Class-Information-Incorporated Principal Component Analysis

Songcan Chen* Tingkai Sun

Dept. of Computer Science & Engineering,

Nanjing University of Aeronautics & Astronautics,

Nanjing, 210016, China

Abstract: Principal component analysis (PCA) is one of the most popular feature extraction methods in pattern recognition and can obtain a set of so-needed projection directions or vectors by maximizing the projected variance of a given training data set in an unsupervised learning way, meaning that PCA does not sufficiently use the class label of given data in feature extraction. In this paper, a class-information-incorporated PCA (CIPCA) is presented with two objectives: one is to sufficiently utilize a given class label in feature extraction and the other is to still follow the same simple mathematical formulation as PCA. The experimental results on 13 benchmark datasets show its feasibility and effectiveness.

Keywords: Principal component analysis (PCA); Class-information-incorporated PCA (CIPCA); Pattern recognition.

1. Introduction

Feature extraction from data or a pattern is a necessary step in pattern recognition and can raise generalization of subsequent classification and avoid notorious curse of dimensionality [1,2]. Among numerous feature extraction methods, PCA [1], also known as Karhunen-Loeve expansion, is one of the most popular, relatively effective and simple methods and widely used in face recognition and computer vision [3-9]. It can obtain a set of projection vectors or principal components for feature extraction from given training patterns through maximizing the variance of the projected patterns with aiming at representing original information as faithfully as possible. However, in its feature extraction for classification tasks, as an unsupervised method, PCA does not sufficiently use class labels of given patterns and its maximization to the variance of the projected patterns might not necessarily be in favor of discrimination among classes, thus naturally it likely loses some useful discriminating

* Corresponding author: Tel: +86-25-84892452; +86-25-84498069. E-mail: s.chen@nuaa.edu.cn (Songcan Chen) and suntingkai@126.com (Tingkai Sun)

information for classification. In order to use such class information as much as possible and more importantly still follow the same simple formulation as PCA, we here propose a CIPCA with PCA being its special case. Two major differences between CIPCA and PCA embody: 1) in seeking so-needed projection vectors, CIPCA uses jointly a pattern and its corresponding class label rather than just the single pattern as in PCA; 2) crucially in extracting features from the pattern with unknown class label and subsequent classification for it, four classification strategies are respectively designed and will be detailed in sections 2.2-2.3. The comparison experimental results on the 13 benchmark datasets used in [10] show that CIPCA outperforms PCA on most of the datasets.

The rest of this paper is organized as follows: In section 2, CIPCA is formulated. The classification results based on the features extracted respectively by CIPCA and PCA are presented in section 3. The conclusion will be drawn in section 4.

2 CIPCA

2.1 Finding projection vectors

Given a set of c-class training sample patterns $\mathbf{x}_i \in R^n$ and the corresponding class labels $\mathbf{y}_i \in \{0,1\}^c$, i=1, 2, …, N. Where $\mathbf{y}_i$ is encoded in 1-of-c ways, i.e., for the jth-class training pattern, just the jth component of $\mathbf{y}_i$ is set to 1, 0 otherwise. Now concatenating each pattern $\mathbf{x}_i$ and its label $\mathbf{y}_i$ to form an augmented column vector $\mathbf{z}_i=(\mathbf{x}_i^T, \mathbf{y}_i^T)^T$, i=1,2,…, N and let $U\in R^{(n+c)xd}$ be a projection matrix to be sought consisting of a set orthogonal unit vectors $\{\mathbf{u}_j\in R^{(n+c)}\}$, j=1,2,…,d, and it can be obtained by maximizing the following criterion $J(U)$ under the constraints of $\mathbf{u}_i^T\mathbf{u}_j$=0 for $i\neq j$, 1 for i=j.

