ON THE DIFFERENCE BETWEEN UPDATING THE MIXING MATRIX AND UPDATING THE SEPARATION MATRIX

Michael Syskind Pedersen, Ulrik Kjems

Oticon A/S,
Strandvejen 58
DK-2900 Hellerup, Denmark
{msp,uk}@oticon.dk

Jan Larsen

Technical University of Denmark
Informatics and Mathematical Modelling
Richard Petersens Plads, Building 321
DK-2800 Kongens Lyngby, Denmark
jl@imm.dtu.dk

ABSTRACT

When the ICA source separation problem is solved by maximum likelihood, a proper choice of the parameters is important. A comparison has been performed between the use of a mixing matrix and the use of the separation matrix as parameters in the likelihood. By looking at a general behavior of the cost function as function of the mixing matrix or as function of the separation matrix, it is explained and illustrated why it is better to select the separation matrix as a parameter than to use the mixing matrix as a parameter. The behavior of the natural gradient in the two cases has been considered as well as the influence of pre-whitening.

1. INTRODUCTION

Consider the independent component analysis (ICA) problem, where n sources $\mathbf{s} = [s_1, \dots, s_n]^T$ are transmitted through a linear mixing system and observed by n sensors. The mixing system is described by the mixing matrix $\mathbf{A}$, and the observations are denoted by $\mathbf{x} = [x_1, \dots, x_n]^T$. This leads to the following equation

$$\mathbf{x} = \mathbf{A}\mathbf{s}, \tag{1}$$

where only the observations $\mathbf{x}$ are known. The objective is to find an estimate $\mathbf{y}$ of the original sources. This can be done by estimating the separation mixing matrix $\mathbf{W} = \mathbf{A}^{-1}$, so that

$$\mathbf{y} = \mathbf{W}\mathbf{x}, \tag{2}$$

Notice, the source estimates may be arbitrarily permuted or scaled. The separation matrix can either be found directly or it can be found by finding an estimate of the mixing matrix and afterwards inverting the mixing matrix, provided that $\mathbf{A}$ is invertible. Here, the classical likelihood source separation is considered.

2. LIKELIHOOD SOURCE SEPARATION

A possible method for solving the ICA problem is the maximum likelihood principle [1]. The ML is closely related to other ICA methods [2] such as the infomax method [3], or maximum a posteriori MAP methods [4], [5]. In maximum likelihood source separation, the probability of a dataset given the parameters $\boldsymbol{\theta}$ of the model should be maximized. In this particular case, for the data $\mathbf{x}$, the parameters are given by either the separation matrix $\mathbf{W}$ or by the mixing matrix $\mathbf{A}$. Thus, the likelihood can be expressed by either

$$p(\mathbf{x}|\mathbf{W}) = |\det \mathbf{W}| \prod_m p_m(\sum_n W_{mn}x_n). \tag{3}$$

or

$$p(\mathbf{x}|\mathbf{A}) = \frac{1}{|\det \mathbf{A}|} \prod_m p_m(\sum_n A^{-1}_{mn}x_n) \tag{4}$$

Here, $p_m(\sum_n A^{-1}_{mn}x_n) = p_m(\sum_n W_{mn}x_n) = p(s_m)$ is the probability density function of the m'th source signal. For source signals such as speech, a heavy tailed source distribution is chosen. One way of maximizing the likelihood, is to minimize the negative logarithm of the likelihood. Given the likelihood functions in (3) and (4), the negative log likelihood functions are given in terms of either $\mathbf{W}$ or in terms of $\mathbf{A}$ as:

$$\begin{aligned} \mathcal{L}(\mathbf{W}) &= -\ln|\det(\mathbf{W})| - \sum_m \ln p_m(\sum_n W_{mn}x_n) & (5)\\ \mathcal{L}(\mathbf{A}) &= \ln|\det(\mathbf{A})| - \sum_m \ln p_m(\sum_n A^{-1}_{mn}x_n). & (6)\end{aligned}$$

