the EM algorithm

that treats unobserved data as random variables, it should be noted that the following method is not an EM algorithm since the objective function is not a likelihood. For brevity of notation, we define $\alpha_{i,n} = P_m(i|\mathbf{x}_n, \mathbf{w}^{(t)}), \beta_{i,n} = P_h(i|\mathbf{x}_n, \mathbf{w}^{(t)}), \gamma_n = 1 - P_o(o_n = 0|\mathcal{D}, \mathbf{w}^{(t)}), \mathcal{W} = \{\mathbf{w} : \|\mathbf{w}\|_2 = 1, \mathbf{w} \geq 0\}, \mathbf{m}_{n,i} = |\mathbf{x}_n - \mathbf{x}_i|$ if $i \in \mathcal{M}_n$, and $\mathbf{h}_{n,i} = |\mathbf{x}_n - \mathbf{x}_i|$ if $i \in \mathcal{H}_n$.

Step-1: After t-th iteration, the Q function is calculated as:

$$\begin{aligned} & Q(\mathbf{w}|\mathbf{w}^{(t)}) = \mathrm{E}_{\{\mathcal{S},\mathbf{o}\}}[C(\mathbf{w})] \ , \\ & = \sum_{n=1}^{N} \gamma_n \Big(\sum_{i \in \mathcal{M}_n} \alpha_{i,n} \|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}} - \sum_{i \in \mathcal{H}_n} \beta_{i,n} \|\mathbf{x}_n - \mathbf{x}_i\|_{\mathbf{w}} \Big) \ , \\ & = \sum_{n=1}^{N} \gamma_n \Big(\sum_j w_j \underbrace{\sum_{i \in \mathcal{M}_n} \alpha_{i,n} m_{n,i}^j}_{\bar{m}_n^j} - \sum_j w_j \underbrace{\sum_{i \in \mathcal{H}_n} \beta_{i,n} h_{n,i}^j}_{\bar{h}_n^j} \Big) \ , \\ & = \mathbf{w}^T \sum_{n=1}^{N} \gamma_n (\bar{\mathbf{m}}_n - \bar{\mathbf{h}}_n) = \mathbf{w}^T \boldsymbol{\nu} \ , \end{aligned} \tag{2}$$

Step-2: The re-estimation of $\mathbf{w}$ in the $(t+1)$-th iteration is: $\mathbf{w}^{(t+1)} = \arg\max_{\mathbf{w} \in \mathcal{W}} Q(\mathbf{w}|\mathbf{w}^{(t)}) = (\boldsymbol{\nu})^+ / \|(\boldsymbol{\nu})^+\|_2$. The above two steps iterate alternatively until convergence, i.e., $\|\mathbf{w}^{(t+1)} - \mathbf{w}^{(t)}\| < \theta$.

We name the above algorithm as iterative RELIEF, or short I-RELIEF. Since P_m, P_h and P_o return us with reasonable probability estimates, and the re-estimation of $\mathbf{w}$ is a convex optimization problem, we expect a good convergence behavior and reasonable performance from I-RELIEF. We provide a convergence analysis below.

3.1. Convergence Analysis

We begin by studying the asymptotic behavior of I-RELIEF. If $\sigma \to +\infty$, we have $\lim_{\sigma \to +\infty} P_m(i|\mathbf{x}_n, \mathbf{w}) = 1/|\mathcal{M}_n|$ for $\forall \mathbf{w} \in \mathcal{W}$ since $\lim_{\sigma \to +\infty} f(d) = 1$. On the other hand, if $\sigma \to 0$, by assuming that for $\forall n, d_{in} \triangleq \|\mathbf{x}_i - \mathbf{x}_n\|_{\mathbf{w}} \neq d_{jn}$ if $i \neq j$, it can be shown that $\lim_{\sigma \to 0} P_m(i|\mathbf{x}_n, \mathbf{w}) = 1$ if $d_{in} = \min_{j \in \mathcal{M}_n} d_{jn}$ and 0 otherwise. $P_h(i|\mathbf{x}_n, \mathbf{w})$ and $P_o(n|\mathbf{w})$ can be computed similarly. We observe that if $\sigma \to 0$, I-RELIEF is equivalent to iterating the original RELIEF (NM = NH = 1) provided that outlier removal is not considered. In our experiments, we rarely observe that the resulting algorithm converges. On the other hand, if $\sigma \to +\infty$, I-RELIEF converges in one step because the term $\boldsymbol{\nu}$ in Eq. (2) is a constant vector for any initial feature weights. This suggests that the convergence behavior of I-RELIEF and the convergent rates are fully controlled by the choice of the kernel width. In the following, we present a proof by using the Banach fixed point theorem. We first state the theorem without proof. For detailed proofs, we refer to [Kress, 1998].

