Roweis-Saul classifier for machine learning

P. Rajamannar* and Gurumurthi V. Ramanan**

AU-KBC Research Centre,
MIT Campus of Anna University,
Chromepet, Chennai 600 044, INDIA.

Abstract. In 2000, Saul and Roweis proposed locally linear embedding as a tool for nonlinear dimensionality reduction [1,2]. In this paper, we modify the LLE algorithm and formulate it as a classifier in a manner reminiscent of He et al [3] and name it after Roweis and Saul. Our experiments with the ORL, YALE, FERET face databases and MNIST handwritten database show that our classifier has recognition rates of 95.41%, 95.55%, 95.41% and 92.50% respectively, clearly outperforming the baseline PCA and LDA classifiers as well as the recently proposed Laplacianfaces. We propose a modification to the training phase of the classifier by perturbing the within class entries of the reconstruction matrix constructed during the training phase. This perturbation leads to an increase in the success rates for some datasets. We point out the relationship between the Roweis-Saul classifier and PCA and LDA. Various hypothesis tests have been suggested for comparing classifiers [4,5]. We apply some of the hypothesis tests suggested by Dietterich and Alpaydin to compare the Roweis-Saul classifier and the Laplacianfaces and show that the Roweis-Saul classifier outperforms the Laplacianfaces for the datasets considered here.

1 Introduction

The last few years have seen many proposals for non linear dimension reduction such as the ISOMAP, LLE, Laplacian eigenmaps, Hessian eigenmaps and GTM[6,1,2,7,8,9,10]. There are many papers where the authors propose a common framework encompassing these techniques under the ambit of spectral clustering [11]. He and Niyogi modified Belkin and Niyogi's Laplacian eigenmaps algorithm for nonlinear dimensionality reduction into a locality preserving projection [12]. All these techniques addressed the issues of finding meaningful low dimensional structures hidden in high dimensional data and analyse the data from the point of view of clustering. Recently, He et al. modified Belkin and Niyogi's Laplacian eigenmaps into a classifier [3,7].

In this paper, we modify the locally linear embedding (LLE) algorithm of Saul and Roweis in a manner which is reminiscent of He et al.'s modification of

* The results of this paper are part of the MS thesis done under the direction of the second author. Email:rajamr@au-kbc.org

** Corresponding author: gurumurthi@au-kbc.org the Laplacian eigenmaps [3]. In the context of clustering, Belkin and Niyogi in their landmark paper pointed out that the LLE algorithm of Saul and Roweis was approximated by Laplacian eigenmaps *under certain assumptions* [7]. We found that these assumptions were far too restrictive even in the context of clustering. In the case of face recognition and handwritten digits classification, these assumptions are not valid. Our experiments with some well known face and handwritten digits datasets show that our classifier outperforms Laplacianfaces.

The paper is organized as follows. Section 2 describes how Saul and Roweis' algorithm can be turned into a classifier. In Section 3 we analyse the Roweis-Saul classifier and indicate the connection with PCA and LDA. Section 4 describes a modification of the training phase of the Roweis-Saul classifier when the class information is available. In Section 5 we measure the recognition rates of the Roweis-Saul classifier and also compare it with Laplacianfaces, PCA and LDA using the ORL, YALE, FERET face databases and the MNIST handwritten digits database. In Section 6, we compare the Roweis-Saul classifier and Laplacianfaces using hypothesis tests. In Section 7, we summarize our results and point out some future directions.

2 Turning LLE into Roweis-Saul classifier

Locally linear embedding (LLE) attempts to compute a low dimensional embedding with the property that '*nearby points in the high dimensional space remain nearby and similarly co-located with respect to one another in the low dimensional space*' [2].

Let $\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_N}$, $\mathbf{x_i} \in \mathbb{R}^D$ be the feature vectors and $\mathbf{X}$ be the $D \times N$ matrix whose columns are the $\mathbf{x_i}'$s, i.e., $\mathbf{X} = [\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_N}]$. The objective of the training phase of the Roweis - Saul classifier is to find a smaller dimension $d \ll D$ and a $D \times d$ transformation $\mathbf{V}$ such that $\mathbf{Y} = \mathbf{V^T X}$, where the neighbourhood relations of the $\mathbf{y_i}$'s which are the columns of $\mathbf{Y}$, is similar to that of $\mathbf{x_i}$'s. We call $\mathbf{V}$, *the Roweis-Saul transform*. During the testing phase, the training vectors as well as the the test vector are projected on this subspace. The test vector is then classified on the basis of the euclidean distance or any other distance measure to one of the training vectors. In some datasets the number N of feature vectors is less than the dimension D of each vector. In these cases, it is not computationally feasible to find the local coordinates as specified by steps 1 and 2 of the algorithm given below. Hence, as a preprocessing step, we may reduce the dimension of the input vectors by principal component analysis (PCA), before proceeding to steps 1 and 2. Steps 1 and 2 coincide with the initial two steps of the LLE algorithm [2,1]. We now describe the training phase of the Roweis-Saul classifier.

