Hierarchical Sparse Dictionary Learning

Xiao Bian[1], Xia Ning[2](✉), and Geoff Jiang[3]

[1] Electrical and Computer Engineering Department,
North Carolina State University, Raleigh, NC 27695, USA
xbian@ncsu.edu

[2] Department of Computer and Information Science,
IUPUI, Indianapolis, IN 46202, USA
xning@cs.iupui.edu

[3] Autonomic Management Department, NEC Labs America,
Princeton, NJ 45237, USA
gfj@neclabs.com

Abstract. Sparse coding plays a key role in high dimensional data analysis. One critical challenge of sparse coding is to design a dictionary that is both adaptive to the training data and generalizable to unseen data of same type. In this paper, we propose a novel dictionary learning method to build an adaptive dictionary regularized by an a-priori over-completed dictionary. This leads to a sparse structure of the learned dictionary over the a-priori dictionary, and a sparse structure of the data over the learned dictionary. We apply the hierarchical sparse dictionary learning approach on both synthetic data and real-world high-dimensional time series data. The experimental results demonstrate that the hierarchical sparse dictionary learning approach reduces overfitting and enhances the generalizability of the learned dictionary. Moreover, the learned dictionary is optimized to adapt to the given data and result in a more compact dictionary and a more robust sparse representation. The experimental results on real datasets demonstrate that the proposed approach can successfully characterize the heterogeneity of the given data, and leads to a better and more robust dictionary.

1 Introduction

Sparse representation has been demonstrated as very powerful in analyzing high dimensional data [1–3], where each data point can be typically represented as a linear combination of a few atoms in an over-complete dictionary. Assume $\mathbf{x} \in R^d$ is a data vector and $\mathbf{D}$ is the dictionary, and then the sparse representation of $\mathbf{x}$ can be formulated as to find the sparse code $\mathbf{w}$ over $\mathbf{D}$ by solving the following optimization problem,

$$\begin{aligned} \min_{\mathbf{w}} \quad & \|\mathbf{w}\|_0 \\ \text{s.t.} \quad & \|\mathbf{D}\mathbf{w} - \mathbf{x}\| \leq \sigma, \end{aligned}$$

where σ is a pre-defined threshold. The pursued sparse code $\mathbf{w}$ can been considered as a robust representation of $\mathbf{x}$, and can be used for clustering [4,5], classification [6] and denoising [2,7].

A. Appice et al. (Eds.): ECML PKDD 2015, Part II, LNAI 9285, pp. 687–700, 2015.
DOI: 10.1007/978-3-319-23525-7_42

One key question is how to construct such an over-complete dictionary that is suitable for sparse representation. There are two major approaches for constructing such dictionaries: analytic approaches and learning-based approaches [8]. In an analytic approach, the dictionary is carefully designed a priori, e.g. with atoms such as wavelets [9], curvelets [10] and shearlets [11,12]. One advantage of the analytic approaches is that the dictionary can be designed as well-conditioned for stable representation, for example, to have a better incoherence condition or restricted isometry property [13,14].

In learning-based approaches, the dictionaries are learned from the given data [2,15,16]. Compared to the designed dictionaries in analytic approaches, the learned dictionaries are usually more adaptive to the given data, and therefore lead to more robust representations. Therefore, the learning-based approaches outperform analytic approaches in many tasks such as denoising and classification, etc [1,17]. The dictionary learning problem in the learning-based approaches is typically formulated as the following optimization problem,

$$\begin{aligned} \min_{\mathbf{D}\in\mathcal{C},\mathbf{W}} \quad & \|\mathbf{X}-\mathbf{DW}\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{W}\|_0 \le k, \end{aligned} \tag{1}$$

where $\mathbf{X}$, $\mathbf{W}$ and $\mathbf{D}$ represent the data, their sparse codes and the dictionary, respectively, and $\mathcal{C}$ is a pre-specified feasible region for $\mathbf{D}$. However, (1) is non-convex and thus it is very difficult to find the global optimal solution or even a good local optimum.

In this paper, we propose to integrate both analytic approaches and learning-based approaches and learn from data a dictionary that is also built upon and regularized by an a-priori dictionary. The learned dictionary will be adaptive to the training data and its size will be determined by the intrinsic complexity of the training data. Meanwhile, due to the regularization from the a-priori dictionary, the non-convex optimization problem will have a more stable and better local minimum solution, and requires fewer training data. We compare the new method with the state-of-the-art methods on various aspects and our experimental results demonstrate superior performance of the new method.

2 Hierarchical Sparse Structures on Dictionaries

In the dictionary learning problem in (1), the constraint $\mathbf{D} \in \mathcal{C}$ is critical to regularize $\mathbf{D}$. In the state of the art, $\mathcal{C}$ is typically specified as $\mathcal{C} = \{\mathbf{D} : \forall \mathbf{d_i} \in \mathbf{D}, \|\mathbf{d_i}\|_2 \le c\}$ [2,15] or $\mathcal{C} = \{\mathbf{D} : \|\mathbf{D}\|_F \le c\}$ [16]. Intuitively, in both cases, $\mathcal{C}$ tames the amplitude of $\mathbf{D}$. However, these constraints do not consider any prior knowledge on $\mathbf{D}$, if available. Prior knowledge is valuable to learn a dictionary that is more powerful to characterize the data. For example, a dictionary for image patches is expected to have finer structures that might be further represented using DCT or wavelets. Incorporating such knowledge into dictionary learning can result in superior results [8,18].

