Fusing Similarities and Euclidean Features with Generative Classifiers

Luca Cazzanti
Applied Physics Laboratory
University of Washington
Seattle, WA, U.S.A.
luca@apl.washington.edu

Maya R. Gupta
Dept. of Electrical Engineering
University of Washington
Seattle, WA, U.S.A.
gupta@ee.washington.edu

Santosh Srivastava
Fred Hutchinson
Cancer Research Center
Seattle, WA, U.S.A.
ssrivast@fhcrc.org

Abstract – *We introduce two generative classifiers that classify based on the pairwise similarities between samples or on the Euclidean features describing the samples: the regularized local similarity discriminant analysis classifier for similarities and the local Bayesian discriminant analysis classifier for Euclidean features. Both new classifiers provide low-variance probability estimates of class labels from low-bias probabilistic models in their respective domains. We combine these two novel classifiers in a naive Bayes framework to form a classifier that fuses similarity and feature information to produce accurate probability estimates for the class labels. Experimental results on several benchmark datasets demonstrate that the two classifiers improve upon the state-of-the-art in their respective domains, and that the fused classifier adaptively uses the best information for classification.*

Keywords: similarity-based classification, regularized local similarity discriminant analysis, local Bayesian discriminant analysis, classifier fusion

1 Learning From Features and Similarities

For standard metric classifiers, each training and test sample is characterized by numerical features represented by a d-dimensional numerical vector in a Euclidean space [1]. In this paper, we refer to these classifiers as feature-based classifiers. An alternative learning architecture is similarity-based learning, in which classifiers learn from a set of pairwise training similarities, training class labels, and from the similarities between a test sample and the training samples [2]; Euclidean features characterizing the samples are not used. We refer to these classifiers as similarity-based classifiers.

In this paper we consider the problem of classifying given both pairwise similarities and Euclidean features. The goal is to fuse the similarity information and Euclidean features to produce accurate probability estimates of the class labels. Probabilistic estimates are important in practice because there may be skewed class priors or asymmetric misclassification costs, or probabilities of each class may be used as an input to the next component in the system or to fuse with probabilistic information about the class label derived from other sources.

One option for combining feature and similarity information is to convert the similarity information into Euclidean features and then use a feature-based classifier on the concatenation. Another option is to use a classifier for each information type, then learn how to weight them to produce good results. However, this requires learning one fixed set of weights for combining the classifiers, but the optimal weighting might be very different for different test samples. Another option is to fuse the similarity information and features by using a k nearest-neighbor classifier that forms a weighted vote of the k-nearest neighbors with respect to each type of information. This has the advantage that the two classifiers are adaptively weighted for each test sample.

Here, we investigate generative classifier solutions for fusing similarities and features. Generative classifiers produce probabilistic outputs, and naturally adapt the weight given to each component classifier according to each classifier's confidence. In this paper we make three contributions. First, we advance the state-of-the-art in feature-based generative classifiers with a local Bayesian version of quadratic discriminant analysis (QDA) we term *local BDA*. We show that local BDA can reduce the high bias of QDA more effectively than Gaussian mixture models. Second, we advance the state-of-the-art in similarity-based generative classifiers with a regularized local similarity discriminant analysis (*regularized local SDA*) classifier that builds on the recently-proposed local similarity discriminant analysis (SDA) classifier [3]. The regularized local SDA classifier reduces the estimation variance of local SDA. Third, we show that fusing the local BDA and local SDA posteriors in a naive Bayes framework can reduce classification errors.

First, we review Bayesian QDA and propose the local BDA classifier. Then in Section 3 we review approaches to similarity-based classification and the local SDA classifier. We propose regularized local SDA in Section 4. Then we detail the proposed fusion of similarity and feature information in Section 5. In Section 6 we demonstrate the effectiveness of all three proposals on eleven real-world datasets. We conclude with a summary of findings and some open questions.

2 Local BDA

The most common approach to generative classification given d-dimensional Euclidean features $\{x_i\}$ for $i = 1, \ldots, n$ is to assume that conditioned on the associated class $y_i \in \{1, 2, \ldots, G\}$, the feature values have a Gaussian distribution. The Gaussian model can be motivated by the central limit theorem and by the maximum entropy principle. The resulting quadratic discriminant analysis (QDA) classifier has many variations, including pooling the covariances between classes (LDA), modeling the class-conditional distribution as a mixture of Gaussians (GMM), and using nonlinear functions of the features (FDA); see [1] for details.