2.1 Algorithm for training phase of Roweis-Saul classifier

Step 0: (optional) Application of PCA: For the sake of notational clarity, assume that the initial feature vectors belong to a $\widehat{D}$ dimensional space. Project the input points to a subspace along the principal components. Let $\mathbf{V_{PCA}}$be the transformation matrix obtained by performing PCA. The dimension of input points will be reduced from $\widehat{D}$ to D, where $D \ll \widehat{D}$.

Step1: Construction of the neighborhood matrix: Construct a neighborhood graph for the N points $\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_N}$. This can be done in any of the following ways.

1. *K - nearest neighbors*: Construct a graph G over all data points by connecting points i and j if i is one of the K nearest neighbors of j.
2. *ε - isomap*: Construct a graph G over all data points by connecting i and j if they are closer than ε.
3. *User defined neighbors*: Force the neighbors to be the points of the same class to which the point belongs. This variation has not been considered previously in the literature [1,2,7]

In the case of segmentation, Shi and Malik, empirically found that one can remove 90 percent of the total connections with each neighborhoods when the neighborhoods are large without affecting the eigenvector solution of the system [13].

Step 2: Computation of local linear coordinates: $\mathbf{W_{ij}}$'s are chosen so that they minimize the reconstruction error

$$\sum_{\mathbf{i}}^{\mathbf{N}} ||\mathbf{x_i} - \sum_{\mathbf{j=1}}^{\mathbf{N}} \mathbf{W_{ij}x_j}||^{\mathbf{2}} \tag{1}$$

with $\mathbf{W_{ij}} = 0$ if $\mathbf{x_j} \notin \{$neighbors of $\mathbf{x_i}\}$ subject to the constraint $\sum_j \mathbf{W_{ij}} = 1 \forall i$. The $\mathbf{W_{ij}}$'s are found using the method of of Lagrange multipliers [2].

Let $\mathbf{W_i} = (\mathbf{W_{i1}}, \mathbf{W_{i2}}, \ldots, \mathbf{W_{iN}})$ where $\mathbf{1} \leq \mathbf{i} \leq \mathbf{N}$. For every fixed $\mathbf{i}$, the Lagrange multiplier λ is given by

$$\frac{\partial \mathbf{f}(\mathbf{W_{i1}}, \mathbf{W_{i2}}, \ldots, \mathbf{W_{iN}})}{\partial \mathbf{W_{ik}}} + \lambda \frac{\partial \mathbf{g}(\mathbf{W_{i1}}, \mathbf{W_{i2}}, \ldots, \mathbf{W_{iN}})}{\partial \mathbf{W_{ik}}} = \mathbf{0} \, \forall \, \mathbf{k} = \mathbf{1}, \mathbf{2}, \ldots\mathbf{N}$$

Minimizing the equation (1) is equivalent to solving the above equation by setting $\mathbf{f}(\mathbf{W_{i1}}, \mathbf{W_{i2}}, \ldots, \mathbf{W_{iN}}) = ||\mathbf{x_i} - \mathbf{\Sigma_{j=1}^{N} W_{ij}x_j}||^{\mathbf{2}}$ and $\mathbf{g}(\mathbf{W_{i1}}, \mathbf{W_{i2}}, \ldots, \mathbf{W_{iN}})$ $= \mathbf{\Sigma_j W_{ij} - 1 = 0}$.

Step 3: Computation of low dimensional embedding: Compute the low dimensional embedding vectors $\mathbf{y_i}$ that are best reconstructed by $\mathbf{W_{ij}}$, by minimizing the equation

$$\Phi(\mathbf{y}) = \sum_{\mathbf{i}} ||\mathbf{y_i} - \sum_{j} \mathbf{W_{ij}y_j}||^{\mathbf{2}} \tag{2}$$

Substituting $\mathbf{y_i} = \mathbf{V^T x_i}$, we can rewrite the above equation as

$$= \sum_{\mathbf{i}} (\mathbf{V^T x_i} - \sum_{\mathbf{j}} \mathbf{W_{ij} V^T x_i})^{\mathbf{T}} (\mathbf{V^T x_i} - \sum_{\mathbf{j}} \mathbf{W_{ij} V^T x_i})$$

