2013 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 22–25, 2013, SOUTHAMPTON, UK

LEARNING INCOHERENT SUBSPACES FOR CLASSIFICATION VIA SUPERVISED ITERATIVE PROJECTIONS AND ROTATIONS

Daniele Barchiesi and Mark D. Plumbley[*]

Centre for Digital Music
Queen Mary University of London
Mile End Road, London E1 4NS, UK

ABSTRACT

In this paper we present the supervised iterative projections and rotations (S-IPR) algorithm, a method to optimise a set of discriminative subspaces for supervised classification. We show how the proposed technique is based on our previous unsupervised iterative projections and rotations (IPR) algorithm for incoherent dictionary learning, and how projecting the features onto the learned sub-spaces can be employed as a feature transform algorithm in the context of classification. Numerical experiments on the FISHERIRIS and on the USPS datasets, and a comparison with the PCA and LDA methods for feature transform demonstrates the value of the proposed technique and its potential as a tool for machine learning.

Index Terms— Feature transforms, sparse approximation, dictionary learning, supervised classification.

1. INTRODUCTION: CLASSIFICATION AND FEATURE TRANSFORM

Supervised classification is one of the classic problems in machine learning where a system is designed to discriminate the category of an observed signal, having previously observed representative examples from the considered classes [1].

Typically, a classification algorithm consists of a training phase where class-specific models are learned from labelled samples, followed by a testing phase where unlabelled data are classified by comparison with the learned models. Both training and testing comprise various stages. Firstly, we observe a signal that measures a process of interest, such as the recording of a sound or image, or a log of the temperatures in a particular geographic area. Then, a set of features are extracted from the raw signals using signal processing techniques. This step is performed in order to reduce the dimensionality of the data and provide a new signal that allows generalisation among examples of the same class, while retaining enough information to discriminate between different classes.

Following the features extraction step, a feature transform can be employed to further reduce the dimensionality of the data and to enhance discrimination between classes. Thus classification benefits from feature transforms especially when features are not separable, that is, when it is not possible to optimise a simple function that maps features belonging to signals of a given class to the corresponding category. A further dimensionalty reduction may be performed when dealing with high dimensional signals (such as audio or high resolution images) by fitting the parameters of global statistical distributions with features learned on portions of the signal. Models learned on different classes are finally compared using a distance metric to the model learned form an unlabelled signal, which is typically assigned to the nearest class.

Standard methods for feature transform will be briefly reviewed in Section 2, as their limitations lead to the main motivation for this work. To provide the context of the proposed method, the incoherent dictionary learning problem will be introduced in Section 3, while Section 4 will contain the main contribution of this paper consisting in learning incoherent subspaces for classification. Numerical experiments are presented in Section 5, and conclusions are drawn in Section 6.

2. ALGORITHMS FOR FEATURE TRANSFORM

Two of the main feature transform techniques include principal component analysis (PCA) [2] and Fisher's linear discriminant analysis (LDA) [1]. This section provides a brief description of their rationale without explicitly derive their expressions to avoid overcrowding the paper.

Let $\left\{\boldsymbol{x}_m \in \mathbb{R}^N\right\}_{m=1}^{M}$ be a set of vectors containing features extracted from M training signals. The goal of PCA is to learn an orthonormal set of basis functions $\left\{\boldsymbol{\phi}_k \in \mathbb{R}^N\right\}_{k=1}^{N}$ such that $||\boldsymbol{\phi}_k||_2 = 1$ and $\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle = 0 \; \forall i \neq j$ that are placed along the columns of a so-called *dictionary* $\boldsymbol{\Phi} \in \mathbb{R}^{N\times N}$. The bases are optimised from the data to identify their principal components, that is, the sub-spaces that retain the maximum

[*]This work has been supported by the Queen Mary University of London School Studentship, the Platform Grant EP/K009559/1 and the Leadership Fellowship EP/G007144/1, both from the UK Engineering and Physical Sciences Research Council (EPSRC).

variance of the features.