The respective gradients of (5) and (6) are given by [6]

$$\begin{aligned} \frac{\partial \mathcal{L}(\mathbf{W})}{\partial \mathbf{W}} &= -(\mathbf{I} + \mathbf{z}\mathbf{y}^T)\mathbf{A}^T & (7)\\ \frac{\partial \mathcal{L}(\mathbf{A})}{\partial \mathbf{A}} &= \mathbf{W}^T(\mathbf{I} + \mathbf{z}\mathbf{y}^T). & (8)\end{aligned}$$

Here $\mathbf{z} = \frac{\partial \ln p_m(\mathbf{y})}{\partial \mathbf{y}}$ is a nonlinear mapping of $\mathbf{y}$. Choosing $\mathbf{z} = -\tanh(\mathbf{y})$ corresponds to a probability density function for $\mathbf{y}$ proportional to $\frac{1}{\cosh(\mathbf{y})}$. This pdf is heavier tailed than e.g. a Gaussian distribution. $\mathbf{I}$ is the identity matrix. The gradient descent update steps are then

$$\begin{aligned} \mathbf{W} &:= \mathbf{W} + \mu_W(\mathbf{I} + \mathbf{z}\mathbf{y}^T)\mathbf{A}^T & (9)\\ \mathbf{A} &:= \mathbf{A} - \mu_A\mathbf{W}^T(\mathbf{I} + \mathbf{z}\mathbf{y}^T), & (10)\end{aligned}$$

where μ_W and μ_A are learning rates. The learning rates can be constant or they can vary as a function of the update step. These algorithms may as well be made into iterative batch versions [7] by averaging over the samples:

$$\begin{aligned} \mathbf{W} &:= \mathbf{W} + \mu_W(\mathbf{I} + E[\mathbf{z}\mathbf{y}^T])\mathbf{A}^T & (11)\\ \mathbf{A} &:= \mathbf{A} - \mu_A\mathbf{W}^T(\mathbf{I} + E[\mathbf{y}\mathbf{z}^T]). & (12)\end{aligned}$$

Here, $E[\cdot]$ denotes the expectation and each sample is assumed to be independent of the other samples.

Thanks to the Oticon Foundation for funding.

3. COMPARISON BETWEEN THE LIKELIHOOD FUNCTIONS

First consider the cost function $\mathcal{L}(\mathbf{A})$. In many source separation problems, the values of the mixing matrix will be relatively close to zero. Large values of **A** are not very likely. If it is assumed that there is a limit on how large $|A_{ij}|$ can be, the n^2-dimensional space occupied by the cost function $\mathcal{L}(\mathbf{A})$ is limited by this maximum value and it is possible to "view" the whole cost function because it only occupies a finite part of the **A**-space. Because the whole cost function is within a finite space, the points where **A** is singular exist in this space too. At a singular point, the cost function $\mathcal{L}(\mathbf{A})$ becomes infinitely large. This makes it hard for gradient descent algorithms to find a minima in a limited space with the existence of infinite values. Now consider $\mathcal{L}(\mathbf{W})$. The space spanned by the $n \times n$ elements in **W** is infinitely large because a limit in the **A**-space doesn't limit the **W**-space. Now consider the behavior of the singular points in the **A**-space when they are mapped into the **W**-space. Recall that the $\{i, j\}$'th element of an inverse matrix can be written as

$$W_{ij} = (\mathbf{A}^{-1})_{ij} = \frac{\text{adj}(A_{ij})}{\det \mathbf{A}}, \qquad (13)$$

where adj is the adjoint matrix. The adjoint matrix, can be found by the following steps:

1. Remove the jth row and the ith column of $(\mathbf{A})_{ij}$.
2. Find the determinant of the remaining part and
3. multiply by $(-1)^{i+j}$.

This means that the $\mathbf{A}^{-1}$ is proportional to $\frac{1}{\det \mathbf{A}}$. At the points where **A** is singular, its determinant is 0. Thus, when **A** becomes singular, **W** becomes infinitely large so all the points in the **A**-space, where $\mathcal{L}(\mathbf{A}) = \infty$ are mapped into the **W**-space far away from the origin and will therefore not disturb the gradient. Because large values of **A** are unlikely, $|\det \mathbf{A}|$ is prevented from becoming too large and hereby, the determinant of **W** is prevented from being close to 0. Hence, it is unlikely that **W** becomes singular.

