THE LAPLACIAN SPECTRAL CLASSIFIER *

Robert Jenssen[†]*, Deniz Erdogmus*[‡]*, Jose C. Principe*[‡] *and Torbjørn Eltoft*[†]

[†] University of Tromsø, N-9037 Tromsø, Norway
[‡] University of Florida, Gainesville, 32611 FL., USA

ABSTRACT

We develop a novel classifier in a kernel feature space defined by the eigenspectrum of the Laplacian data matrix. The classification cost function is derived from a distance measure between probability densities. The Laplacian data matrix is obtained based on a training set, while test data is mapped to the kernel space using the Nyström routine. In that space, the test data is classified based on the angle between the test point and the training data class means. We illustrate the performance of the new classifier on synthetic and real data.

1. INTRODUCTION

Spectral methods for multivariate data analysis are emerging as powerful tools, mostly based on their practical successes, for example in clustering [1]. Spectral methods are typically based on a kernel matrix of pairwise relationships between the samples, from which a more useful data representation can be derived by utilizing its eigenvalue decomposition, or eigenspectrum. Until recently, only those points used to calculate the kernel matrix have been possible to represent in the kernel feature space. Therefore, spectral classifiers have been slow to emerge since these have to be able to represent successively new data points in the kernel feature space. Recently, it was shown how the map new data points into the feature space by using the Nyström routine [2].

In this paper, we propose a new spectral classifier based on the Laplacian pdf distance, which is introduced as a clustering cost function in a recent paper by the current authors [3]. The Laplacian pdf distance exhibits a connection to Mercer kernel based learning theory via the Parzen window technique for density estimation. In a kernel feature space defined by the eigenspectrum of the Laplacian data matrix, this distance measures the cosine of the angle between the class mean vectors. Interestingly, in [3] it was shown that when the prior probabilities of the classes are roughly equal, minimizing the Laplacian pdf distance corresponds to minimizing the probability of error. However, if the prior probabilities are unequal, the Laplacian pdf distance will act as a risk function, emphasizing to classify correctly the least probable class.

Quite importantly, based on the Parzen method, an optimal spectral data transformation can be obtained. We propose to learn the optimal Laplacian data matrix based on a training data set. Hence, the transformation to the kernel feature space is defined by the eigenspectrum of that matrix. In the kernel space, we compute the means of the transformed training data. A test data set, which is to be classified, is mapped to the kernel space by means of the Nyström routine. Based on the Laplacian pdf distance in the kernel space, a spectral classifier is developed. The angle between a test point and the class means is computed. Thereafter, the test point is assigned to the class yielding the smallest such angle.

For the convenience of the reader, we briefly review the theory behind the Laplacian pdf distance in section 2. The material presented here is a compressed version of [3]. We only consider the two-class case, even though multiclass generalizations can easily be made. In section 3, we develop the novel Laplacian spectral classifier. Thereafter, in section 4, we present some experimental studies of the proposed method. Finally, in section 5, we make our concluding remarks.

2. THE LAPLACIAN PDF DISTANCE

2.1. Mercer kernel-based feature spaces

In Mercer kernel-based learning algorithms a nonlinear mapping is potentially performed as

$$\begin{aligned} \mathbf{\Phi} : R^d &\rightarrow \mathcal{F} \\ \mathbf{x} \rightarrow \mathbf{\Phi}(\mathbf{x}) &= [\sqrt{\lambda_1}\phi_1(\mathbf{x}), \sqrt{\lambda_2}\phi_2(\mathbf{x}), \ldots]^T, \end{aligned} \quad (1)$$

where the λ_i's and the ϕ_i's are the eigenvalues and eigenfunctions of a Mercer kernel. Hence, the data $\mathbf{x}_1, \ldots, \mathbf{x}_N \in R^d$ is mapped into $\mathbf{\Phi}(\mathbf{x}_1), \ldots, \mathbf{\Phi}(\mathbf{x}_N) \in \mathcal{F}$. The Mercer kernel computes an inner product in the feature space, that is, $k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{\Phi}(\mathbf{x}), \mathbf{\Phi}(\mathbf{y}) \rangle$ [4]. In practice, the mapping (1) is approximated based on the eigenspectrum