Let $\Psi \overset{\text{def}}{=} \Phi_{1:L} \in \mathbb{R}^{N\times L}$ denote the sub-dictionary constructed from the $L < N$ principal components of the dataset. A new set of transformed features $\boldsymbol{y}_{\text{PCA}} = \Psi\Psi^T\boldsymbol{x}$ is computed by projecting the data onto the sub-space spanned by the columns of $\Phi_{1:L}$ (that is, onto the L-dimensional principal sub-space). This operation reduces the dimensionality of the features by projecting them onto a linear subspace embedded in $\mathbb{R}^N$. It is an unsupervised technique that does not exploit knowledge about the classes associated with the training set, but implicitly relies in the assumption that the principal component directions encode relevant differences between classes.

On the contrary, LDA is a supervised method for feature transform whose objective is to explicitly maximise the separability of classes in the transformed domain. The within-classes scatter matrix is defined to measure how features belonging to the same class are clustered around their mean vector (the vector obtained by averaging all observations associated with a given category), and the between-classes scatter matrix is defined to measure the distances between the mean vectors. A square linear transform $\boldsymbol{M} \in \mathbb{R}^{N\times N}$ is optimised to obtain a new set of features $\boldsymbol{y}_{\text{LDA}} = \boldsymbol{M}\boldsymbol{x}$ according to an objective that promotes features belonging to the same class to be near each other (according to a Euclidean distance), and far apart from features belonging to different classes. LDA explicitly seeks to enhance the discriminative power of features, but does not perform dimensionality reduction.

The method proposed in this paper is aimed at learning discriminative sub-spaces that allow dimensionality reduction, while at the same time enhancing the separability between classes. It is derived from our previous work on learning incoherent dictionaries for sparse approximation.

Related works that extend both PCA and LDA include the supervised PCA proposed by Barshan et al. [3], and methods for manifold learning reviewed by Van Der Maaten et al. [4]. Finally, the sparse sub-space clustering technique developed by Elhamifar and Vidal [5] applies concepts and algorithm from the field of sparse approximation to tackle unsupervised clustering problems.

3. INCOHERENT DICTIONARY LEARNING

A sparse approximation of a signal $\boldsymbol{x} \in \mathbb{R}^N$ is a linear combination of $K \geq N$ basis functions $\left\{\boldsymbol{\phi}_k \in \mathbb{R}^N\right\}_{k=1}^K$ called *atoms* described by:

$$\boldsymbol{x} \approx \tilde{\boldsymbol{x}} = \sum_{k=1}^{K} \alpha_k \boldsymbol{\phi}_k \tag{1}$$

where the vector of coefficients $\boldsymbol{\alpha}$ contains a *small* number of non-zero components, corresponding to a small number of atoms actively contributing to the approximation $\tilde{\boldsymbol{x}}$. Given a signal $\boldsymbol{x}$ and a dictionary, various algorithms have been proposed to find a sparse approximation that minimises the residual error $||\boldsymbol{x} - \tilde{\boldsymbol{x}}||_2$[6].

Dictionary learning aims at optimising a dictionary Φ for sparse approximation given a set of training data. It is an unsupervised technique that can be thought as being a generalisation of PCA, as both methods learn linear subspaces that minimise the approximation error of the signals. Dictionary learning, however, is generally more flexible than PCA because it can be employed to learn more general non-orthogonal over-complete dictionaries [7].

3.1. The incoherent dictionary learning problem

Dictionaries for sparse approximation have important intrinsic properties that describe the relations between their atoms, like the mutual coherence $\mu(\Phi) = \max_{i\neq j}\langle\boldsymbol{\phi}_i, \boldsymbol{\phi}_j\rangle$ that is defined as the maximum inner product between any two different atoms. The goal of incoherent dictionary learning is to learn atoms that are well adapted to sparsely approximate a set of training signals, and that are at the same time mutually incoherent [8].

Given a set of M training signals contained in the columns of the matrix $\boldsymbol{X} \in \mathbb{R}^{N\times M}$ and a matrix $\boldsymbol{A} \in \mathbb{R}^{K\times M}$ indicating the sparse approximation coefficients, the incoherent dictionary learning problem can be expressed as:

$$\begin{aligned} \Phi^\star = \arg\min_{\Phi} \; & ||\boldsymbol{X} - \Phi\boldsymbol{A}||_{\text{F}} \\ \text{such that } & \mu(\Phi) \leq \mu_0 \\ & ||\boldsymbol{\alpha}_m||_0 \leq S \quad \forall m \end{aligned} \tag{2}$$

where μ_0 is a fixed mutual coherence constraint, the ℓ_0 pseudo-norm $||\cdot||_0$ counts the number of non-zero components of its argument and S is a fixed number of active atoms. Algorithms for (incoherent) dictionary learning generally follow an alternate optimisation heuristic, iteratively updating Φ and $\boldsymbol{A}$ until a stopping criterion is met. In the case of the iterative projections and rotations algorithm (IPR) algorithm [8], a dictionary de-correlation step is added after updating the dictionary in order to satisfy the mutual coherence constraint.

