A New Method of Feature Extraction and Its Stability

Nojun Kwak and Chong-Ho Choi

School of Electrical Eng., ASRI, Seoul National University
San 56-1, Shinlim-dong, Kwanak-ku, Seoul 151-742, KOREA
{triplea, chchoi}@csl.snu.ac.kr

Abstract. In the classification on a high dimensional feature space such as in face recognition problems, feature extraction techniques are usually used to overcome the so called 'curse of dimensionality.' In this paper, we propose a new feature extraction method for classification problem based on the conventional independent component analysis. The local stability of the proposed method is also dealt with. The proposed algorithm makes use of the binary class labels to produce two sets of new features; one that does not carry information about the class label – these features will be discarded – and the other that does. The advantage is that general ICA algorithms become available to a task of feature extraction by maximizing the joint mutual information between class labels and new features, although only for two-class problems. Using the new features, we can greatly reduce the dimension of feature space without degrading the performance of classifying systems.

1 Introduction

Feature extraction is very important in many applications and subspace methods such as principle component analysis (PCA) and Fisher's linear discriminant analysis (FLD) have been used for this purpose. Recently, in neural networks and signal processing communities, independent component analysis (ICA), which was devised for blind source separation problems, has received a great deal of attention because of its potential applications in various areas. The advantage of ICA is that it uses higher order statistics, while PCA and FLD use second order statistics. But it leaves much room for improvement since it does not utilize output class information.

In this paper, we propose a new ICA-based feature extraction algorithm (ICA-FX) for classification problem which is an extended version of [1]. It utilizes the output class information in addition to having the advantages of the original ICA method. This method is well-suited for classification problems in the aspect of constructing new features that are strongly related to output class. By combining the proposed algorithm with existing feature selection methods, we can greatly reduce the dimension of feature space while improving classification performance.

J.R. Dorronsoro (Ed.): ICANN 2002, LNCS 2415, pp. 480–485, 2002.

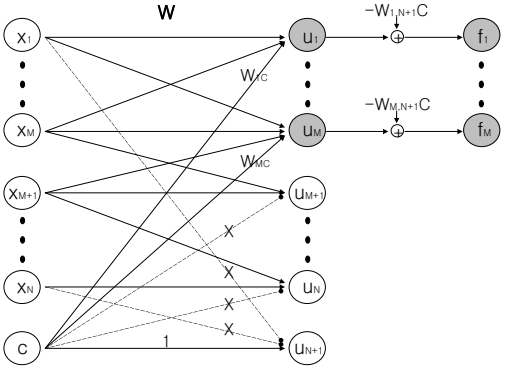

Fig. 1. Structure of ICA-FX

This paper is organized as follows. In Section 2, we propose a new feature extraction algorithm. In Section 3, local stability condition of the proposed algorithm is presented. Simulation results are presented in Section 4 and Conclusions follow in Section 5.

2 Algorithm: ICA-FX

Suppose that there are N zero-mean normalized input features $\boldsymbol{x} = [x_1, \cdots, x_N]^T$ and a binary output class $c \in \{-1, 1\}$. Our purpose of the feature extraction is to find $M(\leq N)$ new features $\boldsymbol{f}_a = [f_1, \cdots, f_M]^T$ from $\boldsymbol{x}$ containing maximal information of the class c.

Using the Fano's inequality and data processing inequality in information theory [2], if we restrict our attention on linear transformations from $\boldsymbol{x}$ to $\boldsymbol{f}_a$, the feature extraction problem can be restated as the following optimization problem:

(FX problem.) Find $K_a \in \Re^{M \times N}$ that maximize $I(c; \boldsymbol{f}_a)$, where $\boldsymbol{f}_a = K_a \boldsymbol{x}$. Here, $I(c; \boldsymbol{f}_a)$ denotes the mutual information between the class label c and the feature vector $\boldsymbol{f}_a$.

To solve this problem, we interpret feature extraction problem in the structure of the blind source separation (BSS) problem in the following.