Given an a-priori over-complete dictionary $\mathbf{\Phi}$ for data $\mathbf{X}$ based on some prior knowledge about $\mathbf{X}$, we aim to learn a dictionary $\mathbf{D}$ based on $\mathbf{\Phi}$ so that $\mathbf{D}$ is more adaptive to $\mathbf{X}$. In specific, we propose hierarchical sparse structures among $\mathbf{\Phi}$, $\mathbf{D}$ and $\mathbf{X}$, that is, $\mathbf{D}$ is constructed from $\mathbf{\Phi}$ via sparse combination of $\mathbf{\Phi}$'s atoms, and $\mathbf{X}$ is constructed from $\mathbf{D}$ via sparse combination of $\mathbf{D}$'s atoms. Mathematically, the hierarchical sparse structure of $\mathbf{D}$ over $\mathbf{\Phi}$ can be specified using the feasible region $\mathcal{C}$ as follows,

$$\mathcal{C} = \{\mathbf{D} : \mathbf{D} = \mathbf{\Phi}\mathbf{U}, \|\mathbf{u}_i\|_0 \leq l, \forall i\}, \tag{2}$$

where $\mathbf{U}$ is the sparse coefficients for $\mathbf{D}$ over $\mathbf{\Phi}$. Given the dictionary $\mathbf{D} = \mathbf{\Phi}\mathbf{U}$, data $\mathbf{X}$ can then be represented as

$$\mathbf{X} = \mathbf{D}\mathbf{W} = \mathbf{\Phi}\mathbf{U}\mathbf{W}, \tag{3}$$

where $\mathbf{W}$ is the sparse coefficients over $\mathbf{D}$.

The hierarchical structures in (3) share some properties with deep architectures of learning models. Deep architectures have been empirically demonstrated as very effective for many complicated AI tasks [19]. Compared to a shallow model, a deep architecture is able to characterize complex data with alleviated overfitting. Our experimental results demonstrate that the hierarchical structures among dictionaries can also reduce overfitting and improve generalizability of the model.

In this paper, we propose a learning framework to learn the dictionary $\mathbf{D}$ and the sparse codes $\mathbf{W}$ in (3). The primary contributions of this paper include

- the proposed hierarchical sparse structures among an a-priori over-complete dictionary $\mathbf{\Phi}$, the pursued dictionary $\mathbf{D}$ and the given training data $\mathbf{X}$ as in (3);
- the formulation of a hierarchical sparse dictionary learning problem to learn $\mathbf{D}$ and $\mathbf{W}$ in Sect. 3; and
- the solution algorithm for the problem in Sect. 3.

3 Hierarchical Sparse Dictionary Learning

We formulate the problem of learning a dictionary $\mathbf{D}$ from data $\mathbf{X}$ and $\mathbf{X}$'s sparse representation $\mathbf{W}$ over $\mathbf{D}$, where $\mathbf{D}$ is built upon an a-priori over-complete dictionary $\mathbf{\Phi}$, as in the following optimization problem.

$$\begin{aligned} \min_{\mathbf{D},\mathbf{W}} \quad & \|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{W}\|_0 \leq k \\ & \mathbf{D} \in \mathcal{C} = \{\mathbf{D} : \mathbf{D} = \mathbf{\Phi}\mathbf{U}, \|\mathbf{u}_i\|_0 \leq l, \forall i\}. \end{aligned} \tag{4}$$

We denote the learning problem in (4) as Hierarchical Sparse Dictionary Learning (HiSDL). The major difficulty in HiSDL is that the feasible region $\mathcal{C}$ is non-convex and even not path-connected, and thus optimization over $\mathcal{C}$ is very challenging. We solve the problem by first giving an approximated sparsity of $\mathbf{D}$ on $\mathbf{\Phi}$ in Sect. 3.1, and then a corresponding optimization algorithm in Sect. 3.2.

3.1 Approximated sparsity of $\mathbf{D}$ on $\mathbf{\Phi}$

We first reformulate the feasible region constraint in (4) as a regularizer in the objective function, and then consider its convex approximation. Specifically, using the ℓ_1 convex relaxation of $\|\cdot\|_0$, we define an $\mathcal{C}$-function of $\mathbf{D}$ as follows,

$$\begin{aligned}\mathcal{C}(\mathbf{D}) &= \sum_i \min_{\mathbf{d}_i} \|\mathbf{u}_i\|_1 \quad \text{s.t.} \quad \mathbf{D} = \mathbf{\Phi}\mathbf{U} \\ &= \min_{\mathbf{D}} \|\mathbf{U}\|_1 \qquad \text{s.t.} \quad \mathbf{D} = \mathbf{\Phi}\mathbf{U}.\end{aligned}$$

Thus, the dictionary learning problem in (4) can be reformulated as

$$\begin{aligned}\min_{\mathbf{D},\mathbf{W}} \quad & \frac{1}{2}\|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F^2 + \gamma\|\mathbf{U}\|_1 \\ \text{s.t.} \quad & \|\mathbf{W}\|_0 \le k, \\ & \mathbf{D} = \mathbf{\Phi}\mathbf{U}.\end{aligned} \tag{5}$$

Then we consider a convex approximation of $\mathcal{C}(\mathbf{D})$ based on the following theorem.

Theorem 1. *Assume a $d \times p$ dictionary $\mathbf{\Phi}$ with incoherence μ, and $\mathbf{D} = \mathbf{\Phi}\mathbf{U}$ with all $\mathbf{u}_i$ k-sparse and $k < 1 + 1/\mu$, then*

$$\alpha\|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1 \le \|\mathbf{U}\|_1 \le \beta\|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1,$$

where $\alpha = \frac{1}{1+(p-1)\mu}$, $\beta = \frac{1}{1-(k-1)\mu}$. In particular, if $\mathbf{\Phi}$ is an orthonormal basis, then $\|\mathbf{U}\|_1 = \|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1$.

