Direct Zero-norm Optimization for Feature Selection

Kaizhu Huang
Department of Engineering Mathematics
University of Bristol
Bristol BS8 1TR, United Kingdom
K.Huang@bris.ac.uk

Irwin King, Michael R. Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{king, lyu}@cse.cuhk.edu.hk

Abstract

Zero-norm, defined as the number of non-zero elements in a vector, is an ideal quantity for feature selection. However, minimization of zero-norm is generally regarded as a combinatorially difficult optimization problem. In contrast to previous methods that usually optimize a surrogate of zero-norm, we propose a direct optimization method to achieve zero-norm for feature selection in this paper. Based on Expectation Maximization (EM), this method boils down to solving a sequence of Quadratic Programming problems and hence can be practically optimized in polynomial time. We show that the proposed optimization technique has a nice Bayesian interpretation and converges to the true zero norm asymptotically, provided that a good starting point is given. Following the scheme of our proposed zero-norm, we even show that an arbitrary-norm based Support Vector Machine can be achieved in polynomial time. A series of experiments demonstrate that our proposed EM based zero-norm outperforms other state-of-the-art methods for feature selection on biological microarray data and UCI data, in terms of both the accuracy and the learning efficiency.

1 Introduction

Zero-norm, defined as $||\mathbf{w}||_0^0 = card\{w_i|w_i \neq 0\}$ for a given n-dimensional vector $\mathbf{w}$ where $card$ means the cardinality of a set, is an important concept in pattern recognition, data mining, and machine learning. More specifically, zero-norm directly conveys the sparse concept and can be used in machine learning [13, 6], especially in feature selection [15]. In the feature selection context, the task is to select a subset of features while preserving the discriminative ability for a classifier. Minimization of zero-norm provides a natural and ideal way to attack such a problem.

However, as shown by Amaldi and Kann, minimization of zero-norm is a combinatorially very difficult problem [1]. In the literature, there are several proposals to deal with this problem. Bradley et al. [4] and Bradley and Mangasarian [3] proposed an approximation method called FSV for feature selection. In this model, the zero-norm is approximated as $||\mathbf{w}||_0^0 = card\{w_i|w_i \neq 0\} \approx \sum_i 1 - \exp\{1 - \alpha|w_i|\}$, where α is a parameter to be tuned. This approximation is further adopted in finding the sparse kernel classifier [8]. In an alternative approach, Weston et al. proposed the so-called two-norm/one-norm approximation of the zero-norm minimization method called AROM [15]. They explored $\sum_i \ln(\epsilon + |w_i|)$ as a surrogate of zero-norm in optimization, where $0 < \epsilon \ll 1$ is a parameter. Both models have demonstrated their effectiveness in performing feature selection. However, there are two shortcomings for these methods. First, both methods are merely approximations to the true zero-norm. On the one hand, it is usually hard to know how accurate such approximations might be; on the other hand, optimizing an approximation term instead of the true objective function might sometimes be wasteful in terms of computer resources. Indeed, as demonstrated later in the paper, these two approaches consume much time in finding a given number of features. Second, as observed from their approximation formulae, two extra parameters are introduced. Although the authors have suggested ideas on how to choose these parameters, it remains uncertain whether the proposed means of setting the parameters works in data with different statistical natures.

In contrast to the previous methods, which attempted to optimize certain surrogates, we propose a direct optimization of zero-norm implementation based on Bayesian learning. The proposed algorithm boils down to solving a sequence of Quadratic Programming problems. Hence the original combinatorial difficult problem can be transformed to one in polynomial time. More importantly, we show that the proposed optimization technique has a nice Bayesian interpretation and converges to the true zero norm asymptotically, provided that a good starting point is given. In addition, different with the above methods, no extra parameters are introduced in our model for feature selection. Indeed, to our best knowledge, this is the first study that can achieve a

1550-4786/08 $25.00 © 2008 IEEE
DOI 10.1109/ICDM.2008.60

IEEE computer society

direct optimization of zero-norm.

The rest of the paper is organized as follows. In the next section, we describe the direct zero-norm implementation in details. The problem definition, Bayesian derivations, main results, and the optimization models will be presented in this section in turn. We then provide a series of experiments to demonstrate the advantages of our proposed approach with respect to the accuracy and the computational time. After that, we discuss some limitations and issues related to this work. Finally, we set out our conclusion with some final remarks.

