Learning Kernels from Indefinite Similarities

Yihua Chen YHCHEN@EE.WASHINGTON.EDU
Maya R. Gupta GUPTA@EE.WASHINGTON.EDU
Department of Electrical Engineering, University of Washington, Seattle, WA 98195, USA

Benjamin Recht BRECHT@IST.CALTECH.EDU
Center for the Mathematics of Information, California Institute of Technology, Pasadena, CA 91125, USA

Abstract

Similarity measures in many real applications generate indefinite similarity matrices. In this paper, we consider the problem of classification based on such indefinite similarities. These indefinite kernels can be problematic for standard kernel-based algorithms as the optimization problems become non-convex and the underlying theory is invalidated. In order to adapt kernel methods for similarity-based learning, we introduce a method that aims to simultaneously find a reproducing kernel Hilbert space based on the given similarities and train a classifier with good generalization in that space. The method is formulated as a convex optimization problem. We propose a simplified version that can reduce overfitting and whose associated convex conic program can be solved efficiently. We compare the proposed simplified version with six other methods on a collection of real data sets.

1. Introduction

Similarity-based learning assumes that only similarities between samples are given, and in the supervised case, also labels of the training samples (Chen et al., 2009). Similarity-based learning arises in computer vision, bioinformatics, information retrieval, natural language processing, and a broad range of other fields.

If the matrix formed by the similarities between samples is positive semidefinite (PSD), then the similarity matrix can be used as a kernel matrix in standard kernel methods. However, for many applications the underlying similarity function does not produce PSD similarity matrices. One can approximate the similarity matrix by a PSD matrix, but different approximations can yield very different results. For example, as shown in Table 1, support vector machine (SVM) classification on the Protein data set has 32% error if the indefinite similarity matrix is made PSD by adding a scaled identity matrix to it (shift), but 9% error if the indefinite similarity matrix is made PSD by setting all the negative eigenvalues to zero (clip). In this paper, we investigate learning a kernel given indefinite similarities that produces a classifier with good generalization.

For the similarity-based classification problem, we take as given an $n \times n$ indefinite matrix S of pairwise similarities between n training samples, and an $n \times 1$ vector y of their class labels. For a test sample x, we take as given an $n \times 1$ vector s of pairwise similarities between x and each of the n training samples, and also its self-similarity. We assume that the underlying similarity function $\psi(x, x')$ is only available in terms of S, s and $\psi(x, x)$; the case where one can directly compute $\psi(x, x')$ for any sample pair can be viewed as a special case.

An intuitive approach to similarity-based classification is the k-nearest neighbor method, where nearest-neighbors are determined by the given similarities. Also, a number of researchers have developed methods that use the similarities as features: the ith row of S is taken to be the feature vector for the ith training sample; s is the feature vector for the test sample. SVMs and other empirical risk minimization classifiers have been applied to these similarity features (Graepel et al., 1998; Liao & Noble, 2003; Hochreiter & Obermayer, 2006). Recently, generative classifiers have been developed for similarity-based learning that model the class-conditional distributions of similarities (Cazzanti & Gupta, 2007). A more com-

Appearing in *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009.

plete review of similarity-based classification can be found in Chen et al. (2009).

In this paper we focus on treating the similarity function as a kernel. We first review the prior art in adapting kernel methods for similarity-based classification in Section 2. Then, in Section 3 we consider methods to find an effective reproducing kernel Hilbert space (RKHS) for learning given indefinite similarities. Our experiments in Section 4 compare the proposed method with other similarity-based classifiers on six real data sets. We conclude in Section 5 with a discussion of extensions of this work.

2. Background and Related Work

To use kernel methods with indefinite similarities, one could simply replace the kernel matrix K with the similarity matrix S, and ignore the fact that S is indefinite. Since the indefinite matrix S does not correspond to an RKHS, one loses the underlying theoretical support for such kernel methods. In practice, the associated optimization problems may become nonconvex, for example, the SVM dual problem:

$$\begin{array}{ll} \underset{\alpha}{\text{maximize}} & \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T \operatorname{diag}(y) K \operatorname{diag}(y)\alpha \\ \text{subject to} & y^T\alpha = 0,\ 0 \le \alpha \le C\mathbf{1}, \end{array} \tag{1}$$

with variable $\alpha \in \mathbb{R}^n$, is no longer convex if one replaces K by S (here $\mathbf{1}$ is the column vector with all entries one, and $\le$ denotes component-wise inequality for vectors). Nevertheless, to solve the problem in (1), Lin & Lin (2003) show that with a simple modification, the sequential minimal optimization (SMO) algorithm will still converge, but to a stationary point, not necessarily a global maximum.[1] Ong et al. (2004) interpret this as finding a stationary point in a reproducing kernel Kreĭn space (RKKS) induced by S, and Haasdonk (2005) shows that this is equivalent to minimizing the distance between reduced convex hulls in a pseudo-Euclidean space induced by S.

