Independent Component Analysis for Blind Source Seperation

Bharath Masetty

bmasetty@utexas.edu — bm37498

1 Introduction

Independent component analysis (ICA) is a popular technique used for the decomposition of multivariate signals into different additive sub-components. This technique is particularly useful when we are presented in the cases which involve studying large dataset which is a result of linearly mixing information from various parallel source with unknown weights. ICA can be considered as a special case of *blind source separation* problem. A popular example is the the *"cocktail party problem"* of focusing on a single person's voice in a noisy room. For this approach to work, it is necessary that the blind sources are statistically independent non-Gaussian signals. These assumptions are the reason for the term *"independent component"* analysis. In this work, we use the ICA to separate independent sources on two datasets. First is a smaller data set with 3 sources and 40 time-steps and the second set comprises of 5 sources and 44000 time-steps.

2 Methods

Assume that we observe n linear mixtures $x_1,...,x_n$ of n independent sources. Following the statistical latent variable model, we drop the time index of the data and assume that each mixture x_j and the independent component x_k are random variables constantly sampled from a distribution (prior). We also assume that both the mixture variables and the independent sources have zero mean. This assumption can be easily satisfied by any data simply by mean normalization.

We can relate the mixed variables and source signals as follows:

$$\mathbf{X} = \mathbf{AS}$$

where A is an unknown mixing matrix. Our aim is to find a matrix **W** such that

$$\mathbf{Y} = \mathbf{WX}$$

where Y is an approximation of decomposed independent sources. Ideally, W is an inverse of the unknown matrix A.
Since we decided to drop that t in the signals and considered them as samples from unknown probability distributions, the source signals can be represented as *likelihood* of series of values.

$$\mathbf{p(s) = p_s(s_1)...p_s(s_n)}$$

similarly, the likelihood for the mixed signals is as follows:

$$\mathbf{p(x) = p_s(w_1{}^T x)...p_s(w_n{}^T x).}|W|$$

Now that we have the formulation ready, we next need to specify the density for individual p_s. For any real-valued random variable z, there exists a cumulative distribution function (cdf) whose derivative would result in the density corresponding to z. So, in order to specify a pdf for the sources S_i, we need to specify a cdf for it. We cannot choose the cdf corresponding to the Gaussian pdf since it is our assumption that out signals are non-Gaussian. As a reasonable choice, we choose the sigmoid function g(s) = 1/(1 + e^{-s}) as it slowly increases from 0 to 1.

$$\mathbf{p_s = g'(s)}$$

Now, our log likelihood is given by

$$l(W) = \sum_{i=1}^{m}(\sum_{j=1}^{n} log(g^{'}(w_j^T x^{(i)} + log|W|)$$

Upon taking the derivatives and using the fact that $\bigtriangledown|W| = |W|(W^T)^{-1}$, the update rule for the stochastic gradient ascent algorithm is as as follows:

$$W = W + \alpha(\begin{bmatrix} 1 - 2g(w_1^T x^{(i)}) \\ 1 - 2g(w_2^T x^{(i)}) \\ . \\ . \\ . \\ 1 - 2g(w_n^T x^{(i)}) \end{bmatrix} x^{iT} + (W^T)^{-1})$$

Following is the step by step implementation of **ICA Algorithm**:
1. Assume $\mathbf{X = AU}$.
2. Initialize the ($\mathbf{n}$ by $\mathbf{m}$) matrix $\mathbf{W}$ with small random values. 3. Calculate $\mathbf{Y = WX}$.
4. Calculate Z where $Z_{i,j} = 1/(1 + e^{-y_{i,j}})$ for $i \in [i, .., n]$ and $j \in [1, .., t]$(t - signal length).
5. Find $\triangle W = \eta(t * I + (1 - 2Z)Y^T W$ where η is a small learning rate.
6. Update $W = W + \triangle W$ and repeat the process from step 3 until convergence.

NOTE: we use an approximated version of the natural gradient presented above to avoid the calculation of inverse of W. However, the update equation presented in the assignment page($\triangle W = \eta(I + (1 - 2Z)Y^T W)$ is slightly erroneous from the formulation used here which initially resulted in slow convergence rate. Upon careful calculation of the approximate gradient, we would get a formulation mentioned in step 4 which enables extremely fast convergence.