(Mixing.) Assume that there exist N independent sources $\boldsymbol{s} = [s_1, \cdots, s_N]^T$ which are also independent of class label c and the observed features $\boldsymbol{x}$ is the linear combination of the sources $\boldsymbol{s}$ and c with the mixing matrix $A \in \Re^{N \times N}$ and $\boldsymbol{b} \in \Re^{N \times 1}$ such that

$$\boldsymbol{x} = A\boldsymbol{s} + \boldsymbol{b}c. \tag{1}$$

(Unmixing.) Our unmixing stage is a little bit different from the BSS problem and this is shown in Fig. 1. Here, $\boldsymbol{x}$ is fully connected to $\boldsymbol{u} = [u_1, \cdots, u_N]$,

c is connected to $\boldsymbol{u}_a = [u_1, \cdots, u_M]$, and $u_{N+1} = c$. Thus, the unmixing matrix $\boldsymbol{W} \in \Re^{(N+1)\times(N+1)}$ becomes

$$\boldsymbol{W} = \begin{pmatrix} w_{1,1} & \cdots & w_{1,N} & w_{1,N+1} \\ \vdots & & \vdots & \vdots \\ w_{M,1} & \cdots & w_{M,N} & w_{M,N+1} \\ w_{M+1,1} & \cdots & w_{M+1,N} & 0 \\ \vdots & & \vdots & \vdots \\ w_{N,1} & \cdots & w_{N,N} & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix}. \tag{2}$$

If we denote the upper left $N \times N$ matrix of $\boldsymbol{W}$ as W and the $(N+1)$th column of $\boldsymbol{W}$ as $\boldsymbol{v} \in \Re^{N\times 1}$, the unmixing equations becomes

$$\boldsymbol{u} = W\boldsymbol{x} + \boldsymbol{v}c. \tag{3}$$

Suppose we have made $\boldsymbol{u}$ somehow equal to the scaled and permuted version of source $\boldsymbol{s}$, i.e.,

$$\boldsymbol{u} = \Lambda\Pi\boldsymbol{s}, \tag{4}$$

where Λ is a diagonal matrix corresponding to appropriate scale and Π is a permutation matrix. Then, u_i's $(i = 1, \cdots, N)$ are independent of class c, and among the elements of $\boldsymbol{f} = W\boldsymbol{x} (= \boldsymbol{u} - \boldsymbol{v}c)$ $\boldsymbol{f}_a = [f_1, \cdots, f_M]^T$ will contain the same amount of the mutual information with class c as original features $\boldsymbol{x}$ have, i.e.,

$$I(c; \boldsymbol{f}_a) = I(c; \boldsymbol{x}). \tag{5}$$

if the unmixing matrix $\boldsymbol{W}$ be nonsingular. Thus, we can extract $M (< N)$ dimensional new feature vector $\boldsymbol{f}_a$ by a linear transformation of $\boldsymbol{x}$ containing the maximal information about the class if we can make (4) hold.

Theorem 1. *In the structure shown in Fig. 1, we can obtain independent u_i's $(i = 1, \cdots, N)$ that are also independent of c by the learning rule*

$$\begin{aligned} W^{(t+1)} &= W^{(t)} + \mu_1[I_N - \boldsymbol{\varphi}(\boldsymbol{u})\boldsymbol{f}^T]W^{(t)} \\ \boldsymbol{v}^{(t+1)} &= \boldsymbol{v}^{(t)} - \mu_2\boldsymbol{\varphi}(\boldsymbol{u}_a)c. \end{aligned} \tag{6}$$

Here $W_{N+1} = [w_{1,N+1}, \cdots, w_{M,N+1}]^T \in \Re^M$, $\boldsymbol{\varphi}(\boldsymbol{u}) = [\varphi_1(u_1), \cdots, \varphi_N(u_N)]^T$, $\boldsymbol{\varphi}(\boldsymbol{u}_a) = [\varphi_1(u_1), \cdots, \varphi_M(u_M)]^T$, where $\varphi_i(u_i) = -\frac{dp_i(u_i)}{du_i}/p_i(u_i)$, $\boldsymbol{f} = W\boldsymbol{x}$, I_N is a $N \times N$ identity matrix, and μ_1 and μ_2 are learning rates that can be set differently.