*THIS WORK WAS PARTIALLY SUPPORTED BY NSF GRANT ECS-0300340.

of the $(N \times N)$ kernel matrix, $\mathbf{K}$, with elements $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j),\ i,j = 1,\ldots,N$, as

$$\Phi(\mathbf{x}_i) \approx [\sqrt{\tilde{\lambda}_1} e_{1i}, \ldots, \sqrt{\tilde{\lambda}_N} e_{Ni}]^T. \quad (2)$$

where $\tilde{\lambda}_j$ is the jth eigenvalue and e_{ji} denotes the ith element of the jth eigenvector of the matrix $\mathbf{K}$.

In [2] it was shown that an estimate of the eigenfunction at a new point, $\mathbf{y}$, can be obtained by he following interpolatory formula, denoted the Nyström routine

$$\phi_j(\mathbf{y}) \approx \frac{\sqrt{N}}{\tilde{\lambda}_j} \sum_{i=1}^{N} e_{ji} k(\mathbf{y}, \mathbf{x}_i). \quad (3)$$

2.2. The Laplacian PDF distance as a kernel feature space cost function

Assume that a data set consists of two clusters. Associate the probability density function $p(\mathbf{x})$ with one of the clusters, and the density $q(\mathbf{x})$ with the other cluster. Let $f(\mathbf{x})$ be the overall probability density function of the data set. A distance measure between the two pdfs can be expressed as

$$D_L = -\log \frac{\langle p, q \rangle_f}{\sqrt{\langle p, p \rangle_f \langle q, q \rangle_f}} \geq 0. \quad (4)$$

where the f^{-1} weighted inner product between $p(\mathbf{x})$ and $q(\mathbf{x})$ is defined as $\langle p, q \rangle_f \equiv \int p(\mathbf{x}) q(\mathbf{x}) f^{-1}(\mathbf{x}) d\mathbf{x}$. By defining the two functions $h(\mathbf{x}) = f^{-\frac{1}{2}}(\mathbf{x}) p(\mathbf{x})$ and $g(\mathbf{x}) = f^{-\frac{1}{2}}(\mathbf{x}) q(\mathbf{x})$, the argument of the log in (4) can be expressed as

$$L = \frac{\int h(\mathbf{x}) g(\mathbf{x}) d\mathbf{x}}{\sqrt{\int h^2(\mathbf{x}) d\mathbf{x} \int g^2(\mathbf{x}) d\mathbf{x}}}. \quad (5)$$

The distance between the two pdfs is greater the smaller (5) is. Assume that we have available the iid training data points $\{\mathbf{x}_i\},\ i = 1,\ldots,N_1$, drawn from $p(\mathbf{x})$, which is the density of class C_1, and the iid $\{\mathbf{x}_j\},\ j = 1,\ldots,N_2$, drawn from $q(\mathbf{x})$, the density of C_2. The union of these two classes constitutes the overall data set. The relevant functions can be estimated based on the Parzen window density estimation technique as

$$\hat{h}(\mathbf{x}) = \frac{1}{N_1} \sum_{i=1}^{N_1} f^{-\frac{1}{2}}(\mathbf{x}_i) W_{\sigma_1^2}(\mathbf{x}, \mathbf{x}_i),$$

$$\hat{g}(\mathbf{x}) = \frac{1}{N_2} \sum_{j=1}^{N_2} f^{-\frac{1}{2}}(\mathbf{x}_j) W_{\sigma_2^2}(\mathbf{x}, \mathbf{x}_j),$$

and $\hat{f}(\mathbf{x}) = \frac{1}{N} \sum_{k=1}^{N} W_{\sigma^2}(\mathbf{x}, \mathbf{x}_k)$, where W is a Gaussian kernel function whose width is determined by the σ^2-parameter in each case. By inserting these estimates into (5), it was shown that it has an equivalent expression in a Mercer kernel feature space as

$$L = \frac{\langle \mathbf{m}_{1_f}, \mathbf{m}_{2_f} \rangle}{||\mathbf{m}_{1_f}|| ||\mathbf{m}_{2_f}||},$$

where $\mathbf{m}_{i_f} = \frac{1}{N_i} \sum_{l=1}^{N_i} \mathbf{\Phi}_f(\mathbf{x}_l),\ i = 1, 2$, that is, the sample mean of the ith class in feature space. The Gaussian Parzen kernel and the Mercer kernel is in fact equivalent in this case. This cost function is quite interesting. It measures the distance between the two classes in the feature space. In that space, the distance is solely based on the means of the classes. The distance is given by the cosine of the *angle* between the class mean vectors.

