A Data-dependent Distance Measure for Transductive Instance-based Learning

Jared Lundell and DanVentura

Abstract— **We consider learning in a transductive setting using instance-based learning (k-NN) and present a method for constructing a data-dependent distance "metric" using both labeled training data as well as available unlabeled data (that is to be classified by the model). This new data-driven measure of distance is empirically studied in the context of various instance-based models and is shown to reduce error (compared to traditional models) under certain learning conditions. Generalizations and improvements are suggested.**

I. Introduction

In many learning scenarios, it is common for data acquisition to be inexpensive compared to data labeling. In such instances, transductive or semi-supervised learning approaches are often useful. These involve not only the traditional machine learning task of model selection but also the question of how to incorporate unlabeled data into the learning process. We consider the application of k-NN classifiers to such situations and address the problem of incorporating unlabeled data into the lazy learning framework of such memory-based models. Our approach involves constructing an explicit, data-driven distance "metric" (in the form of a distance matrix) that incorporates all available data, both labeled and unlabeled. Because we are working in a transductive setting, it makes sense to utilize the additional information provided by the unlabeled data, something many traditional techniques cannot (naturally) do. We therefore create a data-dependent distance matrix using *all* available data and then use that distance matrix in classifying the unlabeled portion of the data. We proceed by initially clustering the data in a semi-supervised manner, and then use the resulting pairwise distances to generate an affinity matrix for use as the distance measure for calculating interpoint distances. We also investigate the effect of preprocessing the data with a nonlinear manifold learner, effectively constraining our clustering to the implicit manifold.

In what follows, we formally describe our approach to constructing the data-dependent distance matrix and present empirical results on several real-world data sets that highlight the characteristic benefits of the technique. Before doing so, however, we briefly mention related work in two significant areas: semi-supervised clustering, and manifold learning.

A. Semi-supervised Clustering

Semi-supervised clustering, a form of transductive inference, is an easier problem than the standard approach of inductive transfer followed by deductive classification. The presence of unlabeled data allows for better estimation of the data's true distribution which can improve classification accuracy. Vapnik [12] has proposed a framework to establish upper bounds on the empirical error of on a set of unlabeled data, D_u, given the empirical error on a set of labeled data, D_l.

Given the widespread adoption of support vector machines, also based on Vapnik's statistical learning theory, it is not surprising that significant effort has gone into combining the principles of transductive learning and SVM's. Broadly, these approaches attempt to combine the maximum margin principle of SVM's with the clustering assumption (points that lie close together are likely to belong to the same class).

An early example of this work is the transductive SVM [6]. In this approach the decision boundary is chosen in such a way to maximize the margin for both labeled and unlabeled data. In other words, the transductive SVM attempts to draw a decision boundary in areas of low data density. Related works that followed include [5] and [10] and more recent work includes variations such as semi-supervised regression [14].

Several approaches to combining the principles of semi-supervised clustering and support vector machines have been proposed. Xu *et. al* [13] use the maximum margin principle to guide clustering, while Chapelle, Weston and Schölkopf [4] generate an affinity matrix that leverages available unlabeled data.

We follow the example of Chapelle *et. al* in our efforts to directly construct a SVM affinity matrix using clustering techniques. In contrast, while their method requires the choice of a "transfer function", ours requires choosing only two scalar parameter values; also, while their method is based on spectral clustering, we use a graph-based clustering more reminiscent of LLE or Isomap with their strong non-linear advantages.

Semi-supervised clustering can be seen as a generalization of the transductive inference problem. Labels or pairwise constraints can be used to modify the standard clustering objective function or to learn an appropriate distance metric. Basu, Bilenko and Moody [2] have proposed several semi-supervised clustering algorithms including Seeded KMeans, HMRF-KMeans, and Pairwise Constraints KMeans.

B. Manifold Learning

Manifold learning, typically for the purpose of dimensionality reduction and/or discovering the intrinsic dimensionality of data, is related to clustering as another approach to

Jared Lundell and Dan Ventura are with the Department of Computer Science, Brigham Young University, Provo, UT 84602, USA (email: jared.lundell@gmail.com, ventura@cs.byu.edu)

computing/discovering distance metrics. Seminal work in this area is represented by the well-known manifold learners Isomap [11] and LLE [9], while more recent work includes extensions of this technique like Spectral Learning [7] and Relevant Component Analysis [1], as well as alternative approaches such as Tensor Voting [8].

