Active Kernel Learning

Steven C.H. Hoi CHHOI@NTU.EDU.SG
School of Computer Engineering, Nanyang Technological University, Singapore
Rong Jin RONGJIN@CSE.MSU.EDU
Department of Computer Science and Engineering, Michigan State University

Abstract

Identifying the appropriate kernel function/matrix for a given dataset is essential to all kernel-based learning techniques. A number of kernel learning algorithms have been proposed to learn kernel functions or matrices from side information (e.g., either labeled examples or pairwise constraints). However, most previous studies are limited to "passive" kernel learning in which side information is provided beforehand. In this paper we present a framework of *Active Kernel Learning* (**AKL**) that actively identifies the most informative pairwise constraints for kernel learning. The key challenge of active kernel learning is how to measure the informativeness of an example pair given its class label is unknown. To this end, we propose a **min-max** approach for active kernel learning that selects the example pair that results in a large classification margin regardless of its assigned class label. We furthermore approximate the related optimization problem into a convex programming problem. We evaluate the effectiveness of the proposed algorithm by comparing it to two other implementations of active kernel learning. Empirical study with nine datasets on semi-supervised data clustering shows that the proposed algorithm is more effective than its competitors.

1. Introduction

Kernel methods have attracted more and more attention of researchers in computer science and engineering due to their superior performance in data clustering, classification, and dimensionality reduction (Scholkopf & Smola, 2002; Vapnik, 1998). Kernel methods have been applied to many fields, such as data mining, pattern recognition, information retrieval, computer vision, and bioinformatics, etc. Since the choice of kernel functions or matrices is often critical to the performance of many kernel-based learning techniques, it becomes a more and more important research problem for how to automatically learn a kernel function/matrix for a given dataset. Recently, a number of kernel learning algorithms (Chapelle et al., 2003; Cristianini et al., 2002; Hoi et al., 2007; Kondor & Lafferty, 2002; Kulis et al., 2006; Lanckriet et al., 2004; Zhu et al., 2005) have been proposed to learn kernel functions or matrices from side information. The side information can be provided in two different forms: either labeled examples or pairwise constraints. In the latter case, two types of pairwise constraints are examined in the previous studies: a must-link pair where two examples should belong to the same class, and a cannot-link pair where two examples should belong to different classes. In this study, we focus on kernel learning with pairwise constraints.

Most kernel learning methods, termed as "passive kernel learning", assume that labeled data is provided beforehand. However, given the labeled data may be expensive to acquire, it is more cost effective if we are able to identify the most informative example pairs such that the kernel can be learned efficiently with only a small number of pairwise constraints. To this end, we focus on **active kernel learning** (AKL) whose goal is to identify the example pairs that are informative to the target kernels. We extends our previous work on non-parametric kernel learning (Hoi et al., 2007) to active kernel learning. As shown in (Hoi et al., 2007), the parametric approaches for kernel learning are often limited by their capacity in fitting diverse patterns of real-world data, and therefore are not as effective as the non-parametric approach for kernel learning.

The simplest approach toward active kernel learning is to measure the informativeness of an example pair by its kernel similarity. Given a pair of examples $(\mathbf{x}_i, \mathbf{x}_j)$, we assume that $K_{i,j}$, the kernel similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, is a large positive number when $\mathbf{x}_i$ and $\mathbf{x}_j$ are in the same class, and a large negative number

Appearing in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008. Copyright 2008 by the author(s)/owner(s).

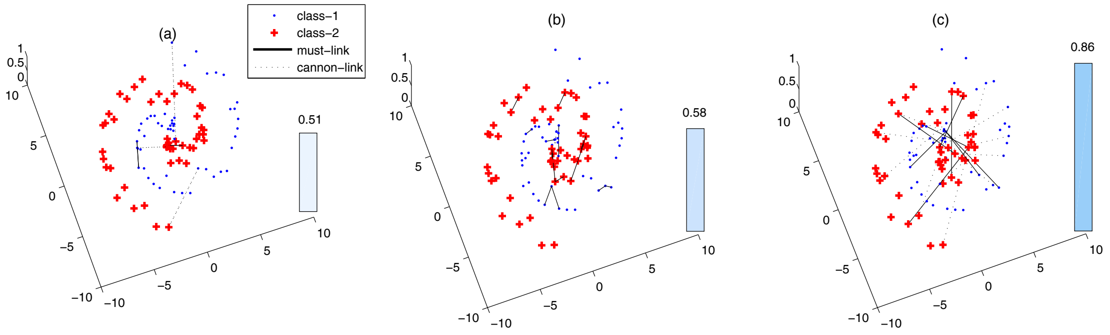

Figure 1. Examples of active kernel learning: (a) double-spiral artificial data with some given pairwise constraints, (b) AKL with the least $|K_{i,j}|$, (c) the proposed AKL method. The right bars show the resulting clustering accuracies using kernel k-means clustering methods.

