Dictionary learning with large step gradient descent for sparse representations

Mailhé, B; Plumbley, MD

Dictionary Learning with Large Step Gradient Descent for Sparse Representations

Boris Mailhé and Mark D. Plumbley

Queen Mary University of London
School of Electronic Engineering and Computer Science
Centre for Digital Music
Mile End Road, London E1 4NS, United Kingdom
firstname.name@eecs.qmul.ac.uk

Abstract. This work presents a new algorithm for dictionary learning. Existing algorithms such as MOD and K-SVD often fail to find the best dictionary because they get trapped in a local minimum. Olshausen and Field's Sparsenet algorithm relies on a fixed step projected gradient descent. With the right step, it can avoid local minima and converge towards the global minimum. The problem then becomes to find the right step size. In this work we provide the expression of the optimal step for the gradient descent but the step we use is twice as large as the optimal step. That large step allows the descent to bypass local minima and yields significantly better results than existing algorithms. The algorithms are compared on synthetic data. Our method outperforms existing algorithms both in approximation quality and in perfect recovery rate if an oracle support for the sparse representation is provided.

Keywords: Dictionary learning, sparse representations, gradient descent

1 Introduction

In the method of sparse representations, a signal is expressed as a linear combination of a few vectors named *atoms* taken from a set called a *dictionary*. The sparsity constraint induces that any given dictionary can only represent a small subset of all possible signals, so the dictionary has to be adapted to the data being represented. Good pre-constructed dictionaries are known for common classes of signals, but sometimes it is not the case, for example when the dictionary has to discriminate against perturbations coming from noise [2]. In that case, the dictionary can be learned from examples of the data to be represented.

Several different algorithms have been proposed to learn the dictionary. Many of them iteratively optimize the dictionary and the decomposition [5,3,1]. The difference between those algorithms is the way they update the dictionary to fit a known decomposition. In particular, Olshausen and Field's Sparsenet algorithm [5] uses a fixed step gradient descent. In this work we observe that all those update methods are suboptimal even if the right support for the decomposition is known.

This work presents a modification to the Sparsenet algorithm that enables it to bypass local minima. We use the fact that the optimal step of the gradient descent can easily be obtained, then multiply it by constant larger than 1. Empirical results show that our method often allows the optimization to reach the global minimum.

2 Dictionary Learning

2.1 Problem

Let $\mathbf{S}$ be a $D \times N$ matrix of N training signals $\{\mathbf{s}_n\}_{n=1}^{N},\ \mathbf{s}_n \in \mathbb{R}^D$. Dictionary learning consists in finding a dictionary $\mathbf{\Phi}$ of size $D \times M$ with $M \geq D$ and sparse coefficients $\mathbf{X}$ such that $\mathbf{S} \approx \mathbf{\Phi X}$. For example, if the exact sparsity level K is known, the problem can be formalized as minimizing the error cost function

$$f(\mathbf{\Phi}, \mathbf{X}) = \|\mathbf{S} - \mathbf{\Phi X}\|_F^2 \tag{1}$$

under the constraints

$$\forall m \in [1, M],\ \|\boldsymbol{\varphi}_m\|_2 = 1 \tag{2}$$

$$\forall n \in [1, N],\ \|\mathbf{x}_n\|_0 \leq K \tag{3}$$

with $\boldsymbol{\varphi}$ an atom (or column) of $\mathbf{\Phi}$ and $\|\mathbf{x}_n\|_0$ the number of non-zero coefficients in the n^{th} column of $\mathbf{X}$.

2.2 Algorithms

Many dictionary learning algorithms follow an alternating optimization method. When the dictionary $\mathbf{\Phi}$ is fixed, estimating the sparse coefficients $\mathbf{X}$ is a sparse representation problem that can be approximately solved by algorithms such as Orthogonal Matching Pursuit (OMP) [6]. Existing algorithms differ in the way they update the dictionary $\mathbf{\Phi}$ once the coefficients $\mathbf{X}$ are fixed:

- Sparsenet [5] uses a projected gradient descent with a fixed step α:

$$\mathbf{R} = \mathbf{S} - \mathbf{\Phi X} \tag{4}$$

$$\nabla f = -\mathbf{R}\mathbf{x}^{mT} \tag{5}$$

$$\boldsymbol{\varphi}_m \leftarrow \boldsymbol{\varphi}_m - \alpha \nabla f \tag{6}$$

$$\boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2} \tag{7}$$

  with $\mathbf{x}^m$ the m^{th} line of $\mathbf{X}$.
- MOD [3] directly computes the dictionary that minimizes the error f when the coefficients are fixed. The result is given by a pseudo-inverse:

$$\mathbf{\Phi} \leftarrow \mathbf{S}\mathbf{X}^{+} \tag{8}$$

$$\forall m \in [1, M],\ \boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2} \tag{9}$$

– K-SVD [1] jointly re-estimates each atom and the amplitude of its non-zero coefficients. For each atom $\boldsymbol{\varphi}_m$, the optimal choice is the principal component of a restricted "error" $\mathbf{E}^{(m)}$ obtained by considering the contribution of $\boldsymbol{\varphi}_m$ alone and removing all other atoms.

$$\mathbf{E}^{(m)} = \mathbf{R} + \boldsymbol{\varphi}_m \mathbf{x}^m \tag{10}$$

$$\boldsymbol{\varphi}_m \leftarrow \underset{\|\boldsymbol{\varphi}\|_2=1}{\operatorname{argmin}} \left\| \mathbf{E}^{(m)} - \boldsymbol{\varphi}\boldsymbol{\varphi}^T \mathbf{E}^{(m)} \right\|_F^2 \tag{11}$$

$$= \underset{\|\boldsymbol{\varphi}\|_2=1}{\operatorname{argmax}} \boldsymbol{\varphi}^T \mathbf{E}^{(m)} {\mathbf{E}^{(m)}}^T \boldsymbol{\varphi} \tag{12}$$

$$\mathbf{x}^m \leftarrow \boldsymbol{\varphi}_m^T \mathbf{E}^{(m)} \tag{13}$$

3 Motivations for an Adaptive Gradient Step Size

This section details an experimental framework used to compare the dictionary update methods presented in Section 2.2. We then show that MOD and K-SVD often get trapped in a local minimum but that with the right step, Sparsenet is more likely to find the global minimum.

3.1 Identifying the Global Optimum: Learning with a Fixed Support

We want to be able to check whether the solution found by an algorithm is the best one. It is easy in the noiseless case: if the training signals are exactly sparse on a dictionary, then there is at least one decomposition that leads to an error of 0: the one used for synthesizing the signals. In that case, a factorization $(\boldsymbol{\Phi}, \mathbf{X})$ is globally optimal if and only if the value of its error cost (1) is 0.

Dictionary learning algorithms often fail at that task because of mistakes done in the sparse representation step: when the dictionary is fixed, tractable sparse approximation algorithms typically fail to recover the best coefficients, although there are particular dictionaries for which the sparse representation is guaranteed to succeed [7]. In order to observe the behavior of the different dictionary update methods, we can simulate a successful sparse representation by using an oracle support: instead of running a sparse representation algorithm, the support used for the synthesis of the training signals is used as an input to the algorithm and only the values of the non-zero coefficients is updated by quadratic optimization. The dictionary learning algorithm is then simplified into Algorithm 1.

3.2 Empirical Observations on Existing Algorithms

We ran a simulation to check whether existing update methods are able to recover the best dictionary once the support is known. Each data set is made of a dictionary containing i.i.d. atoms drawn from a uniform distribution on

Algorithm 1 $(\mathbf{\Phi}, \mathbf{X}) = \text{dict\_learn}(\mathbf{S}, \sigma)$

$\mathbf{\Phi} \leftarrow$ random dictionary
while not converged **do**
  $\forall n, \mathbf{x}_n^{\boldsymbol{\sigma}_n} \leftarrow \mathbf{\Phi}_{\boldsymbol{\sigma}_n}^{+} \mathbf{s}_n$
  $\mathbf{\Phi} \leftarrow \text{dict\_update}(\mathbf{\Phi}, \mathbf{S}, \mathbf{X})$
end while

the unit sphere. For each dictionary, 256 8-sparse signals were synthesized by drawing uniform i.i.d. 8-sparse supports and i.i.d. Gaussian amplitudes. Then each algorithm was run for 1000 iterations starting from a random dictionary. The oracle supports of the representations were provided as explained in Section 3.1.