The mapping $\mathbf{\Phi}_f$ was shown to be determined by the eigenspectrum of the matrix $\mathbf{K}_f$. This matrix can be written as $\mathbf{K}_f = \mathbf{D}^{-\frac{1}{2}} \mathbf{K} \mathbf{D}^{-\frac{1}{2}}$. Here, $\mathbf{K}$ is the kernel matrix with elements $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = W_{(\sigma_t^2 + \sigma_s^2)}(\mathbf{x}_i, \mathbf{x}_j)$, where $\mathbf{x}_i \in C_t, \mathbf{x}_j \in C_s$, for $t, s \in \{1, 2\}$. Furthermore, $\mathbf{D} = \text{diag}(d_1, \ldots, d_N)$, where $d_i = \hat{f}(\mathbf{x}_i) = \frac{1}{N} \sum_{j=1}^{N} W_{\sigma^2}(\mathbf{x}_i, \mathbf{x}_j)$. In fact, $\mathbf{K}_f$ is the *Laplacian data matrix*.

A key point of this paper is that σ_1, σ_2 and σ, can be determined automatically from the training set by optimal Parzen kernel size selection. Thus, the matrix $\mathbf{K}_f$ can also be determined automatically, and so can the mapping to the kernel feature space.

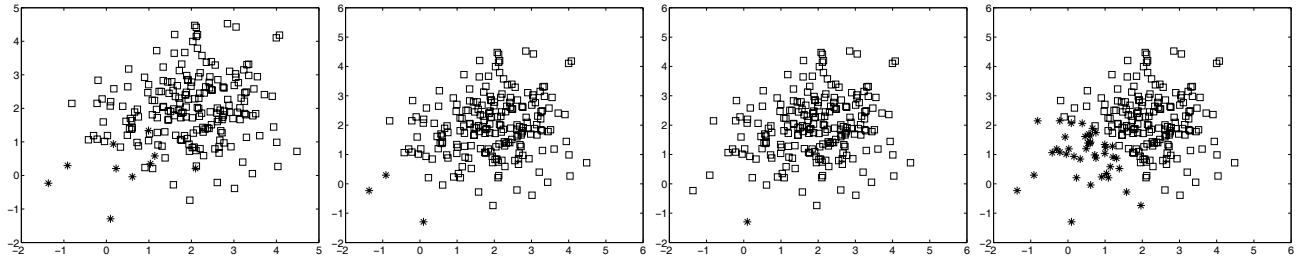
Many approaches have been proposed in order to optimally determine the size of the Parzen window, given a finite sample data set. Silverman [5] discussed this problem, using the mean integrated square error (MISE) between the estimated an the actual pdf as the optimality metric, and proposed the following formula

$$\sigma_{\text{opt}} = \sigma_X \left\{ 4N^{-1}(2d+1)^{-1} \right\}^{\frac{1}{d+4}}, \quad (6)$$

where d is the dimensionality of the data and $\sigma_X^2 = d^{-1} \sum_i \mathbf{\Sigma}_{X_{ii}}$, where $\mathbf{\Sigma}_{X_{ii}}$ are the diagonal elements of the sample covariance matrix.

3. A NOVEL SPECTRAL CLASSIFIER

In this section, we discuss a novel method for developing a spectral classifier based on the Laplacian pdf distance. We have available a labeled training data set. For each of the classes, the optimal Parzen kernel size is determined by (6). The optimal kernel size for the overall data set is also determined by the same formula. Now, the optimal data transformation into the kernel feature space can be performed by (2), after having constructed $\mathbf{K}_f$. Note that the dimensionality of the data in the kernel space equals the number of training data patterns. In that space, the training class mean vectors can be calculated, which can be used to determine



(a) Test data - labeled.

