THE EM ALGORITHM IN INDEPENDENT COMPONENT ANALYSIS

Kaare Brandt Petersen, Ole Winther

ISP, IMM, Technical University of Denmark, Building 321, DK-2800 Kgs. Lyngby, Denmark
kbp@imm.dtu.dk, owi@imm.dtu.dk

ABSTRACT

We investigate two techniques for Independent Component Analysis which use the Expectation-Maximization algorithm. Analysis and simulations show that convergence becomes extraordinary slow for almost all cases, compared to other optimization techniques. The two alternatives considered are "Adaptive Overrelaxed EM" and Ucminf (a BFGS with soft line search), which both improves the convergence dramatically with little or no extra analytical work. We discuss the generality and perspectives of the findings.

1. INTRODUCTION

The EM algorithm, as formulated by Dempster et al. in 1977 [1], has won enormous attention and widespread use as a maximum likelihood estimator in situations of incomplete data. This is presumably due to its guaranteed increase in the likelihood and the computationally appealing framework. The EM algorithm has been applied to a vast range of problems (See e.g. [2] for examples), a clear evidence of the general success of the algorithm, but there has also been many reports on poor convergence properties and attempts to deal with them .

Among the many proposals for accelerating the EM algorithm are the Aitken accelerator [3], different quasi-Newton approaches and the Conjugated Gradient acceleration [4]. As documented in [5], the accelerated methods can improve the performance of the EM algorithm by a factor 10 or more, but common for all the approaches above is, that they demand more analytical computations, which in some cases can be very troublesome. Recently, in 2003 Salakhutdinov et al. [6] proposed an extremely simple method called "Adaptive Overrelaxed EM" , which demands no more analytical work than the basic EM algorithm but is much faster.

In this paper we demonstrate, as also reported in [7] for the low noise limit, that when applying the EM algorithm to Independent Component Analysis (ICA), the convergence properties are so poor that it almost renders the approaches useless in praxis. We also show, however, that using a quasi Newton approach or the simple "Overrelaxed Adaptive EM" the performance is dramatically improved, thus again making the ICA techniques under consideration relevant as practical algorithms.

2. INDEPENDENT COMPONENT ANALYSIS

The two ICA methods we use in this paper, are the Mean Field ICA (MFICA) presented in [8] and the Independent Factor Analysis (IFA) presented in [9].

We assume the observation model $\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \boldsymbol{\eta}_t$ for all time steps t, and collect these vectors into the matrices $\mathbf{X}$ and $\mathbf{S}$. The job of the algorithms is to estimate the mixing matrix $\mathbf{A}$ knowing only the observed signals $\mathbf{X}$. The sources $\mathbf{s}_t$ are assumed statistically independent in the coordinates and time and the noise vector $\boldsymbol{\eta}_t$ is assumed white, zero-mean gaussian with known covariance. The negative log-likelihood $F(\mathbf{A}) = -\ln p(\mathbf{X}|\mathbf{A})$ can be bounded from above

$$F(\mathbf{A}) \leq \int q(\mathbf{S}|\mathbf{X}, \tilde{\mathbf{A}}) \ln \frac{q(\mathbf{S}|\mathbf{X}, \tilde{\mathbf{A}})}{p(\mathbf{S}, \mathbf{X}|\mathbf{A})} d\mathbf{S} \equiv F_v(\tilde{\mathbf{A}}, \mathbf{A})$$

for any distribution $q(\mathbf{S}|\mathbf{X}, \tilde{\mathbf{A}})$. Using that F_v is an upper bound on the negative log-likelihood, F, and that $F = F_v$ when $\mathbf{A} = \tilde{\mathbf{A}}$ and q is the source posterior $p(\mathbf{S}|\mathbf{X}, \mathbf{A})$, we choose F_v to be the cost function. We now use the EM algorithm to minimize F_v:

- *E-step:* Compute $F_v(\mathbf{A}^n, \mathbf{A})$, using the newly updated matrix $\mathbf{A}^n$.
- *M-step:* Set $\mathbf{A}^{n+1} = \text{argmin}_{\mathbf{A}} F_v(\mathbf{A}^n, \mathbf{A})$.