Definition 1. *Let $\mathcal{U}$ be a subset of a norm space $\mathcal{Z}$, and $\|\cdot\|$ is a norm defined in $\mathcal{Z}$. An operator $T : \mathcal{U} \to \mathcal{Z}$ is called a contraction operator if there exists a constant $q \in [0, 1)$ such that $\|T(x) - T(y)\| \leq q\|x - y\|$ for $\forall x, y \in \mathcal{U}$. q is called the contraction number of T.*

Definition 2. *An element of a norm space $\mathcal{Z}$ is called a fixed point of $T : \mathcal{U} \to \mathcal{Z}$ if $T(x) = x$.*

Theorem 1. *Let T be a contraction operator mapping a complete subset $\mathcal{U}$ of a norm space $\mathcal{Z}$ into itself. Then the sequence generated as $x^{(t+1)} = T(x^{(t)})$, $t = 0, 1, 2, \cdots$ with arbitrary $x^{(0)} \in \mathcal{U}$ converges to the unique fixed point x^* of T. Moreover, the following error bounds hold:*

$$\begin{aligned} & \|x^{(t)} - x^*\| \leq \tfrac{q^t}{1-q} \|x^{(1)} - x^{(0)}\| \ , \\ \textit{and} \quad & \|x^{(t)} - x^*\| \leq \tfrac{q}{1-q} \|x^{(t)} - x^{(t-1)}\| \ . \end{aligned} \tag{3}$$

In order to apply the fixed point theorem to prove the convergence of I-RELIEF, the gist is to identify the contraction operator in I-RELIEF and check if all conditions in Theorem 1 are met. To this end, let $\mathcal{P} = \{\mathbf{p} : \mathbf{p} = [P_m, P_h, P_o]\}$ and we specify the two steps of I-RELIEF in a functional form as $A1 : \mathcal{W} \to \mathcal{P}, A1(\mathbf{w}) = \mathbf{p}$ and $A2 : \mathcal{P} \to \mathcal{W}, A2(\mathbf{p}) = \mathbf{w}$. By indicating the functional composition by a circle ($\circ$), I-RELIEF can be written as $\mathbf{w}^{(t)} = (A2 \circ A1)(\mathbf{w}^{(t-1)}) \triangleq T(\mathbf{w}^{(t-1)})$, where $T : \mathcal{W} \to \mathcal{W}$. Since $\mathcal{W}$ is a closed subset of a norm space $\mathcal{R}^I$ and complete, T is an operator mapping a complete subset $\mathcal{W}$ into itself. However, it is difficult to directly verify that T is a contraction operator satisfying Definition 1. Noting that for $\sigma \to +\infty$, I-RELIEF converges with one step, we have $\lim_{\sigma \to +\infty} \|T(\mathbf{w}_1, \sigma) - T(\mathbf{w}_2, \sigma)\| = 0$ for $\forall \mathbf{w}_1, \mathbf{w}_2 \in \mathcal{W}$. Therefore, in the limit, T is a contraction operator with contraction constant $q = 0$, that is, $\lim_{\sigma \to +\infty} q(\sigma) = 0$. Therefore, for $\forall \varepsilon > 0$, there exists a $\bar{\sigma}$ such that $q(\sigma) \leq \varepsilon$ whenever $\sigma > \bar{\sigma}$. By setting $\varepsilon < 1$, the resulting operator T is a contraction operator. Combining the above arguments, we establish the following convergence result for I-RELIEF.

Theorem 2. *Let I-RELIEF be defined as above. There exists a $\bar{\sigma}$ such that $\lim_{t \to +\infty} \|\mathbf{w}^{(t)} - \mathbf{w}^{(t-1)}\| = 0$ for $\forall \sigma > \bar{\sigma}$. Moreover, for a fixed $\sigma > \bar{\sigma}$, I-RELIEF converges to the unique solution for any initial weight $\mathbf{w}^{(0)} \in \mathcal{W}$.*

Theorem 2 ensures the convergence of I-RELIEF but does not tell us how large a kernel width should be. In our experiment, we find that using a relative large σ value, say $\sigma > 0.5$, the convergence is guaranteed. Also, the error bound in Ineq. (3) tells us that the smaller the contraction number q, the tighter the error bound and hence the larger the convergence rate. Since it is difficult to explicitly express q as a function of σ, it is difficult to prove that q monotonically decreases with σ. However, in general, a larger kernel width yields a larger convergence rate, which

is experimentally confirmed in Section 6.2. It is also worthwhile to emphasize that unlike other machine learning algorithms, such as neural networks, the convergence and the solution of I-RELIEF are not affected by the initial value if the kernel width is fixed.