II. Methods

Given a set D_L of n_l labeled data points and a set D_U of n_u unlabeled data points, we use a graph-based semi-supervised clustering algorithm to learn a data-specific distance metric, represented by a $n \times n$ matrix of point-to-point distances for $D_L \cup D_U$, where $n = n_l + n_u$. This matrix is then used in calculating interpoint distances for a k-NN model.

A. Distance Matrix Construction

Our goal is to create the matrix $M_D = d_{ij}$, where $1 \leq i, j \leq n$ and d_{ij} is the (data-dependent) distance from point i to point j. We begin by clustering the data. Given data with m classes and n_l labels, we use hierarchical agglomerative clustering with purity thresholding to cluster the data. The purity threshold θ allows us to leverage any labels that are available. The purity ρ of any cluster c is calculated as

$$\rho_c = \begin{cases} \dfrac{n_{k_c}}{n_{l_c}} & \text{if } n_{l_c} > 0 \\ 1 & \text{if } n_{l_c} = 0 \end{cases}$$

where n_{k_c} is the count of the most common label in c and n_{l_c} is the total number of labeled points in c. Initially, the set C of clusters contains only clusters c_i consisting of a single point and having a purity of 1. Clusters are then iteratively agglomerated as follows. For each cluster c_i, in C, we find it's nearest neighbor, c_j (we use complete link clustering and the Euclidean metric — the distance between any two clusters is defined as the Euclidean distance between their two most distant points). If $\rho_{(c_i \cup c_j)} \geq \theta$, we remove c_i and c_j from C and add the cluster $(c_i \cup c_j)$ to C. This process is repeated until no further clusters can be combined. Finally, any clusters that remain without labeled points are combined with the nearest cluster that has at least one labeled point. C now contains some number of clusters that can each be identified with a dominant label l_c.

For each cluster c, we compute a virtual center of mass point γ^c:

$$\gamma^c = \frac{1}{n_c} \sum_{\mathbf{x} \text{ in } c} \mathbf{x}$$

where n_c is the total number of points in c (both labeled and unlabeled) and the summation is vector addition with the coefficient a scalar applied to each element in the resulting vector. Now, for clusters c_1 and c_2 we define the distance between their centers of mass as

$$\text{dist}(\gamma^{c_1}, \gamma^{c_2}) = \begin{cases} \|\gamma^{c_1} - \gamma^{c_2}\| & \text{if } l_{c_1} \neq l_{c_2} \\ 0 & \text{if } l_{c_1} = l_{c_2} \end{cases}$$

In other words, if two clusters share a common label, the distance between their centers of mass is defined to be 0; if they have different labels, the distance between their centers of mass is the standard Euclidean distance. Next, a shortest path graph traversal algorithm (optimized Floyd-Warshall) is used to propagate the effect of these "wormholes" between clusters to every pair of data points. In essence, this will create m meta-clusters as the matrix M_D is defined [1]

$$d_{ij} = \|\mathbf{x}_i - \gamma^{c_i}\| + \text{dist}(\gamma^{c_i}, \gamma^{c_j}) + \|\mathbf{x}_j - \gamma^{c_j}\| \qquad (1)$$

where $\mathbf{x}_i$ is the vector representation of point i, c_i is the cluster containing point i, γ^{c_i} is the virtual center of mass of c_i and $\|\cdot\|$ is the Euclidean metric. In other words, we combine all clusters with common labels by decreasing the distance between the points included in such clusters (via the "wormholes").

To improve separability we post-process the matrix M_D to push these meta-clusters apart by increasing the distance between each pair of points in different meta-clusters. The distance is increased by κ times the greatest distance between any two data points sharing the same label (we used $\kappa = 2$ in our experiments). That is,

$$d_{ij} = \begin{cases} d_{ij} + \kappa \max\limits_{k} \max\limits_{i,j \in c_k} d_{ij} & \text{if } l_{c_i} \neq l_{c_j} \\ d_{ij} & \text{if } l_{c_i} = l_{c_j} \end{cases}$$

The resulting distance matrix represents a feature space where each point is grouped with similar points, points that share the same label, and points that are similar to points that share the same label.