The derivative of F_v with respect to $\mathbf{A}$ is $\mathbf{W}^{-1}\mathbf{A}\mathbf{G} - \mathbf{W}^{-1}\mathbf{X}\mathbf{m}^T$, where $\mathbf{W}$ is the noise covariance, $\mathbf{m}$ is the mean values $\langle \mathbf{S} \rangle_q$ and $\mathbf{G}$ is the matrix $\langle \mathbf{S}\mathbf{S}^T \rangle_q$. In the M-step, this results in an update equation $\mathbf{A}^{n+1} = \mathbf{X}\mathbf{m}^T\mathbf{G}^{-1}$. As we shall utilize later, this shows that methods different from EM, which uses the derivative of the cost function, can be implemented with very little extra computational effort.

So far the two approaches are completely similar. The difference between IFA and MFICA is in the choice of distribution $q(\mathbf{S}|\mathbf{X}, \mathbf{A})$:

- **MFICA:** $q(\mathbf{S}|\mathbf{X}, \tilde{\mathbf{A}})$ is chosen to be a factorized distribution which in each step is fitted to the source posterior using the Kullback-Leibler divergence.
- **IFA:** $q(\mathbf{S}|\mathbf{X}, \tilde{\mathbf{A}})$ is chosen to be the source posterior which is a mixture of gaussians (MoG) when the priors are MoG's and the noise is gaussian also.

In IFA it is possible to compute the integral in F_v exactly thanks to the suitable choice of the priors. In MFICA, the priors are not restricted, but the integral is approximated using mean field theory. Thus, IFA is a more straight forward but restricted model while MFICA is a good approximation for a wide range of different priors. Note that the MFICA approximation becomes exact, when $\tilde{\mathbf{A}}$ becomes orthogonal and therefore IFA and MFICA are exactly equivalent in the special case of MoG priors and orthogonal mixing matrix.

Thus, in MFICA in general, the entities $\mathbf{m}$ and $\mathbf{G}$ are approximated rather than exactly determined. That makes the minimization in the M-step an approximation, and one could argue that MFICA is not using a true EM update. In that case, the slowdown demonstrated in MFICA could be an effect of the approximation in stead of the EM scheme. To counter this argument, we

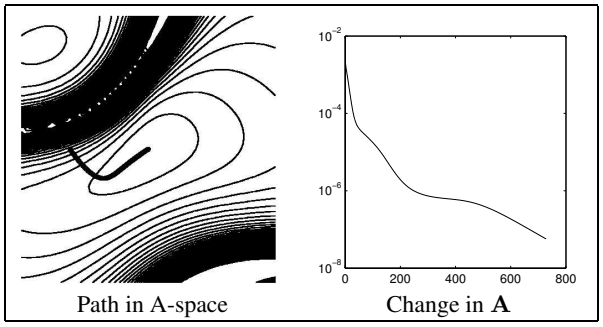

Fig. 1. MFICA: An example of EM slowdown. Left: The (projected) path of EM in A-space. Right: The change of $\mathbf{A}$ over iterations.

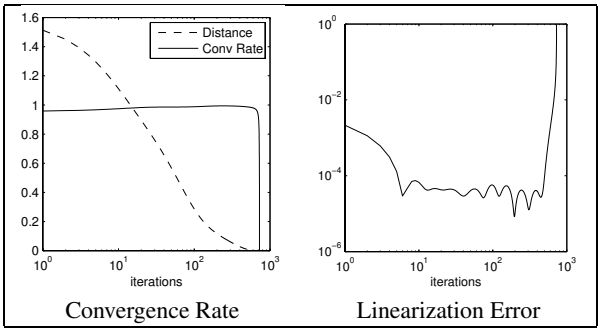

Fig. 2. MFICA: An example of EM slowdown. Left: The convergence rate. Right: The error induced when applying the approximation of Eq. 1

.

have included the IFA, which uses a true, classical EM update, and demonstrates that the slowdowns in MFICA are also present in the IFA and thus a result of the properties of the EM algorithm.

For the rest of this paper we focus on the slowdown of EM in MFICA, using similar results for IFA as a way to ensure that the conclusions on EM in MFICA is not due to the approximations done.