In matrix notation the above equation is

$$\Phi(\mathbf{V}) = (\mathbf{V^T X} - \mathbf{V^T XW})^{\mathbf{T}} (\mathbf{V^T X} - \mathbf{V^T XW}) \tag{3}$$

It is clear that minimizing $\Phi(\mathbf{V})$ is equivalent to minimizing the trace of the RHS of the equation (3)

$$\Phi(\mathbf{V}) = \mathit{Trace} \left\{ (\mathbf{V^T X} - \mathbf{V^T XW})^{\mathbf{T}} (\mathbf{V^T X} - \mathbf{V^T XW}) \right\} \tag{4}$$

We now show how to formulate this minimization as an algebraic eigenvalue problem. Rewrite the RHS of above equation as follows.

$$\Phi(\mathbf{V}) = \mathit{Trace} \left\{ \mathbf{V^T X (I - W)(I - W^T) X^T V} \right\} \tag{5}$$

Let $\mathbf{M} = \mathbf{X(I - W^T)}$

$$\Phi(\mathbf{V}) = \mathit{Trace} \left\{ \mathbf{V^T M M^T V} \right\} \tag{6}$$

Now minimizing equation (6) is equivalent to the algebraic eigenvalue problem

$$\mathbf{V^T M} = \lambda \mathbf{V^T} \tag{7}$$

The transformation $\mathbf{V}$ is found by first computing the eigenvectors corresponding to $(d+1)$ smallest eigenvalues of the matrix $\mathbf{M}$. Then the eigenvector corresponding to the smallest eigenvalue is discarded since it is zero. The remaining d eigenvectors form the rows of the transformation. We note that this step is different from the corresponding step in the case of Laplacianfaces [3] where the generalized eigenvalue problem is solved.

When we use PCA for reducing the dimension of input data, the Roweis-Saul transform for projecting the feature vectors is given by

$$\mathbf{V} = \mathbf{V_{PCA} V} \tag{8}$$

and the low dimensional embedding of the training vectors is given by

$$\mathbf{Y} = \mathbf{V^T X} \tag{9}$$

The test vector $\check{\mathbf{x}}$ is projected using the Roweis-Saul transform $\mathbf{V}$ and classified as belonging to the same class as $\mathbf{y_j}$, using the nearest neighbour rule.

A remark is in order. Belkin and Niyogi showed that the LLE algorithm attempts to minimize $\mathbf{f^T (I - W)^T (I - W) f}$ and *under certain conditions,* it can be interpreted as trying to find the eigenfunctions of the iterated Laplacian $\mathcal{L}^2$, where

$$\mathbf{(I - W)^T(I - W)f \approx \frac{1}{2}\mathcal{L}^2 f}.$$

These conditions imply that pairwise differences of the input feature vectors $\mathbf{x_i - x_j}$ form an orthonormal basis. They also mention in passing that this '*is not usually the case*' [7]. This condition is not valid in many supervised learning problems. We emphasize that this condition is not true for all the datasets considered in this paper. This motivated our modification of the LLE algorithm into a classifier. In the next section, we point out some connections between the Roweis-Saul classifier and the classical PCA and LDA.

3 Connections between Roweis-Saul classifier, PCA and LDA

In the training phase of the RS classifier we minimized the following equation

$$\mathbf{\sum_{i=1}^{N} ||x_i - \sum_{j=1}^{N} W_{ij} x_j||^2} \tag{10}$$

subject to the constraint that $\mathbf{\sum_{j=1}^{N} W_{ij} = 1 \,\forall i}$ using the method of Lagrange multipliers.

3.1 Connection with PCA

Let $\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N})$, where $\mathbf{x_i} \in \mathbb{R}^{\mathbf{D}}$ and $\mathbf{m} = \mathbf{\frac{1}{N}\sum_{j=1}^{N} x_j}$. If we substitute $\mathbf{W_{ij}} = \mathbf{\frac{1}{N}}\,\forall\,\mathbf{i,j}$, in equation (10), it becomes the covariance matrix of the input data. i.e.,

$$\mathbf{\sum_{i=1}^{N} ||x_i - \sum_{j=1}^{N} W_{ij} x_j||}^2 = \mathbf{\sum_{i=1}^{N} (x_i - \sum_{j=1}^{N} W_{ij} x_j)^T (x_i - \sum_{j=1}^{N} W_{ij} x_j)}$$

$$= \mathbf{\sum_{i=1}^{N} (x_i - XW)^T (x_i - XW)}.$$

Using the fact that $\mathbf{XW} = \mathbf{m}$, the RHS of the above equation can be rewritten as

$$\mathbf{\sum_{i=1}^{N} ||x_i - \sum_{j=1}^{N} W_{ij} x_j||}^2 = \mathbf{\sum_{i=1}^{N} (x_i - m)^T (x_i - m)} \tag{11}$$