The Gaussian model requires the estimation of its mean and covariance, and to estimate a full Gaussian distribution the maximum likelihood estimation requires roughly d^2 training samples for d feature dimensions to avoid ill-posed estimates. However, enough training samples may not be available. The standard solution is to regularize the parameter estimation, using for example regularized QDA [4] or eigenvalue-decomposition discriminant analysis [5]. A related approach is Bayesian estimation, which averages all the possible Gaussians weighted by their likelihood to have generated the training samples and their prior probability. Bayesian estimation was first applied to QDA in the 1960's [6, 7], but those proposals did not yield better performance than regularized QDA [8, 9]. Recently, Srivastava et al. [10] proposed a data-dependent prior for Bayesian QDA that uses a coarse estimate of the covariance to determine an appropriate prior for the full covariance matrix. This prior is the inverted Wishart distribution with scale parameter q and matrix parameter B_g. The resulting BDA7 algorithm was shown to perform better than other state-of-the-art QDA methods, particularly for cases where $n < d$.

Using a single Gaussian to model a class-conditional distribution may be too restrictive a model. The standard solution is to learn a Gaussian mixture model (GMM) [1]. Alternatively, we propose that QDA can be made more flexible by applying Bayesian QDA to a neighborhood of the k-nearest neighbors of a given test sample from each class, which we term *local BDA*. For our local BDA classifier, we fix the inverted Wishart prior scalar parameter $q = d + 3$, which makes the prior the standard inverted gamma distribution if $d = 1$ [11]. To avoid any ill-posed numerical problems and to avoid having to cross-validate any hyperparameters, we fix the inverted Wishart prior's matrix-parameter to be

$$B_g = 0.95q\text{diag}\left(\hat{\Sigma}_{ML,g}\right) + 0.05I,$$

where $\text{diag}\left(\hat{\Sigma}_{ML,g}\right)$ is the diagonal of the maximum likelihood estimate of the local class g covariance matrix. For very small k, the diagonal may be ill-posed, and so we regularize it with the identity matrix I; the choice of 5% regularization was chosen without experimentation to be small enough that the emphasis is on the data-dependent diagonal, but large enough to ensure that the inverse of B_g is not ill-posed.

Then, following from [10, Theorem 1], the estimated local class-conditional likelihood for the gth class is given in (1), where $\Gamma(\cdot)$ is the standard gamma function, $I_{(\cdot)}$ is the indicator function, and $\bar{x}_g$ is the average of the k nearest training feature vectors from class g.

3 Review of Similarity-based Classification

In this section we review similarity-based classification and the local SDA classifier. A more complete review can be found in the recent survey paper [2]. Classifying based on similarities between samples differs from standard feature-based learning because similarity functions need not satisfy the metric properties of minimality, symmetry, and the triangle inequality. Similarities are more general than Euclidean distance, and can capture peculiar aspects of the pairwise relationship between samples that are difficult to represent with feature vectors in metric spaces. For example, researchers have gathered extensive evidence that non-metric similarities are the rule rather than the exception when people judge the similarity of concepts [12, 13]. Non-metric similarities arise naturally in cognitive psychology, genomics, proteomics, natural language processing, commerce, and computer vision. Examples of such similarities are the tangent distance used in optical character recognition, the asymmetric color distortion function ΔE_{94} used in color management, and the relative entropy between distributions. In Section 6, we discuss six more examples of similarities that we use in the experiments.

One approach to similarity-based classification is to approximately embed the samples in a metric space and then use the resulting Euclidean features with standard feature-based classifiers. A related approach is to approximate the $n \times n$ similarity matrix by a positive semidefinite (PSD) matrix, and use that PSD matrix as a kernel in a kernel-based learning method such as an SVM [2, 14, 15]. However, these approaches artificially force a metric structure onto the natively nonmetric similarity, at the risk of losing the discriminative information provided by the similarity [12]. This issue is ameliorated if the PSD approximation is less severe because it is done locally, for example by using a local SVM [16] or using kernel ridge interpolation or kernel ridge regression [2].

The standard k-NN classifier is a simple way to classify the given similarities: a test sample x is classified as the most frequent class within a neighborhood of its k most-similar samples chosen from n training samples. Weighting the nearest neighbors with weights that reward both the diversity of the neighbors and their similarities to the test sample has been shown to offer significant reductions in error [2].

Another approach to preserving the similarities is to use

$$\hat{P}(x|g,k) = \frac{\left(\frac{2k}{k+1}\right)^{\frac{d}{2}} \Gamma\left(\frac{k+d+4}{2}\right) \left|\sum_{i=1}^{n}(x_i-\bar{x}_g)(x_i-\bar{x}_g)^T I_{(y_i=g)} + B_g\right|^{\frac{k+d+3}{2}}}{\Gamma\left(\frac{k+4}{2}\right) \left|\sum_{i=1}^{n}(x_i-\bar{x}_g)(x_i-\bar{x}_g)^T I_{(y_i=g)} + \frac{k(x-\bar{x}_g)(x-\bar{x}_g)^T}{k+1} + B_g\right|^{\frac{k+d+4}{2}}} \quad (1)$$

the n similarities to n training samples as features, and apply standard metric learning to that feature space. Training generative classifiers on this space poses difficulties in parameter estimation due to the size of the space growing with n, and researchers have shown on a small set of data that regularized QDA performs about the same as 1-NN unless there is high noise [17].

