Feature selection using order statistics

Raymond Liu
Department of Computing
Imperial College of Science Technology and Medicine
United Kingdom
rl708@imperial.ac.uk

Duncan F. Gillies
Department of Computing
Imperial College of Science Technology and Medicine
United Kingdom
d.gillies@imperial.ac.uk

***Abstract*—One of the fundamental problems in statistical pattern recognition, particularly in face recognition and similar applications, is the intractably high number of dimensions used to represent the input data. Various dimensionality reduction techniques have been studied in recent literature. Many of these techniques, including Principle Component Analysis (PCA), can be divided into two parts: feature extraction and feature selection. Traditional methods for feature selection have been either naive or computationally expensive. In this report, a new noise-resistant method for feature selection based on order statistics is proposed. Experimental results for the selection of PCA features in face recognition show that the new feature selection algorithm gives superior or comparable performance compared to the traditional naive feature selection method in the presence of noise, especially when the number of classes is small compared to the number of training examples per class.**

***Index Terms*—Feature selection, order statistics, pattern recognition.**

ACKNOWLEDGEMENT

This research is supported by the Departmental Teaching Award of Department of Computing, Imperial College London. This award is funded by EPSRC, UK.

I. INTRODUCTION

A. Basic procedure

In classification tasks where input data is represented in (intractably) high dimensional spaces, dimensionality reduction is performed before classification in order to make the data manageable. The most popular dimensionality reduction technique is principal component analysis (PCA) ([1], [2]). In the context of PCA the smaller space is called the eigenspace, and it is represented by a basis that is the set of eigenvectors of the sample covariance matrix of the whole input data. In face recognition these eigenvectors are called eigenfaces, and these are the features available to us, out of which we would like to select a few. We hope that the few features we select will be the most salient ones that describe the data. Roughly, *salient features* means ones that represent the input data in such a way that allows for good inter-class discrimination. These are called *most discriminating features*.

For clarity of exposition, we will study the problem of selecting PCA features in this report. Though, we note that the new techniques proposed in this research is a general feature selection algorithm that is not restricted to PCA or face recognition. We use face recognition as an example application because of the abundant publicly available data, and we use PCA as an example feature extraction algorithm for its theoretical and implementational simplicity.

There is some ambiguity in the face recognition literature in what people mean by the word *feature*, e.g. between [3] and [1]. Here we will conform to the following meaning. A feature is a one-dimensional subspace spanned by a vector in the space in which the input training data is represented, e.g. a principal component (eigenface). This definition is consistent with [3].

B. Current methods for feature selection

Given a set Y of available features, we select a smaller set $X \in Y$ of features. Many classification systems which use PCA as a feature extraction algorithm, e.g. the Bayes plug-in classifier ([4]), employ the naive feature selection algorithm. That is, they select the first few eigenvectors of the sample covariance matrix of the whole data set (eigenfaces), i.e. the ones with the largest associated eigenvalues. This makes sense because those are the directions in the original input space that exhibits the largest variation in the input data. However, while this is sensible for spherical Gaussian data, it is not very suitable for ellipsoidal data with high eccentricity or non-Gaussian, nonlinear data. That is to say, the directions of largest variation in the input data may not be the ones that best discriminate classes in the data, for example when the classes are very flat and very close to each other in the input space. At first sight it seems that this problem could be resolved by not using PCA, but using LDA instead ([2]), since the largest LDA features are the directions which maximise between-class separation and within-class compactness. However, LDA requires inversion of the within-class covariance or scatter matrix, which has much lower rank than the dimensionality of the original input space in the applications we consider.[1] So the within-class covariance or scatter matrix is not invertible in the input space, and we cannot use LDA directly.

Currently, most of the more sophisticated feature selection methods define a criterion function $J(\cdot)$, and select the subset

[1]The within-class scatter matrix is given by

$$S_w = \sum_i \sum_j (x_{ij} - \bar{x_i})(x_{ij} - \bar{x_i})^T,$$

where x_{ij} is the jth data point in class i, and $\bar{x_i}$ is the sample mean of class i. It has rank $N - g$ where N is the total number of data points and g is the number of classes.