4. SIMULATION EXAMPLE

The elements of a 3×3 mixing matrix have been drawn from a Gaussian distribution with zero mean and a standard deviation equal to one:

$$\mathbf{A} = \begin{bmatrix} 0.8644 & 0.8735 & -1.1027 \\ 0.0942 & -0.4380 & 0.3962 \\ -0.8519 & -0.4297 & -0.9649 \end{bmatrix} \qquad (14)$$

Hereby,

$$\mathbf{W} = \mathbf{A}^{-1} = \begin{bmatrix} 0.7872 & 1.7481 & -0.1818 \\ -0.3275 & -2.3546 & -0.5927 \\ -0.5492 & -0.4949 & -0.6120 \end{bmatrix} \qquad (15)$$

In order to find $E[\mathbf{z}\mathbf{y}^T]$, 3×1000 samples have been drawn from the $1/\cosh$-distribution[1]. The behavior of the two cost functions $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ are considered as function of two parameters in **A** and **W**, respectively while the other parameters are kept constant. Figure 1 and figure 2 show the cost function $\mathcal{L}(\mathbf{W})$ as function of W_{11} and W_{21}. Figure 1 shows the negative direction of the gradients. The circle ($\circ$) is placed at the correct values of W_{11} and W_{12}, which also can be seen in (15). It can be seen that the negative gradient directions are pointing toward the global minimum, where the circle is located, or toward a local minimum. When considering figure 2, it can be seen that as $\mathcal{L}(\mathbf{W})$ is increased, when $|W_{11}|$ or $|W_{21}|$ is increased. Now consider figure 3. Here $\mathcal{L}(\mathbf{A})$ is shown as function of A_{11} and A_{21}. Here too, the negative gradient directions point toward the minima. In figure 4 the shape of $\mathcal{L}(\mathbf{A})$ can be seen. The values, where **A** is close to singular can clearly be seen and not far from these singular values, the global minimum exists. Due to these huge differences in the cost function within a quite small range, it can be hard to find the correct solution. This is also illustrated in figure 5. Here the value of the two cost functions $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ are shown as function of the number of iterations. The two learning rates are kept constant. They have been chosen such that the cost functions are minimized as fast as possible. The two learning rates has been found to be $\mu_A = 0.03$ and $\mu_W = 0.3$. It can be seen that more iterations are needed in order to find **A** than to find **W**. Further, it can be seen that $\mathcal{L}(\mathbf{A})$ hasn't reached the minimum after 200 iterations. Actually, after 200 iterations the sources are not separated at all when $\mathcal{L}(\mathbf{A})$ is minimized. This can be explained by considering the cost function in figure 4. At the areas around the minimum of $\mathcal{L}(\mathbf{A})$, the cost function has almost the same value as at the minimum. This makes it very hard to minimize, since the gradient decent steps are very small. Even after 500 iterations, the separation quality [8] of the three sources is only between 13 and 41 dB while the separation quality of the sources, where the $\mathcal{L}(\mathbf{W})$ is minimized is between 36 and 88 dB. Even though only $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ have been investigated as function of two parameters in each matrix, the shown behavior of $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ is believed to be a general behavior for any of the parameters.

4.1. Natural Gradient learning

By using natural gradient descent [9] instead of gradient descent, the cost functions may be minimized with a smaller number of iterations. The natural gradients are obtained by multiplying the gradient in (7) by $\mathbf{W}^T\mathbf{W}$ on the right side and the gradient in (8) by $\mathbf{A}\mathbf{A}^T$ on the left side. Hereby, the natural gradient steps are given by [10]

$$\Delta\mathbf{W}_{NG} = -(\mathbf{I} + E[\mathbf{z}\mathbf{y}^T])\mathbf{W} \qquad (16)$$

$$\Delta\mathbf{A}_{NG} = \mathbf{A}(\mathbf{I} + E[\mathbf{z}\mathbf{y}^T]). \qquad (17)$$

The natural gradient update steps have been used in the separation problem. As it can be seen in figure 5, the separation performance works equally well whether the natural gradient is used in the A-domain or in the W-domain. Hereby, it seems that the natural gradient is able to erase the convergence difference between updating the algorithm in the A-domain and in the W-domain.