Proof. If we assume that $u_1, \cdots, u_N, u_{N+1}$ are independent of each other, the log likelihood of the given data becomes

$$L(\boldsymbol{u}, c, \boldsymbol{W}) = \log|\det \boldsymbol{W}| + \sum_{i=1}^{N} \log p_i(u_i) + \log p(c), \tag{7}$$

because

$$p(\boldsymbol{x}, c) = |\det \boldsymbol{W}|\ p(\boldsymbol{u}, u_c) = |\det \boldsymbol{W}| \prod_{i=1}^{N} p_i(u_i)\ p(c). \tag{8}$$

Using the maximum likelihood estimation criterion, we are to maximize L, and this can be achieved by the steepest ascent method. Because the last term in (7) is a constant, differentiating (7) with respect to $\boldsymbol{W}$ leads to

$$\begin{aligned} \frac{\partial L}{\partial w_{i,j}} &= \frac{adj(w_{j,i})}{|\det \boldsymbol{W}|} - \varphi_i(u_i)x_j \qquad i = 1, \cdots, N,\ j = 1, \cdots, N \\ \frac{\partial L}{\partial v_i} &= -\varphi_i(u_i)c \qquad\qquad\qquad\qquad i = 1, \cdots, M \end{aligned} \tag{9}$$

where $adj(\cdot)$ is adjoint and $\varphi_i(u_i) = -\frac{dp_i(u_i)}{du_i}/p_i(u_i)$. Note that c has categorical values.

We can see that $|\det \boldsymbol{W}| = |\det W|$ and $\frac{adj(w_{j,i})}{|\det \boldsymbol{W}|} = W_{i,j}^{-T}$. Thus the learning rule becomes

$$\begin{aligned} \Delta W &\propto\ W^{-T} - \boldsymbol{\varphi}(\boldsymbol{u})\boldsymbol{x}^T \\ \Delta \boldsymbol{v} &\propto\ -\boldsymbol{\varphi}(\boldsymbol{u}_a)c. \end{aligned} \tag{10}$$

Since the two terms in (10) have different tasks regarding the update of separate matrix W and W_{N+1}, we can divide the learning process and applying natural gradient on updating W, we get (6).

Note that the learning rule for W is the same as the original ICA learning rule [3], and also note that $\boldsymbol{f}_a$ corresponds to the first M elements of $W\boldsymbol{x}$. Therefore, we can extract the optimal features by the proposed algorithm when it find the optimal solution by (6).

3 Stability Conditions

The local stability analysis in this paper undergoes almost the same procedure as that of general ICA algorithms in [4]. In doing so, to cope with troublesome terms, we modify the notation of the weight update rule for $\boldsymbol{v}$ in (6) a little bit with the following:

$$v_i^{(t+1)} = v_i^{(t)} - \mu_i^{(t)} \varphi_i(u_i) c v_i^* v_i^{(t)}, \quad i = 1, \cdots, M \tag{11}$$

where v_i^* is the optimal value for v_i. The learning rate $\mu_i^{(t)} (> 0)$ changes over time t and i, appropriately chosen to equalize $\mu_i^{(t)} v_i^* v_i^{(t)} = \mu_2$. This modification is justified because $v_i^* v_i^{(t)} (\simeq v_i^{*2})$ is positive when $v_i^{(t)}$ is near the optimal value v^*. In weight update rule (11) along with the update rule for W in (6), the local stability condition is obtained in the following theorem.

Theorem 2. *The stability of the stationary point of the proposed algorithm*

$$W = \Lambda \Pi A^{-1}, \quad \boldsymbol{v} = -\Lambda \Pi A^{-1} \boldsymbol{b} \tag{12}$$

is governed by the nonlinear moment

$$\kappa_i = \boldsymbol{E}\dot{\varphi}_i(e_i)\boldsymbol{E}e_i^2 - \boldsymbol{E}\varphi_i(e_i)e_i, \tag{13}$$

and it is stable if

$$1 + \kappa_i > 0, \quad 1 + \kappa_j > 0, \quad (1 + \kappa_i)(1 + \kappa_j) > 1 \tag{14}$$

for all (i, j) *pair. Thus the sufficient condition is*

$$\kappa_i > 0, \quad 1 \le i \le N. \tag{15}$$

Here, $E(\cdot)$ *is the expectation and* $\boldsymbol{e} = [e_1, \cdots, e_N]^T$ *is the scaled and permuted version of* $\boldsymbol{s}$ *(*$\boldsymbol{e} = \Lambda \Pi \boldsymbol{s}$*) which is estimated by* $\boldsymbol{u}$*.*

The proof of the algorithm is omitted because of the space limitation. Note that this is the same as the stability condition for standard ICA in [4].

