PSSDL: Probabilistic Semi-Supervised Dictionary Learning

Behnam Babagholami-Mohamadabadi, Ali Zarghami, Mohammadreza Zolfaghari, and Mahdieh Soleymani Baghshah

Department of Computer Engineering, Sharif University of Technology, Tehran, Iran
{babagholami,Zarghami,mzolfaghari}@ce.sharif.edu, soleymani@sharif.edu

Abstract. While recent supervised dictionary learning methods have attained promising results on the classification tasks, their performance depends on the availability of the large labeled datasets. However, in many real world applications, accessing to sufficient labeled data may be expensive and/or time consuming, but its relatively easy to acquire a large amount of unlabeled data. In this paper, we propose a probabilistic framework for discriminative dictionary learning which uses both the labeled and unlabeled data. Experimental results demonstrate that the performance of the proposed method is significantly better than the state of the art dictionary based classification methods.

Keywords: Dictionary learning, MAP estimation, Gibbs Random Field, Local Linear Embedding, Local Fisher Discriminant Analysis

1 Introduction

In the recent decade, Sparse Representation (SR), and Dictionary Learning (DL) have gained much interest in the computer vision and pattern recognition areas [5]. This attention is due to the fact that many natural signals (like natural images) are sparse in their nature and can be approximated or even fully recovered by their sparse codes. A common SR formulation consists of a sparsity term and a reconstructive term as shown in the following expression

$$[\hat{A}, \hat{D}] = \underset{A,D}{\operatorname{argmin}} \sum_{i=1}^{N} \|x_i - D\alpha_i\|_2^2 + \gamma\|\alpha_i\|_1, \tag{1}$$

where x_i is the i-th input signal, D is the dictionary, $A = [\alpha_1, ..., \alpha_N]$ represents the sparse codes and γ is a regularization term. This problem is not fully convex, but by fixing A or D, and minimizing the other one, the problem can be treated as a convex problem. Methods such as K-SVD [1] can be used to find a proper dictionary and a sparse code simultaneously.
Recently, Supervised Dictionary Learning (SDL) methods [6],[23], [2], [3] have

used DL for classification tasks by adding discriminative terms to the objective function of Eq. 1. [6] added a Fisher Discriminant Analysis (FDA) term to its objective function to make the sparse codes more discriminative. The method proposed in [23] incorporated a logistic loss function into the problem definition and learned a classifier and a dictionary simultaneously. Zhang et al. [2] modified the original K-SVD method by using the classification error as a part of objective function, allowing it to apply as a sparse coding classifier. Wright et al. [3] used the training signals as dictionary atoms (basis). Using this dictionary, new signals can be represented as a sparse linear combination of the training signals. The discrimination will be performed based on the representation error caused by considering only coefficients corresponding to atoms related to each class and ignoring all other atoms.

Despite their merits, SDL methods have two main drawbacks. Firstly, the regularization parameters are usually set using cross-validation technique, which biases their cost functions toward data points that are poorly represented. Hence, they are easily affected by noisy, outlier, and mislabeled training data. Secondly, the performance of the SDL methods is highly dependent on the number of the training samples. Unfortunately, in many pattern classification problems, accessibility to a large set of the labeled data may not be possible due to the fact that labeling data is expensive and time consuming. On the other hand, unlabeled data points are easily available in abundance which have motivated machine learning researchers to develop semi-supervised learning methods which utilize a large amount of unlabeled data, along with the limited number of labeled data, to build better models for classification tasks.

One of the most well-known methods for semi-supervised classification (SSL) is Semi-Supervised Support Vector Machine (S3VM) [4], which regards the class label of unlabeled samples as extra unknowns and optimizes the classifier parameters and unknown labels simultaneously. Another popular algorithm for semi-supervised learning is Co-training [8], which assumes that features (data points) have multiple views. Based on this assumption, this algorithm utilizes the confident samples in one view to update the other view. However, in many applications such as image classification, each image has only one feature vector and hence it is difficult to apply Co-Training.

Recently, Shrivastava et al. [9] have proposed a semi-supervised dictionary learning (SSDL) algorithm for classification tasks. This algorithm uses an iterative process which goes as follows. In the first iteration, the dictionaries (one dictionary for each class) are learned using only the labeled data. Then, the class labels of the unlabeled data points are roughly inferred based on how well they are reconstructed by the dictionaries of the different classes. In the next iterations, the confident unlabeled data points are used to further refine the dictionaries.

In order to improve the discrimination power of the dictionaries, this method imposes some constraints on the DL task, in which the data samples belonging to some particular classes with high confidence, should be well represented by the corresponding dictionaries and poorly represented by other dictionaries. The Fisher Discriminant Analysis (FDA) is also used to enhance the discrimination

of the sparse codes for the labeled data.
Although the results of this method is better than the state of the art discriminative DL methods, it has several shortcomings. Firstly, due to the learning one dictionary for each class, it cannot scale to problems with large number of classes. Secondly, using FDA only for labeled data may result in overfitting due to the fact that the number of the labeled data is much smaller than that of the unlabeled data. Third, it does not consider the underlying geometrical structure of both the labeled and unlabeled data.
To overcome these shortcomings, in this paper, we propose a novel algorithm to learn discriminative dictionaries for semi-supervised classification tasks. More specifically, a single dictionary is learned jointly with a classifier in a MAP setting, by which sharing features among different classes is allowed and it leads to less computational cost and less risk of overfitting. We also introduce a new discriminative term in our probabilistic framework by combining the methods of Local Fisher Discriminant Analysis (LFDA) [7] and Locally linear Embedding (LLE) [11] which preserves the global structure of all samples in addition to enhancing the discrimination power of the dictionary. The contributions of this paper are summarized as follows:

- Our method combines the LFDA, and LLE algorithms to increase the discrimination power of the dictionary as well as preserving the geometrical structure of both labeled and unlabeled data points, by which overfitting to the labeled data is prevented.
- Our method furthermore integrates a multinomial Logistic regression classifier into the proposed probabilistic dictionary learning framework, which improves the discrimination in the sparse codes of signals, and the discrimination in the classifier construction.
- The free parameters are estimated using the MAP estimation technique which allows to avoid parameter tuning based on the cross-validation.
- The MAP parameters are efficiently estimated via the well-known feature-sign search algorithm [12].
- Our approach is validated on various well-known digit recognition, face recognition, and spoken letter classification benchmarks.