To gain the full theoretical and practical benefits of kernel methods, in this paper we focus on approaches to finding a surrogate kernel matrix K derived from S. Previous work in this vein has considered different spectrum modifications to make S PSD, such as spectrum clip, flip and shift (Wu et al., 2005). Assume S is symmetric and thus has eigenvalue decomposition $S = U\Lambda U^T$, where U is an orthogonal matrix and Λ is a diagonal matrix of real eigenvalues, that is, $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Spectrum clip makes S PSD by clipping all the negative eigenvalues to zero:

$$S_{\text{clip}} = U \operatorname{diag}\left(\max(\lambda_1, 0), \ldots, \max(\lambda_n, 0)\right) U^T.$$

A mathematical justification for spectrum clip is that S_{clip} is the nearest PSD matrix to S in terms of the Frobenius norm (Higham, 1988):

$$S_{\text{clip}} = \arg\min_{K \succeq 0} \|K - S\|_F,$$

where $\succeq$ denotes the generalized inequality with respect to the PSD cone for square matrices.[2] Spectrum flip makes S PSD by flipping the sign of the negative eigenvalues:

$$S_{\text{flip}} = U \operatorname{diag}\left(|\lambda_1|, \ldots, |\lambda_n|\right) U^T,$$

which is equivalent to replacing the original eigenvalues of S with its singular values. Spectrum shift makes S PSD by shifting the whole spectrum by the minimum required amount:

$$S_{\text{shift}} = U\left(\Lambda + |\min(\lambda_{\min}(S), 0)|\, I\right) U^T,$$

where $\lambda_{\min}(S)$ is the minimum eigenvalue of S, and I is the identity matrix. Spectrum shift only enhances the self-similarities and does not change the similarity between any two different samples. Roth et al. (2003) show that S_{shift} preserves the group structure when used for clustering nonmetric proximity data.

Some research considers indefinite similarities to be noisy observations of an unknown PSD kernel. A recent paper took this perspective and formulated an extension of the SVM for indefinite kernels (Luss & d'Aspremont, 2007):

$$\begin{array}{ll} \underset{\alpha}{\text{maximize}} & \min\limits_{K \succeq 0} \left(g(\alpha, K) + \rho\|K - S\|_F^2\right) \\ \text{subject to} & y^T\alpha = 0,\ 0 \le \alpha \le C\mathbf{1}, \end{array} \tag{2}$$

where $g(\alpha, K) \triangleq \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T \operatorname{diag}(y) K \operatorname{diag}(y)\alpha$; the variables are $\alpha \in \mathbb{R}^n$ and $K \in \mathbb{R}^{n\times n}$, and $C > 0$ and $\rho > 0$ are two hyperparameters. The problem in (2) is a soft-penalty variant of maximizing the minimum of the objective function of (1) among the PSD matrices close to S. Luss & d'Aspremont (2007) interpret (2) as "a worst-case robust classification problem with bounded uncertainty on the kernel matrix." However, in this paper we offer a better interpretation of (2) by showing that (2) is in fact equivalent to the method we consider in Section 3.1. A fast algorithm to solve (2) has been proposed by Chen & Ye (2008).

[1] Since (1) is to maximize a continuous function on a compact set, the maximum can always be attained.

[2] For $K \in \mathbb{R}^{n\times n}$, $K \succeq 0$ means that K is PSD and thus implies that K is symmetric.

Lastly, we note that Lu et al. (2005) have proposed a multidimensional scaling technique that fits a kernel matrix to the given nonmetric dissimilarities, and the resulting embedding is used for clustering and visualization.

3. Methods for Learning Kernels

Given a similarity function $\psi(x, x')$, we would like to seek a surrogate kernel function $k(x, x')$ that induces an RKHS in which a classifier with good generalization can be learned. However, assuming one only has access to the values of the similarity function for all pairs of the training samples, we pose the problem as: Given an indefinite similarity matrix S, can we find a surrogate PSD matrix K corresponding to an RKHS in which a classifier with good generalization can be learned?

First, in Section 3.1 we consider the surrogate PSD matrix K to be a free parameter in the SVM primal. Then, in Section 3.3 we restrict the surrogate K to be a spectrum modification of S in order to reduce overfitting and yield a more tractable optimization problem.

3.1. Learning the Kernel Matrix

For the development of the investigated methods, we favor the primal form of the SVM due to its clear mathematical interpretation in terms of empirical risk minimization with regularization:

$$\begin{array}{ll} \underset{c,b,\xi}{\text{minimize}} & \frac{1}{n}\mathbf{1}^T\xi + \eta c^T K c \\ \text{subject to} & \text{diag}(y)(Kc + b\mathbf{1}) \geq \mathbf{1} - \xi,\ \xi \geq 0, \end{array} \tag{3}$$

with variables $c \in \mathbb{R}^n$, $b \in \mathbb{R}$ and $\xi \in \mathbb{R}^n$, and regularization parameter $\eta > 0$.