III. Empirical Results

We performed extensive empirical testing using values of $k = \{1, 3, 5, 7, 9\}$ for both traditional and data-dependent k-NN on nine different data sets measuring error rates and the effect of cluster purity on performance, using both unweighted and distance weighted voting. In addition, each of the k-NN models was applied after preprocessing the data with a non-linear manifold learning algorithm (we used Relevant Component Analysis [1]). The data sets (*Chess*, *German*, *Heart*, *Pima*, *Spect*, *Voting*, *WBC1*, *WBC2* and *WBC3*) are taken from the UCI repository [3].

For each data set and for each model, we measured classification error on the unlabeled portion of the data. We varied the amount of unlabeled data, and, in addition, for the data-dependent models, the cluster purity was varied as well. We varied the cluster purity θ over the values $\{0.80, 0.85, 0.90.0.95, 1.00\}$ and the amount of unlabeled data as a percentage of the total data available in 10%

[1] Actually, the distances calculated by the shortest path algorithm cannot be so simply characterized. Eq. 1 is representative of the effects the "wormholes" can have on the final distance matrix; however, there are actually 5 different scenarios that can represent the shortest path between two points (for the 2-class case). Eq. 1 gives one of the five cases, but in practice we compute all five cases for each pair of points and take the minimum, with the result being the same as running a regular Djikstra's algorithm, just (much) faster.

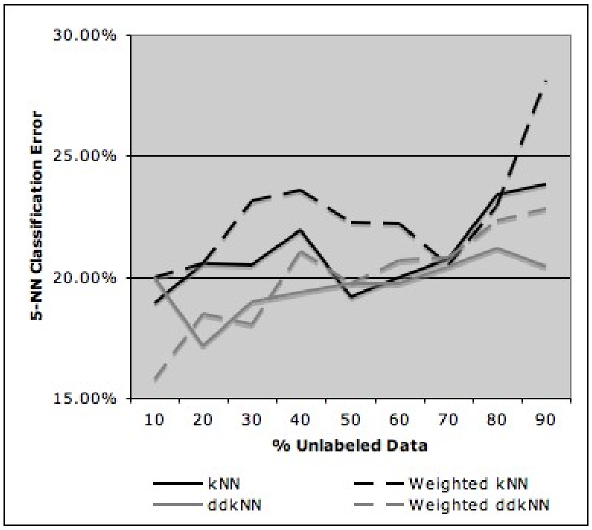

Fig. 1. Classification error using 10-fold cross validation on the *SPECT* data set for varying amounts of unlabeled data. Data points on the curves represent the lowest error value for any value of θ for the data-dependent k-NN. Note that the data-dependent k-NN results in lower error rates across the transductive range and that data-dependent k-NN is more consistent (flatter curves) across the range. Distance weighting is somewhat detrimental for both the traditional and data-dependent cases.

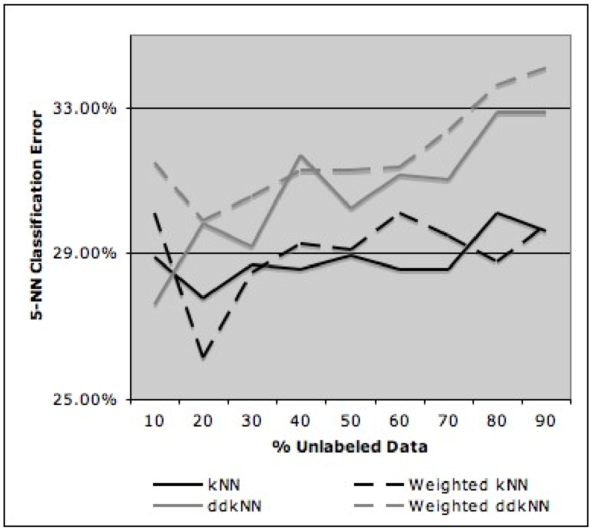

Fig. 2. Classification error using 10-fold cross validation on the *German* data set for varying amounts of unlabeled data. Data points on the curves represent the lowest error value for any value of θ for the data-dependent k-NN. Note that the data-dependent distance results in higher error rates across the transductive range and that traditional k-NN is more consistent (flatter curves) across the range. Distance weighting is somewhat detrimental in the data-dependent case, but has less of an effect in the traditional case.

increments from 10% to 90%. All results were obtained using 10-fold cross validation, resulting in $16,200$ experiments for each of the five data-dependent models and $3,240$ experiments for each of the five traditional models, for a grand total of $97,200$ experimental runs.

