Homework 2: Independent Component Analysis

Thomas Wei
ThomasW219@gmail.com

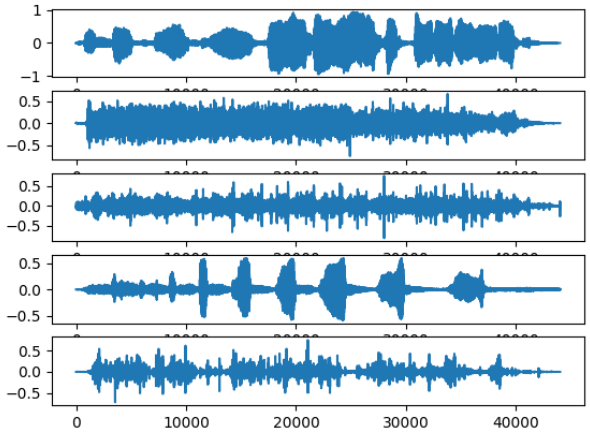
***Abstract*—Consider the "cocktail party problem" in which n people holding different conversations in a room. There are also n microphones arranged throughout the room that record each conversation at different intensities, based on the distance that the microphone is from the speaker. Independent Component Analysis (ICA) provides a way to recover the original independent conversations. Although there are multiple formulations of ICA, in this paper we will consider only the maximum likelihood version.**

I. Introduction

Consider the "cocktail party problem" in which n people holding different conversations in a room. There are also n microphones arranged throughout the room that record each conversation at different intensities, based on the distance that the microphone is from the speaker. Independent Component Analysis (ICA) provides a way to recover the original independent conversations. Although there are multiple formulations of ICA, in this paper we will consider only the maximum likelihood version.

We implement the maximum likelihood version of ICA and test its effectiveness with mixtures of 4 second sound bytes. We assumed the distributions of the independent components were distributed according to a sigmoid cumulative distribution function (cdf) and used gradient ascent to maximize likelihood. We were able to recover the original sources from mixtures of 3 and 4 sounds reliably and had limited success with mixtures of 5 sounds as well.

Fig. 1. The independent signals that are mixed, stored in a 5 by 44000 matrix



II. Method

Let $S \in \mathbb{R}^{n\times t}$ be a matrix made up of n independent sources sampled t times. Then let $X = AS$ be a linear mixture of the independent sources described by the matrix $A \in \mathbb{R}^{n\times n}$. The objective of ICA is to find a matrix W that allows us to unmix the mixed signals, i.e. $S = WX$. If we knew A beforehand, we could simply use $W = A^{-1}$ which would give us $WX = A^{-1}AS = S$ and allow us to recover the original signals exactly. In practice, A is not known beforehand and we can only evaluate the effectiveness of our matrix W by how well the data fits our independent probability distribution function (pdf) given by assumed priors for the signals and W. This means any scalar multiple of the signals may be recovered and the signals may be in any order. Without knowledge of A, any scaled signal and all permutations are equally valid as independent sources. Equivalently, the product WA need only be the product of some diagonal matrix (scaling each signal individually) and some permutation matrix (for any reordering).

Due to the rotational symmetry of Gaussian vectors made up of independent Gaussian random variables with unit variance, it is impossible to recover the n original independent sources. Therefore, we must use a non-Gaussian prior for the distribution of our signals. If we knew their distributions beforehand, the performance of ICA would likely benefit but since we don't we will use a sigmoid cdf for all our signals as it has been shown to work in practice. We will denote the sigmoid function $g(x)$.