The proof of Theorem 1 is presented in the Appendix section.

Since $\mathbf{\Phi}$ is a pre-designed dictionary with a well-constrained incoherence, based on Theorem 1, we choose $\|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1$ to approximate $\mathcal{C}(\mathbf{D})$ and thus to regularize the sparsity of $\mathbf{D}$ on $\mathbf{\Phi}$. Furthermore, we relax and reformulate the sparse constraint of $\mathbf{W}$ as an ℓ_1-norm regularizer in the objective function. The resulting dictionary learning problem is thus as follows.

$$\min_{\mathbf{D},\mathbf{W}} \frac{1}{2}\|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F^2 + \lambda\|\mathbf{W}\|_1 + \gamma\|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1. \tag{6}$$

Due to the convexity of $\|\mathbf{\Phi}^\mathsf{T}\mathbf{D}\|_1$, the objective function in (6) is convex with respect to $\mathbf{D}$.

3.2 Optimization Algorithm

There are two key steps in a typical dictionary learning algorithm: sparse coding and dictionary update. In the sparse coding step, the goal is to find the sparse coefficients $\mathbf{W}$ with a fixed dictionary $\mathbf{D}$ from the last iteration. In the dictionary update step, $\mathbf{D}$ is further optimized with respect to the pursued $\mathbf{W}$. The objective function is therefore minimized in an alternating fashion.

For the objective function as in (6), the sparse coding step is similar to that in [15], that is, it is to find $\mathbf{W}$ by solving the following problem after fixing $\mathbf{D}$.

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F^2 + \lambda \|\mathbf{W}\|_1. \tag{7}$$

It is a classical linear inverse problem with l_1 regularization. We utilize the FISTA algorithm [20], due to its efficiency and robustness, to solve (7).

During the dictionary update step, the objective is to pursue the dictionary $\mathbf{D}$ by solving the following problem after fixing $\mathbf{W}$.

$$\min_{\mathbf{D}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F^2 + \gamma \|\mathbf{\Phi}^\top \mathbf{D}\|_1. \tag{8}$$

To solve the above problem, we introduce an auxiliary variable $\mathbf{H} = \mathbf{\Phi}^\top \mathbf{D}$. Thus, the problem in (8) can be reformulated as follows,

$$\hat{\mathbf{H}} = \arg\min_{\mathbf{H}} \frac{1}{2} \|\mathbf{X} - \mathbf{\Phi}^\dagger \mathbf{H}\mathbf{W}\|_F^2 + \gamma \|\mathbf{H}\|_1, \tag{9}$$

$$\hat{\mathbf{D}} = \mathbf{\Phi}^\dagger \hat{\mathbf{H}}, \tag{10}$$

where $\mathbf{\Phi}^\dagger = (\mathbf{\Phi}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}$.[1] The problem in (9) is again a linear inverse problem with ℓ_1 regularization. We can solve it similarly as for (7) in the sparse coding step. Thus, the entire procedure for solving (6) is presented in Algorithm 1.

Algorithm 1. Hierarchical Sparse Dictionary Learning (HiSDL)

```
Input: Data matrix $\mathbf{X} \in R^{m \times n}$, dictionary $\mathbf{\Phi}$
Initialize: $\lambda$, $\gamma$, $\mathbf{D}_0$
for $t = 1, 2, \ldots, T$ do
    // Sparse coding: solve (7)
    $\mathbf{W}_t = \arg\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}_{t-1}\mathbf{W}\|_F^2 + \lambda \|\mathbf{W}\|_1$
    // Dictionary update: solve (8)
    $\mathbf{H}_t = \arg\min_{\mathbf{H}} \frac{1}{2} \|\mathbf{X} - \mathbf{\Phi}^\dagger \mathbf{H}\mathbf{W}_t\|_F^2 + \gamma \|\mathbf{H}\|_1$
    $\mathbf{D}_t = \mathbf{\Phi}^\dagger \mathbf{H}_t$
end for
return $\mathbf{D}_T$
```

3.3 Analysis of HiSDL Algorithm

Atom selection in HiSDL Generally, the number of atoms in $\mathbf{D}$ is largely determined by the complexity of the given data, and is therefore difficult to determine a priori. Moreover, the non-convex nature of the objective function in (6) inevitably leads to non-global optima. Therefore, it is very challenging to

[1] $\mathbf{\Phi}\mathbf{\Phi}^\top$ is invertible since $\mathbf{\Phi}$ is an over-complete frame [9].

find the correct size of a dictionary $\mathbf{D}$ and its associated atoms that result in a good local minimum [2,21]. Interestingly, HiSDL as in Algorithm 1 has an "atom selection" property. In particular, the obsolete atoms in $\mathbf{D}$ will be automatically eliminated, and thereby the size of $\mathbf{D}$ is well-controlled. To verify this property of HiSDL, we first have the following lemma.

Lemma 1. *For any atom* $\mathbf{d}_i \in \mathbf{D}$*, if* $\mathbf{d}_i^t = \mathbf{0}$*, then* $\mathbf{d}_i^{t+1} = \mathbf{0}$*, where* $\mathbf{d}_i^t$ *is the* i*-th atom of* $\mathbf{D}$ *at the* t*-th iteration as in Algorithm 1.*

The proof of Lemma 1 is presented in the Appendix section.