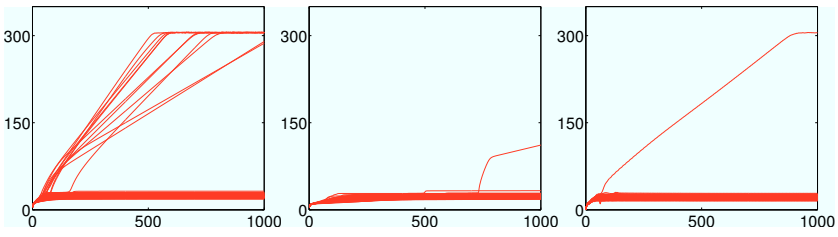
Figure 1 shows the evolution of the SNR $= -10 \log_{10} \frac{\|\mathbf{R}\|_2^2}{\|\mathbf{S}\|_2^2}$ over the execution of the algorithm for each data set. 300dB is the highest SNR that can be reached due to numerical precision. Moreover, we ran some longer simulations and never saw an execution fail to reach 300dB once a threshold of 100dB was passed For each algorithm, the plots show how many runs converged to a global minimum and how fast they did it.

K-SVD found a global minimum in 17 cases and has the best convergence speed of all studied algorithms. MOD only converged to a global minimum in 1 case and shows a tendency to evolve by steps, so even after a large number of iterations it is hard to tell whether the algorithm has converged or not. The best results were obtained when running Sparsenet with a step size $\alpha = 0.05$. In that case most runs converge to a global optimum although the convergence speed is more variable than with K-SVD. The behavior of Sparsenet highly depends on the choice of α. In our case a step of 0.1 is too large and almost always prevented the algorithm to converge, but a step of 0.01 is too small and leads to a very slow convergence.

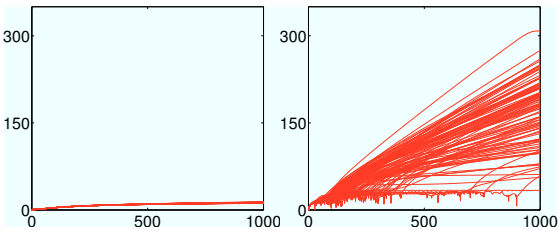
Moreover, Sparsenet outperforms MOD although they both attempt to solve the same least-square problem. MOD finds that minimum in only one iteration, but if each Sparsenet dictionary update was allowed to iterate on its gradient descent with a well chosen step, it would converge towards the result of the MOD update. So the source of the gain is unlikely to be that the step $\alpha = 0.05$ is well adapted to the descent, but rather that it is larger than what an optimal step would be, thus allowing the descent to jump over local minima. The fact that the SNR sometimes decreases at one iteration for Sparsenet with $\alpha = 0.05$ also hints at a larger than optimal step size.

4 Large Step Gradient Descent

This section presents our method to choose the step size of the gradient descent. Our method is based on optimal step gradient descent, but we purposefully choose a step size that is larger than the optimal one.



(a) K-SVD (b) MOD (c) Sparsenet, $\alpha = 0.1$



(d) Sparsenet, $\alpha = 0.01$ (e) Sparsenet, $\alpha = 0.05$

Fig. 1: Approximation SNR depending on the iteration. K-SVD and MOD often get trapped in a local minimum. With $\alpha = 0.05$, Sparsenet avoids local minima, but $\alpha = 0.1$ is too large and $\alpha = 0.01$ is too small.