Given $\boldsymbol{X}$, fixed μ_0, S and a stopping criterion (such as a maximum number of iterations), the optimisation of (2) is tackled by iteratively performing the following steps:

- *Sparse coding*: fix Φ and compute the matrix $\boldsymbol{A}$ using a suitable sparse approximation method.
- *Dictionary update*: fix $\boldsymbol{A}$ and update Φ using a suitable method for dictionary learning.
- *Dictionary de-correlation*: given $\boldsymbol{X}$, Φ and $\boldsymbol{A}$ update the dictionary Φ to reduce its mutual coherence under the level μ_0.

3.2. The iterative projections and rotations algorithm

The IPR algorithm has been proposed in order to solve the dictionary de-correlation step, while ensuring that the updated dictionary provides a sparse approximation with low residual norm, as indicated by the objective function (2) [8].

The IPR algorithm requires the calculation of the Gram matrix $\boldsymbol{G} = \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ which contains the inner products between any two atoms in the dictionary. $\boldsymbol{G}$ is iteratively projected onto two constraint sets, namely the structural constraint set $\mathcal{K}_{\mu_0}$ and the spectral constraint set $\mathcal{F}$. The former is the set of symmetric square matrices with unit diagonal values and off-diagonal values with magnitude smaller or equal than μ_0:

$$\mathcal{K}_{\mu_0} \overset{\text{def}}{=} \left\{ \boldsymbol{K} \in \mathbb{R}^{K \times K} : \boldsymbol{K} = \boldsymbol{K}^T, k_{i,i} = 1, \max_{i>j} |k_{i,j}| \leq \mu_0 \right\}.$$

The latter is the set of symmetric positive semidefinite square matrices with rank smaller than or equal to N:

$$\mathcal{F} \overset{\text{def}}{=} \left\{ \boldsymbol{F} \in \mathbb{R}^{K \times K} : \boldsymbol{F} = \boldsymbol{F}^T, \text{eig}(\boldsymbol{F}) \geq \boldsymbol{0}, \text{rank}(\boldsymbol{F}) \leq N \right\}$$

where the operator $\text{eig}(\cdot)$ returns the vector of eigenvalues of its argument.

Starting from the Gram matrix of an initial dictionary $\boldsymbol{\Phi}$, the IPR method iteratively performs the following operations.

- *Projection onto the structural constraint set.* The projection $\boldsymbol{K} = \mathcal{P}_{\mathcal{K}_{\mu_0}}(\boldsymbol{G})$ can be obtained by:

  1. setting $k_{i,i} = 1$,
  2. limiting the off-diagonal elements so that, for $i \neq j$,

  $$k_{i,j} = \text{Limit}(g_{i,j}, \mu_0) = \begin{cases} g_{i,j} & \text{if} \quad |g_{i,j}| \leq \mu_0 \\ \text{sgn}(g_{i,j})\mu_0 & \text{if} \quad |g_{i,j}| > \mu_0 \end{cases} \tag{3}$$

- *Projection onto the spectral constraint set and factorization.* The projection $\boldsymbol{F} = \mathcal{P}_{\mathcal{F}}(\boldsymbol{G})$ and subsequent factorisation are obtained by:

  1. calculating the eigenvalue decomposition (EVD) $\boldsymbol{G} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$,
  2. thresholding the eigenvalues by keeping only the N largest positive ones.

  $$[\text{Thresh}(\boldsymbol{\Lambda}, N)]_{i,i} = \begin{cases} \lambda_{i,i} & \text{if} \quad i \leq N \text{ and } \lambda_{i,i} > 0 \\ 0 & \text{if} \quad i > N \text{ or } \lambda_{i,i} \leq 0 \end{cases}$$

  where the eigenvalues in $\boldsymbol{\Lambda}$ are ordered from the largest to the smallest. Following this step, at most N eigenvalues of the Gram matrix are different from zero,

