Mean Field ICA

Kaare Brandt Petersen

Kongens Lyngby 2005
IMM-PHD-2005-117

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

IMM-PHD: ISSN 0909-3192

Abstract

This thesis describes investigations and improvements of a technique for Independent Component Analysis (ICA), called "Mean Field ICA".

The main focus of the thesis is the optimization part of the algorithm, the so-called "EM algorithm". Using different approaches it is demonstrated that the EM algorithm is inefficient and therefore an improper choice for a certain class of models. As an alternative, the so-called "Easy Gradient Algorithm" is presented, which makes it possible to apply advanced optimization techniques using the computationally simple E-step and M-step of the EM algorithm. The Easy Gradient Recipe is applicable to a wide selection of models. Furthermore, the Mean Field ICA model is extended to incorporate filtering over time in a so-called "convolutive ICA" model.

Finally, by using mixture of Gaussians as source priors, the generative and filtering approach to ICA is compared in the overcomplete setting, i.e. the situation in which there is more sources than sensors.

Resumé

Denne afhandling omhandler undersøgelser og forbedringer af en teknik til uafhængig komponent analyse (Independent Component Analysis, ICA), kaldet "Mean Field ICA".

Afhandlingens fokus er i særlig grad på optimeringsdelen af algoritmen, den såkaldte EM algoritme, der fra forskellige synsvinkler vises at have betydelige defekter der gør den uegnet i denne type af modeller. Som et alternativ til EM algoritmen foreslås den såkaldte "Easy Gradient Recipe", der er en foreskrift for hvordan man med de samme simple beregninger som indgår i EM algoritmen, kan opnå en langt mere avanceret optimering og dermed hurtigere konvergens. Denne alternative optimeringsform er også brugbar for en lang række andre modeller. Desuden udvides "Mean Field ICA" modellen til at inkorporere tidsligt ekko i de målte signaler - en modeludvidelse der kaldes "foldet ICA" eller på engelsk "convolutive ICA".

Til sidst gives, ved hjælp af gaussiske mixturer som modeller for kilderne, en sammenligning af to forskellige tilgange til ICA: Den generative hvor man opstiller en generativ model, og den filtrerende, hvor de målte signaler behandles med filtre. Sammnenligningen giver et indblik i hvordan de to tilgange er relaterede i tilfælde af flere kilder end sensorer.

Preface

During a trip to China and Thailand in 1997 I one day found myself in a tiny bookstore in Bangkok looking for some literature to keep me entertained: a week at the beaches of Ko Chang was ahead and surely I needed something to read. In this bookstore I found the book 'What the Buddha Taught', a compact presentation of Buddhism from an intellectual non-ritual point of view, which in many ways appealed to me and reminded me that if I were ever to leave atheism for something more spiritual, it would probably be Buddhism. In this book, which is indeed well-written and worth reading, I found the quote

> "There is no point in saying that one should not doubt or one should believe. Just to say 'I believe' does not mean that you understand and see. When a student works with a mathematical problem, he comes to a stage beyond which he does not know how to proceed, and where he is in doubt and perplexity. As long as he has this doubt, he cannot proceed. If he wants to proceed, he must resolve this doubt. And there are ways of resolving that doubt. Just to say 'I believe', or 'I do not doubt' will certainly not solve the problem. To force oneself to believe and accept a thing without understanding it, is political, and not spiritual or intellectual.", *W. Rahula in 'What the Buddha Taught'*

This quote not only parallels the mathematical process of learning with a spiritual one, but indeed does it in a way which expresses a core part of my professional personality: "As long as he has this doubt, he cannot proceed. If he wants to proceed, he must resolve this doubt." Making such impractical approaches to learning and working within science is to make life unnecessary hard, but if not taken to extremes it may indeed prove a useful road for some and earn its place in academia. For my part, I hope there is still room and use for people of that kind.

This Ph.d. Project

The project, of which this thesis is part, was generously funded by the foundation "William Demant's og Hustrus fond" (Oticonfonden). It was initiated in May 2002 and I have handed in this thesis in May 2005. The title of the project, "Mean Field ICA" was indicating that having written a master's thesis on a simpler version of the same algorithm, would give me a head start and make the first confusing phase of the project more orderly and focussed. And indeed it is possible that had I not written my master in a closely related topic, it would have taken even longer to get to the heart of the matter. But there is of course always trouble ahead; in fact I suspect that in the cases of no trouble, students and researchers seek new directions until the challenges and troubles reaches a certain almost intolerable level, and from there start solving the problems as they best can. Regarding this project on the "Mean Field ICA" algorithm, however, no new directions or extra tasks were needed – the basic task of understanding and improving the algorithm proved sufficiently challenging.

In the project description, the main task was clear: Expand the model to non-linear and convolutive mixtures, examine the properties of the algorithm facing over- and underestimated situations, and investigate the two optimization steps to increase performance and robustness. All of these topics are interesting and relevant. Beginning on the project I made the first major mistake believing that the algorithm, which I had been working with for binary sources, were basically sound and efficient for other sources as well. Not paying due attention to robustness from the beginning I started out expanding the model to the convolutive case and spent many months wondering and debugging since it did not really perform as one could expect. Only after more than a year, I decided after consulting my supervisor Ole Winther to make a step backward and investigate the efficiency and robustness for ever simpler setups. Since I was preoccupied with also taking courses, doing teaching, and planning of the external research stay at UCSD, it took almost two years until the use of exact approaches to both the source posteriors and the cost function made it clear: something very fundamental in this approach was causing the trouble. It turned out to be inefficiency of the optimization technique (the EM algorithm) which was the root of the problems and I started writing papers on how and why this was the case. From then on, things went fast, but indeed they had to, since very little time was left.

One could argue that instead of insisting on the performance of the Mean Field ICA, I could have turned my attention to related but different, easier and more rewarding models. But I have kept true to the initial idea doing exactly what I signed up for in the project description – also when it after more than a year without good results seemed as an infinite well of unsolvable problems. Doing that, the satisfaction after in fact solving the efficiency matter and the sense of winning a very long tiring duel, was accordingly great. Were I to start over, I might have done it differently, however, since the rather stubborn approach have brought me perhaps too many frustrating hours after all. Being in the middle of the struggle I have taken great pleasure sharing my views with colleagues and even more in listening to their experiences. One of these experiences I read in a book of Emanuel Dermand, a physicists who eventually turned to the financial sector at Wall Street. He writes in retrospect about his time as a PhD student:

> "Despite the amount of time it took to finish my thesis, I have no real regrets; in a way, I am proud of the struggle. What I learned in those years – perseverance as much as mathematics – has stood me good in stead on Wall Street as well as in academia. When trying to discover something new in any field, one has to spend many years thinking, making false starts, wandering down blind alleys and stumbling into ditches, only to emerge again and keep going. For this, a Ph.D. is a good, if painful, training.", *E. Derman in 'My Life as a Quant'.*

As is often the case, it is for some reason relieving listening to people who have had the same or similar worries as oneself and the way E. Dermand describes the process above resembles the process of the "Mean Field ICA" project quite a bit. In what direction my professional life will go after this thesis is still unknown, but I have no doubt that the experience gained during the these three years as a PhD student is widely applicable and will serve me well, where-ever life takes me.

Scientific Contributions

Chronologically I started out in collaboration with Lars Kai Hansen on a paper on monaural separation in which we applied Mean Field ICA on a special bi-modal gaussian source [8]. From then on, I expanded the Mean Field ICA to the convolutive case but had trouble reaching a satisfying level of robustness and quality. In the fall 2004 Ole Winther and I submitted a paper to the conference ICASSP2005 demonstrating a small scale numerical investigation of the convergence properties of the EM algorithm [42]. Shortly after that, the saddle point analysis led to the demonstration of the poor convergence properties of the EM algorithm for a large class of problems – reaching well beyond ICA – and this was first published as a technical report in [41]. An expansion of this result to the Variational Bayes EM algorithm, was shortly after submitted to Neural Computation in [43]. In early 2005 I did some work in collaboration with Rasmus K. Olsson and Tue Lehn-Schiøler which demonstrated the result on the poor EM convergence for state space models [34]. In the time short before handing in the thesis I submitted the work done with Jiucang Hao and Te-Won Lee comparing the filtering and generative approach to overcomplete ICA. Summarizing my scientific contributions with respect to topics, I have been contributing to the progress within the following areas:

1. Analysis of the slow convergence of the EM algorithm and Variational Bayes EM algorithm. As a proposed solution to the problem I have suggested the Easy Gradient Recipe.

2. Expansion of the Mean Field ICA to the convolutive Mean Field ICA.

3. Analysis of filtering and generative approaches to overcomplete ICA using the so-called Energy Based Models.

The list of publications I have contributed to, is:

Publications (Accepted and in progress)

- K. B. Petersen, O. Winther, L. K. Hansen, "On the slow convergence of EM and VBEM in low noise linear mixtures", Neural Computation 17(9), 2005. [43]

- K. B. Petersen, J. Hao, T-W. Lee "Generative and Filtering Approaches for Overcomplete Representations", Neural Information Processing 8(1), 2005. [40].

- R. K. Olsson, K. B. Petersen, T. Lehn-Schiøler, "State-space models - from the EM algorithm to a gradient approach", Submitted to Neural Computation, 2005. [34]

- K. B. Petersen, O. Winther, "Explaining Slow Convergence of EM in Low Noise Linear Mixutres", IMM-TechRep 2-2005, Technical University of Denmark, 2005. [41]

- K. B. Petersen, O. Winther, The EM Algorithm in Independent Component Analysis, IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, 2005. [42]

- K. B. Petersen, Using Mixture of Gaussians to Compare Approaches to Signal Separation, DSAGM, 2004. [38]

- L. K. Hansen, K. B. Petersen, Monaural ICA of white noise mixtures is hard, Proceedings of ICA'2003 Fourth Int. Symposium on Independent Component Analysis and Blind Signal Separation, Nara Japan, April 4,, pp. 815-820, 2003. [8]

- K. B. Petersen, M. S. Pedersen, The Matrix Cookbook, Technical University of Denmark. Regular updates, latest in 2005. [37]

The last item, The Matrix Cookbook, may deserve some explanation. While visiting UCSD I realized how much time I spend on almost the same analytical calculations, over and over. Most of them being matrix related identities of derivatives, integrals, or reformulations, I turned to the reference literature like

Schaum's for a complete collection. Since I didn't find any, I started collecting identities, inequalities, integrals, derivatives and so on, in a tex document, which became The Matrix Cookbook. In the summer of 2004, encouraged by the interest at my home department, I announced it at two mailing-lists and was quite surprised when it was downloaded more than one thousand times within the first 24 hours. I realized that other researchers were lacking these identities and joined with Michael Syskind Pedersen in producing a larger and better version. Updating and releasing it regularly, it has now been downloaded more than 10000 times, which makes it the most downloaded document from the institute ever. One may argue that collecting knowledge which largely already exists, is not really doing research. For one thing one may disagree with that opinion, but more importantly, the usability and popularity does in my view speak for itself.

Acknowledgements

First of all I would like to thank my primary advisor Ole Winther, who stronger than most have had an open door policy: at almost any time, I could walk into his office to ask a question or debate some issue. Not only is it very good for the daily business to resolve difficulties straight away when possible, it is also strongly motivating to have the opportunity to discuss matters with people who have a deep understanding of the matters. Moreover, Ole was supportive in the darkest hour when, after two years of futile research, I had a hard time keeping up the optimism. I surely hope that I haven't drained Ole from energy reserved for other matters and that other students will also benefit from his knowledge and personality.

In 1999 I was working on a bachelor project and payed the department led by Lars Kai Hansen a short visit. After listening to a PhD defense, my coworker and I spend a few minutes discussing the project with Lars. Then around 2001, when I started looking for a good subject and a supervisor for my master's thesis, I talked to a number of people at the Niels Bohr Institute of physics at University of Copenhagen. Among them good scientists who presented projects which were allegedly perfect for a student like me with my qualifications and so on. But when I met them the day after (literally), they had no idea who I was, seemingly completely oblivion to me and the discussion we had had the day before. Disappointed about their social abilities, but lacking good alternatives, I pondered on what lesser evil to choose, when I one day coincidentally passed Lars from DTU in a hallway in the institute. Although it was two years since I had talked to him for just a few minutes, he briefly looked up, (seemingly) recognized me, and said hello. Since Lars also had a very good reputation as a scientist, I went out to DTU to talk to him about a possible master project and later stayed at the department doing this PhD project. For a large part of the project, Lars Kai Hansen has also been an advisor, with numerous good advices and comments. Lars, truly unique for his combination of good humor and scientific capacity, has been a magnet on student interest and I am sure future PhD students will also be happy working with him.

The good social skills of Lars has also benefitted me when going abroad for a so-called external research stay in San Diego. At the Institute for Neural Computation at UCSD, a small group led by Te-Won Lee was graciously welcoming me. Surrounded by Chinese and Korean people (but no Americans), I learned a lot about both culture and science from Te-Won and his crew. Te-Won was honest, professional and relaxed, and contributed to seven very good and enriching months under the sunny Californian skies. Thank you for the warm welcome and good discussions.

Also thanks to fellow PhD students which through many discussions and joint work, have contributed to both the fun and the science. Among these especially Tue Lehn-Schiøler and Rasmus K. Olsson for joint work on the State Space Models, Jiucang Hao for joint work on the Energy Based Models and Michael Syskind Pedersen for joint work on The Matrix Cookbook. I should also thank the people like Niels and Mogens who made the daily business more easy helping me with a gazillion computer related questions and Ulla who is probably the real reason the ISP Group at DTU is working so well. Thanks also to Anette Jensen, Tue Lehn-Schiøler, Rasmus Kongsgaard Olsson and Anders Meng for proofreading of this thesis.

Last but not least I would like to thank my wife Line, who has been suffering most from the difficulties of being closely related to a PhD student. Apart from the research-wise challenges, one of the side-effects of the work as PhD student is the solitude which sets in the instant one is not by hard work successfully involving other people in common projects. At those times frustrations – which are therefore not resolved with colleagues – turn up in various guises in the private life and thus making partners subject to rather unfair mood swings. Line has been patient at all times and especially helpful on a critical day in the suburbs of San Diego when I, working too hard for too long, had lost perspective. Thank you.

Kaare Brandt Petersen
Copenhagen, May 2005.

Contents

Chapter 1

Introduction

Once upon a time on a dusty African savanna, a prehistoric man – yet unaware of the revolutionary developments which will assists his offspring in survival and proliferation – was pondering the fundamental question: "Where at the dried out river bed is it most likely that the antelope will try to jump?". Although he knew that the deer was agile and therefore able to make it over the slope at least two different places, he chose to hide in the bushes at the lower of the two.

Even in this anniversary year of Hans Christian Andersen, it may require some explanation to bring in fairy tales. The reason is that the task of the prehistoric man is early probabilistic modeling: Not knowing the mind of the deer, it is essentially a random process guessing where the deer will try to escape, but there are clues which make the distribution over the possible outcomes non-uniform. There is the prior knowledge about that kind of animal – knowledge which tells the prehistoric man that the Kudu antelope can easily jump over fences 2 meters high. But there is also the specific knowledge (in analogy to data) that this specific individual has been running from the other tribesmen for quite a while and therefore most likely is tired and not eager to attempt the higher jump. Therefore, the prehistoric man is combining prior knowledge with data and make a decision.

When humans are doing probabilistic modeling it is most often done unconsciously in the sense that the choice is based on intuition, rather than rational reasoning. This reflects that the functionality of the brain for many basic situations is far superior in estimating probabilities compared to rational reasoning. Also, the ability to consciously express the perceived clues in a given situation may be impaired without there being anything wrong with the clues themselves.

The modern development of expressing the probabilities in a mathematical language has very significantly influenced this balance between intuition and reasoning. Early formulation of probability theory gave improvements of simple situations and as the theory has grown more advanced, more and more phenomena has been described and predicted using probabilistic modeling. In many situations, the human ability is incapable of making precise choices either because of complexity or some counter-intuitive mechanism, and mathematical models are in many of these situations now assisting human reasoning. A remaining barrier in that respect, is the computational power: Many probabilities are determinable on principle only, since it is still impractical to estimate them analytically or numerically. This aspect is, of course, ever changing as each new generation of computer processors is developed and where it ends, only time will tell.

This thesis is describing and analyzing the Mean Field ICA algorithm which is an example of modern probabilistic modeling. It is both complicated and computationally intensive and in that sense truly a creature of the new century.

1.1 Structure of the Thesis

Chapter 2 introduces the Bayesian framework for estimation in models with unobserved variables. Only the ideas and overall derivation are taken into account, leaving the practical issues of integration and optimization untouched. Chapter 3 is introducing and analyzing the Expectation-Maximization (EM) algorithm, a method for maximum likelihood estimation. Chapter 4 is introducing the mean field theory as a way to approximate integrals efficiently and show how the work of Chapter 2 and Chapter 3 combines into the Mean Field ICA algorithm. Chapter 4 also shows some properties of the Mean Field ICA algorithm and an extension to the convolutive model. Chapter 5 describes work on the Energy Based Models – some work done during a external research stay at the University of California San Diego.

Chapter 2

Probabilistic Estimation

Probabilistic estimation is the game of learning from data in a way which can be interpreted within a probabilistic framework. In this rather broad definition it also includes estimating probabilities based on experience from previous events. Learning is here meant in a general sense as any kind of knowledge which can be inferred from observations. And since only completely independent processes yields no mutual information, the smallest connection or weakest correlation may be used for inference. In a slightly more technical sense, though, learning means estimation of parameters based on observed data.

The property of an estimation as being interpretable within a probabilistic framework is important and non-trivial. In many applications and situations the possibility of justification, error-estimation or some further data processing is essential, and the probabilistic framework lends itself readily to that task. In many legal or medical applications, for example, it may be inadequate to state black-box results no matter how reliable experience have proved the method to be. Results which can be communicated as a probability, on the other hand, may be acceptable and immediately understandable also to people not familiar with the technical sciences.

The condition that the process must be possible to give probabilistic meaning may seem weak or obvious, but in fact many data analysis techniques does not have this property. Early neural networks, for example, were not formulated in the language of probability theory and well-established techniques, such as Support Vector Machines, and K-nearest neighbors, are not subject to direct meaningful probabilistic interpretation. Thus, probabilistic estimation is both non-trivial, relevant for interpretation and application, and interesting in its own right as an academic research area.

In this chapter a framework for probabilistic estimation in so-called 'hidden variable' models is presented. Focussing on the essence from a theoretical viewpoint, no practical or computational obstacles are considered in this chapter – thus leaving out the optimization and integration issues. I have not read this material in any specific reference but it is largely well-known and included merely for completeness. One exception is the derivation of the probabilistic convolutive ICA, which I derived with focus on the convolutive Mean Field ICA presented in Chapter 4.

2.1 Probabilistic Approaches

2.1.1 The Bayesian Principles

Thomas Bayes (1702-1761) was an English mathematician and a presbyterian minister in a time in which most mathematical progress was done in continental Europe. He almost didn't publish during his lifetime but were influential and became fellow of the Royal Society in 1742. After his death, a friend published in 1763 *Essay Towards Solving a Problem in the Doctrine of Chances*, in which most of the material he is known for today is presented. In the modern communities of mathematical modeling, Bayes and Bayesian approaches are related to the following three statements which can be regarded as principles, postulates, model selection tools, or model building guidelines:

i) Everything can be assigned a probability.

ii) Bayes' theorem $p(A, B) = p(A|B)p(B)$.

iii) Integrate out what you do not know.

Regarding i): The first claim that every variable can be assigned a probability is a postulate which is fiercely debated by frequentists. The frequentists argues that a probability is only well defined as a ratio between successful outcomes and total number of outcomes. The Bayesian viewpoint, on the other hand, is that one may assign a probability or density to any variable as long as the density is fulfilling the ordinary constrains for being a density in the first place. To distinguish between probabilities in the sense of Bayes and frequentists, the Bayesian probabilities are sometimes referred to as 'beliefs' instead of probabilities.

Regarding ii): Bayes's theorem is the most well-known of the three. It simply states a certain connection between the joint and the conditional probabilities of a variable. Since joint probabilities are often difficult to handle and models are well expressed by conditional probabilities, this theorem has had immense importance in probabilistic modelling.

Regarding iii): The third is to integrate out what is unknown. Say the entity x is of interest, but x is somehow dependent on the variable or parameter y about which only $p(y)$ is known. Then the Bayesian approach is to do the following integral using Bayes' theorem, $p(x) = \int p(x|y)p(y)dy$. In this way no assumptions or estimation of the unknown y is made, but instead the lack of knowledge is compensated by the only thing which *is* known, namely the distribution $p(y)$.

In implementations, the third Bayesian point is often utilizing the first two and is the crown jewel of the Bayesian approach: a formalized and sensible way to deal with the lack of certain knowledge. But the clarity is to some extend deceptive since integrating out one unknown entity, almost always is in need of both a model and a prior, which again is unknown and must integrate out using another model and another prior, which again is unknown, and so on. Shortly, there is always something more that one do not know and the 'integrate it out'-scheme formally never ends.

2.1.2 Likelihood versus the Posterior

The Bayesian principles does not themselves induce an estimation scheme or cost function but provide tools and guidelines for how these can be obtained. In this subsection, different cost functions for a data set $\mathbf{X}$ is considered, along with a model in which some model parameter $\boldsymbol{\psi}$ must be estimated. Two approaches are discussed, the maximum likelihood (ML) and the maximum a posteriori (MAP), and in both the Bayesian principles can be implemented to different degrees.

In the basic ML approach, one estimates the model parameters as the parameters which maximize the probability of the observed data set

$$\text{(ML)} \qquad p(\mathbf{X}|\boldsymbol{\psi})\,. \tag{2.1}$$

In a broader and more intuitive language one is choosing the cause (model parameters) which makes the observed event (data set) as probable as possible. A related but slightly different approach is the basic MAP approach, in which one is estimating the model parameters as the parameters which maximizes the parameter posterior, that is, maximizing the cost function

$$\text{(MAP)} \qquad p(\boldsymbol{\psi}|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{\phi}_\psi)}{p(\mathbf{X})}\,. \tag{2.2}$$

Using the MAP estimate one is choosing the most likely cause (model parameter) given the observed event (the data). As the right hand side of Equation 2.2 is demonstrating, the likelihood and posteriori are related through Bayes theorem. From that equation it can be seen that the essential difference between ML and MAP is the parameter prior $p(\boldsymbol{\psi}|\boldsymbol{\phi}_\psi)$ and that if this prior is non-informative, i.e. if one has absolutely no prior knowledge of the parameters, then the ML estimate and the MAP estimate are equivalent.

Up to this point, two of the Bayesian principles have been employed: A prior probability has been assigned to the model parameters and used Bayes theorem to calculate the posterior. The third principle which involves integrating out the unknown entities, however, has not been employed. The unknown entity in this context is the model parameters $\boldsymbol{\psi}$. Further, instead of using a single value as the ML-estimate or MAP-estimate, one could estimate a probability distribution over $\boldsymbol{\psi}$, parameterized by the data $\mathbf{X}$ and the prior parameter $\boldsymbol{\phi}_\psi$, and use that for a more detailed and balanced estimate. Integrating out the parameter in the ML setup, one get the following expression, in which the left hand side is dependent on the hyper-parameter $\boldsymbol{\phi}_\psi$

$$\text{(ML-II)} \qquad p(\mathbf{X}|\boldsymbol{\phi}_\psi) = \int p(\mathbf{X}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{\phi}_\psi)d\boldsymbol{\psi}\,. \tag{2.3}$$

One can now maximize the probability of the observed data set $\mathbf{X}$ given the hyper-parameters to obtain an estimate of the hyper-parameters. Note how this resembles the basic ML, but with respect to integration is taken one step further. For that reason it is sometimes referred to as empirical Bayes, the evidence framework, or ML-II. Note also, however, that one could in principle keep integrating using hyper-hyper-parameters and in that sense a full Bayesian have not been employed, but rather 'a first level' of the Bayesian principles.

As with the basic approaches, one can also for this setup estimate the unknown entities using the posterior instead of the likelihood. The posterior of the hyper-parameter $\boldsymbol{\phi}_\psi$ is derived using Bayes' theorem

$$\text{(MAP-II)} \qquad p(\boldsymbol{\phi}_\psi|\mathbf{X},\boldsymbol{\phi}_{\boldsymbol{\phi}_\psi}) = \frac{p(\mathbf{X}|\boldsymbol{\phi}_\psi)p(\boldsymbol{\phi}_\psi|\boldsymbol{\phi}_{\boldsymbol{\phi}_\psi})}{p(\mathbf{X}|\boldsymbol{\phi}_{\boldsymbol{\phi}_\psi})}\,. \tag{2.4}$$

In order to do that one must choose a prior on the hyper-parameter, $p(\boldsymbol{\phi}_\psi|\boldsymbol{\phi}_{\boldsymbol{\phi}_\psi})$ and include that in the estimation. The infinite Bayesian process of integrating out ends for all practical purposes where one assume some prior to be non-informative. In that sense, a full Bayesian treatment does in fact stop at some point, namely when there is no more prior knowledge to be implemented in the scheme.

So which is the better, the maximum likelihood or the posterior estimate? Since through Bayes' theorem the posterior is incorporating the likelihood, the posterior is the most general framework. However, at the point where one can employ no more prior knowledge, that is, the point at which the prior is non-informative, the two estimates are equivalent.

2.1.3 Hidden Variable Models

The data, the model, and the model parameters are strictly speaking the only concepts needed, but a certain class of models suggest to introduce what is

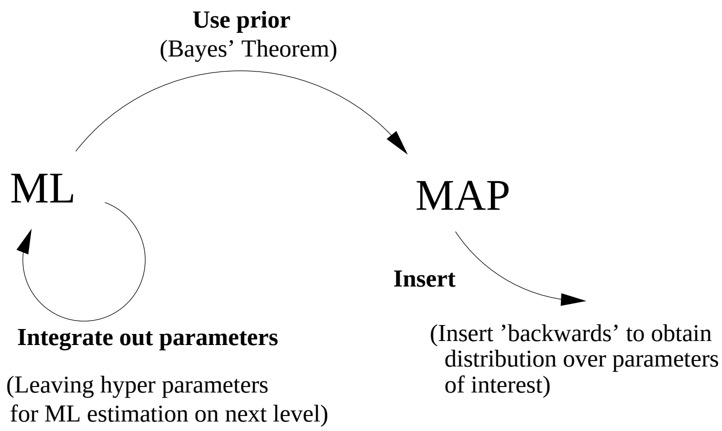


Figure 2.1: The Bayesian Mill: keep integrating out unknown parameters in a maximum likelihood (ML) setting until the prior information is non-informative or impossible to integrate further. At that point use the prior (if informative) to obtain the posterior and use it for estimating the (hyper) parameter at that level. Use the posteriors successively to estimate the distribution over the variables/parameters of interest.

referred to as 'hidden variables', 'latent variables' or 'unobserved data'. The hidden variables are an unknown part of the model and is formally no different from the model parameters. But it may prove useful anyway – at least from a conceptual point of view – to distinguish the hidden variables from the model parameters.

To introduce the hidden variables one must be slightly more concrete: imagine a set of N observed data column-vectors, $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N$ each of m-dimensional and real, i.e. $\mathbf{x}_t \in \mathbb{R}^m$. For notational convenience, these data-vectors are stacked side by side into the large data matrix $\mathbf{X}$, which is then $m \times N$ dimensional. The hidden variables denoted $\mathbf{S}$ is a $d \times N$ matrix and a part of the model which in size scales with the number of observed data vectors N. In the language of statistical physics they are extensive variables as opposed to the model parameters which do not scale with the size of the data set and are referred to as intensive variables.

The important difference between extensive and intensive variables is that having more and more data will in general improve the quality of the estimation of intensive variables but not that of extensive variables. This is, of course, due to the fact that as the amount of data grows, so does the number of extensive variables to be estimated. In that sense it is for extensive variables a tie race between known and unknown entities. Therefore one often, but not necessarily, seeks to integrate out the extensive variables leaving only the parameters of the prior, which in general are intensive variables.

The idea of a hidden variable is an appealing concept when dealing with blind source separation problems, in which each observed data vector $\mathbf{x}_t$ for every time step is a function of some subset of the hidden variables. In simple cases, each data vector $\mathbf{x}_t$ is a function of a column $\mathbf{s}_t$ of the hidden variable matrix. In that picture, the hidden variables are something which for every time step are being transformed, using the model parameters $\boldsymbol{\theta}$, into an observed data vector $\mathbf{x}_t$. But in the most general formulation with additive noise, all time steps of the hidden variables may affect the observation $\mathbf{x}_t$, that is,

$$\mathbf{x}_t = \mathbf{f}_{\theta,t}(\mathbf{S}) + \mathbf{n}_t, \qquad t = 1, ..., N\,. \tag{2.5}$$

Both the noise, the transformation parameters and the hidden variables can be regarded as realizations of stochastic processes each with some probability $p_n(\mathbf{n}|\boldsymbol{\phi}_n)$, $p_\theta(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)$ and $p_s(\mathbf{S}|\boldsymbol{\phi}_s)$. But since the noise realization is different for every time step t, the noise is, like the hidden variable, an extensive variable. Integrating out the noise realization one obtains

$$\begin{aligned} p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \boldsymbol{\phi}_n) &= \int p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \mathbf{n}) p_n(\mathbf{n}|\boldsymbol{\phi}_n) d\mathbf{n} \\ &= \int \delta(\mathbf{n} - (\mathbf{X} - \mathbf{f}_\theta(\mathbf{S}))) p_n(\mathbf{n}|\boldsymbol{\phi}_n) d\mathbf{n} \\ &= p_n(\mathbf{X} - \mathbf{f}_\theta(\mathbf{S})|\boldsymbol{\phi}_n)\,. \end{aligned} \tag{2.6}$$

This conditioned probability $p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \boldsymbol{\phi}_n)$ is the cornerstone for the rest of the chapter: From this basic likelihood, all the probabilistic estimation schemes for hidden variables models are derived – a special status which stems from the choice that the noise realization will always be assumed integrated out.

As discussed earlier, one can either use the likelihood or the posterior and integrate out the unknown entities at different levels – see Table 2.1 for an overview. The notation with the roman numbers indicates to which level the remainder of the unknown entities are integrated out: on level I-I, neither the transformation parameters $\boldsymbol{\theta}$ nor the hidden variables $\mathbf{S}$ are integrated out. One is simply estimating these values directly and as point estimates. On level II-I, the transformation parameters $\boldsymbol{\theta}$ are integrated out, leaving the hyper-parameters $\boldsymbol{\phi}_\theta$ and the hidden variables for estimation. On level I-II it is the other way around, i.e. the hidden variables are integrated out leaving the hyper parameters $\boldsymbol{\phi}_s$ and on level II-II both the transformation parameters and the hidden variables are integrated out. As the dots in the table indicates, one could keep on integrating the hyper-parameters and thus reaching level III-II or whatever would seem fit for the task, but these levels are not considered in this thesis.

A small comment on the more technical side is that one can simplify the computations involved since the logarithm is a strictly increasing function, such

	Est. of $\boldsymbol{\theta}$	Est. of $\mathbf{S}$	ML	MAP
I-I	Directly	Directly	$p(\mathbf{X}\vert\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n)$	$p(\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n\vert\mathbf{X})$
II-I	Integrated out	Directly	$p(\mathbf{X}\vert\boldsymbol{\phi}_\theta,\mathbf{S},\boldsymbol{\phi}_n)$	$p(\boldsymbol{\phi}_\theta,\mathbf{S},\boldsymbol{\phi}_n\vert\mathbf{X})$
I-II	Directly	Integrated out	$p(\mathbf{X}\vert\boldsymbol{\theta},\boldsymbol{\phi}_s,\boldsymbol{\phi}_n)$	$p(\boldsymbol{\theta},\boldsymbol{\phi}_s,\boldsymbol{\phi}_n\vert\mathbf{X})$
II-II	Integrated out	Integrated out	$p(\mathbf{X}\vert\boldsymbol{\phi}_\theta,\boldsymbol{\phi}_s,\boldsymbol{\phi}_n)$	$p(\boldsymbol{\phi}_\theta,\boldsymbol{\phi}_s,\boldsymbol{\phi}_n\vert\mathbf{X})$
...				

Table 2.1: Different approaches to probabilistic estimation in hidden variable models. The model parameters $\boldsymbol{\theta}$ and the hidden variables $\mathbf{S}$ can be integrated out or left for direct point estimation, leaving four combinations for both the maximum likelihood (ML) and the maximum a posteriori (MAP) cost function. The symbols $\boldsymbol{\phi}_\theta$ and $\boldsymbol{\phi}_s$ are parameters of the parameter prior and prior on the hidden variables, i.e. they are so-called hyper-parameters.

that instead of maximizing the density directly, one can maximize the logarithm of the density. This holds of course for any cost function and thus for all the cost functions presented in Table 2.1.

2.1.3.1 ML and MAP on Level I-I

In a setup with no integration of the variables, one would regard the hidden variables as a sort of parameters and therefore seek to maximize $p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta},\boldsymbol{\phi}_n)$ with respect to $\boldsymbol{\theta}$, $\mathbf{S}$ and $\boldsymbol{\phi}_n$. Optimizing the cost function $\ln p(\mathbf{X}|\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n)$ involves computing the derivative and setting it to zero, e.g. for $\boldsymbol{\theta}$

$$\frac{\partial \ln p(\mathbf{X}|\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n)}{\partial \boldsymbol{\theta}} = \mathbf{0}$$

The ML cost function $\ln p(\mathbf{X}|\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n)$ is determined by Equation 2.6, and therefore one needs to define a specific noise model and transformation function $\mathbf{f}_{\theta,t}$ in order to get anything more concrete. The same holds for the MAP estimate maximizing $\ln p(\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n|\mathbf{X})$, which using Bayes' theorem is equal to

$$\mathbf{0} = \frac{\partial \ln p(\boldsymbol{\theta},\mathbf{S},\boldsymbol{\phi}_n|\mathbf{X})}{\partial \boldsymbol{\theta}} = \frac{\partial \ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta},\boldsymbol{\phi}_n)}{\partial \boldsymbol{\theta}} + \frac{\partial \ln p(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)}{\partial \boldsymbol{\theta}}$$

and a similar result when differentiating with respect to $\mathbf{S}$ and $\boldsymbol{\phi}_n$. From this equation it follows that without more specific choices for the noise, the hidden variables and the transformation function, one cannot get more specific about the solutions of the ML and MAP on level I-I, i.e. for the direct point-estimation approach.