(b) Bayes classification of test data.

(c) Kernel density classification of test data.

(d) Spectral classification of test data.

Fig. 1. Result of classifying a data set consisting of two Gaussian classes with very different prior probabilities.

the distance between the classes. This is the training phase of the classifier. For a test data set, which is to be classified, one data point, $\mathbf{y}$, at a time is mapped into the feature space by (3). We use a Gaussian kernel also in (3), where the kernel size, σ, is based on the overall training data set, since we don't know which class $\mathbf{y}$ belongs to. Thereafter, the angle between $\Phi(\mathbf{y})$ and each of the training class mean vectors is computed. Finally, $\mathbf{y}$ is classified to the class for which that angle is the smallest. In summary, the proposed classifier has the following steps

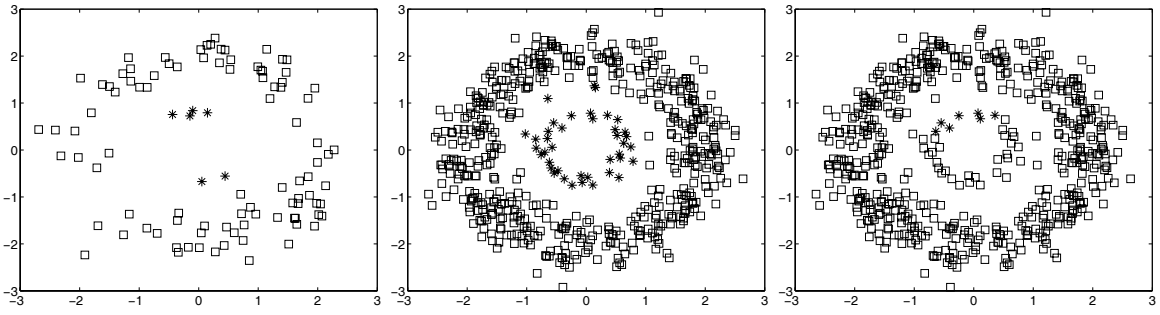
1. Determine σ_1, σ_2 and σ using (6) in each case.
2. Calculate $\mathbf{K}$, $\mathbf{D}$ and $\mathbf{K}_f = \mathbf{D}^{-\frac{1}{2}}\mathbf{K}\mathbf{D}^{-\frac{1}{2}}$.
3. Eigendecompose $\mathbf{K}_f$, and compute $\Phi(\mathbf{x}_i) \approx [\sqrt{\tilde{\lambda}_1}e_{1i}, \ldots, \sqrt{\tilde{\lambda}_N}e_{Ni}]^T, \forall i$.
4. Find $\mathbf{m}_1$ and $\mathbf{m}_2$.
5. for i = 1 : number of test points
   - Map $\mathbf{y}_i$ the the kernel space by (3).
   - Find the angle θ_1 between $\Phi(\mathbf{y}_i)$ and $\mathbf{m}_1$ and the angle θ_2 between $\Phi(\mathbf{y}_i)$ and $\mathbf{m}_2$.
   - Classify: $\mathbf{y}_i \in C_1$ if $\theta_1 < \theta_2$, else $\mathbf{y}_i \in C_2$.

4. EXPERIMENTAL RESULTS

Experiment 1. In the first classification experiment, we classify data points originating from two Gaussian distributions. The purpose is to illustrate the risk function property of the Laplacian spectral classifier. Both distributions have the same spherical covariance structure with unit variance. The mean vector of class one in the input space is $\mu_1 = [2\ 2]^T$. The mean vector of the second class is $\mu_2 = [0.6\ 0.6]^T$. The training data is constructed such that class one is represented by 100 data points, compared to only 5 data points from class two. Hence, $P_1 \approx 0.95$, while $P_2 \approx 0.05$. This means that the two clusters have overlap and that their prior probabilities are very different. Based on this training data set, the new spectral classifier is trained. For comparison, we construct a traditional Gaussian Bayes classifier. Since the covariances of the Gaussian classes are equal, this classifier produces a linear boundary between the classes. We also train a traditional Parzen kernel Bayes classifier [6], using (6) to determine the appropriate kernel size for each of the two classes. Recall that the Bayes classifier is in theory optimal with respect to the probability of error. The test data set is drawn from the same Gaussian distributions as for the training set. The data set consists of 200 data points from class one, and 10 from class two.