3.1 Review of Local SDA

Similarity discriminant analysis (SDA) is a recently proposed generative framework for similarity-based classification [18]. The local SDA classifier applies SDA locally to the test sample, reducing the model bias inherent in the standard SDA approach [3]. Local SDA is competitive with other state-of-the-art similarity-based classifiers [2]. First we describe local SDA, then note why its estimation variance is high, and propose a regularization solution in Section 4.

Let the test and training samples belong to an abstract space of possible samples $\mathcal{B}$, such as the set of all web pages. Let $X \in \mathcal{B}$ be a random test sample with random class label $Y \in \mathcal{G}$ where $\mathcal{G} = \{1, 2, \ldots, G\}$, and let $x \in \mathcal{B}$ denote the realization of X. Consider a similarity function $s : \mathcal{B} \times \mathcal{B} \rightarrow \Omega$, where $\Omega \subset \mathbb{R}$ is a finite discrete space. Let $\mathcal{X} \subset \mathcal{B}$ be the set of n training samples, and let $\mathcal{N}(x) \subset \mathcal{X}$ be the neighborhood of x, defined as the set of x's k most similar training samples.

The local SDA classifier follows from the standard Bayes classifier by making the fundamental assumption that all the information about x's class label depends only on some similarity information from the neighborhood $\mathcal{N}(x)$. In this paper, we use the similarities between x and each class's local centroid, which seems to be the most effective of several variants previously considered [3, 19].

Given a test sample x, let the local centroid $\mu_h(x)$ for class h be the local training sample with maximum sum-similarity to its class,

$$\mu_h(x) = \arg \max_{a \in \mathcal{N}_h(x)} \sum_{z \in \mathcal{N}_h(x)} s(z, a),$$

where $\mathcal{N}_h(x) \subset \mathcal{N}(x)$ is the subset of neighbors of x from class h.

Local SDA assumes the local class-conditional probability $P(x|Y = g)$ is the product $\prod_h P(s(x, \mu_h(x))|Y = g)$, and models each probability mass function (pmf) as an exponential:

$$P(s(x, \mu_h(x))|Y = g) = \gamma_{gh} \exp[\lambda_{gh} s(x, \mu_h(x))]. \quad (2)$$

Then, the resulting minimum expected misclassification cost rule is to classify a test sample x as the class $\hat{y}$ that solves:

$$\arg\min_{f \in \mathcal{G}} \sum_{g \in \mathcal{G}} C(f, g) \prod_{h \in \mathcal{G}} \gamma_{gh} \exp[\lambda_{gh} s(x, \mu_h(x))] P(g),$$

where $C(f, g)$ is the cost of classifying as class f when the true class is g, and $P(g)$ is the a priori probability of class g.

The parameters $\{\lambda_{gh}\}$ in (2) are determined by numerical minimization under the method-of-moments constraint that the expected value of the similarity be equal to the observed average similarity:

$$\sum_{s' \in \Omega} s' P(s(x, \mu_h(x)) = s'|Y = g) = \frac{\sum_{z \in \mathcal{N}_g(x)} s(z, \mu_h(x))}{|\mathcal{N}_g(x)|}, \quad (3)$$

and the parameters $\{\gamma_{gh}\}$ are determined by normalization.

The local SDA classifier has low model bias if used with small neighborhoods, but fitting models to small neighborhoods causes high variance of the probability estimates, and this decreases the classifier's performance. In particular, if a neighborhood $\mathcal{N}(x)$ contains only one training sample from any class g, then each class-conditional pmf for that class would be a Kronecker delta with all its probability concentrated on the self-similarity of the single neighborhood training sample. Unless the test sample happened to have that exact similarity, the likelihood of the test sample's similarities would be zero for that class. This degeneracy propagates to the product $\prod_h P(s(x, \mu_h(x))|Y = g)$, which is incorrectly estimated as identically zero when the pmfs are Kronecker delta functions concentrated on different values of the similarity. Analogous difficulties can arise if there are only a few neighbors from a class, or in general when the set of local similarities is a small subset of Ω.