2.1.3.2 ML and MAP on Level II-I

When integrating out either the model parameters or the hidden variables, one can say slightly more without specifying the rest of the model in details. On level

II-I the model parameters $\boldsymbol{\theta}$ are integrated out leaving the hyper parameters $\boldsymbol{\phi}_\theta$ of the parameter prior for estimation. Differentiating the ML cost function, one get the following

$$\begin{aligned}
\frac{\partial \ln p(\mathbf{X}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n)}{\partial \boldsymbol{\phi}_\theta} &= \frac{\partial}{\partial \boldsymbol{\phi}_\theta} \ln \int p(\mathbf{X}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n) d\boldsymbol{\theta} \\
&= \frac{1}{p(\mathbf{X}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n)} \int \frac{\partial}{\partial \boldsymbol{\phi}_\theta} p(\mathbf{X}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n) d\boldsymbol{\theta} \\
&= \int \frac{p(\mathbf{X}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \mathbf{S})}{p(\mathbf{X}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n)} \frac{\partial}{\partial \boldsymbol{\phi}_\theta} \ln p(\mathbf{X}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n) d\boldsymbol{\theta} \\
&= \left\langle \frac{\partial}{\partial \boldsymbol{\phi}_\theta} \ln p(\mathbf{X}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n) \right\rangle_{p(\boldsymbol{\theta}|\mathbf{X}, \boldsymbol{\phi}_\theta, \mathbf{S}, \boldsymbol{\phi}_n)} . \qquad (2.7)
\end{aligned}$$

Note how general this derivation is: nothing more than Bayes Theorem and the properties of the logarithmic function are used. One therefore ends up with an equation of the same general structure when differentiating with respect to the hidden variables. As usual, the MAP estimate involves the same calculations as the ML, but now adding a prior on the transformation and noise hyper parameters and the hidden variables.

2.1.3.3 ML and MAP on Level I-II

Not surprisingly, level I-II resembles level II-I quite a bit. Within the machine learning community, though, they have received very different amount of attention. This is most likely because it is more sensible to integrate out the extensive variables for which the estimate does not improve when the amount of observed data becomes larger. Similar to level II-I, the derivative of the likelihood is

$$\frac{\partial \ln p(\mathbf{X}|\boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}{\partial \boldsymbol{\theta}} = \left\langle \frac{\partial \ln p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}{\partial \boldsymbol{\theta}} \right\rangle_{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)} \qquad (2.8)$$

and correspondingly for the derivative with respect to $\boldsymbol{\phi}_s$ and $\boldsymbol{\phi}_n$. For additive noise models it is possible to rewrite the joint probability in the data and hidden variables in the following way using Bayes theorem

$$\ln p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) = \ln p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \boldsymbol{\phi}_n) + \ln p(\mathbf{S}|\boldsymbol{\phi}_s) .$$

This simplifies matters of the differentiation by splitting the density into an observation model given by the noise and the transformation function, and a parameter prior.

2.1.3.4 ML and MAP on Level II-II

If one is having prior knowledge of both the transformation parameters and the hidden variables, but not prior knowledge of the parameters of the next level, then level II-II is the most appropriate choice of approach. In that case the likelihood is a function of the prior parameters $\boldsymbol{\phi}_\theta$, $\boldsymbol{\phi}_s$ and $\boldsymbol{\phi}_n$. If one further assume that the prior on the transformation and noise is independent of the prior of the hidden variables, then the task is to maximize the following expression

$$\begin{aligned}
\ln p(\mathbf{X}|\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) &= \ln \int p(\mathbf{X}, \mathbf{S}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) d\mathbf{S} d\boldsymbol{\theta} \\
&= \ln \int p(\mathbf{X}|\mathbf{S}, \boldsymbol{\theta}, \boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) p(\mathbf{S}, \boldsymbol{\theta}|\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) d\mathbf{S} d\boldsymbol{\theta} \\
&= \ln \int p(\mathbf{X}|\mathbf{S}, \boldsymbol{\theta}, \boldsymbol{\phi}_n) p(\mathbf{S}|\boldsymbol{\phi}_s) p(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta) d\mathbf{S} d\boldsymbol{\theta} \,. \qquad (2.9)
\end{aligned}$$

Note that as before the observation density has been assumed not to depend on the prior parameters. Differentiating with respect to the prior parameters yields the following expressions

$$\begin{aligned}
\frac{\partial \ln p(\mathbf{X}|\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}{\partial \boldsymbol{\phi}_\theta} &= \left\langle \frac{\partial \ln p(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)}{\partial \boldsymbol{\phi}_\theta} \right\rangle_{p(\mathbf{S}, \boldsymbol{\theta}|\mathbf{X}, \boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)} \\
\frac{\partial \ln p(\mathbf{X}|\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}{\partial \boldsymbol{\phi}_s} &= \left\langle \frac{\partial \ln p(\mathbf{S}|\boldsymbol{\phi}_s)}{\partial \boldsymbol{\phi}_s} \right\rangle_{p(\mathbf{S}, \boldsymbol{\theta}|\mathbf{X}, \boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}
\end{aligned}$$

Using the logarithm and the assumption that the priors are independent, one obtains for the MAP estimate some minor changes to the equations – changes which are very similar to those of the level I-II approach.

All the approaches presented above have at least one thing in common: in order to be more concrete the model must be specified in more detail. Specifying the model includes choosing a distribution of the additive noise, choosing a transformation function, and a distribution over the hidden variables. The remaining sections of the chapter are dedicated to that: presenting more specific models and to some extend ways to deal with them.

2.2 Noise Model

This section describes details on the noise model. Describing a noise model involves choosing a distribution for the noise variable and showing how to estimate the parameters of this distribution.

2.2.1 Gaussian Noise

The most common noise model, which for many applications is also the most reasonable model, is the zero-mean gaussian assumption, i.e. $\mathbf{n}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{W})$. That is, for this particular noise model, the hyper parameter $\boldsymbol{\phi}_n$ is simply $\boldsymbol{\phi}_n = \mathbf{W}$. With gaussian noise, one can use Equation 2.6 to rewrite the density of the observations since these must then be gaussian with mean values determined by the transformation

$$p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta},\boldsymbol{\phi}_n) = \frac{1}{\sqrt{|2\pi\mathbf{W}|}^N} \exp\left[-\tfrac{1}{2}\sum_{t=1}^{N}(\mathbf{x}_t-\mathbf{f}_t)^T\mathbf{W}^{-1}(\mathbf{x}_t-\mathbf{f}_t)\right]$$

In order to estimate the covariance matrix $\mathbf{W}$ one needs to differentiate the logarithm of the observation density, $\ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta},\boldsymbol{\phi}_n)$, with respect to the noise covariance. Note, however, that a covariance matrix needs to fulfil certain criteria in order to serve as a proper covariance matrix, namely the criteria of being positive definite, and this influences how it should be differentiated. For $\mathbf{W}$ to be a positive definite matrix it must fulfil $\mathbf{x}^T\mathbf{W}\mathbf{x} > 0$ for all $\mathbf{x}$, a constraint which in practice is rather hard to implement directly. Fortunately, differentiating unconstrained and setting the derivative equal to zero one obtains what is in fact the maximum likelihood estimate

$$\begin{aligned} &\frac{\partial}{\partial\mathbf{W}}\ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta},\boldsymbol{\phi}_n) \\ =\; &\frac{\partial}{\partial\mathbf{W}}\left[\ln\frac{1}{\sqrt{|2\pi\mathbf{W}|}^N} - \frac{1}{2}\mathrm{Tr}[(\mathbf{X}-\mathbf{f})^T\mathbf{W}^{-1}(\mathbf{X}-\mathbf{f})]\right] \\ =\; &-\tfrac{N}{2}\mathbf{W}^{-T} + \tfrac{1}{2}\mathbf{W}^{-T}(\mathbf{X}-\mathbf{f})(\mathbf{X}-\mathbf{f})^T\mathbf{W}^{-T} \end{aligned}$$

which gives

$$\mathbf{W} = \tfrac{1}{N}\langle(\mathbf{X}-\mathbf{f})(\mathbf{X}-\mathbf{f})^T\rangle \tag{2.10}$$

where $\langle\cdot\rangle$ denotes average with respect to the source posterior. Note about this estimate that it ensures the solution is positive definite, that it depends on the moments $\langle\mathbf{f}_t\rangle$ and $\langle\mathbf{f}_t\mathbf{f}_t^T\rangle$ and that these moments can be more or less complicated depending of course on the transformation function $\mathbf{f}$.

2.2.2 Non-Gaussian Noise

The gaussian noise is in many cases easily justifiable because of the central limit theorem, but one could indeed imagine having noise which is not gaussian. This would be the case if for some application the hidden variables and observations were forced to be negative or positive or if one was modeling not just a measurement error but also some biased error introduced by a slightly wrong choice of model. In this thesis, however, only gaussian noise is considered.

2.3 Transformation Model

This section describes details on the transformation of the hidden variables as well as the equations for estimation of the transformation parameters. If the transformation parameters are to be estimated by the posterior or integrated out, then a prior on the parameters must be chosen. Which choice of prior is appropriate depends of course on the choice of transformation and is therefore briefly discussed after each presented trasformation function.

2.3.1 General Instantaneous Model

The general instantaneous observation model is, although being general, still sufficiently specific to point out crucial concepts and important relations. The model is

$$\mathbf{x}_t = \mathbf{f}_\theta(\mathbf{s}_t) + \mathbf{n}_t, \qquad t = 1, ..., N\,, \tag{2.11}$$

that is, the observed data vectors $\mathbf{x}_t$ are the result of a possibly non-linear function $\mathbf{f}_\theta : \mathbb{R}^d \to \mathbb{R}^m$ and some additive noise. The subscript $\boldsymbol{\theta}$ indicates that the function $\mathbf{f}$ depends on the model parameters $\boldsymbol{\theta}$.

2.3.1.1 ML on Level I-II

According to Equation 2.7 one needs to derive an expression for the joint distribution over observed data and hidden variables. Using Bayes's rule, the joint distribution can be written as

$$p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n) = p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \boldsymbol{\phi}_n)p(\mathbf{S}|\boldsymbol{\phi}_s)\,.$$

Again, the parameters are split into convenient groups of parameters: the parameters of the transformation function $\boldsymbol{\theta}$, the (hyper) parameters of the noise model $\boldsymbol{\phi}_n$ and the (hyper) parameters of the source prior $\boldsymbol{\phi}_s$. The benefit of factorizing the joint distribution like this is that one can separate the noisy observation model $p(\mathbf{X}|\mathbf{S}, \boldsymbol{\theta}, \boldsymbol{\phi}_n)$ from the source prior $p(\mathbf{S}|\boldsymbol{\phi}_s)$.

Following the general expression in Equation 2.7 and assuming a zero-mean gaussian noise model, the following entity needs to be calculated

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\theta}} \ln p(\mathbf{X}|\boldsymbol{\theta}, \mathbf{S}, \boldsymbol{\phi}_n) &= \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}} \sum_{t=1}^{N} (\mathbf{x}_t - \mathbf{f}(\mathbf{s}_t))^T \mathbf{W}^{-1} (\mathbf{x}_t - \mathbf{f}(\mathbf{s}_t)) \\
&= -\sum_{t=1}^{N} \frac{\partial \mathbf{f}(\mathbf{s}_t)}{\partial \boldsymbol{\theta}^T} \mathbf{W}^{-1} (\mathbf{x}_t - \mathbf{f}(\mathbf{s}_t))\,.
\end{aligned}$$

Thus for a gaussian noise model, the crucial moments for estimating the transformation parameters $\boldsymbol{\theta}$ are

$$\left\langle \frac{\partial \mathbf{f}(\mathbf{s}_t)}{\partial \boldsymbol{\theta}^T} \right\rangle_{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)} \qquad \text{and} \qquad \left\langle \frac{\partial \mathbf{f}(\mathbf{s}_t)}{\partial \boldsymbol{\theta}^T} \mathbf{W}^{-1} \mathbf{f}(\mathbf{s}_t) \right\rangle_{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}_s, \boldsymbol{\phi}_n)}$$

If the notation is abbreviated such that the observation function is given by $\mathbf{f}_t$ and the derivative by $\mathbf{f}'_t$, then the statistics necessary are

$$\langle \mathbf{f}_t \rangle \quad , \quad \langle \mathbf{f}_t \mathbf{f}_t^T \rangle \quad , \quad \langle \mathbf{f}'_t \rangle \quad \text{and} \quad \langle \mathbf{f}'^T_t \mathbf{W}^{-1} \mathbf{f}_t \rangle\,.$$

By 'necessary' is meant necessary to estimate the noise and model parameters. The averages themselves depend on the noise and model parameters and thus constitute a set of non-linear fix point equations. The general instantaneous transformation is too general, however, to make any particular choice of parameter prior the best choice.

2.3.2 Linear Instantaneous Model

The next model considered is the linear instantaneous model – a special case of the general instantaneous model. In the linear instantaneous model, the observation function is linear and therefore it is possible to represent it as a matrix multiplication

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{n}_t, \qquad t = 1, ..., N\,. \tag{2.12}$$

As with the general instantaneous model the observation columns vectors can be concatenated side by side into a larger observed data matrix $\mathbf{X}$. The same can be done with the unobserved column vectors $\mathbf{s}_t$, that is, collecting these into one larger matrix $\mathbf{S}$. Since the observation function is linear and represented by the matrix $\mathbf{A}$, the observation model can be written for all time steps in one expression

$$\mathbf{X} = \mathbf{A}\mathbf{S} + \mathbf{n}\,,$$

where the noise vectors $\mathbf{n}_t$ are stacked into a noise matrix $\mathbf{n}$. The linear observation model is interesting for several reasons. First of all it is the simplest non-trivial function one can construct, since making a Taylor expansion of the observation function to first order will yield exactly the linear model. Secondly the linear instantaneous model is a very versatile model which appears in both blind source separation models such as Independent Component Analysis (ICA) and different approaches for sparse representation. In Chapter 4 a further discussion of the interpretations is presented, but for now, the linear instantaneous model is simply formulated and it is demonstrated what is needed to estimate the model parameters.

The log likelihood is now given by $\mathcal{L} = \ln p(\mathbf{X}|\mathbf{A}, \mathbf{W}, \boldsymbol{\phi}_s)$ and following the notation from Equation 2.7 one can, due to the fact that the observation function is linear, derive the relevant expressions by

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{A}} &= \left\langle \frac{\partial}{\partial \mathbf{A}} \ln p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \mathbf{W}, \boldsymbol{\phi}_s) \right\rangle_{p(\mathbf{S}|\mathbf{X},\mathbf{A},\mathbf{W},\boldsymbol{\phi}_s)} \\
&= \left\langle \frac{\partial}{\partial \mathbf{A}} \sum_{t=1}^{N} \tfrac{-1}{2} (\mathbf{x}_t - \mathbf{A}\mathbf{s}_t)^T \mathbf{W}^{-1} (\mathbf{x}_t - \mathbf{A}\mathbf{s}_t) \right\rangle \\
&= \left\langle \sum_{t=1}^{N} \mathbf{W}^{-1} (\mathbf{x}_t - \mathbf{A}\mathbf{s}_t) \mathbf{s}_t^T \right\rangle \\
&= \mathbf{W}^{-1}\mathbf{X}\langle\mathbf{S}\rangle - \mathbf{W}^{-1}\mathbf{A}\langle\mathbf{S}\mathbf{S}^T\rangle\,.
\end{aligned}$$

Setting this derivative equal to zero yields seemingly an estimate of the mixing matrix

$$\mathbf{A} = \mathbf{X}\langle\mathbf{S}\rangle^T\langle\mathbf{S}\mathbf{S}^T\rangle^{-1}\,. \tag{2.13}$$

Note, however, that the source posterior distribution $p(\mathbf{S}|\mathbf{X}, \mathbf{A}, \mathbf{W}, \boldsymbol{\phi}_s)$, over which the average $\langle\cdot\rangle$ is taken, also depends on $\mathbf{A}$ such that the equation above does not express an isolation of the mixing matrix. Rather, Equation 2.13 is a non-linear constraint on the mixing matrix, which e.g. can be implemented in a fixed point algorithm. With the linear observation model and gaussian noise, the expression for the maximum likelihood estimate of the noise covariance becomes

$$\begin{aligned}\mathbf{W} &= \frac{1}{N}\langle(\mathbf{X}-\mathbf{A}\mathbf{S})(\mathbf{X}-\mathbf{A}\mathbf{S})^T\rangle \\ &= \frac{1}{N}\left(\mathbf{X}\mathbf{X}^T + \mathbf{A}\langle\mathbf{S}\mathbf{S}^T\rangle\mathbf{A}^T - \mathbf{X}\langle\mathbf{S}\rangle^T\mathbf{A}^T - \mathbf{A}\langle\mathbf{S}\rangle\mathbf{X}^T\right)\,. \end{aligned}\tag{2.14}$$

Note, however, that as with the mixing matrix, the simplicity of these equations are somewhat deceptive since the source posterior average denoted $\langle\cdot\rangle$ also depends on $\mathbf{W}$. A nice feature of the linear model as opposed to the general instantaneous model is that the posterior statistics needed in the expression of the mixing matrix and the noise covariance are the same, namely the first and second moments $\langle\mathbf{S}\rangle$ and $\langle\mathbf{S}\mathbf{S}^T\rangle$ of the hidden variables.

For MAP and higher level estimation, one must specify a parameter prior on the matrix $\mathbf{A}$. As a prior on the mixing matrix the following simple gaussian turns out to be useful

$$p(\mathbf{A}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}_A|}}\exp\left[-\frac{1}{2}\mathrm{Tr}(\mathbf{A}^T\boldsymbol{\Sigma}_A^{-1}\mathbf{A})\right]\,.$$

Note that when $\boldsymbol{\Sigma}_A$ is diagonal, which is often the most reasonable, this is a completely factorized distribution in the elements A_{ij} of the mixing matrix.

2.3.3 Convolutive Model

The last transformation model considered in this thesis is the convolutive model. It is the only model of the three which is mixing different time steps in the observation and also a popular and realistic model for many important applications. The basic observation model is

$$\mathbf{x}_t = \mathbf{f}_t + \mathbf{n}_t = \sum_{\tau=0}^{F-1}\mathbf{A}^\tau\mathbf{s}_{t-\tau} + \mathbf{n}_t, \qquad t = 1, ..., N \tag{2.15}$$

where $\mathbf{A}^\tau$ denotes a different matrix for every different τ, and not powers or transpose as the notation also could have indicated. The F matrices $\mathbf{A}^\tau$ are each $m \times d$ dimensional as the mixing matrix in the instantaneous model. The elements A^τ_{ij} for $\tau = 0, ..., F-1$ can be interpreted as a function which describes how the $j.th$ component of the hidden variables (in the BSS interpretation the $j.th$ source), is influencing the $i.th$ measurement.

Assuming a zero-mean gaussian noise model, the entities of the model to be estimated are the matrices $\mathbf{A}^\tau$, the noise covariance $\mathbf{W}$ and the hidden variable prior parameters $\boldsymbol{\phi}_s$. To ease the notation, it will turn out to be useful to define a very large matrix $\underline{\mathbf{A}}$ which consists of *all* the matrices $\mathbf{A}^\tau$ stacked side by side

$$\underline{\mathbf{A}} = \left[\begin{array}{cccc} \mathbf{A}^0 & \mathbf{A}^1 & \dots & \mathbf{A}^{F-1} \end{array} \right] . \tag{2.16}$$

The likelihood $\ln p(\mathbf{X}|\underline{\mathbf{A}}, \mathbf{W}, \boldsymbol{\phi}_s)$ is as always the crucial part of both the maximum likelihood and the maximum a posteriori approach and therefore emphasis is on the maximization of the likelihood. Differentiating it with respect to $\underline{\mathbf{A}}$ demands a corresponding extension of the notation for the hidden variables. Let $\underline{\mathbf{s}}_t$ denote the column vector which is $dF \times 1$ dimensional and is defined by

$$\underline{\mathbf{s}}_t = \left[\begin{array}{cccc} \mathbf{s}_t^T & \mathbf{s}_{t-1}^T & \dots & \mathbf{s}_{t-(F-1)}^T \end{array} \right]^T .$$

Doing that the observation model becomes $\mathbf{x}_t = \underline{\mathbf{A}}\underline{\mathbf{s}}_t + \mathbf{n}_t$ and thus symbolically equivalent to the instantaneous linear model. Differentiation of the likelihood with respect to the parameters $\underline{\mathbf{A}}$ and $\mathbf{W}$ is therefore giving a similar result. If one by $\underline{\mathbf{S}}$ denotes the concatenation of all the vectors $\underline{\mathbf{s}}_t$ then the results are

$$\begin{aligned} \underline{\mathbf{A}} &= \mathbf{X}\langle\underline{\mathbf{S}}\rangle\langle\underline{\mathbf{S}}\underline{\mathbf{S}}^T\rangle^{-1} && (2.17) \\ \mathbf{W} &= \tfrac{1}{N}\langle(\mathbf{X} - \underline{\mathbf{A}}\underline{\mathbf{S}})(\mathbf{X} - \underline{\mathbf{A}}\underline{\mathbf{S}})^T\rangle . && (2.18) \end{aligned}$$

As with the instantaneous linear model, the posterior statistics needed for the transformation function and the noise parameters are the same, namely $\langle\underline{\mathbf{S}}\rangle$ and $\langle\underline{\mathbf{S}}\underline{\mathbf{S}}^T\rangle$. Special attention must now be given when dealing with these moments, since two different vectors $\underline{\mathbf{s}}_t$ and $\underline{\mathbf{s}}_{t'}$ contains the same values if the difference between the times t and t' are smaller than the filter length F. For that reason it may be more appealing to do a single element-wise differentiation, which can be expressed by defining the following matrices (see Section B.0.1 for calculational details),

$$\underline{\mathbf{F}}_* = \left[\begin{array}{cccc} \mathbf{F}_*^0 & \mathbf{F}_*^1 & \dots & \mathbf{F}_*^{F-1} \end{array} \right] \tag{2.19}$$

$$(F_*)_{ij}^\tau = \sum_{t=1}^{N} X_{it}\langle S_{j,t-\tau}\rangle$$

and

$$\underline{\mathbf{G}}_* = \left[\begin{array}{cccc} \mathbf{G}_*^{0,0} & \mathbf{G}_*^{0,1} & \dots & \mathbf{G}_*^{0,F-1} \\ \mathbf{G}_*^{1,0} & \mathbf{G}_*^{1,1} & \dots & \mathbf{G}_*^{1,F-1} \\ \vdots & & & \\ \mathbf{G}^{F-1,0} & \mathbf{G}^{F-1,1} & \dots & \mathbf{G}_*^{F-1,F-1} \end{array} \right] \tag{2.20}$$

$$(G_*)_{ij}^{\tau,\tau'} = \sum_{t=1}^{N}\langle S_{i,t-\tau}S_{j,t-\tau'}\rangle$$

Doing that, the notation of the underlining is not necessary, but the correlations induced by the convolutive transformation function are fundamental and cannot be avoided. In order to make this chapter notational comparable with later chapters, a star has been introduced in the subscript of the matrices. It indicates that the posterior averages are calculated exactly as opposed to approximations introduced later in this thesis. In this notation, one obtains the following equations when setting the derivative of log likelihood equal to zero

$$\underline{\mathbf{A}} = \underline{\mathbf{F}}_* \underline{\mathbf{G}}_*^{-1} \tag{2.21}$$

$$\mathbf{W} = \frac{1}{N}\Big[\mathbf{X}\mathbf{X}^T - \Big(\underline{\mathbf{F}}_* \underline{\mathbf{A}}^T + \underline{\mathbf{A}}\underline{\mathbf{F}}_*^T\Big) + \underline{\mathbf{A}}\underline{\mathbf{G}}_*\underline{\mathbf{A}}^T\Big]\,. \tag{2.22}$$

The seemingly close relation between the instantaneous linear model and the convolutive model is both genuine and deceptive: The convolutive model having a much richer statistical structure is considerably harder to estimate, but in certain limits the relation reveals profound similarities visible as essentially identical properties.

In some applications, the matrix entries A^{τ}_{ij} have an important meaningful interpretation, for example as filters in audio settings. This makes the choice of prior for the entries more complicated and indeed application dependent and it is therefore not included in this chapter.

2.4 Hidden Variable Models

This section describes details on the prior of the hidden variables and the model parameters. Describing the prior models involves choosing a distribution and in principle showing how to estimate the parameters of this distribution. Because of the rather complicated priors needed for the applications included later on in the thesis, the equations for parameter estimation are not included here, but instead postponed for later chapters in the extend they are needed.

2.4.1 Basic Maximum Likelihood

Using maximum likelihood on the hidden variables one is attempting to estimate the hidden variables directly. Differentiating the log likelihood with respect to the hidden variables, yields for the general instantaneous transformation

$$\mathbf{f}'(\mathbf{s}_t)^T \mathbf{W}^{-1}(\mathbf{x}_t - \mathbf{f}(\mathbf{s}_t)) = \mathbf{0}$$

In the linear instantaneous model, this expression simplifies even further and the optimal hidden variables can be isolated giving the result

$$\mathbf{A}^T\mathbf{W}^{-1}\mathbf{A}\mathbf{S} = \mathbf{A}^T\mathbf{W}^{-1}\mathbf{X}$$

which not surprisingly simplifies into $\mathbf{S} = \mathbf{A}^{-1}\mathbf{X}$ when $\mathbf{A}$ is square. When $\mathbf{A}$ is overdetermined, i.e. has more rows than columns, this can be regarded as a generalized pseudo-inverse – generalized in the sense that the metric $\mathbf{W}$ is used instead of the usual $\mathbf{I}$ in $\mathbf{A}^T\mathbf{A}\mathbf{S} = \mathbf{A}^T\mathbf{X}$. When $\mathbf{W}$ has the structure $\mathbf{W} = \sigma^2\mathbf{I}$ then the equation is degenerating into the standard form. Basic maximum likelihood for estimation of hidden variables turns out in many cases to be inadequate for two reasons: the hidden variables are extensive variables and the often important information in the prior is disregarded.

2.4.2 Estimation with Prior

The task is to choose some distribution $p(\mathbf{S}|\boldsymbol{\phi}_s)$ which is sensible from an analytical point of view and with respect to the particular application at hand. But before the prior models relevant for this thesis are presented, a general observation is worth noting: if the source prior is rewritten to be of the form $p(\mathbf{S}|\boldsymbol{\phi}_s) = \exp(\kappa(\mathbf{S}|\boldsymbol{\phi}_s))/Z$, and then setting the derivative of the log likelihood with respect to $\boldsymbol{\phi}_s$ equal to zero, one obtains

$$\left\langle \frac{\partial\kappa(\mathbf{S}|\boldsymbol{\phi}_s)}{\partial\boldsymbol{\phi}_s} \right\rangle_{p(\mathbf{S}|\boldsymbol{\phi}_s)} = \left\langle \frac{\partial\kappa(\mathbf{S}|\boldsymbol{\phi}_s)}{\partial\boldsymbol{\phi}_s} \right\rangle_{p(\mathbf{S}|\mathbf{X},\boldsymbol{\theta},\boldsymbol{\phi}_s)} . \tag{2.23}$$

Depending on the specific source prior considered, this analytically elegant equations may serve as estimation criteria.

Regarding the hidden variable priors presented below one major restriction has been imposed on almost all of them, namely that the elements of the matrix $\mathbf{S}$ are iid, i.e. independent and identical distributed. Doing that, the distribution $p(\mathbf{S}|\boldsymbol{\phi}_s)$ factorizes into dN one dimensional distributions. The reason for choosing in this way is mostly a computational convenience, in fact almost a necessity, utilized in later chapters, but the factorization in the first dimension $(1, ..., d)$ is also in the spirit of the Independent Component Analysis in which it is a basic assumption. The only exception to the complete factorization is the general mixture of gaussians, which does not in general factorize over the in first direction $(1, ..., d)$.

2.4.2.1 General Mixture of Gaussians

The mixture of gaussians (MoG) which is not subject to special restrictions is referred to as the general mixture of gaussians. For a time independent assumption, it is defined by the density

$$p(\mathbf{S}) = \prod_{t=1}^{N}\sum_{k=1}^{K} \frac{\rho_k}{\sqrt{|2\pi\mathbf{D}_k|}} \exp\left[-\tfrac{1}{2}(\mathbf{s}_t - \boldsymbol{\mu}_k)^T\mathbf{D}_k^{-1}(\mathbf{s}_t - \boldsymbol{\mu}_k)\right] ,$$

where the proportions ρ_k must be between 0 and 1 and sum to 1 and each $\mathbf{D}_k$ is a covariance matrix. There are no restrictions on the mean vectors $\boldsymbol{\mu}_k$. The general MoG is a very appealing model density due to its combination of flexibility and computational properties. The flexibility ensures that using sufficiently many components, i.e. choosing the numbers of mixture components K high, one can model a very large number of shapes from asymmetric skewed densities to heavy tailed sparse distributions.

2.4.2.2 Independent Mixture of Gaussians

The independent mixture of gaussians (IMoG), also known as a factorial mixture of gaussians, is a MoG which factorizes in the principal dimension $(1, ..., d)$. The density can be defined by

$$p(\mathbf{S}) = \prod_{it} \sum_{k} \frac{\rho_{ik}}{\sqrt{2\pi\sigma_{ik}^2}} \exp\Big[-\frac{1}{2} \frac{(s_{it} - \mu_{ik})^2}{\sigma_{ik}^2} \Big]$$

in which the usual constrains apply to the ratios and the variances. This is of course a special case of the general MoG, but the assumption of independence imposes constrains on the covariance matrices $\mathbf{D}_k$. These must now fulfil, in a certain sense, to have all combinations of the 1-dimensional variances; i.e. if one has d one dimensional K-component mixtures and want to combine these, then one ends up with a MoG which has K^d components. The IMoG will be described in more details in Chapter 5.

2.4.2.3 Exponential

The exponential distribution over the entire matrix $\mathbf{S}$ is simply the product over dN ordinary one dimensional exponential distributions. The density is

$$p(\mathbf{S}) = \Theta(\mathbf{S}) \left(\prod_{i} \nu_i^N \right) \exp\left[-\sum_{it} \nu_i S_{it} \right]$$

where $\Theta(\mathbf{S})$ denotes a product of heavy-side functions, each factor from an element S_{it} (the heavy-side function is the function which is 0 when the argument is zero or less and 1 otherwise). For the density to be normalizable, the parameters ν_i must all be positive.

2.4.2.4 Laplace

A natural extension of the exponential distribution to the entire real domain is the laplace distribution. It is simply the symmetric extension, which have the density defined by

$$p(\mathbf{S}) = \left[\frac{\eta}{2} \right]^{dN} \cdot \exp\left[-\eta \sum_{t=1}^{N} \sum_{i=1}^{d} |S_{it}| \right]$$

As for the exponential distribution, the parameter η must be positive. The laplace distribution is in some cases a nice practical choice when the hidden variables must be sparse and heavy tailed.

2.4.2.5 Binary

The binary prior, and in general a discrete prior on the hidden variables, is a product of dN ordinary one dimensional binary priors. It is defined by

$$p(\mathbf{S}) = \prod_{it} \Big(\eta\delta(S_{it} - 1) + (1 - \eta)\delta(S_{it} + 1) \Big)$$

where η must be between zero and one, $0 < \eta < 1$. The binary prior is important for application in especially the telecommunication and computer industries.

Summary of Chapter 2

In this chapter, probabilistic estimation in hidden variable models have been introduced. For models with additive noise, one has to choose a transformation function, priors, level of integration and whether to use maximum likelihood or maximum a posterior estimates – see Table 2.2 for an overview.

Table 2.2: Probabilistic Estimation in Hidden Variable Models. An overview of the steps involved and comments on choice for this thesis.

1. Choose a **transformation** function $\mathbf{f}_\theta$.
   In this thesis:
   - General instantaneous model
   - Linear Instantaneous model
   - Convolutive model

2. Choose **prior distributions** for the noise $p_n(\mathbf{n}|\boldsymbol{\phi}_n)$, transformation parameters $p_\theta(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)$, and the hidden variables $p_s(\mathbf{S}|\boldsymbol{\phi}_s)$.
   In this thesis:
   - The noise $p_n(\mathbf{n}|\boldsymbol{\phi}_n)$ is always gaussian
   - The transformation parameter $p_\theta(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)$ is gaussian if used
   - The hidden variables $p_s(\mathbf{S}|\boldsymbol{\phi}_s)$ are chosen among
     – Gaussian
     – Mixture of Gaussians
     – Independent Mixture of Gaussians
     – Laplace
     – Exponential
     – Uniform
     – Binary
     – ...

3. Choose **level of integration** (I, II, ...) for the noise, transformation parameters and the hidden variables.
   In this thesis:
   - The noise is always integrated out: Level II
   - The transformation parameters: Level I or II
   - The hidden variables are always integrated out: Level II

4. Choose whether to use **ML or MAP** estimates.
   In this thesis:
   - Mostly ML estimates are used.

5. Solve **relevant equations** – often non-linear, coupled equations regarded as fixed point equations.
   In this thesis:
   - The equations pose a serious computational challenge, which is discussed in the two next chapters.

Chapter 3

The EM Algorithm

In this chapter, the Expectation-Maximization (EM) algorithm is described and analyzed in detail. The first section introduces the algorithm, sketches its history and a formulation relevant for the Mean Field ICA. The second section presents an analysis of the convergence rate in the low noise limit, published in [41] and [43]. In the third section the so-called "Easy Gradient Recipe" (EGR) is described. EGR is a new variant among the maximum likelihood tools – a quasi-Newton optimization, based on the relatively simple calculations of the EM algorithm [34].

3.1 Introduction to EM

The EM algorithm is a method for obtaining maximum likelihood estimates under difficult conditions. In the situation, that the likelihood of some model is reasonably easy to calculate and differentiate, one can simply maximize the given function with standard methods and there is no need for the EM algorithm. But often it is not so. The models used in a particular setting can be very complicated or the values of some relevant variables unknown, i.e. missing data or so-called latent variables. In either case, the EM algorithm can be a practical approach or in complicated cases the only way forward.

3.1.1 The Dempster Formulation

What in this thesis is referred to as *The Dempster Formulation* is the EM algorithm as it was presented in [7]. The formulation by Dempster et al. is slightly different from the formulation of the EM algorithm used in Mean Field ICA, a formulation which was published in [32], but the Dempster formulation is included here for completeness and perspective.

Before going into details, a comment on notation will be appropriate. In most literature on the EM algorithm, the observed variables are denoted $\mathbf{y}$, the unobserved often as $\mathbf{z}$ and the complete data as $\mathbf{x}$. In this thesis, however, I will for consistency with the other chapters, denote the observed data $\mathbf{X}$, the unobserved data $\mathbf{S}$ and the so-called complete data $\mathbf{U}$. The complete data is the joint set of the observed and hidden variables and in some cases also the parameters. It is simply everything worth knowing in the model.