The remainder of this paper is organized as follows: The proposed probabilistic model (MAP setup) for dictionary learning is introduced in Section 2. The optimization procedure for estimating MAP parameters is discussed in Section 3. Experimental results are presented in Section 4. We conclude and discuss future work in Section 5.

2 Proposed Method

In this section, we present our probabilistic framework for dictionary learning which takes into account both the labeled and unlabeled data. Here, the intuition is to improve the discriminativeness in the dictionary and to prevent overfitting the (small-size) labeled data points by adding a classifier error term and a geometrical preserving term into the proposed MAP setting respectively.

2.1 Problem Formulation

Let $X_L = \{(x_i, y_i), i = 1, ..., N_l\}$ be the set of labeled data, and $X_U = \{x_j, j = N_l + 1, ..., N\}$ be the set of unlabeled data available for learning the dictionary, where N_l and N are the number of labeled and total samples, respectively. Here, $x_j \in R^M$ denotes the j-th sample, $y_i \in \{1, 2, ..., C\}$ is the corresponding class label of the i-th data point, C is the number of classes, and $N_u = N - N_l$ is the number of the unlabeled data points. Let $D = [d_1, ..., d_K] \in R^{M \times K}$ be the dictionary with K atoms and $A = [A_L, A_U]_{K \times N}$ be the matrix of the sparse codes, where $A_L = [\alpha_1, ..., \alpha_l]_{K \times N_l}$ and $A_U = [\alpha_{N_l+1}, ..., \alpha_N]_{K \times N_u}$ show the matrices of the sparse codes of the labeled and unlabeled data respectively.
Here, we assume that each data point $x_i (i = 1, ..., N)$ can be represented as a sparse linear combination of K dictionary atoms with additive zero-mean Gaussian noise ϵ_i ($\epsilon_i \sim \mathcal{N}(0, \sigma_i^2 I)$). Using this assumption, sparse codes can be considered as latent variables of the representation model. Consequently, the likelihood of observing a specific sample x, given the dictionary (D) and its sparse code (α) is modeled as a Gaussian:

$$P(x \mid D, \alpha, \sigma^2) \sim \mathcal{N}(D\alpha, \sigma^2 I). \tag{2}$$

To model the classification process, we use the multinomial logistic regression classifier which is defined as

$$P(y = i \mid \alpha, w_1, ..., w_C) = \frac{exp(w_i^T \alpha)}{\sum_{j=1}^{C} exp(w_j^t \alpha)}, \quad i = 1, ..., C, \tag{3}$$

where α and y are the sparse code of an ordinary sample x and its label respectively, and $W = [w_1, ..., w_C]$ shows the parameter of the classifier.
In order to further enhance the discriminative power of the dictionary, some of the previous DL methods [6], [9] have utilized the FDA algorithm, by which the trace of the within-class scatter matrix of A_L is minimized and the trace of the between-class scatter matrix of A_L is maximized.
However, in situations where the number of labeled data is small, the FDA may overfit the labeled samples. Moreover, in cases where a large set of unlabeled samples is available, FDA cannot make use of unlabeled data. Another drawback of the FDA algorithm is that its performance may be degraded if the samples in a class form several separate clusters [10]. To circumvent these shortcomings, we propose a new discrimination term based on a smooth combination of LFDA algorithm and LLE algorithm, by which the topological structure of all the data is preserved in addition to enhancing the discrimination power of the dictionary. Precisely speaking, using LFDA algorithm, within-class scatter can be computed locally, and so the within-class multimodality can be resolved. Using LLE, reliance on the global structure of all samples and information brought by labeled samples is controlled.
In LFDA method, the local between-class scatter matrix S^{lB} and the local

within-class scatter matrix S^{lW} are defined as [7]

$$S^{(LB)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{i,j}^{(lb)}(\alpha_i - \alpha_j)(\alpha_i - \alpha_j)^T, \tag{4}$$

$$S^{(LW)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{i,j}^{(lw)}(\alpha_i - \alpha_j)(\alpha_i - \alpha_j)^T, \tag{5}$$

where $W_{i,j}^{(lb)}$ and $W_{i,j}^{(lw)}$ are the $N \times N$ matrices which are defined as

$$W_{i,j}^{(lb)} = \begin{cases} P_{i,j}(1/N_l - 1/N_{ly_i}) & \text{if } y_i = y_j \\ 1/N_l & \text{if } y_i \neq y_j \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

$$W_{i,j}^{(lw)} = \begin{cases} P_{i,j}/N_{ly_i} & \text{if } y_i = y_j \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where N_{ly_i} denotes the number of the labeled samples in the class y_i. In the above equations, $P_{i,j}$ shows the affinity value between x_i and x_j which is defined as [7]

$$P_{i,j} = exp\big(-\frac{\|x_i - x_j\|^2}{\gamma_i\gamma_j}\big), \tag{8}$$

where the parameter γ_i represents the local scaling around x_i as

$$\gamma_i = \|x_i - x_i^k\|, \tag{9}$$

and x_i^k is the k-th nearest neighbor of x_i (a heuristic choice of k = 5 was shown to be useful through experiments).

Using LLE, we try to preserve the intrinsic topological structure of the data based on the notion of affinity preserving. In other words, by employing LLE, the geometric structure of the data is retained by maintaining locally linear relationships between sparse codes of close data points.