We consider minimizing the empirical risk with regularization simultaneously over the kernel matrix K and the original SVM variables. Specifically, we form:

$$\begin{array}{ll} \underset{c,b,\xi,K}{\text{minimize}} & \frac{1}{n}\mathbf{1}^T\xi + \eta c^T K c + \gamma\|K - S\|_F \\ \text{subject to} & \text{diag}(y)(Kc + b\mathbf{1}) \geq \mathbf{1} - \xi, \\ & \xi \geq 0,\ K \succeq 0, \end{array} \tag{4}$$

with additional variable $K \in \mathbb{R}^{n\times n}$ and additional regularization parameter $\gamma > 0$. The regularization term $\gamma\|K-S\|_F$ focuses the search for K in the vicinity of S in terms of the Frobenius norm. Recall that spectrum clip yields the nearest PSD matrix to S in terms of the Frobenius norm, thus when γ is set very large, (4) is almost the same as training an SVM with S_{clip}.

Although (4) and (2) are formed from different perspectives, we can show that they are equivalent except for a slight difference in the regularizer of K. Let $\mathcal{A} = \{\alpha \in \mathbb{R}^n \mid y^T\alpha = 0,\ 0 \leq \alpha \leq C\mathbf{1}\}$, and rewrite (2) as

$$\max_{\alpha\in\mathcal{A}} \min_{K\succeq 0}\ g(\alpha, K) + \rho\|K - S\|_F^2. \tag{5}$$

Because $\mathcal{A}$ and the PSD cone are both convex, and $\mathcal{A}$ is compact, and the objective function of (5) is continuous in α and K, concave in α and convex in K, by Sion's minimax theorem (Sion, 1958), we can switch the max and min, that is, (5) is equivalent to

$$\min_{K\succeq 0} \max_{\alpha\in\mathcal{A}}\ g(\alpha, K) + \rho\|K - S\|_F^2. \tag{6}$$

Since (1) is the dual of (3) with zero duality gap, (6) is equivalent to (4) except that they are slightly different in the regularizer of K. Hence a more accurate interpretation of (2) is that it finds the best-case K for classification rather than the worst-case, though we will keep calling (2) the "robust" SVM.

It is not trivial to solve (4) as formulated. We show that by using the following lemma, (4) can in fact be expressed as a convex conic program, which can be solved by a general-purpose convex conic optimizer such as SeDuMi (Strum, 1999) and SDPT3 (Tütüncü et al., 2003).

Lemma 1. *Let $K \in \mathbb{R}^{n\times n}$, $z \in \mathbb{R}^n$ and $u \in \mathbb{R}$. Then*

$$\begin{bmatrix} K & z \\ z^T & u \end{bmatrix} \succeq 0$$

if and only if $K \succeq 0$, z is in the range (column space) of K, and $u - z^T K^\dagger z \geq 0$, where $K^\dagger$ is the Moore-Penrose pseudoinverse of K.

This lemma follows directly from Horn & Zhang (2005, p. 44, Theorem 1.20), which states a basic property of the generalized Schur complement.

Let $z = Kc$, and notice that $c^T K c = z^T K^\dagger z$ because $KK^\dagger K = K$. By introducing slack variables u and v, and applying Lemma 1, we can express (4) as

$$\begin{array}{ll} \underset{z,b,\xi,K,u,v}{\text{minimize}} & \frac{1}{n}\mathbf{1}^T\xi + \eta u + \gamma v \\ \text{subject to} & \text{diag}(y)(z + b\mathbf{1}) \geq \mathbf{1} - \xi,\ \xi \geq 0, \\ & \begin{bmatrix} K & z \\ z^T & u \end{bmatrix} \succeq 0,\ \|K - S\|_F \leq v, \end{array} \tag{7}$$

with variables $z \in \mathbb{R}^n$, $b \in \mathbb{R}$, $\xi \in \mathbb{R}^n$, $K \in \mathbb{R}^{n\times n}$, $u \in \mathbb{R}$ and $v \in \mathbb{R}$. The problem in (7) is a convex conic program since it has a linear objective, a set of affine constraints, a linear matrix inequality (LMI) and a second-order cone (SOC) constraint. Let $z^\star$, $b^\star$ and $K^\star$ denote the optimal solution to (7); we can recover the optimal $c^\star$ by $c^\star = (K^\star)^\dagger z^\star$.

3.2. Consistent Treatment of Training and Test Samples

As stated before, we would like to find a surrogate kernel function $k(x, x')$ for the similarity function $\psi(x, x')$. However, only $K^\star$, the surrogate of S in the PSD cone, is learned from training. Given a test sample x, estimating its label using its unmodified similarities to the training samples s, that is,

$$\hat{y} = \operatorname{sgn}((c^\star)^T s + b^\star),$$

ignores the fact that $c^\star$ is trained on $K^\star$ not on S.

