Sparse ICA: 자연영상의 효율적인 코딩?

최승진, 이오영
충북대학교 전기공학과
schoi@engine.chungbuk.ac.kr

SPARSE ICA: EFFICIENT CODING OF NATURAL SCENES?

Seungjin CHOI and Oyoung LEE
Department of Electrical Engineering, Chungbuk National University, KOREA

요 약

Sparse coding은 최소한의 active한 (non-orthogonal) basis vector를 이용하여 데이터를 표시하는 하나의 방법이다. Sparse coding에서 basis coefficient들이 statistically independent하다는 constraint를 주기에 ㄴparse coding은 independent component analysis(ICA)와 밀접한 관계를 가지고 있다. 본 논문에서는 sparse representation을 위하여 super-Gaussian prior를 이용한 ICA, 즉 sparse ICA 방법을 제시한다. Sparse ICA 방법을 이용하여 natural scenes의 basis vector를 찾고 이와 sparse coding과의 관계를 고찰한다. 여러 가지 super-Gaussian prior들을 고려하고 이들이 ICA에 미치는 영향에 대해 살펴본다.

1. INTRODUCTION

A number of attempts have been made to describe how brain does information coding efficiently in early sensory processing. Early sensory coding is highly related to efficient information representation. Barlow suggested that sensory coding strategies should take advantage of the redundancy in the environment to produce more efficient representations of sensory information [2].

A compact coding scheme was believed to be a way of efficient data representation. The compact coding performs a transformation that allows the input to be represented with a reduced number of vectors with minimal reconstruction error. This approach is related to a method known as principal component analysis (PCA). The PCA aims at finding a set of mutually orthogonal basis vectors that capture the directions of maximum variance in the data and for which the basis coefficients are uncorrelated [7]. Only linear pairwise correlations are used in PCA, so higher-order statistical dependence in the data can not be captured. Thus basis vectors learned by PCA are not localized, and do not resemble cortical receptive fields.

Sparse coding [5] is a method for finding a data data representation in which only a small number of basis coefficients are active (i.e., most of basis coefficients are close to zero). It is recently shown that basis vectors that are learned by a sparse code from natural scenes, bear resemblance to the characteristics of simple cell receptive fields in mammalian striate cortex [8]. The linear sparse code can be considered as a specific form of minimum entropy code in which the probability distribution of each basis coefficient's activity is peaked around zero. In other words, in sparse code, basis coefficients that are active simultaneously are statistically independent and their probability distributions are super-Gaussian. Along this line, sparse code and independent component analysis (ICA) have similarity. Thus ICA with super-Gaussian prior (thus named as sparse ICA) also can result in sparse representation like the sparse code proposed by Olshausen and Field.

In this paper we show the basis vectors learned by the sparse ICA also leads to sparse representation. Some connections between the sparse code and ICA are investigated. Some preliminary work along this direction was done by Bell and Sejnowski [4].

2. DATA REPRESENTATION

In signal transformation or data representation, it is often useful to represent the data as a linear superposition of basis vectors. In complete representation, the n-dimensional data $\boldsymbol{x} = [x_1 \cdots x_n]^T$ is written as

$$\boldsymbol{x} = \sum_{i=0}^{n} s_i \boldsymbol{a}_i \tag{1}$$

where $\{ \boldsymbol{a}_i \}$ is a set of basis vectors that spans the n-dimensional vector space and $\{ s_i \}$ is a set of basis coefficients that represents the activity of the corresponding basis vectors.

If any two basis vectors in $\{ \boldsymbol{a}_i \}$ are orthonormal each other, basis coefficients $\{ \boldsymbol{a}_i \}$ can be easily computed by projecting the data x onto the corresponding basis vectors, i.e.,

$$s_i = \langle \boldsymbol{x}, \boldsymbol{a}_i \rangle, \tag{2}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. In fact, this is known as orthogonal transformation. Fourier transformation is one well-known example.

Data compression (compact coding) requires the data to be approximated by smaller number of basis vectors with minimal reconstruction error, i.e.,

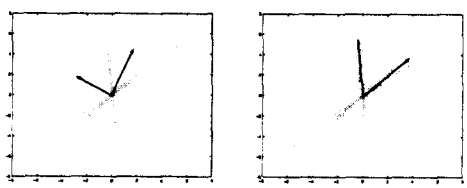
$$\hat{\boldsymbol{x}} = \sum_{i=0}^{m} s_i \boldsymbol{a}_i, \tag{3}$$

where $m < n$ and the reconstruction error $\| \boldsymbol{x} - \hat{\boldsymbol{x}} \|^2$ is minimized ($\| \cdot \|$ is Euclidean norm). Karhunen-Loeve transformation or PCA provides the optimal linear coding in mean square sense. Basis vectors learned by PCA correspond largest m normalized eigenvectors of the data covariance matrix $E\{ \boldsymbol{x}\boldsymbol{x}^T \}$ and basis coefficients become uncorrelated [7].