when they are in different classes. Thus, by following the uncertainty principle of active learning (Tong & Koller, 2000; Hoi et al., 2006), the most informative example pairs should be the ones whose kernel similarities are closest to zero. In other words, the criterion is to select the example pair with the least absolute kernel similarity (i.e., $|K_{i,j}|$). Unfortunately, this simple approach may not always be effective in obtaining informative pairwise constraints for kernel learning. Figure 1 illustrates an example of active kernel learning for data clustering. In this example, Figure 1(a) shows an artificial dataset of two classes together with a few pairwise constraints. Figure 1(b) shows the pairwise constraints with the least $|K_{i,j}|$. We observe that most of them are must-link pairs with two data points separated by a modest distance. Since must-link constraints are not informative to the clustering boundary, a relatively small improvement is observed in clustering accuracy (from 51% to 58%) when using the kernel learned by this simple approach. In contrast, as shown in Figure 1(c), the proposed approach for active kernel learning is able to identify a pool of diverse pairwise constraints, including both must-links and cannot-links. The clustering accuracy is increased significantly, from 51% to 86%, by using the proposed active kernel learning.

The rest of this paper is organized as follows. Section 2 presents the min-max framework for our active kernel learning method, in which the problem is formulated into a convex optimization problem. Section 3 describes the results of the experimental evaluation. Section 4 concludes this work.

2. Active Kernel Learning

Our work extends the previous work on non-parametric kernel learning (Hoi et al., 2007) by introducing the component of actively identifying the example pairs that are most informative to the target kernel. In this section, we will first briefly review the non-parametric approach for kernel learning in (Hoi et al., 2007), followed by the description of the min-max framework for active kernel learning.

2.1. Non-parametric Kernel Learning

Let the entire data collection be denoted by $\mathcal{U} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ where each data point $\mathbf{x}_i \in \mathbb{R}^d$ is a vector of d elements. Let $S \in \mathbb{R}^{N\times N}$ be a symmetric matrix where each $S_{i,j} \geq 0$ represents the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. Unlike the kernel similarity matrix, S does not have to be positive semi-definite. For the convenience of presentation, we set $S_{i,i} = 0$ for all the examples. Then, according to (Hoi et al., 2007), a normalized graph Laplacian L is constructed using the similarity matrix S as follows:

$$L \quad = \quad (1+\delta)I - D^{-1/2}SD^{-1/2}$$

where $D = \mathrm{diag}(d_1, d_2, \ldots, d_N)$ is a diagonal matrix with $d_i = \sum_{j=1}^{N} f(\mathbf{x}_i, \mathbf{x}_j)$. A small $\delta > 0$ is introduced to prevent L from being singular. Let's denote by $\mathcal{T}$ the set of pairwise constraints. We construct a matrix $T \in \mathbb{R}^{N\times N}$ to represent the pairwise constraints in $\mathcal{T}$, i.e.,

$$T_{i,j} = \begin{cases} +1 & (\mathbf{x}_i, \mathbf{x}_j) \text{ is a must-link pair in } \mathcal{T} \\ -1 & (\mathbf{x}_i, \mathbf{x}_j) \text{ is a cannot-link pair in } \mathcal{T} \\ 0 & \text{otherwise} \end{cases}$$

Given the similarity matrix S and the pairwise constraints in $\mathcal{T}$, the goal of kernel learning is to identify a kernel matrix $Z \in \mathbb{R}^{N\times N}$ that is consistent with both $\mathcal{T}$ and S. Following (Hoi et al., 2007), we formulate it into the following convex optimization problem:

$$\begin{aligned} \arg\min_{Z,\varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 \qquad (1) \\ \text{s. t.} \quad & \forall (i,j)\in\mathcal{T},\ Z_{i,j}T_{i,j} \geq 1-\varepsilon_{i,j},\ \varepsilon_{i,j}\geq 0 \\ & Z \succeq 0 \end{aligned}$$

The first term in the above objective function plays a similar role as the manifold regularization (Belkin & andd P. Niyogi, 2004), where the graph Laplacian is used to regularize the classification results. The second term in the above measures the inconsistency between the learned kernel matrix Z and the given pairwise constraints. Note that unlike the formulation in (Hoi et al., 2007), we change $\varepsilon_{i,j}$ in the loss function to $\varepsilon_{i,j}^2$. This modification is specifically designed for active kernel learning, and the reason will be clear later. It is not difficult to see that the problem in (1) is a semi-definite programming problem, and therefore can be solved by the standard software package, such as SeDuMi (Sturm, 1999).

2.2. Min-max Framework for Active Kernel Learning

The simplest approach toward active kernel learning is to follow the uncertainty principle of active learning, and to select the example pair $(\mathbf{x}_i, \mathbf{x}_j)$ with the least $|Z_{i,j}|$ [1]. However, as already discussed in the introduction section, the key problem with this simple approach is that the example pairs with the least $|Z_{i,j}|$ may not necessarily be the the most informative ones, and therefore may not result in an efficient learning of the kernel matrix. To address this problem, we propose a min-max framework for active kernel learning that measures the informativeness of an example pair by how significantly the selected example pair will affect the target kernel matrix.