Given the set of the both labeled and unlabeled data points, LLE assumes that each data point in the original space can be recovered using a linearly weighted average of its neighbors. Based on this assumption, an optimal weight matrix $S^* = [s_{ij}^*]$ is reconstructed by solving the following problem:

$$S^* = \underset{S}{\operatorname{argmin}} \sum_{i=1}^{N} \|x_i - \sum_{x_j \in N_k(x_i)} s_{ij}x_j\|^2, \;\; s.t.\; \forall i, \sum_{x_j \in N_k(x_i)} s_{ij} = 1, \tag{10}$$

where $N_k(x_i)$ demonstrates the set of k nearest neighbor of x_i. The above optimization problem can be solved as a constrained least-squares problem [18].

In order to utilize the information of the unlabeled data points more efficiently, we consider certain assumptions about the general geometric properties of the data. More precisely, in many applications, high dimensional data points are

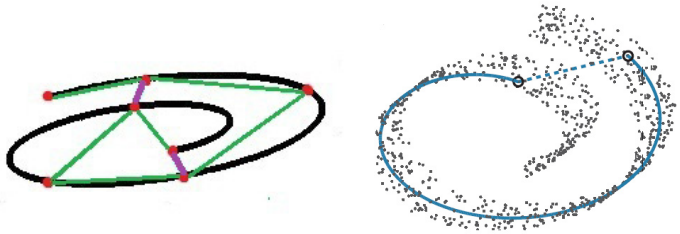

Fig. 1. Left: part of a one dimensional manifold, showing the deficiency of the Euclidian distance (purple edges are shortcut edges), right: Geodesic curve between two points on a manifold (solid line shows the geodesic curve and the dashed line shows the Euclidean curve).

actually samples from a low-dimensional subspace of the actual feature space. In these cases, we can make use of the Manifold assumption which is among the most practical assumptions in semi-supervised learning tasks [19].
In the original LLE and LFDA algorithms, Euclidean distance is considered as a measure of evaluating distance between data points. However, by considering the manifold assumption, Euclidean distance measure may be misleading, since two samples having a small Euclidean distance may be located far apart on the underlying manifold of the data points (Fig. 1).
To circumvent this problem, we make use of geodesic distance as a distance measure between data points which enables us to determine the neighborhood of a data point more precisely. The geodesic distance between two sample x_i and x_j is defined as the length of shortest curve between x_i and x_j lying on the manifold of the data points (Fig. 1).
Since the underlying manifold of the samples is unknown (we only have data which are finite samples of the manifold), we cannot find the exact geodesic distance between each two data points. Hence, in this paper, we use the idea of [20] to approximate the geodesic distance between both labeled and unlabeled data points. In [20], first, a k nearest neighborhood graph of all data is constructed based on the Euclidean distance. Then, an iterative process is done to remove the shortcut edges (shortcut edges connect those points of the graph which are close to each other according to the Euclidean distance, but have large geodesic distance on the manifold [20]). After refining the constructed graph based on the idea of [20], we use an efficient shortest path algorithm to find the k nearest neighbor of each data point. After computing the optimal weight matrix S^* based on the geodesic distance, we try to minimize the following objective function in order to preserve the global structure of data in the sparse representation space.

$$\sum_{i=1}^{N} \|\alpha_i - \sum_{x_j \in N_k^G(x_i)} s_{ij}^* \alpha_j\|^2 = tr\big(AEA^T\big), \tag{11}$$

where $N_k^G(x_i)$ demonstrates the set of k nearest neighbors of x_i based on the geodesic distance, and $E = (I - S^*)^T(I - S^*)$.
Now, we define S^{RLW} and S^{RLB} as the regularized local within-class scatter matrix and the regularized local between-class scatter matrix respectively:

$$S^{RLW} = (1 - \vartheta)AL^{LW}A^T + \vartheta AEA^T \tag{12}$$

$$S^{RLB} = (1 - \vartheta)AL^{LB}A^T + \vartheta I_{K\times K} \tag{13}$$

where $\vartheta \in [0, 1]$ is a trade-off parameter, and L^{LW} and L^{LB} are the graph Laplacian matrix of the local within-scatter (S^{LW}) and the local between-class scatter (S^{LB}) matrices which are defined as

$$L^{LW} = D^{LW} - W^{(lw)}, \quad L^{LB} = D^{LB} - W^{(lb)}, \tag{14}$$

where D^{LW} and D^{LB} are diagonal $N \times N$ matrices with

$$D_{i,i}^{LW} = \sum_{j=1}^{N} W_{i,j}^{(lw)}, \quad D_{i,i}^{LB} = \sum_{j=1}^{N} W_{i,j}^{(lb)}. \tag{15}$$

Another constraint that the sparse codes should satisfy is the sparsity constraint. In order to enforce sparsity on A, we put a well-known Laplacian prior distribution on each sparse code which is shown as

$$\alpha_i \sim Lap(\alpha_i \mid b) = \frac{1}{2b} exp\big(-\frac{\|\alpha_i\|_1}{b}\big), \tag{16}$$

where b is the scale parameter of the laplacian distribution. In order to encode the sparsity constraint, the discriminative constraint by LFDA, and the global constraint by LLE into our probabilistic model, we use Gibbs Random Field (GRF). A set of random variables $\{\alpha_i\}_{i=1}^N$ is said to be a GRF, if and only if their joint distribution follows a Gibbs distribution. Hence, the joint distribution must take the form

$$P(\alpha_1, ..., \alpha_N) = \frac{1}{Z} exp\big(-\frac{1}{T} U(\alpha_1, ..., \alpha_N)\big), \tag{17}$$

where Z is the normalizing constant called the Partition Function, T is a constant called the temperature (in this paper its assumed to be 1), and $U(\alpha_1, ..., \alpha_N)$ is the energy function which in this paper is defined as

$$U(\alpha_1, ..., \alpha_N) = N \log 2b + \frac{1}{b} \sum_{i=1}^{N} \|\alpha_i\|_1 + tr\big(S^{RLW}(A)\big) - tr\big(S^{RLB}(A)\big). \tag{18}$$

For simplicity, we also put a Gaussian prior distribution on the dictionary and the classifier parameters. Hence we have

$$P(D \mid \sigma_d^2) \propto \prod_{i=1}^{K} \mathcal{N}(d_i; 0, \sigma_d^2 I_M), \quad P(W \mid \sigma_w^2) \propto \prod_{j=1}^{C} \mathcal{N}(w_j; 0, \sigma_w^2 I_K). \tag{19}$$

