Linear Dimensionality Reduction for Classification

Ilya Blayvas, Michael Elad, and Ron Kimmel
{blayvas, ron, elad}@cs.technion.ac.il
CS Dept. Technion, Haifa, Israel, 32000
October 28, 2004

Abstract

We consider the problem of dimensionality reduction for the sake of classification. Dimensionality reduction can become a crucial step in classification of high-dimensional data. Usually, the number of training points is limited and can even be smaller than the dimensionality of the data. We show that straightforward pursuing of optimality criteria can lead to ‘overfitting’ and suboptimal performance. We show how regularization can be introduced to obtain better separation between classes in the reduced space and higher classification performance.

1 Introduction

The problem of dimensionality reduction had attracted substantial attention in the last decade [1], [2]. Dimensionality reduction can be considered as a preprocessing for the sake of visualization, regression estimation, or classification. In this paper, we focus only on the classification problem, although the results are applicable to other domains.

The examples of domains, where dimensionality reduction is desirable, include document classification and image recognition. In document classification the document is
represented as a vector in the space, spanned by tens of thousands of words from the
dictionary. In image recognition, an image is represented as a vector in the space which
dimensionality equals to the number of pixels in the image (up to millions).

In both examples and many others, the dimension of the data to be classified is high,
and the number of examples to train on is often substantially smaller. These are the types
of problems we refer to in this work, following [2, 3].

In this paper we outline a framework for dimensionality reduction and show a similarity
between dimensionality reduction and classification. It is known in learning theory [4]
that if the VC dimension of the classifier is too large relatively to the number of training
points, then the classification performance can be suboptimal. The problem is cured
via regularization, which is equivalent to reduction of the VC dimension. Similarly, if the
cardinality of the set of possible dimensionality reductions is too large relatively to the
size of the training set, the optimal reduction will perform better on the training set than
on the test set. We call that overfitting in dimensionality reduction, and demonstrate, how
the performance on the test set can be improved by regularization.

The rest of this paper is organized as following. In Section 2 we present a generic
framework for dimensionality reduction, which will allow us to specify the domain we
are dealing with. It also introduces the notation. Section 3 presents the linear discriminant
analysis for dimensionality reduction and reviews generalized singular value decomposi-
tion, used in the case when the dimensionality exceeds the number of data points. Section
4 discusses the ‘overfitting’ of conventional solution of LDA and proposes a regularized
LDA as a remedy. Section 5 presents some experimental results, illustrating the above
analysis while Section 6 summarizes the paper.
2 Notation, Assumptions and Framework

Assume there are \( n \) data points, \( \{x_1, \ldots, x_n\} \), \( x_i \in \mathbb{R}^D \), belonging to \( K \) different classes \( C_1, \ldots, C_K \) forming a data matrix \( A \in \mathbb{R}^{D \times n} \):

\[
A = [x_1, \ldots, x_n].
\] (1)

The number of points in class \( k \) is denoted by \( n_k \), therefore \( n_1 + \ldots + n_K = n \). The points are assumed to be pre-ordered so that the data matrix \( A = [x_1, \ldots, x_n] \) is partitioned into the \( K \) classes:

\[
A = \underbrace{[x_1, \ldots, x_{n_1}]}_{A_1}, \ldots, \underbrace{[x_{n_{K-1}}, \ldots, x_n]}_{A_K} = [A_1, A_2, \ldots, A_K].
\] (2)

In the general case, dimensionality reduction is a non-linear mapping:

\[
\mathbb{R}^D \xrightarrow{M} \mathbb{R}^d, \quad d < D.
\] (3)

Thus, the data matrix is transformed as

\[
\tilde{A} = [\tilde{x}_1, \ldots, \tilde{x}_n],
\] (4)

\[
\tilde{x}_i = M(x_i); \quad x_i \in \mathbb{R}^D, \quad \tilde{x}_i \in \mathbb{R}^d.
\] (5)

For clarity of presentation, we assume that the data points are in ‘general position’. This means, that for \( n \) data points, \( n \leq D + 1 \), \( \{x_1, \ldots, x_n\} \), \( n \) vectors \( \{v_1 = x_1 - m, \ldots, v_n = x_n - m\} \), where \( m = \frac{1}{n} \sum_{i=1}^{n} x_i \) span \( n - 1 \) dimensional space.