Consider an unlabeled example pair $(\mathbf{x}_k, \mathbf{x}_l) \notin \mathcal{T}$. To measure how this example will affect the kernel matrix, we consider the kernel learning problem with the additional example pair $(\mathbf{x}_k, \mathbf{x}_l)$ labeled by $y \in \{-1,+1\}$, i.e.,

$$\begin{aligned} \min_{Z,\varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 + \frac{c}{2}\varepsilon_{k,l}^2 \qquad (2) \\ \text{s. t.} \quad & T_{i,j}Z_{i,j} \geq 1-\varepsilon_{i,j},\ \forall (i,j)\in\mathcal{T} \\ & yZ_{k,l} \geq 1-\varepsilon_{k,l},\ Z\succeq 0 \end{aligned}$$

Let us denote by $\omega((k,l),y)$ the optimal value of the above optimization problem. Intuitively, $\omega((k,l),y)$ measures the overall classification accuracy with the additional example pair $(\mathbf{x}_k, \mathbf{x}_l)$ labeled by y. To further measure the informativeness of example pair $(\mathbf{x}_k, \mathbf{x}_l)$, we introduce the quantity $\kappa(k,l)$ as follows

$$\kappa(k,l) = \max_{y\in\{-1,+1\}} \omega((k,l),y) \qquad (3)$$

Clearly, $\kappa(k,l)$ measures the worst classification error with the addition of example pair $(\mathbf{x}_k, \mathbf{x}_l)$. Overall, $\kappa(k,l)$ measures how the example pair $(\mathbf{x}_k, \mathbf{x}_l)$ will affect the overall objective function, which indirectly measures the impact of the example pair on the target kernel matrix. To see this, consider an example pair $(\mathbf{x}_k, \mathbf{x}_l)$ that is highly consistent with the current kernel Z with label y (i.e., $Z_{k,l}y \geq 1$). According to the definition $\kappa(k,l)$, we would expect a large $\kappa(k,l)$ for pair $(\mathbf{x}_k, \mathbf{x}_l)$. This is because by assigning a label $-y$ to example pair $(\mathbf{x}_k, \mathbf{x}_l)$, we expect a large classification error and therefore large $\kappa(k,l)$. Hence, we use $\kappa(k,l)$ to measure the *uninformativeness* of example pairs, i.e., the smaller $\kappa(k,l)$, the less informative the example pair is. Therefore, the most informative example pair is found by minimizing $\kappa(k,l)$, i.e.,

$$(k,l)^* \;=\; \arg\min_{(k,l)\notin\mathcal{T}} \max_{t\in\{-1,+1\}} \omega((k,l),t) \qquad (4)$$

Directly solving the min-max optimization problem in (4) is challenging because function $\omega((k,l),t)$ is defined implicitly by the optimization problem in (2). The following theorem allows us to significantly simplify the optimization problem in (4)

Theorem 1. *The optimization problem in (4) is equivalent to the following optimization problem*

$$\begin{aligned} \min_{Z,\varepsilon,(k,l)\notin\mathcal{T}} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 + \frac{c}{2}\varepsilon_{k,l}^2 \qquad (5) \\ s.\ t. \quad & T_{i,j}Z_{i,j} \geq 1-\varepsilon_{i,j}, \varepsilon_{i,j}\geq 0,\ \forall (i,j)\in\mathcal{T} \\ & \varepsilon_{k,l} \geq 1 + |Z_{k,l}|,\ Z\succeq 0 \end{aligned}$$

Proof. The above theorem follows the fact that the solution $y^* \in \{-1,+1\}$ maximizing $\omega((k,l),y)$ is $y^* = -\mathrm{sign}(Z_{k,l})$. This fact allows us to remove the maximization within (4) and obtain the result in the theorem. □

The following corollary shows that the approach of selecting the example pair with the least $|Z_{k,l}|$ indeed corresponds to a special solution for the problem in (5).

Corollary 2. *The optimal solution to (5) with fixed kernel matrix Z is the example pair with the least $|Z_{k,l}|$, i.e.,*

$$(k,l)^* = \arg\min_{(k,l)\notin\mathcal{T}} |Z_{k,l}|$$

[1] Here we assume that $Z_{i,j} > 0$ when $\mathbf{x}_i$ and $\mathbf{x}_j$ are likely to share the same class, and $Z_{i,j} < 0$ when $\mathbf{x}_i$ and $\mathbf{x}_j$ are likely to be assigned to different classes

Proof. By fixing Z, the problem in (5) is simplified as

$$\min_{(k,l)\notin\mathcal{T}} \quad \varepsilon_{k,l}^2 \quad \text{s. t.} \quad \varepsilon_{k,l} \geq 1 + |Z_{k,l}|$$

It is easy to see that the optimal solution to the above problem is the example pair with the least $|Z_{k,l}|$. □

Note that a similar observation is described in the study (Chen & Jin, 2007) for standard active learning.

2.3. Algorithm

The straightforward approach toward the optimization problem in (5) is to try out every example pair $(\mathbf{x}_k, \mathbf{x}_l) \notin \mathcal{T}$. Evidently, this approach will not scale well when the number of example pairs is large.