The situation the EM algorithm is dealing with is the following: assume a probabilistic model consisting of the data $\mathbf{U}$ and the parameters $\boldsymbol{\theta}$. To find a maximum likelihood estimate of the parameters, one needs to optimize $\ln p(\mathbf{U}|\boldsymbol{\theta})$, but the challenge is that not all the data $\mathbf{U}$ is at disposal - one can only observe a part of $\mathbf{U}$, namely $\mathbf{X}$. Since all that is known is $\mathbf{X}$, the best one can do in a maximum likelihood approach is to maximize $\ln p(\mathbf{X}|\boldsymbol{\theta})$ which is related to the model by

$$\ln p(\mathbf{X}|\boldsymbol{\theta}) = \ln \int_{\mathcal{U}(\mathbf{X})} p(\mathbf{U}|\boldsymbol{\theta}) d\mathbf{U}$$

where $\mathcal{U}(\mathbf{X})$ is the set of all data sets $\mathbf{U}$ which is consistent with the observation $\mathbf{X}$. The challenge is that while $p(\mathbf{U}|\boldsymbol{\theta})$ is often explicitly known having an analytical expression because the model has been constructed assuming full knowledge of all data, $p(\mathbf{X}|\boldsymbol{\theta})$ is not known due to the complicated nature of the integral. Thus, the restricted knowledge of the data makes the maximum likelihood estimation considerably more complicated. The EM algorithm defines a function $Q(\boldsymbol{\theta}', \boldsymbol{\theta})$ by

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = \int p(\mathbf{U}|\mathbf{X}, \boldsymbol{\theta}') \ln p(\mathbf{U}|\boldsymbol{\theta}) d\mathbf{U}$$

which, considered as the posterior mean of the complete log likelihood, can be interpreted as a sensible substitution for $\ln p(\mathbf{U}|\boldsymbol{\theta})$. The iterations of the EM algorithm begins with some random $\boldsymbol{\theta}^0$ followed by the repeated use of the two steps

E-step: Compute $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$ as a function of $\boldsymbol{\theta}$.

M-step: Find $\boldsymbol{\theta}^{(n+1)} = \text{argmax}_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$

This produces a sequence of parameter values $\{\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ...\}$ which has the very important property of the log likelihood $\mathcal{L}(\boldsymbol{\theta}) = \ln p(\mathbf{X}|\boldsymbol{\theta})$, that $\mathcal{L}(\boldsymbol{\theta}^n) \leq$

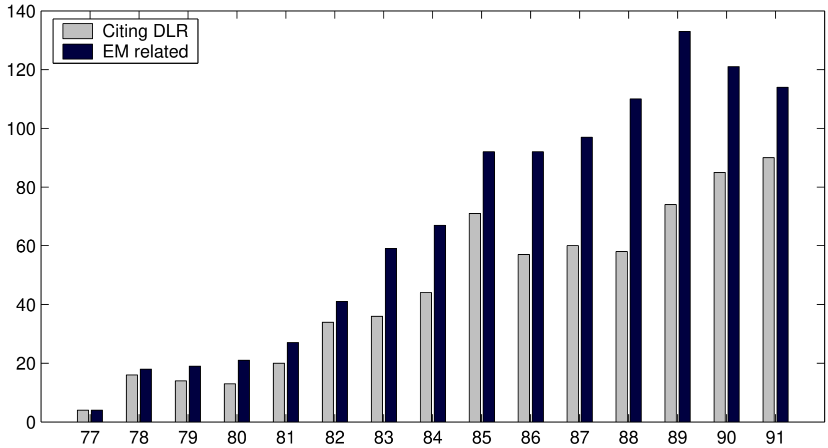


Figure 3.1: Publications relating to the EM algorithm. The impact of the paper of Demspter et al. is quite impressive as is the interest and practical application of the EM algorithm (from [28]).

$\mathcal{L}(\boldsymbol{\theta}^{(n+1)})$, i.e. that the likelihood of the observed data is non-decreasing in each step. Furthermore, if $\boldsymbol{\theta}^*$ is a maximum of the likelihood, then it is fix point of the algorithm, see [7] for details.

The paper [7] of Dempster et al. is of course much more elaborate having a number of theoretical details and generalizations. But the point of this section is just to introduce the reader to the basic concept. A more detailed discussion of the properties of the EM algorithm is to be found in later sections and a comprehensive account can be found in [26].

3.1.2 History of the EM Algorithm

In an attempt to recap the history of developments in a research subject such as the EM algorithm, one would of course like to be exhaustive. The year 1977 is a crucial moment for the EM algorithm, so one could start from there and work through the papers published later, categorize them suitably and write a comprehensive review. In fact, some of the main contributers set out to do exactly that and submitted during the process a short paper on their findings so far [28]. In this preliminary bibliographical review, they report a dramatic increase of papers related to the EM algorithm - an increase which from 1977 to the year 1991 showed no signs of slowing down, see Figure 3.1.

Although Meng and Pedlow in their paper from 1992 had high hopes for their future work, claiming that a comprehensive EM bibliography were around the corner, they seemingly have succumbed to the magnitude of the task. In

[29] by Meng and van Dyk, five years later, a remark in the prologue indicates what happened: "This section shares its title with Meng and Pedlow (1992) who conducted a bibliographical search of EM-related papers (...) This turned out to be an essentially hopeless task". Already in 1992 when Meng and Pedlow published their preliminary findings, there were more than 1000 papers relating to the EM algorithm – a truly daunting amount to summarize.

This should not refrain anyone from trying to organize some part of the EM-related literature into a summary, but rather warn the reader that any recap of the EM algorithm will have to leave out a substantial number of papers and thus be incomplete in some way or another.

3.1.2.1 Prehistory and the Paper of Dempster, Laird and Rubin

The problem of dating the birth of the EM algorithm is discussed in both the book [26] and the well written anniversary paper [29]. The latter have the slightly surprising title "The EM Algorithm - an old Folk-song Sung to a Fast New Tune". The reason for the unusual title is that the authors embrace the comparison of the EM algorithm with a folk song - a comparison originally made by Heartley, one of the commentators of the original Dempster paper from 1977. The similarities allegedly are, that as a folk song can change, develop and reappear in many years before someone record it, the EM algorithm has been around long before 1977 in all sorts of different guises. But it was the paper of Dempster, Laird and Rubin from 1977, that made it popular - or to stay within the music analogy - recorded it and made it a hit.

Ideas and solutions, which today is labeled as an EM algorithm, can be traced as far back as Fisher in 1925, and up through the 20th century, many authors have likewise written EM-related publications. Less than a decade before the Dempster paper, contributions like [2] and [51] were rather close to earning the place which eventually became Dempsters and his colleagues, but they did not have the generality, sufficient clarity and proof for some for the important properties such as strict increase of the likelihood. And then finally in 1977, Dempster, Laird and Rubin published the famous journal paper that according to [50] entered the all time top ten over most important statistics publications.

3.1.2.2 Convergence Issues

In the original paper of Dempster et al., one central property of the generated sequence of updated parameters is that the log likelihood of these values are always non-decreasing, that is $\mathcal{L}(\boldsymbol{\theta}^{(n)}) \leq \mathcal{L}(\boldsymbol{\theta}^{(n+1)})$. But it is not clear that this ensures the property of always finding a stationary point of the likelihood or that the sequence of parameters is converging. Questions like these were investigated in [53], and in [45] failed convergence results were reported - a claim fiercely debated by others.

But measured on the number of reported concerns, the importance of convergence turned out not to be the question if the algorithm converged to a local maxima or not, but how fast it converged. The issue of speed of convergence was already mentioned in [7] by Dempster et al., but perhaps most directly expressed in this quote from [44]:

> "The source for most serious practical problems associated with the algorithm is the speed of convergence (...), which can often be annoyingly or hopelessly slow."

The number of examples in which the EM algorithm converges slowly is quite impressive as is the number of suggestions on how to remedy the slowdown. Most of the reports on slow convergence are empirical in the sense that the EM algorithm has been applied to some problem and the authors have noticed that the simulations run for much longer than they anticipated or required more iterations than were practically possible. These reports include papers such as [44] a paper on mixture densities and [42] on EM in Independent Component Analysis.

But there are also analytical results explaining the bad convergence rate for a specific model or parameter setting. Among these is [13] who proved sublinear convergence for some cases of mixture of two gaussians, multivariate gaussians under certain conditions and some factor analysis cases. Also on low noise cases convergence is a problem which is described in [4] for an ICA problem and more general in [41]. These latter papers help to some extend the understanding of why the EM algorithm in many situations perform rather poorly, but the common picture drawn by the many conclusions is not a simple one. As documented in [41] and suggested by the discussion later in this chapter, the EM algorithm applied to a model with additive noise will perform bad in the low noise limit. This is somewhat surprising in the sense that in the low noise limit, the signal to noise ratio is good. On the other hand, results presented in e.g. [46] reports that it is in the cases of ill-posed problems or other inherently difficult tasks, that EM is performing bad. Most likely these two kind of situations in which EM fails to converge in reasonable time are not closely related. Rather, they point to the fact that the task of understanding the convergence properties of the EM algorithm in a completely general setting is challenging and the problem not yet fully understood.

3.1.2.3 Speeding Up Convergence

As there has been many reports on slow convergence, there has also been many suggested ways to speed the EM algorithm up. Since the goal of the EM algorithm is to find the maximum likelihood estimator, some of the proposed methods are using information and functions such as the log likelihood $\mathcal{L}(\boldsymbol{\theta})$ which are not strictly a part of the EM algorithm. But these functions are still a part of the EM heritage and interesting in their own right.

The simplest methods for accelerating the EM algorithm are the ones which increase the step length. The direction chosen by the EM algorithm is assumed correct or at least sufficiently accurate, and therefore the only problem is that the proposed new value of the EM algorithm $\boldsymbol{\theta}_{EM}^n$ is too close to the current parameter value $\boldsymbol{\theta}^n$. To remedy this, the step is increased by a factor η in some way or another, that is, $\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \eta(\boldsymbol{\theta}_{EM}^n - \boldsymbol{\theta}^n)$. One recent example of the step length approach is the Adaptive Overrelaxed EM, published in [46]. In [42] the Adaptive Overrelaxed EM is applied with a reasonable result, but [18] is summarizing the reports of several step length methods as 'only small gains' compared to more advanced accelerators. A more advanced method which is also using step length were presented in [17]. Here the direction of the EM algorithm is used in a search line approach on the log likelihood, that is, $\mathcal{L}(\boldsymbol{\theta} + \lambda\tilde{\mathbf{g}})$ where λ is chosen to maximize the log likelihood. Since this latter approach is using the log likelihood which is not a part of the standard EM algorithm, it is an example of what is sometimes referred to as a 'hybrid' model.

When describing the advanced methods, the notation in this thesis is following the excellent paper [18]. The mapping of a completed EM update (both an E and an M step) is denoted $\boldsymbol{\theta}^{n+1} = \mathbf{M}(\boldsymbol{\theta}^n)$ and a function $\tilde{\mathbf{g}}$ is defined $\tilde{\mathbf{g}}(\boldsymbol{\theta}) = \mathbf{M}(\boldsymbol{\theta}) - \boldsymbol{\theta}$. In this notation, fix points of the EM algorithm are equivalent to zeros of the function $\tilde{\mathbf{g}}$ which can intuitively be regarded as a gradient of some underlying cost function. A Newton approach is then to write the update $\Delta\boldsymbol{\theta}$ as

$$\Delta\boldsymbol{\theta} = -\mathbf{J}(\boldsymbol{\theta})^{-1}\tilde{\mathbf{g}}(\boldsymbol{\theta}), \qquad \text{where} \qquad \mathbf{J} = \mathbf{M}'(\boldsymbol{\theta}) - \mathbf{I} \tag{3.1}$$

but often $\mathbf{M}'(\boldsymbol{\theta})$ is complicated and needs to be approximated. In fact, most of the Newton-like EM accelerators are approximating $\mathbf{J}^{-1}$ or $\mathbf{M}'$ in some way or another.

The Aitken accelerator, first suggested in [7] but properly reformulated by [19], approximates $\mathbf{M}'$ using a sequence of parameter updates $\{\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, ..., \boldsymbol{\theta}^{p+1}\}$. Constructing a matrix $\mathbf{D}$ of the first p parameter vectors as columns, and $\tilde{\mathbf{D}}$ of the last p, one can utilize the approximation $\mathbf{M}' \cong \tilde{\mathbf{D}}\mathbf{D}^{-1}$. The difficulties are that these columns, due to the small step size of the standard EM algorithm, are very close to being linear dependent and therefore the inversion step may cause numerical problems. Furthermore it has been reported that it in some cases fail to converge at all.

The Louis method published in [24] is approximating $\mathbf{J}$. Motivated from the fact that in the optimal point $\hat{\boldsymbol{\theta}}$, it holds that $\mathbf{J}(\hat{\boldsymbol{\theta}}) = -Q''(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}})^{-1}\mathcal{L}''(\hat{\boldsymbol{\theta}})$, see [7], Louis is using this identity for all $\boldsymbol{\theta}$ as an approximation, that is, $\mathbf{J}(\boldsymbol{\theta}) \cong -Q''(\boldsymbol{\theta}, \boldsymbol{\theta})^{-1}\mathcal{L}''(\boldsymbol{\theta})$. Even if this approximation is good and the resulting method therfore efficient, the approach suffers severely from the fact that both Q'' and $\mathcal{L}''$ typically are hard to compute analytically. And if it is indeed possible to compute $\mathcal{L}''$, it is hard to see why one should not just use a true Newton method on the log likelihood.

The Conjugate Gradient EM published in [16] is based on the so-called generalized conjugated gradient algorithm. The key idea is to regard the EM step as a generalized gradient and then, as one can speed up gradient algorithms using conjugated gradients, they use generalized conjugated gradients to speed up the EM update. A generalized norm on the parameter space is defined by $||\boldsymbol{\theta}||^2 = \boldsymbol{\theta}^T \mathbf{C} \boldsymbol{\theta}$, where $\mathbf{C} = \mathbf{I}$ for ordinary 2-norm. Having some function of the parameters $f(\boldsymbol{\theta})$, the gradient of this function is dependent on the choice of matrix $\mathbf{C}$ in the following way: if $\mathbf{g}(\boldsymbol{\theta})$ is the gradient corresponding to the ordinary 2-norm, and $\tilde{\mathbf{g}}$ is the gradient corresponding to the norm using the matrix $\mathbf{C}$ then these are related to each other as $\tilde{\mathbf{g}}(\boldsymbol{\theta}) = \mathbf{C}^{-1}\mathbf{g}(\boldsymbol{\theta})$. In [16] it is demonstrated that close to the optimal solution $\hat{\boldsymbol{\theta}}$ it approximately holds that $\tilde{\mathbf{g}}(\boldsymbol{\theta}) \cong -Q''(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}})^{-1}\mathbf{g}(\boldsymbol{\theta})$. That is, the EM update is a generalized gradient of the log likelihood. A generalized conjugated gradient algorithm can then be applied – an algorithm which surprisingly does not involve the somewhat difficult matrix $-Q''(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}})^{-1}$, see [16].

The so-called LANGE method from [20] is a quasi Newton algorithm on the log likelihood. It is thus an approximation to the true Newton approach $\mathbf{d} = -\mathbf{J}_{\mathcal{L}}(\boldsymbol{\theta})^{-1}\mathbf{g}(\boldsymbol{\theta})$, where $\mathbf{d}$ is the direction in parameter space used for updating the parameter $\Delta\boldsymbol{\theta} = \lambda\mathbf{d}$ for some suitable λ. The Hessian $\mathbf{J}_{\mathcal{L}}$ is approximated $\mathbf{J}_{\mathcal{L}}(\boldsymbol{\theta}) \cong Q''(\boldsymbol{\theta}, \boldsymbol{\theta}) - \mathbf{B}$, for a matrix $\mathbf{B}$ which is zero in the beginning and must be updated through the process.

The methods denoted QN1 and QN2 are two different quasi Newton algorithms presented in [18]. QN1 is a so-called 'pure' quasi Newton approach in the sense that it is not assuming any extra knowledge than the EM update. The tricky inverse Hessian $\mathbf{J}^{-1}$ in Equation 3.1 is approximated using the Broyden update [6], starting as minus the identity matrix. The QN2 is a quasi Newton method on the log likelihood and the link to the EM algorithm is that the resulting update can be written using a corrected EM direction. The update direction is $\mathbf{d} = \tilde{\mathbf{g}}(\boldsymbol{\theta}) - \boldsymbol{\Psi}\mathbf{g}(\boldsymbol{\theta})$, where $\boldsymbol{\Psi}$ has the role as a correction to the approximated inverse Hessian of the log likelihood. The $\boldsymbol{\Psi}$ is updated using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) rule, see [5], and the parameter is updated using $\Delta\boldsymbol{\theta} = \lambda\mathbf{d}$, where λ is chosen to maximize $\mathcal{L}(\boldsymbol{\theta} + \lambda\mathbf{d})$.

Expectation Conjugate Gradient (ECG) presented in [47] is the last of the presented accelerations. It utilizes a very simple and yet powerful idea that using the formulation of [32] and computing the steps of the EM algorithm, one can obtain the log likelihood and the exact gradient of the log likelihood. These entities can then be applied in a conjugated gradient scheme to obtain faster convergence. Later in the chapter a similar approach called the 'Easy Gradient Approach' is derived in more details.

Table 3.1 is an overview of the described accelerators along with an indication of what entities must be calculated for each of them.

Method	*Published*	$\mathbf{M}$	$\mathbf{M}'$	Q''	$\mathcal{L}$	$\mathcal{L}'$	$\mathcal{L}''$	S.L.
Standard EM	Dem77	x	-	-	-	-	-	-
Aitken	Dem77	x	-	-	-	-	-	-
Louis	Lou82	x	-	x	-	-	x	-
Conj. Grad. EM	Jam93	x	-	-	x	x	-	x
Step length I	Jam94	x	-	-	x	-	-	x
LANGE	Lan95	-	-	x	-	x	-	x
QN1	Jam97	x	-	-	-	-	-	-
QN2	Jam97	x	-	x	-	x	-	x
Adap. Over. EM	Sal03	x	-	-	-	-	-	x
ECG	Sal03	x	-	-	x	-	-	x
Newton on EM	-	x	x	-	-	-	-	-
Newton on $\mathcal{L}$	-	-	-	-	x	x	x	-
EGR (with UCMINF)	Ols05	x	-	-	x	-	-	x

Table 3.1: Analytical costs involved in the maximum likelihood estimation. This table shows which entities are used among the listed methods for maximum likelihood optimization and serves as an overview of the analytical work involved. $\mathbf{M}$ is simply the EM-iteration, i.e. $\boldsymbol{\theta}^{n+1} = \mathbf{M}(\boldsymbol{\theta}^n)$. $\mathbf{M}'$ is the derivative and Q'' is the second order derivative of the Q-function defined for the EM algorithm. $\mathcal{L}, \mathcal{L}'$ and $\mathcal{L}''$ are the true likelihood and the derivatives and 'S.L.' denotes whether or not the method is using search line or step size optimization. The last entry 'EG with BFGS' relates to the Easy Gradient Recipe, explained in Section 3.3.

3.1.3 The Bound Formulation

Early in the nineties [32] made a reformulation of the EM algorithm. A formulation which uses a bound on the log likelihood as the central cost function and which is closely related to the free energy from statistical physics. The relation of the bound and the free energy, will be explained in more detail in Section 4.1.1.

Instead of the posterior mean of the complete log likelihood, a lower bound, B, is introduced on the log-likelihood. The lower bound B is based on the Jensen inequality which is valid for any choice of $\boldsymbol{\theta}'$

$$B(\boldsymbol{\theta}, \boldsymbol{\theta}') = \int p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}') \ln \frac{p(\mathbf{S}, \mathbf{X}|\boldsymbol{\theta})}{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}')} d\mathbf{S} \leq \ln p(\mathbf{X}|\boldsymbol{\theta})$$

The bound B has the important property that choosing the parameters in a certain way, the bound becomes equal to the log likelihood. This is clear when the bound is rewritten using Bayes theorem

$$B(\boldsymbol{\theta}, \boldsymbol{\theta}') = \ln p(\mathbf{X}|\boldsymbol{\theta}) - KL[p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}')||p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})] \tag{3.2}$$

where KL denotes the Kullback-Leibler divergence between the two distributions. The Kullback-Leibler divergence is defined by $KL(q||p) = \int q \ln q/p$, that is, KL is zero if and only if $q = p$ and positive otherwise. Thus if $\boldsymbol{\theta}' = \boldsymbol{\theta}$, then B is equal to the log likelihood. This inspires the two steps of the algorithm: First maximize the bound with respect to the second parameter (E-step) which makes the bound equal to the log likelihood. Then maximize the bound with respect to the first variable (M-step) which is maximizing the current approximation of the log likelihood. That is,

E-step: Find $\boldsymbol{\theta}' = \text{argmax}_{\boldsymbol{\theta}} B(\boldsymbol{\theta}^{(n)}, \boldsymbol{\theta})$

M-step: Find $\boldsymbol{\theta}^{(n+1)} = \text{argmax}_{\boldsymbol{\theta}} B(\boldsymbol{\theta}, \boldsymbol{\theta}')$

Although it may not look like the same algorithm as presented earlier in this chapter, the two formulations are in fact equivalent as is demonstrated next.

3.1.3.1 Connecting the Dempster and the Bound Formulation

In the bound formulation, the complete variable $\mathbf{U}$ is not a natural representation of the problem. So in order to parallel the two formulations and pointing out the differences, the Dempster formulation are rewritten using the observed data $\mathbf{X}$ and the hidden data $\mathbf{S}$ rather than the complete data $\mathbf{U} = \{\mathbf{X}, \mathbf{S}\}$. The central function Q is therefore rewritten in the following way:

$$\begin{aligned} Q(\boldsymbol{\theta}', \boldsymbol{\theta}^{(n)}) &= \int p(\mathbf{U}|\mathbf{X}, \boldsymbol{\theta}^{(n)}) \ln p(\mathbf{U}|\boldsymbol{\theta}') d\mathbf{U} \\ &= \int p(\tilde{\mathbf{X}}, \mathbf{S}|\mathbf{X}, \boldsymbol{\theta}^{(n)}) \ln p(\tilde{\mathbf{X}}, \mathbf{S}|\boldsymbol{\theta}') d\tilde{\mathbf{X}} d\mathbf{S} \\ &= \int p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}^{(n)}) \ln p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}') d\mathbf{S} \\ &= B(\boldsymbol{\theta}', \boldsymbol{\theta}^{(n)}) + H(\boldsymbol{\theta}^{(n)}, \boldsymbol{\theta}^{(n)}) \end{aligned}$$

where H is the entropy of the posterior. Thus, maximizing $Q(\boldsymbol{\theta}', \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}'$ is equivalent to maximizing $B(\boldsymbol{\theta}', \boldsymbol{\theta})$ with respect to the same variable and therefore the M-step of the two formulations are exactly equal.

The E-step is slightly more complicated. First the E-step and M-step is rewritten using the derivation above by noting that computing Q is essentially the same as computing the posterior $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})$ and maximizing Q is equivalent to maximizing B, i.e.

E-step: Compute $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}^{(n)})$.

M-step: Find $\boldsymbol{\theta}^{(n+1)} = \text{argmax}_{\boldsymbol{\theta}} B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$

To complete the connection it must be proven that the posterior $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})$ is the distribution which maximizes B. But this follows directly from rewriting the bound as a sum of the log likelihood and the Kullback-Leibler divergence in Eq. 3.2. The KL divergence is always non-negative and zero if and only if the two distributions are equal.

3.1.3.2 The General Bound Approach

Consider now a more flexible model when bounding the log likelihood. It is here utilized that Jensen's inequality makes the following inequality true for *any* distribution $q(\mathbf{S}|\boldsymbol{\phi})$

$$B(\boldsymbol{\theta}, \boldsymbol{\phi}) = \int q(\mathbf{S}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta})}{q(\mathbf{S}|\boldsymbol{\phi})} d\mathbf{S} \leq \ln p(\mathbf{X}|\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta})$$

If $q(\mathbf{S}|\boldsymbol{\phi})$ is a family of distributions which include as a special case the source posterior $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})$, i.e. if there for every set $(\mathbf{X}, \boldsymbol{\theta})$ exists a $\boldsymbol{\phi}_*$ such that $q(\mathbf{S}|\boldsymbol{\phi}_*) = p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})$, then the exact same bound as before is obtained, and the EM algorithm resulting from it

E-step: Find $\boldsymbol{\phi} = \text{argmax}_{\boldsymbol{\phi}'} B(\boldsymbol{\theta}, \boldsymbol{\phi}')$

M-step: Find $\boldsymbol{\theta} = \text{argmax}_{\boldsymbol{\theta}'} B(\boldsymbol{\theta}', \boldsymbol{\phi})$

will be the standard EM algorithm. But in some cases it is useful to choose a different model family for $q(\mathbf{S}|\boldsymbol{\phi})$ in order to obtain other goals such as an easier handling of the integral involved. In those situations, both the E-step and the M-step becomes different. Taking the E-step first, one may observe that the bound is still closely related to the log likelihood and the posterior,

$$B(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathcal{L}(\boldsymbol{\theta}) - KL[q(\mathbf{S}|\boldsymbol{\phi})||p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})] \tag{3.3}$$

and therefore, maximizing the bound with respect to $\boldsymbol{\phi}$ is minimizing the Kullback-Leibler divergence between the source posterior and the distribution $q(\mathbf{S}|\boldsymbol{\phi})$, sometimes called the variational distribution. In case the source posterior is not realizable within the family of variational distributions, the bound is a bound never reaching equality, and thus never becomes exactly the log likelihood, but if the family of variational distributions is fairly flexible, it may still be a good approximation. To shed light on the M-step, the bound may be rewritten

$$B(\boldsymbol{\theta}, \boldsymbol{\phi}) = \langle \ln p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}) \rangle_{q(\mathbf{S}|\boldsymbol{\phi})} - H(\boldsymbol{\phi}, \boldsymbol{\phi})$$

Clearly the entropy is un-important in this step and the differentiation of the complete likelihood with respect to $\boldsymbol{\theta}$ is of course also unaltered. But the mean is now with respect to a distribution which is not exactly the source posterior, but one that is approximated as well as possible within the variational family.

The upshot of the more general bound is completely dependent on the choice of variational family. If the family is sufficient flexible, the algorithm is essentially a true EM algorithm, and if the family is too simple, the algorithm most likely produces nonsense. But, the general bound approach is included because it plays an important role in Mean Field ICA – a role in which it is simple enough to enable an approximate solution to the integrals for rather advanced models and yet sufficiently flexible to be a good approximation of the log likelihood.

3.1.4 Variational Bayes EM

In the maximum likelihood approach to a model with unobserved variables, there is a fundamental distinction between the unknown parameters $\boldsymbol{\theta}$ to be estimated and the unknown variables $\mathbf{S}$: The unobserved variables are integrated out while the parameters are not. To some extend the distinction makes sense for two reasons: 1) Most often, the number of unobserved variables scales with the size of the problem while the parameters do not and 2) integrating out the parameters, seems like throwing the baby out with bath water. But in fact, it is quite possible both to work in a Bayesian approach, that is 'integrating out what is not known' and still get a very reasonable estimate of the parameters using the parameter posterior $p(\boldsymbol{\theta}|\mathbf{X})$.

The key idea is to construct a bound on the marginal likelihood $\ln p(\mathbf{X})$, i.e. the likelihood in which the parameters are integrated out, sometimes referred to as the *evidence*. As with the bound formulation of the EM algorithm, the following hold for *any* distribution $q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi})$

$$\begin{aligned}
\ln p(\mathbf{X}) &= \ln \int p(\mathbf{X},\mathbf{S},\boldsymbol{\theta}) d\mathbf{S} d\boldsymbol{\theta} \\
&\geq \int q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{X},\mathbf{S},\boldsymbol{\theta})}{q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi})} d\mathbf{S} d\boldsymbol{\theta} \\
&= \ln p(\mathbf{X}) - KL[q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi})||p(\mathbf{S},\boldsymbol{\theta}|\mathbf{X})]
\end{aligned}$$

In principle this is it: Find the variational parameter $\boldsymbol{\phi}$ and use the distribution $q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi})$ as the posterior to estimate both $\mathbf{S}$ and $\boldsymbol{\theta}$. But it is not a practical approach, since either q must be chosen very flexible making the optimization very hard or q must be calculated using Bayes theorem, which as a normalization factor contains the marginal likelihood which is the entity not known. Instead one can factorize the variational distribution $q(\mathbf{S},\boldsymbol{\theta}|\boldsymbol{\phi}) = q(\mathbf{S}|\boldsymbol{\phi}_s)q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)$ and obtain the somewhat easier

$$\begin{aligned}
\ln p(\mathbf{X}) &\geq \int q(\mathbf{S}|\boldsymbol{\phi}_s)q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta) \ln \frac{p(\mathbf{X},\mathbf{S},\boldsymbol{\theta})}{q(\mathbf{S}|\boldsymbol{\phi}_s)q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)} d\mathbf{S} d\boldsymbol{\theta} \\
&= \ln p(\mathbf{X}) - KL[q(\mathbf{S}|\boldsymbol{\phi}_s)q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)||p(\mathbf{S},\boldsymbol{\theta}|\mathbf{X})] \\
&\equiv B_e(\boldsymbol{\phi}_\theta,\boldsymbol{\phi}_s)
\end{aligned}$$

Ideally one would like to optimize over the joint variable $(\boldsymbol{\phi}_\theta,\boldsymbol{\phi}_s)$, but for most systems of interest, the variational family of the unobserved variables $q(\mathbf{S}|\boldsymbol{\phi}_s)$ is either too complicated or has a very large parameter $\boldsymbol{\phi}_s$. Therefore, the Variational Bayes EM approach is to maximize the lower bound B_e in a coordinate ascend manner,

VBE-step: Find $\boldsymbol{\phi}_s = \text{argmax}_{\boldsymbol{\phi}'_s} B(\boldsymbol{\phi}_\theta,\boldsymbol{\phi}'_s)$

VBM-step: Find $\boldsymbol{\phi}_\theta = \text{argmax}_{\boldsymbol{\phi}'_\theta} B(\boldsymbol{\phi}'_\theta,\boldsymbol{\phi}_s)$

To the extend that the variational distribution is sufficiently flexible to include the joint posterior $p(\mathbf{S},\boldsymbol{\theta}|\mathbf{X})$, the VB-EM is a tight bound on the marginal likelihood.

The Variational Bayes EM simplifies into the standard EM algorithm, if $q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta) = \delta(\boldsymbol{\theta} - \boldsymbol{\theta}_{ML})$, i.e. if one chooses the variational distribution over the parameters to be a Dirac delta function centered in the maximum likelihood solution. In that sense is the Variational Bayes EM a way to incorporate the uncertainty of the parameter estimation into an EM framework.

3.2 Low Noise Analysis

Section 3.1 introduced the EM algorithm in different formulations and the Variational Bayes EM variant. Now focus turns to an investigation of the convergence properties of these methods in the limit of low noise. This was originally motivated by the numerical experience that the EM algorithm somewhat surprisingly performed poorly in this limit.

First the posterior moments that are needed in the analysis is approximated rather accurately using the Saddle Point approximation. Then an analysis of the standard EM and the Variational Bayes EM follows. In the end a more general but also slightly less rigorous argument based on the Hessian's is presented.

3.2.1 Saddle Point Approximation

In this subsection, as a general approach to the difficulty of computing the source posterior moments, an approximation of the integral involved in the posterior is made as a Taylor expansion of the posterior moments in orders of the noise. Doing this, some general properties of the algorithm in the low noise limit can be studied.

For notational simplicity, it is assumed for the rest of the section that the noise is isotropic, $\mathbf{W} = \sigma^2 \mathbf{I}$. In my experience this does not change any of the results obtained. Furthermore, since the time steps are assumed independent and the log likelihood therefore a sum over time, one needs only to consider a single time step in the Saddle Point approximation.

3.2.1.1 Approximation of an Integral

The technique applied here is the so-called 'saddle-point approximation' sometimes referred to as the 'Laplace approximation' of an integral. It is well-known in statistical physics, see e.g. [9], and in large parts of applied mathematics, but not all scientific communities are denoting it by the same name. The approximation is described in details below to clear things up.

To lowest order, it is basically an approximation of the integral $I = \int \exp[-f(s)]ds$ which for very large values of $f(s)$ and if $f(s)$ is uni-modal, becomes dominated by its minimum $f(s_0)$. The point s_0 in which $f'(s_0) = 0$ is called the saddle-point, and the approximation is to lowest order $I \cong \exp[-f(s_0)]$. The application of the technique in this section is slightly more complicated since the problem is multidimensional and expanded to higher order. Also, the low noise limit is considered, but all in all, the basic idea is the same. Inspired by the equality $\ln p(\mathbf{x}) = \ln \int e^{\ln p(\mathbf{x},\mathbf{s})} d\mathbf{s}$, the following function is defined

$$g(\mathbf{s}, \mathbf{h}) = -\sigma^2 \left(\ln p(\mathbf{x}|\mathbf{s}) + \ln p(\mathbf{s}) + \mathbf{h}^T \mathbf{s} \right) \tag{3.4}$$

Note that for $\mathbf{h} = \mathbf{0}$ the function g is a scaled version of the joint distribution $\ln p(\mathbf{x}, \mathbf{s})$. The somewhat artificial variable $\mathbf{h}$ is a utility variable which will help obtain the posterior moments. Defining the function $g(\mathbf{s}, \mathbf{h})$ as above, one get that

$$\ln I = \ln \int e^{-\frac{1}{\sigma^2} g(\mathbf{s},\mathbf{h})} d\mathbf{s}$$

is a moment generating function in the sense

$$\langle \mathbf{s} \rangle = \frac{\partial \ln I}{\partial \mathbf{h}}\Big|_{\mathbf{h}=\mathbf{0}} \qquad \langle \mathbf{s}\mathbf{s}^T \rangle - \langle \mathbf{s} \rangle \langle \mathbf{s} \rangle^T = \frac{\partial^2 \ln I}{\partial \mathbf{h} \partial \mathbf{h}^T}\Big|_{\mathbf{h}=\mathbf{0}}$$

where $\langle \cdot \rangle$ denotes average with respect to the source posterior $p(\mathbf{s}|\mathbf{x})$. For small σ^2 one can approximate $\ln I$ by a Taylor expansion of $g(\mathbf{s}, \mathbf{h})$ in the saddle point $\mathbf{s}_0$ defined by $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$. From this the saddle point approximation to the integral in $\ln I$ is obtained

$$\begin{aligned}
\ln I &= \ln \int e^{-\frac{1}{\sigma^2} g(\mathbf{s}, \mathbf{h})} d\mathbf{s} \\
&= \ln \int e^{-\frac{1}{\sigma^2} g(\mathbf{s}_0, \mathbf{h}) - \frac{1}{2}\frac{1}{\sigma^2}(\mathbf{s}-\mathbf{s}_0)^T \mathbf{g}_0''(\mathbf{s}-\mathbf{s}_0)} d\mathbf{s} + \mathcal{O}(\sigma^2) \\
&= -\frac{1}{\sigma^2} g(\mathbf{s}_0) + \ln \sqrt{\det[2\pi\sigma^2 (\mathbf{g}_0'')^{-1}]} + \mathcal{O}(\sigma^2) \qquad (3.5)
\end{aligned}$$

In the limit of small σ^2, the derivative of the second term is scaling as $\mathcal{O}(\sigma^4)$ and thus the approximation is efficiently dominated by the first term $-g(\mathbf{s}_0)/\sigma^2$.

3.2.2 EM for Linear Mixtures

In this subsection, the saddle point approximation is applied to the EM algorithm on a linear model with additive Gaussian noise.

