Reduction of Feature Space Dimension Based on Separability Criterion

A. Nemirko
Saint Petersburg Electrotechnical University “LETI”, Russia
nemirko@yandex.ru

Abstract: *Linear transformation of data in multidimensional feature space based on Fisher’s criterion is considered. The case of two classes is studied. We derived expressions for recurrent calculation of weight vectors which form new features. Examples offered shows that the newly found features which represent the data more accurately make it possible to achieve linear separability of classes which remains impossible using the technique of principal components and the classic Fisher’s linear discriminant.*

Keywords: the classic Fisher’s linear discriminant, separability in pattern recognition.

1. INTRODUCTION

The technique of principal components (PCA) is well-known in pattern recognition theory [1]; it is widely used to reduce dimensionality of feature space and represent the data in the two dimensional space of the first two principal components. PCA is not intended for recognition of objects in feature space. However, it is often used for that purpose due to simplicity of data representation on a plane. Fisher’s linear discriminant (LDF) is also applied to tasks of pattern recognition [2]. It reduces the dimensionality of feature space from its initial value to unit projecting multidimensional data onto a straight line. Unlike PCA, the LDF technique exploits the information on differences between the statistical data on classes to find an optimal solution. It is possible to improve the quality of classification by LDF technique using some larger number of features instead of just one for recognition.

2. FISHER’S LINEAR DISCRIMINANT

Retrieval of the optimal weight vector (Fisher’s linear discriminant) for arbitrary distributions of two classes is described in detail in [2]. Fisher’s linear discriminant may be defined as vector **W** for which the following functional criterion reaches its maximum:

$$J(\mathbf{W}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} . \qquad (1)$$

In this formula m_1 и m_2 - mean values of classes, projected on **W,** s_1^2 and s_2^2 sample intra-class scattering for the two classes. Provided that J(**W**) = max, the distance between the projections of classes onto **W** reaches its maximum for **W**.

According to [2] the equation (1) may be rewritten in the following form

$$J(\mathbf{W}) = \frac{\mathbf{W}^T \mathbf{S}_B \mathbf{W}}{\mathbf{W}^T \mathbf{S}_W \mathbf{W}} . \qquad (2)$$

where $\mathbf{S}_B = (\mathbf{M}_1 - \mathbf{M}_2)(\mathbf{M}_1 - \mathbf{M}_2)^T$ is the matrix of cross-class scattering, $\mathbf{M}_1$ and $\mathbf{M}_2$ vectors of mean values for two classes, $\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2$ - the matrix of intra-class scattering, $\mathbf{S}_1$ and $\mathbf{S}_2$ matrices of intra-class scattering within corresponding classes,

$$\mathbf{S}_j = \sum_{i=1}^{n_j} (\mathbf{X}_i^{(j)} - \mathbf{M}_j)(\mathbf{X}_i^{(j)} - \mathbf{M}_j)^T , \ j = 1,2 ,$$

$\mathbf{X}_i^{(j)}$ - i-th input vector of j-th class, n_1 and n_2 - the number of members of each class. Analysis of this formula demonstrates [2] that the maximum $J(\mathbf{W})$ may be reached when

$$\mathbf{W} = \mathbf{S}_W{}^{-1}(\mathbf{M}_1 - \mathbf{M}_2) \qquad (3)$$

For the original n-dimensional feature space, we can rewrite this expression in the following form

$$\mathbf{W}_n = \mathbf{S}_n^{-1}(\mathbf{m1}_n - \mathbf{m2}_n) . \qquad (4)$$

3. CONSTRUCTING OF FEATURE SPACE WITH THE USE OF FISHER’S CRITERION

We can project all the data onto a plane normal to $\mathbf{W}_n$. Then, using Fisher’s criterion one may find the best weight vector on that plane which is an (n-1)-dimensional feature space in its turn. Apparently, we have on that plane [3, 4]:

$$\mathbf{W}_{n-1} = \mathbf{S}_{n-1}^{-1}(\mathbf{m1}_{n-1} - \mathbf{m2}_{n-1}) , \qquad (5)$$

where

$$\mathbf{m1}_{n-1} = \mathbf{m1}_n - m1_n \mathbf{W}_n = \mathbf{m1}_n - \mathbf{W}_n{}^T \mathbf{m1}_n \mathbf{W}_n ,$$
$$\mathbf{m2}_{n-1} = \mathbf{m2}_n - m2_n \mathbf{W}_n = \mathbf{m2}_n - \mathbf{W}_n{}^T \mathbf{m2}_n \mathbf{W}_n ,$$
$$\mathbf{m1}_{n-1} - \mathbf{m2}_{n-1} = \mathbf{m1}_n - \mathbf{m2}_n - \mathbf{W}_n{}^T (\mathbf{m1}_n - \mathbf{m2}_n) \mathbf{W}_n . \qquad (6)$$

When calculating $\mathbf{S}_{n-1}^{-1}$, the reversible matrix may appear degenerated. In that case one may calculate a pseudo-inverse matrix instead of the explicit inverse one. To that end, the MATLAB system should include pinv(**X**) instead of inv(**X**).

$$\mathbf{S}_{n-1} = \mathbf{S1}_{n-1} + \mathbf{S2}_{n-1} \qquad (7)$$

$$\mathbf{S1}_{n-1} = \sum_{x \in \omega_1} (\mathbf{X}_{n-1} - \mathbf{m1}_{n-1})(\mathbf{X}_{n-1} - \mathbf{m1}_{n-1})^T$$

where

$$\mathbf{X}_{n-1} - \mathbf{m1}_{n-1} = (\mathbf{X}_n - \mathbf{W}_n{}^T \mathbf{X}_n \mathbf{W}_n) - (\mathbf{m1}_n - \mathbf{W}_n{}^T \mathbf{m1}_n \mathbf{W}_n) =$$
$$= (\mathbf{X}_n - \mathbf{m1}_n) - \mathbf{W}_n{}^T (\mathbf{X}_n - \mathbf{m1}_n) \mathbf{W}_n ,$$

$$\begin{aligned}&\mathbf{S}1_{n-1} = \sum_{x\in\omega_1}[(\mathbf{X}_n - \mathbf{m}1_n) - \mathbf{W}_n{}^T(\mathbf{X}_n - \mathbf{m}1_n)\mathbf{W}_n]\times\\ &\times\Big[(\mathbf{X}_n-\mathbf{m}1_n)-\mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n]^T =\\ &= \sum_{x\in\omega_1}[(\mathbf{X}_n-\mathbf{m}1_n)(\mathbf{X}_n-\mathbf{m}1_n)^T -\\ &-\mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n(\mathbf{X}_n-\mathbf{m}1_n)^T -\\ &-(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n{}^T + B],\end{aligned}$$

where

$$B = \mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n\cdot[\mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n]^T .$$

We denote $\alpha = \mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)$. This is a scalar. Note also that

$\alpha = (\mathbf{X}_n-\mathbf{m}1_n)^T\mathbf{W}_n$. Then $B = \alpha\,\mathbf{W}_n\alpha\,\mathbf{W}_n{}^T$ and

$$\begin{aligned}&\mathbf{S}1_{n-1} = \sum_{x\in\omega_1}[(\mathbf{X}_n-\mathbf{m}1_n)(\mathbf{X}_n-\mathbf{m}1_n)^T -\\ &-\alpha\,\mathbf{W}_n(\mathbf{X}_n-\mathbf{m}1_n)^T - \alpha\,(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n{}^T + B]\end{aligned}\tag{8}$$