Such assumption is well founded. Consider a \( D \) dimensional cube with \( j \) distinguishable discrete positions along each coordinate. The volume of such cube is \( j^D \) voxels. \( m \) points span \( m - 1 \) dimensional space, of volume \( j^{m-1} \) voxels. Therefore, the probability that point \( m + 1 \) will fall onto this linear subspace, and not onto the rest of the \( D \) dimensional cube is \( \frac{j^{m-1}}{j^D} = j^{m-1-D} \). In our case \( j = 256 \), and any \( D > m \) results in vanishingly small probability of linear dependence.

The search for an optimal dimensionality reduction can be considered as a 3 step process:
1. Define a set of possible mappings \( \{ M_{\alpha} \} \).

2. Define an optimality criteria, ‘cost’ \( C(M, A) \), as a function of mapping \( M \) for a given data matrix \( A \).

3. Define a procedure, looking for optimal (or approximately optimal) mapping

\[
M_{opt} = \arg \min_{M \in \{ M_{\alpha} \}} C(M, A).
\]

Note, that the same three steps describe 2-class classification process, with the only modification that dimensionality of the reduced space is one, \( \tilde{x} \in \mathbb{R}^1 \) and the sign of the projection in the output space \( \text{sign} (\tilde{x}(x)) \) is taken to predict a class.

Obviously, the power of arbitrary mapping \( M \) allows transformation of arbitrary set of non-coinciding points \((i \neq j \rightarrow x_i \neq x_j)\) into any desirable configuration. However, there is no guarantee that the new point \( \tilde{x} = M(x) \) will be also mapped to the desired position. Thus, similarly to overfitting in classification, we must be aware of overfitting in dimensionality reduction, and we must account for the cardinality of the set of possible mappings \( |\{ M_{\alpha} \}| \), similarly to accounting for the VC dimension (which equals to \( \log_2 \) of cardinality of the set of classifiers) of the classifier [4].

3 Linear Discriminant Analysis

Among many ways to define the set of possible mappings \( \{ M_{\alpha} \} \), restricting search to linear dimensionality reductions is appealing, because of its simplicity, and control on the cardinality of the set. This way we leave the treatment of non-linear dependencies in the data to the classification stage, and look for dimensionality reduction via a matrix \( M \in \mathbb{R}^{d \times D}, d < D \). Among the ways to define the cost \( C(M, A) \) for the labelled training set \( A \), we choose a Linear Discriminant Analysis [5], as a natural linear measure, enjoying linear algebra tools for effective solution.

In linear discriminant analysis, the criteria of the quality of separation is the ratio of scatter between the classes to the scatter within the classes.
Let $i$ be the index running over $n$ points of the training set, $1 \leq i \leq n$. Let $k(i)$ be the class of point $x_i$. Then, we denote by $m_{k(i)} \equiv m_k$ the mean of the class $C_k$, corresponding to point $x_i$:

$$m_k = \frac{1}{n_k} \sum_{i \in C_k} (x_i),$$

(6)

Then the within-class scatter matrix is defined as

$$S_W = \frac{1}{n} \sum_{i=1}^{n} (x_i - m_{k(i)}) (x_i - m_{k(i)})^t,$$

(7)

And the between-class scatter matrix is defined as

$$S_B = \frac{1}{n} \sum_{i=1}^{n} (m_{k(i)} - m) (m_{k(i)} - m)^t,$$

(8)

where

$$m = \frac{1}{n} \sum_{i=1}^{n} x_i,$$

(9)

is the mean over all the data points.

The so constructed scatter matrices allow to evaluate the between-class scatter in the direction $v$ as $|v^t S_B v|$ and within-class scatter as $|v^t S_W v|$. Their ratio

$$\frac{|v^t S_B v|}{|v^t S_W v|}$$

is called a generalized Rayleigh Quotient [5], and its maximization can be done via solution of the generalized eigenvalue problem

$$S_B v = \lambda S_W v.$$

(11)

The generalized eigenvector problem with invertible $S_W$ (7) can be solved following [6]. Multiplying by $S_W^{-1}$, we obtain

$$S_W^{-1} S_B v = \lambda I v,$$

(12)

which is an ordinary eigenvector problem $B w = \lambda w$ for the matrix $S_W^{-1} S_B v$.