Different from other state-of-the-art approaches as in [2,15], Lemma 1 states that if one atom degenerates to $\mathbf{0}$, then it will stay as $\mathbf{0}$ since then. This essentially addresses the dictionary pruning problem, i.e. the unused atoms are automatically set to zero. Indeed, if one atom dose not contribute much to the reduction of the empirical error $\|\mathbf{X} - \mathbf{D}\mathbf{W}\|_F$, then it will be set to zero in the dictionary update step based on the following theorem.

Theorem 2. *At iteration* t_0*, if* $\|\mathbf{\Psi}^\mathsf{T}\mathbf{R}_i\mathbf{W}^\mathsf{T}\|_\infty < \gamma$*, where* $\mathbf{\Psi} = \mathbf{\Phi}^\dagger$*, and* $\mathbf{R}_i = \mathbf{X} - \mathbf{D}_{-i}\mathbf{W}$ *is the empirical error without using* $\mathbf{d}_i$ *in* $\mathbf{D}$*, then* $\mathbf{d}_i = 0$ *for* $t > t_0$*.*

The proof of Theorem 2 is presented in the Appendix section.

Theorem 2 ensures that the unnecessary atoms will degenerate to $\mathbf{0}$ as the empirical error reduces during the learning process. We are therefore able to maintain a compact dictionary in an on-line fashion.

Computational Complexity of HiSDL The sparse coding step (7) and the dictionary update step (8) dominate the computational complexity of HiSDL. In particular, the sparse coding step and the dictionary update step are essentially the same constrained ℓ_1-minimization problem, of which the computational complexity is mainly from matrix multiplication when using soft-thresholding methods such as FISTA [20]. Specifically, if $\mathbf{X}$ is of dimension $d \times m$, $\mathbf{D}$ is of dimension $d \times n$, and $\mathbf{\Phi}$ is of dimension $d \times p$, where typically $m > p > n$ and $m > d, p > d$ [2], then the computational complexity of each soft-thresholding iteration in sparse coding is $O(mnd)$, and similarly $O(pdn)$ for dictionary update.

4 Related Work

Structured dictionary learning has been explored in previous works [22–24] from different perspectives. For example, in [22], a tree-like hierarchical structure is learned among the atoms in a dictionary, instead of treating each atom independently. Group sparsity among atoms is also considered in [23] and is applied to model spatial relations between atoms. In [24], a smooth prior on the sparse

[2] The number of samples m in $\mathbf{X}$ should be larger than p, the number of atoms in $\mathbf{\Phi}$, and $p > n$, the number of atoms in a more compact dictionary D. However, n is determined by the richness of $\mathbf{X}$, and may therefore be larger or smaller than d.

coefficients $\mathbf{W}$ is used in order to get a more stable representation. In contrast to these methods, we introduce a known dictionary representing the prior knowledge of the given data, and the hierarchical structure is imposed on the known dictionary and the learned dictionary rather than among the atoms in the learned dictionary. In addition, in our model, the known dictionary is used directly to regularize dictionary learning rather than to enforce structures in $\mathbf{W}$ as in [22–24]. As shown in this paper later, the use of the known dictionary and the hierarchical structures among the known dictionary and the learned dictionary enable a sparser representation with lower empirical errors.

5 Experimental Results

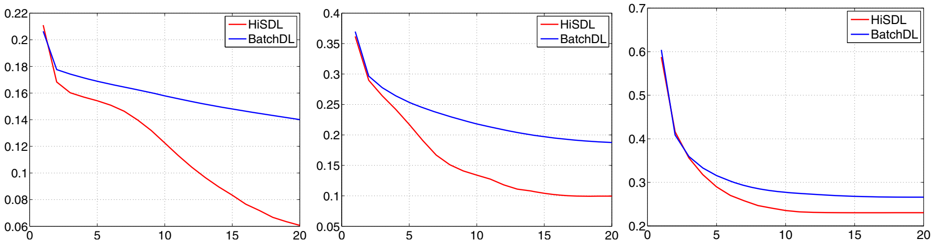
In this section, we present experimental results on synthetic data to empirically evaluate HiSDL. We also demonstrate the applications of HiSDL using real-world data. In particular, we test HiSDL on the following two datasets:

1. Synthetic data: we synthesize 200 time series of length 100 using DCT and Haar wavelets to simulate the real-world time series. DCT and Haar wavelets are composed into the a-priori over-complete dictionary $\mathbf{\Phi}$. Then a few atoms from $\mathbf{\Phi}$ are randomly selected and combined with amplitudes following a uniform distribution in $[-1, 1]$ into an atom in a dictionary $\mathbf{D}$, and in the end $\mathbf{D}$ has 100 atoms. A random sparse matrix $\mathbf{W}$ is then generated and used so as to generate the synthetic time series from $\mathbf{D}$.
2. Chemical plant time series (CPT): This dataset includes 1625 time series from various sensors monitoring an entire manufacture process of a chemical plant. Every time series is the output of one sensor, and each sensor collects one observation every minute. The data exhibit high heterogeneity in nature, e.g., there are both continuous and discrete time series, smooth and non-smooth time series, etc.