Unlike PCA, ICA assumes non-Gaussian structure in data and aims at finding non-orthogonal basis vectors with basis coefficients being statistically independent. Figure 1 illustrates the effectiveness of

ICA in modeling the data having the non-orthogonal density.

In sparse coding, the dimensionality of the representation is maintained or even increases, i.e., the number of basis coefficients remains roughly constant and may even increase (overcomplete representation). However, the number of basis coefficients that are responding to any particular stimulus is minimized. Over the population of likely inputs, every basis coefficient has the same probability of producing a response but probability is low for any given basis coefficient. The goal of sparse coding is to obtain a code where only a few basis coefficients respond to any given input. In addition, basis coefficients $\{s_i\}$ are required to be statistically independent, thus the resulting code is efficient in the viewpoint of Shannon's source coding theorem.



(a) (b)

Figure 1: Basis vectors learned by (a) PCA and (b) ICA

Let us assume that the number of sparse components are equal to the number of sensory signals. Although it is not a necessary requirement, this assumption make it easy for us to fit the sparse coding in the ICA framework. We consider several super-Gaussian priors for basis coefficients for sparse representation and compare them in this paper.

3. ML AND ICA

This section briefly review the maximum likelihood estimation approach to ICA. Let us define the matrix $A=[a_1,\cdots,a_n]$ and the vector $s=[s_1,\cdots,s_n]^T$. Then Eq.(1) can be written as

$$x = As \tag{4}$$

In fact the model (4) is a linear generative model in the limit of zero noise. The goal of ICA is to find both A and s given only x, under the assumption of statistical independence of $\{s_i\}$.

In the framework of latent variable model, basis coefficients $\{s_i\}$ can be viewed as latent variables that are not directly observable to us, but are observed through the data x. The matrix A represents a linear transformation from latent space to data space.

Learning basis vectors can be achieved by maximizing the probability of data given model. For a set of N independent data vectors $\{x_t\}_{t=1}^N$, the likelihood function is given by

$$\prod_{t=1}^{N} p(x_t|A). \tag{5}$$

A single factor in the likelihood function is computed by marginalizing over latent variables [6],

$$p(x|A) = \int p(x|s,A)p(s)\,ds$$

$$= \int \prod_{j=1}^{n} \delta(x_j - \sum_{i=1}^{n} A_{ji}s_i) \prod_{i=1}^{n} p_i(s_i)\,ds \tag{6}$$

$$= |\det A|^{-1} \prod_{i=1}^{n} p_i(\sum_{j=1}^{n} A_{ji}^{-1}x_j) \tag{7}$$

Then we have

$$p(x|A) = |\det A|^{-1} p(A^{-1}x) \tag{8}$$

The log likelihood is

$$\log p(x|A) = -\log|\det A| + \log p(A^{-1}x) \tag{9}$$

It is often customary to learn $W = A^{-1}$ instead of A in ICA. Let us define $y = Wx$. Then the log likelihood (9) can be rewritten as

$$\log p(x|A) = \log|\det W| + \sum_{i=1}^{n} p_i(y_i) \tag{10}$$

The learning algorithm for updating W is obtained by maximizing the log likelihood (10) with natural gradient ascent method [1],

$$\Delta W = \eta_t \{I - \phi(y)y^T\}W \tag{11}$$

where $\eta_t > 0$a learning rate and $\phi(y)$ is a element-wise nonlinear function defined by

$$\phi(y) = [\phi_1(y_1)\cdots\phi_n(y_n)]^T, \tag{12}$$

where $\phi_i(y_i)$ is the negative score function which has the form

$$\phi_i(y_i) = -\frac{d\log p_i(y_i)}{dy_i} \tag{13}$$

In the framework of maximum likelihood for ICA, latent variables $\{s_i\}$ (basis coefficients, or sometimes called sources) are treated as nuisance parameters, thus only their probability densities are required. The nonlinear functions $\{\phi_i(\cdot)\}$ depend on priors for basis coefficients. In order to get sparsely distributed representation, we consider super-Gaussian priors. The super-Gaussian density peaks at zero and has heavier tail than Gaussian, so it is proper choice for sparse coding. Bell and Sejnowski [3,4] used hyperbolic tangent function, $\phi_i(y_i) = \tanh(y_i)$ which can be derived from the density model, $p_i(y_i) = 1/\cosh(y_i)$ [6].