This problem has been addressed by Wu et al. (2005) for the case of spectrum modification. Given s and the self-similarity of the test sample $\psi(x, x)$, their approach is to recompute the same spectrum modification on the augmented $(n+1) \times (n+1)$ similarity matrix

$$S' = \begin{bmatrix} S & s \\ s^T & \psi(x,x) \end{bmatrix}$$

to form $\tilde{S}'$, and then let the modified test similarities $\tilde{s}$ be the first n elements of the last column of $\tilde{S}'$. The classifier trained on the modified training similarities $\tilde{S}$ is then applied on $\tilde{s}$. To implement this approach, they propose a fast algorithm to perform eigenvalue decomposition of S' by using the results of the eigenvalue decomposition of S.

Similarly, given s, $\psi(x, x)$, S and $K^\star$, we propose to find the appropriate $\tilde{s}$ by solving

$$\begin{array}{ll} \underset{k,r}{\text{minimize}} & \left\| \begin{bmatrix} K^\star & k \\ k^T & r \end{bmatrix} - \begin{bmatrix} S & s \\ s^T & \psi(x,x) \end{bmatrix} \right\|_F \\ \text{subject to} & \begin{bmatrix} K^\star & k \\ k^T & r \end{bmatrix} \succeq 0, \end{array} \tag{8}$$

with variables $k \in \mathbb{R}^n$ and $r \in \mathbb{R}$, in the hope that the optimal solution $k^\star$ is related to s in a way that is similar to how $K^\star$ is related to S. The test sample x is then classified as

$$\hat{y} = \operatorname{sgn}((c^\star)^T k^\star + b^\star). \tag{9}$$

By applying Lemma 1 with its condition that k be in the range of $K^\star$ expressed as (Boyd & Vandenberghe, 2004, Appendix A.5.5)

$$\left(I - K^\star (K^\star)^\dagger\right) k = 0,$$

we can reduce (8) to the following quadratically constrained quadratic program (QCQP):

$$\begin{array}{ll} \underset{k,r}{\text{minimize}} & 2\|k - s\|_2^2 + (r - \psi(x,x))^2 \\ \text{subject to} & k^T (K^\star)^\dagger k - r \le 0, \\ & \left(I - K^\star (K^\star)^\dagger\right) k = 0, \end{array}$$

which can be solved very efficiently.

3.3. Learning the Spectrum Modification

Although (7) is a convex optimization problem, the scale of the problem, as measured by the number of variables, grows quadratically with n. Moreover, the flexibility of (4) may lead to a model that overfits the data. Therefore, in this subsection, we propose a computationally simpler method that restricts K to be a spectrum modification of S, inspired by the fact that both spectrum clip and flip can be represented by a linear transformation on S, that is, $\tilde{S} = AS$, where $A = U \operatorname{diag}(a) U^T$ (recall $S = U \Lambda U^T$). For spectrum clip,

$$a_{\text{clip}} = \begin{bmatrix} I_{\{\lambda_1 \ge 0\}} & \dots & I_{\{\lambda_n \ge 0\}} \end{bmatrix}^T,$$

where $I_{\{\cdot\}}$ is the indicator function, and for spectrum flip,

$$a_{\text{flip}} = \begin{bmatrix} \operatorname{sgn}(\lambda_1) & \dots & \operatorname{sgn}(\lambda_n) \end{bmatrix}^T.$$

We propose to treat a as a variable such that the surrogate kernel matrix K_a can be written as a linear transformation of S, that is,

$$K_a = AS = U \operatorname{diag}(a) U^T S = U \operatorname{diag}(a) \Lambda U^T,$$

and we formulate the problem as

$$\begin{array}{ll} \underset{c,b,\xi,a}{\text{minimize}} & \frac{1}{n} \mathbf{1}^T \xi + \eta c^T K_a c + \gamma h(a) \\ \text{subject to} & \operatorname{diag}(y) \left(K_a c + b\mathbf{1}\right) \ge \mathbf{1} - \xi, \\ & \xi \ge 0,\ \Lambda a \ge 0, \end{array} \tag{10}$$

where $h(a)$ is a convex regularizer of a. As in Section 3.1, let $z = \operatorname{diag}(a) \Lambda U^T c$. We note that the LMI constraint on z can be simplified to

$$z_i^2 \le \lambda_i a_i u_i,\ i = 1, \dots, n, \tag{11}$$

where $u_i \ge 0$, $i = 1, \dots, n$, are slack variables. As (11) can be expressed as SOC constraints (Boyd & Vandenberghe, 2004, p. 197), we can write (10) as

$$\begin{array}{ll} \underset{z,b,\xi,a,u,v}{\text{minimize}} & \frac{1}{n} \mathbf{1}^T \xi + \eta \mathbf{1}^T u + \gamma v \\ \text{subject to} & \operatorname{diag}(y)(Uz + b\mathbf{1}) \ge \mathbf{1} - \xi, \\ & \xi \ge 0,\ \Lambda a \ge 0,\ u \ge 0,\ h(a) \le v, \\ & \left\| \begin{bmatrix} 2z_i \\ \lambda_i a_i - u_i \end{bmatrix} \right\|_2 \le \lambda_i a_i + u_i,\ i = 1, \dots, n, \end{array} \tag{12}$$

with variables $z \in \mathbb{R}^n$, $b \in \mathbb{R}$, $\xi \in \mathbb{R}^n$, $a \in \mathbb{R}^n$, $u \in \mathbb{R}^n$ and $v \in \mathbb{R}$. Since here we only learn the spectrum modification, the number of variables grows linearly with n. If one chooses $h(a)$ to make $h(a) \le v$ an SOC constraint, then (12) is a second-order cone program (SOCP), and can be efficiently solved by algorithms

such as the primal-dual interior-point method (Andersen et al., 2003).

