Alex Navarro ajn853 Machine Learning Assignment 2

**Independent Component Analysis**

Introduction

It is often difficult, and usually impossible, for a pure signal to be obtained. Especially for ambient signals, such as sound, there is usually a combination of signals obtained by any measurement device. It is desirable to extract these individual sounds back from the mixed signal so they can be processed in isolation. The process by which this is done is called independent component analysis, or ICA.

The crucial component of ICA, as inferred from the name, is that the combined signals are independent of each other. This assumption that there is no dependency relation between individual components allows for the isolation of the signals. The method to be used a gradient descent optimization of a general cost function for signal separation.

Method

Assume that initially we have *n* individual source signals, each of length *t*. We then stack them in matrix form as the source signal matrix **U**

These signals can then be combined linearly to form a new set of *m* mixed signals. The individual weights of each signal are not known, and indeed it is the aim of this algorithm to determine these weights. The mixed signal matrix is called **X** and is related to the source signal matrix through a linear transformation that we will call **A**.

It follows then that if we want to determine what the original signal matrix **U** is, that we simply need to find a matrix **W** such that . We will assume that **A** is square.

The algorithm is as follows:

* We start with an initial guess of the **W** matrix. Since we do not know anything about our signals beforehand, we initialize matrix **W** with small random values.
* We calculate the current estimate of the source signal matrix, which we will call , where .
* Using the derivative of the log-likelihood function for the matrix **W**, we calculate the size and direction of the next step in the gradient descent.
* Here is the approximate cdf of the individual distributions, and is the learning rate. Smaller learning rates will make the algorithm take longer, while larger learning rates will make convergence unstable
* **W** is updated at each iteration as until the ratio is sufficiently small, or the maximum number of iterations is reached

The Data

Through the homework assignment link, a .mat file with 5 sounds was obtained. These sounds were, in order:

1. A Simpsons line stating, “In this house, we obey the laws of thermodynamics!”
2. A blender blending something
3. A round of applause with some background laughter
4. Someone laughing like a maniac
5. A wrapper or ball of paper being crinkled

These files were mixed with a randomly generated **A** matrix at runtime. For most tests, only 3 of the sounds were mixed at a time.

Additionally, the homework link provided a small test data set with 3 signals of length 40 samples. This was used to determine whether or not the algorithm was working on a fairly simple dataset.  
  
Results

The first test was to see if the small sample dataset could be accurately recovered, given the provided **U** and **A** matrices. After passing it through the ICA algorithm with a learning rate of 0.01 and an iteration count of 1,000,000 (the same as the example) the results were as follows:

Chart, line chart

Description automatically generated

We can see that the results are very good. . As is evident, the order of the signals does not necessarily match the order of the original signals that were mixed in the first place. This is because the ICA algorithm cannot distinguish between permutations of signals in which the rows swap places. They both provide the exact same mathematical representation.

However, a count of 1,000,000 iterations is quite a lot, and even with this small sample set took over a minute to run. What were to happen if instead we used a termination condition as described in the Method section? The following plot shows the results where the algorithm was halted when the condition is met.

Chart, line chart

Description automatically generated

This run terminated in 878 iterations. It is still easy to identify which output signal corresponds with which input signal, but the quality of the isolation has dropped. If we instead make the termination condition stricter, say having , then we do indeed get a better result, but with more iterations (25,558 in this case).

Chart, line chart

Description automatically generated

Testing With Real Data

It is important to remember in this section that since the initialization of the **A** and **W** matrices are random, the results of these experiments are stochastic, and success of the algorithm varies from run to run. The termination condition was tightened by one more order of magnitude to account for the complexity of the problem to . Additionally, the correlation coefficient between signals has been added as a performance metric.

Here is a successful run from the combination of sounds 2, 3, and 4. These sounds seemed to be the easiest to separate in initial testing.

Timeline

Description automatically generated

Colour has been added to help correlate which result corresponds with which source signal. This is done by associating the colours with largest correlation coefficient, meaning that multiple result signals can be assigned the same source signal if signal isolation failed.

Here the correlation coefficients are all greater than 0.95, meaning that the ICA algorithm worked extremely well. Listening to the resulting sound audio, I can say that the sounds sounded almost exactly like their sources, with the first one having only a slight hint of another sound in the background. The algorithm converged in only 1,011 iterations.

However, not all runs went as well as this one. Running the algorithm again with the exact same settings gave the following result. The only difference was the randomly generated **A** and **W** matrices, yet the algorithm was unable to properly separate the signals into their components.

Timeline

Description automatically generated with medium confidence

This run still converged in around 1000 iterations, but the output sounds were still heavy mixes of the source signals. The ICA algorithm reached a local minimum and was unable to continue to the true solution. Two of the output solutions matched input sound 2, but none of them matched input sound 3.

Other sound combinations were more difficult to separate. Sound 1 (the Homer Simpson voice) in particular had a habit of remaining in the background of all recovered signals, although there were occasionally some good instances where it could be filtered completely. Additionally, the sound of the blender (sound 2) and the sound of crinkling paper (sound 5) were often very difficult to separate. This is likely because they both have a “white noise” type of effect, and so assumption of independence may not hold in that case since they are so similar.  
  
Performance

The main tunable parameter for the ICA algorithm is the step size. As in any optimization problem, the step size can have a large effect on both the convergence rate and the solution accuracy. To test the effect of step size, the algorithm was run with the aforementioned settings (using sounds 2, 3, and 4 with termination condition ). The step size was varied from 0.01 to 0.10 in increments of 0.01.

Chart, line chart

Description automatically generated

As expected, the iterations needed to converge decreased significantly with increased step size. Increasing the step size by a factor of 10 also reduced the number of iterations by a factor of 10. But what effect does this increased efficiency have on the efficacy of the algorithm?

Chart, line chart

Description automatically generated

There is no clear trend here. As the step size increased, the correlation results decreased until suddenly spiking up to around 0.98 at a step size of 0.08. This can be attributed to the stochastic nature of the results. After running it a few more times, the mean R value for a step size of 0.08 was in the range of 0.7-0.8. Sometimes you just get lucky or unlucky with the initial seed. Smaller step sizes were more consistent at successfully isolating the source signals. Determining if an optimal seed exists, and what it is, is interesting but beyond the scope of this assignment.