It is important to note, that at most $K - 1$ generalized eigenvalues of (11) are non-zero, since $\text{rank}(S_B) \leq K - 1$. This reflects the fact that $K$ class centers can span at most a $K - 1$ dimensional space. By the same argument, $\text{rank}(S_W) \leq n - K$. 

5
We solved (11) assuming that $S_W$ is invertible. However, here we consider the case $n \leq D$, in which the matrix $S_W$ (7) is rank deficient and therefore is not invertible. In order to treat this case, Howland et al. [2], utilize generalized singular value decomposition, which is described in the following section.

3.1 Generalized Singular Value Decomposition

The following theorem presents the GSVD [7], with somewhat altered notation of indices for the sake of clarity of our presentation.

**Theorem (Paige and Saunders)**

Given $A \in \mathbb{R}^{n \times D}$ and $B \in \mathbb{R}^{p \times D}$ with

$$C = \begin{bmatrix} A \\ B \end{bmatrix},$$

and $k = \text{rank}(C)$, there exist unitary matrices $U \in \mathbb{R}^{n \times n}$, $V \in \mathbb{R}^{p \times p}$, $W \in \mathbb{R}^{k \times k}$, and $Q \in \mathbb{R}^{D \times D}$ giving

$$U^T A Q = \Sigma_A \begin{bmatrix} W^T R & 0 \end{bmatrix},$$

$$V^T B Q = \Sigma_B \begin{bmatrix} W^T R & 0 \end{bmatrix},$$

where $R \in \mathbb{R}^{k \times k}$ is non-singular with singular values equal to the non-zero singular values of $C$. The diagonal $C_A(S_B)$ has non-increasing (non-decreasing) values $c_i(s_i)$ and $c_i^2 + s_i^2 = 1$.

The theorem statement (13) can be rewritten as

$$U^T A Q \begin{bmatrix} R^{-1}W \\ I \end{bmatrix} = \Sigma_A$$

$$V^T B Q \begin{bmatrix} R^{-1}W \\ I \end{bmatrix} = \Sigma_B.$$
Substituting
\[ A \equiv H_B^T = \sqrt{\frac{1}{n}} [m_{k(1)} - m, \ldots, m_{k(n)} - m] \]
\[ B \equiv H_W^T = \sqrt{\frac{1}{n}} [x_1 - m_{k(1)}, \ldots, x_n - m_{k(n)}], \]
(in our case \( H_W^T \in \mathbb{R}^{n \times D} \)), we obtain that the first rows of
\[ \begin{bmatrix} R^{-1}W \\ I \end{bmatrix} \]
maximize the separation in the sense of (10,11).

4 Overfitting and regularization

From (16) it follows that \( \text{rank}(H_B) \leq K - 1 \), and therefore only first \( K - 1 \) dimensions are significant to maximize (10,11). This reflects the fact that \( K \) classes span a \( K - 1 \) dimensional space. Similarly, \( \text{rank}(H_W) \leq n - K \), as can be seen from (17). This reflects the fact that that \( n_k \) points in class \( k \) span a \( n_k - 1 \) dimensional space. We are considering the case where the dimensionality \( D \) of a training data \( x \in \mathbb{R}^D \) exceeds the number of training points \( n \). Thus, there exists a \( D - n + K \) dimensional subspace, in which there is a zero within-class scatter.

\[ \operatorname{dim} (\text{null}(H_W)) = D - n + K. \]  \hspace{1cm} (19)

Since, in general, the between-class scatter is non-zero in this subspace, the generalized eigenvectors of (11), corresponding to maximum (here infinite) generalized eigenvalues lie in this subspace. This allows us to propose a new algorithm for maximization of (10):

1. Find QR decomposition of \( H_W \): \( H_W = [Q_{W1} \quad Q_{W2}] R \). Here \( \text{ran} (Q_{w1}) = \text{ran} (H_W) \), and \( \text{ran} (Q_{W2}) = \text{null} (H_W) \).