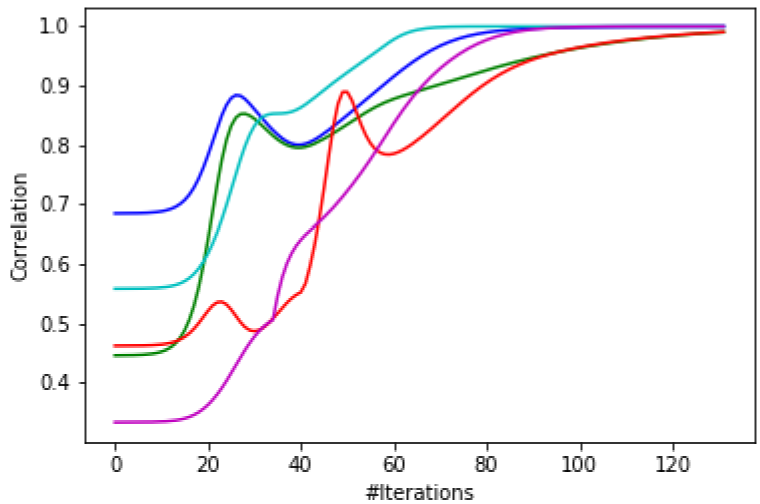


Figure 1: Variation of correlation with number of iterations for 5 sources.

3 Results

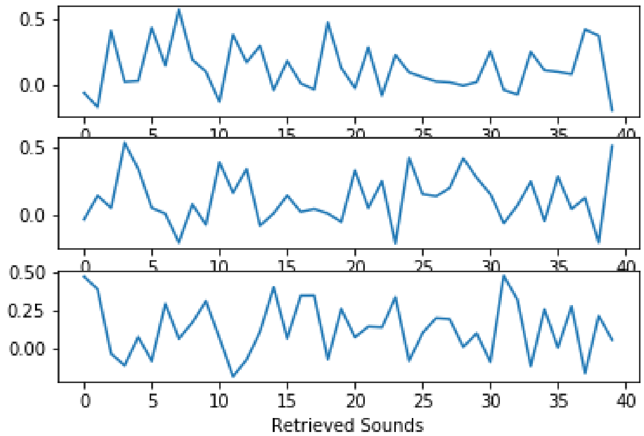
There are two parts in our experimentation. First, we test the algorithm presented above on a small test with 3 independent sources and 40 time steps. The value of learning rate used in this is $\eta = 0.01$ and number of iterations = 1000000. In this part, we used the update equation ($\triangle W = \eta(I + (1 - 2Z)Y^T W)$. The results are shown in Figure 2. Upon comparing 2(a) and 2(c), it is clear that seen that the extracted signals exactly match with the actual sound signals which tells us that the approach we use is correct. It should also be observed that the extracted signals are differ in the order and scale as compared to the original signals. This is the result of matrix transformations happening in every iteration to obtain a a set of vectors W that are perpendicular to the set of vectors A.

The second part of the experiment is to test our approach on a larger data set (5 sources and 44000 time-steps). To obtain faster convergence, the updated equation presented in step 4 of the algorithm is used in this case. These updates are orders of magnitude larger as compared to the previous case and we need to choose an extremely small learning rate to obtain convergence. The setting used for this part are $\eta = 0.000005$, max-iterations = 500 and the condition for convergence is a minimum of 0.95 correlation between extracted and original signals.

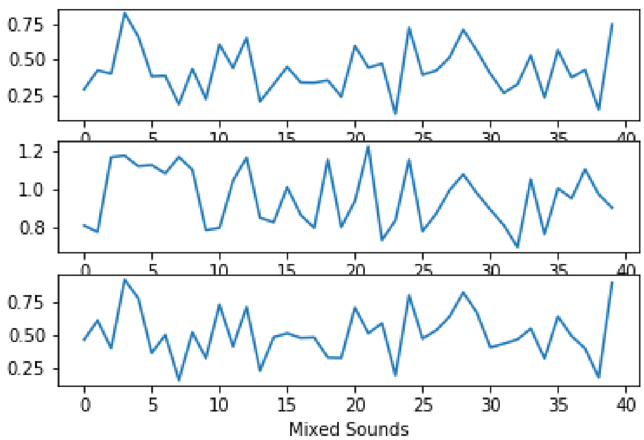
The signals for this part are presented in Figure 3 variation of correlation for each signal with number of iterations is shown in Figure 1. Upon comparing Figure 3(a) and 3(c), it is clear that the algorithm is successfully able to retrieve the independent source signals. From Figure 1, we can see that convergence is achieved on the bigger data set in only 135 iterations. This is a significant improvement (135 << 1000000 and 44000 >> 40) as compared to the configuration used for the test case in Figure 2 which almost took a million iterations on a smaller data set. This conclusively proves that the modified update equation used on the larger data set is correct and improves the performance of the algorithm significantly.