  3. factorizing the projected Gram matrix into the product $\boldsymbol{G} = \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ by setting:

  $$\boldsymbol{\Phi} = \boldsymbol{\Lambda}^{1/2} \boldsymbol{Q}^T. \tag{4}$$

- *Dictionary rotation.* Rotate the dictionary $\boldsymbol{\Phi}$ to align it to the training set by solving the problem:

  $$\boldsymbol{W}^\star = \underset{\boldsymbol{W}\boldsymbol{W}^T = \boldsymbol{I}}{\arg\min} \ ||\boldsymbol{X} - \boldsymbol{W}\boldsymbol{\Phi}\boldsymbol{A}||_{\text{F}}. \tag{5}$$

  The optimal rotation matrix can be calculated by:

  1. computing the sample covariance between the observed signals and their approximations $\boldsymbol{C} \overset{\text{def}}{=} (\boldsymbol{\Phi}\boldsymbol{A})\boldsymbol{X}^T$,
  2. calculating the SVD of the covariance $\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$,
  3. setting the optimal rotation matrix to $\boldsymbol{W}^\star = \boldsymbol{V}\boldsymbol{U}^T$,
  4. rotating the dictionary $\boldsymbol{\Phi} \leftarrow \boldsymbol{W}^\star \boldsymbol{\Phi}$.

More details about the IPR algorithm can be found in [8], including details of its computational cost.

4. LEARNING INCOHERENT SUBSPACES

The IPR algorithm learns a dictionary where all the atoms are mutually incoherent. Therefore, given any two disjoint sets $\Lambda \bigcap \Gamma = \emptyset$ that identify non-overlapping collections of atoms, the sub-dictionaries $\boldsymbol{\Phi}_\Lambda, \boldsymbol{\Phi}_\Gamma$ are also mutually incoherent.

Starting from this observation, the main intuition driving the development of a supervised IPR (S-IPR) algorithm for classification is to learn mutually incoherent sub-dictionaries that approximate features from different classes of signals. The sub-dictionaries are in turn used to define incoherent subspaces, and features are projected onto these sub-spaces yielding discriminative dimensionality reduction.

4.1. The supervised IPR algorithm

Let $\{c_m \in \mathcal{C}\}_{m=1}^M$, $\mathcal{C} = \{C_1, C_2, \dots, C_P\}$ be a set of labels that identify the category of the vectors of features $\boldsymbol{x}_m$, whose elements belong to a set $\mathcal{C}$ of P possible categories. The columns of the matrix $\boldsymbol{X}_p$ contain a selection of the features extracted from signals belonging to the p-th category.

To learn incoherent sub-dictionaries from the entire set of features, we must first cluster the atoms to different classes[1], and then only proceed with their de-correlation if they are assigned to different categories (while allowing coherent atoms to approximate features from the same class). To this aim, we employ the matrix $\boldsymbol{A}$ to measure the contribution of every atom to the approximation of features belonging to each class.

Let $\boldsymbol{\alpha}_p^k$ indicate the k-th row of the matrix $\boldsymbol{A}_p$ containing the coefficients that contribute to the approximation of $\boldsymbol{X}_p$, and N_p indicate the number of its elements. A coefficient $\gamma_{k,p}$ is defined as:

$$\gamma_{k,p} \overset{\text{def}}{=} \frac{1}{N_p} \left|\left|\boldsymbol{\alpha}_p^k\right|\right|_1, \tag{6}$$

[1] Note that the term *cluster* implies that a this stage the algorithm needs to make an unsupervised decision, since there is no any a-priori reason to assign a given atom to any particular class.

and every atom $\boldsymbol{\phi}_k$ is associated with the category to which it maximally contributes $p_k^\star = \arg\max_p \{\gamma_{k,p}\}$.

Grouping together atoms that have been assigned to the same class leads to a set of sub-dictionaries whose size and rank depends on the number of atoms for each class, and to their linear dependence. As a general heuristic, if features corresponding to different classes do not occupy the same sub-space (according to the active elements in $\boldsymbol{A}$), a full-rank dictionary $\boldsymbol{\Phi}$ with $K \geq N \gg P$ ensures that $p_k^\star$ identify P non-empty and disjoint sub-dictionaries $\{\boldsymbol{\Phi}_p\}_{p=1}^P$.