$$J(U)=\mathrm{tr}(U^TC_{\mathbf{zz}}U) \quad (1)$$

Where $C_{\mathbf{zz}}=ZZ^T$, a sample covariance matrix for the sample set $\{\mathbf{z}_i=(\mathbf{x}_i^T, \mathbf{y}_i^T)^T, i=1,2,\ldots, N\}$. $Z=(\mathbf{z}_1, \mathbf{z}_2,\ldots,\mathbf{z}_N)$. For convenience of description, we do not here discriminate the centered $\mathbf{z}$ from the un-centered or original $\mathbf{z}$. Obviously, Eq. (1) is formally completely equivalent to the PCA formulation. So, we can simply obtain the U through the same process as PCA, i.e., solving the eigen-equation as follows:

$$C_{\mathbf{zz}}U=U\Lambda \quad (2)$$

Hence, the U is formed by the first d eigenvectors $\{\mathbf{u}_j, j=1,2,\ldots,d\}$, corresponding to the first d largest eigenvalues $\{\lambda_j, j=1,2,\ldots,d\}$, in a decreasing order. Where the number, d, of the projection vectors to be kept is determined in terms of $\sum_{j-1}^{d}\lambda_j/\sum_{j-1}^{n+c}\lambda_j \geq\alpha$ (set to 0.95 here both for CIPCA and PCA). And

Λ=diag($\lambda_1, \lambda_2, \ldots, \lambda_d$).

In order to examine whether such an introduction of class label to PCA is helpful for discrimination, let us take a 4-dimensional Iris dataset with 3 classes as an example from which we select its linearly non-separable two classes, class2 and class3, and then use the projection vectors obtained respectively by PCA and CIPCA to project the first 25 training patterns from each class to the first 2 principal components, the results are shown in Fig.1 from which we can observe that the separable effect between the two classes is enhanced. So it should be intuitively helpful for classification.

2.2 Feature extraction

For a pattern to be classified, we cannot directly use the U to extract so-needed features from it due to its unknown label. In order to perform an extraction task, we must decompose the found U into two parts $U_{\mathbf{x}}$ and U_{y} satisfying $U = (\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d) = \binom{U_{\mathrm{x}}}{U_{\mathrm{y}}}$, $\mathbf{u}_i = \binom{\mathbf{u}_{\mathrm{x}i}}{\mathbf{u}_{\mathrm{y}i}}$, i=1,2,..., d, $U_{\mathbf{x}} = (\mathbf{u}_{\mathrm{x}1}, \mathbf{u}_{\mathrm{x}2}, \cdots, \mathbf{u}_{\mathrm{x}d}) \in R^{n\times d}$ and $U_{\mathrm{y}} = (\mathbf{u}_{\mathrm{y}1}, \mathbf{u}_{\mathrm{y}2}, \cdots, \mathbf{u}_{\mathrm{y}d}) \in R^{c\times d}$. Notice that U is an orthogonal matrix but individual $U_{\mathbf{x}}$ and U_{y} need not be. Though so, we do not intend to orthogonalize them, instead we use them directly to extract features from a given pattern **x**. Now we can not only get so-needed features directly from **x** but also estimate its corresponding class label indirectly from its extracted features according to the following formulation:

For the **x**, let its unknown class label be **y**, the corresponding augmented vector **z** can be approximately represented by a set of the found principal components {$\mathbf{u}_j$, j=1,2,…,d} as

$$\mathbf{z} \approx \sum_{i=1}^{d} a_{zi}\mathbf{u}_i = \begin{pmatrix} \sum_{i=1}^{d} a_{\mathbf{z}i}\mathbf{u}_{\mathrm{x}i} \\ \sum_{i=1}^{d} a_{\mathbf{z}i}\mathbf{u}_{\mathrm{y}i} \end{pmatrix} = \begin{pmatrix} U_{\mathrm{x}} a_{\mathbf{z}} \\ U_{\mathrm{y}} a_{\mathbf{z}} \end{pmatrix} \approx \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \qquad (3).$$

Where $a_{\mathbf{z}}=(a_{\mathbf{z}1}, a_{\mathbf{z}2},\ldots, a_{\mathbf{z}d})^T$ is a column vector formed by the d projection coefficients. Now from (3), we have an equation $\mathbf{x} \approx U_{\mathbf{x}}a_{\mathbf{z}}$ and thus can obtain its least squared approximate solution to $a_{\mathbf{z}}$, i.e., $a_{\mathbf{z}}$ $=(U_{\mathbf{x}})^{+}\mathbf{x}$, and in turn substitute it into the equation $\mathbf{y} \approx U_{\mathrm{y}}a_{\mathbf{z}}$ to obtain the approximate class label $\hat{\mathbf{y}} = U_{\mathrm{y}} U_{\mathbf{x}}^{+}\mathbf{x}$ for **x**. Here $(U_{\mathbf{x}})^{+}$ is a Moore-Penrose generalized inverse matrix of $U_{\mathbf{x}}$**.** It is such an explicitly obtained estimation to the class label that makes CIPCA distinctly different from PCA where we cannot acquire such estimation.