3.2.2.1 Finding the Saddle Point

In the above the saddle point has been used without explicitly knowing its value. To find it, one make use of the definition $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$ but also an approximation of the saddle point in the low noise regime, i.e. $\mathbf{s}_0$ is written as a Taylor expansion in σ^2

$$\mathbf{s}_0 = \hat{\mathbf{s}}_0 + \sigma^2 \tilde{\mathbf{s}}_0 + \mathcal{O}(\sigma^4)$$

In the zero noise limit, the source prior and utility term of Equation (3.4) vanishes and leaves only $\ln p(\mathbf{x}|\mathbf{s})$. Since the noise is Gaussian, the solution $\hat{\mathbf{s}}_0$ which dominates in this limit, is the least squares solution

$$\hat{\mathbf{s}}_0 = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{x}$$

which simplifies further when $\mathbf{A}$ is square. An expression for the first order term $\tilde{\mathbf{s}}_0$ is now wanted and to find this the combined expression for $\mathbf{s}_0$ is inserted into the equation $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$ and the terms to first order in σ^2 rearranged. Doing that one obtains

$$\tilde{\mathbf{s}}_0 = (\mathbf{A}^T \mathbf{A})^{-1} \left[\frac{\mathbf{p}'(\hat{\mathbf{s}}_0)}{p(\hat{\mathbf{s}}_0)} + \mathbf{h} \right]$$

Having an expression for the saddle point in orders of σ^2, makes it possible to use the approximation of the integral to obtain the expressions for the posterior moments.

3.2.2.2 The Posterior Moments

Inserting the expression for $\mathbf{s}_0$ into Equation (3.5) and performing the calculations, one obtains

$$\langle \mathbf{s} \rangle = \left.\frac{\partial \ln I}{\partial \mathbf{h}}\right|_{\mathbf{h}=\mathbf{0}} = \hat{\mathbf{s}}_0 + \sigma^2 \tilde{\mathbf{s}}_0 + \mathcal{O}(\sigma^4) \quad (3.6)$$

$$\langle \mathbf{s}\mathbf{s}^T \rangle - \langle \mathbf{s} \rangle \langle \mathbf{s} \rangle^T = \left.\frac{\partial^2 \ln I}{\partial \mathbf{h} \partial \mathbf{h}^T}\right|_{\mathbf{h}=\mathbf{0}} = \sigma^2 (\mathbf{A}^T\mathbf{A})^{-1} + \mathcal{O}(\sigma^2) \quad (3.7)$$

Note that the posterior mean is the saddle point. This is due to the fact that a minimization of $g(\mathbf{s}, \mathbf{h})$ corresponds to a maximization of $\ln[p(\mathbf{x}, \mathbf{s})]$ which through Bayes theorem has the same maximum as the posterior $p(\mathbf{s}|\mathbf{x})$. In the approximation of the integral it has been implicitly assumed that the joint distribution is quadratic, i.e the source posterior uni-modal and symmetric, and therefore the mean is the same as the most probable value. The variance provides the second moment through

$$\langle \mathbf{s}\mathbf{s}^T \rangle = \hat{\mathbf{s}}\hat{\mathbf{s}}^T + \sigma^2 \mathbf{B}_t + \mathcal{O}(\sigma^4)$$

$$\mathbf{B}_t = (\mathbf{A}^T\mathbf{A})^{-1} + (\mathbf{A}^T\mathbf{A})^{-1} \frac{\mathbf{p}'(\hat{\mathbf{s}})}{p(\hat{\mathbf{s}})} \hat{\mathbf{s}}^T + \hat{\mathbf{s}} \frac{\mathbf{p}'(\hat{\mathbf{s}})}{p(\hat{\mathbf{s}})}^T (\mathbf{A}^T\mathbf{A})^{-1} \quad (3.8)$$

To summarize, this yields approximations for the posterior moments in the low noise regime. In order to investigate the quality of this approximation, the exact and approximated moments for a mixture of gaussians has been computed. The result is plotted in Fig. 3.2. As expected the approximation is not good for large noise variance, but for noise variance in the area of 10^{-2} to 10^{-3} and smaller, the saddle point approximation is very accurate.

Combining the posterior moments for each time step $\mathbf{s}$ into larger matrices $\mathbf{S}$ for the entire data set, one obtains

$$\langle \mathbf{S} \rangle = \hat{\mathbf{S}} + \sigma^2 \tilde{\mathbf{S}} + \mathcal{O}(\sigma^4) \quad (3.9)$$

$$\langle \mathbf{S}\mathbf{S}^T \rangle = \hat{\mathbf{S}}\hat{\mathbf{S}}^T + \sigma^2 \mathbf{B} + \mathcal{O}(\sigma^4) \quad (3.10)$$

where $\mathbf{B} = \sum_t \mathbf{B}_t$ using the expression for $\mathbf{B}_t$ in Equation (3.8).

3.2.2.3 The EM Updates

The approximated posterior moments, can be directly inserted into Eq. 2.13 and 2.14 to obtain the EM updates of $\mathbf{A}$ and σ^2

$$\mathbf{A}_{n+1} = \mathbf{X}\hat{\mathbf{S}}(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1} + \sigma^2 \mathbf{X}\mathbf{F}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1} + \mathcal{O}(\sigma^4) \quad (3.11)$$

$$\sigma^2_{n+1} = \sigma^2_{bias} + \sigma^2 \Big(\frac{\mathrm{rank}(\mathbf{A})}{m} - \frac{2}{m}\mathrm{Tr}(\mathbf{U}) \Big) + \mathcal{O}(\sigma^4) \quad (3.12)$$

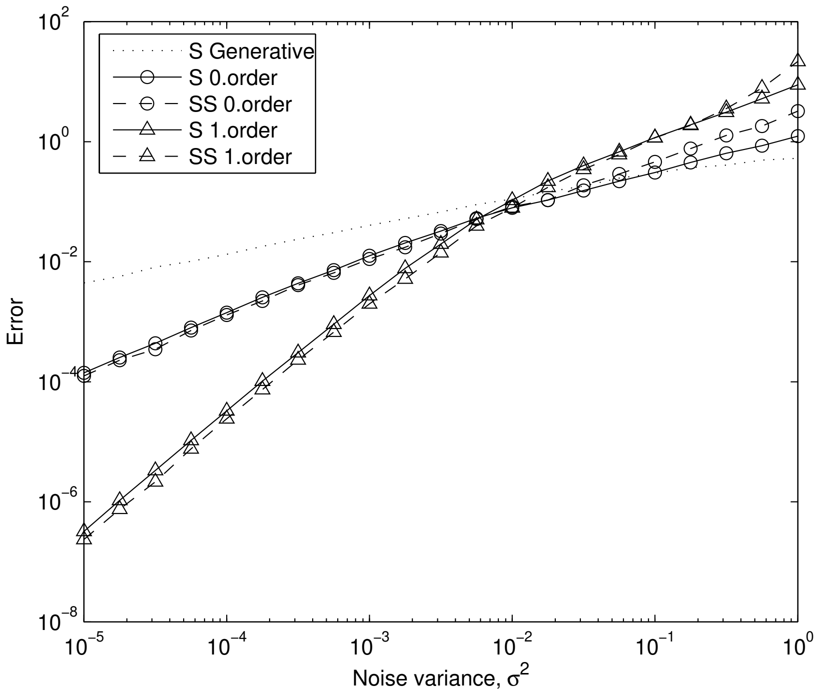


Figure 3.2: The accuracy of the saddle point approximation. For a MoG source prior and given a data set $\mathbf{X}$ and the generative mixing matrix, both the exact posterior means and the saddle point approximations to the posterior means can be computed. Doing that, the approximation can be compared to the exact in order to evaluate the accuracy of the saddle point approximation. The error is calculated as the root of sum of squared differences averaged over number of variables. Also, comparing the generative sources with the exact posterior means, the error on the generative sources are included for perspective. This plot shows that the approximation of the source posterior moments are very accurate and that the first order (in the noise variance) is more accurate than the zeroth order for low noise values as expected.

where $\mathbf{F} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{p}'(\hat{\mathbf{S}})/p(\hat{\mathbf{S}}) - \hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}\mathbf{B}$ and $\sigma^2_{bias}, \mathbf{U}$ will be explained shortly. These equations are general in the sense that they apply for both square and overdetermined mixing matrix and they simplify considerably in the more specific cases. In the square case $\sigma^2_{bias} = 0$ and $\mathbf{U} = \mathbf{0}$, and the expressions simplify into

$$\begin{aligned} \mathbf{A}_{n+1} &= \mathbf{A}_n + \sigma^2\tilde{\mathbf{A}}_n + \mathcal{O}(\sigma^4) \\ \sigma^2_{n+1} &= \sigma^2_n + \mathcal{O}(\sigma^4) \end{aligned}$$

$$\tilde{\mathbf{A}}_n = \mathbf{A}^{-T}\left(\mathbf{A}^T + \tfrac{1}{N}\frac{\mathbf{p}'(\hat{\mathbf{S}})}{p(\hat{\mathbf{S}})}\mathbf{X}^T\right)\mathbf{A}$$

Many important conclusions can be reached from these equations: (i) the update of the mixing matrix is "freezing" in the sense that to zeroth order the new estimated mixing matrix is identical to the previous one. (ii) The first order correction $\tilde{\mathbf{A}}$ of the mixing matrix is scaling with the change derived from the noise less Bell & Sejnowski algorithm (the expression in the parenthesis). This shows that to first order in the noise variance, there is no improvement of the noisy model compared to the noiseless counterpart. The consequence is that only for noise variances sufficiently large to make σ^4 important, is the noise model different (and hopefully better) than the noiseless model. (iii) The update of the noise is scaling with σ^4 which makes the change in the estimated noise extremely slow to converge when the noise levels are small.

In the overdetermined case, the equations simplifies into

$$\mathbf{A}_{n+1} = \mathbf{A}_n + \sigma^2\tilde{\mathbf{A}}_n + \mathcal{O}(\sigma^4)$$

$$\tilde{\mathbf{A}}_n = (\mathbf{I} - \mathbf{A}\mathbf{A}^+)\frac{\mathbf{X}\mathbf{p}'(\hat{\mathbf{S}})^T}{Np(\hat{\mathbf{S}})} - \mathbf{A} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\frac{\mathbf{p}'(\hat{\mathbf{S}})\mathbf{X}^T}{Np(\hat{\mathbf{S}})}\mathbf{A}$$

$$\sigma^2_{bias} = 1 - \frac{\mathrm{rank}(\mathbf{A})}{m}, \qquad \mathbf{U} = \mathbf{\Delta}^T\mathbf{X}\frac{\mathbf{p}'(\hat{\mathbf{S}})}{p(\hat{\mathbf{S}})}\hat{\mathbf{A}}^+/N$$

where $\mathbf{\Delta} = \mathbf{I} - \mathbf{A}\mathbf{A}^+$. The "+" denotes the Moore-Penrose pseudo-inverse, i.e. for overdetermined non-aquare matrices of full rank, $\mathbf{A}^+ = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$. The following should be noted about these equations: (i) As in the square case, the update in the mixing matrix in freezing, resulting in extremely slow convergence in the low noise limit. (ii) In the first order correction for the mixing matrix, one can recognize the structure which leads to the Bell & Sejnowski solution in the square case, and see that it is the term $(\mathbf{I} - \mathbf{A}\mathbf{A}^+)\mathbf{X}\mathbf{p}'(\hat{\mathbf{S}})^T/Np(\hat{\mathbf{S}})$ which makes the difference. Not surprisingly, this term is increasing as the ratio of sensors and sources are increasing. (iii) The noise now also has a bias term. On an intuitive level this can be explained as the spill-over from the 'misfit' of the number of sources.

3.2.2.4 Step Length Methods and Discussion

The EM-variants using a step length modification are very easy and general applicable speedup suggestions. Basically one is enlarging the step size proposed by EM with a factor η

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

When the low noise analysis is combined with this idea one gets

$$\mathbf{A}_{n+1} = \mathbf{A}_n + \eta\sigma^2\tilde{\mathbf{A}} + \mathcal{O}(\sigma^4)$$

That is, the trick of the step length methods is directly countering the problem of the small noise variance. On the downside there is no longer a guaranteed increase in the log likelihood for each step, but this can be controlled with a test-step modification or similar hacks.

The saddle point analysis of EM applied to the ordinary ICA problem reveals that the EM algorithm in the low noise limit converges infinitely slowly. On the other hand if the noise variance is sufficiently large, the analysis does not apply and thus one might be able to see some improvement in the large noise regime. This is not the case in practice, however, since the enhanced noise levels for the same amount of data makes estimation correspondingly more difficult. In my experience, the window in which the noise is large enough to prevent the EM algorithm to get stuck and yet small enough to provide good estimates with reasonable size data sets, is indeed very small.

3.2.3 Variational Bayes EM for Linear Mixtures

The standard EM does not work well on a classical ICA problem, probabilistic PCA (pPCA), and Factor Analysis (FA), in the low noise limit. But this does not mean that other and more advanced variants will fail too. In this subsection the Variational Bayes EM approach is investigated in the low noise limit.

When using the Variational Bayes EM algorithm, the variational distributions need to be known. These are readily obtained in a general form using functional derivatives and constraining the variational distribution appropriately to be a density, that is, always positive and normalized. The result is

$$\begin{aligned}
\frac{\delta B(\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s)}{\delta q(\mathbf{S}|\boldsymbol{\phi}_s)} = 0 \quad &\Rightarrow \quad q(\mathbf{S}|\boldsymbol{\phi}_s) \propto \exp[\langle \ln p(\mathbf{X}, \mathbf{S}, \boldsymbol{\theta})\rangle_{q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)}] \\
\frac{\delta B(\boldsymbol{\phi}_\theta, \boldsymbol{\phi}_s)}{\delta q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta)} = 0 \quad &\Rightarrow \quad q(\boldsymbol{\theta}|\boldsymbol{\phi}_\theta) \propto \exp[\langle \ln p(\mathbf{X}, \mathbf{S}, \boldsymbol{\theta})\rangle_{q(\mathbf{S}|\boldsymbol{\phi}_s)}]
\end{aligned}$$

More detailed examples of completely factorized variational distributions will be given in Chapter 4. In the following a linear mixing with mixing matrix $\mathbf{A}$ and isotropic Gaussian noise $\mathbf{W} = \sigma^2\mathbf{I}$ is considered. Thus, the three unknown entities of interest are the sources $\mathbf{S}$, the mixing matrix $\mathbf{A}$ and the noise variance σ^2.

3.2.3.1 Variational Distributions and Conjugate Priors

For all three entities $\mathbf{S}, \mathbf{A}, \sigma^2$ one needs to find the variational distribution q and since this distribution includes the prior, the prior of each of the entities must be considered. The variational distribution over the sources is seemingly rather simple to compute, when leaving out the trouble of finding the normalization constant,

$$\begin{aligned} q(\mathbf{S}) &\propto p(\mathbf{S}) \exp(\langle p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \sigma^2) \rangle_{q(\mathbf{A}), q(\sigma^2)}) \\ &\propto p(\mathbf{S}) \exp\left[-\frac{1}{2}\mathrm{Tr}(\mathbf{S}^T\mathbf{J}\mathbf{S}) + \mathrm{Tr}(\mathbf{h}^T\mathbf{S})\right] \end{aligned} \tag{3.13}$$

where the matrices $\mathbf{J}$ and $\mathbf{h}$ are now redefined to fit the variational Bayes framework

$$\begin{aligned} \mathbf{J} &= \langle 1/\sigma^2 \rangle \langle \mathbf{A}^T\mathbf{A} \rangle \\ \mathbf{h} &= \langle 1/\sigma^2 \rangle \langle \mathbf{A} \rangle^T \mathbf{X} \end{aligned}$$

are placeholders, i.e. shorthand notation for convenience. Equation (3.13) may look simple since the quadratic form of the Gaussian noise is inherited, but a prior have not been chosen yet – a choice which can make the variational distribution rather complicated. The quadratic form suggests that a Gaussian prior would make everything work just fine: Choosing the prior on the sources to be Gaussian, the variational distribution would it self be a Gaussian and moments such as mean and covariance would be easy to find. But the problem is that sources with a Gaussian prior cannot be separated using ICA, a fundamental fact of the independent component analysis; see e.g. [14]. So one has to choose a different source prior and deal with the problems that stems from the somewhat complicated variational distribution. For now, assume that a source prior is chosen and that all needed entities relating to the variational distribution of the sources can be computed.

The situation that one cannot choose the prior which makes the variational distribution simple, is special for the sources and does not apply for the parameters $\mathbf{A}$ and σ^2. If the mixing matrix prior are chosen to be a zero mean Gaussian with covariance matrix $\mathbf{\Sigma}_{p(a)}$

$$p(\mathbf{A}) = \frac{1}{\sqrt{\det(2\pi\mathbf{\Sigma}_{p(a)})}} \exp\left[-\frac{1}{2}\mathrm{Tr}(\mathbf{A}\mathbf{\Sigma}_{p(a)}^{-1}\mathbf{A}^T)\right]$$

then the variational distribution becomes reasonably simple due to the fact that the noisy observation $p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \sigma^2)$ and the prior on $\mathbf{A}$ are chosen carefully to match each other

$$\begin{aligned} q(\mathbf{A}) &\propto p(\mathbf{A}) \exp\left[\langle \ln p(\mathbf{X}, \mathbf{S}|\mathbf{A}, \sigma^2) \rangle_{q(\mathbf{S}), q(\sigma^2)}\right] \\ &\propto \exp\left[-\tfrac{1}{2}\mathrm{Tr}[(\mathbf{A} - \mathbf{M}_{q(a)})^T \mathbf{\Sigma}_{q(a)}^{-1} (\mathbf{A} - \mathbf{M}_{q(a)})]\right] \end{aligned} \tag{3.14}$$

where

$$\begin{aligned}\mathbf{\Sigma}_{q(a)}^{-1} &= \langle 1/\sigma^2\rangle\langle \mathbf{SS}^T\rangle + \mathbf{\Sigma}_{p(a)}^{-1}\\ \mathbf{M}_{q(a)} &= \mathbf{X}\langle \mathbf{S}\rangle^T\left[\langle \mathbf{SS}^T\rangle + 1/\langle 1/\sigma^2\rangle \mathbf{\Sigma}_{p(a)}^{-1}\right]^{-1}\end{aligned}$$

However, the prior and the distribution from the noisy observation model are from the same family of distributions – in this case a Gaussian. Priors with this property are called *conjugate priors* and they make the analytical and numerical work considerably easier.

But knowing something about the parameter in advance, it will be unlikely that this prior knowledge almost magically matches the observation model. The answer to that objection is that situations in which the conjugate prior in fact matches the a priori knowledge are very rare and that the reason for using conjugate priors are analytical tractability: one simply choose the conjugate priors not because they are true but because they are convenient. That being said, one can argue that conjugate priors make an ideal test case for artificial data when testing an algorithm. Investigating a new algorithm, it is reasonable to choose simplified situations in order to keep the analysis clear of unnecessary sources of problems.

While the conjugate prior for the mixing matrix was simply just a Gaussian, the conjugate prior for the noise variance is more complicated. The variational distribution is

$$\begin{aligned}q(\sigma^2) &\propto p(\sigma^2)\exp\left[\langle \ln p(\mathbf{X},\mathbf{S}|\mathbf{A},\sigma^2)\rangle_{q(\mathbf{S}),q(\mathbf{A})}\right]\\ &\propto p(\sigma^2)\exp[\ln[(\sigma^2)^{-\alpha_q}] - \beta_q/\sigma^2] \qquad (3.15)\end{aligned}$$

That is, the prior must be of the same form as the expression in the exponential function. The conjugate prior for this form is the so-called inverse gamma distribution

$$p(\sigma^2) = \frac{1}{Z}(\sigma^2)^{-(\alpha_p-1)}e^{-\beta_p/\sigma^2}$$

such that the variational distribution is given by

$$q(\sigma^2) = \frac{\beta^\alpha}{\Gamma(\alpha)}(\sigma^2)^{-(\alpha-1)}e^{-\beta/\sigma^2}$$

where

$$\begin{aligned}\alpha &= \alpha_p + \frac{mN}{2}\\ \beta &= \beta_p + \frac{1}{2}\Big(\mathrm{Tr}(\mathbf{X}^T\mathbf{X}) + \mathrm{Tr}(\langle \mathbf{A}^T\mathbf{A}\rangle\langle \mathbf{SS}^T\rangle) - 2\mathrm{Tr}(\mathbf{X}^T\langle \mathbf{A}\rangle\langle \mathbf{S}\rangle^T)\Big)\end{aligned}$$

This provides the variational distributions for the sources and the parameters and therefore in principle paving the way for the Variational Bayes EM. The purpose of this is not to use the Variational Bayes EM algorithm, however, but rather to analyze it in the zero noise limit and thereby investigate if Variational Bayes EM suffers form the same defects as the standard EM algorithm or not.

3.2.3.2 Zero Noise Limit of the Updates

When analyzing the zero noise limit of the updates of both the sources and the parameters, the first thing which needs to be clarified is what the zero noise limit now means. In the standard EM it was rather obvious that the zero noise limit were the situation in which $\sigma^2 \to 0$, but in the Variational Bayes EM, σ^2 is integrated out and does not appear in the marginal likelihood. But, the variational distributions of the sources and the mixing matrix shows, that the mean $\langle 1/\sigma^2 \rangle$ have much the same role, as $1/\sigma^2$ had in the standard EM. Therefore a new variable $r^2 \equiv 1/\langle 1/\sigma^2 \rangle$ is defined and instead the limit of this new variable $r^2 \to 0$ is considered.

The zero noise limit of the source posterior moments can be found by remembering that in the limit, the source prior does not matter and the correlation between the sources vanishes. Assuming the data to have unit covariance one obtains

$$\begin{aligned} \langle \mathbf{S} \rangle_{n+1} &\to \mathbf{J}^{-1}\mathbf{h} \\ &\to (\langle \mathbf{A} \rangle_n^T \langle \mathbf{A} \rangle_n)^{-1} \langle \mathbf{A} \rangle_n^T \mathbf{X} \\ \langle \mathbf{S}\mathbf{S}^T \rangle_{n+1} &\to \langle \mathbf{S} \rangle_{n+1} \langle \mathbf{S} \rangle_{n+1}^T \\ &\to N(\langle \mathbf{A} \rangle_n^T \langle \mathbf{A} \rangle_n)^{-1} \end{aligned}$$

Since the variational distribution of the mixing matrix is Gaussian, it is sufficient to know the update of the mean and covariance in order to update the entire distribution. These can be found to be

$$\begin{aligned} \langle \mathbf{A} \rangle_{n+1} &= \mathbf{X} \langle \mathbf{S} \rangle_{n+1}^T \langle \mathbf{S}\mathbf{S}^T \rangle_{n+1}^{-1} &&\to \langle \mathbf{A} \rangle_n \\ \mathrm{Var}(\mathbf{A}) &= r^2(\langle \mathbf{S}\mathbf{S}^T \rangle + r^2 \mathbf{W}^{-1})^{-1} &&\to \mathbf{0} \end{aligned}$$

while the updates of the parameters of the variational distribution for the noise variance are given by

$$\begin{aligned} \alpha_{n+1} &= \alpha_p + \tfrac{mN}{2} &&= \alpha_n \\ \beta_{n+1} &= \beta_p + N\beta_{n+1}^{bias} &&\to \beta_n \end{aligned}$$

where $\beta_{bias} = \frac{1}{2}\mathrm{Tr}(\hat{\boldsymbol{\Delta}}\hat{\boldsymbol{\Delta}}^T)$, $\hat{\boldsymbol{\Delta}} = \mathbf{I} - \langle \mathbf{A} \rangle \langle \mathbf{A} \rangle^+$.

Important to note about the updates is that after one completed iteration of the Variational Bayes EM algorithm, i.e. one VBE-Step and one VBM-Step, the parameters are unchanged to zeroth order. That is, the Variational Bayes EM is stuck in the low noise limit in exactly the same way as the standard EM algorithm.

3.2.3.3 Comments and Observations

Since the order parameter $r^2 = 1/\langle 1/\sigma^2 \rangle$ is substituting in the role of the parameters of interest, one may wonder how the mean and variance of σ^2 are behaving in the limit $r^2 \to 0$. These can be calculated rather easily with the result

$$\begin{aligned}\mathrm{E}(\sigma^2) &= \frac{\beta^2}{\alpha - 1} \\ &\to \frac{\beta_p + N\beta_{bias}}{\alpha_p + \frac{mN}{2} - 1} \\ &\sim \begin{cases} \frac{1}{mN} & \text{(Square)} \\ \frac{1}{m} & \text{(Overdetermined)} \end{cases}\end{aligned}$$

and

$$\begin{aligned}\mathrm{Var}(\sigma^2) &= \frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)} \\ &\to \frac{(\beta_p + N\beta_{bias})^2}{(\alpha_p + \frac{1}{2}mN - 1)^2(\alpha_p + \frac{1}{2}mN - 2)} \\ &\sim \begin{cases} \frac{1}{(mN)^3} & \text{(Square)} \\ \frac{1}{m^3N} & \text{(Overdetermined)} \end{cases}\end{aligned}$$

Note that in the square case that both the mean and variance of σ^2 is going to zero as $N \to \infty$, while this is not the case for the overdetermined case. Seemingly it is the 'misfit' of number of sources in the overdetermined case that influences the noise to have a 'ground-level' as in the traditional EM algorithm.

3.2.4 The Hessian Argument

This subsection describes a different approach to understanding the slowdown of the standard EM algorithm. The approach is more general than the saddle point analysis and based on the Hessian of the bound and the Hessian of the log likelihood. The figures and examples are assuming an observation model using linear mixtures, but in fact the same arguments work for a very general observation model using only the assumption of additive Gaussian noise.

The Hessian of the bound can be derived using the identity in Equation 3.3 $B(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathcal{L}(\boldsymbol{\theta}) - KL$. For notational convenience in the next few lines, the variables of the distributions are omitted, such that p denotes $p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})$ and q denotes $q(\mathbf{S}|\boldsymbol{\phi})$. In that notation it holds that

$$\begin{aligned}\mathbf{H}_B &= \mathbf{H}_\mathcal{L} - \frac{\partial^2 KL(q||p)}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T} \\ &= \mathbf{H}_\mathcal{L} - \int q \frac{\partial \ln p}{\partial\boldsymbol{\theta}} \frac{\partial \ln p}{\partial\boldsymbol{\theta}} d\mathbf{S} + \int \frac{q}{p} \frac{\partial^2 p}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T} d\mathbf{S} \\ &= \mathbf{H}_\mathcal{L} - \int q \frac{\partial \ln p}{\partial\boldsymbol{\theta}} \frac{\partial \ln p}{\partial\boldsymbol{\theta}} d\mathbf{S} \qquad (3.16)\end{aligned}$$

where the last identity is true when $q = p$, i.e. after a completed E-step, for the following reason: When $q = p$, the differentiation can be moved outside

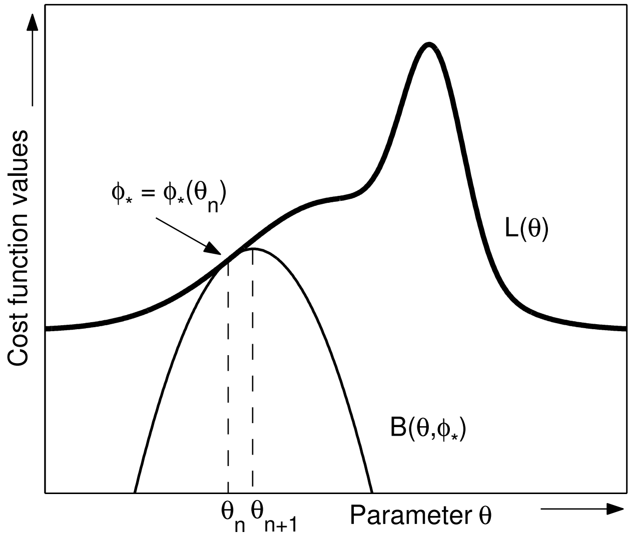


Figure 3.3: A schematic drawing of the EM algorithm. In the point $\boldsymbol{\theta}_n$ one finds the corresponding $\boldsymbol{\phi}_* = \boldsymbol{\phi}(\boldsymbol{\theta}_n)$ and regard the bound as a function of $\boldsymbol{\theta}$. Since the bound, in the point $\boldsymbol{\theta}_n$, must have the same function value and gradient as the log likelihood, but still have to be a bound, the Hessian will often be rather 'curved' and the next parameter value $\boldsymbol{\theta}_{n+1}$ is therefore close to $\boldsymbol{\theta}_n$.

the integral, which then evaluates to 1 as over any density. Differentiating 1 produces zero and thus gives the stated result. Since the last term of the expression is an outer product, the eigenvalues of the bound Hessian are always more negative than the eigenvalues of the of the log likelihood Hessian. This demonstrates that the Hessian of the bound is more sharply peaked compared to the Hessian of the log likelihood.

A schematic drawing of the situation is shown in Figure 3.3: In the point $\boldsymbol{\theta}_n$, it holds that $\boldsymbol{\phi} = \boldsymbol{\phi}_*$ such that the bound equals the log likelihood. Furthermore, as is described in details in Section 3.3, the gradient of the bound equals the gradient of the log likelihood in this point. But the Hessian do not, as has been demonstrated, and the Hessian of the bound in the point $\boldsymbol{\theta}_n$ is more curved than that of the log likelihood. Since at the same time $B \leq \mathcal{L}$, the peak of a quadratic bound must be close to the point $\boldsymbol{\theta}_n$. The peak is exactly pointing to the next value of $\boldsymbol{\theta}$ since in the M-step, the bound is maximized with respect to $\boldsymbol{\theta}$ and thus this peak indicates the next parameter value $\boldsymbol{\theta}_{n+1}$. In Figure 3.3 the bound is drawn as a quadratic function in $\boldsymbol{\theta}$, as is the case for linear mixing.

This analysis leads to a different understanding of why the EM algorithm slows down: When the Hessian of the bound is always more curved than the Hessian of the log likelihood, the next maximum of the bound must be close to the current parameter value and thus cause the combined algorithm to make a relatively smaller step in the parameter space, compared to a corresponding step based on the log likelihood.

3.2.4.1 Scaling and Noise

So far it has been demonstrated that the Hessian of the bound has larger negative eigenvalues than the Hessian of the log likelihood, but not shown how large the effect is or how it scales with the noise variance. To do this, one must go into more details assuming the observation model $\mathbf{x}_t = \mathbf{f}_\theta(\mathbf{s}_t) + \mathbf{n}_t$ where $\mathbf{n}_t$ is additive Gaussian noise with noise variance σ^2. With this assumption the difference of the two hessians can be rewritten using Bayes theorem and assuming that the parameter $\boldsymbol{\theta}$ is a parameter of the mixing only. That is, not a noise parameter and not a parameter of the source prior. It holds that

$$
\begin{aligned}
\frac{\partial \ln p(\mathbf{S}|\mathbf{X},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial \ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} - \frac{\partial \ln p(\mathbf{X}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial \ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} - \left\langle \frac{\partial \ln p(\mathbf{X}|\mathbf{S},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right\rangle_{p(\mathbf{S}|\mathbf{X},\boldsymbol{\theta})} \\
&= -\frac{1}{\sigma^2}\frac{\partial y}{\partial \boldsymbol{\theta}} + \frac{1}{\sigma^2}\left\langle \frac{\partial y}{\partial \boldsymbol{\theta}} \right\rangle_{p(\mathbf{S}|\mathbf{X},\boldsymbol{\theta})}
\end{aligned}
$$

where $y = \frac{1}{2}||\mathbf{X} - \mathbf{f}(\mathbf{S})||^2$ is shorthand notation for the square term of the Gaussian noise and $\langle\cdot\rangle$ is average with respect to the source posterior. Defining the notation

$$\boldsymbol{\chi}_y \equiv \left\langle \frac{\partial y}{\partial \boldsymbol{\theta}} \frac{\partial y}{\partial \boldsymbol{\theta}^T} \right\rangle - \left\langle \frac{\partial y}{\partial \boldsymbol{\theta}} \right\rangle \left\langle \frac{\partial y}{\partial \boldsymbol{\theta}^T} \right\rangle$$

$$\boldsymbol{\varrho}_y \equiv \left\langle \frac{\partial^2 y}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right\rangle = \left\langle \mathbf{f}'^T \mathbf{f}' - (\mathbf{X} - \mathbf{f})^T \mathbf{f}'' \right\rangle \tag{3.17}$$

the hessians of the log likelihood and the bound can be expressed as follows assuming that the q distribution equals the source posterior as after an E-step,

$$\mathbf{H}_{\mathcal{L}} = \frac{1}{\sigma^4} \boldsymbol{\chi}_y - \frac{1}{\sigma^2} \boldsymbol{\varrho}_y \tag{3.18}$$

$$\mathbf{H}_B = \mathbf{H}_{\mathcal{L}} - \frac{1}{\sigma^4} \boldsymbol{\chi}_y = \frac{-1}{\sigma^2} \boldsymbol{\varrho}_y \tag{3.19}$$

Note that although $\boldsymbol{\varrho}_y$ contains the difference $(\mathbf{X} - \mathbf{f})$ it also contains a term which does not include this difference or other traces of the noise variance. Thus, $\boldsymbol{\varrho}_y$ does not scale with the noise variance and therefore, since $\mathbf{H}_B = -\boldsymbol{\varrho}_y/\sigma^2$, the Hessian of the bound is diverging for low noise levels. In some sense, this is concluding a very general argument that in low noise hidden variable models, the EM algorithm is 'freezing', i.e. updating the parameters infinitely slowly. But for the scaling of the Hessian $\mathbf{H}_B$ to be a serious problem – which is unique to the EM approach – one needs to demonstrate that the Hessian of the true log likelihood $\mathbf{H}_{\mathcal{L}}$, is not suffering from the same defect, i.e. is not also diverging in the the low noise limit. This point is supported by the following derivation, were p_s denotes the source prior density:

$$\begin{aligned}
\lim_{\sigma^2 \to 0} \mathbf{H}_{\mathcal{L}} &= \lim_{\sigma^2 \to 0} \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln p(\mathbf{X}|\boldsymbol{\theta}) \\
&= \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln \lim_{\sigma^2 \to 0} \int p(\mathbf{X}|\mathbf{S}, \boldsymbol{\theta}) p_s(\mathbf{S}) d\mathbf{S} \\
&= \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln \int \delta(\mathbf{X} - f_{\boldsymbol{\theta}}(\mathbf{S})) p_s(\mathbf{S}) d\mathbf{S} \\
&= \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln \frac{1}{|\mathbf{f}'(f_{\boldsymbol{\theta}}^{-1}(\mathbf{X}))|} p_s(f_{\boldsymbol{\theta}}^{-1}(\mathbf{X}))
\end{aligned}$$

These calculations assume, of course, that the limit of the noise and the two limits in the differentiation and the integration are interchangeable.