Once the atoms have been clustered, the Gram matrix $\boldsymbol{G}$ is computed and iteratively projected as in the method described in Section 3.2, with the difference that equation (3) is modified in order to only constraint the mutual coherence between atoms assigned to different categories

$$\text{Limit}(g_{i,j}, \mu_0, \boldsymbol{p}^\star) = \begin{cases} g_{i,j} & \text{if} \quad |g_{i,j}| \leq \mu_0 \,\text{or}\, p_i^\star = p_j^\star \\ \text{sgn}(g_{i,j})\mu_0 & \text{if} \quad |g_{i,j}| > \mu_0 \,\text{and}\, p_i^\star \neq p_j^\star \end{cases} \tag{7}$$

A further modification of the standard IPR algorithm presented in [8] consists in the update of the Gram matrix, performed by computing its element-wise average with the projection $\boldsymbol{K} = \mathcal{P}_{\mathcal{K}_{\mu_0}}(\boldsymbol{G})$ (rather than by using the projection alone). This heuristic has led to improved empirical results by preventing $\boldsymbol{G}$ from changing too abruptly.

The complete supervised S-IPR method is summarised in Algorithm 1. Note that the mutual coherence $\mu_{p^\star}(\boldsymbol{\Phi}) = \arg\max_{p_i^\star \neq p_j^\star} \langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle$ indicated in this algorithm measures the inner product between any two atoms assigned to different categories since atoms assigned to the same category are allowed to be mutually coherent.

4.2. Classification via incoherent subspaces

The S-IPR algorithm allows to learn a set of sub-dictionaries $\{\boldsymbol{\Phi}_p\}$ that contain mutually incoherent atoms. These cannot be directly used to define discriminative subspaces because, depending on N and on the rank of each sub-dictionary, atoms belonging to disjoint sub-dictionaries might span identical subspaces. Instead, we fix a rank $Q \leq \lfloor N/P \rfloor$ and choose a collection of Q linearly independent atoms from each sub-dictionary $\boldsymbol{\Phi}_p$, using the largest values of $\gamma_{k,p}$ to define a picking order. Thus, we obtain a set $\{\boldsymbol{\Psi}_p\}_{p=1}^P$ of incoherent sub-spaces of rank Q embedded in the space $\mathbb{R}^N$, and use them to derive a feature transform for classification.

Each feature vector $\boldsymbol{x}_m$ that belongs to the class c_m is projected onto the relative subspace, yielding a set of transformed features $\{\boldsymbol{y}_m\}_{m=1}^M$.

$$\boldsymbol{y}_m = \boldsymbol{\Psi}_{c_m} \boldsymbol{\Psi}_{c_m}^\dagger \boldsymbol{x}_m \tag{8}$$

where $\boldsymbol{\Psi}^\dagger$ denotes the Moore-Penrose pseudo-inverse of the matrix $\boldsymbol{\Psi}$ and needs to be used in place of the transposition operator because the columns of $\boldsymbol{\Psi}$ are in general not orthogonal.