In practice this degeneracy problem occurs often. In fact, consider the standard practice of cross-validating the neighborhood size k. The degeneracy of the local model for small k can overshadow the effectiveness of the local SDA classifier, and the cross-validation procedure can produce a suboptimal value for k, leading to inflated error rates: the local SDA classifier never has a full opportunity to succeed for small neighborhood sizes.

In the original local SDA paper [3], the degeneracy problem was handled by reverting to a local nearest centroid classifier [20] for all the classes if any class had exactly one

neighbor. This is suboptimal for multiple reasons: degeneracies can happen even if there is more than one neighbor; for classification problems with many classes it is likely that some class will only have one neighbor in a test sample's neighborhood; reverting to a local nearest centroid classifier does not produce probability estimates; and reverting to a local nearest centroid classifier creates a conceptual and empirical discontinuity as the neighborhood size changes.

Another approach to cope with the local scarcity of samples from one or more classes is to increase the size of the neighborhood k until satisfactory estimates of each class-conditional pmf can be achieved. This approach increases the model bias because a larger k must be used.

We argue that a more effective way to cope with this problem is to regularize the local SDA classifier. Next, we describe our regularization strategy.

4 Regularized Local SDA

We propose regularizing the local SDA classifier as a way to solve the degenerate model problem described in Section 3.1 and reduce estimation variance. We considered three methods for regularizing the local SDA classifier: (i) regularizing the parameters $\{\lambda_{gh}\}$, (ii) regularizing the mean similarity constraint in formula (3), and (iii) regularizing the class-conditional pmfs $P(s(x,\mu_h(x))|Y=g)$. Based on preliminary experiments and analysis, we argue that this third option affords the best bias-variance trade-off and is most consistent with the intuition behind the SDA model.

Specifically, we propose to regularize the local class-conditional pmfs themselves by linearly combining them with the average of the local class-conditional pmfs computed from the training set. This regularization should reduce estimation variance, but also reduces model bias by enlarging the model space to include non-exponential distributions, because the average of exponentials is not exponential.

Let $z \in \mathcal{X}$ be a training sample, and let $\mathcal{N}(z) \subset \mathcal{X}$ be its neighborhood consisting of its k most similar samples from $\mathcal{X}$. Let training sample z's class h local centroid $\mu_h(z)$ be computed from $\mathcal{N}(z)$, that is

$$\mu_h(z) = \arg \max_{a \in \mathcal{N}_h(z)} \sum_{v \in \mathcal{N}_h(z)} s(v,a).$$

Let $P_z(s(v,\mu_h(z))|Y = g)$ denote the class-g conditional exponential pmf computed from z's neighborhood that solves the empirical mean constraint (3), that is

$$\sum_{s' \in \Omega} s' P(s(v,\mu_h(z)) = s'|Y=g) = \frac{\sum_{v' \in \mathcal{N}_g(z)} s(v',\mu_h(z))}{|\mathcal{N}_g(z)|}.$$

Then, to each training sample z there correspond G^2 local class-conditional pmfs $P_z(s(v,\mu_h(z))|Y = g)$ for $g = 1,\ldots,G$ and $h = 1,\ldots,G$, and repeating the process for each of the n training samples creates a set of n pmfs $\{P_z(v,\mu_h(z))|Y=g)\}$ for each of the possible G^2 choices of g and h. We average the n training local class-conditional pmfs for each choice of g and h:

$$P_{ave}(s_h|g) \triangleq \frac{1}{|\mathcal{X}|} \sum_{z \in \mathcal{X}} P_z(s(v,\mu_h(z))|Y=g). \quad (4)$$

We will use the G^2 pmfs given by (4) to regularize each joint pmf to form the a posteriori probabilities compared in classification. For notational simplicity, denote by $\mathcal{T}(x) \triangleq \{s(x,\mu_1(x)),\ldots,s(x,\mu_G(x))\}$, the set of similarities of test sample x to its local class centroids. Then the local SDA class-conditional pmf is

$$P(\mathcal{T}(x)|Y=g) \triangleq \prod_{h \in \mathcal{G}} \gamma_{gh} \exp[\lambda_{gh} s(x,\mu_h(x))],$$

and we define the regularization term to be

$$\tilde{P}(\mathcal{T}(x)|Y=g) \triangleq \prod_{h \in \mathcal{G}} P_{ave}(s_h|g).$$

Then, the proposed regularized pmf is

$$\hat{P}(\mathcal{T}(x),g) = (1-\alpha)P(\mathcal{T}(x)|Y=g)P(g) + \alpha\tilde{P}(\mathcal{T}(x)|Y=g)\tilde{P}(g), \quad (5)$$

where $\alpha = [0,1]$ is the regularization parameter, and $P(g)$ and $\tilde{P}(g)$ are the g class prior probabilities estimated from $\mathcal{N}(x)$ and from $\mathcal{X}$ respectively. The resulting *regularized local SDA* classification rule is

$$\arg\min_{f \in \mathcal{G}} \sum_{g \in \mathcal{G}} C(f,g)\hat{P}(\mathcal{T}(x),g).$$