5.1 Evaluation on Empirical Errors

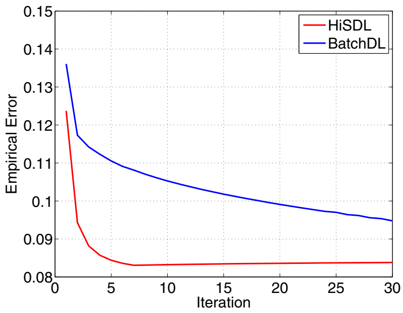
Fig. 1 shows the empirical errors of HiSDL and of the state-of-the-art method [15], denoted as `BatchDL`, during learning iterations with different parameter λ values on the synthetic data ($\gamma = 0.05\lambda$; other λ values give similar trends; the optimal λ and γ combinations are from grid search). For each of the λ values, the sparsity of the learned W is relatively similar from both HiSDL and `BatchDL`. However, HiSDL consistently achieves smaller empirical errors than `BatchDL` after each learning iteration. This demonstrates that by introducing a regularization of $\mathbf{D}$ with respect to an a-priori over-complete $\mathbf{\Phi}$, the optimization process in (4) may have a better chance to end up at a better local minimum within the reduced (and better) search space. In addition, HiSDL achieves smaller empirical errors faster than `BatchDL`. This implies that HiSDL can quickly find a more accurate sparse representation than `BatchDL`.

We further compare the performance of HiSDL and `BatchDL` on the CPT dataset. We randomly pick one-day data in the dataset for dictionary learning,



(a) $\lambda = 0.05$ (b) $\lambda = 0.1$ (c) $\lambda = 0.2$

Fig. 1. Empirical errors vs learning iterations on synthetic data



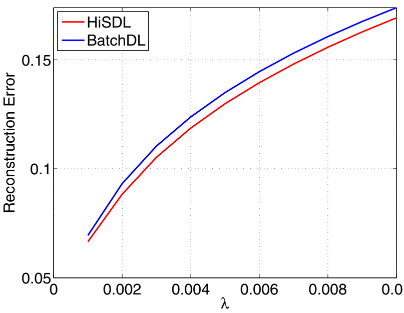


Fig. 2. Empirical errors vs learning iterations on CPT

Fig. 3. Reconstruction errors on CPT testing data

and the data from a later day for testing. For CPT, $\mathbf{\Phi}$ is constructed as a combination of DCT and Haar wavelets, of which the number of atoms is twice as the length of time series. However, the learned dictionary is composed of only 120 atoms. Fig. 2 shows the empirical errors during learning iterations with $\lambda = 0.001$ and $\gamma = 0.02\lambda$ (the λ and γ values and combinations are optimized from grid search). Again, on the real dataset, HiSDL achieves smaller empirical errors faster than `BatchDL`. Fig. 3 shows the reconstruction errors of HiSDL and `BatchDL` on CPT testing data with different λ values. In Fig. 3, HiSDL consistently achieves smaller reconstruction errors than `BatchDL`, which implies that HiSDL is able to find more robust and generalizable dictionaries than `BatchDL`.

5.2 Evaluation on Atom Recovery

Fig. 4 presents some sample atoms learned from HiSDL on the synthetic data. These atoms exhibit finer structures as a linear combination of DCT and Haar wavelets, which demonstrates the capability of HiSDL recovering the building structures of the data. However, as shown in Fig. 5, the learned atoms by `BatchDL` on the synthetic data appear less structured, more homogeneous and do not conform to the true structures underlying the data. This is due to that fact that

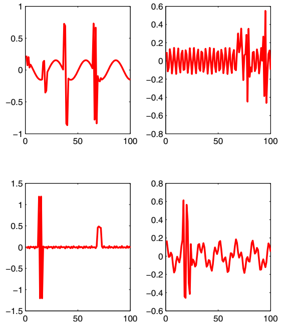


Fig. 4. Sample atoms from HiSDL on synthetic data

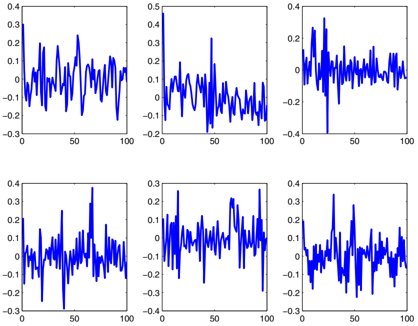
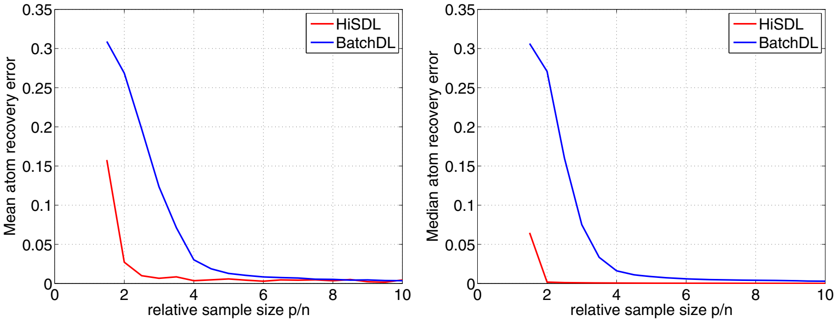