A. Transductive Range

We consider the performance of the various models across a range of transductive scenarios. To do this, we treat varying amounts of our data as unlabeled and use only the remaining labeled data as the memory for our nearest neighbor classifiers. For the data-dependent models, the unlabeled data is made use of during the learning of the distance matrix M_d as discussed in Section II (i.e. only the data locations are used to help populate the distance matrix, without regard for the data labels). For the traditional models, since the distance metric is fixed (Euclidean) the unlabeled data is not used transductively.

The results are mixed and expose some interesting behaviors in this learning setting. Tables I and II in the Appendix present a representative sample of performance evaluation data, though due to the number of experiments run, these are still a small sub-sample of the complete set of results. Given enough data and using single nearest neighbor, the data-dependent approach reduces error rates on all nine data sets when compared to the Euclidean metric (see the first two rows of Table I). As the number of neighbors is increased, results become mixed, indicating that perhaps the distance matrix should be coupled to the model in the sense of tuning the distance matrix to the choice of k.

Figure 1 shows *transduction curves* for both traditional and data-dependent k-NN (with and without distance weighted voting) for the *SPECT* data set. Points on the curves are average classification errors for 5-NN, computed using 10-fold cross validation, and, in the data-dependent case represent the lowest average for any value of the clustering threshold θ. Notice that for this data set, the data-dependent k-NN results in lower error rates across the transductive range and that data-dependent k-NN is more consistent (flatter curves) across the range.

In contrast, the curves in Figure 2 reveal that for the *German* data set, the opposite is true — the data-dependent distance results in higher error rates across the transductive range and traditional k-NN is more consistent (flatter curves) across the range.

Figure 3, which gives curves for the *WBC2* data set, highlights a situation in which the data-dependent model performs better than the traditional one at the low end of the curve (that is, when the percentage of unlabeled data is relatively low). However, at the high end of the curve, the situation is reversed. In fact, across all data sets we observed a trend of negative correlation between percentage of unlabeled data and efficacy of the data-dependent models — as the percentage of unlabeled data increases, the merit of the data-dependent model decreases relative to the traditional approach.

To illustrates this trend, we counted the number of data sets for which the data-dependent model had the lowest error and subtracted that number from the number of data sets for which the traditional model had the lowest error. The bars in Figure 4 show this differential, with bars below the midline indicating an advantage for the data-dependent approach and bars above the midline indicating an advantage for the traditional approach. The graph shows differentials for

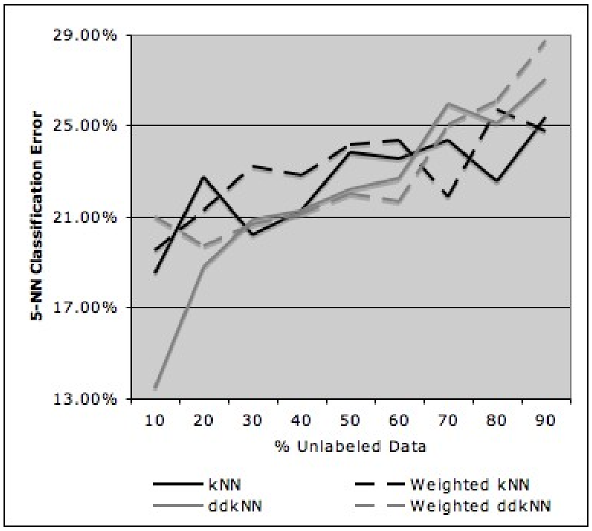

Fig. 3. Classification error using 10-fold cross validation on the *WBC2* data set for varying amounts of unlabeled data. Data points on the curves represent the lowest error value for any value of θ for the data-dependent k-NN. Note that the data-dependent k-NN results in lower error rates at the lower end (less unlabeled data) of the transductive range but become detrimental to performance at the upper end of the range. Distance weighting does not have a significant effect for either the traditional or the data-dependent case.