2 Asymptotically True Zero-norm

In this section, we first present the problem definition. Following that, we show that a hierarchical Bayesian model can asymptotically achieve the zero-norm directly. In line with the Bayesian approach, we then present the main results and demonstrate how to achieve the zero-norm for feature selection as well as sparse classification .

2.1 Problem Definition

Suppose we are given a training data set $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$, where the input pattern $\mathbf{x}_i \in R^n$ is i.i.d. sampled from $\mathcal{X}$ and the output label $y_i \in \{\pm 1\}$[1]. The goal of feature selection is to select a minimum number of features while preserving or even increasing the discriminative ability of the classifier, defined as $f(\mathbf{w}, b) = \mathbf{w} \cdot \mathbf{x} + b$ ($\mathbf{w} \neq 0 \in R^n$, $b \in R$). The problem of using zero-norm for feature selection can be formulated as follows:

$$\min_{\mathbf{w},b} ||\mathbf{w}||_0^0 + C \sum_{i=1}^{l} \xi_i \quad (1)$$

$$\text{s.t.} \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, l, (2)$$

where C is a penalty parameter used to trade off the empirical error $\sum_{i=1}^{l} \xi_i$ and the zero-norm term.

The above optimization naturally achieves the goal of feature selection, i.e., the number of features is minimized by $||\mathbf{w}||_0^0$, while maintaining the accuracy of the classifier $f(\mathbf{w}, b) = \mathbf{w} \cdot \mathbf{x} + b$ by minimization of the empirical error $\sum_{i=1}^{l} \xi_i$. However, as shown in [1], the problem involving the zero-norm is combinatorially difficult to optimize. In the following, motivated from a hierarchical Bayesian model within the context of classification, we show how to achieve the true zero-norm asymptotically for feature selection.

2.2 Hierarchical Bayesian Model

The Bayesian approaches often treat the output z of the learned linear classifier as corrupted by a zero-mean and unit-variance variable o, i.e., $z(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{h}(\mathbf{x}) + o$, where $\mathbf{h}(\mathbf{x})$ can either be a linear vector function, $\mathbf{h}(\mathbf{x}) = [1, \mathbf{x}]^T$ or be defined as a kernel vector $[1, k(\mathbf{x}, \mathbf{x}_1), \ldots, k(\mathbf{x}, \mathbf{x}_l)]^T$, where $k(\mathbf{x}, \cdot)$ is a given kernel function. Here the bias term b is incorporated as the first element in $\mathbf{w}$.

Given the training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$, we could simply write the Gaussian noise corrupted formula as a matrix form $\mathbf{z} = \mathbf{H}\mathbf{w} + \mathbf{o}$, where $\mathbf{H}$ is defined as $[\mathbf{h}(\mathbf{x}_1), \ldots, \mathbf{h}(\mathbf{x}_l)]^T$, and $\mathbf{o}$ is a vector with each element as a zero-mean and unit-variance Gaussian variable. We further assume a hierarchical prior probability for $\mathbf{w}$ in two stages as follows [6, 7].

$$\begin{array}{ll} \text{Stage I} & p(w_i|\tau_i) = \mathcal{N}(w_i|0, \tau_i) \\ \text{Stage II} & p(\tau_i) \propto 1/\tau_i, \tau_i > 0 \end{array} \quad (3)$$

If $\mathbf{z}$ is treated as missing variables, the EM algorithm [5] can be used to find the Maximum A Posterior $\mathbf{w}$ iteratively.

More specifically, in the E-step, since z_i is a Gaussian distribution centered at $\mathbf{w} \cdot \mathbf{h}(\mathbf{x}_i)$, but left-truncated at zero if $y_i = +1$ and right-truncated at zero if $y_i = -1$, the expectation of z_i can be expressed in a closed form as

$$\mathrm{E}[z_i|\widehat{\mathbf{w}}_{(t)}, \mathbf{y}] = \begin{cases} \widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i) + \frac{\mathcal{N}(\widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i)|0,1)}{1 - \mathcal{S}(-\widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i)|0,1)} & \text{if} \quad y_i = 1 \\ \widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i) - \frac{\mathcal{N}(\widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i)|0,1)}{\mathcal{S}(-\widehat{\mathbf{w}}_{(t)}^T \mathbf{h}(\mathbf{x}_i)|0,1)} & \text{if} \quad y_i = -1 \end{cases} \quad (4)$$

where $\mathcal{S}(.|0, 1)$ denotes the probability under a cumulative normal distribution, and the subscript t represents the t-th step in the EM procedure.