Fig. 5. Sample atoms from `BatchDL` on synthetic data



(a) Mean recovery error

(b) Median recovery error

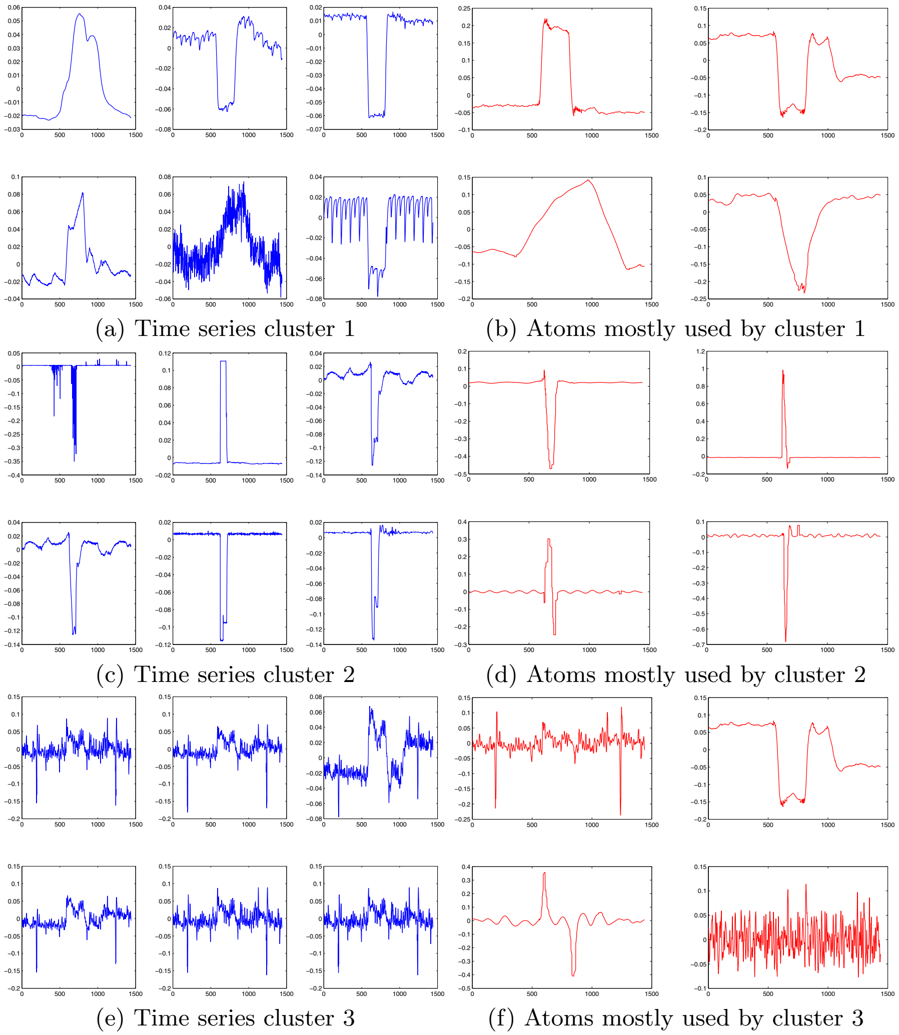
Fig. 6. Recovery errors vs sample size

`BatchDL` constraints the norm of each atom and thus biases the search of the atoms towards a bad local minimum.

To further test the performance of the methods on the discovery of latent atoms, we evaluate HiSDL and `BatchDL` on the blind source separation problem [25,26] on a set of synthetic datasets. These synthetic datasets have the same a-priori over-complete dictionary $\mathbf{\Phi}$ and dictionary $\mathbf{D}$ as generated as before, but different number of time series (150, 200, 250 up to 1000). The success of the recovery of latent atoms relies on the ratio of the given sample size to the number of latent atoms. Intuitively, we can only expect to recover all latent atoms when every atom has been sufficiently used in the given sample set. Naturally, this recovery goal is more likely to be achieved when we have a large dataset.

Denote the learned dictionary as $\hat{\mathbf{D}}$, and the relative recovery error of each atom $\mathbf{d}_i \in \mathbf{D}$ is then defined as follows,

$$r_i = \min_{\hat{\mathbf{d}}_j \in \hat{\mathbf{D}}} \{1 - \cos \Theta(\hat{\mathbf{d}}_j, \mathbf{d}_i)\}, \tag{11}$$



(a) Time series cluster 1 (b) Atoms mostly used by cluster 1

(c) Time series cluster 2 (d) Atoms mostly used by cluster 2

(e) Time series cluster 3 (f) Atoms mostly used by cluster 3

Fig. 7. Sample time series clusters in CPT and the corresponding mostly used atoms

where $\Theta(\hat{\mathbf{d}}_j, \mathbf{d}_i)$ is the angle between $\hat{\mathbf{d}}_j$ and $\mathbf{d}_i$. In specific, $r_i \in [0, 1]$, and if there exists $\hat{\mathbf{d}}_j \in \hat{\mathbf{D}}$ that satisfies $\mathbf{d}_i = \hat{\mathbf{d}}_j$, then $r_i = 0$. In Fig. 6, we show the mean/median atom recovery error vs the relative sample size (i.e., p/n in Fig. 6, the ratio of sample sizes over the dictionary size) on the synthetic datasets.

In this set of experiments, the initial dictionary size for both algorithms is 150, and the dictionary of ground truth is of size 100. Each point in Fig. 6 is the mean of 5 experiments under the same training set, but with different initialization. We can see that when the sample size is large, i.e. sufficient information is provided, both HiSDL and `BatchDL` work well. However, when the sample size is small, such as two to four times the number of latent atoms, HiSDL shows a substantially superior performance. It further demonstrates that by integrating a priori knowledge of the given dataset, HiSDL achieves better generalizability.