To model the prior of the parameter $\Xi = \{\sigma_i (i = 1, ..., N), b, \sigma_d, \sigma_w\}$, we choose the objective non-parametric Jeffreys prior, which has been demonstrated to perform well for regression and classification tasks [21]. So, we have

$$P(\Xi) \propto \prod_{i=1}^{N} \frac{1}{\sigma_i^2} \times \frac{1}{b} \times \frac{1}{\sigma_d^2} \times \frac{1}{\sigma_w^2}. \tag{20}$$

The prior over σ_i encourages a low variance representation which means the training data should properly fit the proposed representation model. The prior over b encourages a sparser solution for the sparse codes which is the main aim of the sparse representation based methods, and the prior over σ_d and σ_w decreases the risk of overfitting the dictionary and the classifier respectively.
After defining the prior and the likelihood distributions, the posterior distribution of the latent variables (W, D, A, Ξ) given the observations $(X_L, X_U, \{y_i\}_{i=1}^{N})$ can be computed as

$$P(W, D, A, \Xi \mid X_L, X_U, \{y_i\}_{i=1}^{N_l}) \propto \prod_{i=1}^{N} \frac{1}{\sigma_i^2} \times \frac{1}{b} \times \frac{1}{\sigma_d^2} \times \frac{1}{\sigma_w^2} \times e^{\left(-U(\alpha_1,\ldots,\alpha_N)\right)} \times$$
$$\prod_{i=1}^{N} \mathcal{N}(x_i; D\alpha_i, \sigma_i^2) \prod_{j=1}^{N_l} \frac{exp(w_{y_j}^T \alpha_j)}{\sum_{c=1}^{C} exp(w_c^T \alpha_j)} \prod_{i=1}^{K} \mathcal{N}(d_i; 0, \sigma_d^2 I_M) \prod_{j=1}^{C} \mathcal{N}(w_j; 0, \sigma_w^2 I_K). \tag{21}$$

In order to determine the most probable point estimate for the latent variables, we compute the MAP estimation of the above posterior distribution which is easy to show that it is equal to the following minimization problem.

$$\begin{aligned}
[\hat{W}, \hat{D}, \hat{A}, \hat{\Xi}] = \underset{W,D,A,\Xi}{\operatorname{argmin}} & \sum_{i=1}^{N} \frac{\|x_i - D\alpha_i\|_2^2}{2\sigma_i^2} + \sum_{i=1}^{N} \log \sigma_i^{M+2} - \sum_{j=1}^{N_l} w_{y_j}^T \alpha_j \\
& + \sum_{j=1}^{N_l} \log \left(\sum_{c=1}^{C} exp(w_c^T \alpha_j) \right) + \frac{1}{2\sigma_w^2} \sum_{j=1}^{C} \|w_j\|_2^2 + C \log \sigma_w^{K+2} \\
& + \frac{1}{2\sigma_d^2} \sum_{j=1}^{K} \|d_j\|_2^2 + K \log \sigma_d^{M+2} + (N+1) \log b + \frac{1}{b} \sum_{i=1}^{N} \|\alpha_i\|_1 \\
& + tr\big(S^{RLW}(\alpha_1, ..., \alpha_N)\big) - tr\big(S^{RLB}(\alpha_1, ..., \alpha_N)\big).
\end{aligned} \tag{22}$$

3 Optimization Procedure

In this section, we describe the optimization procedure for the proposed objective function (Eq. 22). Solving (22) is a challenging task because of two reasons. Firstly, the objective function is not convex respect to W, D, A and Ξ simultaneously. Secondly, the log-sum-exp term $(\log(\sum_{c=1}^{C} exp(w_c^T \alpha_j)))$ in the objective function prevents us using efficient methods such as feature search sign algorithm

[12] to compute the sparse codes efficiently. To address the first problem, we can easily observe that the objective function is convex with respect to each of the parameters when the others are fixed. Hence, we resort to a coordinate descent method (alternating optimization), in which unknown parameters are updated through an iterative process which updates each parameter by fixing the other parameters in each step. To circumvent the second problem, we utilize a suitable upper bound to the log-sum-exp function proposed by [22] which states that for any $\beta \in \mathbb{R}$ and $\xi_k \in [0, \infty)$, $k = 1, ..., K$

$$\log(\sum_{k=1}^{K} e^{g_k}) \leq \beta + \sum_{k=1}^{K} \Big(\frac{g_k - \beta - \xi_k}{2} + \lambda(\xi_k)\big((g_k - \beta)^2 - \xi_k^2\big) + \log(1 + e^{\xi_k}) \Big), \quad (23)$$

where

$$\lambda(\xi) = \frac{1}{2\xi}\big(\frac{1}{1 + e^{-\xi}} - \frac{1}{2}\big), \quad (24)$$

where β and $\{\xi_k\}_{k=1}^{K}$ are the variational parameters which can be optimized to get the tightest possible bound. So, by replacing the log-sum-exp term of the objective function with the upper bound of Eq. 23, we have

$$\begin{aligned}
[\hat{W}, \hat{D}, \hat{A}, \hat{\Xi}, \hat{\Theta}] = \underset{W,D,A,\Xi,\Theta}{\operatorname{argmin}} & \sum_{i=1}^{N} \frac{\|x_i - D\alpha_i\|_2^2}{2\sigma_i^2} + \sum_{i=1}^{N} \log \sigma_i^{M+2} - \sum_{j=1}^{N_l} w_{y_j}^T \alpha_j \\
& + \sum_{j=1}^{N_l} \beta_j + \frac{1}{2} \sum_{j=1}^{N_l} \sum_{c=1}^{C} (w_c^T \alpha_j - \beta_j - \xi_{jc}) + \sum_{j=1}^{N_l} \sum_{c=1}^{C} \lambda(\xi_{jc})\big((w_c^T \alpha_j - \beta_j)^2 - \xi_{jc}^2\big) \\
& + \sum_{j=1}^{N_l} \sum_{c=1}^{C} (\log(1 + e^{\xi_{jc}})) + \frac{1}{2\sigma_w^2} \sum_{j=1}^{C} \|w_j\|_2^2 + C \log \sigma_w^{K+2} + \frac{1}{2\sigma_d^2} \sum_{j=1}^{K} \|d_j\|_2^2 \\
& + K \log \sigma_d^{M+2} + (N+1) \log b + \frac{1}{b} \sum_{i=1}^{N} \|\alpha_i\|_1 + tr\big(A^T A \Gamma\big), \quad (25)
\end{aligned}$$