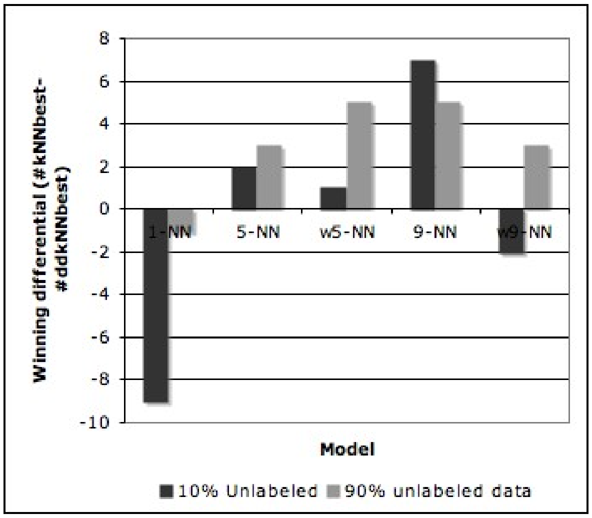

Fig. 4. Winning differential for several different models, calculated over the nine data sets by counting the number of times t that traditional k-NN has better accuracy and the number of times d that data-dependent k-NN has better accuracy, and computing $t-a$. Bars with smaller value favor data-dependent k-NN. Note the upward skew when comparing 10% unlabeled data with 90% unlabeled data

several different models at the extremes of the transduction curve (10% and 90%). Notice the significant upward trend across all models between these two extremes.

This effect is, in fact, not surprising because we are working with a limited amount of data and our clustering for the distance matrix M_d depends upon a densely sampled manifold in order to build a representative distance matrix.

B. RCA Preprocessing

We now consider the effect of performing manifold learning using RCA as a preprocessing step. As examples, Figures 5 and 6 show transduction curves for the *Chess* and *Voting* data sets. Both traditional and data-dependent models were used with and without RCA. For *Chess*, RCA provides a reduction in error rate for both models, helping the data-dependent approach more than it does the traditional one. For *Voting*, both approaches benefit significantly from the RCA preprocessing. Figure 7 shows the result of averaging curves over all nine data sets. Interestingly, in the average case, traditional k-NN derives benefit from the RCA preprocessing while data-dependent k-NN does not. The *average* data-dependent distance matrix appears to be implicitly incorporating the useful properties associated with (nonlinear) manifold learning techniques.

IV. Discussion

We present a method for constructing a data-dependent distance matrix and consider its efficacy in the context of instance based learning. The approach works well for 1-NN, significantly outperforming the standard Euclidean model. Results for models using more neighbors are mixed, likely due to the fact that the distance matrix is uncoupled from the learning model (that is, it does not know the value k). We also consider the use of a nonlinear manifold learner as a preprocessing step and show that on average, the traditional models benefit from this while the data-dependent models do not, suggesting that the distance matrix is (on average) implicitly learning the latent manifold.

In addition to incorporating the value of k into the distance matrix construction, the approach can benefit from the automatic selection of appropriate values for the parameters κ and θ (we empirically chose $\kappa = 2$ and performed a simple grid search over several values of θ).

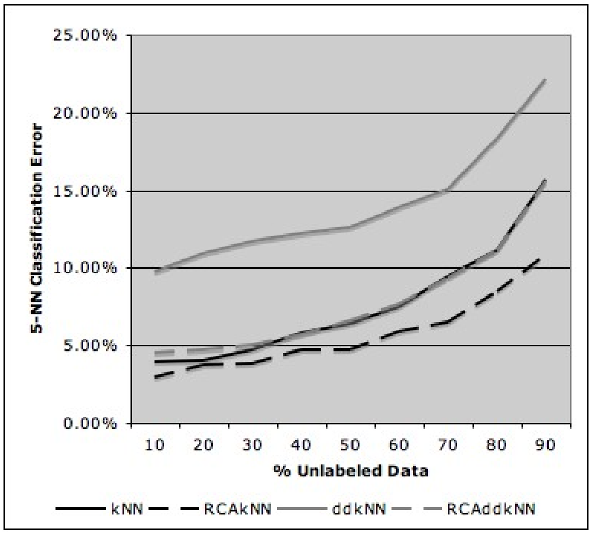

Fig. 5. The effect of RCA preprocessing on the *Chess* data set. In this case, RCA helps the data-dependent approach more than it does the traditional one.

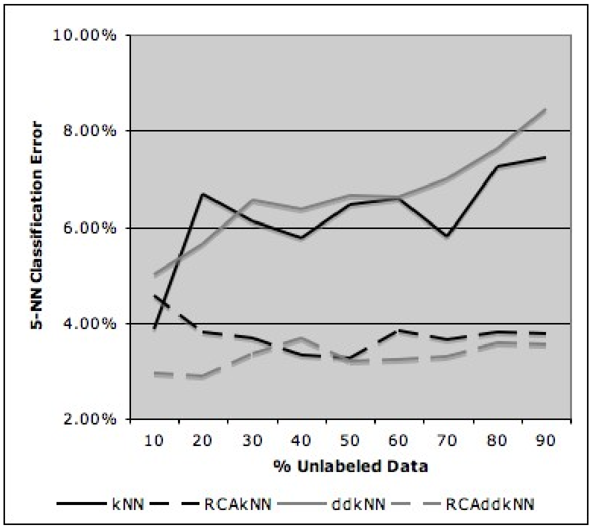

Fig. 6. The effect of RCA preprocessing on the *Voting* data set. In this case, RCA helps both the traditional and data-dependent approaches.