Our first attempt toward solving the problem (5) is to turn it into a continuous optimization problem. To this purpose, we introduce variable $p_{k,l} \geq 0$ to represent the probability of selecting the example pair $(k,l) \notin \mathcal{T}$. Using this notation, we have the optimization problem in (5) rewritten as

$$\begin{aligned}\min_{Z\succeq 0, p, \varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 + \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} p_{k,l}\varepsilon_{k,l}^2 \quad (6)\\ \text{s. t.} \quad & T_{i,j}Z_{i,j} \geq 1 - \varepsilon_{i,j}, \ \forall (i,j)\in\mathcal{T}\\ & \varepsilon_{k,l} - 1 \geq Z_{k,l} \geq 1 - \varepsilon_{k,l}, \forall (k,l)\notin\mathcal{T}\\ & \sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq 1, \ p_{k,l} \geq 0, \forall (k,l)\notin\mathcal{T}\end{aligned}$$

The following theorem shows the relationship between (6) and (5).

Theorem 3. *Any global optimal solution to (5) is also a global optimal solution to (6).*

The proof of the above theorem can be found in Appendix A.

Unfortunately, the optimization problem in (6) is nonconvex because of the term $p_{k,l}\varepsilon_{k,l}^2$. It is therefore difficult to find the global optimal solution for (6). In order to turn (6) into a convex optimization problem, we view the constraint $\sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq 1$ as a bound for the arithmetic mean of $p_{k,l}$, i.e.,

$$\frac{1}{m}\sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq \frac{1}{m}$$

where $m = |\{(k,l)|(k,l)\notin\mathcal{T}\}|$. We then relax this constraint by the harmonic mean of $p_{k,l}$, i.e.,

$$\frac{m}{\sum_{(k,l)\notin\mathcal{T}} p_{k,l}^{-1}} \geq \frac{1}{m}, \text{ or } \sum_{(k,l)\notin\mathcal{T}} p_{k,l}^{-1} \leq m^2$$

The above relaxation is based on the property that a harmonic mean is no larger than an arithmetic mean. By replacing the constraint $\sum_{(k,l)\notin\mathcal{T}} p_{k,l} \leq 1$ with (7), we have (6) relaxed into the following optimization problem

$$\begin{aligned}\min_{Z\succeq 0, p, \varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 + \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} p_{k,l}\varepsilon_{k,l}^2 \quad (7)\\ \text{s. t.} \quad & T_{i,j}Z_{i,j} \geq 1 - \varepsilon_{i,j}, \varepsilon_{i,j} \geq 0, \ \forall (i,j)\in\mathcal{T}\\ & \varepsilon_{k,l} - 1 \geq Z_{k,l} \geq 1 - \varepsilon_{k,l}, \forall (k,l)\notin\mathcal{T}\\ & \sum_{(k,l)\notin\mathcal{T}} p_{k,l}^{-1} \leq m^2, \ 0 \leq p_{k,l} \leq 1, \forall (k,l)\notin\mathcal{T}\end{aligned}$$

By defining variable $h_{k,l} = p_{k,l}^{-1}$, we have

$$\begin{aligned}\min_{Z\succeq 0, h, \varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \varepsilon_{i,j}^2 + \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} \frac{\varepsilon_{k,l}^2}{h_{k,l}} \quad (8)\\ \text{s. t.} \quad & T_{i,j}Z_{i,j} \geq 1 - \varepsilon_{i,j}, \varepsilon_{i,j} \geq 0, \ \forall (i,j)\in\mathcal{T}\\ & \varepsilon_{k,l} - 1 \geq Z_{k,l} \geq 1 - \varepsilon_{k,l}, \forall (k,l)\notin\mathcal{T}\\ & \sum_{(k,l)\notin\mathcal{T}} h_{k,l} \leq m^2, \ h_{k,l} \geq 1, \forall (k,l)\notin\mathcal{T}\end{aligned}$$

Notice that constraint $0 \leq p_{k,l} \leq 1$ is transferred into $h_{k,l} \geq 1$. The following theorem shows the property of the formulation in (8)

Theorem 4. *We have the following properties for (8)*

- *(8) is a semi-definite programming (SDP) problem.*
- *Any feasible solution to (8) is also a feasible solution to (5) with $p_{k,l} = h_{k,l}^{-1}$, and the optimal value for (6) is upper bounded by that for (8).*

The proof is provided in Appendix B. Note that using $\varepsilon_{i,j}^2$ instead of $\varepsilon_{i,j}$ for the loss function is key to turning (6) into a convex optimization problem. The second property stated in Theorem 4 indicates that by minimizing (8), we guarantee a small value for the objective function in (6).

The following theorem shows the dual problem of (8), which is the key to the efficient computation.

Theorem 5. *The dual problem of (8) is*

$$\begin{aligned}\max_{Q,W} \quad & \sum_{(i,j)\in\mathcal{T}} \left(Q_{i,j} - \frac{Q_{i,j}^2}{2c}\right) + \sum_{(k,l)\notin\mathcal{T}} \left(|W_{k,l}| - \frac{W_{k,l}^2}{2c}\right)\\ & -\frac{2(m^2-m)}{c}\lambda \qquad (9)\\ s.\ t \quad & L \succeq Q\otimes T + W\otimes\bar{T}\\ & \forall (i,j)\in\mathcal{T}, Q_{i,j}\geq 0, \quad \lambda \geq W_{k,l}^2, \forall (k,l)\notin\mathcal{T}\end{aligned}$$

where matrix $\bar{T}$ is defined as

$$\bar{T}_{i,j} = \left\{\begin{array}{ll} 0 & (i,j)\in\mathcal{T} \\ 1 & \textit{otherwise}\end{array}\right.,$$

and $\otimes$ stands for the element wise product of matrices.