Since

$$\begin{aligned}&\sum_{x\in\omega_1}[-\alpha\,\mathbf{W}_n(\mathbf{X}_n-\mathbf{m}1_n)^T - \alpha\,(\mathbf{X}_n-\mathbf{m}1_n)\mathbf{W}_n{}^T] =\\ &=\sum_{x\in\omega_1}[(\alpha\,\mathbf{W}_n\mathbf{m}1_n^T - \alpha\,\mathbf{W}_n\mathbf{X}_n^T) + (\alpha\,\mathbf{m}1_n\mathbf{W}_n{}^T - \alpha\,\mathbf{X}_n\mathbf{W}_n{}^T)] =\\ &=(\alpha\,N_1\mathbf{W}_n\mathbf{m}1_n^T - \alpha\,N_1\mathbf{W}_n\mathbf{m}1_n^T) +\\ &+(\alpha\,N_1\mathbf{m}1_n\mathbf{W}_n{}^T - \alpha\,N_1\mathbf{m}1_n\mathbf{W}_n{}^T) = 0,\end{aligned}$$

where N_1 is the number of elements in set ω_1, it follows from (7) that

$$\begin{aligned}&\mathbf{S}1_{n-1} = \sum_{x\in\omega_1}(\mathbf{X}_n-\mathbf{m}1_n)(\mathbf{X}_n-\mathbf{m}1_n)^T + \sum_{x\in\omega_1}B =\\ &=\mathbf{S}1_n + \sum_{x\in\omega_1}B.\end{aligned}\tag{9}$$

Since

$$\begin{aligned}&\sum_{x\in\omega_1}B = \sum_{x\in\omega_1}\alpha\,\mathbf{W}_n\alpha\,\mathbf{W}_n{}^T =\\ &=\sum_{x\in\omega_1}\mathbf{W}_n{}^T(\mathbf{X}_n-\mathbf{m}1_n)(\mathbf{X}_n-\mathbf{m}1_n)^T\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T) =\\ &=\mathbf{W}_n{}^T[\sum_{x\in\omega_1}(\mathbf{X}_n-\mathbf{m}1_n)(\mathbf{X}_n-\mathbf{m}1_n)^T]\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T) =\\ &\mathbf{W}_n{}^T\mathbf{S}1_n\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T),\end{aligned}$$

then

$$\mathbf{S}1_{n-1} = \mathbf{S}1_n + \mathbf{W}_n{}^T\mathbf{S}1_n\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T)$$

and

$$\begin{aligned}&\mathbf{S}_{n-1} = \mathbf{S}1_{n-1} + \mathbf{S}2_{n-1} =\\ &=\mathbf{S}1_n + \mathbf{S}2_n + \mathbf{W}_n{}^T(\mathbf{S}1_n + \mathbf{S}2_n)\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T) =\\ &=\mathbf{S}_n + \mathbf{W}_n{}^T\mathbf{S}_n\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T).\end{aligned}\tag{10}$$

Finally, in consideration of (6) we derive from (5)

$$\mathbf{W}_{n-1} = \mathbf{S}_{n-1}^{-1}[(\mathbf{m}1_n - \mathbf{m}2_n) - \mathbf{W}_n{}^T(\mathbf{m}1_n - \mathbf{m2}_n)\mathbf{W}_n],\tag{11}$$

or substituting expression (10) in (11) we obtain

$$\begin{aligned}&\mathbf{W}_{n-1} = [\mathbf{S}_n + \mathbf{W}_n{}^T\mathbf{S}_n\mathbf{W}_n(\mathbf{W}_n\mathbf{W}_n{}^T)]^{-1}\times\\ &\times[(\mathbf{m}1_n - \mathbf{m}2_n) - \mathbf{W}_n{}^T(\mathbf{m}1_n - \mathbf{m2}_n)\mathbf{W}_n]\end{aligned}\tag{12}$$

In an (n-2)-dimensional feature space we derive, respectively

$$\begin{aligned}&\mathbf{W}_{n-2} = [\mathbf{S}_{n-1} + \mathbf{W}_{n-1}{}^T\mathbf{S}_{n-1}\mathbf{W}_{n-1}(\mathbf{W}_{n-1}\mathbf{W}_{n-1}{}^T)]^{-1}\times\\ &\times[(\mathbf{m}1_{n-1} - \mathbf{m}2_{n-1}) - \mathbf{W}_{n-1}{}^T(\mathbf{m}1_{n-1} - \mathbf{m2}_{n-1})\mathbf{W}_{n-1}]\end{aligned}$$

and so on.