4.2. Pre-whitening

Pre-whitening of the data may simplify the separation problem. After pre-whitening the data **x** is uncorrelated and

$$E[\mathbf{x}\mathbf{x}^T] = \mathbf{I} \qquad (18)$$

The update equations (11), (12), (16) and (17) have been applied in the case, where the data has been pre-whitened. Figure 6 shows

[1] Artificial data y which is $\frac{1}{\cosh}$-distributed can be generated from uniformly distributed data as $Y = \ln|\tan(X)|$, where X is a uniformly distributed random variable over the interval $0 < x < \pi$.

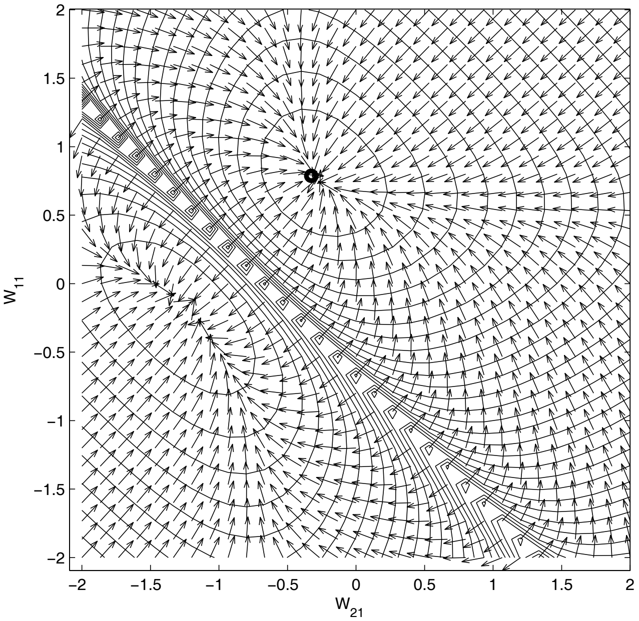

Fig. 1. The cost function $\mathcal{L}(\mathbf{W})$ as function of W_{11} and W_{21}. The other elements in **W** are held constant by their true value. The direction of the gradients are shown as well.

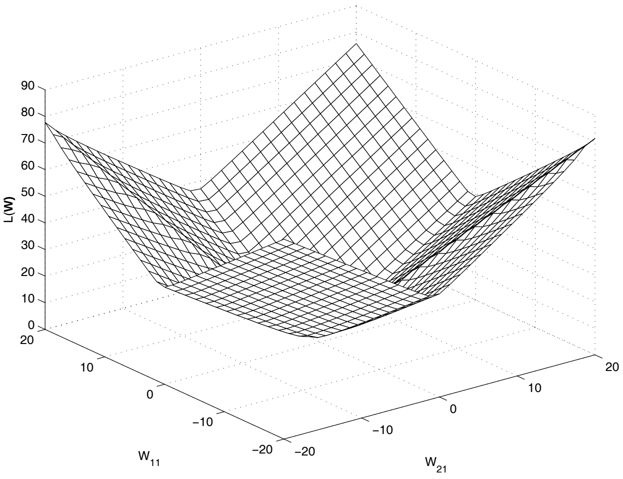

Fig. 2. The cost function $\mathcal{L}(\mathbf{W})$ as function of W_{11} and W_{21}. As it can be seen, as the parameters in **W** are increased, $\mathcal{L}(\mathbf{W})$ is increased too.

how the cost functions are minimized as function of the number of iterations. As it can be seen, the convergence time is significantly improved. Still, when **A** is updated in the **A**-domain without the natural gradient, convergence is slow compared to updating in the **W**-domain.