Let $z^\star$, $b^\star$ and $a^\star$ denote the optimal solution to (12). Then, one can recover the optimal $c^\star$ by

$$c^\star = U \left(\operatorname{diag}(a^\star)\Lambda\right)^\dagger z^\star.$$

Solving (10) produces a linear transformation

$$A = U \operatorname{diag}(a^\star) U^T$$

on S. For a test sample x, we propose to apply the same linear transformation A on s such that in (9) we use $k^\star = As$. This method for modifying the test similarities is low-cost and is also *consistent* in the sense that if any training sample is taken as a test sample, its similarities will be modified in the same way during training and during test, in line with the spirit of empirical risk minimization.

3.4. Regularizer Selection

In (4), we regularize the search for K toward S. Since $\|K_a - S\|_F = \|\Lambda(a - \mathbf{1})\|_2$, $\|\Lambda(a - \mathbf{1})\|_2$ could be a reasonable option for $h(a)$ in (10). As described before, when the regularization parameter γ is set large, this is very close to training an SVM with S_{clip}, and we expect that using $h(a) = \|a - a_{\text{clip}}\|_2$ will achieve similar results, which we have verified experimentally.

In fact, instead of searching in the vicinity of S, one might want to regularize the search for K_a toward other approximations of S. For example, one can use other regularizers such as $h(a) = \|a - a_{\text{flip}}\|_2$.

We suggest that in practice one should select the regularizer by cross-validation. Specifically, we propose to select the regularizer from three choices based on the cross-validation error of the SVMs using spectrum clip and flip. If the SVM using spectrum clip has lower cross-validation error, we use $h(a) = \|a - a_{\text{clip}}\|_2$; if the SVM using spectrum flip has lower cross-validation error, we use $h(a) = \|a - a_{\text{flip}}\|_2$; if they have equal cross-validation error, we use $h(a) = \|a - a_{\text{clip}}\|_2 + \|a - a_{\text{flip}}\|_2$.

4. Experiments

We compare the SVM proposed in Section 3.3 (for learning the spectrum modification) using the regularizer selection procedure detailed in Section 3.4 with six other algorithms: k-NN on the similarities, the SVMs using the similarities as a kernel via spectrum clip, flip, and shift, a linear SVM acting on the similarities as features, and the robust SVM given in (2), which we have shown is equivalent to the method we consider in Section 3.1 that learns the full kernel matrix.

4.1. Data Sets

We ran the experiments on six real data sets[3] representing a diverse set of indefinite similarities. Figure 1 shows the similarity matrices for all the samples for each data set. The spectra of these similarity matrices are shown in Figure 2.

The *Amazon* data set, created for this paper, consists of 96 fiction and nonfiction books, by 23 different authors, including some authors who write both fiction and nonfiction. The problem is to correctly classify each book as one of the 36 nonfiction books or one of the 60 fiction books based on its similarities to the training books. The similarity between book A and book B is $\frac{1}{2}\left(P(A,B) + P(B,A)\right)$, where $P(A,B)$ is the percentage of customers who bought book A after viewing book B, as reported by `amazon.com`. The similarity matrix is very sparse and has integer similarities between 0 and 100.

The *Aural Sonar* data set was developed to investigate the human ability to distinguish different types of sonar signals by ear (Philips et al., 2006), and consists of 100 samples. Each pairwise similarity is the sum of the similarity scores of two human subjects for that pair. The problem is to classify the 50 target-of-interest signals from the 50 clutter signals.

The *Protein* data set has sequence-alignment similarities for 226 proteins from 9 classes (Hoffmann & Buhmann, 1997). Here, we treat the problem as classifying the two most confusable classes, each of which has 72 samples.

The *Voting* data set comes from the UCI Repository (Asuncion & Newman, 2007). It is a binary classification problem with 435 samples, where each sample is a categorical feature vector with 16 components and three possibilities for each component. We compute the value difference metric (Stanfill & Waltz, 1986) from the categorical data, which is a dissimilarity that uses the training class labels to weight different components differently so as to achieve maximum probability of class separation. We normalize the dissimilarities such that $d(x, x') \in [0, 1]$ and convert them to similarities by letting $\psi(x, x') = 1 - d(x, x')$. Though the magnitude of the negative eigenvalues of its similarity matrix is very small as seen in Figure 2, those negative eigenvalues do cause different spectrum modifications to perform differently.