X of features that maximises $J(X)$ ([3]). The most popular choice for the criterion function $J(\cdot)$ is the classification error estimated by cross-validation or a similar technique. The aim is to try to find the optimal subset X of the full set of features Y by some search method. According to this paradigm, the first setback is that for a general criterion function $J(\cdot)$ based on classification error, there is no way to guarantee the optimal subset of features without exhaustive search ([5]). Jain et al. briefly describe some of the well-known search procedures in Section 4.2 of [3]. Another setback is that if the criterion function $J(\cdot)$ is based on classification error, we will have to re-evaluate the classification error for every subset X of features, whatever search strategy we use. This will lead to some computational overhead. Moreover, the classification error often cannot be reliably evaluated (e.g. by cross-validation), especially when the training sets are small. Even when we do have some confidence in our estimation of the classification error, different esitimates may be obtained depending on the classifier we use, and thus we may obtain different 'optimal' sets of features. However the latter may or may not be a problem.

II. Feature selection based on order statistics

A. Notation and setup

Let there be g classes, N_i data points in each class c_i, and N data points in total. The jth data point from the class c_i is x_{ij}, which is a d-dimensional column vector, i.e. a point in $\mathbb{R}^d$, the data space. For any feature extraction method, e.g. PCA, let the rank of the transformation matrix be p. We will assume that $p < d$. For PCA, assuming that the data points are in general position, we will have $p = N - 1$. We will call the space of dimensionality p the *transformed space*. Of these p features we wish to select m where $m < p$ (usually $m \ll p$). These m features will span another smaller subspace, which we call the *classification space*. The classification space is the one in which classification will be performed. Thus in short we have the input space, the transformed space which is obtained after feature extraction, and the classification space which is obtained after feature selection.

A data point x'_{ij} from class c_i in the transformed space will be a p-dimensional vector, with $x'_{ij} = P^{-1}x_{ij}$ where P is the change of basis matrix from the original data space to the transformed space. We will often omit the prime and just write x_{ij} for a data point in the transformed space, the context should make it clear what we mean.

B. Impurity and quality of variation (QoV)

Consider a class c_i of data points along the kth feature, i.e. the kth coordinates of the data points. If there is no data point x_j from another class c_j between any two data points in c_i, then we say that the class c_i is *clean*. Otherwise, we say that there is *overlap* between classes c_i and c_j. Now let's consider the orders of the data points along the kth feature.

For a class c_i and a feature l, let the projection orders of the points of this class along a feature l be $o_{i1}, \ldots, o_{iN_i}$. Note that $o_{i1}, \ldots, o_{iN_i}$ are scalar integers.

Definition 1. The *class order scatter* $\mathrm{OS}_l(c_i)$ of class c_i along a feature l is the quantity

$$\mathrm{OS}_l(c_i) := \sum_j (o_{ij} - \bar{o}_i)^2$$

where $\bar{o}_i$ is the sample mean order of class c_i along the feature l.

So the class order scatter along a feature l is almost the same as the usual *class scatter*[2], except we use data point orders instead of data point values. Now we take note of an interesting property of class order scatters. If a class c is clean along the feature l, (see above), we see that its data points $o_1, \ldots, o_{N_c}$ must be consecutive integers in some order, and so it can be easily shown that in this case $\mathrm{OS}_l(c) = \frac{N_c}{12}(N_c^2 - 1)$. If c is not clean along l, then clearly $\mathrm{OS}_l(c)$ will be larger. This minimum value property motivates our definition of class impurity, which is defined to be the normalised difference between the actual class order scatter and that of a clean class of the same size.

Definition 2. Let c_i be a class of data points of size N_i and let c be a set of consecutive integers that has the same size (equivalent to a clean class of the same size). The *class impurity* $\mathrm{imp}(c_i)$ of class c_i along a feature l is

$$\mathrm{imp}_l(c_i) := \frac{\mathrm{OS}_l(c_i) - \mathrm{OS}_l(c)}{N_i(N_i^2 - 1)} = \frac{\mathrm{OS}_l(c_i)}{N_i(N_i^2 - 1)} - \frac{1}{12}.$$

It can be easily seen that for any class c and along any feature l, we have $\mathrm{imp}_l(c) \geq 0$, with equality iff c is clean. Now that we have a definition of class impurity of one class along a feature, we want to consider the average, or weighted average, of the impurities of all classes along a feature.