3.2.4.2 Discussion

The Hessian argument demonstrates a different way to understand the observed slow convergence of the EM algorithm. Using the general instantaneous observation model and therefore no special trade of the linear model, it provides a very general conclusion: Any hidden variable model with Gaussian noise, will have slow convergence using the EM algorithm. The only dodgy point is the general argument stating that the Hessian of the log likelihood does not diverge in the low noise limit, but in most non-pathologic setups this is in fact the case.

3.3 The Easy Gradient Recipe

This sections describes the so-called 'Easy Gradient Recipe' which, as the name suggests, is not an algorithm, but rather a recipe for constructing algorithms. It prescribes a way to obtain the likelihood and the gradient of the likelihood using the often simpler calculations involved in the EM algorithm. The likelihood and the corresponding gradient can then be used in advanced optimizers to obtain strongly improved results.

3.3.1 Details of the Recipe

The key idea is to regard the bound as a function of $\boldsymbol{\theta}$ only, instead of a function of both the parameters $\boldsymbol{\theta}$ and the variational parameters $\boldsymbol{\phi}$. Doing that the lower bound can be used as a reformulation of the log likelihood

$$\mathcal{L}(\boldsymbol{\theta}) = B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$$

where $\boldsymbol{\phi}_* = \boldsymbol{\phi}_*(\boldsymbol{\theta})$ fulfils the constraint $q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$. Since $\boldsymbol{\phi}_*$ is exactly minimizing the KL-divergence, the partial derivative of the bound with respect to $\boldsymbol{\phi}$ evaluated in the point $\boldsymbol{\phi}_*$, is equal to zero

$$q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta}) \quad \Rightarrow \quad \frac{\partial KL}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}_*} = 0 \quad \Rightarrow \quad \frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}_*} = 0$$

But setting the variational distribution equal to the source posterior has further consequences. The derivative of the Kullback-Leibler divergence with respect to the parameter $\boldsymbol{\theta}$ turn out to be zero, as simple calculations demonstrate

$$\begin{aligned}
\frac{dKL}{d\boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*} &= \frac{\partial KL}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*} + \frac{\partial KL}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}_*} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*} \\
&= \frac{\partial}{\partial \boldsymbol{\theta}} \left(\int q(\mathbf{S}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{S}|\boldsymbol{\phi})} d\mathbf{S} \right) \Big|_{\boldsymbol{\phi}_*} + 0 \\
&= \int q(\mathbf{S}|\boldsymbol{\phi}_*) \frac{1}{p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta})} \frac{\partial}{\partial \boldsymbol{\theta}} p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}) d\mathbf{S} \\
&= \frac{\partial}{\partial \boldsymbol{\theta}} \int p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}) d\mathbf{S} \quad = \quad \mathbf{0}
\end{aligned}$$

The second equality is due to the implication above and the fourth follows from the fact that $q(\mathbf{S}|\boldsymbol{\phi}_*)/p(\mathbf{S}|\mathbf{X}, \boldsymbol{\theta}) = 1$, which is of course also a consequence of the choice of $\boldsymbol{\phi}_*$. Using these results for the derivative of the bound and the

Kullback-Leibler divergence one is able to find the gradient of the log likelihood in the simple expression

$$\begin{aligned}
\frac{d\mathcal{L}(\boldsymbol{\theta})}{d\boldsymbol{\theta}} &= \frac{dB(\boldsymbol{\theta},\boldsymbol{\phi}_*)}{d\boldsymbol{\theta}} + \frac{dKL}{d\boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*} \\
&= \frac{\partial B(\boldsymbol{\theta},\boldsymbol{\phi}_*)}{\partial\boldsymbol{\theta}} + \frac{\partial B(\boldsymbol{\theta},\boldsymbol{\phi}_*)}{\partial\boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}_*} \frac{\partial\boldsymbol{\phi}}{\partial\boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*} \\
&= \frac{\partial B(\boldsymbol{\theta},\boldsymbol{\phi})}{\partial\boldsymbol{\theta}}\Big|_{\boldsymbol{\phi}_*}
\end{aligned}$$

In this way values and gradients of the true log likelihood can be obtained using the lower bound. The observation and use of this is not entirely new, as it is similar to an approach used in [47]. The novelty of the recipe is the practical recycling of computations already done using the EM algorithm for a much more efficient optimization carefully chosen for the occasion. Expressed in MATLAB-style pseudo-code one just need to construct the following: A function `fct` is receiving as argument the parameter $\boldsymbol{\theta}$ and returning both the function value and the gradient

$$\begin{array}{ll}
 & [\mathcal{L}, \frac{d\mathcal{L}}{d\boldsymbol{\theta}}] = \mathtt{fct}(\boldsymbol{\theta}) \\
\mathbf{1)} & \text{Find } \boldsymbol{\phi}^* \text{ such that } \frac{\partial B}{\partial\boldsymbol{\phi}}\big|_{\boldsymbol{\phi}^*} = 0 \\
\mathbf{2)} & \text{Calculate } \mathcal{L}(\boldsymbol{\theta}) = B(\boldsymbol{\theta},\boldsymbol{\phi}^*) \\
\mathbf{3)} & \text{Calculate } \frac{d\mathcal{L}(\boldsymbol{\theta})}{d\boldsymbol{\theta}} = \frac{\partial B(\boldsymbol{\theta},\boldsymbol{\phi}^*)}{\partial\boldsymbol{\theta}}
\end{array}$$

Compared to the EM algorithm, the advantage is that the function value and gradient can be fed to any advanced gradient based optimizer, which in most cases results in a strong improvement of the convergence properties. In that sense one is able to work with the easy framework of EM with respect to the computations while 'standing on the shoulders of giants' with respect to optimization.

Compared to working with directly with the log likelihood, the benefit of formulating the log likelihood using the bound function depends on the task at hand. In the PhD project of which this thesis is a major part, the Easy Gradient Recipe was developed to speed up the Mean Field ICA but turned out to work well also for Kalman filters. In the Kalman filter a brute force computation of the gradient of the log likelihood is very expensive, since computation of the gradients in that setting scales as d^2 times the cost of one Kalman Filter filter sweep. Using the formulation of the bound, it can be seen that computation of the gradient roughly scales as 2 times the cost of a Kalman filter sweep. Since the dimensionality d in State-space models often is very high this renders the Easy Gradient approach significantly less expensive. The somehow counter intuitive statement, that one can achieve substantial benefits from a reformulation of the problem, is also noted in [48] in their investigation of linear state space models.

The Easy Gradient Recipe is of course not restricted to ICA problems or Kalman filters. As it is clear from the derivation, the approach is possible for any maximum likelihood estimation with hidden variables as long as one can compute the bound. The combination of a widely applicable approach with the possibility of using state of the art optimization tools makes it a very strong tool for the community interested in practical maximum likelihood estimation.

3.3.2 Constrained Gradients of the Linear Model

Switching from the EM algorithm to a gradient based optimization, some of the control of the parameter updates is lost. Gradient based optimizers may use all kind of tricks and hacks to obtain a good result fast and one is not at all guaranteed that whatever constrains of the system are fulfilled by the next gradient based update. In this subsection issues about the constraint gradients for the linear model is discussed.

First a general property is considered regarding the gradients of the cost function B with respect to the model parameters $\boldsymbol{\theta}$. The cost function is a function of the model parameters $\boldsymbol{\theta}$ and the variational parameters $\boldsymbol{\phi}$ and any gradient must in principle take into account all dependencies. As an example the gradient with respect to the mixing matrix $\mathbf{A}$ is considered

$$\frac{dB}{d\mathbf{A}} = \frac{\partial B}{\partial \mathbf{A}} + \sum_{\{\boldsymbol{\phi}\}} \frac{\partial B}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{A}} + \sum_{\{\boldsymbol{\theta}\}\backslash\mathbf{A}} \frac{\partial B}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}}{\partial \mathbf{A}} \tag{3.20}$$

Since the model parameters are mutually independent, e.g. the mixing matrix is independent of the prior and the noise, the last sum is equal to zero. When choosing the variational parameters $\boldsymbol{\phi}$ to maximize B they are not independent of the model parameters $\boldsymbol{\theta}$, but instead, the variational parameters are by construction making the derivatives of B equal to zero,

$$\frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}\bigg|_{(\boldsymbol{\theta}, \boldsymbol{\phi}^*)} = \mathbf{0}$$

such that the first sum of the gradient is equal to zero too. By that argument, all the gradients of B with respect to the model parameters will, in the points $\boldsymbol{\phi}^*$ simply be the direct partial derivative of B. That is, if the evaluations in the optimal point is denoted by $\cdot|_*$ then

$$\frac{dB}{d\mathbf{A}}\bigg|_* = \frac{\partial B}{\partial \mathbf{A}}\bigg|_* \qquad \frac{dB}{d\mathbf{W}}\bigg|_* = \frac{\partial B}{\partial \mathbf{W}}\bigg|_* \qquad \frac{dB}{d\boldsymbol{\nu}}\bigg|_* = \frac{\partial B}{\partial \boldsymbol{\nu}}\bigg|_*$$

This significantly simplifies the calculations involved when the gradients are computed.

3.3.2.1 Constrained Parameters

The standard unconstrained gradient approach does not take into account that some of the parameters of the model must fulfil certain constrains to act as proper parameters. This is the case for the noise covariance, which must be positive definite to be a proper covariance, and e.g. the ratios in a mixture of gaussians that must be between zero and one and sum to one.

In order to incorporate these essential constrains into a standard gradient based unconstrained optimization, the constrained variables must be rewritten as functions of unconstrained variables and the underlying unconstrained variables be optimized. One could also, in a more Bayesian approach, construct priors on the parameters, which are zero in appropriate domains, but this is somewhat more troublesome and therefore not the approach followed in this thesis.

Mixing Matrix: In many situations there are obvious constrains on the mixing matrix. If in an overcomplete setting one will make a sparse representation of image data and therefore regard the columns of $\mathbf{A}$ as basis images, these will have to be positive to comply with a gray scale image interpretation. Positivity is fairly easy to obtain using the exponential function

$$(\mathbf{A})_{ij} = e^{(\boldsymbol{\alpha})_{ij}} \qquad \text{(Positive)}$$

It is now $\boldsymbol{\alpha}$ which is the variable which is optimized unconstrained. If the sources are speech recordings, fixed positions of the microphones will impose a geometrically originated constraint on the mixing matrix, and one should ideally incorporate this structure in the optimization. The constrains imposed by a geometrical knowledge of this kind, however, are to some extend situation specific, and for that reason omitted here.

Noise Covariance: The matrix representing the estimated noise covariance is considerably more complicated. For $\mathbf{W}$ to be a covariance matrix it must be positive definite, that is, $\mathbf{x}^T\mathbf{W}\mathbf{x} \geq 0$ for all $\mathbf{x}$. This constraint is difficult to implement directly, but some other structure which is sufficient to make it positive definite can be assumed.

$$\begin{array}{rcll}
\mathbf{W} & = & e^{\beta}\mathbf{I} & \text{(Isotropic)} \\
\mathbf{W} & = & \text{diag}(e^{\beta_1}, ..., e^{\beta_n}) & \text{(Diagonal)} \\
\mathbf{W} & = & \boldsymbol{\beta}\boldsymbol{\beta}^T & \text{(Positive Definite)}
\end{array}$$

In the last equation, $\boldsymbol{\beta}$ is assumed full rank but otherwise unconstrained. This way of writing the covariance matrix is sufficient and necessary in the sense that $\mathbf{W}$ is positive definite if and only if there exists a non-singular $\boldsymbol{\beta}$ such that $\mathbf{W} = \boldsymbol{\beta}\boldsymbol{\beta}^T$, see [49].

Prior Parameters: The constrains on the prior parameters cannot be described explicitly without choosing a certain prior. But most constrains are along the same lines as for the mixing and noise parameters: They must be positive or positive definite, and these constrains can be handled in the same way as described above. One important exception however, is the ratios in a mixture model, which must be between zero and one and sum to one. This can be obtained by defining the ratios using the unconstrained variables r_k in the following way

$$\rho_k = \frac{\exp(r_k)}{\sum_{k'} \exp(r_{k'})} \qquad \text{(Ratios)}$$

3.3.2.2 Gradients of Constrained Parameters

In this subsection the partial derivatives of the cost function B with respect to the constrained model parameters $\mathbf{A}, \mathbf{W}, \boldsymbol{\nu}$ will be computed, or rather, their underlying unconstrained counterparts. For the mixing matrix this involves both unconstrained mixing and positive mixing matrix and for the noise covariance different complicated structures are taken into consideration. For the prior parameters one must make a choice of prior distribution and in this thesis, a Laplace distribution and two variants of Mixture of Gaussians are considered.

Unconstrained Mixing Matrix: In the case of unconstrained mixing matrix, no special structure of $\mathbf{A}$ is assumed and therefore one has no additional variables to take care of. In that case, the result is easily found to be

$$\frac{\partial B}{\partial \mathbf{A}} = \mathbf{W}^{-1}\mathbf{A}\langle \mathbf{S}\mathbf{S}^T\rangle_q - \mathbf{W}^{-1}\mathbf{X}\langle \mathbf{S}\rangle_q^T \tag{3.21}$$

Positive Mixing Matrix: When assuming a positive mixing matrix, the new variable α_{ij} is introduced which is related to the mixing matrix through the equation $(\mathbf{A})_{ij} = e^{\alpha_{ij}}$. Thus, $\mathbf{A}$ is no longer regarded a variable but a function of the variable α_{ij} and the gradient is therefore expressed as

$$\frac{dB}{d\alpha_{ij}} = \sum_{kl} \frac{\partial B}{\partial A_{kl}} \frac{\partial A_{kl}}{\partial \alpha_{ij}} = \left[\frac{\partial B}{\partial \mathbf{A}}\right]_{ij} \cdot A_{ij}$$

Note that the notation $\partial B/\partial \mathbf{A}$ denotes an 'unconditioned' differentiation, i.e. a differentiation in which dependencies between variables are ignored.

The Noise Covariance: In all, three different kinds of structures of $\mathbf{W}$ is considered. All of them utilize the following expression in the computation of the gradient

$$\frac{\partial B}{\partial \mathbf{W}} = \frac{1}{2}\mathbf{W}^{-1} - \frac{1}{2}\mathbf{W}^{-1}(\mathbf{X}\mathbf{X}^T + \mathbf{A}\langle \mathbf{S}\mathbf{S}^T\rangle_q\mathbf{A}^T - \mathbf{X}\langle \mathbf{S}\rangle_q^T\mathbf{A} - \mathbf{A}\langle \mathbf{S}\rangle_q\mathbf{X}^T)\mathbf{W}^{-1} \tag{3.22}$$

Isotropic Covariance: The simplest structure of the covariance is an isotropic assumptions $\mathbf{W} = \sigma_\beta^2\mathbf{I} = e^\beta\mathbf{I}$. In this assumption the noise is both uncorrelated and the same in all directions. Al though this may seem restrictive it is a very commonly used assumption. The gradient with respect to β is

$$\frac{dB}{d\beta} = \sum_{kl} \frac{\partial B}{\partial W_{kl}} \frac{\partial W_{kl}}{\partial \beta} = e^\beta \mathrm{Tr}\left[\frac{\partial B}{\partial \mathbf{W}}\right]$$

Diagonal Covariance: More advanced is the uncorrelated noise assumption in which $\mathbf{W}$ is assumed diagonal with positive elements $(\mathbf{W})_{ij} = \delta_{ij} e^{\beta_i}$. Under this assumption the vector $\boldsymbol{\beta}$ is the variable which can be optimizes unconstrained and the gradient is given by

$$\frac{dB}{d\beta_i} = \sum_{kl} \frac{\partial B}{\partial W_{kl}} \frac{\partial W_{kl}}{\partial \beta_i} = \left[\frac{\partial B}{\partial \mathbf{W}}\right]_{ii} e^{\beta_i}$$

Positive Definite Covariance: The full covariance model, in which no special assumptions about the noise covariance structure is made, is the model where one only restricts $\mathbf{W}$ to be positive definite, i.e. any covariance matrix. One can now optimize unconstrained on the matrix $\boldsymbol{\beta}$ and the gradient is

$$\frac{dB}{d\boldsymbol{\beta}} = \sum_{kl} \frac{\partial B}{\partial W_{kl}} \frac{\partial W_{kl}}{\partial \boldsymbol{\beta}} = \left(\left[\frac{\partial B}{\partial \mathbf{W}}\right] + \left[\frac{\partial B}{\partial \mathbf{W}}\right]^T \right) \boldsymbol{\beta}$$

The Prior in General: If one also want to estimate the parameters of the source priors, the gradient given by the following needs to be calculated

$$\frac{dB}{d\boldsymbol{\nu}} = \int q(\mathbf{S}) \frac{\partial \ln p(\mathbf{S}|\boldsymbol{\nu})}{\partial \boldsymbol{\nu}} d\mathbf{S} = \left\langle \frac{\partial \ln p(\mathbf{S}|\boldsymbol{\nu})}{\partial \boldsymbol{\nu}} \right\rangle_q \tag{3.23}$$

This integral is obviously strongly dependent on the choice of prior and, for some but not all choices of prior distributions, rather complicated.

Exponential Prior: In the case of an exponential prior as defined in Eq. 2.4.2.3 the gradient is straight forward

$$\langle \frac{\partial \ln p(\mathbf{S}|\boldsymbol{\nu})}{\partial \nu_j} \rangle_q = \frac{\partial \ln \delta_{[\mathbf{S}>0]}}{\partial \nu_j} + \frac{N}{\nu_j} - \sum_t \langle \mathbf{S} \rangle_{q_{jt}}$$

where the first term is 0 if all elements of $\mathbf{S}$ are positive and infinite otherwise.

Mixture of Gaussians Prior: In the following, the case of zero-mean Gaussian mixtures is discussed - or rather, the case of mixtures of zero-mean gaussians. These distributions are important special cases of Gaussian mixtures well suited for flexible sparse priors. In the case of Mixture of gaussians prior the integral is surprisingly rather tricky.

$$\begin{aligned} \frac{\partial B}{\partial \boldsymbol{\nu}} &= \left\langle \frac{\partial \ln p(\mathbf{S}|\boldsymbol{\nu})}{\partial \boldsymbol{\nu}} \right\rangle_q \\ &= \sum_{t=1}^{N} \left\langle \frac{1}{\sum_{\kappa'} \rho_{\kappa'} \mathcal{N}_{\mathbf{s}_t}(\boldsymbol{\mu}_{\kappa'}, \mathbf{D}_{\kappa'})} \frac{\partial}{\partial \boldsymbol{\nu}} \sum_{\kappa} \rho_\kappa \mathcal{N}_{\mathbf{s}_t}(\boldsymbol{\mu}_\kappa, \mathbf{D}_\kappa) \right\rangle_q \end{aligned}$$

The challenge is that while for the gradients for $\mathbf{A}$ and $\mathbf{W}$ one needed the moments $\langle \mathbf{S} \rangle_q$ and $\langle \mathbf{S}\mathbf{S}^T \rangle_q$, now the moments of some very complicated function involving sums of exponentials in a fraction, must be known.

In order to circumvent this, a slight change of perspective with respect to the prior parameters is made. The prior parameter is simply fixed to be the same in $q(\mathbf{S})$ as in the prior $p(\mathbf{S}|\boldsymbol{\nu})$ at all times, i.e. $\boldsymbol{\nu}' \equiv \boldsymbol{\nu}$. Doing this one obtains

$$\begin{aligned}
\frac{dB}{d\boldsymbol{\nu}} &= \frac{\partial}{\partial\boldsymbol{\nu}} \int q(\mathbf{S}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{X},\mathbf{S}|\boldsymbol{\theta})}{q(\mathbf{S}|\boldsymbol{\phi})} d\mathbf{S} \\
&= \frac{\partial}{\partial\boldsymbol{\nu}} \int q(\mathbf{S}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{X}|\mathbf{S},\mathbf{A},\mathbf{W})p(\mathbf{S}|\boldsymbol{\nu})}{\frac{1}{Z_q} p(\mathbf{S}|\boldsymbol{\nu}) \exp[-\frac{1}{2}\mathrm{Tr}(\mathbf{S}\boldsymbol{\lambda}\mathbf{S}) + \mathrm{Tr}(\boldsymbol{\gamma}^T\mathbf{S})]} d\mathbf{S} \\
&= \frac{\partial}{\partial\boldsymbol{\nu}} \int q(\mathbf{S}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{X}|\mathbf{S},\mathbf{A},\mathbf{W})}{\frac{1}{Z_q} \exp[-\frac{1}{2}\mathrm{Tr}(\mathbf{S}\boldsymbol{\lambda}\mathbf{S}) + \mathrm{Tr}(\boldsymbol{\gamma}^T\mathbf{S})]} d\mathbf{S} \\
&= \frac{\partial}{\partial\boldsymbol{\nu}} \int q(\mathbf{S}|\boldsymbol{\phi}) \ln Z_q(\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\nu}) \\
&= \frac{\partial \ln Z_q}{\partial\boldsymbol{\nu}}
\end{aligned}$$

which is much easier since one can compute the normalization as is the case for gaussians and mixtures of gaussians. Often, when setting the variational parameters equal to the model parameters at all times, it complicate matters, but in this case it does exactly the opposite because this eliminates the need for averaging over complicated functions.

The gradients of the cost function – here expressed as the gradient of the partition function $\ln Z_q$ – must be calculated with respect to the unconstrained variables $\mathbf{r}$ and $\mathbf{w}$ – see below. Due to the appealing feature of mixture of gaussians that the partition function can in fact be obtained in a nice compact manner, these gradients can be computed with relative ease. Differentiation with respect to the mixture proportions yields

$$\frac{dB}{dr_{mn}} = \sum_{jl} \frac{\partial B}{\partial\rho_{jl}} \frac{\partial\rho_{jl}}{\partial r_{mn}} = \sum_{jl} \frac{\partial \ln Z_q}{\partial\rho_{jl}} \frac{\partial\rho_{jl}}{\partial r_{mn}}$$

$$\frac{\partial \ln Z_q}{\partial\rho_{jl}} = \sum_t \frac{\rho_{jl}\xi_{jlt}}{\sum_{k'} \rho_{jk'}\xi_{jtk'}} \frac{1}{\rho_{jl}} \qquad \frac{\partial\rho_{jl}}{\partial r_{mn}} = \rho_{jl}(\delta_{jm}\delta_{nl} - \rho_{mn})$$

$$\xi_{jlt} = \sqrt{\frac{1}{1+\sigma_{ik}^2\lambda_{ii}}} \exp\left[\frac{\gamma_{it}^2}{2} \frac{\sigma_{ik}^2}{1+\sigma_{ik}^2\lambda_{ii}}\right]$$

Differentiation with respect to the variances yields

$$\frac{dB}{dw_{jl}} = \frac{\partial B}{\partial\sigma_{jl}^2} \frac{\partial\sigma_{jl}^2}{\partial w_{jl}} = \frac{\partial \ln Z_q}{\partial\sigma_{jl}^2} \sigma_{jl}^2$$

$$\frac{\partial \ln Z_q}{\partial\sigma_{jl}^2} = \sum_t \frac{\rho_{jl}\xi_{jtl}}{\sum_{k'} \rho_{jk'}\xi_{jtk'}} \frac{1}{2} \frac{1}{(1+\sigma_{jl}^2\lambda_{jj})} \left[\frac{\gamma_{jt}^2}{1+\sigma_{jl}^2\lambda_{jj}} - \lambda_{jj}\right]$$

These equations together express the gradient of the unconstrained underlying variables of the Gaussian mixture.

3.3.3 The Optimizer UCMINF

With the gradient of the relevant unconstrained variables, the scene is now set for an unconstrained optimization. Any gradient based optimization is at disposal and quasi Newton approach with a number of tricks and hacks into one very efficient package has been chosen. The person behind the product is Hans Bruun Nielsen from Technical University of Denmark, and therefore the optimizer is sometimes referred to as the 'Hans Bruun' optimizer, although he in [33] named it UCMINF himself.

The UCMINF is constructed for nonlinear unconstrained optimization and is basically a quasi Newton optimizer with line search and trust regions. The quasi Newton part is a BFGS update starting from the identity matrix and the line search direction is the difference of two consecutive parameter values - see [33] for details.

Since the Easy Gradient approach with UCMINF is optimizing the true log likelihood and not the bound, it is not a true EM algorithm. Rather it is associated with EM variants such as LANGE [20] or QN2 [18]. The concept of explicitly using the EM to create the gradients for use on some method is also found in [47] in which a similar approach is implemented to produce function values and gradients of the log likelihood and these are used in a conjugated gradient approach for improved optimization.

Summary of Chapter 3

In this Chapter, the EM algorithm and various bound variants have been introduced. In the section on the low noise analysis it has been demonstrated, that both the traditional EM and the variational Bayes EM are 'freezing' the parameters update and thus making the combined algorithm intolerably slow. As a remedy, the Easy Gradient Recipe is introduced in Section 3.3 along with a discussion on how to ensure the constrains on the model parameters.

CHAPTER 4

Mean Field ICA

Chapter 2 presented the Bayesian probabilistic estimation for a number of hidden variable models with additive Gaussian noise. Among them were the so-called linear instantaneous model which for every time step is linearly mixing the hidden variables $\mathbf{s}_t$ into an observation vector $\mathbf{x}_t$, that is, $\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{n}_t$ for all $t = 1, ..., N$. If the prior over hidden variables $p_s(\mathbf{S}|\boldsymbol{\phi}_s)$ is chosen to be independent in the coordinates of the rows

$$p_s(\mathbf{S}|\boldsymbol{\phi}_s) = \prod_{i=1}^{d} p_i((\mathbf{S})_i|\boldsymbol{\phi}_s^i)$$

then estimating the parameters of this model is an Independent Component Analysis (ICA) technique. ICA is a so-called blind source separation technique, i.e. a technique for recovering sources which has been mixed in an unknown way. In the ICA framework, the hidden variables are regarded as sources in the following sense: Each coordinate of the hidden variables, considered over all time steps, S_{it} for $t = 1, ..., N$ is a signal of some sort which has been generated independently of the other signals S_{jt}, where $j \neq i$. These signals, being independent, are regarded as d different 'sources', which in audio applications could be different sound sources (voices, bird singing, engine noise, ...) mixed during the recording. ICA is a very successful and popular blind source separation technique with numerous applications – see [15] or [14] for some general introductions to ICA.

Since maximum likelihood is often troublesome to work with directly, Chapter 3 introduced the expectation-maximization (EM) algorithm and the Easy

Gradient Recipe as alternatives for practical optimization. Using the EM algorithm for estimation in ICA models, however, is nothing new. Seemingly, the authors of [3] were the first to use the EM algorithm for noisy ICA. In this paper the discrete source prior of QAM4 symbols – a prior which have applications in telecommunication – was applied, and the E-step of the EM algorithm was computed exactly due to the discreteness of the prior. For the noisy ICA model, the E-step of the EM algorithm consists of computing the first and second moments $\langle \mathbf{s}_t \rangle$ and $\langle \mathbf{s}_t \mathbf{s}_t^T \rangle$ of the often very complicated source posterior. Using QAM4 symbols, the number of combinations required in the source posterior moments is not unreasonably large, but one could imagine a larger set of symbols – taken to extremes a discretization of the prior space. This would make the model much more flexible with respect to the prior, while keeping the posterior means of the E-step exactly computable. But not computable in reasonable time, since for n symbols and d sources, the number of terms in the sum involved would be of order n^d, a number which very fast becomes prohibitively large.

Five years later, [30] tackled this problem in a way which is both efficient and fairly flexible. Using mixtures of gaussians (MoG) as source priors, the Gaussian noise makes the posterior itself a MoG and therefore the posterior moments of the E-step easily and exactly computable. A few years later in [1], the idea was further generalized and since then, many contributions have been using MoG as a prior. MoG's are very flexible and can, if enough components are included, model almost any continuous density. But, it suffers from the following two drawbacks: 1) one must evaluate a sum, which – with K components and d sources – for each time step involves K^d terms. For many realistic settings, this number becomes extremely large. And, 2) since a Gaussian is non-zero everywhere, it cannot for finite K model a prior which have zero density in some regions such as positive priors.

As an answer to the challenge that the number of terms scales as K^d, the so-called 'variational' or 'ensemble' approach sacrificed some of the exactness on the alter of efficiency. Early work using variational methods for ICA includes [21] and [22]. In these approaches, the complicated posterior distribution was approximated by a completely factorizing variational distribution. This is equivalent to maximizing a simpler lower bound on the likelihood instead of the likelihood itself. In effect, the K^d terms are substituted by dK terms and thus embracing a much larger class of problems for practical solutions.

This simplification paved the way for a solution to the second problem of the restrictions of the MoG's. The simplification by the variational approach allows for more complicated source priors because it splits the complicated integral over d dimensions into a product of d one-dimensional integrals. Getting rid of the high-dimensional challenge makes integral approximations such as the flexible and yet accurate Mean Field (MF) approach possible. The resulting variational MF approach was presented in [12] and it opened up for application of other source priors such as Laplace, exponential, uniform, and binary, among others.

This chapter, combines the content of the two previous chapters with an approximation of the posterior moments into the Mean Field ICA algorithm. In Section 4.1 the mean field theory, which approximates the posterior moments, is presented. Section 4.2 summarizes the Mean Field ICA algorithm along with examples of the the results regardring the convergence. Finally Section 4.3 presents the extension to convolutive mixtures.

4.0.3.1 The Posterior Moments

In the framework of linear transformations and Gaussian noise, the posterior mean and second moment of the sources are central. For the EM algorithm, the moments are the E-step and they are crucial for the estimation of both the noise and the mixing matrix because they are parts of the expression for the gradients. The source posterior moments are defined by

$$\begin{aligned} \langle \mathbf{S} \rangle &= \int \mathbf{S} \cdot p(\mathbf{S}|\mathbf{X}, \mathbf{A}, \mathbf{W}, \boldsymbol{\nu}) d\mathbf{S} \\ \langle \mathbf{S}\mathbf{S}^T \rangle &= \int \mathbf{S}\mathbf{S}^T \cdot p(\mathbf{S}|\mathbf{X}, \mathbf{A}, \mathbf{W}, \boldsymbol{\nu}) d\mathbf{S} \end{aligned}$$

Even integrals in one dimension can be an analytical challenge or a numerical difficulty. If the noise is not time-independent, the above integrals are formally Nd dimensional, where at least N, the number of data points is usually a very large number and d, the number of sources, in some applications is rather large too. In short, the above integrals can only be solved in a practical framework, if they to some degree factorize into many integrals of lesser dimension. The source posterior distribution is, by the use of Bayes theorem, given by

$$\begin{aligned} p(\mathbf{S}|\mathbf{X}, \mathbf{A}, \mathbf{W}, \boldsymbol{\nu}) &= \frac{p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \mathbf{W}) p(\mathbf{S}|\boldsymbol{\nu})}{p(\mathbf{X}|\mathbf{A}, \mathbf{W}, \boldsymbol{\nu})} \\ &= \frac{1}{Z} p(\mathbf{S}|\boldsymbol{\nu}) \exp\left[-\tfrac{1}{2}\mathrm{Tr}(\mathbf{S}^T\mathbf{J}\mathbf{S}) + \mathrm{Tr}(\mathbf{h}^T\mathbf{S})\right] \end{aligned}$$

where $\mathbf{J}$ and $\mathbf{h}$ is convenient notation for

$$\mathbf{J} = \mathbf{A}^T\mathbf{W}^{-1}\mathbf{A} \tag{4.1}$$
$$\mathbf{h} = \mathbf{A}^T\mathbf{W}^{-1}\mathbf{X} \tag{4.2}$$

Note that the source prior factorizes by assumption in both the sources and the time steps, and the second term of the exponential function involving $\mathbf{h}$ is obviously also factorizing. So if $\mathbf{J}$ is diagonal, the source posterior is completely factorized, but this is rarely the case for real world problems. For rather simple isotropic noise assumptions, $\mathbf{J}$ is only diagonal if the columns of $\mathbf{A}$ are orthogonal and this assumption is often too restrictive. The question whether or not $\mathbf{J}$ is diagonal is central in the discussion ahead, and therefore some convenient notation is introduced: $\tilde{\mathbf{J}}$ is the diagonal-free variant defined by $\mathbf{J} = \mathrm{diag}(\mathbf{J}) + \tilde{\mathbf{J}}$.

Thus the source posterior is in general only factorizing in the time steps and leaving the challenge of integrating in d dimensions. For most choices of noise and source prior this is not tractable and for a Gaussian noise model, only the Gaussian mixtures and discrete distributions provide nice analytical results. Therefore one must resort to approximations among which different Monte Carlo schemes or other sampling methods could be chosen, but common for these are that they are intolerably slow for practical purposes.

Instead, different variational approaches are presented. Approaches which have parallels in the Mean Field theory of statistical physics and which are related to the so-called Bound Formulation in Section 3.1.3. Basically the task is to find a variational distribution $q(\mathbf{S})$ which is sufficiently flexible to model the source posterior well and still computationally tractable with respect to the integrals. How to choose $q(\mathbf{S})$ is not at all trivial but the Mean Field theory from statistical physics – originally developed for magnetic materials in condensed matter physics – has turned out to be useful for this task.

4.1 Mean Field Theory

4.1.1 Naive Mean Field

The so-called Naive Mean Field (NMF) theory is not named in this way to indicate that it is useless or unreasonably inaccurate, but rather that it can be regarded as the starting point for more advanced approaches. The basic idea of the Naive Mean Field approach is to model a system without interactions – or rather to substitute the interaction with an additional external field. This corresponds to a completely factorized variational distribution

$$q(\mathbf{S}) = \prod_{t=1}^{N} \prod_{i=1}^{d} q_{it}(S_{it})$$

which is adapted to the source posterior. It can be done in two ways: 1) Choosing some suitable parametric form for the variational distribution and then minimizing the difference to the source posterior. This is referred to as the 'fixed form derivation'. Or 2) Make a constrained functional differentiation of the bound to find the optimal distribution, a procedure referred to as the 'freeform derivation'.

4.1.1.1 Fixed and Freeform Derivation

The fixed form derivation first: Enlightened by hindsight or experience from an attempted factorization of the source posterior, the following form for the variational distribution is chosen, where $\boldsymbol{\lambda}$ is assumed diagonal,

$$\begin{aligned} q(\mathbf{S}|\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\nu}') &= \frac{1}{Z_q} p(\mathbf{S}|\boldsymbol{\nu}') \exp\left[-\frac{1}{2}\mathrm{Tr}(\mathbf{S}^T\boldsymbol{\lambda}\mathbf{S}) + \mathrm{Tr}(\boldsymbol{\gamma}^T\mathbf{S})\right] \\ &= \prod_{t=1}^{N}\prod_{i=1}^{d} \frac{1}{Z_{it}} p(S_{it}|\nu_{it}) \exp\left[-\tfrac{1}{2}\lambda_{ii}S_{it}^2 + \gamma_{it}S_{it}\right] \end{aligned}$$

As described in the Bound Formulation in Section 3.1.3, optimization of the bound to the true log likelihood with respect to the variational parameters $\boldsymbol{\lambda}, \boldsymbol{\gamma}$ and $\boldsymbol{\nu}$, is done by minimizing the Kullback-Leibler divergence between the variational distribution and the source posterior. Doing this, the following solution is obtained

$$\begin{aligned} \lambda_{ii} &= J_{ii} \\ \gamma_{it} &= h_{it} - \textstyle\sum_k \tilde{J}_{ik}\langle S_{kt}\rangle_q \\ \nu'_{it} &= \nu_{it} \end{aligned}$$

While the equations for λ and ν are simple and in closed form, the equation for γ is dependent on the mean value $\langle S_{it}\rangle_q$, i.e. of the distribution q and its parameters. Thus the equations are not closed and a way to deal with this is to iterate the calculations as fixed point equations until convergence – more on that when the freeform derivation has been presented.