2.3 Classification strategy

Though having obtained the class label estimation, we still need to determine the true class to which the unknown pattern belongs using the nearest neighbor classifier (1NN) based on Euclidean distance. Now facing the extracted features $a_{\mathbf{z}}=(U_{\mathbf{x}})^{+}\mathbf{x}$ and an estimated label $\hat{\mathbf{y}}$,we can in fact design four classification strategies all based on 1NN as follows: 1) just use $a_{\mathbf{z}}$ (S1); 2) just use $\hat{\mathbf{y}}$ (S2); 3) jointly use $a_{\mathbf{z}}$ and $\hat{\mathbf{y}}$ (S3); 4) majority voting (S4) based on the three classification results from S1, S2 and S3.

In order to further gain insight into the essence of CIPCA for classification, in the following, we derive their corresponding decision functions using 1NN classification rule for S1, S2 and S3 except for S4 because of non-existence of its analytical formulation.

1) In S1, for unknown pattern **x**, let $a_{\mathbf{z}}=(U_{\mathbf{x}})^{+}\mathbf{x}=a_{\mathbf{z}}(\mathbf{x})$ be its projection vector, we will make a decision of **x** belonging to the same class as its nearest neighbor $\mathbf{x}_i$ in the training set if the following inequality is satisfied:

$$\left\|a_Z(\mathbf{x})-a_Z(\mathbf{x}_i)\right\|<\left\|a_Z(\mathbf{x})-a_Z(\mathbf{x}_j)\right\| \quad \text{for all } j\neq i \text{ and } j=1, 2, \ldots, N. \qquad (4)$$

where ||.|| is Euclidean distance. The inequality (4) can be equivalent to (5) below after some mathematical manipulations:

$$(\mathbf{x}-\tfrac{\mathbf{x}_i+\mathbf{x}_j}{2})^T U_{\mathbf{x}}^{+} U_{\mathbf{x}}^{+T}(\mathbf{x}_i-\mathbf{x}_j)>0 \quad \text{for all } j\neq i \text{ and } j=1, 2, \ldots, N. \qquad (5)$$

For PCA, there is also a similar decision rule but using a different projection matrix, meaning that both will yield different results of classification for the same task. With a similar derivation to the above (5), we have

2) In S2, If $(\mathbf{x}-\tfrac{\mathbf{x}_i+\mathbf{x}_j}{2})^T U_{\mathbf{y}} U_{\mathbf{x}}^{+}(U_{\mathbf{y}} U_{\mathbf{x}}^{+})^T(\mathbf{x}_i-\mathbf{x}_j)>0$ for all $j\neq i$ and j=1, 2, …, N. (6)

then the **x** is decided to the same class as its nearest neighbor $\mathbf{x}_i$ in the training set.

3) In S3, a corresponding rule is

If $(\mathbf{x}-\tfrac{\mathbf{x}_i+\mathbf{x}_j}{2})^T [U_{\mathbf{x}}^{+} U_{\mathbf{x}}^{+T}+U_{\mathbf{y}} U_{\mathbf{x}}^{+}(U_{\mathbf{y}} U_{\mathbf{x}}^{+})^T](\mathbf{x}_i-\mathbf{x}_j)>0$ for all $j\neq i$ and j=1, 2, …, N. (7)

then the similar decision for **x** is made.

In summary, rules (5-7) induce different classifying methods and thus different performances of classification as shown in our experiments below. However, just from the above rules, we cannot claim directly which will yield better results of classification unless a series of necessary experimental verifications are made and supported.