Algorithm 1: Supervised IPR

```
Input: $\boldsymbol{X}, \boldsymbol{\Phi}, \boldsymbol{A}, \mu_0, \boldsymbol{c}, I$
Output: $\boldsymbol{\Phi}^\star$
$i \leftarrow 1$;
// Cluster atoms
$\boldsymbol{A}_p \leftarrow [\boldsymbol{\alpha}_j] \forall j \in C_p$;
$\gamma_{k,p} \leftarrow \left|\left|\boldsymbol{\alpha}_p^k\right|\right|_1 / N_p$;
$p_k^\star = \arg\max_p \{\gamma_{k,p}\}$;
while $i \leq I$ and $\mu_{p^\star}(\boldsymbol{\Phi}) > \mu_0$ do
    // Calculate Gram matrix
    $\boldsymbol{G} \leftarrow \boldsymbol{\Phi}^T \boldsymbol{\Phi}$;
    // Project onto structural c.s.
    $\text{diag}(\boldsymbol{K}) \leftarrow \mathbf{1}$;
    $\boldsymbol{K} \leftarrow \text{Limit}(\boldsymbol{G}, \mu_0, \boldsymbol{p}^\star)$;
    $\boldsymbol{G} \leftarrow \frac{1}{2}\boldsymbol{G} + \frac{1}{2}\boldsymbol{K}$;
    // Project onto spectral c.s.  and
       factorize
    $[\boldsymbol{Q}, \boldsymbol{\Lambda}] \leftarrow \text{EVD}(\boldsymbol{G})$;
    $\boldsymbol{\Lambda} \leftarrow \text{Thresh}(\boldsymbol{\Lambda}, N)$;
    $\boldsymbol{\Phi} \leftarrow \boldsymbol{\Lambda}^{1/2} \boldsymbol{Q}^T$;
    // Rotate dictionary
    $\boldsymbol{C} \leftarrow \boldsymbol{X}(\boldsymbol{\Phi}\boldsymbol{A})^T$;
    $[\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}] \leftarrow \text{SVD}(\boldsymbol{C})$;
    $\boldsymbol{W} \leftarrow \boldsymbol{V}\boldsymbol{U}^T$;
    $\boldsymbol{\Phi} \leftarrow \boldsymbol{W}\boldsymbol{\Phi}$;
    $i \leftarrow i + 1$;
end
```

When an unlabelled signal is presented to the classifier, the corresponding vector of features $\boldsymbol{x}$ is projected onto all the learned sub-spaces. Then, the nearest sub-space is chosen using an Euclidean distance measure, and the corresponding projection $\boldsymbol{y}$ used as the transformed feature.

$$p^\star = \arg\min_p \left|\left|\boldsymbol{x} - \boldsymbol{\Psi}_p \boldsymbol{\Psi}_p^\dagger \boldsymbol{x}\right|\right|_2 \tag{9}$$

$$\boldsymbol{y} = \boldsymbol{\Psi}_{p^\star} \boldsymbol{\Psi}_{p^\star}^\dagger \boldsymbol{x} \tag{10}$$

The subspace $p^\star$ can be directly used as an estimator of the category of the signal $c^\star$. Alternatively, a simple *k-neaerst neighbour* classifier can be employed on the transformed features, and a class can be inferred as:

$$c^\star = \texttt{knn}(\boldsymbol{y}, \boldsymbol{Y}, \boldsymbol{c}) \tag{11}$$

where $\boldsymbol{Y}$ represents the matrix of training features after the transform stage. This latter approach is especially suitable when working with a large number of classes in a space of relatively small dimension (as in the numerical experiment presented in Section 5.1 where $P = 3$ and $N = 3$), as in this case multiple classes might be assigned to the same subspace.

5. NUMERICAL EXPERIMENTS

5.1. Classifying the FISHERIRIS dataset

To illustrate the S-IPR algorithm for feature transform, we apply it to the FISHERIRIS dataset [9]. This contains data for the classification of 150 iris specimens into the classes *setosa*, *versicolor* and *virginica*. The features corresponding to measurements on the sepal length, sepal width and petal length for the flowers are stored in the matrix $\boldsymbol{X} \in \mathbb{R}^{3\times 150}$. As a pre-processing step, we subtract the mean and normalise the standard deviation of the features to avoid large variations or offsets in any of the dimensions of the data.

The dictionary learning is run using the SMALLBox toolbox and the Incoherent dictionary learning add-on[2]. We learn an over-complete dictionary $\boldsymbol{\Phi}$ consisting of $K = 10$ atoms and a sparse approximation $\boldsymbol{A}$ that uses $S = 2$ active atoms for each signal. Incoherent sub-dictionaries are learned by the S-IPR method, setting the mutual coherence to the theoretical lower bound attainable by a $N \times K$ dictionary $\mu_{\min} = \sqrt{(K-N)/N(K-1)} \approx 0.5$ [10]. A maximum number of $I = 20$ iterations is set as the stopping criterion. These parameters are within the range used in previous studies of incoherent dictionary learning [8], and have not been optimised for this task.