The freeform derivation is simply based on three assumptions only: 1) q must be a distribution, 2) q must be completely factorized and 3) q must optimize the bound B. Taking the last thing first, the optimization is obtained by the functional derivative of the marginal distribution $q(S_{it})$

$$\frac{\delta B}{\delta q(S_{it})} = 0$$

which gives

$$\begin{aligned} q(S_{it}) &\propto \exp\left[\langle \ln p(\mathbf{X},\mathbf{S}|\boldsymbol{\theta})\rangle_{\backslash it}\right] \\ &= \frac{1}{Z_q} p_i(S_{it}|\boldsymbol{\nu}') \exp\left[-\frac{1}{2}\lambda_{ii}S_{it}^2 + \gamma_{it}S_{it}\right] \end{aligned} \tag{4.3}$$

where $\boldsymbol{\nu}', \boldsymbol{\lambda}, \boldsymbol{\gamma}$ here formally are convenient placeholder variables to ease the notation

$$\boldsymbol{\lambda} = \mathrm{diag}(\mathbf{J}) \tag{4.4}$$
$$\boldsymbol{\gamma} = \mathbf{h} - \tilde{\mathbf{J}}\langle \mathbf{S}\rangle_q \tag{4.5}$$
$$\boldsymbol{\nu}' = \boldsymbol{\nu} \tag{4.6}$$

Note that the role of $\boldsymbol{\lambda}$ is similar to that of $\mathbf{J}$ but that $\boldsymbol{\lambda}$ is strictly diagonal, thus ignoring what is often referred to as the 'interaction' between the variables. Only the self-interaction is taken into consideration.

Writing the equations componentwise it is clear that they are equivalent to the result of the fixed form derivation. Thus, for a specific and reasonable choice of parametrization, fixed form and freeform derivation lead to the same result, but freeform derivation is a generally applicable approach which also works when one cannot choose cleverly by hindsight.

4.1.1.2 Integrals as Functions

Using a completely factorized model the intractable integral over d dimensions can be reduced to many integrals of one dimension, but depending on the source prior, this integral may still be very difficult to compute. The solution is to regard the integrals involved in the moments as a function – potentially a very complicated function – of the parameters

$$\begin{aligned}
\langle S_{it} \rangle &= \int S_{it} \frac{1}{Z_{it}} p(S_{it}|\nu_{it}) \exp[-\tfrac{1}{2}\lambda_{ii} S_{it}^2 + \gamma_{it} S_{it}] dS_{it} \\
&\equiv f(\lambda_{ii}, \gamma_{it}, \nu_{it}) \\
\langle S_{it}^2 \rangle &= \int S_{it}^2 \frac{1}{Z_{it}} p(S_{it}|\nu_{it}) \exp[-\tfrac{1}{2}\lambda_{ii} S_{it}^2 + \gamma_{it} S_{it}] dS_{it} \\
&= \frac{\partial}{\partial \gamma_{it}} f(\lambda_{ii}, \gamma_{it}, \nu_{it})
\end{aligned}$$

The crucial function $f(\lambda_{ii}, \gamma_{it}, \nu_{it})$ is different for each new source prior and there is no guarantee that there exists a nice closed expression for every different prior. But for a wide selection of relevant and flexible source priors, one can compute the function once and for all - see [12].

If the function $f(\lambda_{ii}, \gamma_{it}, \nu_{it})$ is reasonably efficient from a numerical perspective, then the coupled equation involving γ_{it} can be solved by repeated iterations as a fixed point algorithm

$$\begin{aligned}
\langle S_{it} \rangle_q &= f(\lambda_{ii}, \gamma_{it}, \nu_{it}) \\
\gamma_{it} &= h_{it} - \textstyle\sum_k \tilde{J}_{ik} \langle S_{kt} \rangle_q
\end{aligned}$$

In fact it can be shown that each coordinate in each step is increasing the bound and therefore a sequential update will make B converge under reasonably mild assumptions. After convergence is reached, a set of parameters $\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}$ and mean values $\langle \mathbf{S} \rangle_q, \langle \mathbf{S}\mathbf{S}^T \rangle_q$ are estimated which are consistent and an optimal approximation within the completely factorized model.

4.1.1.3 From the Viewpoint of Statistical Physics

So far the Naive Mean Field approach has only been considered from a variational probabilistic viewpoint. This section presents the same story with the terminology and concepts of statistical physics and does so not only to give due credit to the area from which it historically came, but also to set up the framework within which more advanced approaches are considered such as linear response (LR) and the Thouless, Anderson and Palmer (TAP) approximation.

To construct a Hamiltonian from the probabilistic framework, one must combine the fundamental relation between the energy function (the Hamiltonian) $H(\mathbf{S})$ and the probability of finding a physical system in that state. One can do this with Bayes theorem assuming a canonical ensemble and fixed unit temperature

$$\frac{e^{-\beta H(\mathbf{S})}}{Z} = p(\mathbf{S}|\mathbf{X}, \mathbf{A}, \mathbf{W}, \boldsymbol{\nu}) = \frac{p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \mathbf{W})p(\mathbf{S}|\boldsymbol{\nu})}{p(\mathbf{X}|\mathbf{A}, \mathbf{W}, \boldsymbol{\nu})}$$

This Hamiltonian now defines a physical system to which all sorts of methods from the repository of statistical physics can be applied

$$\begin{aligned} H(\mathbf{S}) &= -\ln\left[p(\mathbf{X}|\mathbf{S}, \mathbf{A}, \mathbf{W})p(\mathbf{S}|\boldsymbol{\nu})\right] && (4.7) \\ &= -\ln p(\mathbf{S}|\boldsymbol{\nu}) + \tfrac{1}{2}\mathrm{Tr}(\mathbf{S}^T\mathbf{J}\mathbf{S}) - \mathrm{Tr}(\mathbf{h}^T\mathbf{S}) + H_c && (4.8) \end{aligned}$$

where $\mathbf{J}, \mathbf{h}$ is defined by Eq. 4.1 and 4.2, and H_c by

$$H_c = \tfrac{1}{2}\mathrm{Tr}(\mathbf{X}^T\mathbf{W}^{-1}\mathbf{X}) + \tfrac{N}{2}\ln\det(2\pi\mathbf{W})$$

As the probabilistic framework had the problem that the model did not factorize, so does the physical counterpart. Since $\mathbf{J}$ is not diagonal, the variables S_{it}, which now are referred to as 'spins' in analogy with a magnetic system, are coupled through the energy function, H. To tackle this challenge, a different and simpler Hamiltonian, H_q, is defined

$$H_q(\mathbf{S}|\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}) = -\ln p(\mathbf{S}|\boldsymbol{\nu}) + \tfrac{1}{2}\mathrm{Tr}(\mathbf{S}^T\boldsymbol{\lambda}\mathbf{S}) - \mathrm{Tr}(\boldsymbol{\gamma}^T\mathbf{S})$$

$$\boldsymbol{\lambda} = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_d)$$

and this Hamiltonian has the property that the spins do not interact. Using Jensen's inequality one get the following bound on the partition function Z

$$\begin{aligned} Z &= \int e^{-H(\mathbf{S})} d\mathbf{S} \\ &= \frac{\int \exp[-(H(\mathbf{S}) - H_q(\mathbf{S}))]\exp[-H_q(\mathbf{S})]d\mathbf{S}}{\int e^{-H_q(\mathbf{S}'')}d\mathbf{S}''} \int e^{-H_q(\mathbf{S}')}d\mathbf{S}' \\ &= \langle \exp[-(H(\mathbf{S}) - H_q(\mathbf{S}))]\rangle_q \int e^{-H_q(\mathbf{S}')}d\mathbf{S}' \\ &\geq \exp[-\langle H(\mathbf{S}) - H_q(\mathbf{S})\rangle_q] \int e^{-H_q(\mathbf{S}')}d\mathbf{S}' \end{aligned}$$

Note that the above holds for *any* choice of H_q and choosing H_q to be non-interactive is just a choice of convenience. The result can be reformulated into

$$F \leq F_q + \langle H - H_q\rangle_q \equiv F_{var} \qquad (4.9)$$

in which F_{var} is referred to as the 'variational free energy' in contrast to the true free energy F. The true free energy is in fact the negative log likelihood and the variational free energy is the negative bound function

$$F_{var} = -B(\boldsymbol{\theta}, \boldsymbol{\phi}) = -\ln p(\mathbf{X}|\boldsymbol{\theta}) + KL(q_{\boldsymbol{\phi}}||p_{\boldsymbol{\theta}})$$

Using the variational free energy as a cost function, which must now be minimized, and considering it as a function of both the parameters and the moments, one get

$$\begin{array}{rclcrcl}
0 & = & \frac{\partial F_{var}}{\partial \langle S_{it}^2 \rangle_q} & \Rightarrow & \boldsymbol{\lambda} & = & \mathrm{diag}(\mathbf{J}) \\
0 & = & \frac{\partial F_{var}}{\partial \langle S_{it} \rangle_q} & \Rightarrow & \boldsymbol{\gamma} & = & \mathbf{h} - \tilde{\mathbf{J}} \langle \mathbf{S} \rangle_q \\
0 & = & \frac{\partial F_{var}}{\partial \boldsymbol{\lambda}} & \Rightarrow & \langle S_{it}^2 \rangle_q & = & \int \frac{e^{-H_q(S_{it})}}{Z_{it}} S_{it}^2 dS_{it} \\
0 & = & \frac{\partial F_{var}}{\partial \boldsymbol{\gamma}} & \Rightarrow & \langle S_{it} \rangle_q & = & \int \frac{e^{-H_q(S_{it})}}{Z_{it}} S_{it} dS_{it}
\end{array} \tag{4.10}$$

which are the so-called 'Mean Field Equations'. Iterating these equations until convergence and thereby regarding the moments as variables too, will after convergence produce a consistent set of mean values and parameters. Apart from being central to the mean field equations above, the variational free energy is a measure of convergence and interesting due to its connection with the bound on the log likelihood. It is computed by

$$\begin{aligned}
F_q &= -\ln \int e^{-H_q(\mathbf{S})} d\mathbf{S} \\
&= -\ln \prod_{t=1}^{N} \prod_{k=1}^{D} \int dS_{kt} P(S_{kt}) \exp[-\frac{1}{2}\lambda_{kt} S_{kt}^2 + \gamma_{kt} S_{kt}] \\
&= \sum_{t=1}^{N} \sum_{k=1}^{D} \Big(-\ln \int dS_{kt} P(S_{kt}) \exp[-\frac{1}{2}\lambda_{kt} S_{kt}^2 + \gamma_{kt} S_{kt}] \Big)
\end{aligned}$$

which is of course highly dependent on the choice of source prior. It is often averaged with respect to number of dimensions d and data points N to give comparable values between different setups.

The statistical physics viewpoint, defining the entire system by the equation $p = e^{-H}/Z$ may seem rather foreign and artificial to scientists with no background in physics. But in fact it can be viewed as nothing but a specific rewriting of the source posterior probability and variational distributions

$$p(\mathbf{S}|\mathbf{X}, ...) = \frac{e^{-H(\mathbf{S})}}{Z} \qquad q(\mathbf{S}|\boldsymbol{\phi}) = \frac{e^{-H_q(\mathbf{S})}}{Z_q}$$

not unlike the assumptions on the source prior in Section 2.4.2. There are no limitations or anything mystical about it, just a slight change of perspective and interpretation.

4.1.1.4 The Covariance

The methods described above gives the moment $\langle S_{it}\rangle_q$ and $\langle S_{it}^2\rangle_q$. In many contexts, the covariance of the sources is central and it can be rewritten with respect to both the true source posterior and the variational distribution q

$$\begin{array}{rcll} \chi_{ij}^{tt} & = & \langle S_{it}S_{jt'}\rangle - \langle S_{it}\rangle\langle S_{jt'}\rangle & \text{(Exact Posterior)} \\ \chi_{ij}^{tt} & = & \delta_{ij}(\langle S_{it}^2\rangle_q - \langle S_{it}\rangle_q^2) & \text{(Using NMF)} \end{array}$$

Thus not surprisingly, a completely factorized model gives diagonal covariance estimates. As an example with three sources, the covariance is

$$\boldsymbol{\chi}^t = \begin{bmatrix} \frac{\partial\langle S_{1t}\rangle_q}{\partial\gamma_{1t}} & 0 & 0 \\ 0 & \frac{\partial\langle S_{2t}\rangle_q}{\partial\gamma_{2t}} & 0 \\ 0 & 0 & \frac{\partial\langle S_{3t}\rangle_q}{\partial\gamma_{3t}} \end{bmatrix} \tag{4.11}$$

which will serve later as comparison with covariance estimates of more advanced models. Note that $\langle S_{it}^2\rangle_q - \langle S_{it}\rangle_q^2$ has been expressed as $\partial\langle S_{it}\rangle_q/\partial\gamma_{it}$. This relation is important and will be discussed in more detail in Section 4.1.2 which is about the so-called Linear Response.

4.1.1.5 More Complicated Models

The completely factorized model is not the only possible choice. In the so-called Bethe approximation, the variational distribution takes into account the dependencies between pairs of variables. Doing that, the interaction can be modeled more accurately and if the dependency structure is so-called tree-like the inference can be done exactly with a suitable choice of prior densities, see e.g. [54].

The principle can be expanded and in the Kikuchi approximation even higher order interactions are taken into account. These approaches gives at least in principle a better approximation, but are also much more analytically complicated and computationally costly, and are not considered in this thesis.

4.1.2 Linear Response

Due to the fact that the Naive Mean Field approach is a completely factorized model, i.e. the variables are completely independent, the covariance in Eq. 4.11 is diagonal. This is indeed the heart of the matter with respect to the simplification, but of course also the main problem if the true system has strong interactions. The linear response (LR) correction is a way to obtain more advanced estimates of the covariance – in particular estimates which are not diagonal.

For the true source posterior moments, the log likelihood is a moment generating function in the sense that the mean and covariance can be obtained as derivatives of the of the log likelihood with respect to the variable $\mathbf{h}$. Utilizing this one can connect the mean and covariance by

$$\langle \mathbf{S}_t \mathbf{S}_t^T \rangle - \langle \mathbf{S}_t \rangle \langle \mathbf{S}_t \rangle^T = \frac{\partial \langle \mathbf{S}_t \rangle}{\partial \mathbf{h}_t}$$

The connection between the covariance and the derivative of the mean with respect to the external field $\mathbf{h}_t$, is important. It is a connection between fluctuations (the covariance) and the systems susceptibility, i.e. sensibility of the mean to changes in $\mathbf{h}_t$. This connection does not automatically follow from any choice of system and variables. If one imagines some physical system defined by the Hamiltonian H which, among other variables, depends on the variable u, then

$$\frac{\partial \langle s \rangle}{\partial u} = \langle s \frac{\partial H}{\partial u} \rangle - \langle s \rangle \langle \frac{\partial H}{\partial u} \rangle$$

Since H_q depends linearly on the sources with respect to the variable γ, one could differentiate $\langle S_{it} \rangle_q$ with respect to $\gamma_{jt'}$. And indeed one *do* get the same covariance as in Eq. 4.11, but there is not much new obtained. The approach of the linear response correction is to assume the moments $\langle S_{it} \rangle_q$ to be good approximations of the true source posterior moments $\langle S_{it} \rangle$, when the parameters are optimized. Thus, considering the parameters $\boldsymbol{\lambda}, \boldsymbol{\gamma}$ and $\boldsymbol{\nu}'$ as functions of $\mathbf{h}_t$, one can mimic the connection of the true moments by the following calculations, where $\tilde{\boldsymbol{\chi}}^t$ is the new estimate of the covariance at the time step t

$$\begin{aligned}
(\tilde{\boldsymbol{\chi}}^t)_{ij} &= \frac{\partial \langle S_{it} \rangle_q}{\partial h_{jt}} \\
&= \sum_{i'j'} \frac{\partial \langle S_{it} \rangle_q}{\partial \lambda_{i'j'}} \frac{\partial \lambda_{i'j'}}{\partial h_{jt}} + \sum_{i't'} \frac{\partial \langle S_{it} \rangle_q}{\partial \gamma_{i't'}} \frac{\partial \gamma_{i't'}}{\partial h_{jt}} \\
&= \frac{\partial \langle S_{it} \rangle_q}{\partial \gamma_{it}} \frac{\partial \gamma_{it}}{\partial h_{jt}} \\
&= \frac{\partial \langle S_{it} \rangle_q}{\partial \gamma_{it}} \left(\delta_{ij} + \sum_k \tilde{J}_{ik} \frac{\partial \langle S_{kt} \rangle}{\partial h_{jt}} \right) \\
&= \frac{\partial \langle S_{it} \rangle_q}{\partial \gamma_{it}} \left(\delta_{ij} + \sum_k \tilde{J}_{ik} (\tilde{\boldsymbol{\chi}}^t)_{kj} \right)
\end{aligned}$$

The first and last equation are linear in $\tilde{\boldsymbol{\chi}}^t$ and can be solved to obtain an expression for the new estimate of the source covariance

$$\tilde{\boldsymbol{\chi}}^t = (\boldsymbol{\Lambda}_t + \tilde{\mathbf{J}})^{-1} \qquad (\Lambda_t)_{ij} = \delta_{ij} \left[\frac{\partial \langle S_{it} \rangle_q}{\partial \gamma_{it}} \right]^{-1}$$

Compared to the (inverse) covariance of the naive mean field approach, the linear response correction has the values of $\mathbf{J}$ on the non-diagonal entries on the inverse of the covariance. In a 3 by 3 example this means

$$\tilde{\chi}^t = \begin{bmatrix} \left(\frac{\partial \langle S_{1t}\rangle_q}{\partial \gamma_{1t}}\right)^{-1} & J_{12} & J_{13} \\ J_{21} & \left(\frac{\partial \langle S_{2t}\rangle_q}{\partial \gamma_{2t}}\right)^{-1} & J_{23} \\ J_{31} & J_{32} & \left(\frac{\partial \langle S_{3t}\rangle_q}{\partial \gamma_{3t}}\right)^{-1} \end{bmatrix}^{-1}$$

In the limit of no interactions, that is, in the limit where the off-diagonal elements of $\mathbf{J}$ becomes zero, the naive mean field approach becomes exact. Note that the linear response correction approach in this limit becomes equivalent to the exact source posterior too, and by this demonstrates some of the necessary fundamental properties.

4.1.3 Thouless, Anderson and Palmer - TAP

The Thouless, Anderson and Palmer (TAP) approach is a higher order approximation to the source posterior, which includes the naive mean field approximation as a special case. The basic idea is to reformulate the marginal source posterior such that the interaction can be Taylor expanded to whatever order appropriate. In that sense, the TAP approach aims at ending up with a marginal of the form

$$q_{it}(S_{it}) = \frac{1}{Z_{it}} p(S_{it}|\nu_{it}) \exp[-\tfrac{1}{2}\lambda_{it} S_{it}^2 + \gamma_{it} S_{it}]$$

as in the naive mean field approach, but with λ_{it} and γ_{it} defined differently and to higher order in the interactions. Note that in the naive mean field approach, the system is assumed completely factorized which is not the case for the TAP approach. Thus, although the marginal distribution of NMF and TAP have a similar form, their joint distributions are very different.

4.1.3.1 The Cavity Distribution

The derivation of the TAP approach starts out by considering the true source posterior. Using that the noise and source prior by assumption is factorizing in time, one obtains

$$
\begin{aligned}
p(\mathbf{S}|\mathbf{X},\mathbf{J},\mathbf{h}) &= \frac{1}{Z}\exp[-H(\mathbf{S}|\mathbf{X},\mathbf{J},\mathbf{h})] \\
&= \prod_{t=1}^{N}\frac{1}{Z_t}\exp(-H_t(\mathbf{S}_t)) \\
&= \prod_{t=1}^{N}\frac{1}{Z_t}\exp\left[-\tfrac{1}{2}\sum_{i,j=1}^{d}S_{it}J_{ij}S_{jt}+\sum_{i=1}^{d}h_{it}S_{it}+\sum_{i=1}^{d}\ln p(S_{it})-\frac{H_0}{N}\right] \\
&= \prod_{t=1}^{N}\left\{\frac{1}{Z_t}\exp\left[\sum_{j=1}^{d}(\tfrac{1}{2}\tilde{h}_{it}+h_{it})S_{it}\right]\prod_{i=1}^{d}\varrho_t(S_{it})\right\}
\end{aligned}
$$

where

$$
\begin{aligned}
\varrho_t(S_{it}) &= \exp\left[\ln p(S_{it})-\tfrac{1}{2}J_{ii}S_{it}^2\right] \\
\tilde{h}_{it} &= -\sum_{j=1}^{d}\tilde{J}_{ij}S_{jt}
\end{aligned}
$$

That is, $\tilde{h}_{it}$ is the effect on S_{it} as a result of the interaction with the other variables – or perhaps more correctly by the correlation between the variables induced by the observed data. With this notation the marginal source posterior is

$$
\begin{aligned}
& p(S_{it}|\mathbf{X},\mathbf{J},\mathbf{h}) \\
=\; & \langle p(\mathbf{S}_t|\mathbf{X},\mathbf{J},\mathbf{h})\rangle_{\setminus it} \\
=\; & \left\langle\frac{1}{Z_t}\exp\left[\sum_{j=1}^{d}(\tfrac{1}{2}\tilde{h}_{jt}+h_{jt})S_{jt}\right]\prod_{j=1}^{d}\varrho_t(S_{jt})\right\rangle_{\setminus it} \\
=\; & \varrho_t(S_{it})e^{h_{it}S_{it}}\left\langle\frac{1}{Z_t}\exp\left[\tfrac{1}{2}\tilde{h}_{it}S_{it}+\sum_{j\neq i}^{d}(\tfrac{1}{2}\tilde{h}_{jt}+h_{jt})S_{jt}\right]\prod_{j\neq i}^{d}\varrho_t(S_{jt})\right\rangle_{\setminus it}
\end{aligned}
$$

Note that the reaction field $\tilde{h}_{jt}$ is dependent on all other variables than j, and that $\tilde{h}_{jt}$ therefore is dependent on S_{it}. To obtain an expression which is as factorized as possible, the reaction field $\tilde{h}_{it}$ is rewritten such that the dependency on S_{it} becomes explicit, using the the so-called 'cavity field'

$$\begin{aligned}\sum_{j\neq i}^{d} \tfrac{1}{2}\tilde{h}_{jt}S_{jt} &= \sum_{j\neq i}^{d} \tfrac{1}{2}\left(-\sum_{k=1}^{d}\tilde{J}_{jk}S_{kt}\right)S_{jt} \\ &= \sum_{j\neq i}^{d} \tfrac{1}{2}\left(-\sum_{k\neq i}^{d}\tilde{J}_{jk}S_{kt} - \tilde{J}_{ji}S_{it}\right)S_{jt} \\ &= \sum_{j\neq i}^{d} \tfrac{1}{2}\tilde{h}_{jt}^{\backslash i}S_{jt} + \tfrac{1}{2}\tilde{h}_{it}S_{it}\end{aligned}$$

Inserting this into the expression for the marginal source posterior one obtains

$$\begin{aligned}& p(S_{it}|\mathbf{X},\mathbf{J},\mathbf{h}) \\ = \;& \varrho_t(S_{it})e^{h_{it}S_{it}}\left\langle \frac{1}{Z_t}\exp\left[\sum_{j\neq i}^{d}(\tfrac{1}{2}\tilde{h}_{jt}^{\backslash it} + h_{jt})S_{jt}\right] e^{\tilde{h}_{it}S_{it}}\prod_{j\neq i}^{d}\varrho_t(S_{jt})\right\rangle_{\backslash it} \\ = \;& \varrho_t(S_{it})e^{h_{it}S_{it}}\frac{Z_{\backslash it}}{Z_t}\langle e^{\tilde{h}_{it}S_{it}}\rangle_{\backslash it} \\ = \;& \frac{1}{Z_{it}}\varrho_t(S_{it})\exp[h_{it}S_{it} + E_{it}(S_{it})]\end{aligned}$$

where the function E_{it} is defined $E_{it}(S_{it}) = \ln\langle\exp[\tilde{h}_{it}S_{it}]\rangle_{\backslash it}$. So far no approximations have been made and essentially nothing but rewriting the problem has been done. The challenge of the interaction is now reformulated to the function E_{it} and since E_{it} looks rather difficult it may seem that very little is gained. But using a cumulant expansion to second order, one obtains an expression of the form

$$\begin{aligned}E_{it}(S_{it}) &= \ln\langle\exp[\tilde{h}_{it}S_{it}]\rangle_{\backslash i} \\ &= \sum_{k=1}^{\infty}\frac{c_{it}^{k}}{k!}S_{it}^{k} \\ &\cong \langle\tilde{h}_{it}\rangle_{\backslash i}S_{it} + \tfrac{1}{2}(\langle\tilde{h}_{it}^{2}\rangle_{\backslash i} - \langle\tilde{h}_{it}\rangle_{\backslash i}^{2})S_{it}^{2} \\ &\equiv \langle\tilde{h}_{it}\rangle_{\backslash i}S_{it} + \tfrac{1}{2}V_{it}S_{it}^{2}\end{aligned}$$

This derivation is the central approximation in the TAP approach: The interactions expressed by the function E_{it}, are assumed to be sufficiently accurately described by the mean and variance. The question of how the values for $\langle\tilde{h}_{it}\rangle_{\backslash i}$ and the fluctuations V_{it} should be obtained is still unaddressed, and will be answered in two very different ways: The mean $\langle\tilde{h}_{it}\rangle_{\backslash i}$ can be calculated directly, while the values V_{it} will be defined shortly. The mean of the reaction field $\langle\tilde{h}_{it}\rangle_{\backslash i}$

is given by the following calculations

$$\begin{aligned}
\langle\tilde{h}_{it}\rangle &= \int p(S_{it}|\mathbf{X},\mathbf{J},\mathbf{h})\tilde{h}_{it}dS_{it}\\
&= \int \frac{1}{Z_{it}}\varrho_t(S_{it})\exp[h_{it}S_{it}+E_{it}(S_{it})]\tilde{h}_{it}dS_{it}\\
&= \int \frac{1}{Z_{it}}\varrho_t(S_{it})\exp[h_{it}S_{it}+E_{it}(S_{it})]\frac{\partial}{\partial S_{it}}E_{it}(S_{it})dS_{it}\\
&\cong \int \frac{1}{Z_{it}}\varrho_t(S_{it})\exp[h_{it}S_{it}+E_{it}(S_{it})]\frac{\partial}{\partial S_{it}}\left[\langle\tilde{h}_{it}\rangle_{\setminus i}S_{it}+\tfrac{1}{2}V_{it}S_{it}^2\right]dS_{it}\\
&= \int \frac{1}{Z_{it}}\varrho_t(S_{it})\exp[h_{it}S_{it}+E_{it}(S_{it})]\left[\langle\tilde{h}_{it}\rangle_{\setminus i}+V_{it}S_{it}\right]dS_{it}\\
&= \langle\tilde{h}_{it}\rangle_{\setminus i}+V_{it}\langle S_{it}\rangle
\end{aligned}$$

that is,

$$\langle\tilde{h}_{it}\rangle_{\setminus i}\cong\langle\tilde{h}_{it}\rangle-V_{it}\langle S_{it}\rangle$$

Thus, disregarding that it is still not known how to estimate the fluctuations V_{it}, one can write the approximation for marginal source posterior as

$$\begin{aligned}
p(S_{it}|\mathbf{X},\mathbf{J},\mathbf{h}) &= \frac{1}{Z_{it}}\varrho_t(S_{it})\exp\left[h_{it}S_{it}+E_{it}(S_{it})\right]\\
&\cong \frac{1}{Z_{it}}\varrho_t(S_{it})\exp\left[h_{it}S_{it}+\langle\tilde{h}_{it}\rangle_{\setminus i}S_{it}+\tfrac{1}{2}V_{it}S_{it}^2\right]\\
&= \frac{1}{Z_{it}}\varrho_t(S_{it})\exp\left[(h_{it}+\langle\tilde{h}_{it}\rangle-V_{it}\langle S_{it}\rangle)S_{it}+\tfrac{1}{2}V_{it}S_{it}^2\right]\\
&= \frac{1}{Z_{it}}\varrho_t(S_{it})\exp\left[(h_{it}-\sum_k\tilde{J}_{ik}\langle S_{kt}\rangle-V_{it}\langle S_{it}\rangle)S_{it}+\tfrac{1}{2}V_{it}S_{it}^2\right]\\
&= \frac{1}{Z_{it}}\exp p(S_{it})\left[-\tfrac{1}{2}\lambda_i S_{it}^2+\gamma_{it}S_{it}\right]
\end{aligned}$$

where the parameters λ_{it} and γ_{it} are defined by

$$\begin{aligned}
\lambda_{it} &= J_{ii}-V_{it}\\
\gamma_{it} &= h_{it}-\sum_k\tilde{J}_{ik}\langle S_{kt}\rangle-V_{it}\langle S_{it}\rangle
\end{aligned}$$

The fluctuations V_{it} are estimated using the Expectation Consistency approach, also known as 'Adaptive TAP', described in the next subsection.

4.1.3.2 Adaptive TAP

The TAP approach is a higher order model in which the marginal source posterior $p(S_{it}|\mathbf{X},\mathbf{J},\mathbf{h})$ is estimated. With this marginal source posterior one can estimate $\langle S_{it}\rangle$ and $\langle S_{it}^2\rangle$ more accurately, but in order to use the TAP approximation in the ICA setup, estimating V_{it} and the rest of the covariance matrix is still missing. The Expectation Consistency approach is in some sense using the first problem to battle the second.

If one assumes for a moment the fluctuations V_{it} are known, then the diagonal of the covariance matrix is known. In a 3 by 3 example it would be

$$\boldsymbol{\chi}^t = \begin{bmatrix} \frac{\partial\langle S_{1t}\rangle_q}{\partial\gamma_{1t}} & ? & ? \\ ? & \frac{\partial\langle S_{2t}\rangle_q}{\partial\gamma_{2t}} & ? \\ ? & ? & \frac{\partial\langle S_{3t}\rangle_q}{\partial\gamma_{3t}} \end{bmatrix}$$

where "?" denotes that this entity has not been calculated yet. But, since the mean values $\langle S_{it}\rangle_q$ are known, this is not the only approach to computing the covariance matrix. One can also use the mean value in a linear response setup as with the naive mean field approach

$$\begin{aligned}
(\tilde{\boldsymbol{\chi}}^t)_{ij} &= \frac{\partial\langle S_{it}\rangle_q}{\partial h_{jt}} \\
&= \sum_{i'j'} \frac{\partial\langle S_{it}\rangle_q}{\partial\lambda_{i'j'}}\frac{\partial\lambda_{i'j'}}{\partial h_{jt}} + \sum_{i't'} \frac{\partial\langle S_{it}\rangle_q}{\partial\gamma_{i't'}}\frac{\partial\gamma_{i't'}}{\partial h_{jt}} \\
&= \frac{\partial\langle S_{it}\rangle_q}{\partial\gamma_{it}}\frac{\partial\gamma_{it}}{\partial h_{jt}} \\
&= \frac{\partial\langle S_{it}\rangle_q}{\partial\gamma_{it}}\left(\delta_{ij} - \sum_k (\tilde{J}_{ik} + \delta_{ik}V_{it})\frac{\partial\langle S_{kt}\rangle}{\partial h_{jt}}\right) \\
&= \frac{\partial\langle S_{it}\rangle_q}{\partial\gamma_{it}}\left(\delta_{ij} - \sum_k (\tilde{J}_{ik} + \delta_{ik}V_{it})(\tilde{\boldsymbol{\chi}}^t)_{kj}\right)
\end{aligned}$$

As with the linear response correction for the naive mean field approach, this is a linear equation in $\tilde{\boldsymbol{\chi}}^t$ and has the solution

$$\tilde{\boldsymbol{\chi}}^t = (\boldsymbol{\Lambda}^t + \tilde{\mathbf{J}})^{-1}, \qquad \Lambda^t_{ij} = \delta_{ij}\left(\frac{\partial\langle S_{it}\rangle_q}{\partial\gamma_{it}} + V_{it}\right) \tag{4.12}$$

That is, in a 3 by 3 example, the linear response solution to the TAP approach is

$$\tilde{\boldsymbol{\chi}}^t = \begin{bmatrix} \left(\frac{\partial\langle S_{1t}\rangle_q}{\partial\gamma_{1t}}\right)^{-1} + V_{it} & J_{12} & J_{13} \\ J_{21} & \left(\frac{\partial\langle S_{2t}\rangle_q}{\partial\gamma_{2t}}\right)^{-1} + V_{2t} & J_{23} \\ J_{31} & J_{32} & \left(\frac{\partial\langle S_{3t}\rangle_q}{\partial\gamma_{3t}}\right)^{-1} + V_{3t} \end{bmatrix}^{-1}$$

Now two seemingly different estimates of the diagonal of the covariance is in play. The idea of the EC (aka Adaptive TAP) is to adjust the fluctuations V_{it} to make these two estimates equal and thereby making the linear response solution consistent. That is, adjust V_{it} such that

$$\mathrm{diag}(\boldsymbol{\chi}^t) = \mathrm{diag}(\tilde{\boldsymbol{\chi}}^t)$$

In practical terms it is not possible to invert the matrix $\tilde{\boldsymbol{\chi}}^t$ to obtain a closed form expression for V_{it}. Instead, one iterates the fixed point equation

$$V_{it}^{(n+1)} = V_{it}^{(n)} + \frac{1}{\chi_{ii}^t} - \frac{1}{\tilde{\chi}_{ii}^t}$$

When convergence has been reached, an estimate of V_{it} is given which provides a non-diagonal covariance estimate while at the same time keeping the direct covariance and the linear response approach consistent.

4.1.4 Source Models and Mean Functions

Interpreting the integral of the moment as a function of the parameters is of course no magic trick which can make all the problems disappear at once. Any integral involving some parameters is formally a function of these parameters (at least if it converges) and in that sense nothing has been solved.