3. THE PROBLEM USING EM

When implementing MFICA and IFA we find from practical experience that they both suffer from very slow convergence. I order to investigate the reason for this we have taken a typical example and done an analytical and numerical investigation presented in this and next section.

The slow convergence in MFICA is evident in Fig. 1: The dots of the iterations on the contour plot (left) has turned into a solid line and the changes in $\mathbf{A}$ to the right are smaller than 10^{-6} for more than 400 iterations which, compared to other optimization techniques in the same setting, is extraordinarily slow.

In the following, $\mathbf{a}^n$ is the vector-form of $\mathbf{A}^n$, that is, $\mathbf{a}^n = \text{vec}(\mathbf{A}^n)$. In each step, $\mathbf{a}^n$ is updated to $\mathbf{a}^{n+1}$ and we can write this as a mapping $\mathbf{a}^{n+1} = \mathbf{M}(\mathbf{a}^n)$. Close to the optimal $\mathbf{a}^*$, we can make a first order Taylor-expansion of this mapping and obtain

$$\mathbf{a}^{n+1} \cong \mathbf{a}^* + \mathbf{M}'_*(\mathbf{a}^n - \mathbf{a}^*) \tag{1}$$

If $(\mathbf{a}^n - \mathbf{a}^*)$ is an eigenvector to the matrix $\mathbf{M}'_*$ with eigenvalue 1, the algorithm gets stuck *even* when there is still a large difference between $\mathbf{a}^n$ and the optimal $\mathbf{a}^*$. We can find the matrix $\mathbf{M}'_*$ from the matrix formulation by using $\mathbf{A}^{n+1} = \mathbf{X}\mathbf{m}^T\mathbf{G}^{-1}$ where $\mathbf{m}$ and $\mathbf{G}$ are functions of $\mathbf{A}^n$. We can therefore differentiate $\mathbf{A}^{n+1}$ with respect to $\mathbf{A}^n$

$$\frac{\partial \mathbf{A}^{n+1}}{\partial A^n_{ij}} = \mathbf{X}\Big(\frac{d\mathbf{m}^T}{dA^n_{ij}}\mathbf{G}^{-1} + \mathbf{m}^T\frac{d\mathbf{G}^{-1}}{dA^n_{ij}}\Big)$$

in which all entities can be computed with some effort. From this expression we can determine $\mathbf{M}'_*$ by rearranging the indices suitably and inserting $\mathbf{a}^*$ (found numerically from the algorithm). One can also obtain an approximation of $\mathbf{M}'_*$ from the vectors $\mathbf{a}^n$ by estimating the least square solution to Eq. 1, that is, defining $\boldsymbol{\alpha}_0 = [\mathbf{a}^{t'-1}...\mathbf{a}^{t-1}]$ and $\boldsymbol{\alpha}_1 = [\mathbf{a}^{t'}...\mathbf{a}^{t}]$, for some suitable choice of t and t', then $\mathbf{M}'_* \cong \boldsymbol{\alpha}_1\boldsymbol{\alpha}_0^T(\boldsymbol{\alpha}_0\boldsymbol{\alpha}_0^T)^{-1}$. Knowing $\mathbf{M}'_*$ we are able to understand the convergence properties of EM at least in the neighborhood of $\mathbf{a}^*$. In that sense, $\mathbf{M}'_*$ is one measure of convergence among several others.

Another more direct measure of convergence often found in the literature [10, 2], is the rate of convergence r, which is defined by the equation

$$r = \lim_{n\to\infty} \frac{||\mathbf{a}^{n+1} - \mathbf{a}^*||}{||\mathbf{a}^n - \mathbf{a}^*||}$$

Under certain regularity conditions, we have further that $r \cong \lambda_{max}$ where λ_{max} is the largest of the eigenvalues $\lambda_i = \text{eig}(\mathbf{M}'_*)$. Since the iterations in practice end at some point, we often investigate the convergence rate of time n, denoted r_n, in stead of r.

Using the methods presented above, we can find the matrix $\mathbf{M}'_*$ and the convergence rate r_n for the problematic example presented in Fig. 1.