A scatter plot of the labeled test data set is shown in Fig. 1 (a). The squares indicate class one, and the stars class two. It can be seen that the data sets overlap, such that classification errors are unavoidable. The classification result using the Gaussian Bayes classifier is shown in Fig. 1 (b). It performs very well in terms of classification errors. It misclassifies only 7 data points. All the misclassified data points belong to class two. The Parzen kernel Bayes classifier performs worse, only detecting one of the class two data points, as shown in Fig. 1 (c). The spectral classifier obtains the result shown in Fig. 1 (d). The result is significantly different from that obtained by the Bayes classifiers. It classifies correctly 9 of the class two data points. However, it also erroneously assigns 31 class one data points to class two. One class two data point is wrongly assigned to class one. Clearly, the Laplacian spectral classifier emphasizes more to classify correctly the least probable class, i.e. the class two data points in this case. This property may be useful in many applications.

The results presented in this experiment vary somewhat depending on the training data and the test data, which is drawn at random from the Gaussian distributions. However, these differences are small, and the result presented here is representative for most cases. It should be mentioned that for $P_1 \approx P_2$, the classifiers perform almost equally good.



(a) Training data.

(b) Spectral classifi cation of test data.

(c) Kernel density classifi cation of test data.

Fig. 2. Result of classifying a data set consisting of two ring-shaped classes.

Experiment 2. The purpose of the second experiment is to show that the spectral classifier can handle highly irregular data shapes. Fig. 2 (a) shows the labeled training data set, which consists of two ring-shaped classes. There are 100 training samples, 6 from the inner-most ring and 94 from the outer-most ring. Fig. 2 (b) shows the spectral classification result for a test set consisting of 844 test samples, drawn from the same ring-shaped distributions. The classification result is nearly completely correct, for this very challenging data set. The classification result is fairly stable over repeated experiments. For comparison, Fig. 2 (c) shows the classification result using the Parzen kernel Bayes classifier. Again, it has problems with the sparse class.

Experiment 3. In this experiment, we classify a breast-cancer data set into the two classes *benign* and *malignant*. The purpose is to show that the proposed classifier also performs well on a real data set of higher dimensionality than for the previous two data sets. The Wisconsin Breast-Cancer (WBC) database is the source of this dataset, which consists of 683 data points (444 benign and 239 malignant). WBC is a nine-dimensional dataset. For the training data, 100 data points were selected at random from the data set. We performed the classification 20 times, each time selecting different training data at random. The test set consisted in each case of 583 data patterns. The average correct classification rate was 96.0%, with a standard deviation of 0.01%. The Parzen kernel Bayes classifier performs almost equally good in this case, probably because the prior probabilities are not extremely different.

5. CONCLUSIONS

We have presented a new fully automatic (no user-specified parameters) spectral classifier based on the Laplacian pdf distance. The training data set is optimally mapped to the feature space using the eigenspectrum of the Laplacian data matrix. New data points are mapped to the feature space by the Nyström routine, where they are classified based on the angle with the means of the transformed training data. The new classifier has been shown to perform well on irregular and real data. Also, it exhibits the interesting property that it emphasizes to classify correctly the least probable data points.

As for most kernel-based methods, proper kernel size selection may be problematic in very high-dimensional data spaces.

6. REFERENCES

[1] Y. Weiss, "Segmentation Using Eigenvectors: A Unifying View," in *International Conference on Computer Vision*, 1999, pp. 975–982.

[2] C. Williams and M. Seeger, "Using the Nyström Method to Speed Up Kernel Machines," in *Advances in Neural Information Processing Systems 13*, MIT Press, 2001, pp. 682–688.

[3] R. Jenssen, D. Erdogmus, J. C. Principe, and T. Eltoft, "The Laplacian PDF Distance: A Cost Function for Clustering in a Kernel Feature Space," in *Advances in Neural Information Processing Systems 17*, MIT Press, 2005.

[4] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, 1995.

[5] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, 1986.

[6] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 1999.