5.3 Evaluation on Sparse Codes

We also explore the ability of HiSDL to process heterogeneous time series data by studying the clustering results using sparse codes. Intuitively, if the learned dictionary successfully characterizes the given data, the clustering generated from their associated sparse codes should exhibit good structures. Fig. 7 shows some clusters of time series from CPT data and their frequently used dictionary atoms. The clustering is done by using spectral clustering algorithm [27] on the sparse codes. As Fig. 7 shows, the CPT data are heterogeneous including step signals, piece-wise linear signals, periodical signals and even brownian motion-like signals. However, after representing the time series over the learned hierarchical dictionaries, the clustering over their sparse codes shows high homogeneity within each cluster, and the frequently used atoms for each cluster represent dominant features of the cluster. This demonstrates that HiSDL has the capability of capturing the most representative features from even highly heterogeneous time series.

6 Conclusion

In this paper, we introduce a novel dictionary learning framework HiSDL which utilizes a hierarchical sparse structure to characterize observed data. The experiments demonstrate that the hierarchical sparse structure within the model regularizes potential solutions, and enables smaller empirical errors. In addition, HiSDL is able to identify the most representative latent atoms from a few training samples, and thus well characterizes the training data. Future work may include constructing nonlinear and deep structures on dictionary learning models. Also it would be interesting to see a more thorough evaluation of HiSDL on other types of data, such as videos and images.

Appendix

Proof of Theorem 1

Proof. We first show the right part of the inequality, $\beta\|\mathbf{\Phi}^\top\mathbf{D}\|_1 \geq \|\mathbf{U}\|_1$.

since $\mathbf{D} = \mathbf{\Phi}\mathbf{U}$, $\|\mathbf{u}_i\|_0 \leq k, \forall \mathbf{u}_i \in \mathbf{U}$, it follows that

$$\begin{aligned}\|\mathbf{\Phi}^\top\mathbf{D}\|_1 &= \|\mathbf{\Phi}^\top\mathbf{\Phi}\mathbf{U}\|_1 \\ &= \sum_i \|\mathbf{\Phi}^\top\mathbf{\Phi}\mathbf{u}_i\|_1. \end{aligned} \tag{12}$$

Moreover, according to the definition of incoherence,

$$\begin{aligned}\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{\Phi}\mathbf{u}_i\|_1 &\geq \sum_j (1-(k-1)\mu)|u_{ij}|_1 \\ &= (1-(k-1)\mu)\|\mathbf{u}_i\|_1. \end{aligned} \tag{13}$$

Consequently, we have

$$\begin{aligned}\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1 &\geq \sum_i \|\mathbf{u}_i\|_1(1-(k-1)\mu) \\ &= (1-(k-1)\mu)\|\mathbf{U}\|_1, \end{aligned} \tag{14}$$

and let $\beta = \frac{1}{1-(k-1)\mu}$, it follows that $\beta\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1 \geq \|\mathbf{U}\|_1$.

We next prove the left part of the inequality, $\alpha\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1 \leq \|\mathbf{U}\|_1$.

Proceeding from (12), we further have

$$\begin{aligned}\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{\Phi}\mathbf{u}_i\|_1 &\leq \sum_j (1-(p-1)\mu)|u_{ij}|_1 \\ &= (1-(p-1)\mu)\|\mathbf{u}_i\|_1. \end{aligned} \tag{15}$$

and as a result,

$$\begin{aligned}\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1 &= \sum_i \|\mathbf{\Phi}^{\mathsf{T}}\mathbf{\Phi}\mathbf{u}_i\|_1 \\ &\leq (1-(p-1)\mu)\|\mathbf{U}\|_1. \end{aligned} \tag{16}$$

Since $\alpha = \frac{1}{1+(p-1)\mu}$, we therefore have $\alpha\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1 \leq \|\mathbf{U}\|_1$. ■

Proofs of Lemma 1 and Theorem 2

We first show the proof of Lemma 1.

Proof. If $\mathbf{d}_i^t = \mathbf{0}$, then

$$\mathbf{W}_{t+1} = \arg\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{X} - \mathbf{D}_t\mathbf{W}\|_F^2 + \lambda\|\mathbf{W}\|_1 \tag{17}$$

implies that the ith row of $\mathbf{W}_{t+1}$, $\mathbf{w}_{t+1}^i$, is also $\mathbf{0}$.

Now consider

$$\mathbf{D}_{t+1} = \arg\min_{\mathbf{D}} = \frac{1}{2}\|\mathbf{X} - \mathbf{D}\mathbf{W}_{t+1}\|_F^2 + \lambda\|\mathbf{\Phi}^{\mathsf{T}}\mathbf{D}\|_1, \tag{18}$$

since $\mathbf{w}_{t+1}^i = \mathbf{0}$, we therefore have $\mathbf{d}_i^{t+1} = \mathbf{0}$. ■

Having Lemma 1 proved, we can then proceed to prove Theorem 2.

Proof. At iteration t_0, let $g(\mathbf{H}) = \gamma\|\mathbf{H}\|_1 + \frac{1}{2}\|\mathbf{X} - \boldsymbol{\Psi}\mathbf{H}\mathbf{W}\|_F^2$, and assume $\hat{\mathbf{H}} = \arg\min_H g(\mathbf{H})$, we then have

$$\partial g(\hat{\mathbf{H}}) = \gamma\partial\|\hat{\mathbf{H}}\|_1 + \boldsymbol{\Psi}^\top(\mathbf{X} - \boldsymbol{\Psi}\hat{\mathbf{H}}\mathbf{W})\mathbf{W}^\top \ni \mathbf{0}. \tag{19}$$