where $\Theta = \{\beta_j, \xi_{jc}\}_{j=1,\, c=1}^{j=N_l, c=C}$ is the set of variational parameters, and Γ is a $N \times N$ matrix which is defined as

$$\Gamma = (1 - \vartheta) L^{LW} + \vartheta E - (1 - \vartheta) L^{LB}. \quad (26)$$

Its obvious from Eq. 25 that it is convex in one parameter when the other parameters are fixed. Using the upper bound of Eq. 23, we are able to solve the above optimization problem by an efficient feature-sign search algorithm [12] which goes as follows.

Computing Sparse Codes A with Fixed W, D, Ξ and Θ: We optimize each sparse code $\alpha_i (i = 1, ..., N)$ by fixing sparse codes $\alpha_j (j \neq i)$ of other signals. Hence, for each sparse code α_i, if $x_i \in X_L$, we must solve

$$\hat{\alpha_i} = \underset{\alpha_i}{\operatorname{argmin}}\, F_L(\alpha_i), \quad (27)$$

and if $x_i \in X_U$, we must solve

$$\hat{\alpha_i} = \underset{\alpha_i}{\operatorname{argmin}} F_U(\alpha_i), \tag{28}$$

where

$$\begin{aligned} F_L(\alpha_i) =& \frac{1}{2\sigma_i^2}\|x_i - D\alpha_i\|_2^2 - w_{y_i}^T\alpha_i + \frac{1}{2}\sum_{c=1}^{C} w_c^T\alpha_i + \sum_{c=1}^{C}\lambda(\xi_{ic})(w_c^T\alpha_i - \beta_i)^2 \\ &+ 2\alpha_i^T(A\Gamma_i) - \alpha_i^T\alpha_i\Gamma_{i,i} + \frac{1}{b}\|\alpha_i\|_1, \end{aligned} \tag{29}$$

$$F_U(\alpha_i) = \frac{1}{2\sigma_i^2}\|x_i - D\alpha_i\|_2^2 + 2\alpha_i^T(A\Gamma_i) - \alpha_i^T\alpha_i\Gamma_{i,i} + \frac{1}{b}\|\alpha_i\|_1, \tag{30}$$

where Γ_i is the i-th column of Γ and $\Gamma_{i,i}$ is the (i,i) element of Γ.
The functions in Eqs. 29 and 30 are exactly the objective functions that the feature-sign search algorithm can minimize. This algorithm iteratively searches for the coefficient sign vector θ_i of x_i, hence (27) and (28) reduce to a standard, unconstrained quadratic optimization problem (QP). Precisely speaking, after finding the optimal coefficient sign of the sparse code α_i, $\|\alpha_i\|_1$ can be replaced by $\theta_i\alpha_i$, by which α_i can be computed analytically by setting the derivative of $F_L(\alpha_i)$ and $F_U(\alpha_i)$ respect to α_i equal to zero. The gradient of $F_L(\alpha_i)$ and $F_U(\alpha_i)$ can be calculated as

$$\begin{aligned} \frac{\partial F_L(\alpha_i)}{\partial\alpha_i} =& \frac{1}{\sigma_i^2}D^T(D\alpha_i - x_i) - w_{y_i} + \frac{1}{2}\sum_{c=1}^{C} w_c - 2\beta_i\sum_{c=1}^{C}\lambda(\xi_{ic})w_c \\ &+ 2\Big(\sum_{c=1}^{C}\lambda(\xi_{ic})w_c w_c^T\Big)\alpha_i + 2A\Gamma_i + \frac{\theta_i}{b}, \end{aligned} \tag{31}$$

$$\frac{\partial F_U(\alpha_i)}{\partial\alpha_i} = \frac{1}{\sigma_i^2}D^T(D\alpha_i - x_i) + 2A\Gamma_i + \frac{\theta_i}{b}. \tag{32}$$

Finally, the analytic solution of α_i can be obtained when we have $\frac{\partial F_L(\alpha_i)}{\partial\alpha_i} = 0$, if $x_i \in X_L$ and $\frac{\partial F_u(\alpha_i)}{\partial\alpha_i} = 0$, if $x_i \in X_U$:

$$\begin{aligned} \hat{\alpha_i} =& \Big(\frac{1}{\sigma_i^2}D^TD + 2\Big(\sum_{c=1}^{C}\lambda(\xi_{ic})w_c w_c^T\Big) + 2\Gamma_{i,i}I\Big)^{-1}\Big(\frac{1}{\sigma_i^2}D^Tx_i + w_{y_i} - \frac{1}{2}\sum_{c=1}^{C} w_c \\ &+ 2\beta_i\sum_{c=1}^{C}\lambda(\xi_{ic})w_c - 2\sum_{j\neq i}\Gamma_{j,i}\alpha_j - \frac{\theta_i}{b}\Big), \qquad if\ \ x_i \in X_L, \end{aligned} \tag{33}$$

$$\hat{\alpha_i} = \Big(\frac{1}{\sigma_i^2}D^TD + 2\Gamma_{i,i}I\Big)^{-1}\Big(\frac{1}{\sigma_i^2}D^Tx_i - 2\sum_{j\neq i}\Gamma_{j,i}\alpha_j - \frac{\theta_i}{b}\Big), \qquad if\ \ x_i \in X_U. \tag{34}$$