The proof can be found in Appendix C. In the dual problem, variables $Q_{i,j}$ and $W_{i,j}$ are the dual variables that indicate the importance of labeled example pairs and unlabeled examples, respectively. We thus will select the unlabeled example pair with the largest $|W_{i,j}|$. To speed up the computation, in our experiment, we first select a subset of example pairs (fixed 200) with smallest $|Z_{i,j}|$ using the current kernel matrix Z. We then set all $W_{k,l}$ to be zero if the corresponding pair is not selected. In this way, we significantly reduce the number of variables in the dual problem in (9), thus simplifying the computation.

3. Experimental Results

In our experiments, we follow the work (Hoi et al., 2007), and evaluate the proposed algorithm for active kernel learning by the experiments of data clustering. More specifically, we first apply the active kernel learning algorithm to identify the most informative example pairs, and then solicit the class labels for the selected example pairs. A kernel matrix will be learned from the labeled example pairs, and the learned kernel matrix will be used by the clustering algorithm to find the right cluster structure.

3.1. Experimental Setup

We use the same datasets as the ones described in (Hoi et al., 2007). Table 1 summarizes the information about the nine datasets used in our study. We adopt the clustering accuracy defined in (Xing et al., 2002) as the evaluation metric. It is defined as follows

$$Accuracy = \sum_{i>j} \frac{\mathbf{1}\{\mathbf{1}\{c_i = c_j\} = \mathbf{1}\{\hat{c}_i = \hat{c}_j\}\}}{0.5n(n-1)}, \quad (10)$$

where $\mathbf{1}\{\cdot\}$ is the indicator function that outputs 1 when the input argument is true and 0 otherwise. c_i and $\hat{c}_i$ denote the true cluster membership and the predicted cluster membership of the ith data point, respectively. n is the number of examples in the dataset. For the graph Laplacian L used by the nonparametric kernel learning, we apply the standard method for all experiments, i.e., by calculating the distance matrix by Euclidean distance, then constructing the adjacency matrix with five nearest neighbors, and finally normalizing the graph to achieve the final Laplacian matrix.

3.2. Performance Evaluation

To evaluate the quality of the learned kernels, we extend the proposed kernel learning algorithm to solve clustering problems with pairwise constraints. In the experiments, we employ the kernel k-means as the clustering method, in which the kernel is learned by the proposed non-parametric kernel learning method. In addition to the proposed active kernel learning method, two baseline approaches are implemented to select informative example pairs for kernel learning. Totally we have:

- **Random**: This baseline method randomly samples example pairs from the pool of unlabeled pairs.

- **AKL-min-|Z|**: This baseline method chooses the pair examples with the least $|Z_{k,l}|$, where matrix Z is learned by the non-parametric kernel learning method. As already discussed in the introduction section, this approach may not find the most informative example pairs.

- **AKL-min-H**: This is the proposed AKL algorithm. It selects the example pairs with least $H_{k,l}$ that corresponds to the maximal selection probability $P_{k,l}$.

To examine the performance of the proposed AKL algorithm in a full spectrum, we evaluate the clustering results with respect to different sampling sizes. Specifically, for each experiment, we first randomly sample N_c pairwise constraints as the initially labeled pair examples. We then employ the nonparametric kernel learning method to learn a kernel from the given pairwise constraints. This learned kernel is engaged by the kernel k-means method for data clustering. Next, we apply the AKL method to sample 20 pair examples (i.e. 20 pairwise constraints) for labeling in an iteration, and then examine the clustering results based on the kernel that is learned from the augmented set of example pairs in each iteration.

Table 1. The nine datasets used in our experiments. The first two are the artificial datasets from (Hoi et al., 2007) and the others are from the UCI machine learning repository.

Dataset	#Classes	#Instances	#Features
Chessboard	2	100	2
Double-Spiral	2	100	3
Glass	6	214	9
Heart	2	270	13
Iris	3	150	4
Protein	6	116	20
Sonar	2	208	60
Soybean	4	47	35
Wine	3	178	12

Each experiment is repeated 50 times with multiple restarts for clustering. Fig. 2 shows the experimental results on the nine datasets with five active kernel learning iterations. First of all, we observe that AKL-min-$|Z|$, i.e., the naive AKL approach that samples the example pairs with the least $|Z|$, does not always outperform the random sampling approach. In fact, it only outperforms the random sampling approach on five out of the nine datasets. It performs noticeably worse than the random approach on dataset “sonar” and “heart”. Compared with the two baseline approaches, the proposed AKL algorithm (i.e., AKL-min-H) achieves considerably better performance for most datasets. For example, for the “Double-Spiral” dataset, after 3 active kernel learning iterations, the proposed algorithm is able to achieve the clustering accuracy of 99.6%, but the clustering accuracies of the other two methods are less than 98.8%. These experimental results show the effectiveness of the proposed algorithm as a promising approach for active kernel learning.