2. Project the data matrix \( A \) (1) onto \( \text{null} (H_W) \): \( \tilde{A}^{D-n+K \times n} = Q_{W2}^T A \). (Now there is a zero within-class scatter in \( \tilde{A} \)).

3. Find \( \tilde{H}_B \) from \( \tilde{A} \) following (16).
4. Find QR decomposition of $\tilde{H}_B$: $\tilde{H}_B = [Q_{B_1} \quad Q_{B_2}] \, R_B$. Here $\text{ran} \ (Q_{B_1}) = \text{ran} \ (\tilde{H}_B)$, and $\text{ran} \ (Q_{B_1}) = \text{null} \ (\tilde{H}_B)$. (The subspace $\text{ran} \ (Q_{B_2})$ is useless since there is a zero between-class scatter in this subspace.)

5. The data matrix in the reduced dimension is given by $A^{K-1 \times n} = Q_{B_2}^T \hat{A} = Q_{B_1} Q_{W_2}^T A$.

4.1 Regularized Linear Discriminant Analysis

Although the reduction of within-class scatter to zero sounds very attractive, it is obviously a bad solution. Existence of the subspace zeroing the within-class scatter is not an intrinsic feature of the data, but a simple consequence of the fact that there was not enough data to span all the feature space. Therefore, the new data points will probably (almost certainly, see Section 2) deviate from other class members, reduced to a point. One can say that there is an ‘overfitting’ in dimensionality reduction since we had too many free parameters for choice of reduction matrix, relatively to amount of data. The standard recipe to cure this ill-posed problem is regularization [8].

To compensate for the lack of data, we modify (7), substituting for each point $x_i$ a cloud of points scattered randomly at distance $\delta$ around $x_i$

$$\tilde{S}_W = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{n} \sum_{i=1}^{n} \left( x_i + \delta_j - m_{k(i)} \right) \left( x_i + \delta_j - m_{k(i)} \right)^t,$$

Since the expected value of $\delta = 0$, this results in

$$\tilde{S}_W = S_W + \delta^2 I.$$  \hspace{1cm} (21)

We can not derive the ‘optimal’ value of $\delta^2$, however it is estimated to be about the smallest eigenvalues of $S_W$. It is not hard to be convinced that if the underlying data distribution is anisotropic, the null space of $S_W$ will be along the directions of the smallest variance.
5 Experimental Results

5.1 MNIST Handwritten Digits

The MNIST database [9] consists of 60000 labelled handwritten digits ‘0’, . . . , ‘9’. The digits are stored as $28 \times 28$ images with 256 levels of gray. We choose a random subsets of 550, 600, 1000, 2000 and 4000 images (with about $\frac{1}{10}$ of this quantity of each digit) to calculate the $K - 1 = 9$ dimensional LDA basis $\{v_1, \ldots, v_9\}$.

The LDA basis was calculated by solving generalized eigenvalue problem (11), with negligible regularization added to $S_W$: $S_W + \delta I$, $\delta = 10^{-8}$ in order to make it invertible and to use solution (12). We found the optimal subspace via generalized eigenvalue problem 11. There were only 9 non-zero eigenvalues in accordance with the number of classes $K = 10$.

Both the training set (the set of points, used to find the LDA space) and the test set (500 points randomly chosen from the rest of the MNIST database) were projected onto the LDA space. The obtained points in 9 dimensional space were visualized via 5 two dimensional plots, presenting respectively the coordinates 1 and 2, . . . , 7 and 8, 9 and 1. A subset of 500 digits was chosen from the training set in order to not clutter the figures.

Although one might expect that digits from $28 \times 28$ images occupy the $28^2 = 784$ dimensional space, practically it was only $\sim 650$ in the limit of large size of the training set, since some pixels near the corners and boundary of the $28 \times 28$ frame are identical zeros for all the images.

Figures 1-2 show the scatter in the dimensions 1, 2 of the training and test sets without regularization, while figures 3-4 show the same scatter in the less discriminative dimensions 7, 8. One can see the cluster structure of the digit distribution, and that the overlapping digits on one figure are separated on the other. Obviously, full 9 dimensions yield better separation than can be shown on a two dimensional plot.