For the *Yeast-5-7* and *Yeast-5-12* data sets (Lanckriet et al., 2004), the problem is to predict the func-

[3] These data sets are available at `http://idl.ee.washington.edu/similaritylearning/`.

tions of yeast proteins. The original Yeast data set contains 3588 samples and each sample is a yeast protein sequence. There are 13 classes and some samples belong to more than one class due to their multiple roles. To simplify the problem, we choose a subset of 200 samples, called Yeast-5-7, by selecting the first 100 samples that exclusively belong to class 5 and the first 100 samples that exclusively belong to class 7. We select another subset called Yeast-5-12 by repeating the same procedure on class 5 and class 12. The Smith-Waterman E-value is used here to measure the similarity between two protein sequences.

4.2. Experimental Setup

We normalized the entries of all the similarity matrices to the range of $[0, 1]$. For each data set, we randomly partitioned the data 20 times into 20% test and 80% training. For each of the 20 partitions, we selected parameters by a 10-fold cross-validation on the training set. The regularization parameters η and γ for the proposed method, the hyperparameter C for the traditional C-SVM, the regularization parameter ρ for the robust SVM, and the neighborhood size k for k-NN were cross-validated from the following sets:

$$\begin{aligned} \eta &\in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}, \\ \gamma, C, \rho &\in \{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}, \\ k &\in \{1, 2, 3, \ldots, 16, 32\}. \end{aligned}$$

All the convex optimization problems for the proposed SVMs were solved by the semidefinite-quadratic-linear program solver SDPT3 (Tütüncü et al., 2003).

4.3. Results

The test errors averaged over the 20 randomized test/training partitions along with the standard deviations (in parentheses) are shown in Table 1. For each data set, the bold results denote the classifier with the lowest average error and those not statistically significantly worse according to a one-sided Wilcoxon signed-rank test at a significance level of 5%. One can see that the proposed SVM with spectrum modification learning is among the top performers on five out of the six data sets. The performance of the robust SVM, which learns the full kernel matrix, is very close to the proposed SVM except on the Amazon and Protein data sets. The robust SVM preforms poorly on the Protein data set, but on the Amazon data set, it is obviously a winner. We conjecture this is because the rank-one update given by (4) in Luss & d'Aspremont (2007) makes the kernel matrix of the Amazon data set less sparse and helps infer some hidden relationships between samples. The results indicate that on some data sets, the proposed SVM with spectrum modification learning can achieve statistically significant improvement over the SVMs with simple spectrum modification such as clip, flip and shift.

Our experiments also verify the necessity of treating training and test similarities consistently.[4] For example, on the Protein data set, if unmodified s is used, the errors of the proposed SVM, the SVM with spectrum flip and the robust SVM would increase from 2.93% to 48.10%, 4.14% to 41.38%, and 18.28% to 38.62%, respectively.

5. Discussion and Conclusions

For learning from indefinite similarities, we framed the problem as finding a surrogate RKHS, and investigated two methods to simultaneously learn the kernel matrix and minimize the empirical risk with regularization. Experimental evidence suggests that learning a spectrum modification provides an effective trade-off between increased model flexibility and overfitting. We consider it worthwhile to investigate other forms of regularizers for learning the kernel matrix from indefinite similarities.

We showed that these kernel learning ideas can be formulated as convex optimization problems. In fact, there exist general algorithms to solve (12) efficiently.

For supervised learning, the test similarities may have to be modified in a second step as shown in Section 3.2, but these kernel-learning methods would not need this second step if applied to transductive or local SVMs because the new regularization term can be used to solve for an augmented kernel matrix that includes the test sample(s). The focus here was on SVMs, but we hypothesize that this research might be useful for other kernel methods.

Acknowledgments

This work was funded by the Office of Naval Research.

References

Andersen, E. D., Roos, C., & Terlaky, T. (2003). On implementing a primal-dual interior-point method for conic quadratic optimization. *Math. Programming*, *95*, 249–277.

Asuncion, A., & Newman, D. (2007). UCI machine learning repository.

[4] This includes the robust SVM. The code provided by Luss & d'Aspremont (2007) uses a heuristic based on spectrum clip to achieve the consistency.