4 Summary

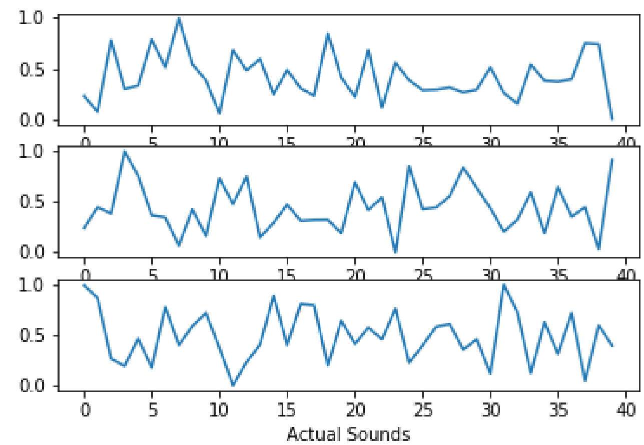
In this work, we the ICA as a special case of blind source separation method. We listed the underlying assumptions of ICA and discussed the step-by-step derivation of gradient ascent approach of ICA. We have also used a modified and correct update equation in the algorithm to appreciate the extremely fast convergence obtained on the larger data set. Overall, this work provides necessary mathematical background and physical intuition to apply ICA to any other blind source separation problem.



(a)

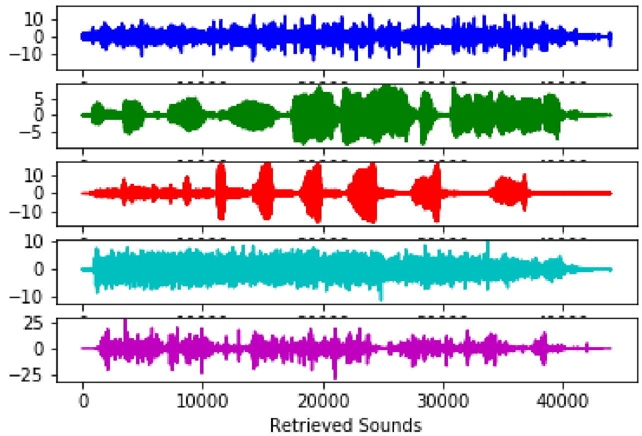


(b)

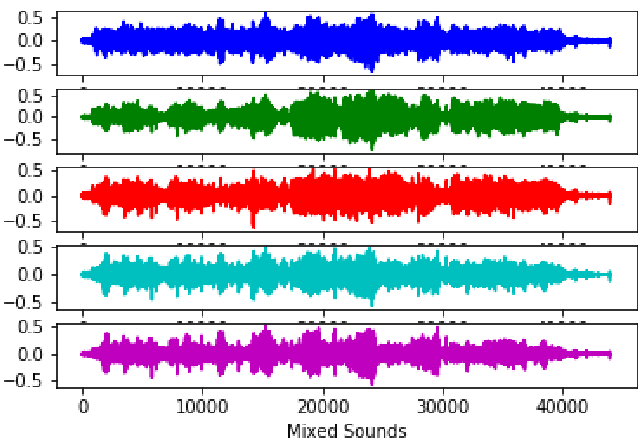


(c)

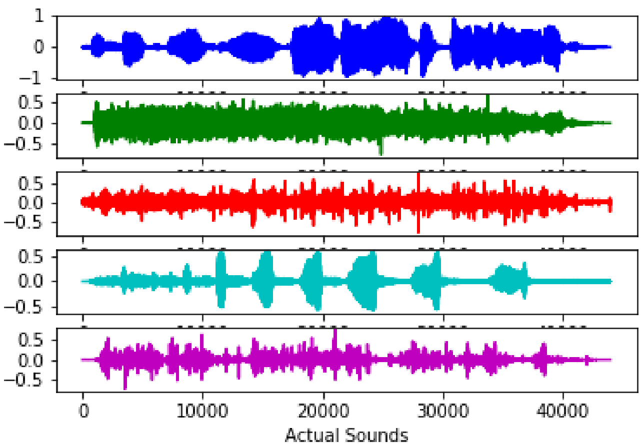
Figure 2: (a) Retrieved source signals for the test data, (b) Mixed inputs of test data, (c) Original test source signals.



(a)



(b)



(c)

Figure 3: (a) Retrieved source signals, (b) Mixed inputs, (c) Original source signals.