4. Extension to Multiclass RELIEF

The original RELIEF algorithm can only handle binary problems. RELIEF-F overcomes this limitation by modifying the weight update rule as: $w_i = w_i + \sum_{\{c\in\mathcal{Y},c\neq y(\mathbf{x})\}} \frac{P(c)}{1-P(c)}|\mathbf{x}^{(i)} - \mathrm{NM}_c^{(i)}(\mathbf{x})| - |\mathbf{x}^{(i)} - \mathrm{NH}^{(i)}(\mathbf{x})|$, where $\mathcal{Y} = \{1,\cdots,C\}$ is the label space, $\mathrm{NM}_c(\mathbf{x})$ is the nearest miss of $\mathbf{x}$ from class c, and $P(c)$ is the *a priori* probability of class c. By using the conclusions drawn in Section 2, it can be shown that RELIEF-F is equivalent to defining a sample margin as: $\rho = \sum_{\{c\in\mathcal{Y},c\neq y(\mathbf{x})\}} \frac{P(c)}{1-P(c)} d(\mathbf{x} - \mathrm{NM}_c(\mathbf{x})) - d(\mathbf{x} - \mathrm{NH}(\mathbf{x}))$. Note that a positive sample margin does not necessarily imply a correct classification. The extension of RELIEF-F to the iterative version is quite straightforward, and therefore we skip the detailed derivations here. We name the resulting algorithm as I-RELIEF-1.

From the commonly used margin definition for multiclass problems, however, it is more natural to define a margin as:

$$\begin{aligned}\rho &= \min_{\{c\in\mathcal{Y},c\neq y(\mathbf{x})\}} d(\mathbf{x} - \mathrm{NM}_c(\mathbf{x})) - d(\mathbf{x} - \mathrm{NH}(\mathbf{x}))\ , \\ &= \min_{\{\mathbf{x}_i\in\mathcal{D}\setminus\mathcal{D}_{y(\mathbf{x})}\}} d(\mathbf{x} - \mathbf{x}_i) - d(\mathbf{x} - \mathrm{NH}(\mathbf{x})),\end{aligned}$$

where $\mathcal{D}_c$ is a subset of $\mathcal{D}$ containing only the patterns from class c. Compared to the first definition, this definition regains the property that a positive sample margin corresponds to a correct classification. The derivation of the iterative version of multiclass RELIEF using the new margin definition, which we call I-RELIEF-2, is straightforward.

5. Online Learning

I-RELIEF is based on batch learning, i.e. feature weights are updated after seeing all of the training data. In the cases where the amount of training data is huge, online learning is computationally much more attractive than batch learning. In this section, we derive an online algorithm for I-RELIEF. Convergence analysis is also presented.

Recall that in I-RELIEF, one needs to compute $\boldsymbol{\nu} = \sum_{n=1}^{N} \gamma_n(\bar{\mathbf{m}}_n - \bar{\mathbf{h}}_n)$. Analogously, in online learning, after the T-th iteration, we may consider computing $\boldsymbol{\nu}^{(T)} = \frac{1}{T}\sum_{t=1}^{T} \gamma^{(t)}(\bar{\mathbf{m}}^{(t)} - \bar{\mathbf{h}}^{(t)})$. Denote $\boldsymbol{\pi}^{(t)} = \gamma^{(t)}(\bar{\mathbf{m}}^{(t)} - \bar{\mathbf{h}}^{(t)})$. It is easy to show that $\boldsymbol{\nu}^{(T)} = \boldsymbol{\nu}^{(T-1)} + \frac{1}{T}(\boldsymbol{\pi}^{(T)} - \boldsymbol{\nu}^{(T-1)})$. By defining $\eta^{(T)} = 1/T$ as a learning rate, the above formulation states that the current estimate can be simply computed as a linear combination of the previous estimate and the current observation. Moreover, it suggests that other learning rates are possible. One simple example is to set $\eta^{(T)} = 1/aT$ with $a \in (0, 1]$. Due to space limitation, more comprehensive consideration of online I-RELIEF is presented elsewhere. Below we establish the convergence property of online I-RELIEF. We first present a useful lemma without proof.

Lemma 1. *Let $\{a_n\}$ be a bounded sequence, i.e. for $\forall n$, $M_1 \leq a_n \leq M_2$. If $\lim_{n\to+\infty} a_n = a^*$, then $\lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^{n} a_i = a^*$.*

Theorem 3. *Online I-RELIEF converges when the learning rate is appropriately selected. If both algorithms converge, I-RELIEF and online I-RELIEF converge to the same solution.*

Proof. The proof of the first part of the theorem can be easily done by recognizing that the above formulation has the same form as the Robbins-Moron stochastic approximation algorithm [Kushner & Yin, 2003]. The conditions on the learning rate $\eta^{(t)} : \lim_{t\to+\infty} \eta^{(t)} = 0, \sum_{t=1}^{+\infty} \eta^{(t)} = +\infty$, and $\sum_{t=1}^{+\infty} (\eta^{(t)})^2 < +\infty$ ensure the convergence of online I-RELIEF. $\eta^{(t)} = 1/t$ meets the above conditions.