4.1 Optimal step projected gradient descent

When fixing the coefficients and the whole dictionary but one atom $\boldsymbol{\varphi}_m$, there is a closed-form solution for the best atom $\boldsymbol{\varphi}_m^*$ that minimizes the cost function (1) [4].

$$\boldsymbol{\varphi}_m^* = \underset{\|\boldsymbol{\varphi}_m\|_2=1}{\operatorname{argmin}} \|\mathbf{S} - \boldsymbol{\Phi}\mathbf{X}\|_F^2 \tag{14}$$

$$= \underset{\|\boldsymbol{\varphi}_m\|_2=1}{\operatorname{argmin}} \left\|\mathbf{E}^{(m)} - \boldsymbol{\varphi}_m \mathbf{x}^m\right\|_F^2 \tag{15}$$

with $\mathbf{E}^{(m)}$ the restricted errors described for K-SVD in Equation (10).

$$\left\|\mathbf{E}^{(m)} - \boldsymbol{\varphi}_m \mathbf{x}^m\right\|_F^2 = \left\|\mathbf{E}_k^{(m)}\right\|_F^2 - 2\left\langle \mathbf{E}_k^{(m)}, \boldsymbol{\varphi}_m \mathbf{x}^m \right\rangle + \|\boldsymbol{\varphi}_m \mathbf{x}^m\|_F^2 \tag{16}$$

$\left\|\mathbf{E}_k^{(m)}\right\|_F^2$ is constant with respect to $\boldsymbol{\varphi}_m$. If $\boldsymbol{\varphi}_m$ is constrained to be unitary, then $\|\boldsymbol{\varphi}_m \mathbf{x}^m\|_F^2 = \|\mathbf{x}^m\|_2^2$ is also constant with respect to $\boldsymbol{\varphi}_m$. So the only variable

term is the inner product and the expression of the optimum $\boldsymbol{\varphi}_m^*$ is given by:

$$\boldsymbol{\varphi}_m^* = \underset{\|\boldsymbol{\varphi}_m\|_2=1}{\operatorname{argmax}} \left\langle \mathbf{E}^{(m)} \mathbf{x}^{mT}, \boldsymbol{\varphi}_m \right\rangle \tag{17}$$

$$= \frac{\mathbf{E}^{(m)} \mathbf{x}^{mT}}{\left\| \mathbf{E}^{(m)} \mathbf{x}^{mT} \right\|_2} \quad . \tag{18}$$

The link with the gradient appears when developing the Expression (18):

$$\boldsymbol{\varphi}_m^* \propto \left(\mathbf{R} + \boldsymbol{\varphi}_m \mathbf{x}^m\right) \mathbf{x}^{mT} \tag{19}$$

$$\propto \boldsymbol{\varphi}_m + \frac{1}{\|\mathbf{x}^m\|_2^2} \mathbf{R} \mathbf{x}^{mT} \quad . \tag{20}$$

Starting from the original atom, the global best atom $\boldsymbol{\varphi}_m^*$ can be obtained with only one iteration of gradient descent and the optimal step α^* of the descent is the inverse of the energy of the amplitude coefficients.

$$\alpha^* = \frac{1}{\|x^m\|_2^2} \tag{21}$$

5 Experimental Validation

This section presents dictionary learning experiments using gradient descent dictionary updates with the step sizes α^* and $2\alpha^*$. The comparison between them shows that the use of a larger than optimal step size improves the results.

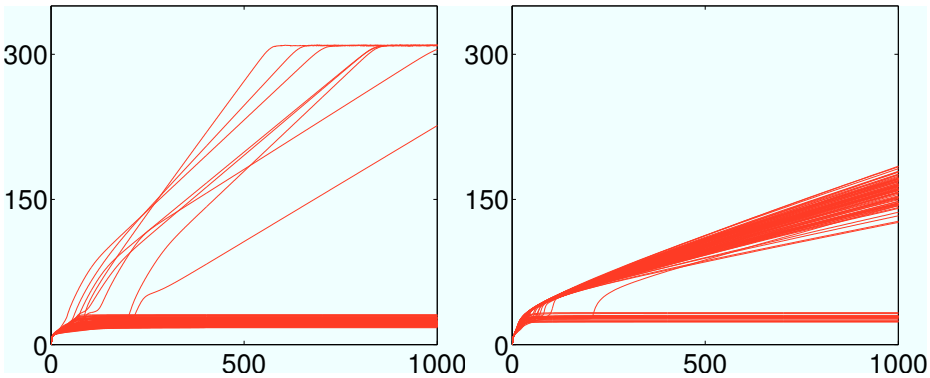
5.1 Learning with a Fixed Support

This experiment uses the same setup as the one presented in Section 3.2. We ran Sparsenet with the optimal step size α^* defined in Equation (21) and a larger step size $2\alpha^*$. As expected, the optimal step gradient descent almost always gets trapped in a local minimum. Doubling that step greatly improves the recovery rate from 8% to 79%.