Figures 5-12 present the training and test set scattering in the dimensions 1,2 of LDA space build respectively for 2000, 1000, 600 and 550 training points. As predicted, de-
crease of the size of the training set leads to decrease of the scatter of the training set and simultaneous increase of the scatter of the test set.

Regularization (21), results in increased scattering of the training set but decreased scattering of the test set. Figures 13-16 show the training and test set scattering in dimensions 1 and 2 for 4000 and 550 points in the regularized case.

Table 1 presents the within-class scatter for the training and the test points, in the LDA basis build from 550, . . . , 4000 training points. One can see, that regularization decreases the within-class scatter of the test set, while increasing the scatter of the training set, as expected. Value of regularization parameter $\delta$ is given in parenthesis.

### 5.2 Classification

Table 1 demonstrates that the within class scatter can be controlled by regularization. In order to evaluate the influence of regularization on classification, the 10 nearest neighbor classifier was applied to the 9 dimensional data, obtained after the reduction. The size of both training and test sets was 2000 points, randomly and disjointedly chosen from the 60000 points of the MNIST database. The training points were used for both dimensionality reduction and the ‘training’ of the classifier. The 9 first generalized eigenvectors, corresponding to largest eigenvalues, obtained in the dimensionality reduction of the training set (11) with regularization (21) were used for dimensionality reduction of

| Train. Set Size | $|S_W|_{Train.}$ | $|S_W|_{Test}$ | $|S_W|_{Train.\,Reg.}$ | $|S_W|_{TestReg.}$ | Rank(Sw) |
|-----------------|----------------|--------------|----------------|----------------|-----------|
| 550             | 0.0±0          | 50±50        | 0.27±.01       | 0.34±.01       | 540±0     |
| 600             | .096±.001      | 25±10        | 0.28±.01       | 0.34±.01       | 560±5     |
| 1000            | .198±.002      | 10±2         | 0.28±.01       | 0.32±.02       | 580±10    |
| 2000            | .230±.005      | .70±.05      | 0.27±.01       | 0.29±.02       | 610±5     |
| 4000            | .230±.005      | 0.40±.1      | 0.26±.01       | 0.27±.01       | 640±5     |

Table 1: Within-class scatter ($\pm$Variance) on training and test sets in the LDA basis, build for 550, . . . , 4000 training points
Figure 1: Training points in dimensions 1,2 of LDA space formed for 4000 digits

Figure 2: Test points in dimensions 1,2 of LDA space formed for 4000 digits

Figure 3: Training points in dimensions 7,8 of LDA space formed for 4000 digits

Figure 4: Test points in dimensions 7,8 of LDA space formed for 4000 digits

Figure 5: Training points in dimensions 1,2 of LDA space formed for 2000 digits

Figure 6: Test points in dimensions 1,2 of LDA space formed for 2000 digits
Figure 7: Training points in dimensions 1,2 of LDA space formed for 1000 digits

Figure 8: Test points in dimensions 1,2 of LDA space formed for 1000 digits

Figure 9: Training points in dimensions 1,2 of LDA space formed for 600 digits

Figure 10: Test points in dimensions 1,2 of LDA space formed for 600 digits

Figure 11: Training points in dimensions 1,2 of LDA space formed for 550 digits

Figure 12: Test points in dimensions 1,2 of LDA space formed for 550 digits
Figure 13: Training points in dimensions 1.2 of LDA space formed for 4000 digits, with regularization \( \delta = 0.01 \)

Figure 14: Test points in dimensions 1.2 of LDA space formed for 4000 digits, with regularization \( \delta = 0.01 \)

Figure 15: Training points in dimensions 1.2 of LDA space formed for 550 digits, with regularization \( \delta = 0.01 \)

Figure 16: Test points in dimensions 1.2 of LDA space formed for 550 digits, with regularization \( \delta = 0.01 \)