The technique of PCA involves choosing the eigenvectors corresponding to the leading eigenvalues of the covariance matrix as the basis for projection of the input data [14,15]. This shows the relationship between Step 2 (finding local linear coordinates) of the training phase of the Roweis-Saul classifier and PCA. During the training phase of the Laplacianfaces, this step is absent because the linear coordinates are chosen by taking them to be the Euclidean distance or the neighbourhood relationships between the input points [3].

3.2 Connection with LDA

Let $\mathbf{C}$ be the number of classes in the given dataset and $\mathbf{c_i}$, the number of samples in each of the class, where $\mathbf{1} \leq \mathbf{i} \leq \mathbf{C}$. Let us represent $\mathbf{x_j}$ and $\mathbf{W_{jk}}$ as $\mathbf{x^i_j}$ and $\mathbf{W^i_{jk}}$ if $\mathbf{x_j} \in \mathbf{c_i}$ and where $\mathbf{1} \leq \mathbf{j}, \mathbf{k} \leq \mathbf{N}$ and $\mathbf{m^i} = \frac{\mathbf{1}}{\mathbf{c_i}} \sum_{\mathbf{j=1}}^{\mathbf{c_i}} \mathbf{x^i_j}$. The equation (10) can be rewritten as

$$\sum_{\mathbf{i=1}}^{\mathbf{N}} ||\mathbf{x_i} - \sum_{\mathbf{j=1}}^{\mathbf{N}} \mathbf{W_{ij}x_j}||^{\mathbf{2}} = \sum_{\mathbf{i=1}}^{\mathbf{C}} \sum_{\mathbf{j=1}}^{\mathbf{c_i}} ||\mathbf{x^i_j} - \sum_{\mathbf{k=1}}^{\mathbf{c_i}} \mathbf{W^i_{jk}x^i_k}||^{\mathbf{2}}$$

If $\mathbf{W^i_{jk}} = \frac{\mathbf{1}}{\mathbf{c_i}} \forall \mathbf{i}, \mathbf{j}, \mathbf{k}$ then $\sum_{\mathbf{k=1}}^{\mathbf{c_i}} \mathbf{W^i_{jk}x^i_k}$ becomes the class mean of the input samples (i.e.,) $\mathbf{m^i}$

$$= \sum_{\mathbf{i=1}}^{\mathbf{C}} \sum_{\mathbf{j=1}}^{\mathbf{c_i}} ||\mathbf{x^i_j} - \mathbf{m^i}||^{\mathbf{2}}$$

$$= \sum_{\mathbf{i=1}}^{\mathbf{C}} \sum_{\mathbf{j=1}}^{\mathbf{c_i}} (\mathbf{x^i_j} - \mathbf{m^i})^{\mathbf{T}} (\mathbf{x^i_j} - \mathbf{m^i}) \tag{12}$$

The RHS of equation (12) is the within class scatter matrix $\mathbf{S_W}$that is minimized in Linear Discriminant Analysis (LDA). In LDA, the denominator of the criterion function

$$\mathbf{J(W_i)} = \frac{\mathbf{W_i^T S_B W_i}}{\mathbf{W_i^T S_w W_i}} \tag{13}$$

is minimized in order to maximize the criterion function (13) where $\mathbf{S_B}$ is the between class scatter matrix and $\mathbf{S_w}$is the within class scatter matrix [15]. From this it is clear that during Step 2, we are minimizing an analogue (equation (10)) of the within class scatter matrix in order to find the local coordinates.

4 Roweis-Saul with an α perturbation

In this section, we propose a modification of the Roweis-Saul classifier described in the previous section. First we assume that the input data consists of a labelled training set and is ordered using the class information. In Step 3 of training phase, described in (2.1), we add a matrix $\mathbf{W}_\alpha$ to $\mathbf{W}$, *i.e.,*

$$\mathbf{W}_{\mathbf{RS}\alpha} = \mathbf{W}_\alpha + \mathbf{W}. \tag{14}$$

We identify the matrix $\mathbf{W}_\alpha$ in the above equation (14) with the matrix whose rows and columns are indexed by the input data points. The $(i, j)^{\text{th}}$ entry of $\mathbf{W}_\alpha$ is α if the i^{th} and the j^{th} points belong to the same class and $1 - \alpha$ if they belong to different classes. Hence equation (4) becomes

$$\Phi(\mathbf{V}) = Trace\left\{\mathbf{V}^{\mathbf{T}}\mathbf{X}(\mathbf{I} - \mathbf{W}_{\mathbf{RS}\alpha})(\mathbf{I} - \mathbf{W}^{\mathbf{T}}_{\mathbf{RS}\alpha})\mathbf{X}^{\mathbf{T}}\mathbf{V}\right\}.$$