Since τ_i^{-1} is also missing, we perform the expectation over τ_i^{-1} as follows:

$$\begin{aligned} \mathrm{E}[\tau_i^{-1}|\widehat{\mathbf{w}}_{(t)}, \mathbf{y}] &= \frac{\int_0^{+\infty} \frac{1}{\tau_i} p(\tau_i|\widehat{\mathbf{w}}_{(t)}, \mathbf{y}) d\tau_i}{\int_0^{+\infty} p(\tau_i|\widehat{\mathbf{w}}_{(t)}, \mathbf{y}) d\tau_i} \\ &= \frac{\int_0^{+\infty} \frac{1}{\tau_i} p(\tau_i) p(\widehat{\mathbf{w}}_{(t)}|\tau_i) d\tau_i}{\int_0^{+\infty} p(\tau_i) p(\widehat{\mathbf{w}}_{(t)}|\tau_i) d\tau_i} \\ &= |\widehat{w}_{i,(t)}|^{-2} \,. \end{aligned} \quad (5)$$

On the other hand, the complete log-posterior to be maximized in M-step can be written as follows:

$$\begin{aligned} \log p(\mathbf{w}|\mathbf{y}, \mathbf{z}) &\propto \log p(\mathbf{z}|\mathbf{w}) + \log p(\mathbf{w}) \\ &\propto -||\mathbf{H}\mathbf{w} - \mathbf{z}||^2 - \mathbf{w}^T \mathbf{\Lambda} \mathbf{w}, \end{aligned} \quad (6)$$

where $\mathbf{\Lambda} = \mathrm{diag}(1/\tau_1, \ldots, 1/\tau_l)$. The first term corresponds to the errors between the output of the learned classifier $f(\mathbf{x}) = \mathbf{w}^T \mathbf{h}(\mathbf{x})$ and the actual output $z(\mathbf{x}, \mathbf{w})$; the

[1] In this paper, only two-category problems are considered. Multi-category problems can be easily approached by using a One Vs. One or One Vs. Others strategy.

second term represents the prior imposed by the assumption over $\mathbf{w}$.

If expectations (4), (5) are substituted into (6), the above maximization with respect to $\mathbf{w}$ can readily be computed in a closed form. The E and M steps are then conducted iteratively until a stable solution for $\mathbf{w}$ is obtained.

2.3 Main Results

In the following we propose three Propositions as a summary of the above derivations, showing the asymptotical equivalence between the hierarchical Bayesian model and the zero-norm.

Proposition 1. *The 2-level hierarchical-Bayes model $p(w_i|\tau_i) = N(w_i|0, \tau_i)$, $p(\tau_i) = 1/\tau_i$, $\tau_i > 0$ over w_i is equivalent to the zero-norm regularized classifier asymptotically.*

Proof. In the case $t \rightarrow \infty$, since the term $\widehat{w}_{i,(t)} = \widehat{w}_{i,(t+1)} = w_i$, maximizing the complete log-posterior in M-step $\log p(\mathbf{w}|\mathbf{y}, \mathbf{z})$ changes to maximizing $-||\mathbf{Hw} - \mathbf{z}||^2 - \mathbf{w}^T \mathbf{\Lambda w}$. This is equivalent to minimizing $||\mathbf{Hw} - \mathbf{z}||^2 + card\{w_i|w_i \neq 0\}$. The first term represents the empirical errors incurred by the classifier $\mathbf{w}$, while the second term is the zero-norm. Hence the 2-level hierarchical model is exactly equivalent to the zero-norm regularized classifier.

From the above proof, we know that the prior assumed in the zero-norm is only related to the second term. This directly elicits Proposition 2.

Proposition 2.*The prior assumed in zero-norm is only related to the term $\mathbf{w}^T \mathbf{\Lambda w}$ as defined in the EM process, where $\mathbf{\Lambda} = diag(1/\tau_1, \dots, 1/\tau_l)$, $1/\tau_i$ $(i = 1, \dots, l)$ can be iteratively updated by $|\widehat{w}_{i,(t)}|^{-2}$ for the zero-norm regularization.*

Interestingly, as shown in [7], another 2-level hierarchical-Bayes model $p(w_i|\tau_i) = N(w_i|0, \tau_i)$, $p(\tau_i) = (\gamma/2)\text{exp}(-\gamma\tau_i/2)$, $\tau_i > 0$ over w_i is equivalent to the one-norm regularized classifier $||\mathbf{w}||_1^1$ asymptotically [2]. Similarly, the one-norm is only related to the prior term $\mathbf{w}^T \mathbf{\Lambda w}$, where $\mathbf{\Lambda} = \text{diag}(1/\tau_1, \dots, 1/\tau_l)$ and $1/\tau_i$ $(i = 1, \dots, l)$ is updated by $\gamma|\widehat{w}_{i,(t)}|^{-1}$. Developed from the above propositions and the fact in [7], we make the following Proposition 3 for p-norm[3].