The proposed approach of regularizing the pmfs has several desirable properties. First, any degenerate joint pmfs that arise as solutions for a local pmf $P(\mathcal{T}(x)|Y=g)$ are regularized by the smoother pmf $\tilde{P}(\mathcal{T}(x)|Y=g)$. The regularizing marginal pmf $P_{ave}(s_h|g)$ is itself smooth because it is an average of many local exponential pmfs computed from the training set, and any Kronecker delta functions that arise as solutions for any particular $P_z(s(v,\mu_h(z))|Y=g)$ are averaged with the rest of the local training pmfs.

A second desirable property of regularizing the local class-conditional pmfs is the increased modeling flexibility. In general, each $P_{ave}(s_h|g)$ is not exponential, but rather a weighted sum of class-conditional exponential functions over a discrete similarity domain. In fact, because the SDA framework allows both positive and negative values for $\{\lambda_{gh}\}$, the regularized posterior can flexibly model complicated distributions.

A third desirable property of the proposed regularization is that it regularizes using analogous quantities, retaining the intuitive local quality of the models: the test sample's local pmf is regularized by other samples' local pmfs.

5 Classifier Fusion

In this section we detail the proposed approach to fusing similarity-based and feature-based generative classifiers. Given a test sample, we denote its Euclidean features vector by x and its local similarity statistics by $\mathcal{T}(x)$. We make the fundamental assumption that the features and the similarities are statistically independent. We do not expect this independence assumption to be always true, but we hypothesize it will be effective, in part because assuming the posteriors are independent removes the risk of overfitting associated with modeling dependencies. Using the independence assumption, we write the fused minimum expected misclassification rule as the product of probabilities defined in similarity space and Euclidean space

$$\arg\min_{f\in\mathcal{G}} \sum_{g=1}^{G} C(f,g)\hat{P}(\mathcal{T}(x),g)\hat{P}(x|g,k)\hat{P}(g), \quad (6)$$

where $\hat{P}(\mathcal{T}(x),g)$ is the regularized local SDA estimate (5), $\hat{P}(x|g,k)$ is the local BDA estimate in (1), and $\hat{P}(g)$ is the g-th class prior probability estimated from the Euclidean-space, which we assume for simplicity is just $\hat{P}(g) = 1/G$.

The result is a fused regularized local SDA/local BDA classifier that produces class probability estimates with low model bias and low variance, remains well-posed even in high-dimensional Euclidean spaces, and flexibly combines similarities and Euclidean features.

6 Experiments

First, we describe experiments comparing the proposed local BDA classifier to other feature-based classifiers in Section 6.1. Then we detail experiments in Section 6.2 testing the proposed regularized local SDA and the SDA/BDA fusion.

6.1 Experiments with Local BDA

On benchmark Euclidean-features classification problems, we compared the proposed local BDA classifier to a GMM, k-NN, local nearest means [21], and a local SVM trained on each test point's k-NN termed *SVM-KNN* [16].

6.1.1 Feature-based Classifier Experiment Details

For all of the classifiers except the GMM, the only parameter cross-validated is a neighborhood size parameter k, where $k \in \{1, 2, \ldots, 20, 30, 40, \ldots, 100\}$, unless there were fewer neighbors of one class, in which case the maximum k was taken to be the maximum number of neighbors in the smallest class. For local BDA and local nearest means, the k nearest neighbors from each class are used to fit that class's model. For SVM-KNN, the k nearest neighbors from the entire training set are used to train the SVMs [16].

For the GMM, the number of mixture components was determined by cross-validation, where the maximum number of components was $c = \min_g \text{floor}(n_g/d)$, where n_g is the number of training samples from class g, and each class was modeled as a mixture with c components. The mixture weights, means, and full covariances were estimated using the EM algorithm. Occasionally, the EM algorithm produces estimated Gaussians with ill-posed covariance matrices; in these cases we regularized the covariance matrix by adding the scaled identity matrix $10^{-6}I$.