In practice, the Hessian matrices of F_L (Eq. 35), and F_U (Eq. 36) may not be positive semidifinite. So, a very small η (ηI) is added to the Hessian matrices to make them positive semidifinite, hence F_L and F_U are convex.

$$H_{F_L} = \frac{1}{\sigma_i^2} D^T D + 2\big(\sum_{c=1}^{C} \lambda(\xi_{ic}) w_c w_c^T\big) + 2\Gamma_{i,i} I, \tag{35}$$

$$H_{F_U} = \frac{1}{\sigma_i^2} D^T D + 2\Gamma_{i,i} I. \tag{36}$$

Updating Dictionary D with Fixed A, W, Ξ and Θ: Given A, W, Ξ and Θ, the optimization problem for D can be formulated as

$$\hat{D} = \underset{D}{\operatorname{argmin}} \sum_{i=1}^{N} \frac{\|x_i - D\alpha_i\|_2^2}{2\sigma_i^2} + \frac{1}{2\sigma_d^2}\|D\|_F^2. \tag{37}$$

The above problem is an unconstrained quadratic programming, for which D can be computed analytically as

$$\hat{D} = X\Sigma A^T (A\Sigma A^T + \frac{1}{\sigma_d^2} I)^{-1}, \tag{38}$$

where Σ is a diagonal $N \times N$ matrix with

$$\Sigma_{i,i} = \frac{1}{\sigma_i^2}, \;\; i = 1, 2, ..., N. \tag{39}$$

Updating the Classifier parameter W with Fixed A, D, Ξ and Θ: Without loss of generality, we assume that the first N_L^c labeled samples belong to the c-th class. So, given A, Ξ, D and Θ, the optimization problem for w_c can be formulated as

$$\hat{w}_c = \underset{w_c}{\operatorname{argmin}} \frac{1}{2}\big(\sum_{j=1}^{N_L} \alpha_j^T\big) w_c - \big(\sum_{j=1}^{N_L^c} \alpha_j^T\big) w_c + \sum_{j=1}^{N_L} \lambda(\xi_{jc})(\alpha_j^T w_c - \beta_j)^2 + \frac{1}{2\sigma_w^2} w_c^T w_c. \tag{40}$$

By setting the derivative of the objective function of the above equation respect to w_c equal to zero, we can compute w_c analytically as

$$\hat{w}_c = \Big(2\sum_{j=1}^{N_L} \lambda(\xi_{jc})\alpha_j \alpha_j^T + \frac{1}{\sigma_w^2} I\Big)^{-1} \Big(\sum_{j=1}^{N_L^c} \alpha_j + \sum_{j=1}^{N_L} \big(2\lambda(\xi_{jc})\beta_j - \frac{1}{2}\big)\alpha_j\Big). \tag{41}$$

Updating the free Parameter Ξ with Fixed A, W, D and Θ: Given A, W, D and Θ, each parameter can be computed analytically as:

$$\hat{\sigma_i}^2 = (\frac{1}{M+2})\|x_i - D\alpha_i\|_2^2, \quad i = 1, 2, ..., N, \tag{42}$$

$$\hat{\sigma_d}^2 = (\frac{1}{K(M+2)})\|D\|_F^2, \tag{43}$$

$$\hat{\sigma_w}^2 = (\frac{1}{C(K+2)})\|W\|_F^2, \tag{44}$$

$$\hat{b} = (\frac{1}{N+1})\sum_{i=1}^{N}\|\alpha_i\|_1. \tag{45}$$

Updating the Variational parameter $\boldsymbol{\Theta}$ with Fixed $\boldsymbol{A, W, D}$ and $\boldsymbol{\Xi}$: Given A, W, D and Ξ, we first compute the updates for $\{\beta_j\}_{j=1}^{N_L}$ by fixing other variational parameters $(\{\xi_{jc}\}_{j=1,c=1}^{N_L,C})$ which leads to the following solution.

$$\hat{\beta}_j = \Big(2\big(\sum_{c=1}^{C}\lambda(\xi_{jc})w_c^T\big)\alpha_j + \frac{C}{2} - 1\Big)\Big/2\big(\sum_{c=1}^{C}\lambda(\xi_{jc})\big), \quad j = 1, ..., N_L. \tag{46}$$

Secondly, we fix $\{\beta_j\}_{j=1}^{N_L}$, and update $\{\xi_{jc}\}_{j=1,c=1}^{N_L,C}$ by solving the following problem.

$$\hat{\xi}_{jc} = \underset{\xi_{jc}}{\operatorname{argmin}}\ \lambda(\xi_{jc})\big((w_c^T\alpha_j - \beta_j)^2 - \xi_{jc}^2\big) - \frac{1}{2}\xi_{jc} + \log(1 + e^{\xi_{jc}}). \tag{47}$$

By setting the derivative of the objective function of the above equation respect to ξ_{jc} equal to zero, we have

$$\lambda'(\xi_{jc})\big((w_c^T\alpha_j - \beta_j)^2 - \xi_{jc}^2\big) - 2\lambda(\xi_{jc})\xi_{jc} - \frac{1}{2} + \frac{1}{1 + e^{-\xi_{jc}}} = 0. \tag{48}$$

Using the definition of $\lambda(\xi_{jc})$, the above equation can be simplified as

$$\lambda'(\xi_{jc})\big((w_c^T\alpha_j - \beta_j)^2 - \xi_{jc}^2\big) = 0. \tag{49}$$

Since $\xi_{jc} \in [0, \infty]$, $\lambda'(\xi_{jc}) \neq 0$, hence we can compute ξ_{jc} analytically as

$$\hat{\xi}_{jc} = \mid w_c^T\alpha_j - \beta_j \mid, \quad j = 1, ..., N_L, \ \ c = 1, ..., C, \tag{50}$$

3.1 Class Label Prediction

After learning A, D, W and Ξ, classifying a new signal x with an unknown label y is performed by solving the following optimization problem.

$$\hat{y} = \underset{y\in\{1,...,C\}}{\operatorname{argmax}}\ P(y \mid x, D, W, \Xi). \tag{51}$$