Proposition 3.*The priors assumed in $||\mathbf{w}||_p^p$ $(0 \leq p \leq 2$ or $p = \infty)$ are only related to the term $\mathbf{w}^T \mathbf{\Lambda w}$ as defined in the EM process, where $\mathbf{\Lambda} = diag(1/\tau_1, \dots, 1/\tau_l)$, $1/\tau_i$ $(i = 1, \dots, l)$ can be iteratively updated by $\gamma|\widehat{w}_{i,(t)}|^{-(2-p)}$ respectively.*

Proposition 3 posits the intriguing outcome that we can achieve the same effect without knowing the prior for p-norm explicitly. More interestingly, we can also define an L$_\infty$-norm [4] where $\mathbf{\Lambda} = \text{diag}(0, \dots, 0, 1/w_{i_{max},(t)}, 0, \dots, 0)$ is updated iteratively by a matrix with $w_{i_{max},(t)} = \max_i w_{i,(t)}$.

Remarks: Note that we exploit the above EM process to implement the zero-norm. Although EM has been widely used and also proved to be very successful in various fields covering pattern recognition, data mining, and machine learning, it only guarantees the convergence to local optimums [10]. When a good starting point is chosen, the EM might converge to a global optimum and hence achieves an asymptotically true zero-norm implementation. Despite of its local optimum property, our proposed EM implementation presents the first study that can directly achieve zero-norm for feature selection. This distinguishes our framework from those surrogate methods. Later empirical study also demonstrates the advantages of our novel approach over the traditional approximating methods, in terms of both the accuracy and the learning efficiency.

2.4 Zero-norm SVM for Feature Selection

Integrating the result from the previous section, we can achieve the zero-norm SVM for feature selection iteratively as follows:

$$\{\mathbf{w}^{(t)}, b^{(t)}\} = \arg\min_{\mathbf{w},b} C \sum_{i=1}^{l} \xi_i + \mathbf{w}^T \Lambda^{(t-1)} \mathbf{w} \quad (7)$$

$$\text{s.t.} \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, l \quad (8)$$

$$\Lambda^{(t)} = diag(1/|w_1^{(t-1)}|^2, \dots, 1/|w_n^{(t-1)}|^2). \quad (9)$$

The above objective function (7) is very similar to (6), which is maximized in the M-step. As mentioned before, the first term in (6) represents the loss between the output of the learned classifier $f(\mathbf{x}) = \mathbf{w}^T \mathbf{h}(\mathbf{x})$ and the ground truth output $z(\mathbf{x}, \mathbf{w})$. Similarly, the first term in (7) stands for the hinge loss among the training data. The optimization process is very similar to the EM process, except that the the maximized function in the M-step is changed to the negative of (7). Similar to the EM process, the above optimization will converge rapidly. Furthermore, at each iteration, the above optimization is easily verified to be a Quadratic Programming problem, since $\Lambda^{(t)}$, constructed by the vector $\mathbf{w}$ at the previous step $(t-1)$, is a constant at the current step. Hence the optimization finally boils downs to solving a sequence of Quadratic Programming problems and hence can be solved in polynomial time.

2.5 Using Zero-norm in Dual Space

In the above, zero-norm is exploited in the primal space, where the goal is to find the minimal number of features

[2] The one-norm of an vector $\mathbf{w}$ is defined as $||\mathbf{w}||_1^1 = \sum_i |w_i|$

[3] The p-norm of an vector $\mathbf{w}$ is defined as $||\mathbf{w}||_p = (\sum_i |w_i|^p)^{\frac{1}{p}}$

[4] The infinity-norm of an vector $\mathbf{w}$ is defined as $||\mathbf{w}||_\infty = \max_i |w_i|$

while preserving the discriminative ability. When used in the dual space, the objective changes to finding the minimal number of data points or Support vectors (SV) selected for representing the decision function.