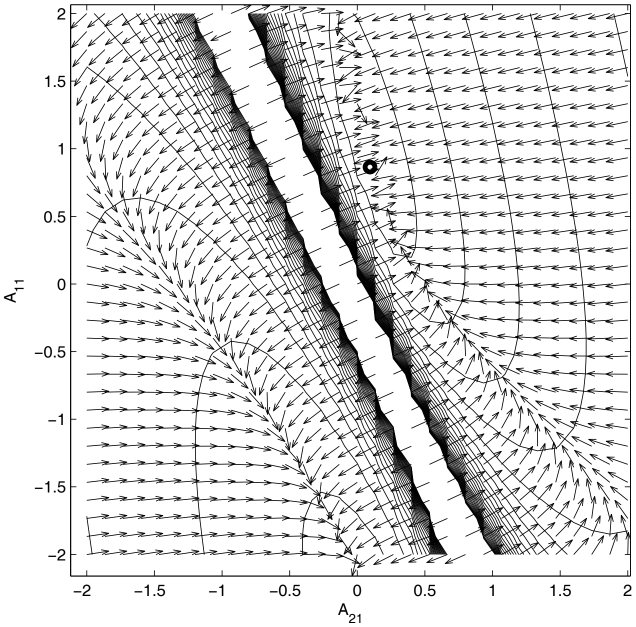

Fig. 3. The cost function $\mathcal{L}(\mathbf{A})$ as function of A_{11} and A_{21}. The other elements in **A** are held constant by their true value. The direction of the gradients are shown as well.

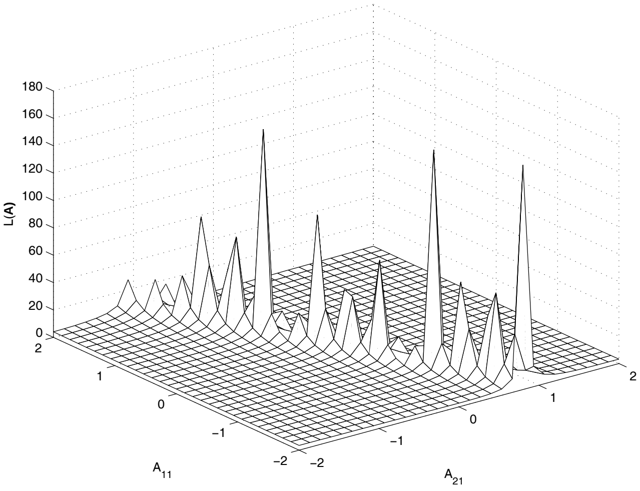

Fig. 4. The cost function $\mathcal{L}(\mathbf{A})$ as function of A_{11} and A_{21}. As it can be seen, the cost function is dominated by a high ridge, where the mixing matrix is close to singular, and some flat areas. Compared to the cost function in figure 2, it is much harder to find the global minima.

5. CONCLUSION

When performing source separation based on minimizing a cost function by gradient descent, the shape of the cost function is important. By comparing the negative log likelihood cost function as either function of the mixing matrix or as function of the separation matrix, the contours of the cost functions are very different. Due