Thereafter, we follow the steps mentioned in (2.1) of the Roweis-Saul training phase to compute the embedding. This matrix is motivated by our desire to bring the within class points closer and can be viewed as a perturbation of the reconstruction matrix constructed during Step 3 of the training phase. We choose α between 1.0 and 0.0, and the typical value used in this paper is 0.9. [Table 2].

Experiments with the databases using the RS classifier with α graph perturbation show that it has lower error rate when compared to the RS classifier and Laplacianfaces. Since this variation makes use of the class information of the dataset, it can be considered as a supervised learning analogue of the Roweis-Saul classifier. A similar modification when applied to the Laplacianfaces, also yields a similar boost in the performance [Table 2]. This phenomenon is interesting and needs to be validated for larger and more diverse datasets than considered here.

5 Experiments

5.1 Experiments with face databases

The YALE face database, ORL face database and the FERET database were used to measure the recognition rates of the Roweis-Saul classifier and compared with the Laplacianfaces [16,17,18]. He et al. had downsampled the images to 32 × 32 before performing classification [3]. In our experiments we used three different image sizes, namely 32 × 32, 64 × 64 and 128 × 128. In the case of the FERET database, we manually selected 106 people each having 4 frontal images. In all these cases we have set the number of neighbors as$(K =)\,4$.

Table 1. Face databases used in the experiments

Database	No. of people	No. of images per person
ORL	40	10
YALE	15	11
FERET	106	4

Table 2. Recognition rates for Roweis-Saul, Laplacianfaces and their variations

Dataset	Size	RS	RS α	Laplacian	Laplacian α	PCA	LDA
ORL 4 training	32	95.41%	95.41%	91.66%	92.50%	87.91%	93.33%
	64	94.58%	95.41%	91.25%	91.66%	87.08%	92.92%
	128	94.58%	95.41%	91.25%	91.66%	87.08%	92.50%
YALE 5 training	32	95.55%	96.66%	90.00%	94.44%	87.78%	95.56%
	64	94.44%	95.55%	88.89%	93.33%	84.44%	94.44%
	128	94.44%	95.55%	88.89%	91.11%	84.44%	93.33%
FERET 2 training	32	95.41%	95.41%	91.66%	92.50%	87.91%	92.22%
	64	94.58%	95.41%	91.25%	91.66	87.08%	92.22%
	128	92.92%	95.41%	91.25%	91.66	87.08%	92.92%

Fig. 1. Recognition rates for the RS classifier and Laplacianfaces with increasing number of training samples for ORL and YALE databases.

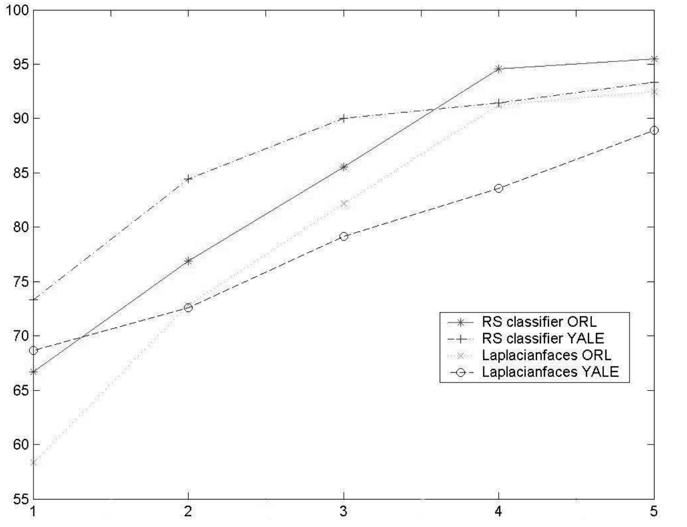

Table-1 gives information about the databases used for our experiments. Figure-1 shows how the recognition rates of the RS classifier and Laplacianfaces improve when the number of training samples increase. Table-2 compares the recognition rates of the Roweis-Saul classifier and Laplacianfaces, PCA and LDA for different image sizes. We observe that recognition rates decrease with increasing image size.