Formally speaking, we assume the decision function is $\mathbf{w} \cdot \Phi(x) + b$, where Φ is a mapping function from the input space to the kernel/feature space. According to the Representer theory [12, 9], $\mathbf{w}$ can be further represented as the linear combination of all the mapped training samples, i.e, $\mathbf{w} = \sum_{i=1}^{l} \alpha_i \Phi(\mathbf{x}_i)$. In most kernel machines, e.g., SVM, only a portion of α_i's are non-zero. The samples $\mathbf{x}_i$'s associated with non-zero α_i's are called Support Vectors . The task in the dual space is to minimize the number of support vectors so as to make the final decision function as sparse as possible. The task can be formulated as follows:

$$\{\alpha^{(t)}, b\} = \arg\min_{\alpha,b} \alpha^T \Lambda^{(t-1)} \alpha + C \sum_{i=1}^{l} \xi_i, \quad (10)$$

$$\text{s.t.} \quad y_i(\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, l$$

$$\Lambda^{(t)} = diag(1/|\alpha_1^{(t-1)}|^2, \ldots, 1/|\alpha_l^{(t-1)}|^2).$$

It is easily verified that, by substituting $\mathbf{w}$ with $\sum_{i=1}^{l} \alpha_i \Phi(\mathbf{x}_i)$ into the optimizationthe, the above problem is still a Quadratic Programming problem with respect to α and b. More specifically, at each iteration, the problem is almost the same as the standard two-norm SVM. The only difference lies in the first term of (10). Two-norm SVM uses the two-norm as the regularization to control the structure risk. In comparison, such risk is avoided by exploiting the asymptotically true zero-norm, where it is still a two-norm formulation at each iteration. The sub problem at each iteration can be similarly solved by using the Sequential Minimum Optimization method [11], incurring very small time complexity and space complexity. In practice, zero-norm can even be combined with two-norm by replacing the objective function with $\{\alpha^{(t)}, b\} = \arg\min_{\alpha,b} \alpha^T H \alpha + C_0 \alpha^T \Lambda^{(t-1)} \alpha + C \sum_{i=1}^{l} \xi_i$, where $\alpha^T H \alpha$ ($\mathbf{H}$ is the matrix with $h_{ij} = y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$), represents the two-norm regularization term describing the maximum margin between two classes of data. C_0 is used to control the trade-off between the two-norm and the zero-norm. Since elaboration of the combination of zero-norm and other norms is beyond the scope of this paper, we leave this topic as future work.

3 Experiments

In this section, we compare the proposed zero-norm for feature selection with three other competitive models: (1) the AROM SVM [15], (2) FSV SVM [3], and (3) the standard SVM [5], on four machine learning benchmark data sets and two microarray gene sets. We employ the implementation in the matlab toolbox of Spider [6] for the comparison with the proposed algorithm. In addition, we also test the performance of the proposed zero-norm in the dual space for reducing the number of Support Vectors so as to reduce the test time. We first report the experimental setup and then present the detailed results.

3.1 Setup

Four data sets are used to evaluate the proposed zero-norm for feature selection, including Sonar, Breast, Colon, and Lymphoma. The first two are from the UCI machine learning repository [2], while the last two are microarray gene data sets. Table 1 describes the detailed information of these data sets.

Data set	Dimension	# Sample
Sonar	60	208
Breast	9	683
Colon	2000	62
Lymphoma	4026	96

Table 1. Data Description

We randomly partition these data sets into 80% as the training set and 20% as the test data. We first use these algorithms to select a given number of features. Then the standard two-norm SVM is adopted as the classifier to conduct training and testing over the selected features. The final reported results are the average over 10 runs. The parameters for each algorithm are all tuned on the training set using cross validation. In order to make our model (including the AROM and the FSV method) choose exactly the given number of features (say r features), we follow the setup in [15] and stop at the last iteration where $||\mathbf{w}||_0^0 \leq r$. We then choose the r largest features of $\mathbf{w}$. All the experiments are conducted in a PC with 4G RAM and a 3.00GHz CPU.