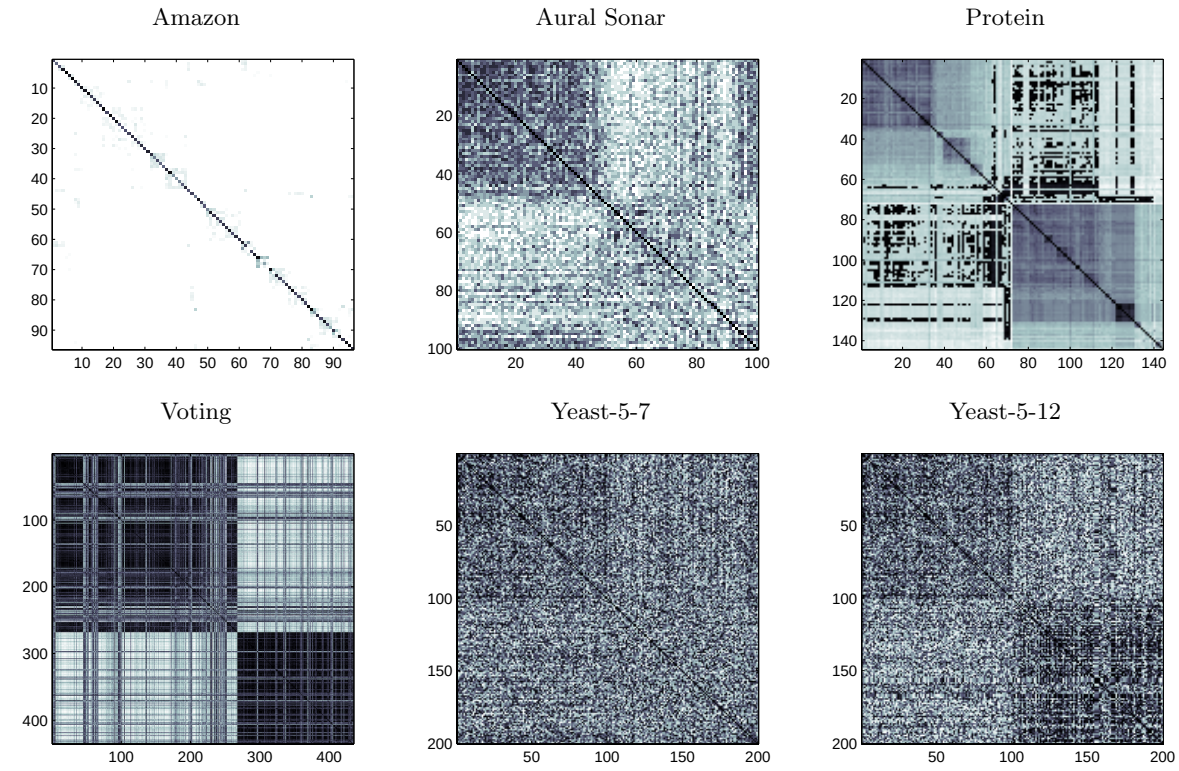

Figure 1. Similarity matrices of the six data sets; black corresponds to maximum similarity and white to zero.

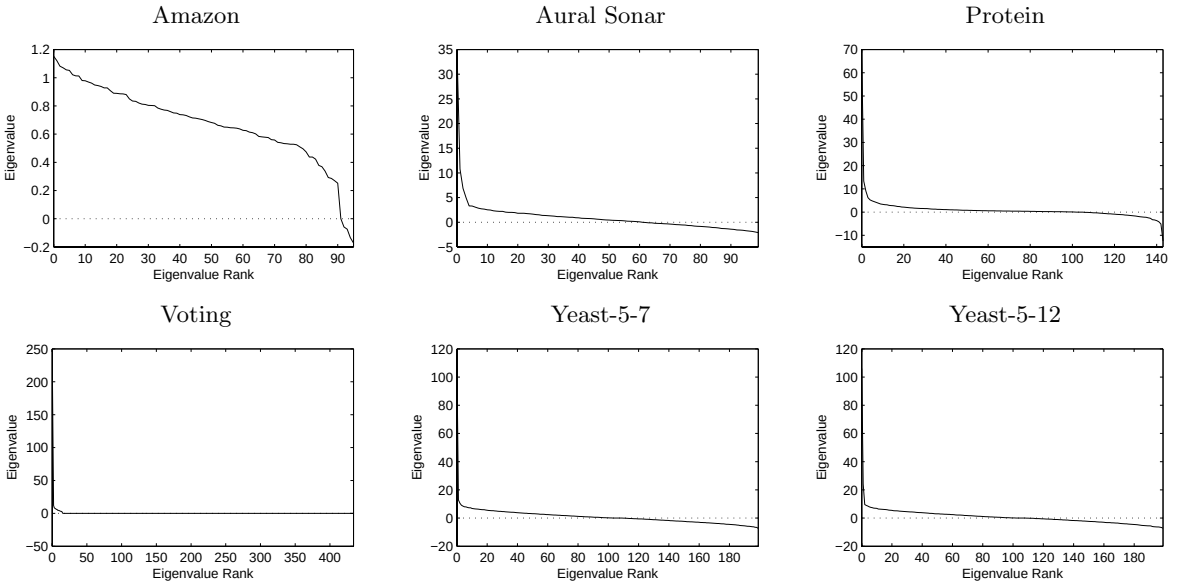

Figure 2. Eigenvalue spectra of the similarity matrices shown in Figure 1.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Cazzanti, L., & Gupta, M. R. (2007). Local similarity discriminant analysis. *Proc. Intl. Conf. Mach. Learning*.