Now we prove the second part of the theorem. To eliminate the randomness, instead of randomly selecting a pattern from $\mathcal{D}$, we divide the data into blocks, denoted as $\mathcal{B}^{(m)} = \mathcal{D}$. Online I-RELIEF successively performs online learning over $\mathcal{B}^{(m)}, m = 1, 2, \cdots$. For each block, denote $\tilde{\boldsymbol{\pi}}^{(m)} = \frac{1}{N}\sum_{t=(m-1)\times N+1}^{m\times N} \boldsymbol{\pi}^{(t)}$. After running over M blocks of data, we have $\boldsymbol{\nu}^{(M\times N)} = \frac{1}{M\times N}\sum_{t=1}^{M\times N} \boldsymbol{\pi}^{(t)} = \frac{1}{M}\sum_{m=1}^{M} \tilde{\boldsymbol{\pi}}^{(m)}$. From the proof of the first part, we know that $\lim_{t\to+\infty} \boldsymbol{\nu}^{(t)} = \boldsymbol{\nu}^*$. It follows that $\lim_{m\to+\infty} \tilde{\boldsymbol{\pi}}^{(m)} = \tilde{\boldsymbol{\pi}}^*$. Using Lemma 1, we have $\lim_{M\to+\infty} \boldsymbol{\nu}^{(M\times N)} = \tilde{\boldsymbol{\pi}}^* = \boldsymbol{\nu}^*$. The last equality is due to the fact that a convergent sequence cannot have two limits.

We prove the convergence of online I-RELIEF to I-RELIEF by using the uniqueness of the fixed point for a contraction operator. Recall that if the kernel width is appropriately selected, $T : \mathcal{W} \to \mathcal{W}$ is a contraction operator for I-RELIEF, i.e., $T(\mathbf{w}^*) = \mathbf{w}^*$. We then construct an operator $\widetilde{T} : \mathcal{W} \to \mathcal{W}$ for online I-RELIEF, which, in the m-th iteration, uses $\tilde{\mathbf{w}}^{(m-1)} = (\boldsymbol{\nu}^{((m-1)\times N)})^+/\|(\boldsymbol{\nu}^{((m-1)\times N)})^+\|_2$ as input, and then computes $\boldsymbol{\nu}^{(m\times N)}$ by performing online learning on $\mathcal{B}^{(m)}$, and finally returns $\tilde{\mathbf{w}}^{(m)} = (\tilde{\boldsymbol{\pi}}^{(m)})^+/\|(\tilde{\boldsymbol{\pi}}^{(m)})^+\|_2$. Since $\lim_{t\to+\infty} \boldsymbol{\nu}^{(t)} = \boldsymbol{\nu}^* = \tilde{\boldsymbol{\pi}}^*$, it follows that as $m \to +\infty$, we have $\widetilde{T}(\tilde{\mathbf{w}}^*) = \tilde{\mathbf{w}}^*$, where $\tilde{\mathbf{w}}^* = (\boldsymbol{\nu}^*)^+/\|\boldsymbol{\nu}^*\|_2$. Therefore, $\tilde{\mathbf{w}}^*$ is the fixed point of $\widetilde{T}$. The only difference between T and $\widetilde{T}$ is that $\widetilde{T}$ performs online learning while T does not. Since $\{\boldsymbol{\nu}^{(t)}\}$ is convergent, it is also a Cauchy sequence. In other words, as $m \to +\infty$, the difference between every pair of $\boldsymbol{\nu}$ within one block goes to zero with respect to some norms. The operator $\widetilde{T}$, therefore, is identical to T in the limit. It follows that $\tilde{\mathbf{w}}^* = \mathbf{w}^*$, since

otherwise there would be two fixed points for a contraction operator, which contradicts Theorem 1. □

One major advantage of RELIEF and its variations over other algorithms is their computational efficiency. The complexity of RELIEF, I-RELIEF and online I-RELIEF are $\mathcal{O}(TNI)$, $\mathcal{O}(TN^2I)$ and $\mathcal{O}(TNI)$, respectively, where T is the total number of iterations, I is the feature dimensionality and N is the number of data points. If RELIEF runs over the entire dataset, i.e., $T = N$, then the complexity is $\mathcal{O}(N^2I)$. In the following section, we show that online I-RELIEF can attain similar solutions to I-RELIEF after one pass of the training data. Therefore, online I-RELIEF has the same computational cost as RELIEF.