The proposed method is used to learn $Q = 1$ dimensional sub-spaces, and is compared to PCA with a number of components $L = 2$ and to LDA. Figure 1 shows the results of the experiment. The original features show a clear cluster corresponding to the setosa class, and overlapping data for the other two classes. Applying PCA reduces the dimensionality of the 3-dimensional features to a rank 2 principal sub-space, but worsens the separation between classes. Both LDA and the proposed method are able to improve the separation of the three classes, but in remarkably different ways: while the former optimises the linear separability of different classes, features derived from S-IPR are projected onto low-dimensional discriminative subspaces.

To assess the classification performance of the various techniques, we created a random 5-fold partition of the training set used for cross-validation, and classified the transformed features using KNN with 5 neighbours. The misclassification rate MCR defined as the number of misclassified samples divided by the total number of samples is displayed for the various methods in the first row of Table 1. S-IPR and LDA are both able to attain a 7% misclassification error, improving on the 10% obtained on the original features. The worst result is achieved by PCA, whose dimensionality reduction has the effect of mixing together features belonging to different classes, and achieves an error of 44%.

	none	PCA	LDA	S-IPR
MCR - FISHERIRIS	0.10	0.44	0.07	0.07
MCR - USPS	0.343	0.433	0.359	0.315

Table 1: Misclassification error evaluated using different feature transform methods on the FISHERIRIS and USPS datasets (none indicates no feature transform).

5.2. Classification of the USPS digits dataset

In order to evaluate the proposed method on a more challenging dataset that contains signals of higher dimension, we run a similar experiment on the USPS digits dataset, which consists of a collection of 16×16 pixels images of hand-written digits [3]. We selected the digits 1, 3 and 8 resulting in a total of 1405 examples stored in the matrix $\boldsymbol{X} \in \mathbb{R}^{256\times 1405}$.

We investigated the effect of the parameters $K = \{256, 512, 1024\}$, $Q = \{2, 4, 8, 16, 32, 64, 128\}$ and $S = \{2, 25, 128, 256\}$ on the misclassification error, and compared the S-IPR to PCA and LDA. For PCA, the number of principal components L is automatically set by the algorithm to choose an approximation that retains 95% of the variance of the data. The second row of Table 1 shows the results obtained with the best choice of parameters $K^\star = 512$ (corresponding to a 2 times over-complete dictionary), $Q^\star = 64$ and $S^\star = 128$. In this case, S-IPR achieves a 31.5% misclassification error, while the other two techniques do not improve the K-NN classification on to the original features, which scores 34.3%.

Analysing the MCR as a function of the parameters K, Q and S, the most relevant trend observed empirically on this dataset is that a value $Q < 16$ steeply increases the misclassification error. This suggests that the dimensionality of the incoherent sub-spaces must be large enough to retain discriminative characteristics of the original high-dimensional data. The dependency of MCR on the other two parameters is less significant, suggesting that dictionary learning leads to discriminative subspaces regardless of the number of active atoms S or the number of atoms in the dictionary K.

6. CONCLUSION

We have presented the S-IPR algorithm for learning incoherent subspaces, and employed it as a feature transform method in the context of supervised classification. The encouraging experimental results obtained on the FISHERIRIS and the USPS datasets suggest that the proposed algorithm can overcome limitations of standard feature transforms methods, making it a viable tool for machine learning research and practice. A better theoretical understanding of the performance of S-IPR (especially in relation to other methods for

[2] http://code.soundsoftware.ac.uk/projects/smallbox.

[3] http://www.cs.nyu.edu/ roweis/data.html.