Rewrite $\mathbf{H}$ as $\mathbf{H} = \mathbf{H}_i + \mathbf{H}_{-i}$, where $\mathbf{H}_i = [\mathbf{0}, \ldots, \mathbf{0}, \mathbf{h}_i, \mathbf{0}, \ldots, \mathbf{0}]$ and $\mathbf{H}_{-i} = [\mathbf{h}_1, \ldots, \mathbf{h}_{i-1}, \mathbf{0}, \mathbf{h}_{i+1}, \ldots, \mathbf{h}_n]$, then we have

$$\partial g(\hat{\mathbf{H}}_i) \ni \mathbf{0}, \partial g(\hat{\mathbf{H}}_{-i}) \ni \mathbf{0}. \tag{20}$$

Note that

$$\partial g(\hat{\mathbf{H}}_i) = \gamma\partial\|\hat{\mathbf{H}}_i\|_1 + \boldsymbol{\Psi}^\top(\mathbf{R}_i - \boldsymbol{\Psi}\hat{\mathbf{H}}_i\mathbf{W})\mathbf{W}^\top, \tag{21}$$

where $\mathbf{R}_i = \mathbf{X} - \boldsymbol{\Psi}\hat{\mathbf{H}}_{-i}\mathbf{W} = \mathbf{X} - \mathbf{D}_{-i}\mathbf{W}$.

Consider the condition $\|\boldsymbol{\Psi}^\top\mathbf{R}_i\mathbf{W}^\top\|_\infty < \gamma$, combined with the subgradient of ℓ_1-norm that $\partial\|x\|_1 = (-1, 1)$, we have

$$\hat{\mathbf{H}}_i = \mathbf{0} \Leftrightarrow \partial g(\hat{\mathbf{H}}_i) \ni \mathbf{0}. \tag{22}$$

When $\mathbf{h}_i = \mathbf{0}$, since $\mathbf{D} = \boldsymbol{\Psi}\mathbf{H}$, it follows that $\mathbf{d}_i = \mathbf{0}$ at $t_0 + 1$. According to Lemma 1, we consequently have $\mathbf{d}_i = 0$ for $t > t_0$. ■

References

1. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Supervised dictionary learning. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) NIPS, Curran Associates Inc., pp. 1033–1040 (2008)
2. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing **54**, 4311–4322 (2006)
3. Elad, M.: Sparse and redundant representation modeling: What next? IEEE Signal Processing Letters **19**, 922–928 (2012)
4. Bian, X., Krim, H.: Robust Subspace Recovery via Bi-Sparsity Pursuit (2014). ArXiv e-prints
5. Soltanolkotabi, M., Elhamifar, E., Candes, E.: Robust subspace clustering (2013). arXiv preprint arXiv:1301.2603
6. Huang, K., Aviyente, S.: Sparse representation for signal classification. In: Advances in neural information processing systems, pp. 609–616 (2006)
7. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image Processing **15**, 3736–3745 (2006)
8. Rubinstein, R., Zibulevsky, M., Elad, M.: Double sparsity: Learning sparse dictionaries for sparse signal approximation. IEEE Transactions on Signal Processing **58**, 1553–1564 (2010)
9. Mallat, S.: A wavelet tour of signal processing. Academic press (1999)
10. Candes, E.J., Donoho, D.L.: Curvelets: A surprisingly effective nonadaptive representation for objects with edges. Technical report, DTIC Document (2000)

11. Yi, S., Labate, D., Easley, G.R., Krim, H.: A shearlet approach to edge analysis and detection. IEEE Transactions on Image Processing **18**, 929–941 (2009)
12. Labate, D., Lim, W.Q., Kutyniok, G., Weiss, G.: Sparse multidimensional representation using shearlets. In: Optics & Photonics 2005, International Society for Optics and Photonics, pp. 59140U–59140U (2005)
13. Eldar, Y.C., Kutyniok, G.: Compressed sensing: theory and applications. Cambridge University Press (2012)
14. Candès, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Communications on pure and applied mathematics **59**, 1207–1223 (2006)
15. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th Annual International Conference on Machine Learning, 689–696. ACM (2009)
16. Kreutz-Delgado, K., Murray, J.F., Rao, B.D., Engan, K., Lee, T.W., Sejnowski, T.J.: Dictionary learning algorithms for sparse representation. Neural computation **15**, 349–396 (2003)
17. Elad, M., Figueiredo, M.A., Ma, Y.: On the role of sparse and redundant representations in image processing. Proceedings of the IEEE **98**, 972–982 (2010)
18. Lee, H., Ekanadham, C., Ng, A.: Sparse deep belief net model for visual area v2. In: Advances in neural information processing systems, pp. 873–880 (2007)
19. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning **2**, 1–127 (2009)
20. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences **2**, 183–202 (2009)
21. Rubinstein, R., Faktor, T., Elad, M.: K-svd dictionary-learning for the analysis sparse model. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5405–5408. IEEE (2012)
22. Jenatton, R., Mairal, J., Obozinski, G., Bach, F.: Proximal methods for hierarchical sparse coding. The Journal of Machine Learning Research **12**, 2297–2334 (2011)
23. Jenatton, R., Audibert, J.Y., Bach, F.: Structured variable selection with sparsity-inducing norms. The Journal of Machine Learning Research **12**, 2777–2824 (2011)
24. Bradley, D.M., Bagnell, J.A.: Differential sparse coding (2008)
25. Zibulevsky, M., Pearlmutter, B.: Blind source separation by sparse decomposition in a signal dictionary. Neural Computation **13**, 863–882 (2001)
26. Li, Y., Cichocki, A., Amari, S.: Analysis of sparse representation and blind source separation. Neural Computation **16**, 1193–1234 (2004)
27. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: Advances in Neural Information Processing Systems, pp. 849–856. MIT Press (2001)