4. Conclusion

In this paper we proposed a min-max framework for active kernel learning that specifically addresses the problem of how to identify the informative pair examples for efficient kernel learning. A promising algorithm is presented that approximates the original min-max optimization problem into a convex programming problem. Empirical evaluation based on the performance of data clustering showed that our proposed algorithm for active kernel learning is effective in identifying informative example pairs for the learning of kernel matrix.

Acknowledgments

The work was supported in part by the National Science Foundation (IIS-0643494), National Institute of Health (1R01-GM079688-01), and Singapore NTU AcRF Tier-1 Research Grant (RG67/07). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF and NIH.

Appendix A: Proof of Theorem 3

Proof. First, for any global optimal solution to (6), we have $\sum_{(k,l)\notin\mathcal{T}} p_{k,l} = 1$ though the constraint in (6) is $\sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq 1$. This is because we can always scale down $p_{k,l}$ if $\sum_{(k,l)\notin\mathcal{T}} p_{k,l} > 1$, which guarantees to reduce the objective function. Second, any extreme point solution (i.e., $p_{k,l} = 1$ for one example pair and zero for other pairs) to (6) is a global optimal solution to (5). This is because (6) is a relaxed version of (5). Third, one of the global optimal solutions to (6) is an extreme point. This is because the first order condition of optimality requires $p^*_{k,l}$ to be a solution to the following problem:

$$\begin{aligned}\min_{p} \quad & \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} p_{k,l}[\varepsilon^*_{k,l}]^2 \qquad (11)\\ \text{s. t.} \quad & \sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq 1,\ p_{k,l} \geq 0, \forall (k,l) \notin \mathcal{T}\end{aligned}$$

where $\varepsilon^*_{k,l}$ is the optimal solution for $\varepsilon_{k,l}$. Since (11) is a linear optimization problem, it is well known that one of its global optimal solutions is an extreme point. Combining the above arguments together, we prove there exists a global solution to (5), denoted by $((k,l)^*, Z^*, \varepsilon^*_{i,j})$ that is also a global solution to (6) with $p_{(k,l)^*} = 1$. We extend this conclusion to any other global solution $((k,l)', Z', \varepsilon'_{i,j})$ to (5) because $((k,l)', Z', \varepsilon'_{i,j})$ results in the same value for the problem in (6) as solution $((k,l)^*, Z^*, \varepsilon^*_{i,j})$. This completes our proof. □

Appendix B: Proof of Theorem 4

Proof. To show (8) is a SDP problem, we introduce slack variables for both labeled and unlabeled example pairs, i.e., $\eta_{i,j} \geq \varepsilon^2_{i,j}$ and $\eta_{k,l} \geq \varepsilon^2_{k,l}/h_{k,l}$. We can turn these two nonlinear constraints into LMI constraints, i.e.,

$$\begin{pmatrix} \eta_{i,j} & \varepsilon_{i,j} \\ \varepsilon_{i,j} & 1 \end{pmatrix} \succeq 0, \quad \begin{pmatrix} \eta_{k,l} & \varepsilon_{k,l} \\ \varepsilon_{k,l} & h_{k,l} \end{pmatrix} \succeq 0$$

Using the slack variables, we rewrite (8) as

$$\begin{aligned}\min_{Z\succeq 0, h, \varepsilon} \quad & \mathrm{tr}(LZ) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \eta_{i,j} + \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} \eta_{k,l} (12)\\ \text{s. t.} \quad & T_{i,j}Z_{i,j} \geq 1 - \varepsilon_{i,j}, \varepsilon_{i,j} \geq 0,\ \forall (i,j) \in \mathcal{T}\\ & \varepsilon_{k,l} - 1 \geq Z_{k,l} \geq 1 - \varepsilon_{k,l}, \forall (k,l) \notin \mathcal{T}\\ & \sum_{(k,l)\notin\mathcal{T}} h_{k,l} \leq m^2,\ h_{k,l} \geq 1, \forall (k,l) \notin \mathcal{T}\\ & \begin{pmatrix} \eta_{i,j} & \varepsilon_{i,j} \\ \varepsilon_{i,j} & 1 \end{pmatrix} \succeq 0, \forall (i,j) \in \mathcal{T}\\ & \begin{pmatrix} \eta_{k,l} & \varepsilon_{k,l} \\ \varepsilon_{k,l} & h_{k,l} \end{pmatrix} \succeq 0, \forall (k,l) \notin \mathcal{T},\end{aligned}$$

which is clearly a SDP problem.