5.2 Experiments with handwritten digit database

The MNIST handwritten digit database was used to compare the recognition rates of RS classifier and Laplacianfaces. The MNIST handwritten digit database has 60,000 training digits and 10,000 testing digits each of size 28X28. The authors of the training and testing set were disjoint. We took 500 training images for each digit and considered the entire testing set for classification. The results summarized in the Table-3 shows that RS classifier has low error rate when compared to the Laplacianfaces.

Table 3. Recognition rates for MNIST handwritten digit database

Classifier	Recognition rate
RS classifier with α graph	93.50%
RS classifier	92.50%
Laplacianfaces with α graph	91.50%
Laplacianfaces	90.00%
PCA	87.50%
LDA	89.00%

6 Comparing classifiers using hypothesis tests

In this section, we apply McNemar's test, K- fold cross validated paired test, 5×2 cross validated paired t test and 5×2 cross validated paired F test to compare the classification rates of Roweis-Saul classifier and Laplacianfaces. The results of our experiments with the ORL face database show that RS classifier has lower error rate when compared to Laplacianfaces. The comparisons for the YALE face database indicate that they both have same error rate. For a detailed introduction to these tests we refer the reader to the paper by Dietterich and the recent book by Alpaydin and the references therein [4,5].

6.1 McNemar's test for comparing classifiers

We performed McNemar's test for all the three databases mentioned previously. For the ORL database, 4 images per person were used for training and 6 images per person for testing. The YALE database was tested with 5 images per person in training and the remaining for testing. The FERET database was tested using 2 images per person for training and the remaining 2 images for testing. This test takes into account the number of samples misclassified by the Laplacianfaces when compared to the Roweis-Saul classifier and vice-versa. The null hypothesis (H_0) is that both the classifiers have equal error rate *i.e.* $e_{01} = e_{10}$.

e_{00} : No. of samples misclassified by both classifiers
e_{01} : No. of samples misclassified by Laplacianfaces but not Roweis-Saul classifier
e_{10} : No. of samples misclassified by Roweis-Saul classifier but not Laplacianfaces
e_{11} : No. of samples correctly classified by both classifiers

The values of $\mathbf{e_{ij}}$'s and the results are available in Table-4. We have the chi-square statistic with one degree of freedom $\chi^2 = \frac{(|e_{01}-e_{10}|-1)^2}{e_{01}+e_{10}}$. McNemar's test accepts the null hypothesis at significance level α if $\chi^2 \leq \chi^2_{\alpha,1}$. The typical value for α is 0.05 and $\chi^2_{0.05,1} = 3.84$. Since the null hypothesis is rejected and e_{01} is greater than e_{10} for ORL and FERET database, we conclude that the Roweis-Saul classifier outperforms the Laplacianfaces for these databases. Even though e_{01} is greater than e_{10} for the Yale database, the χ^2 value is in the acceptable range. Hence, we conclude that both the classifiers have same error rate. The results of McNemar's test are shown in Table-4.

Table 4. Results of McNemar's test

Database	e_{00}	e_{01}	e_{10}	e_{11}	χ	Null hypothesis
ORL	11	8	2	219	4.9	REJECT
FERET	14	9	1	188	5.6	REJECT
Yale	4	5	1	80	1.5	ACCEPT

6.2 *K* - fold cross validated paired t test

In the K-fold cross validated paired test the dataset is divided randomly into K equal parts [5]. To get the training and testing set, we combine $K - 1$ parts to form the training set and the remaining part for testing. In the subsequent round, we take another one of K parts for testing and combine the remaining $K-1$ parts for training. We perform this experiment till all the K parts have been tested. We record the error rates p_i^1 for the first classifier and p_i^2 for the second classifier, where $1 \leq i \leq K$ and calculate the difference in error rates $p_i = p_i^1 - p_i^2$. The null hypothesis H_0 assumed here is that mean μ of the distribution of difference in error rates is zero. We perform a t test to check whether the mean of the distribution of difference in error rates falls in the acceptable range. If it does not fall in the acceptable range, we reject the null hypothesis. The t test statistic with $K-1$ degrees of freedom is given by $t_{K-1} = \frac{\sqrt{K}m}{S}$, where $m = \frac{\sum_{i=1}^{K} p_i}{K}$ and $S = \frac{\sum_{i=1}^{K}(p_i - m)^2}{K-1}$.

The K-fold cross validated paired t test accepts the null hypothesis that both the classifiers have same error rate at significance level α if t_{K-1} is in the interval $(-t_{\alpha/2,K-1}, t_{\alpha/2,K-1})$. If $t_{K-1} < -t_{\alpha/2,K-1}$, the first classifier has a higher error rate when compared to the second classifier. If $t_{K-1} > t_{\alpha/2,K-1}$, the second classifier has a higher error rate when compared to the first one.