3.2 Results

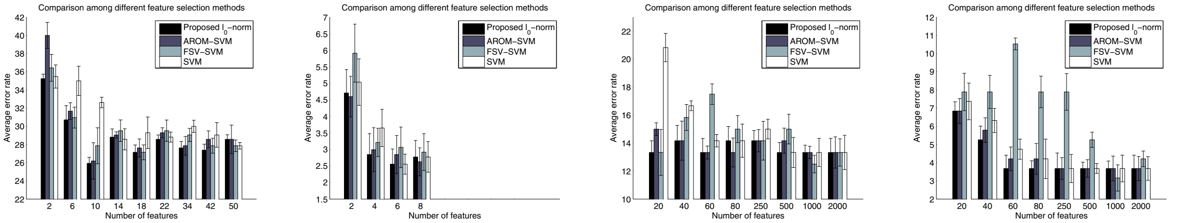
We compare our approach with the other methods in terms of the accuracy and the training speed. We also examine the performance of our methods used in the dual space, where the purpose is to generate a sparse solution in the kernel space.

3.2.1 Accuracy

We first evaluate the performance of the two-norm SVM on four data sets when a full set of features is used. The recognition error rates are respectively 27.86%, 2.63%, 13.33%, 4.21% for Sonar, Breast, Colon, and Lymphoma. We then

[5] The features with the largest weights are chosen when the standard SVM is used for feature selection.

[6] This toolbox can be downloaded from the web site http://www.kyb.tuebingen.mpg.de/bs/people/spider/index.html.



(a) Sonar data (b) Breast data (c) Colon gene data (d) Lymphoma gene data

Figure 1. Comparisons on four data sets among the proposed true zero-norm SVM, and the other state-of-the-art methods, i.e., AROM SVM, FSV SVM, and the modified SVM for feature selection

draw the corresponding error rates of different algorithms against different number of features in Figure 1. Several important points are highlighted as follows. First, the recognition error rates after feature selection are lower than or almost the same as those without feature selection (i.e., using the full set of features for training). This fact shows that feature selection is necessary before conducting learning on data. Second, when the given feature number is small, the "advanced" algorithms, i.e., our proposed zero-norm, the AROM SVM, and the FSV SVM perform much stable than the "naive" SVM in feature selection. The "naive" SVM sometimes demonstrates very higher error rates, e.g., in WDBC and Colon when the feature number is set to 2 and 20 respectively. As the feature number increases, the difference among all the algorithms becomes smaller. Finally and most importantly, although the proposed zero-norm algorithm does not win in all the cases against various number of features on four data sets, it clearly demonstrates an overall best performance. This can be readily observed from Figure 1: the error bars of the proposed true zero-norm are usually shorter than the ones of the other methods. This shows the advantages of our algorithm exploiting a direct optimization of zero-norm over those using a surrogate of zero-norm.

3.2.2 Computational Time

In order to evaluate the efficiency, we also report the average computational time in feature selection for all the four methods in the following. For brevity, we only report the running time when the feature number is set to 2 in the first four data sets and 20 in the remaining two microarray data sets. The results are described in Table 2.

Clearly, SVM demonstrates the fastest speed, since it uses the naive approach to select features. Our proposed method significantly outperforms the other two approximating approaches. Since our model asymptotically achieves the true zero-norm, its optimization directly hits the target of feature selection. In comparison, the AROM SVM and FSV SVM models minimize an approximation of the zero-norm, which might be wasteful in some sense. We observe that the AROM and the FSV approaches sometimes are stucked in selecting the features. For example, the FSV spends much time in choosing features in Breast data, while the AROM takes over 500 seconds in Bci data. Furthermore, we notice that both our approach and the AROM SVM spends much less time than the FSV SVM in Colon and Lymphoma, where the number of training samples is far fewer than the feature dimensionality. The reason is that both our proposed approach and the AROM SVM can take advantage of the dual optimization, while the FSV SVM cannot; this makes the FSV SVM scale with respect to the number of features rather than the number of samples. Hence it is much slower in such tasks.

3.2.3 Performance in the Dual Space

We also examine the performance of the proposed algorithm in the dual space, where the target is to make the final decision function as sparse as possible, i.e., the objective is to select as few as possible Support Vectors rather than choosing a compact set of features. As the test speed and the required space of many kernel methods is proportional to the number of SVs, achieving the true zero-norm immediately leads to a high test speed as well as a small space complexity.