Table 1. Data Summary of 6 UCI and 6 Microarray Datasets

Dataset	Train	Test	Feature	Class
twonorm	400	7000	20	2
waveform	400	4600	21	2
ringnorm	400	7000	20	2
f-solar	666	400	9	2
thyroid	140	75	5	2
segmentation	210	2100	19	7
9-tumors	60	/	5726	9
Brain-tumor2	60	/	10367	4
Leukemia-1	72	/	5327	3
Prostate-tumors	83	/	2308	4
SRBCT	102	/	10509	2
DLBCL	77	/	5469	2

6. Experiments

We conduct large-scale experiments to demonstrate the effectiveness of the proposed algorithms. The ultimate goal of this study is for gene selection based on microarray data, where the true gene set is typically unknown. It is necessary to conduct experiments in a controlled manner. Therefore, we perform experiments on two test-beds. The first test-bed is composed of 6 datasets: *twonorm*, *waveform*, *ringnorm*, *f-solar*, *thyroid*, and *segmentation*, all publicly available at the UCI Machine Learning Repository. The data information is summarized in Table 1. We add 50 independently Gaussian distributed irrelevant features to each pattern, representing different levels of signal-to-noise ratios. In real applications, it is also possible that some patterns are mislabelled. To evaluate the robustness of each algorithm against mislabelling, we introduce noise to training data but keep test data intact. The level of noise represents a percentage of randomly selected training data whose class labels are changed. The second test-bed contains six microarray datasets: *9-tumors*, *Brain-tumor2*, *Leukemia-1*, *prostate-tumors*, *DLBCL* and *SRBCT*. One characteristic of microarray data, different from most of the classification problems, is the large feature dimensionality compared to small sample numbers. For more detailed information on these data, see [Statnikov et al., 2005] and the references therein. For all datasets, except for a simple re-scaling of each feature value to be between 0 and 1 as required in RELIEF, no other pre-processing is performed.

We use two metrics to evaluate the performance of the algorithms. The first is the classification errors commonly used in the literature. The second is the ROC (receiver operating characteristic) based metric, where we treat feature selection as a target recognition problem. Though features in the original feature sets may be weakly relevant or even useless, it is reasonable to assume that these features contain at least the same or more information than the useless ones artificially added. Therefore, by changing a threshold, we can plot a ROC curve, which gives us a direct view on the capabilities of each algorithm to identify useful features and at the same time rule out useless ones.

6.1. Experiments on UCI Datasets

We first perform experiments on the UCI datasets. To make the experiment feasible, a KNN classifier is used to compute the classification errors for each algorithm. The number of the nearest neighbors K, the kernel width σ of I-RELIEF and the number of NH and NM of RELIEF-F are estimated through a stratified 10-fold cross validation (CV) using training data. The code of Simba used in the study is downloaded from [Gilad-Bachrach et al., 2004]. The parameters are set to be the default values, but we increase the number of the passes of training data to be 5 instead of the default value 1.

To reduce statistical variations, each algorithm is run 20 times for each dataset. In each run, a dataset is randomly partitioned into training and testing. The testing results, measured with two performance metrics, are plotted in Fig.1. We see that with respect to classification errors, in nearly all datasets, I-RELIEF performs the best, RELIEF-F the second and Simba the worst. For a more rigorous comparison between I-RELIEF and RELIEF-F, a significance test is also performed. The optimum number of features used in KNN is estimated through 10-fold CV using training data. We report that at the 0.05 p-value level, I-RELIEF wins on 7 cases (*ringnorm* $(50/10)$, *twonorm* $(50/10)$, *thyroid* $(50/0)$, *waveform* and *f-solar*), and ties with RELIEF-F on the remaining 5 cases. (In the notation $50/10$, the first number refers to the number of irrelevant features and the second one the percentage of mislabelled samples.) The reason that I-RELIEF ties with RELIEF-F on *segmentation* is self-explained in Fig.1. We also check with the ROC metric: in almost all datasets, I-RELIEF has the largest area under ROC curves, RELIEF-F the second and Simba the smallest. We find that for *thyroid* and *ringnorm* $(50/0)$, though there are no significant differences in classification errors, it is clear from the ROC metric that I-RELIEF has