Table 1: % Test Error for Classifiers Using Euclidean Features

Dataset	k-NN	Local Nearest Means	Local BDA	SVM KNN	GMM
Letter Rec.	5.20	4.43	**3.23**	3.93	12.20
Opt. Digits	3.23	2.73	2.78	**2.67**	9.29
Pen Digits	2.77	2.29	**1.89**	2.14	13.95
Image Seg.	12.76	13.67	**10.95**	11.52	16.86
Vowel	49.78	**43.72**	44.59	49.78	62.34

All of the datasets come from the UCI Machine Learning Repository [22]. The datasets were standardized so that each feature had zero-mean and unit standard deviation. The normalizing means and standard deviations were calculated on the training sets and then applied to the test sets. We used a randomized 10-fold cross-validation: For each of 100 runs, the training dataset was randomly divided into a set with 9/10 of the data, and a set with 1/10 of the data. The 9/10 data set was used to build models for each of the choices of the parameter k or number of mixture components for the GMM, and each model was then tested on the remaining 1/10 data set. For each parameter setting, the cross-validation error is the average error on the 100 randomly drawn 1/10 datasets.

6.1.2 Local BDA Results

Results are shown in Table 1. Local BDA performs better than the GMM for all five datasets. Local BDA is the best classifier in three of the five cases, and is close to the best classifier for the remaining two datasets.

6.2 Similarity and Fusion Experiments

We used six similarity datasets that are available from the Similarity-based Learning Repository[1].

We compared to using only local SDA on the similarities, using only regularized local SDA on the similarities, the proposed fusion classifier (fusing regularized local SDA and local BDA), using k-NN only on the similarities, using k-NN only on the Euclidean features, and using a fused k-NN rule where the class label is estimated by the combined

[1] idl.ee.washington.edu/similaritylearning

majority vote of the k nearest neighbors in terms of similarity and the k nearest neighbors in the Euclidean space (some samples received two votes).

6.2.1 Data with Separate Similarity and Features

The *Internet Ads problem* is to classify the images embedded in web pages as advertisements or non-advertisements. For this dataset, the similarity and the Euclidean features describe different aspects of the data. We pruned the original dataset [22] of the samples that had missing features, leaving 2359 samples, each one consisting of three Euclidean features and 1556 binary features (1=has that feature, 0=does not have that feature). The binary features were used to compute the Tversky linear contrast model similarity [23].

6.2.2 Data with Similarity and Derived Features

In previous research into similarity-based learning [2] it has become clear that different perspectives on the same similarity data can lead to very different classification performance. In particular, treating the similarities to the n training samples as Euclidean features sometimes works quite well, and sometimes works very poorly [2]. We hypothesize that fusing different perspectives of the similarity data could lead to better classification, or automatically sort out the best way to use the similarity data. To test this hypothesis we took five datasets for which we only have similarity data, and we computed an n-dimensional Euclidean feature vector for a sample where the ith dimension is the similarity between x and the ith training sample: $s(x, x_i)$. Then we attempted to fuse this data using the proposed local SDA/BDA classifier. Clearly the independence assumption is violated here because the Euclidean features are derived from the given similarities, which made this a particularly intriguing experiment.

The *Amazon problem* is to classify books as fiction or non-fiction, where the similarity between two books is the symmetrization of the percentage of customers who bought one book after viewing the other book. There are 96 samples in this dataset, 36 from class Non-fiction, and 60 from class Fiction. This similarity function strongly violates the triangle inequality. This dataset is also especially interesting because this similarity strongly violates the minimality property that says a sample should be maximally similar to itself, because customers often buy a different book if they first view a poorly-reviewed book.

The *Aural Sonar problem* is to distinguish 50 target sonar signals from 50 clutter sonar signals. Listeners perceptually evaluated the similarity between two sonar signals on a scale from 1 to 5. The pairwise similarities are the sum of the evaluations from two listeners, resulting in a perceptual similarity from 2 to 10 [24]. This dataset is interesting because often similarities stemming from human judgement are non-metric.

The *Patrol problem* is to classify 241 people into one of 8 patrol units based on who people claimed was in their unit when asked to name five people in their unit [25]. Like the Amazon dataset, this is a sparse dataset, as most of the similarities are zero.

The *Protein problem* is to classify 213 proteins into one of four protein classes based on a sequence-alignment similarity [26]. This problem is very well-suited to treating similarities as features because many of the first class proteins are consistently more similar to proteins from the second class.

The *Voting problem* is to classify 435 representatives into two political parties based on their votes [22]. The categorical feature vector of yes/no/abstain votes was converted into pairwise similarities using the value difference metric, which is a (dis)similarity designed to be useful for classification [27]. This similarity is almost metric.