3. Classification experiments

3.1 Dataset

We use 13 datasets (available at http://ida.first.fraunhofer.de/projects/bench/benchmarks.htm) used in [10] to perform comparison experiments. These datasets all contain 2 classes. We use the training and testing sets offered by the database and give a brief description as follows: 1) Image: (Dimension, Training set size, Testing set size)=(18, 1300,1010); 2) Splice: (60,1000,2175); 3)Banana: (2,400,4900); 4) B.-Cancer: (9,200,77); 5) Diabetis: (8,468,300); 6) F.-Solar: (9,666,400); 7) German: (20,700,300); 8) Heart: (13,170,100); 9) Ringnorm: (20,400,7000); 10) Thyroid: (5,140,75); 11) Titanic: (3,150,2051); 12) Twonorm: (20,400,7000); 13) Waveform: (21,400,4600).

3.2 Experimental results and analysis

In the experiments, we performed independently repeatedly 20 times and 100 times respectively for the first two datasets and for the latter 11 datasets and then averaged their classification accuracies for each dataset corresponding to four different classification strategies. The final classification accuracies are tabulated in Table 1. For a comprehensive comparison, we also list the corresponding classifying results obtained by linear discriminant analysis (LDA) [11] which is a supervised feature extraction method but implicitly uses class information without the class encoding. From Table 1, we can observe that PCA, CIPCA and LDA exhibit their own advantages: on all datasets, both PCA and CIPCA with strategy S1 have, on the whole, comparable classification performance. CIPCA with strategy S2 seems, on the average, better on 6 datasets in classification than the other methods. Different methods show different performances with relatively large changing range on some datasets, for example, on the datasets Image, Splice, Banana, B.-Cancer, Diabetis, Ringnorm and Thyroid, the respective highest accuracies (underlined) are respectively 96.02% in CIPCA with strategy S1, 83.37% in CIPCA with strategy S2, 86.36% in PCA and CIPCA with strategy S1, 67.13% in CIPCA with strategy S1, 75.57% in CIPCA with strategy S2, 74.78% in CIPCA with strategy S2 and 95.68% in PCA while the corresponding lowest accuracies are respectively 77.34% in CIPCA with strategy S2, 70.38% in CIPCA with strategy S1, 54.64% in CIPCA with strategy S2, 54.84% in CIPCA with strategy S2, 68.70% in LDA, 65.02% in CIPCA with strategy S1 and 82.55% in LDA. On the remaining datasets, their performance changes are relatively small. Surprisingly, though exhibiting better performance on the 6 datasets, CIPCA with strategy S2 behaves particularly badly on the Image, Banana and B.-Cancer datasets, and meanwhile though using implicitly class information, LDA seems to behave relatively badly as well at least 13 datasets used here probably due to the fact that it can *only* obtain one discriminating projection vector for classification for any 2-class problem. The experimental results here remind us that no method *always* possesses superiority in all datasets. Therefore, in order to obtain as good performance as possible, we must make a choice from various methods for various datasets or problems.

4. Conclusions

In this paper, we proposed a PCA incorporating class label information by concatenating pattern data and its class label. When removing the class label from CIPCA, CIPCA will reduce to PCA. Unlike LDA which uses implicitly the class information, CIPCA *explicitly* incorporates the class information. The experimental comparison is made and the corresponding results show that CIPCA is feasible, and like PCA and LDA, can serve as an alternative for feature extraction in pattern recognition and data analysis. It is worth noting that, although we only adopted the batch methods for CIPCA and PCA computation in our experiments, in practice we can borrow the idea from the Hebbian-based neural networks [12] to more effectively implement CIPCA with aiming at overcoming the batch methods' restrictions on storage and reducing their computational cost and in this way making CIPCA also applicable to build practical system such as face recognition ones which will be our next goal.

Acknowledgements

We thank the anonymous reviewers' constructive comments for improving presentation of this paper and also Natural Science of Jiangsu under Grant No. BK2005122，National Science Foundations of China under Grant No. 60473035, Jiangsu "QingLan" Project Foundation and the Returnee's Foundation of China Scholarship Council for partial supports respectively.

References

[1] I.T. Jolliffe, Principal Component Analysis, Springer-Verlag, New York, 1986.

[2] R. A. Duda, P. E. Hart and D. G. Stork, Pattern Classification, John Wiley & Sons, 2nd Edition, 2001.

[3] M. Kirby, L. Sirovich, Application of the Karhunen-Loeve procedure for the characterization of human faces, IEEE Trans. Patt. Analy. And Mach. Intell., 12 (1) (Jan. 1990) 103-108.