The results of this test for the ORL and YALE face datasets are summarized in Table-5. Since the null hypothesis is rejected for ORL database and it exceeds $t_{\alpha/2,K-1}$, we conclude that Laplacianfaces has a higher error rate when compared to RS classifier. Since the t_{K-1}value of YALE database is in the acceptable range, we accept the null hypothesis and conclude that both Laplacianfaces and RS classifier have the same error rates.

Table 5. K **- fold cv paired t test with** $K = \mathbf{10}$ **&** $\alpha = \mathbf{0.05}$

Database	$-t_{\alpha/2,K-1}$	t_{K-1}	$t_{\alpha/2,K-1}$	Null Hypothesis
ORL	-2.26	2.5038	2.26	REJECT
YALE	-2.26	1.0437	2.26	ACCEPT

6.3 5×2 cross validated paired t test

In 5×2 cv paired t test proposed by Dietterich, we divide the dataset randomly into two equal parts, to get the first pair of training and testing set [4]. Then we swap the role of training and testing sets. To get the second pair we shuffle the dataset and again divide it randomly into two equal parts. We repeat this for three more pairs, and get ten training and testing sets. In all, we perform five replications of twofold cross-validation. Even though it is possible to get more training and testing pairs, Dietterich points out that after five folds, the training and testing sets overlap and the error rates calculated becomes dependent [4].

Let $p_i^{(j)}$ be the difference between the error rates of the two classifiers on the fold $j = 1, 2$ of replication $i = 1, \ldots, 5$. Let the average on the replication i be $\overline{p_i} = (p_i^{(1)} + p_i^{(2)})/2$, and the variance be $s_i^2 = (p_i^{(1)} - \overline{p_i})^2/(p_i^{(2)} - \overline{p_i})^2$.

The null hypothesis is that the two classification algorithms have the same error rate $p_i^{(j)}$. Here $p_i^{(j)}$ can be treated as approximately normal distributed with 0 mean and unknown variance σ^2. Then $p_i^{(j)}/\sigma$ is approximately unit normal. If we assume $p_i^{(1)}$ and $p_i^{(2)}$ are independent normals, then s_i^2/σ^2 has a χ^2 distribution with one degree of freedom. If each of s_i^2 are independent, then their sum is χ^2 with five degrees of freedom.

$M = \frac{\sum_{i=1}^{5} s_i^2}{\sigma^2} \sim \chi_5^2$ and

$$t = \frac{p_1^{(1)}}{\sqrt{M/5}} = \frac{p_1^{(1)}}{\sqrt{\sum_{i=1}^{5} s_i^2/5}} \sim t_5 \tag{15}$$

The above equation is a t statistic with five degrees of freedom. We accept the null hypothesis that both the classifiers have the same error rate at significance level α if this value is in the interval $(-t_{\alpha/2,5}, t_{\alpha/2,5})$. $t_{0.025,5} = 2.57$. Since the t value for ORL face database is not in the acceptable range, the null hypothesis is rejected and we conclude that Laplacianfaces has more error rate when compared to RS classifier. Since the t value of YALE database is in the acceptable range we accept the null hypothesis and conclude that both Laplacianfaces and RS classifier have same error rates. The results are summarized in Table-6.

Table 6. 5×2 **cv paired t test with** $\alpha = \mathbf{0.05}$

Database	ORL	YALE
$\overline{p_1}$	5.00	1.33
$\overline{p_2}$	2.25	2.33
$\overline{p_3}$	2.75	1.00
$\overline{p_4}$	6.25	2.00
$\overline{p_5}$	3.25	4.00
s_1^2	0.5000	0.8712
s_2^2	3.3125	0.2178
s_3^2	3.125	2.0000
s_4^2	3.125	0.8712
s_5^2	1.125	8.0000
t	3.6770	1.2932
$t_{0.025,5}$	2.57	2.57
Null Hypothesis	REJECT	ACCEPT

6.4 5×2 cross validated paired F test

The numerator in equation (15) is arbitrary and can take ten different values namely $p_i^{(j)}, j = 1, 2, i = 1, \ldots, 5$ and thus we get ten different t statistics.

$$t_i^{(j)} = \frac{p_i^{(j)}}{\sqrt{\sum_{i=1}^{5} s_i^2/5}}$$

Alpaydin proposed an extension to the 5×2 cross validated t test by considering the ten possible statistics. If $p_i^{(j)}/\sigma \sim Z$, then $(p_i^{(j)})^2/\sigma^2 \sim \chi^2$ and their sum is chi-square with ten degrees of freedom.