6.2.3 Similarity Experiment Details

Each dataset was partitioned 20 times into disjoint benchmark partitions of 80% training samples and 20% test samples. For each of the 20 partitions of each dataset we chose parameters using ten-fold cross-validation for each of the classifiers. Cross-validation parameter sets were based on recommendations in previously published papers and popular usage. The choice of neighborhood size was $k \in \{1, 2, 3, \ldots, 16, 32, 64, \min(n, 128)\}$, for all classifiers, except for local BDA, for which the neighborhood was defined by the k nearest neighbors from each class, and k was limited by the size of the smallest class. The k for each similarity-based classifier and its paired Euclidean-based classifier was cross-validated. For regularized local SDA, the choices for the convex regularizing parameter were $\alpha \in \{10^{-6}, 10^{-3}, 0.01, 0.1, 0.5, 0.9\}$, and the SDA parameters $\{\lambda_{gh}\}$ were computed with the Nelder-Mead optimizer[2]. We considered binary classification costs $C(f, g)$, thus adopting the standard maximum a posteriori rule to classify each test sample.

6.2.4 Regularized Local SDA Results

Results comparing local SDA to the proposed regularized local SDA are given in Table 2 for six similarity datasets. The results show that the regularization helps for every one of the six datasets.

6.2.5 Fusion Results

The fusion results are shown in Table 2. The hypothesis that fusing different information in the form of similarities and Euclidean features should lower classification error was tested with the Internet Ads dataset. Here, the results show that the proposed fused generative classifier decreased error by 27% compared to either classifier alone. In contrast, fusing the two pieces of information with k-NN actually increased error by 16% over using k-NN only on the Euclidean features. In addition, the fused generative classifier made only $1/3$ the errors of the fused k-NN.

[2] We used the `fminsearch()` function in MATLAB

Our second hypothesis is that fusing different perspectives on the similarity can increase classification performance compared to any single perspective. This hypothesis is more tenuous because of the obvious violation of the independence assumption. In fact, fusing different perspectives only reduces error on the Voting dataset, by 5% compared to using regularized local SDA only.

However, in practice it is difficult to choose a classifier to use a priori. Cross-validation rates cannot be effectively compared across classifiers because of differences in model flexibility and overfitting. Thus it would be useful if a fused classifier always provided at least as good results as the best single classifier, because one cannot know a priori which will be the best classifier. In fact, for Amazon, Patrol, and Voting the regularized local SDA is better, and for Aural Sonar and Protein the local BDA is better. The fused local SDA/BDA is as good or better than the best single classifier for three of the five datasets (Amazon, Protein, Voting), diminishes performance by a small amount for Aural Sonar compared to using the best single classifier, and for Patrol it is robust to the inclusion of the very poor Euclidean features. In comparison, the fused k-NN is surprisingly never better than the single best k-NN classifier, and actually seems to lean towards the worse k-NN classifier rather than the better.

An important difference between the fused k-NN classifier and the fused generative classifier is the weighting of the different information. With the fused k-NN classifier the two information sources are always weighted equally. An alternative would be to attempt to train a relative weighting parameter. This risks overfitting and constrains the relative weighting to be fixed for the entire problem rather than adjust based on a given test sample. If more information sources are to be combined, the risks of overfitting and the time and complexity of training also increase. In contrast, the fused generative classifier combines the two information sources depending on how confident they each are.

7 Conclusions and Open Questions

We proposed a classification framework for fusing similarities and Euclidean features using generative classifiers. We introduced three technical contributions: (i) local BDA, a generative classifier for Euclidean features that advances the state of the art in metric learning; (ii) regularized local SDA, a generative similarity-based classifier that advances the state of the art in similarity-based classification; (iii) a naive Bayes framework for fusing local BDA and regularized local SDA. Our experiments on six real-world benchmark datasets show that our approach successfully classifies data where the similarities and the Euclidean features represent complementary information about a problem and where the similarities and features represent two different perspective about the same data.

The SDA framework can seamlessly accommodate arbitrary similarities and dissimilarities and makes it straightforward to extend our proposed fusion framework to any number of similarities. Several complementary Euclidean feature representations of the data may also be fused together with multiple local BDA classifiers. While comparing its effectiveness to other techniques such as multiple kernel learning is an open area of research, the proposed fusion approach can enable the development and extension of practical systems that integrate diverse components.

References

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer-Verlag, New York, 2001.

[2] Y. Chen, E. K. Garcia, M. R. Gupta, A. Rahimi, and L. Cazzanti, "Similarity-based classification: Concepts and algorithms," *Journal of Machine Learning Research*, March 2009.

[3] L. Cazzanti and M. R. Gupta, "Local similarity discriminant analysis," in *Proc. Intl. Conf. Machine Learning*, 2007.

[4] J. H. Friedman, "Regularized discriminant analysis," *Journal American Statistical Association*, vol. 84, no. 405, pp. 165–175, 1989.

[5] H. Bensmail and G. Celeux, "Regularized Gaussian discriminant analysis through eigenvalue decomposition," *Journal of the American Statistical Association*, vol. 91, pp. 1743–1748, 1996.