5.2 Complete learning

We also compared the different update rules in the context of a complete dictionary learning, i.e. without the use of an oracle support. The sparse decomposition step was performed using OMP.

Figure 3 shows the repartition of the SNR obtained by each algorithm. The different algorithms are sorted by increasing average SNR. For Sparsenet we used the step size $\alpha = 0.05$ which was well suited to the fixed support case. With that choice Sparsenet slightly outperforms K-SVD by 0.01 dB, but in practical cases one might not have access to such previous knowledge to finely tune the step size α. Our large step gradient achieved the best average SNR. It outperforms K-SVD and the fixed step Sparsenet by an average 0.5 dB and converged to a better solution than K-SVD in 98 cases over 100.



(a) Optimal step gradient descent, $\alpha = \alpha^*$ (b) Large step gradient descent, $\alpha = 2\alpha^*$

Fig. 2: Approximation SNR depending on the iteration. The optimal gradient descent only succeeds 8 times whereas using a $2\alpha^*$ step succeeds 79 times.

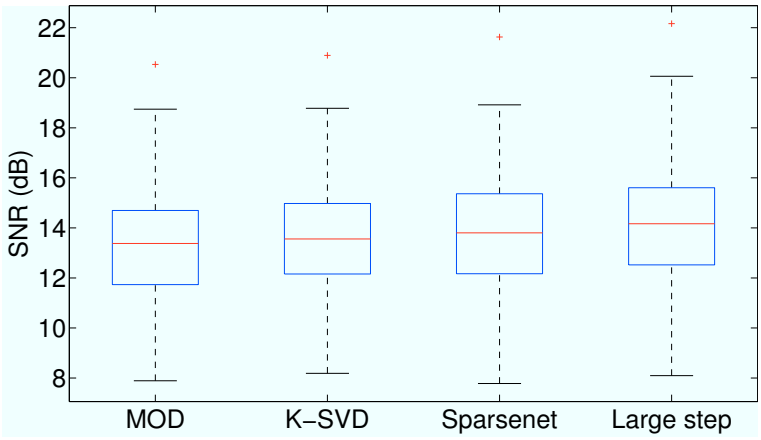


Fig. 3: Repartition of the SNR after learning dictionaries on 100 random data sets with different algorithms. The proposed large step gradient descent results in an average 0.5dB improvement over K-SVD.

6 Conclusion

We have presented a dictionary learning algorithm capable of better approximation quality of the training signals than K-SVD. That algorithm uses a gradient descent with an adaptive step guaranteed to be higher than the optimal step. The large step allows the descent to bypass local minima and converge towards the global minimum.

While our algorithm yields much better recovery rates than the existing ones, it can still be improved. With the step size $2\alpha^*$, the descent still gets trapped in a local minimum in 21% of the cases in our experiments. One could think of using an even larger step, but the algorithm then becomes unstable and fails to converge at all. The solution could be to use a hybrid algorithm that starts with large step gradient descent to find the attraction basin of a global minimum, then switches to one of the fast converging algorithms such as K-SVD to find the minimum itself.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing 54(11), 4311 –4322 (nov 2006)
2. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image Processing 15(12), 3736 – 3745 (Dec 2006)
3. Engan, K., Aase, S., Hakon Husoy, J.: Method of optimal directions for frame design. In: Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on. vol. 5, pp. 2443 –2446 vol.5 (1999)
4. Mailhé, B., Lesage, S., Gribonval, R., , Vandergheynst, P., Bimbot, F.: Shift-invariant dictionary learning for sparse representations: extending k-svd. In: in Proc. EUSIPCO (2008)
5. Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381, 607–609 (jun 1996)
6. Pati, Y., Rezaiifar, R., Krishnaprasad, P.: Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In: Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on. pp. 40 –44 vol.1 (Nov 1993)
7. Tropp, J.: Greed is good: algorithmic results for sparse approximation. IEEE Transactions on Information Theory 50(10), 2231 – 2242 (Oct 2004)