If W is the transformation that recovers independent signals $\mathbf{s} = (s_1, \ldots, s_n)^T$ from vector $\mathbf{x} = (x_1, \ldots, x_n)^T$ then:

$$p_{\mathbf{x}}(\mathbf{x}) = p_{\mathbf{s}}(W\mathbf{x})|W|$$

Furthermore, since s_i are mutually independent:

$$p_{\mathbf{s}}(W\mathbf{x})|W| = |W|\prod_{i=1}^{n} p_{s_i}(w_i^T x) = |W|\prod_{i=1}^{n} g(w_i^T x)$$

Where w_i^T are the rows of W. If we also assume that samples of X are independent, we can write the likelihood of W being the transformation that recovers independent components (given data $X \in \mathbb{R}^{n\times t}$) as:

$$L(W; X) = \prod_{i=1}^{t}\left(|W|\prod_{j=1}^{n} g(w_j^T x_i)\right) \tag{1}$$

Taking the log of both sides we get

$$\log(L(W; X)) = \sum_{i=1}^{t}\left(\log(|W|) + \sum_{j=1}^{n}\log(g(w_j^T x_i))\right) \tag{2}$$

By taking the gradient with respect to W for a particular sample x_i we can see that:

$$\nabla_W \log(L(W; x_i)) \propto \begin{bmatrix} 1 - 2g(w_1^T x_i) \\ \vdots \\ 1 - 2g(w_n^T x_i) \end{bmatrix} x_i^T + (W^T)^{-1} \quad (3)$$

If we wanted to use more samples to get a more reliable sample expectation for the gradient at that particular W we could use the fact that a sum of outer products is equivalent to matrix multiplication of the appropriate matrices. Let the matrix $X \in \mathbb{R}^{n \times k}$ be the k compiled samples, then we get the expression:

$$\frac{1}{k} \begin{bmatrix} 1 - 2g(w_1^T x_1) & \dots & 1 - 2g(w_1^T x_k) \\ \vdots & & \vdots \\ 1 - 2g(w_n^T x_i) & \dots & 1 - 2g(w_n^T x_k) \end{bmatrix} X^T + (W^T)^{-1} \quad (4)$$

which converges to the expected value of the gradient as k gets large.

Since calculating inverses of matrices is computationally expensive, we want to achieve the same effect using an expression without an inverse. To do so we instead consider the natural gradient which is simply the gradient right multiplied by $W^T W$. To simply notation, we will let $Y = WX$ which leads to $y_{i,j} = w_i^T x_j$. Using the appropriate substitutions, we get the natural gradient to be:

$$\left(\frac{1}{k} \begin{bmatrix} 1 - 2g(y_{1,1}) & \dots & 1 - 2g(y_{1,k}) \\ \vdots & & \vdots \\ 1 - 2g(y_{n,1}) & \dots & 1 - 2g(y_{n,k}) \end{bmatrix} Y^T + I \right) W \quad (5)$$

We can use any of the above expressions for the gradient to perform gradient ascent to maximize our likelihood function in terms of W. We use the rule:

$$W_{i+1} = W_i + \eta \nabla_W log(L(W_i; X)) \quad (6)$$

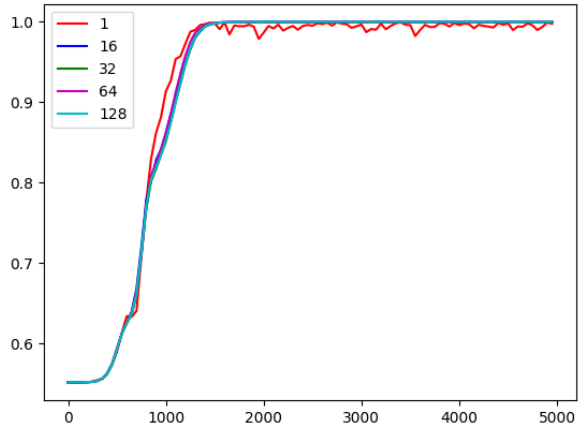
Where optionally the natural gradient can be substituted in for the normal gradient. Repeating this process with an appropriate value of η should allow W_i to converge to a matrix that recovers the independent components.

III. Results

To measure the effectiveness of our implementation of ICA in recovering the original signal, we calculated the correlation between the source signals and the recovered signals and performed a matching so that the highest average correlation would be achieved but every signal would be matched with another. We used random mixes of the source signals and initialized W_0 to be random matrices with entries in $[0, 0.1)$.