Since the example has very slow convergence, it is not surprising to find that the convergence rate r_n is close to 1 for a large part of the iterations. This is plotted in Fig 2 (left) together with the distance $d_n = ||\mathbf{a}^n - \mathbf{a}^*||$. Note on this plot that the rate of convergence is not changing notably as the distance to the optimal point changes. That is, the slowdown is not an effect which sets in close to the optimal point, but rather during the entire optimization.

The matrix $\mathbf{M}'_*$ of the example is found to be $\mathbf{M}'_* \cong \mathbf{I}$ and Eq. 1 a good approximation. This comes as no surprise close to the optimal solution, but as Fig. 2 (right) demonstrates, Eq. 1 is a good approximation far from the optimal solution. Fig. 2 (right) shows the linearization error defined as

$$\epsilon_n = \frac{\sqrt{||(\mathbf{a}^{n+1} - \mathbf{a}^*) - \mathbf{M}'_*(\mathbf{a}^n - \mathbf{a}^*)||^2}}{\sqrt{||(\mathbf{a}^n - \mathbf{a}^*)||^2}}$$

and, rather surprisingly, the errors are small in the beginning far from the optimum. The errors in the end of the iterations blow up due to a combination of numerical round off errors and very small distances to the optimum.

The general picture from the example is that EM is slow because it already from a very early stage is essentially a linear update with $\mathbf{M}'_* \cong \mathbf{I}$. Note that this was not obviously the case: for $\mathbf{a}^n$ far from the optimal point, the Taylor expansion in Eq. 1 could be wrong due to the influence of higher order terms, which would make the EM-update non-linear.

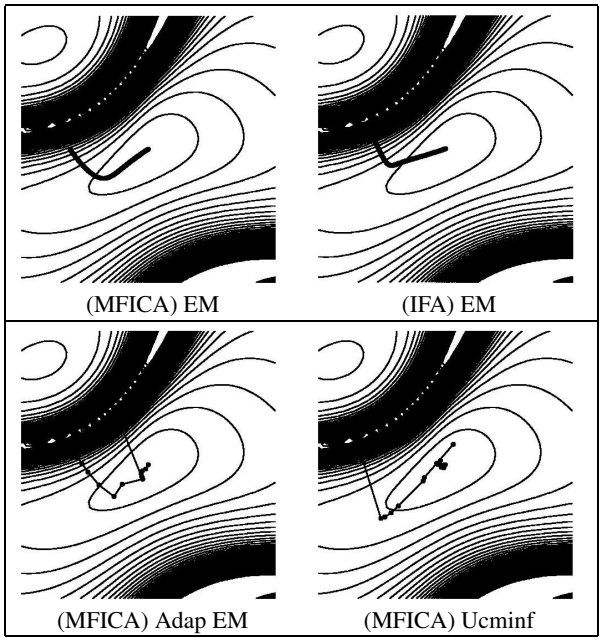

Fig. 3. The (projected) path in A-space for different optimization techniques on the example of Sec. 3.

4. OPTIMIZATION ALTERNATIVES

In this section we present two alternatives to EM which are applicable to the MFICA (and in fact also IFA). The first is applicable to any EM update, while the second demands, that it is possible to compute the gradient of the cost function.

Adaptive Overrelaxed EM: The Adaptive Overrelaxed EM (AdapEM), is a method presented in [6]: In each step, the direction of the EM is enhanced by a step size η

$$\mathbf{A}^{n+1} = \mathbf{A}^n + \eta(\mathbf{A}_{EM}^{n+1} - \mathbf{A}^n)$$

which is increased by a factor until the normally decreasing cost function is suddenly increasing. When this happens, the step size is reset to 1 and the algorithm take a step back from the value causing the increase of the cost function - recognizing that this was "a mistake".

Ucminf: The algorithm Ucminf is sometimes called the "Hans Bruun Minimizer" named after the person who combined the different parts into one algorithm with very good performance. It is a Quasi-Newton algorithm with BFGS update of the inverse hessian and soft line search (see [11] for details). Ucminf is using the E-step to calculate both $F_v(\tilde{\mathbf{A}}, \mathbf{A})$ and the gradient with respect to $\mathbf{A}$ and is then substituting the M-step with a more advanced minimization. Not surprisingly more advanced methods provide faster convergence - the role of the Ucminf algorithm in this setting is twofold: 1) to demonstrate that with modest extra work on the analytical part one can obtain very strong improvements and 2) to provide an upper limit of how fast one at best can expect EM and AdapEM to perform.