[6] S. Geisser, "Posterior odds for multivariate normal distributions," *Journal of the Royal Society Series B Methodological*, vol. 26, pp. 69–76, 1964.

[7] D. G. Keehn, "A note on learning for Gaussian properties," *IEEE Trans. on Information Theory*, vol. 11, pp. 126–132, 1965.

[8] S. Geisser, *Predictive Inference: An Introduction*, Chapman and Hall, New York, 1993.

[9] B. Ripley, *Pattern recognition and neural nets*, Cambridge University Press, Cambridge, 2001.

[10] S. Srivastava, M. R. Gupta, and B. A. Frigyik, "Bayesian quadratic discriminant analysis," *Journal of Machine Learning Research*, vol. 8, pp. 1277–1305, 2007.

[11] S. Srivastava and M. R. Gupta, "Distribution-based Bayesian minimum expected risk for discriminant analysis," *Proc. of the IEEE Intl. Symposium on Information Theory*, 2006.

[12] J. Laub, V. Roth, J. M. Buhmann, and K.-R. Müller, "On the information and representation of non-Euclidean pairwise data," *Pattern Recognition*, vol. 39, pp. 1815–1826, 2006.

Table 2: Percent Test Error Averaged over 20 Random Test/Train Splits.

Dataset	Amazon (2 classes)	Aural Sonar (2 classes)	Patrol (8 classes)	Protein (4 classes)	Voting (2 classes)	Internet Ads (2 classes)
Local SDA	12.11	17.75	11.77	17.44	6.38	4.92
Regularized Local SDA	**8.95**	16.25	**11.35**	16.40	5.40	4.53
Local BDA	15.79	**12.25**	42.92	**0.7**	5.92	5.28
Fused local SDA/BDA	**8.95**	12.75	18.85	**0.7**	**5.11**	**3.31**
k-NN on pairwise similarities	12.63	15.50	11.87	30.35	5.86	11.06
k-NN on Euclidean features	36.58	13.25	43.13	**0.7**	6.72	8.20
Fused k-NN	25.79	15.00	25.94	28.14	6.15	9.48

[13] Itamar Gati and Amos Tversky, "Weighting common and distinctive features in perceptual and conceptual judgments," *Cognitive Psychology*, pp. 341–370, 1984.

[14] G. Wu, E. Y. Chang, and Z. Zhang, "An analysis of transformation on non-positive semidefinite similarity matrix for kernel machines," Tech. Rep., University of California, Santa Barbara, March 2005.

[15] V. Roth, J. Laub, M. Kawanabe, and J. M. Buhmann, "Optimal cluster preserving embedding of nonmetric proximity data," *IEEE Trans. Pattern Anal. and Machine Intel.*, vol. 25, no. 12, pp. 1540–1551, Dec. 2003.

[16] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: discriminative nearest neighbor classification for visual category recognition," *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2126 – 2136, 2006.

[17] E. Pekalska, P. Paclíc, and R. P. W. Duin, "A generalized kernel approach to dissimilarity-based classification," *Journal of Machine Learning Research*, pp. 175–211, 2001.

[18] L. Cazzanti, M. R. Gupta, and A. J. Koppal, "Generative models for similarity-based classification," *Pattern Recognition*, vol. 41, no. 7, pp. 2289–2297, July 2008.

[19] L. Cazzanti, *Generative Models for Similarity-based Classification*, Ph.D. thesis, Department of Electrical Engineerng, University of Washington, 2007.

[20] Y. Mitani and Y. Hamamoto, "Classifier design based on the use of nearest neighbor samples," *Proc. of the Intl. Conf. on Pattern Recognition*, pp. 769–772, 2000.

[21] Y. Mitani and Y. Hamamoto, "A local mean-based nonparametric classifier," *Pattern Recognition Letters*, vol. 27, pp. 1151–1159, 2006.

[22] A. Asuncion and D. J. Newman, "UCI machine learning repository," 2007.

[23] Amos Tversky, "Features of similarity," *Psychological Review*, pp. 327–352, 1977.

[24] S. Philips, J. Pitton, and L. Atlas, "Perceptual feature identification for active sonar echoes," in *Proc. of the 2006 IEEE OCEANS Conf.*, 2006.

[25] J. E. Driskell and T. McDonald, "Identification of incomplete networks," *Florida Maxima Corporation Technical Report*, vol. 08-01, 2008.

[26] T. Hofmann and J.M. Buhmann, "Pairwise data clustering by deterministic annealing," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, January 1997.

[27] Craig Stanfill and David Waltz, "Toward memory-based reasoning," *Communications of the ACM*, vol. 29, no. 12, pp. 1213–1228, December 1986.