Chen, J., & Ye, J. (2008). Training SVM with indefinite kernels. *Proc. Intl. Conf. Mach. Learning*.

Table 1. Mean and standard deviation (in parentheses) of the test errors (in percentage) across the 20 test/training partitions. For each data set, the lowest mean error and those not statistically significantly worse are boldfaced. The proposed SVM is for learning the spectrum modification as detailed in Section 3.3 and Section 3.4.

	Amazon		Aural Sonar		Protein		Voting		Yeast-5-7		Yeast-5-12	
Proposed SVM	11.32	(7.86)	**12.00**	(4.85)	**2.93**	(2.73)	**4.72**	(1.96)	**25.25**	(4.93)	**8.63**	(5.09)
Robust SVM	**7.63**	(6.55)	**12.00**	(5.79)	18.28	(5.57)	**5.00**	(1.79)	**26.50**	(4.50)	**9.00**	(5.39)
SVM w/ clip	12.37	(7.68)	**12.00**	(5.34)	9.14	(4.26)	**5.00**	(1.79)	28.25	(6.38)	10.50	(4.72)
SVM w/ flip	20.79	(10.97)	13.25	(5.07)	4.14	(3.01)	**4.83**	(2.17)	35.75	(7.75)	10.75	(5.13)
SVM w/ shift	15.53	(13.05)	17.25	(9.93)	31.55	(17.63)	6.55	(5.34)	46.00	(9.40)	43.38	(8.41)
Linear SVM	16.05	(11.59)	14.25	(6.94)	**2.59**	(3.06)	5.34	(1.93)	**26.88**	(6.22)	10.75	(4.82)
k-NN	9.47	(6.57)	18.25	(5.97)	43.45	(5.71)	5.46	(1.74)	30.63	(5.80)	12.75	(4.39)

Chen, Y., Garcia, E. K., Gupta, M. R., Rahimi, A., & Cazzanti, L. (2009). Similarity-based classification: Concepts and algorithms. *J. Mach. Learning Res.*, *10*, 747–776.

Graepel, T., Herbrich, R., Bollmann-Sdorra, P., & Obermayer, K. (1998). Classification on pairwise proximity data. *Advances in Neural Information Processing Systems*.

Haasdonk, B. (2005). Feature space interpretation of SVMs with indefinite kernels. *IEEE Trans. Pattern Anal. and Mach. Intel.*, *27*, 482–492.

Higham, N. J. (1988). Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, *103*, 103–118.

Hochreiter, S., & Obermayer, K. (2006). Support vector machines for dyadic data. *Neural Computation*, *18*, 1472–1510.

Hoffmann, T., & Buhmann, J. M. (1997). Pairwise data clustering by deterministic annealing. *IEEE Trans. Pattern Anal. and Mach. Intel.*, *19*, 1–14.

Horn, R. A., & Zhang, F. (2005). Basic properties of the Schur complement. In *The Schur complement and its applications*. Springer.

Lanckriet, G. R. G., Deng, M., Cristianini, N., Jordan, M. I., & Noble, W. S. (2004). Kernel-based data fusion and its application to protein function prediction in yeast. *Proc. Pacific Symposium Biocomputing*.

Liao, L., & Noble, W. S. (2003). Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships. *J. Computational Biology*, *10*, 857–868.

Lin, H.-T., & Lin, C.-J. (2003). *A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods* (Technical Report). National Taiwan University.

Lu, F., Keles, S., Wright, S. J., & Wahba, G. (2005). Framework for kernel regularization with application to protein clustering. *Proc. Natl. Acad. Sci. USA*, *102*, 12332–12337.

Luss, R., & d'Aspremont, A. (2007). Support vector machine classification with indefinite kernels. *Advances in Neural Information Processing Systems*.

Ong, C. S., Mary, X., Canu, S., & Smola, A. J. (2004). Learning with non-positive kernels. *Proc. Intl. Conf. Mach. Learning*.

Philips, S., Pitton, J., & Atlas, L. (2006). Perceptual feature identification for active sonar echoes. *Proc. IEEE OCEANS Conf.*

Roth, V., Laub, J., Kawanabe, M., & Buhmann, J. M. (2003). Optimal cluster preserving embedding of nonmetric proximity data. *IEEE Trans. Pattern Anal. and Mach. Intel.*, *25*, 1540–1551.

Sion, M. (1958). On general minimax theorems. *Pacific J. Math.*, *8*, 171–176.

Stanfill, C., & Waltz, D. (1986). Toward memory-based reasoning. *Comm. ACM*, *29*, 1213–1228.

Strum, J. F. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, *11*, 625–653.

Tütüncü, R. H., Toh, K. C., & Todd, M. J. (2003). Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Programming*, *95*, 189–217.

Wu, G., Chang, E. Y., & Zhang, Z. (2005). *An analysis of transformation on non-positive semidefinite similarity matrix for kernel machines* (Technical Report). University of California, Santa Barbara.