We compare the proposed algorithm with the standard two-norm SVM, and the state-of-the-art model in attaining the sparsity, the Relevance Vector Machine (RVM) [13, 14]. We evaluate the algorithms on two large data sets Twonorm and Titanic from UCI. Twonorm consists of $7,400$ 20-dimensional samples and Titanic has $2,201$ 3-dimensional data points. We perform the evaluations using 10-fold cross validation for Titanic data and 5-fold cross validation for the relatively larger data set Twonorm. The kernel function used is the RBF kernel. Similarly, the trade-off parameter C and the width parameter σ in the RBF function are chosen via cross validation.

The experimental results including the test set accuracy (TSA) and the number of SVs are shown in Table 3. From

Data Set	Proposed Algorithm	AROM SVM	FSV SVM	SVM
Sonar	0.8061 ± 0.02	6.1431 ± 1.05	2.2888 ± 0.41	0.0146 ± 0.00
Breast	0.3203 ± 0.01	0.6247 ± 0.06	290.4822 ± 13.27	0.0461 ± 0.00
Colon	0.0223 ± 0.00	1.3558 ± 0.29	2.6941 ± 0.25	0.0018 ± 0.00
Lymphoma	0.1766 ± 0.01	2.3809 ± 0.21	23.640 ± 3.16	0.0057 ± 0.00

Table 2. Comparisons of computational time (seconds) among different feature selection algorithms

Data Set	Proposed Algorithm		SVM		RVM	
	TSA	#SVs	TSA	#SVs	TSA	#SVs
Twonorm	97.81	16.60	97.70	537.40	97.47	39.20
Titanic	78.82	256.70	78.86	1981.00	77.81	1768.92

Table 3. Comparisons in the dual space on Twonorm and Titanic

the table, it is clear that our proposed zero-norm algorithm can significantly reduce the number of SVs while maintaining the accuracy. When compared with RVM, we also observe a much smaller number of SVs in our proposed algorithm compared with RVM. These results clearly demonstrate the advantages of the proposed algorithm.

4 Conclusion

We have proposed a direct optimization of zero-norm for feature selection in this paper. This approach distinguishes itself from traditional methods that usually optimizes a surrogate of zero-norm. We have present detailed theoretical justifications and interpret the model based on a Bayesian viewpoint. We have demonstrated how the proposed algorithm is elegantly used for feature selection in the primal space and kernel minimization in the dual space. We have conducted a series of experiments to evaluate the proposed asymptotically true zero-norm. The experimental results on both biological microarray data and UCI data have demonstrated the advantages of the proposed zero-norm for feature selection in terms of both the accuracy and the efficiency. Moreover, experiments in the dual space have also shown very promising results.

Acknowledgements

The work described in this paper was fully supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4150/07E and No. CUHK4125/07).

References

[1] E. Amaldi and V. Kann. On the approximability of minimizing non zero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209:237–260, 1998.

[2] C. L. Blake and C. J. Merz. Repository of machine learning databases, University of California, Irvine, http://www.ics.uci.edu/∼mlearn/mlrepository.html, 1998.

[3] P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In *Proceedings of International Conference on Machine Learning (ICML-1998)*, pages 82–90, 1998.

[4] P. S. Bradley, O. L. Mangasarian, and W. N. Street. Approximating discrete probability distributions with dependence trees. *INFORMS Journal on Computing archive*, 10:209–217, February 1998.

[5] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[6] M. Figueiredo and A. K. Jain. Unsupervised selection and estimation of finite mixture models. In *Proceedings of the International Conference on Pattern Recognition (ICPR-2000)*, pages 335–338, 2000.

[7] M. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(3):381–396, 2002.

[8] G. M. Fung, O. L. Mangasarian, and A. J. Smola. Minimal kernel classifers. *Journal of Machine Learning Research*, 3:303–321, 2002.

[9] K. Huang, H. Yang, I. King, M. R. Lyu, and L. Chan. The minimum error minimax probability machine. *Journal of Machine Learning Research*, 5:1253–1286, 2004.

[10] M. I. Jordan and L. Xu. Convergence results for the em approach to mixtures of experts architectures. *Neural Networks*, 8(9):1409–1431, 1995.

[11] J. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. *Technical Report MSR-TR-98-14*, 1998.

[12] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[13] M. Tipping. The relevance vector machine. In *Advances in Neural Information Processing Systems 12 (NIPS)*, 2000.

[14] M. Tipping and A. C. Faul. Fast marginal likelihood maximization for sparse bayesian models. In *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics. Key West, FL*, 2003.

[15] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping. Use of the zero norm with linear models and kernel methods. *Journal of Machine Learning Research*, 3:1439–1461, 2003.