To show the second part of theorem, we follow the inequality that a harmonic mean is upper bounded by an arithmetic mean, i.e.,

$$\frac{1}{m}\sum_{(k,l)\notin\mathcal{T}} p_{k,l} \geq \frac{m}{\sum_{(k,l)\notin\mathcal{T}} p^{-1}_{k,l}} = \frac{m}{\sum_{(k,l)\notin\mathcal{T}} h_{k,l}} \geq \frac{1}{m}$$

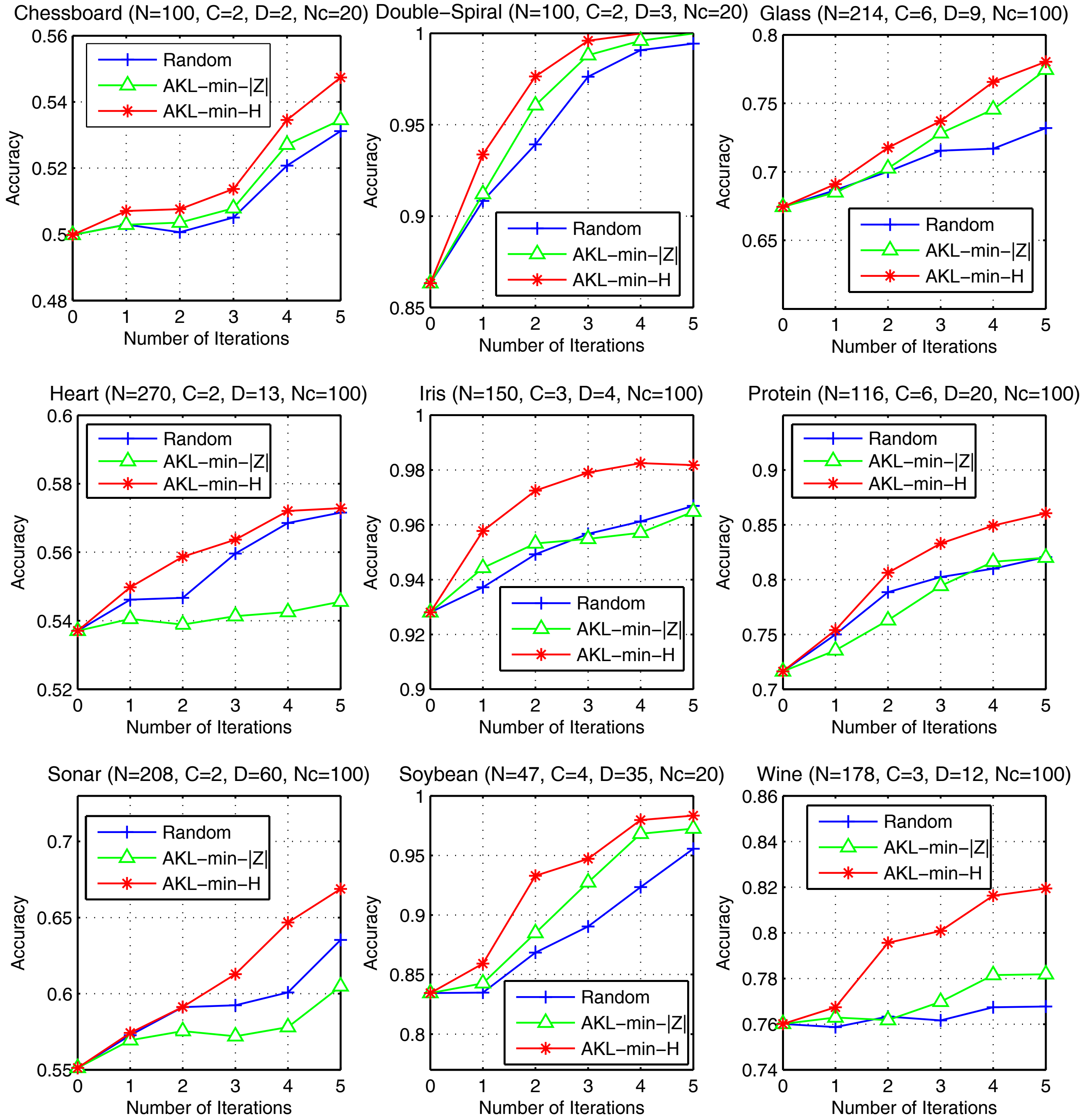

Figure 2. The clustering accuracy of different AKL methods for kernel k-means algorithms with nonparametric kernels learned from pairwise constraints. In each individual diagram, the three curves are respectively the random sampling method, the active kernel learning method for selecting pair examples with the least $|Z_{k,l}|$ (AKL-min-$|Z|$), and the active kernel learning method with minimal H values learned from our proposed algorithm (AKL-min-H). The details of the datasets are also shown in each diagram. In particular, N, C, D, and Nc respectively denote the dataset size, the number of classes, the number of features, and the number of initially sampling pairwise constraints. In each of the five iterations, 20 pair examples are sampled for labeling by the compared algorithms.