We also consider Laplacian prior, $p_i(y_i) = (\lambda/2)e^{-\lambda|y_i|}$. With Laplacian prior, the nonlinear function $\phi_i(y_i)$ is signum function, $\phi_i(y_i) = \mathrm{sgn}(y_i)$.

4. SIMULATIONS

We have used several natural scenes as inputs to PCA and ICA filters. Every natural scenes were converted to 8-bit gray scale images (values between 0 and 255). From gray scale images, we have generated 17160 12 × 12 image patches and have constituted 144-dimensional data vector x with 17160 samples. Then the data vector x was preprocessed by subtracting the mean and whitening.

The ICA filter matrix W was initialized as the identity matrix. We have applied both on-line ICA algorithm in Eq. (11) and its batch version. In batch ICA algorithm, the sample average of $\phi(y)y^T$ was used for updating W. For batch ICA algorithm, the learning rate η_t was set as .1 and for on-line ICA algorithm, we used $\eta_t = .001$ for first 20 sweeps and $\eta_t = .0005$ for next 30 sweeps. All the

results were similar. First 36 basis functions obtained from PCA and ICA, are shown in Figures 2, 3, and 4. It can be observed that that basis functions learned by ICA are localized and oriented, whereas PCA filters are not. Moreover, Laplacian prior gave more sparse distribution than the hyper-Cauchy prior (see Figure 5).

5. CONCLUSIONS

We have applied ICA with super-Gaussian prior in order to obtain sparse representation of natural scenes. Hyper-Cauchy prior and Laplacian prior were used and compared in terms of sparseness. Through computer simulations, we verified sparse ICA also gave localized and oriented basis functions like the ones obtained by sparse coding.

6. ACKNOWLEDGMENT

This work was supported by Braintech 21, Ministry of Science and Technology.

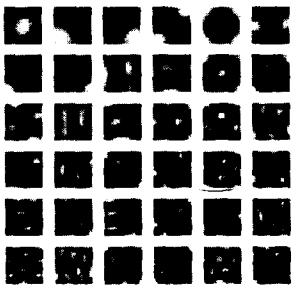


Figure 2: First 36 principal components are shown by column, then by row.

7. REFERENCES

[1] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10:251-276, 1998.

[2] H. B. Barlow. Unsupervised learning. *Neural Computation*, 1:295-311, 1989.

[3] A. Bell and T. Sejnowski. An information maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129-1159, 1995.

[4] A. Bell and T. Sejnowski. The independent components of natural scenes are edge filters. *Vision Research*, 37(23):3327-3338, 1997.

[5] D. Field. What is the goal of sensory coding? *Neural Computation*, 6(4):559-601, 1994.

[6] D. J. C. MacKay. Maximum likelihood and covariant algorithms for independent component analysis, 1996. University of Cambridge, Cavendish Laboratory, Draft 3.7.

[7] E. Oja. Neural networks, principal component analysis, and subspaces. *International Journal of Neural Systems*, 1:61-68, 1989.

[8] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607-609, 1996.

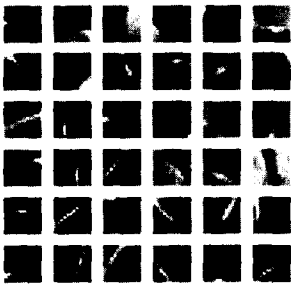


Figure 3: First 36 basis vectors learned by ICA with $\phi(y_i) = \tanh(y_i)$ are shown by column, then by row.

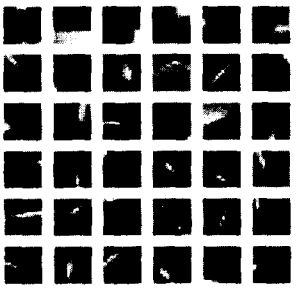


Figure 4: First 36 basis vectors learned by ICA with $\phi(y_i) = sgn(y_i)$ are shown by column, then by row.

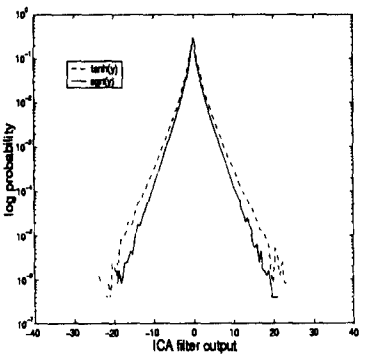


Figure 5: Lod distribution of ICA filter outputs learned by the nonlinear functions $\phi(y_i) = \tanh(y_i)$ (dotted line) and $\phi(y_i) = sgn(y_i)$ (solid line). Both are peaky distributions and solid line is more peakier(i.e., more sparse).