4 Simulation Results

(Simple problem.) Suppose we have two input features x_1 and x_2 uniformly distributed on [-1,1] for a binary classification, and the output class y is determined as follows:

$$y = \begin{cases} 0 & \text{if } x_1 + x_2 < 0 \\ 1 & \text{if } x_1 + x_2 \ge 0 \end{cases}$$

Here, $y = 0$ corrsponds to $c = -1$ and $y = 1$ corresponds to $c = 1$.

This problem is linearly separable, and we can easily distinguish $x_1 + x_2$ as an important feature. But feature extraction algorithms based on conventional unsupervised learning, such as the conventional PCA and ICA, cannot extract $x_1 + x_2$ as a new feature because they only consider the input distribution; i.e., they only examine (x_1, x_2) space.

For this problem, we performed ICA-FX with $M = 1$ and could get $u_1 = 43.59x_1 + 46.12x_2 + 36.78y$ from which a new feature $f_1 = 43.59x_1 + 46.12x_2$ is obtained.

(Pima indian diabetes.) This data set consists of 768 instances in which 500 are class 0 and the other 268 are class 1. It has 8 numeric features with no missing value. This dataset is used in [5] to test their feature extraction algorithm MMI. We have compared the performance of ICA-FX with those of LDA, PCA, and MMI.

As in [5], we used LVQ-PAK[1] to train the data and separated the data into 500 training data and the other 268 as test data. The meta-parameters of LVQ-PAK were set to the same as those in [5].

[1] http://www.cis.hut.fi/research/software.shtml.

In Table 1, classification performances are presented. The classification rates of MMI, LDA, and PCA are from [5]. As shown in the table, the performance of ICA-FX is about 3∼10% better than those of the other methods. Even with one feature, the classification rate is as good as the cases where more features are used.

Table 1. Classification performance for Pima data

No. of features	Classification performance (%) (LVQ-PAK)			
	PCA	LDA	MMI	ICA-FX
1	64.4	65.8	72.0	81.3
2	73.0	–	77.5	81.7
4	74.1	–	78.5	81.5
6	74.7	–	78.3	81.0
8	74.7	–	74.7	81.0

5 Conclusions

In this paper, we have proposed an algorithm for feature extraction. The proposed algorithm is based on ICA and can generate appropriate features for classification problems.

In the proposed algorithm, we added class information in training ICA. The added class information plays a critical role in the extraction of useful features for classification. With the additional class information we can extract new features containing maximal information about the class. The number of extracted features can be arbitrarily chosen. We presented the justification of the proposed algorithm.

Since it uses the standard feed-forward structure and learning algorithm of ICA, it is easy to implement and train. Experimental results show that the proposed algorithm generates good features that outperform the original features and other features extracted from other methods for classification problems. The proposed algorithm is useful for two-class problems, and more works are needed to extend the proposed method for multi-class problems.

References

1. N. Kwak, C.-H. Choi, and C.-Y. Choi, "Feature extraction using ica," in *Proc. Int'l Conf. on Artificial Neural Networks 2001*, Vienna Austria, Aug. 2001, pp. 568–573.
2. T.M. Cover, and J.A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.
3. A.J. Bell and T.J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol 7, pp. 1129–1159, 1995.
4. J.-F. Cardoso, "On the stability of source separation algorithms," *Journal of VLSI Signal Processing Systems*, vol. 26, no 1/2, pp. 7–14, Aug. 2000,
5. K. Torkkola and W.M. Campbell, "Mutual information in learning feature transformations," in *Proc. Int'l Conf. Machine Learning*, Stanford, CA, 2000.