But if the integral can be expressed as some function, which evaluates much faster than a sampling to approximate the integral, then quite a bit has been gained in terms of computational speed. As an example, one can consider the case of a binary source prior. For a binary source prior, the Gaussian noise assumption leads to a mean function which is especially simple, namely

$$\begin{aligned} f(\lambda_{it}, \gamma_{it}, \nu_{it}) &= \int S_{it} \frac{1}{Z_q} p(S_{it}) \exp[-\tfrac{1}{2}\lambda_{it} S_{it}^2 + \gamma_{it} S_{it}] dS_{it} \\ &= \tanh(\gamma_{it}) \end{aligned}$$

Inserting the expression for the mean value and the optimal gamma parameter, one obtains the equation

$$\langle S_{it} \rangle_q = \tanh[h_{it} - \textstyle\sum_k \tilde{J}_{ik} \langle S_{kt} \rangle_q]$$

In this expression, h_{it} and $\tilde{J}$ are constants and one can iterate until the value of $\langle S_{it} \rangle_q$ is converged. Doing this, a difficult integral has essentially been substituted with a repeated evaluation of a nonlinear function – a tradeoff which in almost all cases will speed up the computational time. Not all source priors lead to nice compact expressions like the above, but a fairly large and flexible selection is in fact available - see below for some of the important examples and [12] for a wider selection.

The exponential quadratic form in the variational distribution stems from the Gaussian noise assumption and this influences strongly which source priors give nice analytical results. With the binary case as an exception, mostly source priors which to some degree are related to the exponential family, give closed form results. This relationship to the Gaussian distribution makes the following functions relevant, following the notation from [12]

$$D(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5x^2} \qquad \Phi(x) = \int_{-\infty}^{x} e^{-0.5t^2} dt$$

Using this notation the mean function for three important source priors is presented. The mean functions are included here for a set of important source priors to give the reader a better understanding of the matter using some concrete examples.

4.1.4.1 Exponential

The so-called exponential prior is used for sources which are positive and sparse. The probability density is defined by

$$p(S_{it}) = \eta e^{-\eta S_{it}}, \qquad S_{it} > 0$$

and zero if $S_{it} < 0$. Dividing the integrals at suitable boundaries one can compute the mean function using the functions $D(x)$ and $\Phi(x)$. The result is

$$f(\lambda_{it}, \gamma_{it}, \nu_{it}) \quad = \quad \frac{1}{\sqrt{\lambda_{it}}} \frac{\xi \Phi(\xi) + D(\xi)}{\Phi(\xi)}, \qquad \xi = \frac{\gamma_{it} - \eta}{\sqrt{\lambda_{it}}}$$

As with the other source priors, the function for the second moments are the derivative of the mean function with respect to γ_{it}. This function can, along with the second moment function for the other source priors, be found in [12].

4.1.4.2 Laplace

The Laplace source distribution is a heavy tailed symmetric distribution with the density

$$p(S_{it}) = \frac{\eta}{2} e^{-\eta |S_{it}|}$$

It is often used as a prior for speech or other sparse and symmetric sources. The mean function can be calculated using the functions $D(x)$ and $\Phi(x)$, and the result is

$$\begin{aligned} f(\lambda_{it}, \gamma_{it}, \nu_{it}) &= \frac{1}{\sqrt{\lambda_{it}}} \frac{\xi_+ \kappa_+ + \xi_- \kappa_-}{\kappa_+ + \kappa_-} \\ \xi_\pm &= \frac{\gamma_{it} \mp \eta}{\sqrt{\lambda_{it}}} \\ \kappa_\pm &= \Phi(\pm \xi_\pm) D(\xi_\mp) \end{aligned}$$

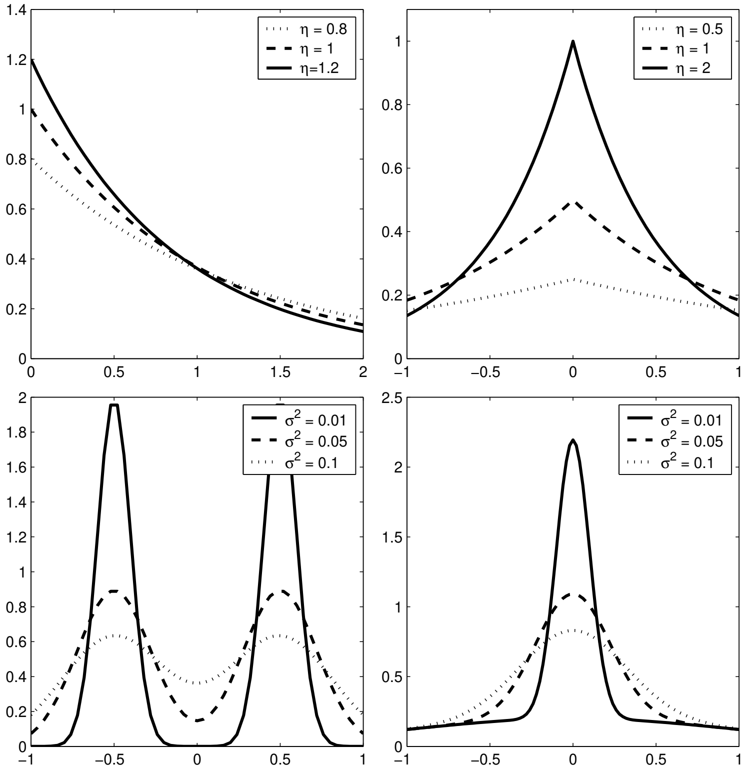


Figure 4.1: Examples of source priors: Exponential (top left), Laplace (top right), Bimodal MoG (Bottom left), and Sparse MoG (bottom right).

For the second moment one needs to differentiate the mean function with respect to γ_{it} – see e.g. [12] for more details.

4.1.4.3 Mixture of Gaussians, MoG

While the exponential and the Laplacian distributions are rather limited in the form presented above because they are assumed to have zero mean – the mixture of Gaussians considered here is an extremely flexible source prior. In fact, one can in principle model any other density which is nonzero everywhere using a mixture of gaussians. The density is defined by

$$p(S_{it}) = \sum_{k=1}^{K} \frac{\rho_k}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{1}{2}\frac{(S_{it}-\mu_k)^2}{\sigma_k^2}\right]$$

Since they are gaussians, their integral with the Gaussian of the noise can be computed analytically and without the special functions used for the other priors. The result is

$$f(\lambda_{it}, \gamma_{it}, \nu_{it}) = \sum_k u_k \frac{\gamma_{it}\sigma_k^2 + \mu_k}{\lambda_{it}\sigma_k^2 + 1}$$

where

$$\begin{aligned} u_k &= \frac{\kappa_k \exp(\xi_k)}{\sum_{k'} \kappa_{k'} \exp(\xi_{k'})} \\ \kappa_k &= \frac{\rho_k}{\sqrt{\lambda_{it}\sigma_k^2 + 1}} \\ \xi_k &= -\frac{1}{2}\left[\frac{\mu_k^2}{\sigma_k^2} - \frac{(\gamma_{it}\sigma_k + \mu_k\sigma_k)^2}{\lambda_{it}\sigma_k^2 + 1}\right] \end{aligned}$$

In fact there is, of course, strictly speaking no need for mean field theory when using mixtures of gaussians, since the nice analytical properties of the Gaussian density ensures that the mean values can be calculated exactly.

4.1.4.4 Other Source Priors

One could compute all sorts of source priors for whatever purpose. As mentioned not all source priors result in an analytical mean function, as not all mean functions have an analytical prior. But more can be computed than is presented here. In [12] one can also find mean functions for binary (very easy), bi-Gaussian special case, uniform prior, general heavy tailed and positive Gaussian.

4.1.5 Accuracy

While the use of mixture of gaussians as source priors makes it possible to calculate the source posterior moments $\langle \mathbf{S} \rangle$ and $\langle \mathbf{S}\mathbf{S}^T \rangle$ exactly, the Variational approach doesn't. A reasonable question therefore is: How accurate is the variational approximation to the source posterior moments? If the true source posterior is factorizing completely, i.e. if $\mathbf{J} = \mathbf{A}^T \mathbf{\Sigma}^{-1} \mathbf{A}$ is diagonal, then the variational distribution q becomes exactly the source posterior and the variational factorized approach becomes exact. When $\mathbf{J}$ is not diagonal, the variational approach cannot model the moments exactly, but still have a certain compensating ability which may work well for weak dependencies. In this subsection, the accuracy of the variational approximation is investigated when challenged with non-zero dependencies. The dependency is quantified in the following way for a system with three sources[1]:

$$\mathbf{J} = \mathbf{A}^T \mathbf{\Sigma}^{-1} \mathbf{A} = \begin{bmatrix} 1 & \alpha & \alpha \\ \alpha & 1 & \alpha \\ \alpha & \alpha & 1 \end{bmatrix}$$

That is, α expresses the dependency between the variables in the true source posterior: $\alpha = 0$ means no dependency and the factorized approximation is exact, and $\alpha = 1$ means maximum dependency and the factorized approximation is presumably bad.

In Figure 4.2, the exact moments $\langle S_{it} \rangle$ are plotted against the variational approximation for different combinations of mixing matrix dimensions and degree of dependency. The representative examples are plotted of the matrix dimensions 2x3, 3x3 and 4x3, and $\alpha = 0.50$, 0.90, and 0.98. For weak dependencies ($\alpha = 0.50$), the 3x3 and 4x3 examples show very good accuracy while the underdetermined 2x3 setting does not. This indicates that the underdetermined case is posing a special challenge not only with respect to source recovery but also with respect to the accuracy of the source posterior moments. For moderate dependencies ($\alpha = 0.90$), the accuracy of the square and overdetermined case becomes poor, and for strong dependencies ($\alpha = 0.98$), the approximation breaks down as expected.

In [39] it is demonstrated for binary sources that in the limit of a factorized true source posterior, the off-diagonal elements of $\boldsymbol{\chi}$ in LR are more accurate than in the completely factorized model (aka Naive Mean Field, NMF), but that this is only the case for the diagonal elements when the so-called 'external field' $\mathbf{h}_t$ is not too small. Other more general but less detailed arguments, as the Parisi argument presented in [35], expresses the accuracy in the difference ε between the true source posterior and the variational distribution and shows how the error on the covariance of the factorized model is $\mathcal{O}(\varepsilon)$ while the covariance obtained by LR is $\mathcal{O}(\varepsilon^2)$.

An investigation of the accuracy has been done by constructing 1000 random mixing matrices (2x3, 3x3 and 4x3) with columns normalized to length 1. Using MoG's as source priors, the exact source posterior covariance can be computed along with the factorized and the LR correction variants. Comparing the elements of these approximations using root mean square error,the result in table 4.1 is obtained. The table shows that for 3x3 and 4x3, the LR is often the better estimator on the source posterior second moments $\langle S_{it} S_{jt'} \rangle$, while for the underdetermined 2x3 case, it is the factorized model which is most accurate. Note that when LR is better it is by a small margin, while the cases where LR is not an improvement, the error is relatively large, making the average unfavorable for the LR approach.

But, as a sneak preview of the combined algorithm, Figure 4.3 shows a crucial benefit of using linear response. In this data set, which is a mixing of

[1] For a 2x3 case, it is not possible to construct exactly this form for all choices of dependency. In those cases, α expresses the value of the largest non-diagonal entry.

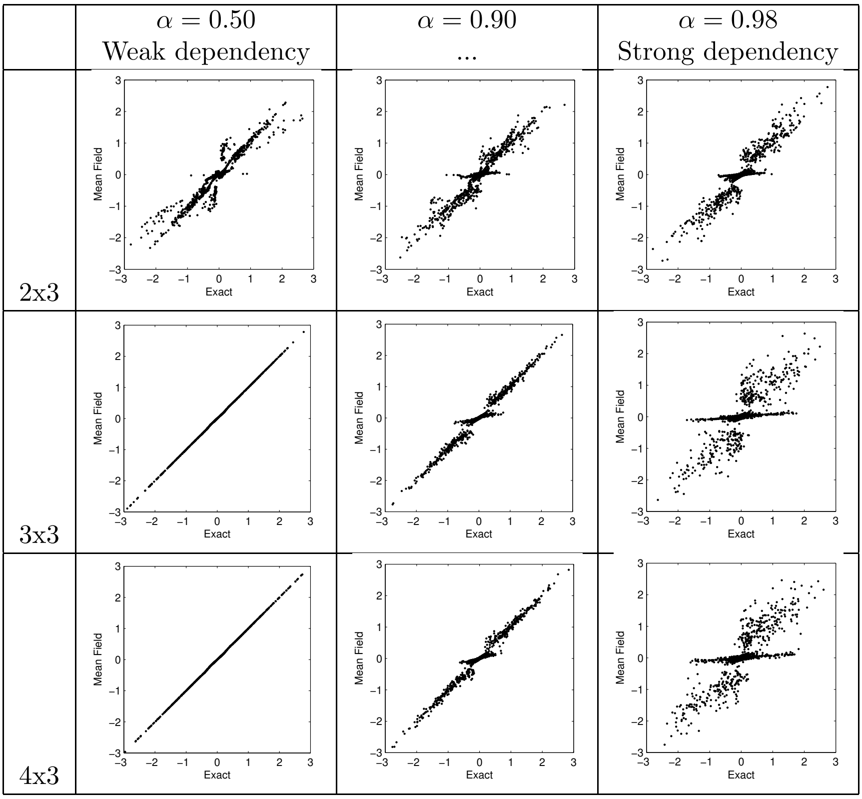


Figure 4.2: Accuracy of the variational approach: The moments $\langle S_{it} \rangle$ have been computed for both the exact soruce posterior and the variational approximation, and plotted against each other such that a 'straight diagonal line' means a perfect fit. In the case of square and overdetermined mixing and weak dependency, the variational approximation is very accurate, while the other cases are proving increasingly difficult. Note that at $\alpha = 0.90$, the approximation is still reasonable for square and overdetermined, while the underdetermined (overcomplete) is posing problems for all dependencies.

		Factorized	LR
2x3	Mean Error $\langle S_{it}S_{jt}\rangle$	0.0088	0.0313
	Median Error $\langle S_{it}S_{jt}\rangle$	0.0069	0.0115
	Best Estimator	**97.3%**	2.7%
3x3	Mean Error $\langle S_{it}S_{jt}\rangle$	0.0015	0.0039
	Median Error $\langle S_{it}S_{jt}\rangle$	$6.1 \cdot 10^{-5}$	$5.3 \cdot 10^{-5}$
	Best Estimator	35.9%	**64.1%**
4x3	Mean Error $\langle S_{it}S_{jt}\rangle$	0.0003	0.0007
	Median Error $\langle S_{it}S_{jt}\rangle$	$1.4 \cdot 10^{-5}$	$0.8 \cdot 10^{-5}$
	Best Estimator	11.5%	**88.5%**

Table 4.1: Accuracy of linear response (LR) versus the completely factorized approach. For each of the settings 2x3, 3x3 and 4x3, a random set of 1000 mixing matrices with columns normalized to length one was generated. Using MoG source priors, the error of the factorized and LR approach were calculated as a root mean square error. Note that LR is typically (median) doing better for square and overdetermined case, but not on average (mean). For 2x3 the factorized approach is most accurate, but later examples show that LR still offers the best overall performance of the algorithm.

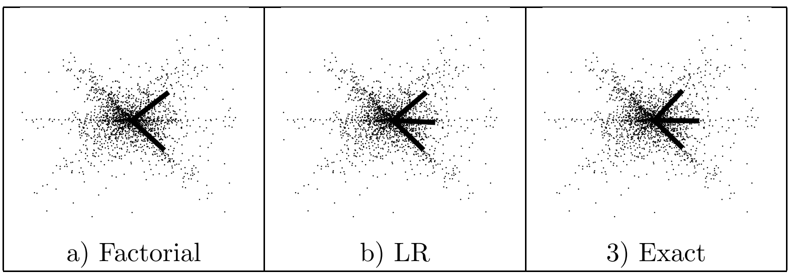


Figure 4.3: Three speech signals in two channels. In this example, the speech signals were modelled using MoG priors enabling us to compare the factorized and LR approach with the exact posterior moments. The factorized approach 'turns off' one of the channels, i.e. is setting a column in the mixing matrix to zero, and thereby finds only two of the three directions in data. The LR and the exact approaches, on the other hand, finds all three directions. This demonstrates that LR is an important extension to the factorized variational approach.

three speech signals into two channels, the completely factorized approach is not able to find all three columns of the mixing matrix. Instead it 'turns off' one of the columns by setting it to zero and find two of the remaining directions, Figure 4.3 (a). Linear response on the other hand is able to find all directions as is the exact method without the variational approximation, Figure 4.3 (c). In this sense, the linear response is outperforming the factorial approach also in the overcomplete setting even if the numerical accuracy is not better.

An in depth general analysis to why this is the case is not given here, but the following argument may shed some light on the process. In a 2x3 example with isotropic noise and known noise variance, one may assume the first column of the mixing matrix to be zeros. This ensures, that the interaction matrix $\mathbf{J}$ corresponding to the mixing matrix, is zero in the first column and first row and that the remaining columns – if chosen to be orthogonal – can fulfil the factorization constraint and make $\mathbf{J}$ diagonal

$$\mathbf{J} = \mathbf{A}^T\mathbf{W}^{-1}\mathbf{A} = \frac{1}{\sigma^2}\begin{bmatrix} 0 & 0 & 0 \\ 0 & \mathbf{a}_2^T\mathbf{a}_2 & 0 \\ 0 & 0 & \mathbf{a}_3^T\mathbf{a}_3 \end{bmatrix}$$

where $\mathbf{a}_i$ are the column vectors of $\mathbf{A}$. In this case, the top row of the external field $\mathbf{h} = \mathbf{A}^T\mathbf{W}^{-1}\mathbf{X}$ will also be zeros, and the parameters λ_{11} and γ_{1t} of the variational q-distribution are all zero. Therefore, $q(S_{1t})$ is simply the prior, which in the case of the mixed speech signals are zero-mean, such that $\langle S_{1t}\rangle_q = 0$ for all $t = 1, ..., N$. The next update of the mixing matrix involved the data, presumably 'corrects' the situation, but the update of $\mathbf{A}$, i.e. $\mathbf{A} = \mathbf{X}\langle\mathbf{S}\rangle_q\langle\mathbf{S}\mathbf{S}^T\rangle_q^{-1}$ is in fact not changing anything: Let $\mathbf{G} = \langle\mathbf{S}\mathbf{S}^T\rangle_q$, such that when q is completely factorized and $\langle S_{1t}\rangle_q = 0$, then

$$\mathbf{G}^{-1} = \begin{bmatrix} G_{11} & 0 & 0 \\ 0 & G_{22} & G_{23} \\ 0 & G_{32} & G_{33} \end{bmatrix}^{-1} = \begin{bmatrix} G_{11}^{-1} & 0 & 0 \\ 0 & \tilde{G}_{33} & -\tilde{G}_{23} \\ 0 & -\tilde{G}_{32} & \tilde{G}_{22} \end{bmatrix}$$

where $\tilde{G}_{ij} = G_{ij}/(G_{22}G_{33} - G_{23}G_{32})$. Inserting this into the update expression for $\mathbf{A}$ reveals that the updated mixing matrix still has zeros in the first column. That is, the first column having zeros, together with the constraint that the second and third column are orthogonal, is a fix point for the algorithm. This simplified example explains the behavior observed in Figure 4.3 a), where one column of the estimated mixing matrix ends as zeros, and the two remaining columns orthogonal.

4.2 Mean Field ICA

The probabilistic estimation presented in Chapter 2 can now be linked to the approximate bound method presented in Chapter 3 and the mean field approaches

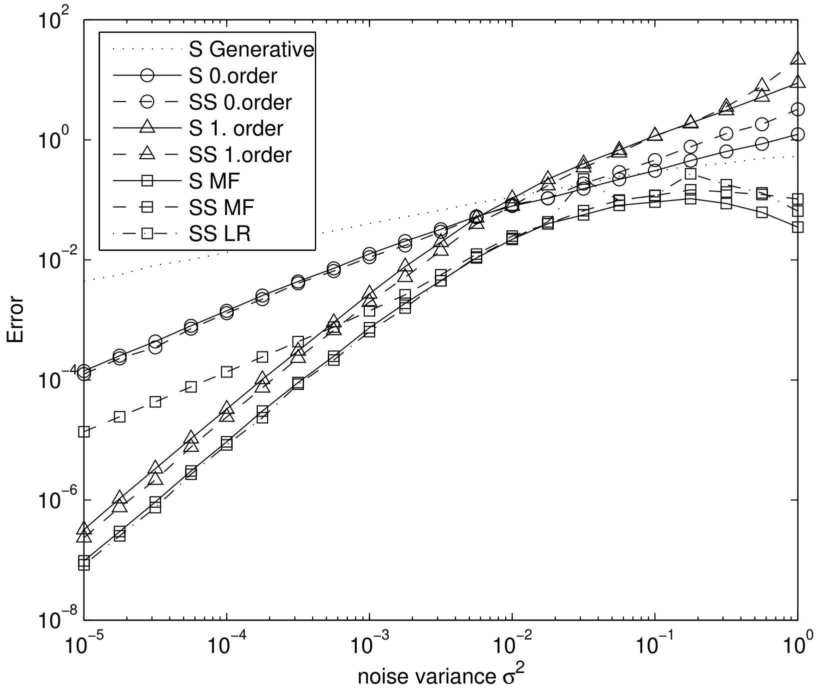


Figure 4.4: Accuracy of Factorized approach (MF), Linear response (LR) and zeroth and first order saddle point approximation. Using a MoG source prior the exact moments can be computed as well as the different sets of approximations. The error is, $Err = \sqrt{\sum_{it}(\langle S_{it}\rangle_{exc} - \langle S_{it}\rangle_{app})^2}/(Nd)$. Apart from the above mentioned approximations, also the generative source have been included for perspective. That is, the true generative source has been used as an approximation to the source posterior mean, to see how well the exact posterior moments matches the entity which it ultimately is used to estimate. Interesting in this examples is that the linear response correction works well (and better than the factorized approach) for all noise levels.

in Section 4.1. The result is the Mean Field ICA, a flexible and yet powerful algorithm for blind source separation.

4.2.1 The Algorithm

Chapter 2 presented a way to do probabilistic estimation disregarding all practical problems with respect to integration and optimization. The formal result of the approach for the linear instantaneous model is the non-linear fixed point equations for the mixing matrix, noise covariance and the source prior parameters, which are expressed using integrals. The integrals are the averages with respect to the source posterior – a distribution which in most cases is very complicated rendering the integrals also very complicated.

Chapter 3 presented the EM algorithm, which in the bound formulation of Section 3.1.3 makes it possible to substitute the complicated source posterior averages with averages over the variational distribution $q(\mathbf{S}|\boldsymbol{\phi})$. The result of this approximation is that one is working with a lower bound on the likelihood instead of the true likelihood. A lower bound which presumably is less complicated. Since the EM algorithm was proven to converge slowly at least in the case of small noise variance, the Easy Gradient Recipe was developed as a practical alternative.

Section 4.1 presents a practical choice of this variational distribution, namely choosing $q(\mathbf{S}|\boldsymbol{\phi})$ to be completely factorized. Section 4.1 also provides the optimal choices for the parameters $\boldsymbol{\phi} = \{\boldsymbol{\lambda}, \gamma, \boldsymbol{\nu}\}$ of $q(\mathbf{S}|\boldsymbol{\phi})$; optimal in the sense that they make the Kullback-Leibler divergence to the true source posterior as small as possible. Section 4.1 furthermore presents the LR and TAP extensions to the factorial choice and how to deal with these more advanced methods.

All these parts are now ready to be combined into one flexible and powerful algorithm: The Mean Field ICA. Since the parameters $\boldsymbol{\phi}$ are formally independent of the mixing matrix, noise covariance and source prior parameter, differentiating the bound $B(\boldsymbol{\theta}, \boldsymbol{\phi})$ with respect to $\boldsymbol{\theta}$ yields

$$\frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\theta}} = \left\langle \frac{\partial \ln p(\mathbf{X}, \mathbf{S}|\boldsymbol{\theta}, \boldsymbol{\phi}))}{\partial \boldsymbol{\theta}} \right\rangle_q .$$

That is, one obtains the same equations as in Chapter 2, where the true source posterior is substituted with the variational distribution. Equation 2.13 and 2.14 can be used as presented in Chapter 2 changing only the average from the true source posterior to the variational distribution. One now need to optimize the bound with respect to both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ using either the EM algorithm or the Easy Gradient Recipe with some optimizer:

1) The EM optimization is an iteration of two steps as presented in Section 3.1.3: First estimate the moments of the variational distributions and second use these estimates to update the estimate for the mixing matrix, noise covariance and source prior parameters. See Table 4.2 for an overview.

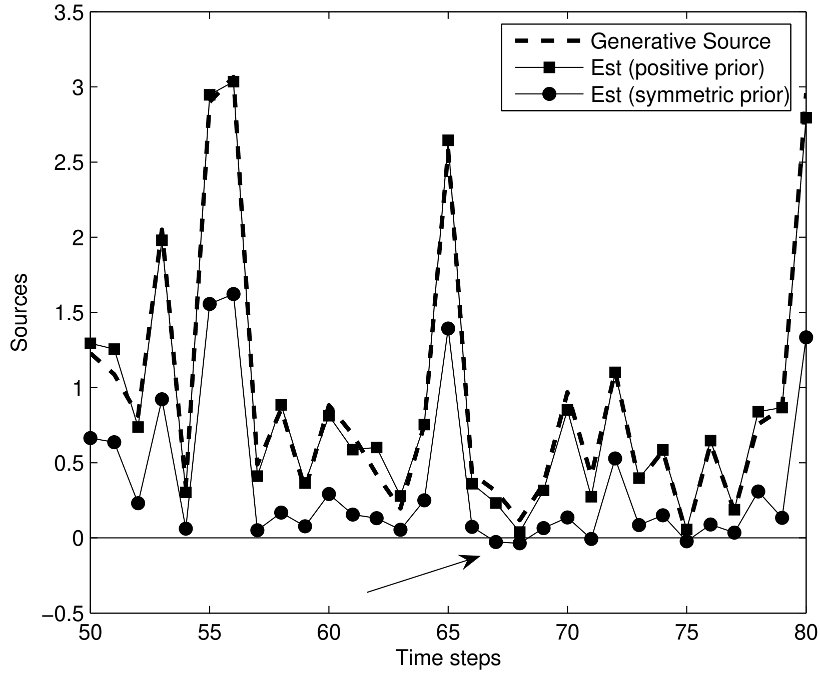


Figure 4.5: The beneficial effect of a positive prior. On this plot, a 2×2 data set constructed with positive sources has been attempted separated assuming both positive source (Exponential) and sparse symmetric sources (Zero-mean MoG). Two expected effects are visible: 1) The estimation using the correct prior is much better, and 2) Using a symmetric prior, one may get results which are in contradiction to the true model – one may get negative estimated source values.

2) The Easy Gradient Recipe computes the value and gradient of the bound (which equals the value and gradient of the true log likelihood) and apply the optimizer UCMINF to make an update of the $\boldsymbol{\theta}$ parameters – see Table 4.3 for an overview.

4.2.2 Numerical Examples

One of the major benefits of the Mean Field ICA, is the flexibility with respect to constraints and the source prior distribution. Many approaches to ICA cannot change the prior and are constructed with some fixed, e.g. a symmetric, prior.

An example of the benefits of choosing a prior suitable for a certain problem, a 2x2 setting with positive sources, i.e. $S_{it} > 0$ is considered. The source data is artificially constructed and for the separation, two different prior assumptions

E-step:	Calculate $\langle \mathbf{S} \rangle_q$ and $\langle \mathbf{S}\mathbf{S}^T \rangle_q$.
	$\boldsymbol{\lambda} = \mathrm{diag}(\mathbf{J})$
	$\boldsymbol{\gamma} = \mathbf{h} - \tilde{\mathbf{J}} \langle \mathbf{S} \rangle_q$
	$\boldsymbol{\nu}' = \boldsymbol{\nu}$
	$\langle \mathbf{S} \rangle_q \cong f_1(\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}')$ (Mean Field)
	$\langle \mathbf{S}\mathbf{S}^T \rangle_q \cong f_2(\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}')$ (Mean Field)
	Reiterate until convergence before going to M-step.
M-step:	Update estimate of $\mathbf{A}, \mathbf{W}$ and $\boldsymbol{\nu}$:
	$\mathbf{A} = \mathbf{X} \langle \mathbf{S} \rangle_q^T \langle \mathbf{S}\mathbf{S}^T \rangle_q^{-1}$
	$\mathbf{W} = \frac{1}{N} \mathrm{Tr}[\langle (\mathbf{X} - \mathbf{A}\mathbf{S})(\mathbf{X} - \mathbf{A}\mathbf{S})^T \rangle_q]$
	$\boldsymbol{\nu} = \ldots$ (Prior dependent)
	$\mathbf{J} = \mathbf{A}^T \mathbf{W}^{-1} \mathbf{A}$
	$\mathbf{h} = \mathbf{A}^T \mathbf{W}^{-1} \mathbf{X}$
	Goto E-step.

Table 4.2: Mean Field ICA (EM)

	Notation:
	$\boldsymbol{\theta}$ is short for $\mathbf{A}, \mathbf{W}$ and $\boldsymbol{\nu}$.
	$\boldsymbol{\phi}$ is short for $\boldsymbol{\lambda}, \boldsymbol{\gamma}, \langle \mathbf{S} \rangle_q \langle \mathbf{S}\mathbf{S}^T \rangle_q$
Step 1:	For a given $\boldsymbol{\theta}$, calculate $\boldsymbol{\phi}^* = \boldsymbol{\phi}(\boldsymbol{\theta})$ such that $\frac{\partial B}{\partial \boldsymbol{\phi}}\|_* = \mathbf{0}$.
Step 2:	With $\boldsymbol{\theta}, \boldsymbol{\phi}^*$ calculate $B(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$ and $\frac{dB}{d\boldsymbol{\theta}}(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$
Step 3:	Use the set $[\mathcal{L}, \frac{d\mathcal{L}}{d\boldsymbol{\theta}}] = [B, \frac{dB}{d\boldsymbol{\theta}}]$ in some advanced optimization method to find the next $\boldsymbol{\theta}^{n+1}$.
	Re-iterate until convergence is reached.

Table 4.3: Mean Field ICA (Easy Gradient)

are used: 1) A symmetric source prior, chosen as a mixture of two zero-mean gaussians, and 2) a positive prior, chosen as an exponential distribution. The error on the estimated $\hat{\mathbf{S}}$ is the element-average matrix norm from the generative $\mathbf{S}$, defined as $Err = ||\hat{\mathbf{S}} - \mathbf{S}||_2/(Nd)$. The result is

	Error
Symmetric prior	0.0173
Positive prior	0.0023

Clearly the positive prior is outperforming the symmetric prior - in this case with an order of magnitude. This is also evident in the general quality of the source estimation, shown in Figure 4.5; the estimated sources using the positive prior are very close to the generative sources compared to counterpart with symmetric prior. Furthermore, the estimated source values using the symmetric prior are not always positive and thus in contradiction with the model assumption.

As for the source estimation, the benefits of constraining the structure of the mixing matrix or the noise covariance are multiple. First of all, any kind of constraints on the solution space help the optimization and often also the convergence speed. Second, it may be important that the obtained estimates fulfil some requirements. If the data is interpreted as a sparse representation of images, for example, then at least the mixing matrix must be positive to reflect gray-level intensities. Images often results in rather large mixing matrices, but as a demonstration of the principle a 3x3 mixing matrix is considered

$$\mathbf{A}_{gen} = \begin{bmatrix} 0.0008 & 1.0074 & 0.0130 \\ 0.7100 & 0.0024 & 0.0055 \\ 0.0069 & 0.0794 & 1.0241 \end{bmatrix}.$$

Using this matrix, artificial data were generated and from this two different separation processes attempted: 1) A standard unconstrained and 2) a separation in which $\mathbf{A}$ is constrained to be positive. The estimated mixing matrices are

$$\begin{aligned} \mathbf{A}_{unc} &= \begin{bmatrix} -0.0219 & 1.0030 & 0.0392 \\ 0.6597 & 0.0272 & -0.0094 \\ 0.0269 & 0.0291 & 1.0078 \end{bmatrix} \\ \mathbf{A}_{pos} &= \begin{bmatrix} 0.0000 & 1.0031 & 0.0397 \\ 0.6595 & 0.0118 & 0.0000 \\ 0.0203 & 0.0277 & 1.0077 \end{bmatrix} \end{aligned}$$

where $\mathbf{A}_{unc}$ is the result of the unconstrained optimization, while $\mathbf{A}_{pos}$ is the result of the estimation in which $\mathbf{A}$ is restricted to being positive. In this example it made a significant difference since some of the entries in $\mathbf{A}_{unc}$ are negative, in contradiction with the model assumptions. For finite and especially smaller data sets, this is what one should expect, since even unbiased unconstrained estimators may have negative entries at the optimal solution.

		Fixed $\mathbf{W}$	Simple $\mathbf{W}$	Diagonal $\mathbf{W}$	Pos.Def. $\mathbf{W}$
Err($\mathbf{A}$)	mean	0.0690	0.0361	0.0337	0.0433
	median	0.0383	0.0321	0.0312	0.0321
Err($\mathbf{W}$)	mean	0	0.0008	0.0014	0.0047
	median	0	0.0007	0.0012	0.0021
Err($\mathbf{S}$)	mean	0.0118	0.0074	0.0070	0.0241
	median	0.0072	0.0065	0.0064	0.0068

Table 4.4: The impact of constraining the estimated noise covariance. The data was generated using the noise covariance $\mathbf{W} = \sigma^2\mathbf{I}$ and the result clearly reflects that through the lower error on the constrained noise covariance.

Regarding the covariance matrix $\mathbf{W}$, three different possible constraints have been demonstrated, as well as the case in which the covariance is not estimated but assumed known a priori. In a 2x2 example with sparse source prior, the parameters and sources of the model for 100 different source and noise realizations but same mixing matrix has been estimated. In each repeated estimation, the result is compared to the the generative values and the entry-average norm-error calculated. Table 4.4 shows the mean and median of the errors. The generative covariance is $\mathbf{W}_{gen} = \sigma^2\mathbf{I}$, not surprisingly, the error on $\mathbf{W}_{est}$ is increasing as the constraint is relaxed to be more flexible. More surprisingly, though, is that the effect on the other variables of interest, $\mathbf{A}$ and $\mathbf{S}$, is almost non-existent.

Another example of the use of positive constrains, a decomposition of the digit '3' from the MNIST database of handwritten digits has been considered, see Figure 4.6. The set of images of the digit '3' has been decomposed into 25 components using the symmetric MoG priors (left) and exponential priors (right), both with a mixing matrix which are constrained to be positive. The images are the columns of the estimated mixing matrix, and clearly the MoG result on the left is not decomposing the image into strokes as one would wish for, while the approach using positive priors to the right does a much better job.

4.2.3 Slow Convergence Issues

As an example of the slow convergence consider Figure 4.7: The EM algorithm makes the lower bound B increase in each step, but the increase in the first 1500 iterations is very small and has very modest effect on the estimated mixing matrix (inserts in the Figure). After about 1800 iterations, the convergence is suddenly reaching a reasonable speed, but the overall conclusion is unchanged: The EM algorithm spends an excessive amount of iterations with very little change in the parameters. Since the gradients are not available in the EM algorithm, the stopping criteria must be based on the change in either the parameters or the cost function (the lower bound B on the log likelihood).