C. Feature selection criterion

The general strategy is as follows. Given a set of p features, we calculate for each feature the average class impurity of all the classes along the said feature. The *quality of variation* (QoV) of a feature is then the inverse of the average impurity of all the classes along the feature. We select the m features with the best QoV, and those will form the basis of our classification space (see above). New test points to be classified will be transformed into that space, and then any classification method can be applied in that space to classify that point.

Definition 3. The *quality of variation* of a feature l is given by

$$\mathrm{QoV}(l) = \frac{1}{\sum_i \mathrm{imp}_l(c_i)/g},$$

where g is the number of classes.

For our experiments, the number of features we select will be $\min(\min_i(N_i - 1), g - 1)$. In short, the $\min_i(N_i - 1)$ is because we will be using Mahanalobis distance-based classifiers on the classification space, and we need to make sure that the sample class covariance matrix is invertible; the $g - 1$

[2] $S_i := \sum_j (x_{ij} - \bar{x}_i)^2$ where the x_{ij} are the data point values and $\bar{x}_i$ is the class sample mean.

Fig. 1. Example class of noiseless data.

Fig. 2. Example class of noisy data.

is a sensible choice because a scatter of g points in general position span $g-1$ dimensions.

III. Experiments

We test our new feature selection method on frontal face recognition with PCA feature extraction, as compared with the naive feature selection method. Recall that the naive PCA feature selection method selects the first m PCA components (see Section I-B). We test three learning algorithms: naive PCA feature selection with Mahalanobis distance, naive PCA feature selection with quadratic discriminant analysis (QDA), and our PCA feature selection based on order statistics with Mahalanobis distance.

To illustrate that our new feature selection method is resistant to noise, we will study two data sets of coloured frontal face images: one without added noise and another with added artificial noise. There are 25 images available per individual, and we will randomly select 5 to test and use the remaining 20 to train the classifiers. The added artificial noise comprises the conjunction of assigning a random colour to the background, occluding the left half of each face by a random colour, and finally occluding the top half of each face by a random colour. Figures 1 and 2 show a noiseless and a noisy example class respectively.

We look at the performance of the three algorithms mentioned above in terms of their classification accuracy as it varies with the number of classes. For each fixed number of classes, we perform the aforementioned cross-validation 10 times, and record the average accuracy of each algorithm and its standard deviation.

A. Results

For notational convenience we will make some abbreviations here. We will write MD for Mahalanobis distance. Our new feature selection method based on order statistics will be written as QoV selection. Tables I and II show the results for noiseless and noisy data respectively.

TABLE I
ACCURACY(STANDARD DEVIATION) FOR NOISELESS DATA. NMD = NAIVE SELECTION WITH MAHALANOBIS DISTANCE, NQDA = NAIVE SELECTION WITH QDA, QOVMD = QOV SELECTION WITH MAHALANOBIS DISTANCE.

	Number of classes		
	4	8	12
NMD	93.5%(6.34)	91%(4.77)	90.3%(4.14)
NQDA	95.5%(5.67)	90.5%(4.30)	91%(4.42)
QoVMD	93.5%(6.34)	90%(4.47)	89.8%(3.91)

TABLE II
ACCURACY(STANDARD DEVIATION) FOR DATA WITH ADDED ARTIFICIAL NOISE. NMD = NAIVE SELECTION WITH MAHALANOBIS DISTANCE, NQDA = NAIVE SELECTION WITH QDA, QOVMD = QOV SELECTION WITH MAHALANOBIS DISTANCE.