4. EXPERIMENTS WITH LINEARLY SEPARABLE CLASSES

For experimental studies, we have chosen two sets of 3-dimensional data f1 and f2 presented in Table 1.

Table 1

f1		
0.7	0.3	1.2
0.5	0.7	1.0
0.4	1.0	0.4
0.7	0.7	1.0
0.6	0.6	1.5
0.6	0.6	1.2
0.6	0.5	1.0
0.4	0.9	0.6
0.5	0.6	1.1
0.8	0.3	1.2
f2		
0.4	0.2	0.8
0.2	0.2	0.7
0.9	0.3	0.5
0.8	0.3	0.6
0.5	0.6	0.4
0.6	0.5	0.7
0.4	0.4	1.2
0.6	0.3	1.0
0.3	0.2	0.6
0.5	0.5	0.8

It is known that these sets are linearly separable in 3-dimensional space. Let us apply to them consecutively the technique of principal components, Fisher's linear discriminant and Fisher's linear discriminant with an extra feature. Calculations are carried out using the MATLAB system.

```
% TECHNIQUE OF PRINCIPAL COMPONENTS
A=load('f1.txt')
B=load('f2.txt')
C=[A;B]
gg=['A'; 'A'; 'A'; 'A'; 'A'; 'A'; …
'A'; 'A'; 'A'; 'A'; 'B'; 'B'; 'B';…
'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'];
g=cellstr(gg)
[PC,SCORE]=princomp(C)
gscatter(SCORE(:,1),SCORE(:,2),…
g,'','xos')
```

Results of running the above program are presented in Fig. 1. It shows clearly that the technique of principal components does not ensure linear separability of classes.

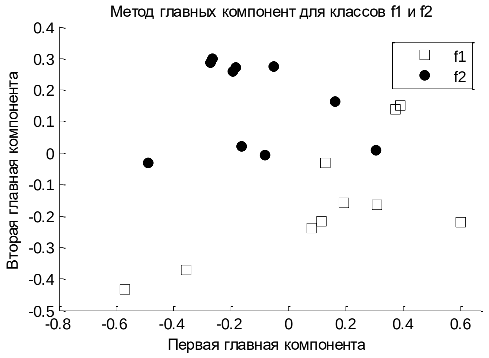

Fig. 1

The next program implements the classic technique of Fisher’s linear discriminant and the technique of Fisher’s linear discriminant with an extra feature.

```
% CLASSIC FISHER’S LINEAR DISCRIMINANT
load ('f1.txt') % loading of the
% matrix of Class 1
load ('f2.txt') % loading of the
% matrix of Class 2
A1=f1
A2=f2
n1=length(A1) % number of members of
% Class 1
n2=length(A2) % number of members of
% Class 2
m1= mean(A1)
m2= mean(A2)
E1=(n1-1)*cov(A1)
E2=(n2-1)*cov(A2)
E=E1+E2
W=inv(E)*(m1-m2)'
w=W/norm(W) % normalization
X1=A1*w % Class 1 projections on w
X2=A2*w % Class 2 projections on w
t=0.5:0.1:1.40
x1=hist(X1,t)
x2=hist(X2,t)
figure
plot(x1,'-*g')
hold on
plot(x2,'-or')
```