5. RESULTS

We compare the EM with the AdapEM and the Ucminf. First we let them solve the analyzed example of Sec 3, and second we randomly generate many mixing matrices and corresponding data sets (300 in the 2x2 case, and 130 in the 3x3 case), and see how they perform on these. The performance is measured in iterations and not floating point operations for convenience since the extra operations per iteration used in AdapEM and Ucminf are very few compared to the total number of operations.

For both the example and the randomly generated data sets, the prior is a MoG's where each coordinate prior is a sum of centered gaussians with variance 1 and 0.01, i.e. a sparse prior. The noise variance is $\sigma^2 = 0.01$ and the number of time steps is 500. When the relative change in the cost function $||F_v^n - F_v^{n-1}||/||F_v^n||$ becomes less than 10^{-5} the algorithms terminate. The random mixing matrices are drawn from a centered univariate normal distribution.

In Fig. 3, we see the (projected) path of the methods and contours of the cost function. The EM algorithm takes so many steps that the dots are have formed a solid line, while the other methods are doing significantly better. The result in numbers is presented in Table 1 a), which very clearly demonstrates the short comings of EM: Compared to AdapEM, standard EM in MFICA is using 45 times more iterations. The fact that IFA with 645 iterations is also performing very poorly, indicates that the slow convergence is a property of the EM algorithm and not artifacts of MFICA. Surprisingly, the AdapEM performs better than Ucminf in this particular example, but as we shall se below, the Ucminf algorithm has a better over all performance.

For the random data sets, we first draw a random generative mixing matrix, the corresponding data set and then a random initial guess on the matrix. Each of the four algorithms are then applied to the same data sets using the same initial condition, such that the basis for comparing them is as fair as possible. With respect to the results, we have grouped them into 3 bins according to the number of iterations the algorithm used to converge.

The performance for 2x2 mixing matrices is presented in Table 1 b): Using standard EM, both IFA and MFICA are most of the time using more than 100 iterations, while Ucminf never uses more than 50. An interesting point is that while AdapEM computationally is almost as easy as standard EM, it performs much better with only 17% of the data sets demanding more than 100 iterations.

The performance on 3x3 mixing matrices is presented in Table 1 c) and the picture is much the same. All techniques are using more iterations than before emphasizing that the problem in the higher dimensionality is simply more difficult, but what this table does not show is how much more than 100 iterations EM usually needs to converge. Practical experience indicates that the relative difference in iterations needed for convergence is increasing with the dimensionality, i.e., the performance of EM compared to other methods is getting worse when the mixing matrix becomes larger.

In Fig. 4 we have plotted the determinant of the 2x2 mixing matrix versus number of iterations for (MFICA) EM. From this plot we, somewhat surprisingly see that when the determinant of $\mathbf{A}$ is large, the EM is using more than 100 iterations. Close inspection reveals that determinants smaller than 0.1 also makes the number of iterations large. This indicates that (MFICA) EM only performs well in a certain interval regarding the determinant of the mixing matrix.

		Iterations
IFA	EM	645
MFICA	EM	729
MFICA	Adap EM	16
MFICA	Bruun	25

a) The example of Sec. 3.

	Iteration bins	*0-50*	*50-100*	*100-∞*
IFA	EM	7 %	10 %	**83%**
MFICA	EM	1%	6%	**93%**
MFICA	AdapEM	**51%**	32%	17%
MFICA	Ucminf	**100%**	0%	0%

b) When $\mathbf{A}$ is 2x2.

	Iteration bins	*0-50*	*50-100*	*100-∞*
IFA	EM	5 %	6 %	**89%**
MFICA	EM	0%	2%	**98%**
MFICA	AdapEM	14 %	**47%**	39%
MFICA	Ucminf	**93%**	7%	0%

c) When $\mathbf{A}$ is 3x3.

Table 1. Performance of the different approaches for the example and general 2x2 and 3x3 mixing matrices.