	Number of classes		
	4	8	12
NMD	5%(3.16)	55.5%(6.96)	57.8%(2.79)
NQDA	4.5%(3.5)	58%(7.23)	60%(4.08)
QoVMD	79%(8.31)	72%(7.14)	64.2%(5.44)

	Number of classes	
	16	20
NMD	52.1%(3.63)	30.4%(5.2)
NQDA	54.6%(3.63)	30.7%(5.25)
QoVMD	52.4%(5.85)	28.5%(2.94)

The results shown in Table I indicate that for noiseless data, there is no significant difference in performance between QoV selection and naive selection of features. This is not surprising, since for noiseless data, we expect that the 'best' PCA components to be the the first few, i.e. the ones that exhibit the most variation in the data. These results also serve to support the claim that QoV selection does indeed select the most discriminating features. Indeed, we find that the actual PCA features selected by our QoV principle in this noiseless experiment are exactly the first few PCA components.

The results in Table II are more interesting. Here there is added artificial noise as described previously. We see that in the cases where the number of classes is small (4 and 8), QoV selection performs significantly better than traditional naive selection. A deeper look at the exact features selected by our QoV principle elucidates what is going on. In a training run with 8 classes for example, the features selected by QoV selection, in the order of most discriminating first (i.e. decreasing QoV), are: 5, 4, 7, 6, 9, 11, and 13, where each number i represents the ith PCA component. Obviously,

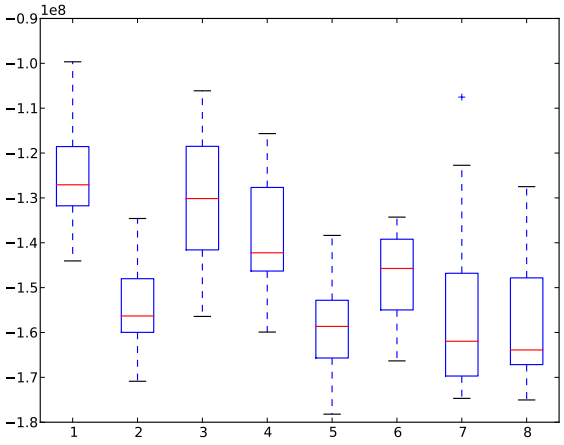

Fig. 3. Sample distribution of data along the 5th PCA component, where each class is represented by a box plot.

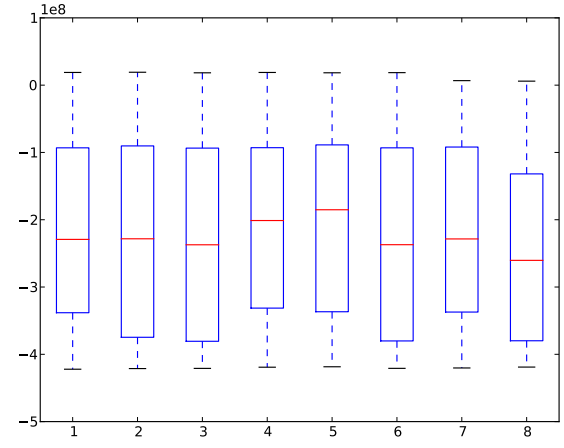

Fig. 4. Sample distribution of data along the 1st PCA component, where each class is represented by a box plot.

naive method selected features 1, . . ., 7. So the two selection methods only agree on features 4, 5, 6 and 7. Notice that QoV selection did not choose the first three PCA components to use because they predominantly exhibit only noise. Figures 3 and 4 illustrate how some features may have much better QoV than others. Figure 3 shows how the classes are spread along the 5th PCA component, which is the most discriminating feature according to our QoV principle. Each class is represented by a box plot. In contrast to Figure 4 which shows the class spread along the 1st PCA component, we can clearly see that the 5th component contains much more discriminating information about classes than does the 1st component, despite the fact that the 1st component exhibits considerably more variation. This is because the variation along the 1st PCA component is largely due to noise.