[4] M. Turk, A. Pentland, Eigenfaces for recognition, J. Cognitive neuroscience, 3 (1) (Jan. 1991) 71-86.

[5] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, Eigenfaces vs. Fisherfaces : Recognition using class specific linear projection, IEEE Trans. Pattern Analysis and Machine Intelligence, 19 (7) (July 1997) 711-720.

[6] L. Zhao, Y. Yang, Theoretical analysis of illumination in PCA-based vision systems, Pattern Recognition, 32 (4) (Apr. 1999) 547-564.

[7] R. Gottumukkal, V. K. Asari, An improved face recognition technique based on modular PCA approach, Pattern Recognition Letters, 25 (4) (Apr. 2004) 429-436.

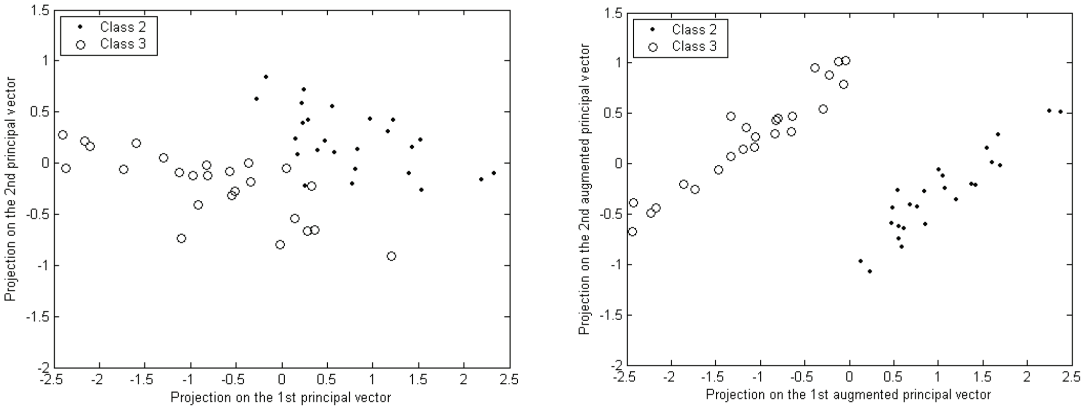
[8] S. C. Chen, Y. L. Zhu, Subpattern-based principal component analysis, Pattern Recognition, 37 (1) (Jan. 2004) 1081-1083.

[9] K. R. Tan, S. C. Chen, Adaptively Weighted Sub-pattern PCA for Face Recognition, Neurocomputing, 64 (Mar. 2005) 505-511.

[10] G. Ratsch, T. Onoda, K.-R. Müller, Soft margins for AdaBoost, Machine Learning, 42 (3) (Mar. 2001) 287-320.

[11] G. J. Mclachlan, Discriminant Analysis and Statistical Pattern Recognition, John Wiley & Sons, New York, 1992.

[12] S. Becker and M. Plumbley, Unsupervised neural network learning procedures for feature extraction and classification, Journal of Applied Intelligence, 6 (3) (July 1996) 185-205.



(a) Projection plot of PCA

(b) Projection plot of CIPCA

Fig. 1 Visual projection plots respectively for (a) PCA and (b) CIPCA for the two classes of Iris dataset

Table 1 Classification Accuracy (%) at $\alpha = 0.95$.

Dataset	PCA	S1	S2	S3	S4	LDA
Image	95.89	96.02	77.34	92.51	89.62	77.91
Splice	71.38	70.38	83.37	70.94	73.64	79.08
Banana	86.36	86.36	54.64	71.83	71.83	61.66
B.-Cancer	67.04	67.13	54.84	60.71	60.43	64.95
Diabetis	69.45	69.87	75.57	72.32	73.93	68.70
F.-Solar	60.91	60.94	62.75	62.81	62.81	61.10
German	70.07	70.03	75.64	70.75	73.60	67.77
Heart	77.02	76.76	82.59	77.20	79.83	76.88
Ringnorm	65.63	65.02	74.78	65.74	68.69	68.22
Thyroid	95.68	95.67	85.76	94.13	92.44	82.55
Titanic	67.00	67.00	67.69	73.66	73.66	67.04
Twonorm	93.45	93.11	97.32	93.30	94.64	96.47
Waveform	84.43	84.25	80.34	84.48	85.24	81.43