Here we have considered transductive learning (that is, in addition to our labeled training data, we know *a priori* which data we want to classify). In the more general inductive case (where we don't know our test data in advance), the distance matrix we construct in Section II will likely will not contain entries for future test data. Then, how should one find nearest neighbors in the inductive scenario? One approach would be to find the m closest (via Euclidean distance) points that are in the matrix and compute some linear combination of their outputs. A reasonable argument can be made for doing this — we expect that as the space is warped during the distance matrix construction process that *local* Euclidean distances are roughly preserved. It is interesting to note that most, if not all, manifold learning algorithms make this same locality assumption. In effect, the subset of the data that are labeled are driving a *global* change in the meaning of distance, while at the same time dragging their close (in the Euclidean sense) unlabeled neighbors along for the ride.

Finally, it will be interesting to ask whether the data-dependent k-NN model can be compressed while preserving performance. This could be done using any of the traditional methods for compressing memory based models, though these may need to be modified to take into account the affect that eliminating data has on the distance matrix. As an alternative compression, one can consider using the virtual cluster centers γ^c as means for a radial basis function network (setting the widths of the basis functions by considering cluster variance). The distance metric may also prove useful as the kernel of an SVM.

References

[1] Aharon Bar-Hillel, Tomer Hertz, Noam Shental, and Daphna Weinshall. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6(Jun):937–965, 2005.

[2] Sugato Basu, Mikhail Bilenko, and Raymond J. Mooney. A probabilistic framework for semi-supervised clustering. In *KDD04*, pages 59–68, Seattle, WA, Aug 2004.

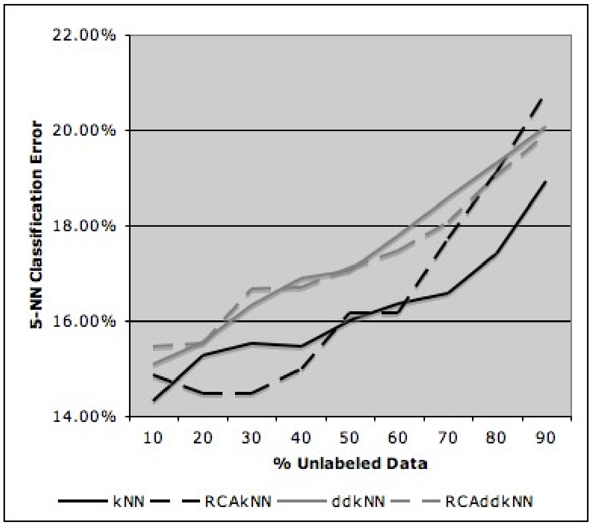

Fig. 7. The effect of RCA preprocessing on average error (across all nine data sets) across the transductive range. Notice that RCA, on average, has little effect on the data-dependent method across the entire range. In contrast, RCA preprocessing helps traditional k-NN at the low end of the range while hurting its performance at the high end.

[3] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.

[4] Olivier Chapelle, Jason Weston, and Bernhard Schölkopf. Cluster kernels for semi-supervised learning. In *Advances in Neural Information Processing Systems 15*, pages 585–592, 2003.

[5] T. Joachims. Transductive learning via spectral graph partitioning. In *Proceedings of the International Conference on Machine Learning*, pages 290–297, 2003.

[6] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the International Conference on Machine Learning*, pages 200–209, 1999.

[7] S. Kamvar, D. Klein, and C. Manning. Spectral learning. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, pages 561 – 566, 2003.

[8] Philippos Mordohai and Gérard Medioni. Unsupervised dimensionality estimation and manifold learning in high-dimensional spaces by tensor voting. In *Proceedings of the Joint Conferences on Artificial Intelligence*, pages 908–913, 2005.

[9] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[10] Martin Szummer and Tommi Jaakkola. Partially labeled classification with markov random walks. In *Advances in Neural Information Processing Systems 14*, pages 945–952, 2001.