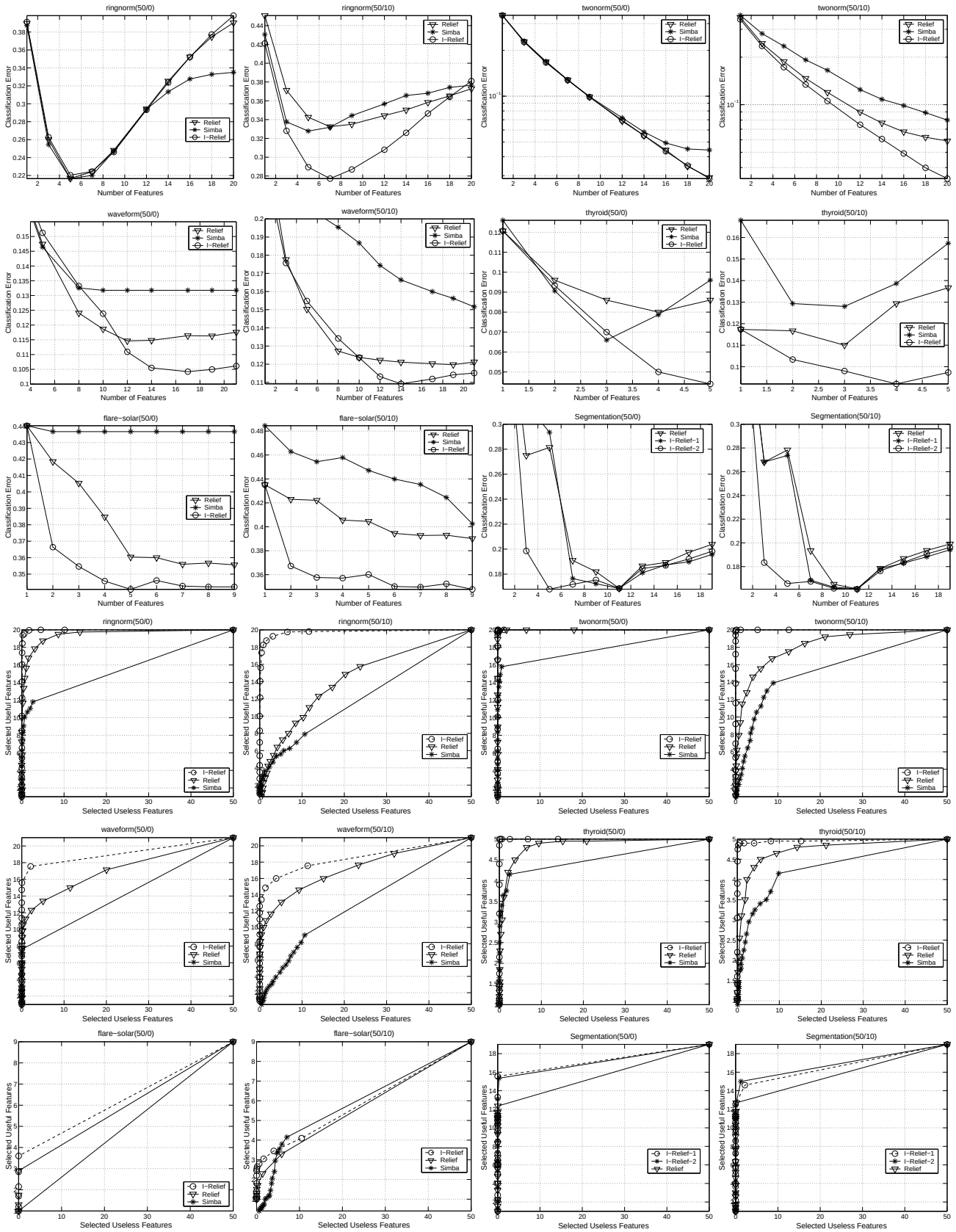


Figure 1. Comparison of three algorithms using the classification error and ROC metrics on 6 UCI datasets.