IV. Discussion and related research

A. Complexity

The reduction from input space to PCA eigenspace takes $\mathcal{O}(Nd^2)$ time ([6]), where N is the total number of data points and d is the dimensionality of the input space. In our applications we always have $d > N$ (in fact usually $d \gg N$), so PCA is at least $\mathcal{O}(N^3)$. Even if we do not use PCA, most other effective feature extraction methods are at least $\mathcal{O}(N^3)$ complex. Our QoV feature selection method based on order statistics requires to sort the data points along each of the axes in the transformed space (e.g. the PCA space). If we use a feature extraction method that extracts $\mathcal{O}(N)$ features (PCA extracts $\leq N-1$), we need to sort the data points $\mathcal{O}(N)$ times. Each sort has complexity $\mathcal{O}(N \log N)$. Thus the total computational overhead of our feature extraction algorithm using QoV is $\mathcal{O}(N^2 \log N)$, which is smaller than $\mathcal{O}(N^3)$. Thus the total complexity of performing both feature extraction and selection would be $\mathcal{O}(N^3) + \mathcal{O}(N^2 \log N) = \mathcal{O}(N^3)$

B. Relation to the small sample size problem

Classifiers based on distance metrics that rely on individual class covariances, such as the Mahalanobis distance and QDA, suffer from the *small sample size* problem [7]. When the available training data is small or not much bigger compared to the dimensionality of the classification space, either the inversion of the class covariance matrices is impossible because they're singular, or the estimate of those matrices is *unstable*.[3] A natural remedy for the small sample size problem is to have more training data, but that is not always available. A popular paradigm of alternative methods for getting around the problem involve modifying the estimated covariance matrix so that the severely underestimated variances (the unstable ones) are increased ([4], [8], [9], [10]). In all of these methods, the general approach is to artificially increase or create variance where there is little or no variance evident from the training data, because the training data set is too small. This is based on some often restrictive assumptions about the shape of the distribution of the classes.

We can think of QoV feature selection as a different, non-parametric way of getting around the small sample size problem. With the naive feature seletion method, a relatively high number of dimensions (features) need to be retained for the classification space in order to achieve satisfactory classfication accuracy, especially in the presence of noise. But with QoV selection, the results in Section III-A imply that we can afford to select fewer features and still obtain similar or superior classification accuracy. Indeed, from Table II we see that QoV selection gives similar or superior performance when it selects the same number of features as the naive method, and there are cases where the difference in performance is large. This means that the classfication space will have a smaller

[3] An *unstable* estimate is one that is vastly different from the true class covariance matrix. It occurs when a class does not have enough data points to span the full classification space, in particular when the number of data points in a class is not bigger than the dimensionality of the classification space.

dimensionality, and thus may avoid the small sample size problem. The idea is that if we choose the right features to use, we would not need a lot of them in order to achieve satisfactory classification accuracy.

REFERENCES

[1] M. Turk, "A random walk through eigenspace," *IEICE Transactions on Information and Systems E series D*, vol. 84, no. 12, pp. 1586–1595, 2001.

[2] A. Martinez and A. Kak, "Pca versus lda," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 2, pp. 228 –233, Feb. 2001.

[3] A. Jain, R. Duin, and J. Mao, "Statistical pattern recognition: a review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 1, pp. 4 –37, Jan. 2000.

[4] C. Thomaz, "Maximum entropy covariance estimate for statistical pattern estimation," Ph.D. dissertation, Imperial College of Science Technology and Medicine, London, United Kingdom, 2004.

[5] T. M. Cover and J. M. Van Campenhout, "On the possible orderings in the measurement selection problem," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 7, no. 9, pp. 657 –661, Sep. 1977.

[6] Q. Du and J. Fowler, "Low-complexity principal component analysis for hyperspectral image compression," *International Journal of High Performance Computing Applications*, vol. 22, no. 4, p. 438, 2008.

[7] C. Thomaz and D. Gillies, ""Small sample size": a methodological problem in Bayes plug-in classifier for image recognition," Imperial College of Science Technology and Medicine, London, United Kingdom, Tech. Rep., 2001.

[8] C. Thomaz, "Using mixture covariance matrices to improve face and facial expression recognitions," *Pattern Recognition Letters*, vol. 24, no. 13, pp. 2159–2165, Sep. 2003.

[9] J. Hoffbeck and D. Landgrebe, "Covariance matrix estimation and classification with limited training data," *Pattern Analysis and Machine*, vol. 18, no. 7, pp. 763–767, 2002.

[10] J. Friedman, "Regularized discriminant analysis," *Journal of the American statistical association*, vol. 84, no. 405, pp. 165–175, 1989.