[11] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

[12] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[13] Linli Xu, James Neufeld, Bryce Larson, and Dale Schuurmans. Maximum margin clustering. In *Advances in Neural Information Processing Systems 17*, pages 1537–1544, 2005.

[14] Zhi-Hua Zhou and Ming Li. Semi-supervised regression with co-training. In *Proceedings of the Joint Conferences on Artificial Intelligence*, pages 908–913, 2005.

TABLE I

Performance comparison for $k = \{1, 5, 9\}$ for traditional k-NN and data-dependent k-NN (labels containing 'dd'), with and without distance weighting (labels are prefaced with a 'w'). Results reported are the lowest average error (10-fold cross validation) for any value of θ using 10% unlabeled data. Bold entries indicate the lower error rate between the traditional and the data-dependent approach for each case.

	Chess	German	Heart	Pima	Spect	Vote	WBC1	WBC2	WBC3
1-NN	9.78%	33.90%	26.67%	32.21%	24.07%	6.36%	4.86%	28.00%	7.54%
1-ddNN	**8.94**%	**30.60**%	**24.44**%	**30.00**%	**20.37**%	**4.09**%	**3.14**%	**19.00**%	**4.21**%
5-NN	**3.94**%	28.90%	**17.04**%	**27.53**%	**18.89**%	**3.86**%	5.86%	18.50%	**4.39**%
5-ddNN	9.78%	**27.60**%	22.96%	28.57%	20.00%	5.00%	**3.86**%	**13.50**%	**4.39**%
w5-NN	**3.16**%	**30.10**%	**18.89**%	29.22%	20.00%	6.36%	6.43%	**19.50**%	**4.74**%
w5-ddNN	9.91%	31.50%	22.59%	**28.83**%	**15.74**%	**4.77**%	**4.86**%	21.00%	5.00%
9-NN	**4.56**%	**25.80**%	**15.56**%	**27.27**%	**19.63**%	**3.86**%	7.57%	**17.00**%	**4.21**%
9-ddNN	9.53%	30.90%	19.63%	28.83%	20.37%	4.77%	**5.14**%	18.50%	5.61%
w9-NN	**3.88**%	**28.60**%	**18.52**%	28.96%	19.63%	6.36%	6.86%	**21.00**%	6.14%
w9-ddNN	10.41%	31.40%	22.59%	**28.83**%	**15.56**%	**4.77**%	**5.71**%	**21.00**%	**5.09**%

TABLE II

Performance comparison for $k = \{1, 5, 9\}$ for traditional k-NN and data-dependent k-NN (labels containing 'dd'), with and without distance weighting (labels are prefaced with a 'w'). Results reported are the lowest average error (10-fold cross validation) for any value of θ using 90% unlabeled data. Bold entries indicate the lower error rate between the traditional and the data-dependent approach for each case.

	Chess	German	Heart	Pima	Spect	Vote	WBC1	WBC2	WBC3
1-NN	**18.03**%	**34.06**%	**26.42**%	32.74%	28.38%	8.04%	7.90%	31.63%	**7.01**%
1-ddNN	21.28%	34.51%	27.33%	**32.53**%	**23.71**%	**7.55**%	**6.07**%	**29.61**%	8.28%
5-NN	**15.69**%	**29.60**%	**21.73**%	30.19%	23.83%	**7.42**%	9.76%	**25.34**%	**6.82**%
5-ddNN	22.19%	32.88%	25.14%	**29.39**%	**20.42**%	8.44%	**6.92**%	27.02%	8.26%
w5-NN	**15.69**%	**29.77**%	**19.44**%	**30.64**%	28.10%	**8.55**%	10.13%	**24.78**%	**6.81**%
w5-ddNN	21.94%	34.10%	23.79%	31.84%	**22.83**%	9.36%	**7.81**%	28.76%	7.73%
9-NN	**15.88**%	**28.20**%	**21.85**%	**28.86**%	22.75%	**7.93**%	12.05%	**24.61**%	**7.23**%
9-ddNN	22.02%	32.71%	26.01%	31.59%	**21.38**%	9.62%	**5.98**%	27.58%	8.42%
w9-NN	**15.36**%	**29.19**%	**18.97**%	**30.75**%	27.27%	9.36%	12.45%	**24.16**%	**7.36**%
w9-ddNN	21.82%	33.88%	23.70%	32.00%	**22.58**%	**9.31**%	**7.81**%	28.43%	8.07%