Using the Bayes' rule formula, the above problem can be expressed as

$$\hat{y} = \underset{y\in\{1,...,C\}}{\operatorname{argmax}} \iint P(y \mid \alpha, W)P(x \mid D, \alpha, \sigma^2)P(\alpha \mid b)P(\sigma)\, d\alpha\, d\sigma. \tag{52}$$

Table 1. Classification accuracy of different methods.

dataset	SVM	S3VM	FDDL	SDL-G	SDL-D	S2D2	PM
MNIST	79.3 ± 1.9	83.3 ± 1.1	81.8 ± 1.8	82.1 ± 1.4	79.9 ± 2.1	86.1 ± 1.0	**87.4 ± 1.2**
USPS	80.7 ± 1.6	82.5 ± 0.9	81.1 ± 1.7	81.9 ± 1.3	80.1 ± 1.9	85.6 ± 0.9	**86.9 ± 1.0**
AR	70.4 ± 2.1	77.1 ± 1.7	74.2 ± 2.1	75.3 ± 1.5	74.1 ± 2.3	85.9 ± 1.4	**86.7 ± 1.5**
E-Yale B	72.1 ± 1.9	75.1 ± 1.7	65.9 ± 2.3	69.4 ± 1.8	67.9 ± 2.1	79.3 ± 1.8	**80.8 ± 1.8**
ISOLET	85.8 ± 1.7	87.3 ± 1.6	82.6 ± 1.9	83.4 ± 1.7	82.5 ± 1.9	89.9 ± 1.8	**91.4 ± 1.1**

where α is the sparse representation of x, and σ^2 is the representation noise variance of x (Eq. 2). By assuming that the posterior distribution over α and σ ($P(\alpha, \sigma \mid x, D, \Xi)$) can be approximated as a unit point measure at the MAP value (α_t, σ_t) of that distribution, the above problem can be replaced with the following minimization problem.

$$\hat{y} = \underset{y\in\{1,2,\ldots,C\}}{\operatorname{argmin}} \Bigg[\underset{\alpha_t,\sigma_t}{min} \frac{\|x - D\alpha_t\|_2^2}{2\sigma_t^2} + (M+2)\log\sigma_t + \frac{1}{b}\|\alpha_t\|_1 - w_y^T\alpha_t + \log\left(\sum_{c=1}^{C} exp(w_c^T\alpha_t)\right)\Bigg]. \tag{53}$$

Again, using the upper bound of Eq. 23, we can efficiently solve the above problem (details omitted due to space limitations).

4 Experimental Results

To illustrate the efficacy of our method, we present experimental results on applications such as Face Recognition (FR), Handwritten Digit Recognition (HDR), and Letter Recognition (LR). For comparison purposes, we compare our method with some state of the art SDL methods such as FDDL [6], SDL-G [23], SDL-D [23], and two well-known classification methods SVM and S3VM [4]. We also compare our method with S2D2 [9] which is a recently introduced semi-supervised DL algorithm. In all of our experiments, the parameter ϑ is set equal to 0.5 (the results of all experiments are almost unchanged for $0.1 \leq \vartheta \leq 0.9$). In order to determine an appropriate number of dictionary atoms (K), and nearest neighbors of data samples (k) for computing the LLE matrix (E), Five-fold cross validation is performed to find the best pair (K, k). The tested values for K are $\{64, 128, 256, 512\}$ and for k, $\{3, 5, 7, 9, 11\}$.

Digit Recognition: We apply the proposed method on two HDR datasets MNIST [24], and USPS [25]. The MNIST dataset consists of 70,000 28×28 images, 60,000 for training, 10,000 for testing, each of them containing one handwritten digit. USPS is composed of 7291 training images and 2007 test images

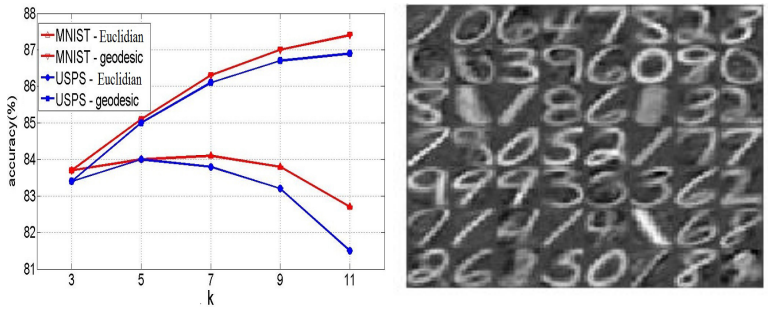

Fig. 2. Left: accuracy of the proposed method using the geodesic and the Euclidian distance for MNIST and USPS datasets, right: the learned D (K=64) for USPS dataset.

of size 16×16. For these datasets, 25 samples per class are randomly chosen from the training data as the labeled samples and the rest of the training data is used as the unlabeled data (we use the whole image as the feature vector in the digit datasets). The average recognition accuracies over 10 runs together with the standard deviation is shown in Table 1, from which we can see that the proposed method outperforms significantly the SDL methods. The improvement in performance compared to SDL methods is because of two reasons. Firstly, the number of labeled data is small, hence the SDL methods may overfit to the labeled data. Secondly, these methods cannot utilize unlabeled data for learning dictionary. Moreover, S3VM and S2D2 does not consider the topological structure of all data, hence both of them are less accurate than our method. We also provide a visualization of the learned D for USPS dataset for $K = 64$ (Fig. 2). In order to demonstrate the superiority of the geodesic distance over the Euclidian distance, we compute the recognition performance of the proposed method on MNIST, and USPS dataset, using both the geodesic and the Euclidian distances to find the k nearest neighbors of data points. The results are presented in Fig. 2, for various number of k. The figure shows two major points. Firstly, it is obvious that using the geodesic distance leads to better performance than the Euclidian distance, because the Euclidian distance ignores the fact that the data points lie on a low dimensional manifold. Secondly, when the number of nearest neighbors grows, the recognition accuracy decreases for the Euclidian distance and increases for the geodesic distance. This is due to the fact that using Euclidian distance, by increasing k, samples from different classes are more likely to be selected as the neighbors of data points. Hence, the matrix E which captures the locality information of data points may be misleading. On the other hand, the geodesic distance considers the underlying manifold of samples, by which the neighbors of data points can be found more accurately, and hence the matrix E encodes the locality of data points more precisely.