Hence, any feasible solution to (8) is also a feasible solution to (6), and (8) is a restricted version of (8), which leads to the conclusion that the optimal output value for (8) provides the upper bound for that of (6). □

Appendix C: Proof of Theorem 5

Proof. We first constructe the Lagrangian function for the above problem

$$\begin{aligned}\mathcal{L} = {} & \mathrm{tr}(L^\top Z) + \frac{c}{2}\sum_{(i,j)\in\mathcal{T}} \eta_{i,j} + \frac{c}{2}\sum_{(k,l)\notin\mathcal{T}} \eta_{k,l} \\ & - \sum_{(i,j)\in\mathcal{T}} Q_{i,j}(T_{i,j}Z_{i,j} + \varepsilon_{i,j} - 1) \\ & - \sum_{(i,j)\in\mathcal{T}} (\alpha_{i,j}\eta_{i,j} + \tau_{i,j}/2 - 2\beta_{i,j}\varepsilon_{i,j}) - \mathrm{tr}(MZ) \\ & - \sum_{(k,l)\notin\mathcal{T}} s_{k,l}(h_{k,l} - 1) - \lambda\left(m^2 - \sum_{(k,l)\notin\mathcal{T}} h_{k,l}\right) \\ & - \sum_{(k,l)\notin\mathcal{T}} (\alpha_{k,l}\eta_{k,l} + \tau_{k,l}h_{k,l}/2 - 2\beta_{k,l}\varepsilon_{k,l}) \\ & - \sum_{(k,l)\notin\mathcal{T}} W_{k,l}Z_{k,l} + (\varepsilon_{k,l} - 1)|W_{k,l}|\end{aligned}$$

In the above, we introduce Lagrangian multiplier

$$\begin{pmatrix} \alpha_{i,j} & -\beta_{i,j} \\ -\beta_{i,j} & \tau_{i,j}/2 \end{pmatrix}$$

for constraints

$$\begin{pmatrix} \eta_{i,j} & \varepsilon_{i,j} \\ \varepsilon_{i,j} & 1 \end{pmatrix} \succeq 0 \text{ and } \begin{pmatrix} \eta_{k,l} & \varepsilon_{k,l} \\ \varepsilon_{k,l} & h_{k,l} \end{pmatrix} \succeq 0$$

By setting the derivative to be zero, we have

$$\begin{aligned} \max \; & \sum_{(i,j)\in\mathcal{T}} \left(Q_{i,j} - \frac{\tau_{i,j}}{2}\right) + \sum_{(k,l)\notin\mathcal{T}} \left(|W_{k,l}| - \frac{\tau_{k,l}}{2}\right) \quad (13) \\ & -(m^2 - 1)\lambda \\ \text{s. t} \; & L \succeq Q \otimes T + W \otimes \bar{T} \\ & \begin{pmatrix} c & -Q_{i,j} \\ -Q_{i,j} & \tau_{i,j} \end{pmatrix} \succeq 0, Q_{i,j} \geq 0, \; \forall (i,j) \in \mathcal{T} \\ & 0 \leq \tau_{k,l} \leq 2\lambda, \; \forall (k,l) \notin \mathcal{T} \\ & \begin{pmatrix} c & -|W_{k,l}| \\ -|W_{k,l}| & \tau_{k,l} \end{pmatrix} \succeq 0, \; \forall (k,l) \notin \mathcal{T} \end{aligned}$$

The two LMI constraints can be simplified as

$$\tau_{i,j} \geq 2Q_{i,j}^2/c, \quad \tau_{k,l} \geq 2Q_{k,l}^2/c$$

Substituting the above constraints into (13), we have (9). □

References

Belkin, M., & andd P. Niyogi, I. M. (2004). Regularization and semi-supervised learning on large graphs. *Intl. Conf. on Learning Theory (COLT)*.

Chapelle, O., Weston, J., & Schölkopf, B. (2003). Cluster kernels for semi-supervised learning. .

Chen, F., & Jin, R. (2007). Active algorithm selection. *Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI)*.

Cristianini, N., Shawe-Taylor, J., & Elisseeff, A. (2002). On kernel-target alignment. *JMLR*.

Hoi, S. C. H., Jin, R., & Lyu, M. R. (2007). Learning nonparametric kernel matrices from pairwise constraints. *ICML2007*. Corvallis, OR, US.

Hoi, S. C. H., Jin, R., Zhu, J., & Lyu, M. R. (2006). Batch mode active learning and its application to medical image classification. *ICML2006* (pp. 417–424). Pittsburgh, Pennsylvania.

Kondor, R., & Lafferty, J. (2002). Diffusion kernels on graphs and other discrete structures. *ICML'2002*.

Kulis, B., Sustik, M., & Dhillon, I. S. (2006). Learning low-rank kernel matrices. *ICML2006* (pp. 505–512).

Lanckriet, G., Cristianini, N., Bartlett, P., Ghaoui, L. E., & Jordan, M. (2004). Learning the kernel matrix with semi-definite programming. *JMLR*, *5*, 27–72.

Scholkopf, B., & Smola, A. (2002). *Learning with kernels*. MIT Press.

Sturm, J. (1999). Using sedumi: a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, *11–12*, 625–653.

Tong, S., & Koller, D. (2000). Support vector machine active learning with applications to text classification. *ICML2000* (pp. 999–1006). Stanford, US.

Vapnik, V. N. (1998). *Statistical learning theory*. John Wiley & Sons.

Xing, E. P., Ng, A. Y., Jordan, M. I., & Russell, S. (2002). Distance metric learning with application to clustering with side-information. *NIPS2002*.

Zhu, X., Kandola, J., Ghahramani, Z., & Lafferty, J. (2005). Nonparametric transforms of graph kernels for semi-supervised learning. *NIPS2005*.