$$N = \frac{\sum_{i=1}^{5} \sum_{j=1}^{2} (p_i^{(j)})^2}{\sigma^2} = \chi_{10}^2$$

If we replace the numerator of equation (15) by the above equation, the resulting statistic is the ratio of two chi-square distributed random variables. Alpaydin notes that *'two such variables divided by their corresponding degrees of freedom is F distributed with ten and five degrees of freedom'* [5].

$$f = \frac{N/10}{M/5} = \frac{\sum_{i=1}^{5} \sum_{j=1}^{2} (p_i^{(j)})^2}{2 \sum_{i=1}^{5} s_i^2} \sim F_{10,5}$$

This test accepts the null hypothesis that both the classifiers have the same error rate for a significance level α if the value is less than $F_{\alpha,10,5}$. $F_{0.05,10,5} = 4.74$.

Since the f value of ORL face database is not less than $F_{\alpha,10,5}$, we reject the null hypothesis and we conclude that Laplacianfaces has more error rate when compared to RS classifier for ORL. Since the f value of YALE database is less than $F_{\alpha,10,5}$, we accept the null hypothesis and conclude that both Laplacianfaces and RS classifier has same error rates for YALE face database. The results are summarized in Table-7.

Table 7. **5×2 cv paired F test α = 0.05**

Database	N	M	f	$F_{\alpha,10,5}$	Null Hypothesis
ORL	188.50	11.5625	8.4246	4.74	REJECT
YALE	68.3824	11.9602	2.8587	4.74	ACCEPT

7 Conclusion

We modified the LLE algorithm for nonlinear dimensionality reduction and formulated it as a classifier. We measured the performance of this classifier using the ORL, YALE and FERET face databases and the MNIST handwritten digit database against classifiers such as PCA, LDA and the related Laplacianfaces and found that it outperformed them. We also indicated the relationships between the Roweis-Saul classifier and PCA and LDA. We modified the training phase of the classifier by perturbing the within class entries of the reconstruction matrix constructed during the training phase. We proved that this perturbation leads to a small increase in the success rates for some datasets. We used the hypothesis tests suggested by Dietterich and Alpaydin to compare the Roweis-Saul classifier and the Laplacianfaces. The results from these tests seem to indicate that Roweis-Saul performs better than the Laplacianfaces for most cases considered in this paper. Since the datasets considered in this paper consist of images, the results of this paper need to be validated for more diverse datasets. Classifiers such as PCA have been viewed from other perspectives such as reconstruction, compression and ease of computation. Bengio et al touch upon some computational issues as well as a generalized framework in the case of LLE [11]. A similar understanding of the Roweis-Saul classifier and its variation is required for applying it to larger and more diverse datasets.

References

1. Saul, L.K., Roweis, S.T.: Think globally, fit locally: Unsupervised learning of low dimensional manifold. Journal of Machine Learning Research **4** (2003) 119–155
2. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by local linear embedding. Science **290** (2000) 2323–2326
3. He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H.: Face recognition using Laplacianfaces. IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005) 1–13
4. Dietterich, T.G.: Approximate statistical test for comparing supervised classification learning algorithms. Neural Computation **10** (1998) 1895–1923
5. Alpaydin, E.: Introduction to Machine Learning. The MIT Press (2004)
6. Tenenbaum, J., Desilva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science **290** (2000) 2319–2323
7. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. **15** (2003) 1373–1396
8. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In Dietterich, T.G., Becker, S., Ghahramani, Z., eds.: Advances in Neural Information Processing Systems 14, Cambridge, MA, MIT Press (2002)
9. Donoho, D., Grimes, C.: Hessian eigenmaps: new locally linear embedding techniques for highdimensional data. (2003)
10. Bishop, C.M., Svensen, M., Williams, C.K.I.: GTM: The generative topographic mapping. Neural Computation **10** (1998) 215–234
11. Bengio, Y., Paiement, J., Vincent, P., Delalleau, O., Le Roux, N., Ouimet, M.: Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering. (2004)
12. He, X., Niyogi, P.: Locality preserving projections (2002)
13. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22** (2000) 888–905
14. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86
15. Duda, R.O., E.Hart, P., Stork, D.G.: Pattern Classification. John Wiley Sons, Inc (2002)
16. Yale University: (YALE face database)
17. AT&TLaboratories: (ORL face database)
18. Phillips, P., Wechsler, H., Huang, J., Rauss, P.: The FERET database and evaluation procedure for face recognition algorithms. Image Vision Comput. **16** (1998) 295–306