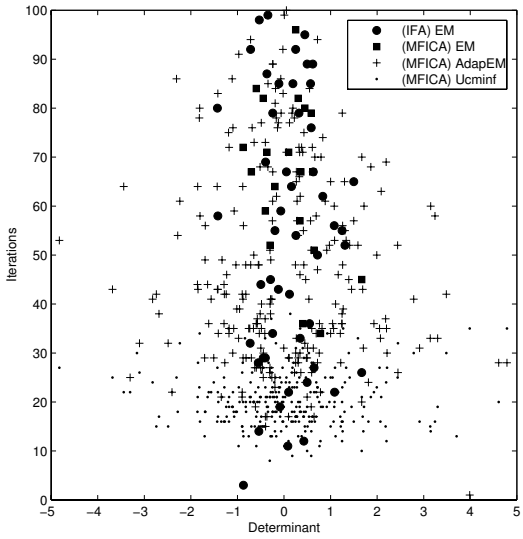

Fig. 4. The influence on the iterations of the determinant for 2x2 mixing matrices. On the first axis is $\det(\mathbf{A})$ and on the second axis is number of iterations for EM (MFICA).

6. CONCLUSIONS AND OUTLOOK

The EM algorithm as applied to the ICA techniques IFA and MFICA, has a problem of slow convergence. Analysis and simulations show that this slowdown is due to the fact that EM, also far from the optimal point, approximately becomes a linear update with very bad convergence properties. The determinant have an influence on the performance of EM in the sense that both too small and too large determinants of the mixing matrix, makes the convergence of EM slow.

Numerical investigations demonstrates that using different optimization techniques, one can improve the convergence dramatically at a very modest analytical cost. The Adaptive Overrelaxed EM and especially the quasi-Newton methods Ucminf is applied with very good results. The Adaptive Overrelaxed EM is computationally as uncomplicated as EM but is much faster and therefore a very good alternative for MFICA and IFA and perhabs any maximum likelihood estimation using EM.

Using either of the two improved optimizations this broadens the applicability of MFICA and IFA. With respect to MFICA this constitutes an important step forward because it makes the technique both flexible with respect to priors, dimensionality and constraints on the mixing matrix and possibly noise estimation *and* numerically efficient. In this perspective, we believe that using improved optimization we have not only speeded up the process but also opened up a new area of problems for practical and efficient solutions.

7. REFERENCES

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of Royal Statistics Society, Series B*, vol. 39, pp. 1–38, 1977.

[2] G. J. Lachlan and T. Krishnan, *The EM Algorithm and Extensions*, John Wiley and Sons, 1997.

[3] T. A. Louis, "Finding the observed information matrix when using the em algorithm," *Journal of Royal Statistical Society, Series B*, vol. 44, pp. 226–233, 1982.

[4] Mortaza Jamshidian and Robert I. Jennrich, "Conjugate gradient acceleration of the em algorithm," *Journal of the American Statistical Association*, vol. 88, no. 421, 1993.

[5] Mortaza Jamshidian and Robert I. Jennrich, "Acceleration of the em algorithm using quasi-newton methods," *Journal of Royal Statistical Society, Series B*, vol. 59, no. 3, pp. 569–587, 1997.

[6] R. Salakhutdinov and S. Roweis, "Adaptive overrelaxed bound optimization methods," International Conference on Machine Learning, ICML, 2003.

[7] O. Bermond and Jean Francois Cardoso, "Approximate likelihood for noisy mixtures," in *Proceedings of the ICA Conference*, 1999.

[8] Pedro Hojen-Sorensen, O. Winther, and L. K. Hansen, "Mean-field approaches to independent component analysis," *Neural Computation*, vol. 14, pp. 889–918, 2002.

[9] Hagai Attias, "Independent factor analysis," *Neural Computation*, , no. 11, pp. 803–851, 1999.

[10] X. L. Meng, "On the rate of convergence of the em algorithm," *The Annals of Statistics*, vol. 22, pp. 326–339, 1994.

[11] Hans Bruun Nielsen, "Ucminf - an algorithm for unconstrained nonlinear optimization," Tech. Rep. IMM-Rep-2000-19, Technical University of Denmark, 2000.