The result of classic Fisher’s discriminant analysis can be shown in the figure [4]. Apparently, no linear separability of classes is found in this case either.

```
% PROGRAM CONTINUED
% FISHER’S LINEAR DISCRIMINANT
% WITH AN EXTRA FEATURE.
for i=1:10
B1(i,:)=A1(i,:)-X1(i,:)*w'
end
for k=1:10
B2(k,:)=A2(k,:)-X2(k,:)*w'
end
mB1=mean(B1)
mB2=mean(B2)
h1=inv(w'*E*w*(w*w')+E) % further we
% use the formula(12)derived above
h3=(m1-m2)'
h5=w'*(m1-m2)'*w
h6=h3-h5
W22=h1*h6
w22=W22/norm(W22)
Y1=B1*w22 % Class 1 projections on w2
Y2=B2*w22 % Class 2 projections on w2
figure
scatter(X1,Y1)
hold on
scatter(X2,Y2)
% number of errors is equal to 0
```

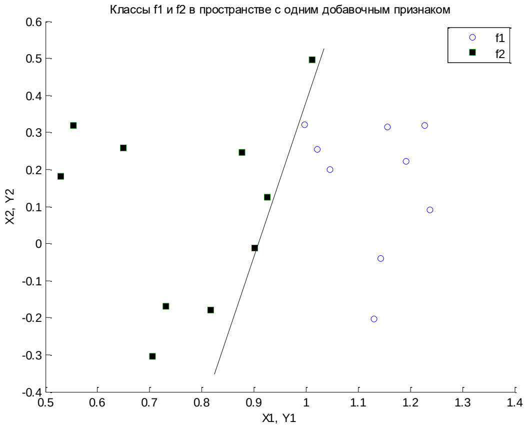

Fig. 2

The result of running this program (Fig. 2) shows clearly that an extra feature helped to find a weight vector ensuring complete linear separability of Classes f1 and f2. The same result was obtained when finding weight vector with a simple single-layer perceptron.

We assume the degree of classes separation and the classification error as the proportion of points that fall in the intersection of classes (intersection of classes convex hulls). In the case of complex configurations of classes distributions the winnings can be substantial. For two classes shown in Fig. 3, the use of the Fisher test gives the weight vector w shown in the figure, with a total classification error equal to 33%. On the plane (Fig. 3) it is easy to find a vector w1 that separates these classes accurately.

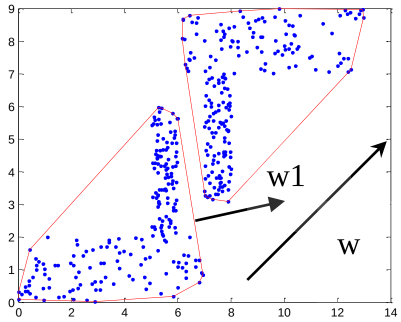

Fig. 3

Experiments conducted with other data demonstrated that an extra feature does improve separability of classes. It may be important for certain tasks, especially those requiring complete separability of classes.

5. CONCLUSION

The technique of Fisher's linear discriminant is considered. To improve the quality of linear recognition of two classes it is suggested to use extra features that may be retrieved with the use of Fisher's criterion as orthogonal weight vectors in spaces of lower dimensionality. We derived a recurrent expression for calculating such extra features consecutively. The examples presented above demonstrates that using a single extra feature a complete linear separability of two classes may be achieved, not detectable by either the technique of principal components or the classic Fisher's linear discriminant. Other criteria of distance between classes may also be used for the suggested approach.

6. REFERENCES

[1] Ayvazian S.A., Bezhaeva Z.I., Staroverov O.V. Classification of Multidimensional Observations. Moscow: Statistika, 1974. - 240 p.

[2] Duda R., Hart P. Pattern Recognition and Scene Analysis. Transl. from Eng. Moscow: Mir, 1976. – 511 p.

[3] Manilo L.A. Ordering of spectral attributes by empirical estimations of intergroup distance in problems of biosignals classification. – Izvestia vuzov Rossii. Radioelectronika. Vyp.3, 2006 – p.20 – 29.

[4] Nemirko A.P. Transformation of feature space based on Fisher's linear discriminant. Pattern Recognition and Image Analysi00s. Advances in Mathematical Theory and Applications. - Volume 26 / 2016, No 2. pp. 257-261.