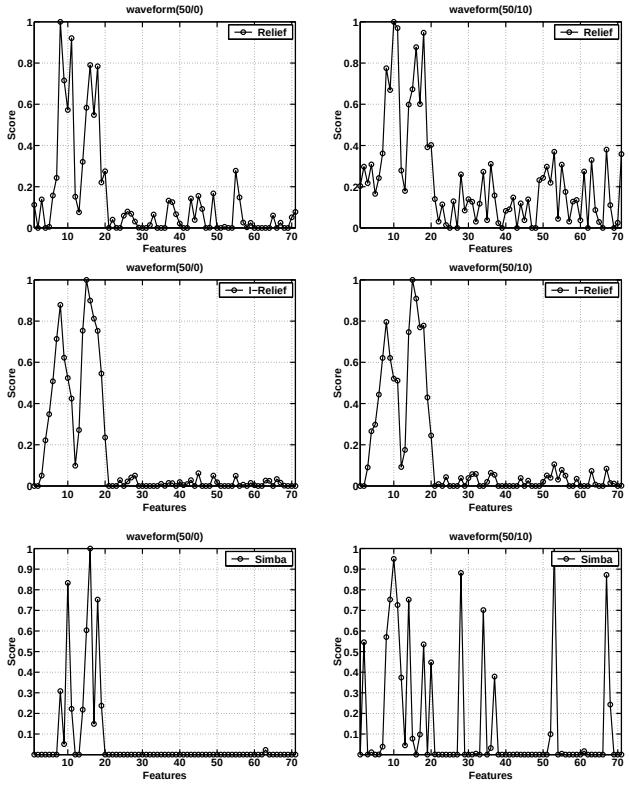


Figure 2. Feature weights learned in three algorithms using *waveform* dataset. The first 21 features are presumably useful.

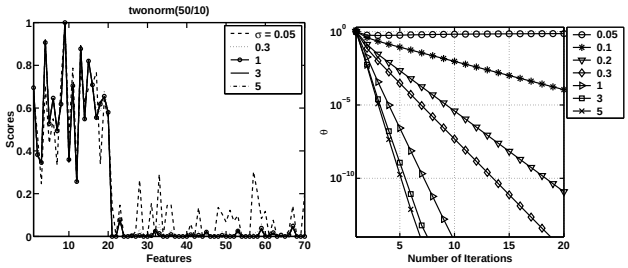
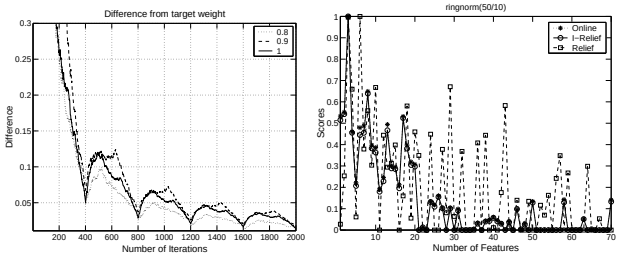


Figure 3. Feature weights and convergence rates with different σ using *twonorm* dataset.



(a) (b)

Figure 4. Convergence analysis of online I-RELIEF on *ringnorm* dataset.

better solution quality than RELIEF-F.

To further demonstrate the behavior of each algorithm, we particularly focus on the dataset *waveform*. We plot the learned feature weights of one realization in Fig. 2. Without mislabelling, the weights learned in RELIEF-F are similar to those of I-RELIEF but the former have larger weights on the useless features than the latter. It is interesting to note that in *waveform* (50/0), Simba assigns zero weights to not only useless features but also some presumably useful features. In this case, we need to go back to the classification error metric. We observe that the test error of Simba flats after the tenth feature since except for these 10 features, the weights of the remaining features are all zeros. It indicates that Simba in effect does not identify all of the useful features. With 10% mislabelling, the solution qualities of both RELIEF-F and Simba degrade significantly while I-RELIEF performs similarly as before. For example, Simba mistakenly identifies an irrelevant feature as the top feature. These observations imply that both Simba and RELIEF are not robust against label noise.

6.2. Choice of Kernel Width

The kernel width σ is the only parameter of I-RELIEF and can be estimated through CV on training data. It is well-known that the CV method may result in an estimate with a large variance. Fortunately, this problem does not pose a serious concern. In Fig. 3, we plot the feature weights and the convergence rates of I-RELIEF with different σ values using the *twonorm* dataset. We observe that the algorithm diverges when $\sigma = 0.05$; but for relative large σ values, the algorithm always converges, and the resulting feature weights do not have much difference. This indicates that the performance of I-RELIEF is not sensitive to the choice of σ values, which makes model selection easy in real applications. Moreover, with the increase of σ value, the convergence becomes faster.

6.3. Online Learning

We perform some experiments to verify the convergence results established in Section 5. The feature weights learned in I-RELIEF are used as the target vector. The stopping criterion θ is set to be 10^{-5} to ensure that the target vector is a good approximation of the true solution (c.f., Eq.(3)). We only present the results of *ringnorm* since the results for other datasets are almost identical. The convergence results with different learning rates ($\eta^{(t)} = 1/at$), averaged from 20 runs, are plotted in Fig. 4(a). We observe that online I-RELIEF converges to I-RELIEF, which confirms the theoretical findings in Theorem 3. We also find that after 400 iterations (*ringnorm* has 400 training samples), the feature weights are already very close to the tar-

get vector (Fig. 4(b)). For comparison, the feature weights learned in RELIEF-F are also plotted. From this experiment, we conclude that online I-RELIEF can greatly reduce the computational cost of I-RELIEF while retaining its performance.