<table>
<thead>
<tr>
<th>Regularization</th>
<th>(10^{-12})</th>
<th>(10^{-9})</th>
<th>(10^{-6})</th>
<th>(10^{-3})</th>
<th>(10^{-2})</th>
<th>(10^0)</th>
<th>(10^2)</th>
<th>(10^6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train. 550</td>
<td>2 ( \cdot ) 10^{-4}</td>
<td>0.02</td>
<td>0.09</td>
<td>0.21</td>
<td>0.27</td>
<td>0.34</td>
<td>0.35 ( \pm ) 0.01</td>
<td>0.35 ( \pm ) 0.01</td>
</tr>
<tr>
<td>Test 550</td>
<td>50</td>
<td>8</td>
<td>1.5</td>
<td>0.39</td>
<td>0.34</td>
<td>0.36</td>
<td>0.35 ( \pm ) 0.01</td>
<td>0.35 ( \pm ) 0.01</td>
</tr>
<tr>
<td>Train. 1000</td>
<td>0.20 ( \pm ) 0.01</td>
<td>0.20 ( \pm ) 0.01</td>
<td>0.205 ( \pm ) 0.005</td>
<td>0.24 ( \pm ) 0.01</td>
<td>0.27 ( \pm ) 0.01</td>
<td>0.32 ( \pm ) 0.01</td>
<td>0.34 ( \pm ) 0.01</td>
<td>0.35 ( \pm ) 0.01</td>
</tr>
<tr>
<td>Test 1000</td>
<td>7 ( \pm ) 3</td>
<td>2 ( \pm ) 0.5</td>
<td>0.60 ( \pm ) 0.02</td>
<td>0.35 ( \pm ) 0.01</td>
<td>0.32 ( \pm ) 0.01</td>
<td>0.34 ( \pm ) 0.01</td>
<td>0.35 ( \pm ) 0.01</td>
<td>0.35 ( \pm ) 0.01</td>
</tr>
<tr>
<td>Train. 4000</td>
<td>0.23 ( \pm ) 0.005</td>
<td>0.23 ( \pm ) 0.005</td>
<td>0.23 ( \pm ) 0.05</td>
<td>0.25 ( \pm ) 0.01</td>
<td>0.26 ( \pm ) 0.01</td>
<td>0.307 ( \pm ) 0.01</td>
<td>0.31 ( \pm ) 0.01</td>
<td>0.30 ( \pm ) 0.01</td>
</tr>
<tr>
<td>Test 4000</td>
<td>1.2 ( \pm ) 0.5</td>
<td>0.44 ( \pm ) 0.03</td>
<td>0.29 ( \pm ) 0.01</td>
<td>0.26</td>
<td>0.27 ( \pm ) 0.01</td>
<td>0.312 ( \pm ) 0.01</td>
<td>0.32 ( \pm ) 0.01</td>
<td>0.31 ( \pm ) 0.01</td>
</tr>
</tbody>
</table>

Table 2: Within-class scatter (\(\pm Variance\)) on the Training and Test set for different values of regularization
the test set.

Table 3 shows the classification performance for several values of regularization. The classification performance fall short comparing to the > 99% state of the art for this problem [9], but serves the purpose of illustrating the ideas discussed in our paper.

<table>
<thead>
<tr>
<th>Regularization</th>
<th>Performance (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-12}$</td>
<td>64.0 ± 1</td>
</tr>
<tr>
<td>$10^{-9}$</td>
<td>66.2 ± 1</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>67.8 ± 1</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>83.2 ± 8</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>85.9 ± 7</td>
</tr>
<tr>
<td>$10^0$</td>
<td>86.0 ± 6</td>
</tr>
<tr>
<td>$10^2$</td>
<td>85.8 ± 8</td>
</tr>
<tr>
<td>$10^6$</td>
<td>85.6 ± 8</td>
</tr>
</tbody>
</table>

Table 3: Classification performance of the 10-nearest neighbor classifier on MNIST handwritten digits database. Training on 1000 samples in 9 dimensional space obtained with regularized LDA. Test on 1000 samples reduced to the same space.

6 Discussion and Summary

We discussed the problem of linear dimensionality reduction for classification. We have shown that the classical Linear Discriminant Analysis approach can fail in the cases when the dimensionality exceeds the number of training points. In such cases the LDA approach achieves zero scattering within the classes, and yet sub-optimal classification performance.

The regularization approach cures this problem resulting in lower training and higher test performance. We show, that regularization is equivalent to boosting, when each data point generates the cloud of the points of the same class in its vicinity.

References