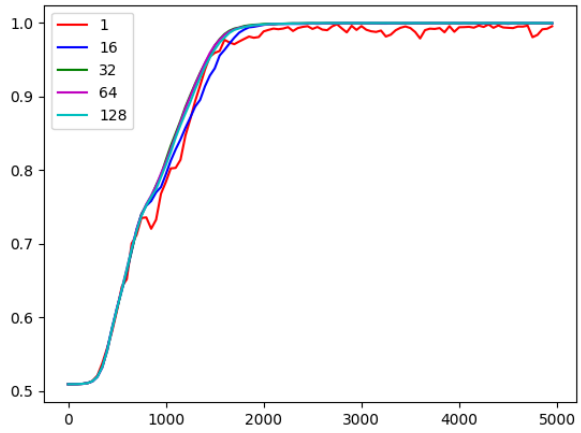
We evaluated our ICA implementation on mixes of 3, 4, and 5 signals. For each mix of signals, we performed ICA with different batch sizes for each step. Starting from using one sample to update W_i in each iteration (stochastic gradient ascent) and then increasing to use 16, 32, 64, and 128 samples per iteration. These samples were chosen randomly from X at each iteration.

Fig. 2. Correlation over iterations for the mix of 3 signals, different batch sizes are shown



In figure 2 we can see that the larger batch size experiments had smoother curves as they optimized for maximum likelihood. In this case stochastic gradient ascent (batch size 1) exceeded the largest batch size for a considerable number of iterations, however, had the results for stochastic gradient ascent been averaged over many different runs, it would likely look similar to the larger batch size executions.

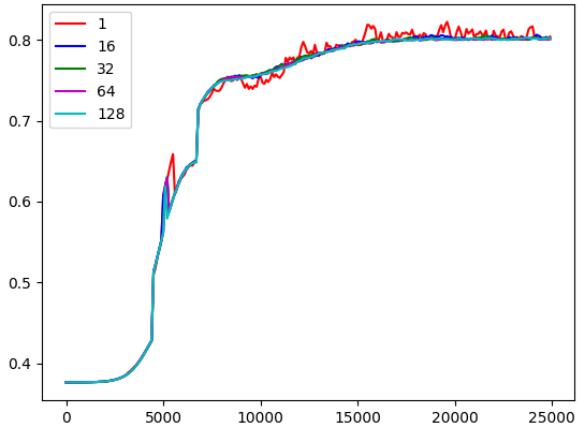
Fig. 3. Correlation over iterations for the mix of 4 signals, different batch sizes are shown



In figure 3 we see similar behavior to the mix of 3 signals in figure 2. Although it took longer to converge, the correlations eventually did reach 1. In both experiments in figures 2 and 3 we ran gradient ascent for 5,000 steps with a learning rate (η) of 0.01.

When mixing 5 signals in figure 4, we saw a more chaotic rise in correlation that did not improve past 0.8. Even with

Fig. 4. Correlation over iterations for the mix of 5 signals, different batch sizes are shown



25,000 iterations of gradient ascent and a learning rate reduced to 0.001 to reduce divergence issues, our implementation of ICA still could not match its performance on mixes of fewer sources. It is possible that more iterations and an even smaller learning rate could yield better results given more time, however, we didn't have the time to try.

A visualization of our results for our mix of 3 sounds in figure 2 is shown in figure 5. As shown in the figure, we were able to recover scaled and permuted versions of the original signals.

IV. Summary

ICA allows us to recover the n independent sources of information as long as they do not have a Gaussian distribution and we have at least n mixed sources to recover them from. The recovered signals may be a scalar multiple of the original and the sources may be permuted but for many applications, such as audio separation, these variations from the original are not an issue.

Fig. 5. The top three signals are the unmixed and independent signals, the three in the middle are the mixed signals, the last three are the recovered signals