6.4. Experiments on Microarray

We apply RELIEF-F, I-RELIEF-1 and I-RELIEF-2 to six microarray datasets. Due to the limited sample numbers, the leave-one-out method is used to evaluate the performance of each algorithm.

The classification errors of KNN as a function of the 500 top ranked features are plotted in Fig. 5. Since *Prostate-Tumor* and *DLBCL* are binary problems, I-RELIEF-1 is equivalent to I-RELIEF-2. From the figure, we observe that, except for *DLBCL*, for which I-RELIEF performs similarly to RELIEF-F, I-RELIEF-2 is the clear winner among the three algorithms. Also, I-RELIEF-1 ties with RELIEF-F on three datasets (*9-Tumors*, *DLBCL* and *Brain-Tumors*) but outperforms RELIEF-F on the remaining three datasets. For comparison, we report the classification errors of KNN using all genes. We can see that gene selection can significantly improve the KNN performance.

We note that the numbers of genes found by I-RELIEF are all less than 200. With these small gene sets, oncologists may be able to work on them directly to infer the molecular mechanism underlying disease causes. Currently, we are working closely with oncologists to check the biological significance of the top ranked genes identified by our algorithms. Also, if for classification purposes, some computationally expensive methods (e.g.wrapper methods) can be used to further filter out some redundant genes. By using some sophisticated classification algorithms such as SVM, much improvement on classification performance is expected. Building such a classification system is our future work.

7. Conclusion

We have proposed several new feature weighting algorithms, all stemming from a simple yet informative explanation of RELIEF. We have experimentally demonstrated that our algorithms perform significantly better than RELIEF and Simba. Moreover, considering many heuristical approaches used in feature selection, we believe that the contribution of this paper is not merely limited to the algorithmic aspects. The I-RELIEF algorithms, as one of the first feature weighting methods that have a clearly defined objective function and can be solved through numerical analysis instead of combinatorial searching, provide a promising direction for more rigorous treatment of the feature weighting and selection problems.

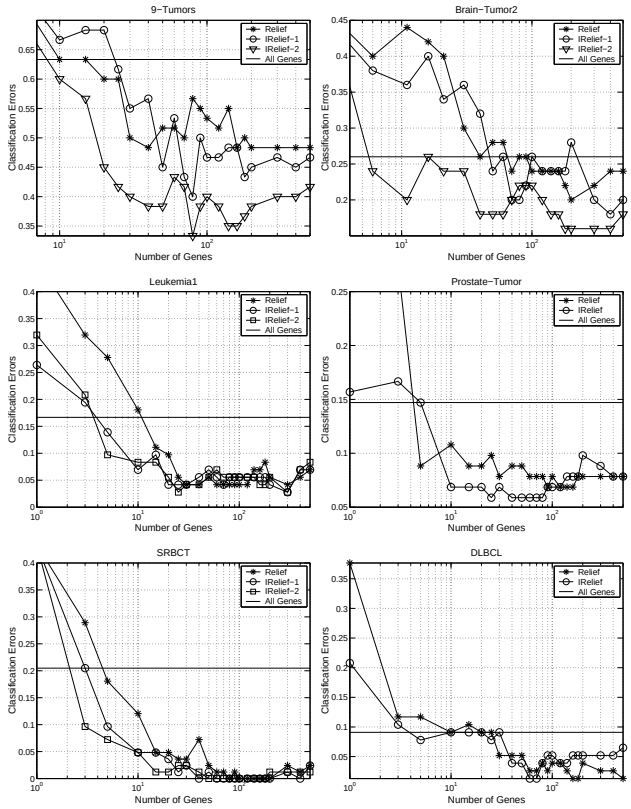


Figure 5. Classification errors on six microarray datasets.

References

Dietterich, T. G. (1997). Machine learning research: Four current directions. *AI Magazine*, *18*, 97–136.

Gilad-Bachrach, R., Navot, A., & Tishby, N. (2004). Margin based feature selection - theory and algorithms. *the 21st International Conference on Machine Learning*.

Kira, K., & Rendell, L. A. (1992). A practical approach to feature selection. *the 9th International Conference on Machine Learning* (pp. 249 – 256). Morgan Kaufmann.

Kononenko, I. (1994). Estimating attributes: Analysis and extensions of RELIEF. *European Conference on Machine Learning* (pp. 171–182).

Kress, R. (1998). *Numerical analysis*. New York: Springer-Verlag.

Kushner, H., & Yin, G. (2003). *Stochastic approximation and recursive algorithms and applications*. New York: Springer-Verlag. 2 edition.

Statnikov, A., Aliferis, C., Tsamardinos, I., Hardin, D., & Levy, S. (2005). A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, *21*, 631–643.