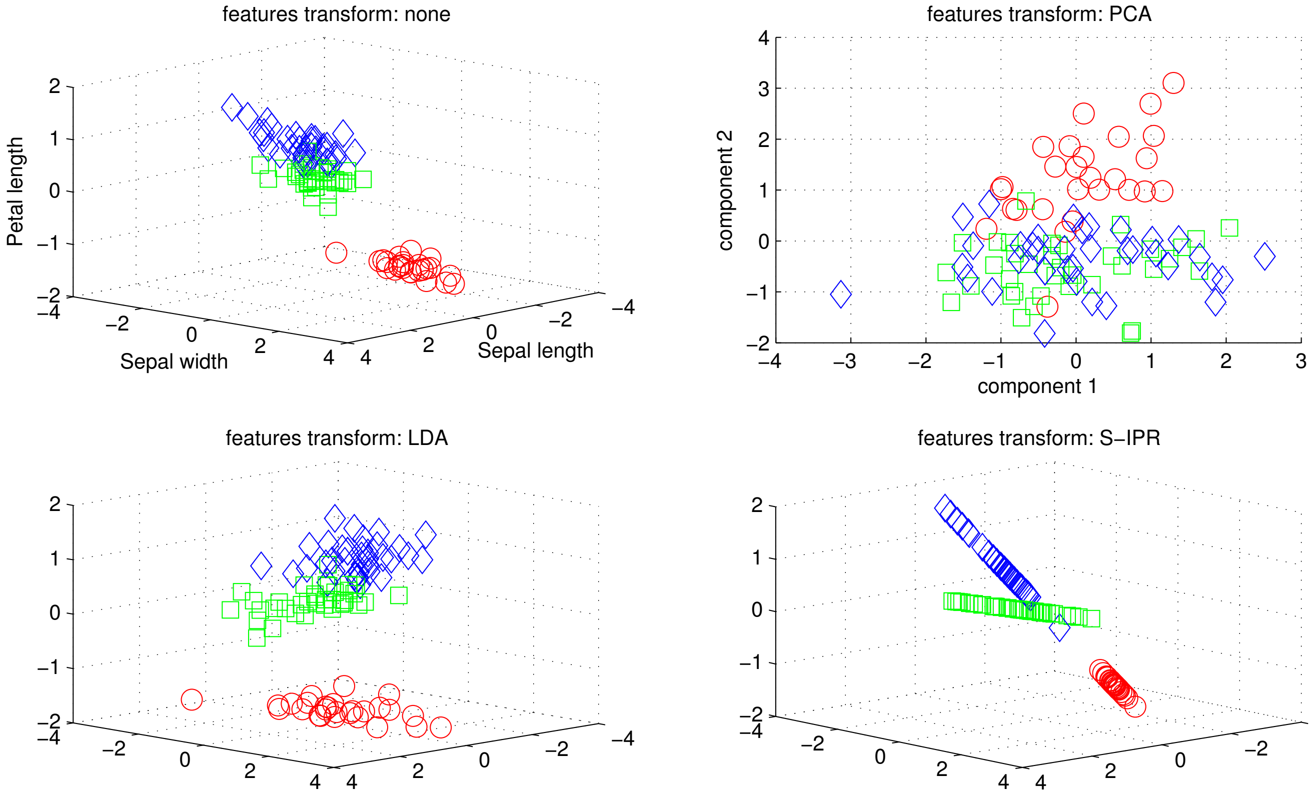

Fig. 1: Features transform for the fisheriris dataset. Red circular markers correspond to setosa, green square markers to versicolor and blue diamond markers to virginica.

subspace and manifold learning), and further validations of the technique on challenging classification problems constitute the most promising avenues for future research.

7. REFERENCES

[1] R. Duda and P. E. Hart, *Pattern classification and scene analysis*, Wiley and Sons, 1973.

[2] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science, Sixth Series*, vol. 2, pp. 559–572, 1901.

[3] E. Barshan, A. Ghodsi, Z. Azimifar, and M. Z. Jahromi, "Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds," *Pattern Recognition*, vol. 44, no. 7, pp. 1357–1371, 2011.

[4] L. Van Der Maaten, E. Postma, and J. Van Den Herik, "Dimensionality reduction: A comparative review," Tech. Rep., TiCC, Tilburg University, 2009.

[5] E. Elhamifar and R. Vidal, "Sparse subspace clustering: algorithm, theory, and applications," To appear in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013.

[6] M. Elad, *Sparse and redundant representations*, Springer, 2010.

[7] R. Rubinstein, A. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, Jun. 2010.

[8] D. Barchiesi and M. D. Plumbley, "Learning incoherent dictionaries for sparse approximation using iterative projections and rotations," *IEEE Trans. on Signal Processing on Signal Processing*, vol. 61, no. 8, pp. 2055–2065, Apr. 2013.

[9] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annuals of Eugenics*, vol. 7, pp. 179–188, 1936.

[10] T. Strohmer and R. W. Jr. Heath, "Grassmannian frames with applications to coding and communication," *Applied and Computational Harmonic Analysis*, vol. 14, no. 3, pp. 257–275, 2003.