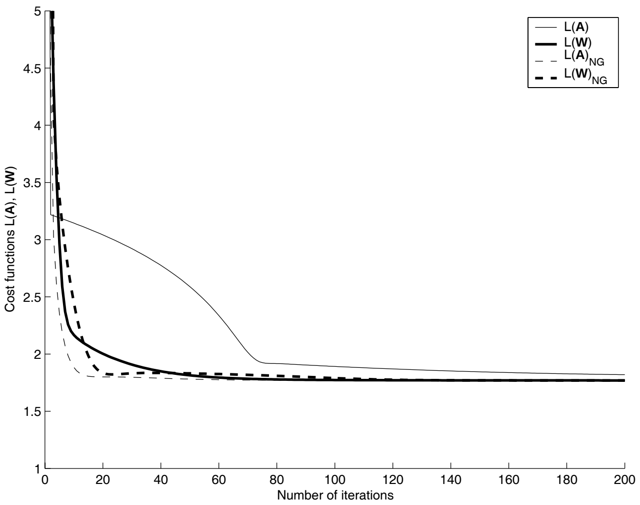

Fig. 5. The cost function $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ as function of the number of iterations. The constant learning rates are selected in order to minimize the number of iterations in order to ensure convergence. After 200 iterations, only $\mathcal{L}(\mathbf{W})$ has been minimized. Even though $\mathcal{L}(\mathbf{A})$ seems to have been minimized as well, $\mathbf{A}$ has not been correctly estimated. Only a value of $\mathbf{A}$ somewhere at the flat areas in figure 4 has been found, and much more iterations are needed in order to find the correct value of $\mathbf{A}$. Also, the minimization as function of the iterative update by use of the natural gradient is shown. Here, the cost functions are minimized by use of a smaller number of iterations and fast convergence is achieved for the update of $\mathbf{A}$ as well as $\mathbf{W}$. By using the natural gradient, the difference between updating the algorithm in the two domains seems to have disappeared.

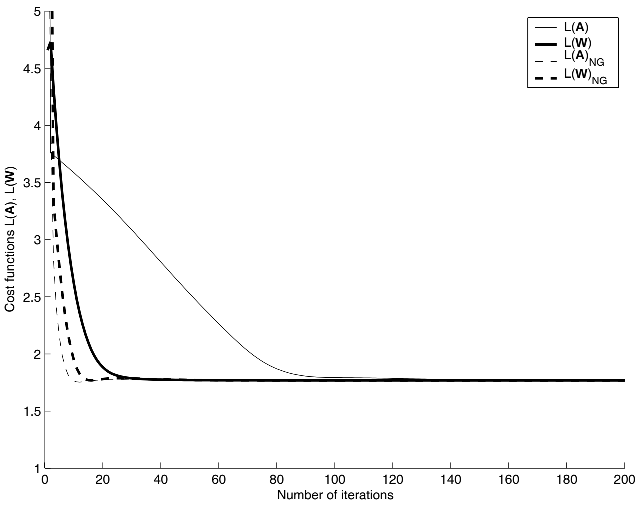

Fig. 6. The cost function $\mathcal{L}(\mathbf{A})$ and $\mathcal{L}(\mathbf{W})$ as function of the number of iterations as in figure 5. Contrary to figure 5, the data has been pre-whitened before the four update equations have been applied to the data. As it can be seen, pre-whitening increases the convergence speed significantly. Still, the update in the $\mathbf{A}$-domain without the natural gradient has the slowest convergence speed.

to these different behaviors of the cost functions, it has been found that it is much easier to minimize the negative log likelihood, when it is a function of the separation matrix than as function of the mixing matrix. If the natural gradient is applied in the mixing domain, it is able to cope with the difficult contour of the cost function. But in problems, where the natural gradient is hard to find, a proper choice of the parameters may be crucial. Also pre-whitening has been considered. By pre-whitening the data before applying the ICA algorithm, the convergence is significantly increased. The results may be generalized to more difficult problems such as e.g. convolutive ICA.

6. REFERENCES

[1] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja, *Independent Component Analysis*, Wiley, 2001.

[2] Jean-François Cardoso, "Blind signal separation: Statistical principles," *Proceedings of the IEEE*, vol. 9, no. 10, pp. 2009–2025, October 1998.

[3] Anthony J. Bell and Terrence J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[4] Pedro A.d.F.R. Højen-Sørensen, Ole Winther, and Lars Kai Hansen, "Mean field approaches to independent component analysis," *Neural Computation*, vol. 14, pp. 889–918, 2002.

[5] Kevin H. Knuth, "Bayesian source separation and localization," in *Proceedings of the SPIE Conference on Bayesian Infernce on Inverse problems*, San Diego, California, July 1998, pp. 147–158.

[6] David J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 1st edition, 2003.

[7] Nikos Vlassis and Yoichi Motomura, "Efficient source adaptivity in independent component analysis," *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 559–565, May 2001.

[8] D.W.E. Schobben, K. Torkkola, and P. Smaragdis, "Evaluation of blind signal separation methods," in *Int. Workshop Independent Component Analysis and Blind Signal Separation*, Aussois, France, January 11–15 1999, pp. 261–266.

[9] Shun-ichi Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, pp. 251–276, 1998.

[10] Andrezej Cichocki and Shun-ichi Amari, *Adaptive Blind Signal and Image Processing*, Wiley, 2002.