Face Recognition: We then perform the face recognition task on the widely used E-Yale B [26], and AR [27], face databases. The E-Yale B database con-

sists of 2,414 frontal-face images from 38 individuals (about 64 images per individual), and the AR database consists of over 4,000 frontal images from 126 individuals generated in two sessions, each of them consists of 14 images per individual (seven image for training, and seven image for testing). The E-Yale B and AR images are normalized to 54×48 and 60×40 respectively. We then perform a Principal Component Analysis (PCA) on the images to obtain 300 dimensional feature vectors. For AR dataset, we randomly choose two samples of the training session to form the labeled data and use the remaining five of that session as the unlabeled data. For E-Yale B dataset, for each class, we randomly select ten images as labeled data, twenty images as unlabeled data, and the remaining ones for testing. The average recognition accuracies over 5 runs together with the standard deviation are presented in the forth, and fifth rows of Table 1. Again, due to the small number of the labeled data, SDL methods have lower accuracy than SSDL methods. Moreover, because of considering the geometrical structure of data, our method has better performance than S3VM and S2D2 methods.

Letter Recognition: Finally, we apply our method on the ISOLET database [28], from UCI Machine Learning Repository which consists of 6238 examples and 26 classes corresponding to letters of the alphabet. We reduced the input dimensionality (originally at 617) by projecting the data onto its leading 100 principal components. For each class, We randomly select 10 samples as labeled data, 100 samples as unlabeled data, and the remaining ones for testing. The average recognition accuracies over 5 runs together with the standard deviation are presented in the last row of Table 1, from which we can see that the proposed method performs significantly better than the other algorithms.

5 Conclusion

In this paper, we proposed a probabilistic method which uses the information of unlabeled data as well as labeled data for learning discriminative dictionaries. The proposed method improves the discrimination of the dictionary and the sparse codes by incorporating a classifier error term and a discrimination term based on LFDA into the model. The topological structure of all data is also preserved based on LLE method which prevents overfitting the small labeled data. Moreover, instead of Euclidian distance, we utilized the geodesic distance which allows us to find the neighbors of data points more accurately. Experiments using various benchmark datasets demonstrate the superiority of the proposed method over the state-of-the-art SDL and SSDL methods.

References

1. Aharon, M., Elad, M., Bruckstein, A.:k-svd: An algorithm for designing dictionaries for sparse representation. IEEE Transactions on Signal Processing. (2006)

2. Zhang, Q., Li, X.: Discriminative k-svd for dictionary learning in face recognition. In: CVPR (2010)
3. Wright, J., Yang, A., Ganesh, A., Sastry, S., Ma, Y.: Robust Face Recognition via Sparse Representation. IEEE PAMI (2009)
4. Sindhwani, V., Keerthi, S.S.:Large scale semi-supervised linear svms. In: ACM SIGIR (2006)
5. Wright, J., Ma, Y., Mairal, J., Sapiro, G., Yan, S.: Sparse representation for computer vision and pattern recognition. In: Proceedings of the IEEE (2010)
6. Yang, M., Zhang, L., Feng, X., Zhang,D.: Fisher discrimination dictionary learning for sparse representation. In: IEEE ICCV (2011)
7. Sugiyama, M.: Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. Journal of Machine Learning Research (2007)
8. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: ACM COLT (1998)
9. Shrivastava, A., Jaishanker, K.P., Patel, V.M, Chellappa, R.: Learning discriminative dictionaries with partially labeled data. In: IEEE ICIP (2012)
10. Sugiyama, M., Ide, T., Nakajima, S., Sese, J.: Semi-Supervised Local Fisher Discriminant Analysis for Dimensionality Reduction. J. machine Learning. (2010)
11. Roweis, S.T, Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. J. Science. (2000)
12. Lee, H., Battle, A., Ng, A.Y.: Efficient sparse coding algorithms. In: NIPS (2007)
13. Huang, K., Aviyente, S.: Sparse representation for classification. In: NIPS (2007)
14. Boureau, Y., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: CVPR (2010)
15. Grosse, R., Raina, R., Kwong, H., Ng, A.Y.: Shift-invariant sparse coding for audio classification. In: Conf. on Uncertainty in AI (2007)
16. Mairal, J., Leordeanu, M., Bach, F., Hebert, M., Ponce, J.: Discriminative sparse image models for edage detection and image interpretation. In: ECCV (2008)
17. Zhang, W., Surve, A., Fern, X., Dietterich, T.: Learning non-redundant codebooks for classifying complex objects. In ICML (2009)
18. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science, 290(5500):2323-2326, (2000).
19. Chapelle, O., Scholkopf, B., Zien, A.: Semi-supervised learning, vol. 2. MIT press Cambridge (2006)
20. Ghazvininejad, M., Mahdieh, M., Rabiee, H.R., Khanipour, P., Rohban, M.H.: Isograph: Neighbourhood Graph Construction Based on Geodesic Distance for Semi-Supervised Learning. In: ICDM (2010)
21. Figueiredo, M.: Adaptive Sparseness using Jeffreys Prior. In: NIPS (2002).
22. Bouchard, G.: Efficient bounds for the softmax function. In: NIPS (2007).
23. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Supervised dictionary learning. In: NIPS (2009)
24. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Proc. of the IEEE (1998)
25. http://www-i6.informatik.rwth-aachen.de/ keysers/usps.html.
26. Lee, K., Ho, J., Kriegman, D.: Acquiring linear subspaces for face recognition under variable lighting. IEEE TPAMI. (2005)
27. Martinez, A., benavente, R.: The AR face database. CVC Tech. Report (1998)
28. Frank, A., Asuncion, A.: UCI Machine Learning Repository (2010)