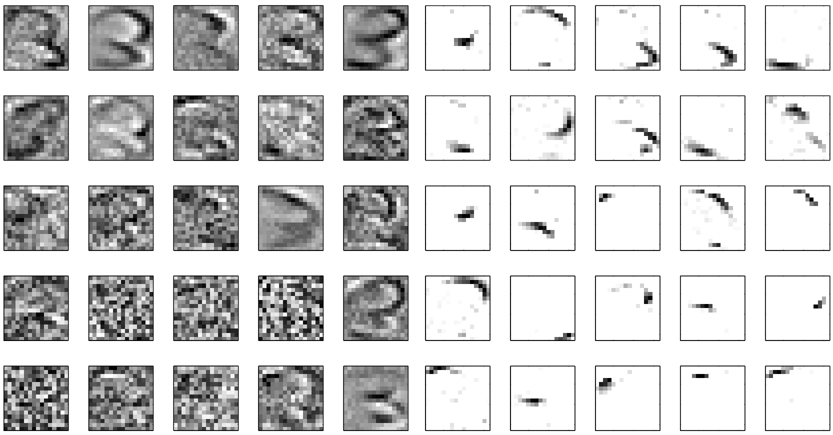


Figure 4.6: Positivity constrains on image decomposition. A data set of handwritten digits (the cipher 3) has been decomposed using a mixing matrix which is 256×25, i.e. the image data has 256 entries and these are decomposed into a sparse set of 25 basis vectors (the columns of $\mathbf{A}$). On the left, no positivity constrains are applied, and the result is by no means a decomposition of the strokes in the digits. On the right, both the mixing matrix and the sources are assumed positive, yielding a much better result.

Iteration bins	*0-50*	*50-100*	*100-∞*
EM	1%	6%	**93%**
AdapEM	**51%**	32%	17%
Ucminf	**100%**	0%	0%

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

Iteration bins	*0-50*	*50-100*	*100-∞*
EM	0%	2%	**98%**
AdapEM	14 %	**47%**	39%
Ucminf	**93%**	7%	0%

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

Table 4.5: Performance of the different approaches for the example and general 2x2 and 3x3 mixing matrices.

Situations like the first 1500 iterations in Figure 4.7 therefore poses a serious threat because it 'lurks' the algorithm to stop prematurely and there is no way of knowing better.

The slow convergence of the EM algorithm is also evident in Fig. 4.8: 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. The three other optimization methods applied, are the adaptive overrelaxed EM, and two gradient based approach: Simple gradient ascend and the quasi-Newton UCMINF.

4.2.3.1 Linearization and Convergence Rate

The slow convergence can be analyzed in different ways. One of the typical approaches is to Taylor expand the EM update to first order, close to the optimal point - an approach followed in this subsection.

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 it can be written as a mapping $\mathbf{a}^{n+1} = \mathbf{M}(\mathbf{a}^n)$. Close to the optimal $\mathbf{a}^*$, a first order Taylor-expansion of this mapping can be made to obtain

$$\mathbf{a}^{n+1} \cong \mathbf{a}^* + \mathbf{M}'_*(\mathbf{a}^n - \mathbf{a}^*) \tag{4.13}$$

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 $\mathbf{a}^*$. The matrix $\mathbf{M}'_*$ can be found from the matrix formulation by using $\mathbf{A}^{n+1} = \mathbf{X}\mathbf{m}^T\mathbf{G}^{-1}$ where $\mathbf{m} = \langle\mathbf{S}\rangle_q$ and $\mathbf{G} = \langle\mathbf{S}\mathbf{S}^T\rangle_q$ are functions of $\mathbf{A}^n$. Therefore

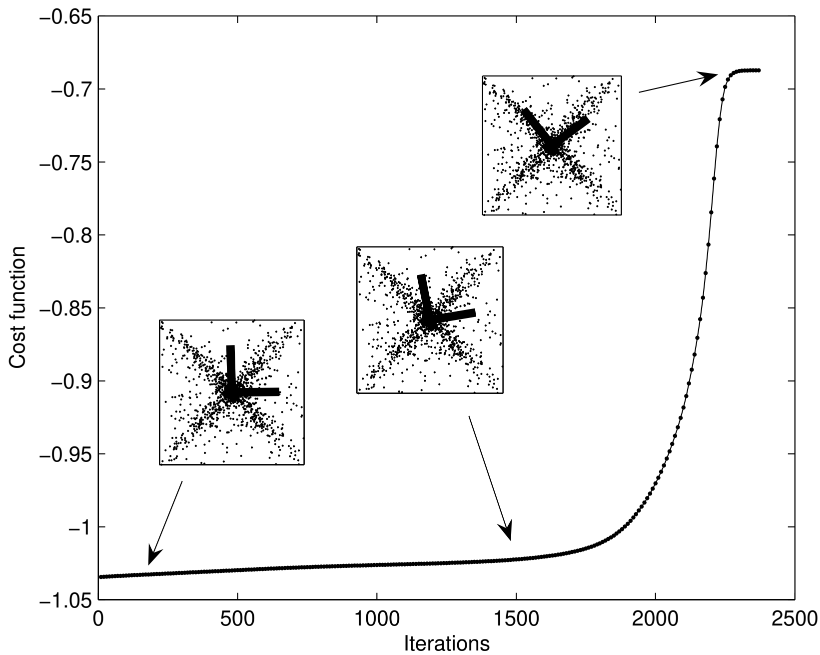
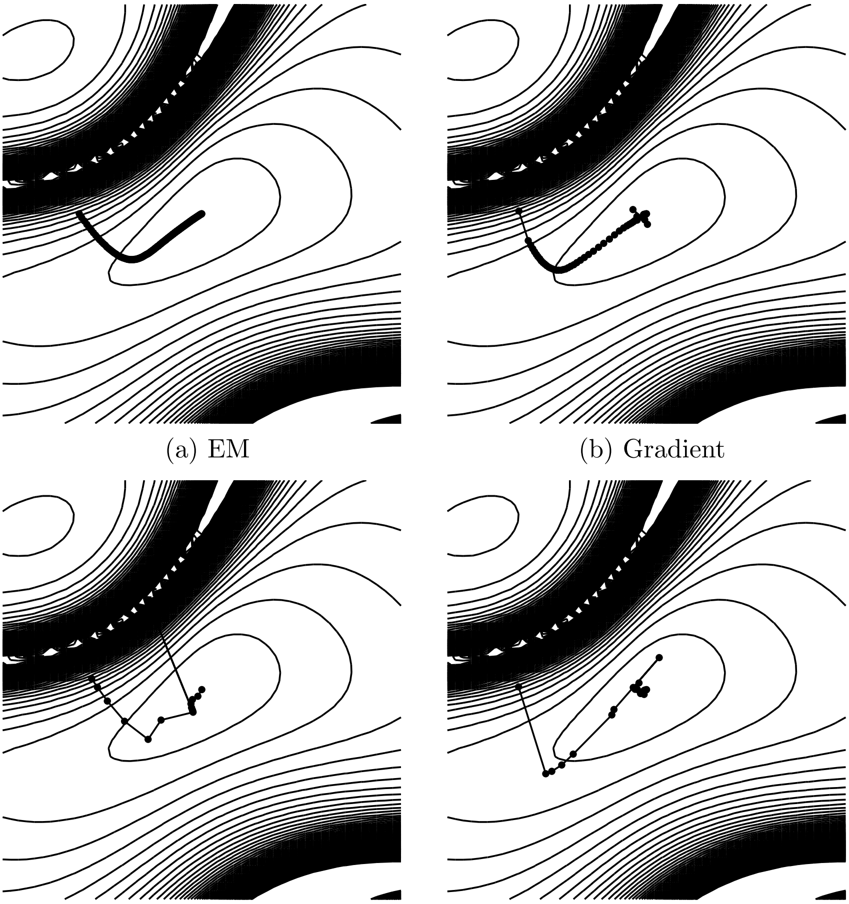


Figure 4.7: Convergence of the EM algorithm. This plot shows the convergence of the EM algorithm in a 2x2 setting: The cost function (the lower bound B) is in the first 1500 iterations only increasing very slowly and the inserts shows that the estimate of the mixing matrix in this part of the process is not improving much. In the end, the convergence speed is improving strongly, but the problem is that the flat areas is posing a real threat to the use of the EM algorithm: Not only are the 'flat areas' of the cost function making the entire process very slow, they also make us believe that convergence is reached because there is no way of knowing better.



(a) EM (b) Gradient

(c) Adap EM (d) UCMINF

Figure 4.8: Paths of optimization in parameter space. The plots show how different optimization schemes performs on a problem of estimating a 2×2 matrix with symmetric sparse priors. The likelihood (contours) and the path (dots) are projected onto a 2 dimensional plane for improved visualization. Plot (a) shows the performance of the basic EM algorithm; Plot (b) a basic gradient optimization using the Easy Gradient Recipe for computation of the gradient; Plot (c) the Adaptive Overrelaxed EM; and Plot (d) the UCMINF (Hans Bruun) optimizer on gradients provided by the Easy Gradient Recipe. Note that the step size of the ordinary EM algorithm is so small, that the dots form a solid line, while Adaptive Overrelaxed EM and UCMINF are taking visible distinct steps. Note on Plot (c) the rejected test step from close to the optimum and to the 'ridge' of the cost function.

$\mathbf{A}^{n+1}$ can be differentiated with respect to $\mathbf{A}^n$

$$\frac{\partial \mathbf{A}^{n+1}}{\partial A^n_{ij}} = \mathbf{X}\Big(\frac{d\mathbf{m}^T}{dA^n_{ij}}\mathbf{G}^{-1} + \mathbf{m}^T\frac{d\mathbf{G}^{-1}}{dA^n_{ij}}\Big)\,, \tag{4.14}$$

in which all entities can be computed with some effort. From this expression $\mathbf{M}'_*$ can be determined 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. 4.13, 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 $\mathbf{M}'_*$ the convergence properties of EM can be understood 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 [27, 26], 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, it holds 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, one often investigate the convergence rate at time n, denoted r_n, instead of r. Using the methods presented above, one can find the matrix $\mathbf{M}'_*$ and the convergence rate r_n for the problematic example presented in Fig. 4.8.

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 4.9 (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. 4.13 a good approximation. This comes as no surprise close to the optimal solution, but as Fig. 4.9 (right) demonstrates, Eq. 4.13 is a good approximation far from the optimal solution. Fig. 4.9 (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 analysis confirms the findings numerically, but since the expressions in Equation 4.14 are typically rather complicated – in fact too complicated for practical purposes – this approach to an analysis of the EM algorithm is in my view not that useful for many problems.

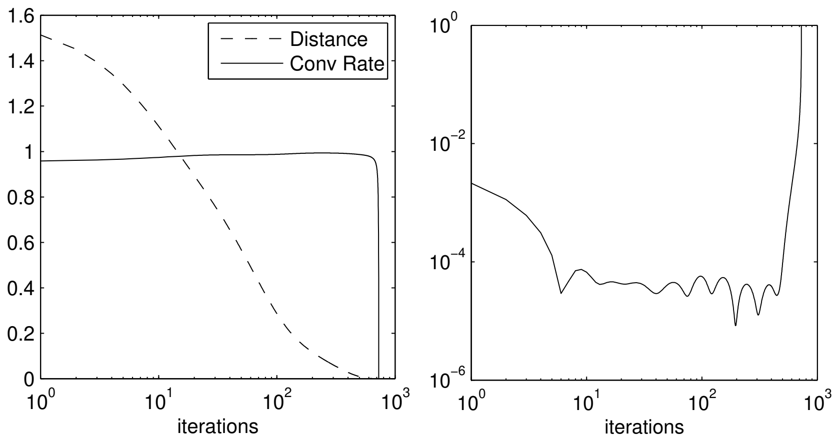


Figure 4.9: Example of linearization of the EM update in 2×2 setup. On the left is a plot of two important entities in the space of the matrix $\mathbf{A}$: 1) the euclidian distance (dashed) from the estimated to the optimal and 2) the convergence ratio r_n. As expected, the distance to the optimal point goes to zero, but the convergence ratio $r_n \to 1$ until very close to the optimal point. This indicates that from a very early state the convergence rate is poor and that it decreases until a certain point very close the optimal solution. On the right is the error of assuming the EM update to linear with some matrix $\mathbf{M}$. This plot confirms the indication that very early in the process, the EM update of the mixing matrix is essentially a linear update very close to the identity.

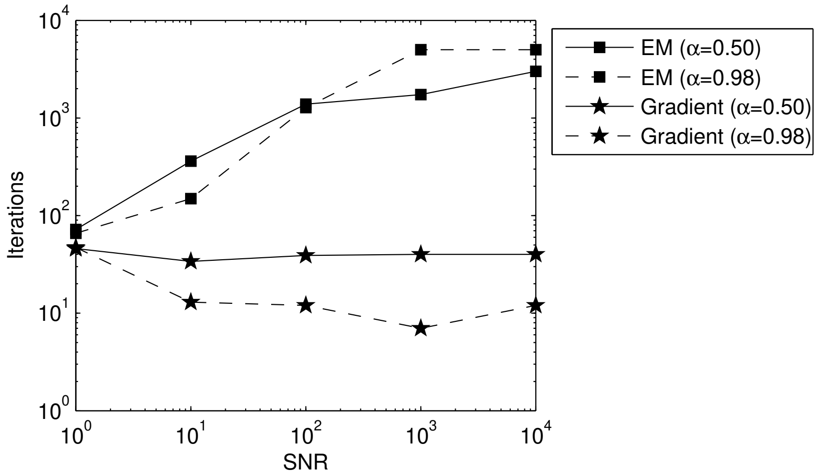


Figure 4.10: Convergence and signal to noise ratio (SNR). This plot demonstrates that the Easy Gradient Recipe with a quasi-Newton method ('Gradient') is far more efficient than the EM algorithm ('EM'). Especially at high SNR the difference is clear: While the gradient based optimization is more or less constant, the EM algorithm uses more and more iterations as SNR becomes large, i.e. in the low noise limit. This is in agreement with the analytical results. Note that strong dependencies slightly increases the number of iterations needed in the EM algorithm. The number of iterations is of course not exactly the number of floating point operations and thus one should in principle compare FLOPS rather than iterations. In this setting, however, more than 95% of the time is spend in operations common to the EM algorithm and the gradient based approach and thus making the iterations a good basis for comparing the two. SNR is defined as $SNR = \mathrm{Tr}(\mathbf{AC}_s\mathbf{A}^T)/(m\sigma^2)$, where $\mathbf{C}_s$ is the signal covariance.

4.2.3.2 Scaling with the Noise Level

In Figure 4.10 typical examples are plotted demonstrating the influence of the noise level: A mixing matrix with weak ($\alpha = 0.50$) dependencies and another with strong dependencies ($\alpha = 0.98$) are estimated. Clearly the number of iterations needed for the EM algorithm are extraordinary, and for low noise levels, prohibitively large. The Easy Gradient Recipe with the quasi-Newton optimizer UCMINF, on the other hand, is performing very well. The number of function evaluations is below forty for all noise levels and both weak and strong dependencies. These plots demonstrates that the Easy Gradient Recipe is performing much better than EM, simply recycling the analytical and computational results in a different way.

4.3 Convolutive Mean Field ICA

This section describes the extension of the Mean Field ICA to the convolutive mixing model, that is, each observation is a linear mixture of vectors F time steps backward, i.e.

$$\mathbf{x}_t = \sum_{\tau=0}^{F-1} \mathbf{A}^\tau \mathbf{s}_{t-\tau} + \mathbf{n}_t, \qquad t = 1, ..., N .$$

Each of the mixing matrices $\mathbf{A}^\tau$ are different – possibly even statistically independent – and the noise is zero-mean Gaussian as usual. Convolutive mixtures are a natural extension of the instantaneous mixing model and important for a number of applications.

4.3.1 Factorized Variational Distribution

In the instantaneous version of Mean Field ICA, the interactions in d dimensions are substituted by a model of a so-called 'external field' – an extra linear factor on each of the variables S_{it}. Extending this concept to the convolutive model involves modelling dependencies over Fd variables, where F is the filter length, by an external field. There are at least two aspects which makes the factorized approach less exact: 1) having many variables which interact and 2) having strong interactions. If the interactions are many but weak, then the factorized approach to convolutive mixtures may indeed work well.

4.3.1.1 Translating into Statistical Physics

As with the instantaneous Mean Field ICA, the link between the probabilistic framework and the framework of statistical physics is the fundamental equation $p = e^{-H}/Z$. Using Bayes' theorem, one can therefore find the Hamiltonian corresponding to the posterior of the hidden variables by the following identity

$$\frac{P(\mathbf{X}|\mathbf{A},\mathbf{W},\mathbf{S})P(\mathbf{S}|\boldsymbol{\nu})}{P(\mathbf{X}|\mathbf{A},\mathbf{W},\boldsymbol{\nu})} = P(\mathbf{S}|\mathbf{X},\mathbf{A},\mathbf{W},\boldsymbol{\nu}) = \frac{e^{-H(\mathbf{S})}}{Z} .$$

That is

$$H(\mathbf{S}) = -\ln P(\mathbf{X}|\mathbf{A},\mathbf{W},\mathbf{S}) - \ln P(\mathbf{S}|\boldsymbol{\nu}) .$$

The second term, the logarithm of the hidden variables prior, can be rather complicated in the sense that the prior of the hidden variables can be complicated, but it is, by assumption always simple in the sense that is factorizing completely. Also, the hidden variables prior is completely indifferent to the choices of mixing model and therefore the corresponding results from the instantaneous model can be directly applied in the convolutive model. It is therefore the first term which demands the attention, rewriting it as clearly as possible as a function of the hidden variables. Using the notation $\mathbf{f}_t = \sum_\tau \mathbf{A}^\tau \mathbf{s}_{t-\tau}$ the initial rewriting is

$$\begin{aligned}
& -\ln P(\mathbf{X}|\mathbf{A},\mathbf{W},\mathbf{S}) \\
= \; & -\ln C + \frac{1}{2}\mathrm{Tr}\Big[(\mathbf{X}-\mathbf{f})^T\mathbf{W}^{-1}(\mathbf{X}-\mathbf{f})\Big] \\
= \; & -\ln C + \frac{1}{2}\mathrm{Tr}(\mathbf{X}^T\mathbf{W}^{-1}\mathbf{X}) + \frac{1}{2}\mathrm{Tr}(\mathbf{f}^T\mathbf{W}^{-1}\mathbf{f}) - \mathrm{Tr}(\mathbf{X}^T\mathbf{W}^{-1}\mathbf{f}) \\
= \; & \frac{1}{2}\mathrm{Tr}(\mathbf{f}^T\mathbf{W}^{-1}\mathbf{f}) - \mathrm{Tr}(\mathbf{X}^T\mathbf{W}^{-1}\mathbf{f}) + H_c(\mathbf{X},\mathbf{W})
\end{aligned}$$

which is of course general with respect to the choice of transformation function $\mathbf{f}$. By defining similar but expanded versions of the interaction matrix and the external field

$$\mathbf{J}^{\tau\tau'} = (\mathbf{A}^\tau)^T\mathbf{W}^{-1}\mathbf{A}^{\tau'} \tag{4.15}$$

$$\mathbf{h}^{\tau} = (\mathbf{A}^\tau)^T\mathbf{W}^{-1}\mathbf{X} \tag{4.16}$$

the physics-notation of the instantaneous mixing can be generalized in the following way. Focusing first on the interaction term,

$$\begin{aligned}
\mathrm{Tr}(\mathbf{f}^T\mathbf{W}^{-1}\mathbf{f}) &= \sum_{t=1}^{N}\Big\{\Big(\sum_{\tau=0}^{F-1}\mathbf{A}^\tau\mathbf{s}_{t-\tau}\Big)^T\mathbf{W}^{-1}\Big(\sum_{\tau'=0}^{F-1}\mathbf{A}^{\tau'}\mathbf{s}_{t-\tau'}\Big)\Big\} \\
&= \sum_{t=1}^{N}\sum_{\tau,\tau'=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T(\mathbf{A}^\tau)^T\mathbf{W}^{-1}\mathbf{A}^{\tau'}\mathbf{s}_{t-\tau'}\Big) \\
&= \sum_{t=1}^{N}\sum_{\tau,\tau'=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\mathbf{J}^{\tau\tau'}\mathbf{s}_{t-\tau'}\Big)
\end{aligned}$$

and then on the part with the external field, one get

$$\mathrm{Tr}(\mathbf{X}^T\mathbf{W}^{-1}\mathbf{f}) = \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\mathbf{x}_t^T\mathbf{W}^{-1}\mathbf{A}^\tau\mathbf{s}_{t-\tau} = \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\mathbf{h}_t^\tau)^T\mathbf{s}_{t-\tau}$$

Collecting these into the expression above, one get a formulation of the Hamiltonian which matches the posterior of the hidden variables,

$$H = \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau,\tau'=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\mathbf{J}^{\tau\tau'}\mathbf{s}_{t-\tau'}\Big) - \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\mathbf{h}_t^\tau)^T\mathbf{s}_{t-\tau} + H_c - \ln p(\mathbf{S})$$

a Hamiltonian which makes it clear how the hidden variables, in the physics interpretation known as 'spins', are subject to interaction with all other spins within d steps in the dimension of the hidden variables (as usual) and F steps in time (the convolution). As with the instantaneous transformation, the interaction is induced by the statistical correlation imposed by the observations: the observation at time t is carrying information about the hidden variables in all dimensions from time t and back to time $t-(F-1)$.

4.3.1.2 Factorizing and the Variational Free Energy

In the framework of statistical physics, the ordinary way to make a factorized approximation as close as possible to the true Hamiltonian, is to define a factorized variational Hamiltonian H_q and then write up the variational free energy defined by $F_{var} = F_q + \langle H - H_q \rangle_q$. This variational free energy must then be minimized with respect to a suitable choice of parameters, which for the instantaneous model includes the first and second moments.

The variational Hamiltonian is defined by the following expression in which $\hat{\boldsymbol{\lambda}}^\tau$ is assumed diagonal for all τ

$$H_q = \frac{1}{2} \sum_{t=1}^{N} \sum_{\tau=0}^{F-1} \mathbf{s}_{t-\tau}^T \hat{\boldsymbol{\lambda}}^\tau \mathbf{s}_{t-\tau} - \sum_{t=1}^{N} \sum_{\tau=0}^{F-1} (\hat{\gamma}_t^\tau)^T \mathbf{s}_{t-\tau} - \ln p(\mathbf{S})\,. \tag{4.17}$$

The sum over t and τ, and the index $t-\tau$ of the sources, obscures the fact that this Hamiltonian is factorizing and essentially identical to the Hamiltonian of the instantaneous case. Defining

$$\boldsymbol{\lambda}_t \equiv \sum_{l=1}^{\xi} \hat{\boldsymbol{\lambda}}^l \qquad \boldsymbol{\gamma}_t \equiv \sum_{\tau=1}^{\xi} \hat{\gamma}_{t+\tau}^\tau \qquad \xi \equiv \min\{F-1, t\} \tag{4.18}$$

and changing the sum appropriately the factorial Hamiltonian can be written simply as

$$H_q = \tfrac{1}{2} \sum_{t=1}^{N} \mathbf{s}_t^T \boldsymbol{\lambda}_t \mathbf{s}_t - \sum_{t=1}^{N} \boldsymbol{\gamma}_t^T \mathbf{s}_t - \ln p(\mathbf{S})\,. \tag{4.19}$$

Using a rewriting approach for the convolutive case, one needs to rewrite the true Hamiltonian appropriately: Using the notation $\bar{\mathbf{J}}$ for the matrix in which non-diagonal entries have been set to zero $\mathbf{J} = \bar{\mathbf{J}} + \tilde{\mathbf{J}}$, the interaction term becomes

$$
\begin{aligned}
& \tfrac{1}{2}\mathrm{Tr}(\mathbf{f}^T\mathbf{W}^{-1}\mathbf{f}) \\
= \; & \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\mathbf{J}^{\tau\tau}\mathbf{s}_{t-\tau}\Big) + \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau\neq\tau'}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\mathbf{J}^{\tau\tau'}\mathbf{s}_{t-\tau'}\Big) \\
= \; & \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\bar{\mathbf{J}}^{\tau\tau}\mathbf{s}_{t-\tau}\Big) + \\
& \quad \tfrac{1}{2}\sum_{t=1}^{N}\Big\{\sum_{\tau=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\tilde{\mathbf{J}}^{\tau\tau}\mathbf{s}_{t-\tau}\Big) + \sum_{\tau\neq\tau'}^{F-1}\Big(\mathbf{s}_{t-\tau'}^T\mathbf{J}^{\tau'\tau}\mathbf{s}_{t-\tau}\Big)\Big\} \\
= \; & \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\bar{\mathbf{J}}^{\tau\tau}\mathbf{s}_{t-\tau}\Big) + \\
& \quad \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\Big\{\Big(\mathbf{s}_{t-\tau}^T\tilde{\mathbf{J}}^{\tau\tau}\Big) + \sum_{\tau'=1}^{F-1}(1-\delta_{\tau\tau'})\Big(\mathbf{s}_{t-\tau'}^T\mathbf{J}^{\tau'\tau}\Big)\Big\}\mathbf{s}_{t-\tau} \\
\equiv \; & \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\Big(\mathbf{s}_{t-\tau}^T\bar{\mathbf{J}}^{\tau\tau}\mathbf{s}_{t-\tau}\Big) + \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\check{\mathbf{h}}_t^{\tau})^T\mathbf{s}_{t-\tau}\,,
\end{aligned}
$$

where $\check{\mathbf{h}}_t^{\tau}$ is defined by

$$
\check{\mathbf{h}}_t^{\tau} \equiv \tfrac{1}{2}\Big\{\Big(\mathbf{s}_{t-\tau}^T\tilde{\mathbf{J}}^{\tau\tau}\Big) + \sum_{\tau'=1}^{F-1}(1-\delta_{\tau\tau'})\Big(\mathbf{s}_{t-\tau'}^T\mathbf{J}^{\tau'\tau}\Big)\Big\}\,.
$$

The stage is now set for rewriting the variational free energy. The variational free energy for the convolutive mixtures are defined in the same way as for any transformation function, since the variational free energy is derived using only Jensen's inequality and the properties of the exponential function. That is,

$$
\begin{aligned}
F_{var} &= F_q + \langle H - H_q\rangle_q \\
&= F_q + \Big\langle\big(\tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau,\tau'}^{F-1}\mathbf{s}_{t-\tau}^T\mathbf{J}^{\tau\tau'}\mathbf{s}_{t-\tau'} - \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\mathbf{h}_t^{\tau})^T\mathbf{s}_{t-\tau} + H_c - \ln P(\mathbf{S})\big) \\
&\qquad -\big(\tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\mathbf{s}_{t-\tau}^T\hat{\boldsymbol{\lambda}}^{\tau}\mathbf{s}_{t-\tau} - \sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\tilde{\gamma}_t^{\tau})^T\mathbf{s}_{t-\tau} - \ln P(\mathbf{S})\big)\Big\rangle_q \\
&= F_q + H_c + \tfrac{1}{2}\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}\langle\mathbf{s}_{t-\tau}^T(\bar{\mathbf{J}}^{\tau\tau} - \hat{\boldsymbol{\lambda}}^{\tau})\mathbf{s}_{t-\tau}\rangle_q \\
&\qquad -\sum_{t=1}^{N}\sum_{\tau=0}^{F-1}(\mathbf{h}_t^{\tau} - \hat{\mathbf{h}}_t^{\tau} - \hat{\gamma}_t^{\tau})^T\langle\mathbf{s}_{t-\tau}\rangle_q
\end{aligned}
$$

where

$$\hat{\mathbf{h}}_t^\tau \equiv \langle \check{\mathbf{h}}_t^\tau \rangle_q = \tfrac{1}{2}\Big\{ \Big(\langle \mathbf{s}_{t-\tau} \rangle_q^T \tilde{\mathbf{J}}^{\tau\tau} \Big) + \sum_{\tau'=1}^{F-1} (1-\delta_{\tau\tau'}) \Big(\langle \mathbf{s}_{t-\tau'} \rangle_q^T \mathbf{J}^{\tau'\tau} \Big) \Big\} \,.$$

Note that the last line in which $\langle \check{\mathbf{h}}_t^\tau \mathbf{s}_{t-\tau} \rangle_q = \langle \check{\mathbf{h}}_t^\tau \rangle_q \langle \mathbf{s}_{t-\tau} \rangle_q$ is using that q is factorizing and that $\check{\mathbf{h}}_t^\tau$ is independent of $\mathbf{s}_{t-\tau}$. As in the instantaneous case, the variational free energy is now differentiated with respect to both the parameters and the moments resulting in the following mean field equations:

$$\begin{aligned}
\hat{\boldsymbol{\lambda}}^\tau &= \mathrm{diag}(\mathbf{J}^{\tau\tau}) && (4.20)\\
\hat{\gamma}_t^\tau &= \mathbf{h}_t^\tau - \hat{\mathbf{h}}_t^\tau && (4.21)\\
\langle S_{it} \rangle_q &= \int \frac{e^{-H_q(S_{it})}}{Z} S_{it} dS_{it} && (4.22)\\
\langle S_{it}^2 \rangle_q &= \int \frac{e^{-H_q(S_{it})}}{Z} S_{it}^2 dS_{it} \,. && (4.23)
\end{aligned}$$

The solution to these mean field equations provides the first and second source posterior moments $\langle \mathbf{S} \rangle_q$ and $\langle \mathbf{S}\mathbf{S}^T \rangle_q$ which can be used in the combined algorithm, described in next subsection.

4.3.2 The Algorithm

With Equations 4.20 - 4.23 a method for choosing the parameters of the factorized distribution has been shown, such that they fit the variational distribution to the source posterior of a convolutive model. Combining these with Equation 2.21 and 2.22, which updates $\underline{\mathbf{A}}$ and $\mathbf{W}$ – and as with the instantaneous model substituting the average over the source posterior with the average over the variational distribution – one has the basic building blocks of an optimization of the convolutive model. As always, one can optimize it using either EM or the Easy Gradient Recipe with some advanced optimizer – see Table 4.6 and 4.7 for an overview.

As an example of the algorithm, consider Figure 4.11 and 4.12. In this case, a mixture of two laplacian sources have been mixed with a filter length about 40 and total signal length $N = 1000$. Figure 4.11 shows the four combinations of generative and estimated sources plotted against each other, and the agreement is rather good. Figure 4.12 shows the generative and estimated mixing matrices in the following form: To the upper left is the values for A_{11}, i.e. then generative (upper) and estimated (lower) values of $A_{11}^0, A_{11}^1, ..., A_{11}^{39}$ plotted as time lag sequences. Clearly these are very alike, leading to the conclusion that the convolutive Mean Field ICA worked rather well on this example.

The convolutive problem, however, is far more complicated than the instantaneous. Experience show that not only is the convolutive case subject to the

E-step:	Calculate $\langle \mathbf{S} \rangle_q$ and $\langle \mathbf{S}\mathbf{S}^T \rangle_q$.
	$\boldsymbol{\lambda} = \sum_{l=1}^{\xi} \hat{\boldsymbol{\lambda}}^l, \qquad \hat{\boldsymbol{\lambda}}^\tau = \mathrm{diag}(\mathbf{J}^{\tau\tau})$
	$\gamma_t = \sum_{l=1}^{\xi} \hat{\gamma}_{t+l}^l, \qquad \hat{\gamma}_t^\tau = \mathbf{h}_t^\tau - \hat{\mathbf{h}}_t^\tau$
	$\boldsymbol{\nu}' = \boldsymbol{\nu}$
	$\langle \mathbf{S} \rangle_q \cong f_1(\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}')$ (Mean Field)
	$\langle \mathbf{S}\mathbf{S}^T \rangle_q \cong f_2(\boldsymbol{\lambda}, \boldsymbol{\gamma}, \boldsymbol{\nu}')$ (Mean Field)
	$(F_*)_{ij}^\tau = \sum_{t=1}^N X_{it} \langle S_{j,t-\tau} \rangle$
	$(G_*)_{ij}^{\tau,\tau'} = \sum_{t=1}^N \langle S_{i,t-\tau} S_{j,t-\tau'} \rangle$
	Reiterate until convergence before going to M-step.
M-step:	Update estimate of $\mathbf{A}, \mathbf{W}$ and $\boldsymbol{\nu}$:
	$\underline{\mathbf{A}} = \underline{\mathbf{F}\mathbf{G}}^{-1}$
	$\mathbf{W} = \frac{1}{N}\Big[\mathbf{X}\mathbf{X}^T - \Big(\underline{\mathbf{F}\mathbf{A}}^T + \underline{\mathbf{A}\mathbf{F}}^T\Big) + \underline{\mathbf{A}\mathbf{G}\mathbf{A}}^T\Big]$
	$\boldsymbol{\nu} = \ldots$ (Prior dependent)
	$\mathbf{J}^{\tau\tau'} = (\mathbf{A}^\tau)^T \mathbf{W}^{-1} \mathbf{A}^{\tau'}$
	$\mathbf{h}^\tau = (\mathbf{A}^\tau)^T \mathbf{W}^{-1} \mathbf{X}$
	$\hat{\mathbf{h}}_t^\tau = \frac{1}{2}\Big\{\Big(\langle \mathbf{s}_{t-\tau}\rangle_q^T \tilde{\mathbf{J}}^{\tau\tau}\Big) + \sum_{\tau'=0}^{F-1}(1-\delta_{\tau\tau'})\Big(\langle \mathbf{s}_{t-\tau'}\rangle_q^T \mathbf{J}^{\tau'\tau}\Big)\Big\}$
	Goto E-step.

Table 4.6: Convolutive Mean Field ICA (EM)

	Notation:
	$\boldsymbol{\theta}$ is short for $\mathbf{A}, \mathbf{W}$ and $\boldsymbol{\nu}$.
	$\boldsymbol{\phi}$ is short for $\boldsymbol{\lambda}, \boldsymbol{\gamma}, \langle \mathbf{S} \rangle_q \langle \mathbf{S}\mathbf{S}^T \rangle_q$
Step 1:	For a given $\boldsymbol{\theta}$, calculate $\boldsymbol{\phi}^* = \boldsymbol{\phi}(\boldsymbol{\theta})$ such that $\frac{\partial B}{\partial \boldsymbol{\phi}}\|_* = \mathbf{0}$. (This essentially an E-step)
Step 2:	With $\boldsymbol{\theta}, \boldsymbol{\phi}^*$ calculate $B(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$ and $\frac{dB}{d\boldsymbol{\theta}}(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$ (This is essentially an M-step)
Step 3:	Use the set $[\mathcal{L}, \frac{d\mathcal{L}}{d\boldsymbol{\theta}}] = [B, \frac{dB}{d\boldsymbol{\theta}}]$ in some advanced optimization method to find the next $\boldsymbol{\theta}^{n+1}$.
	Re-iterate until convergence is reached.

Table 4.7: Convolutive Mean Field ICA (Easy Gradient)

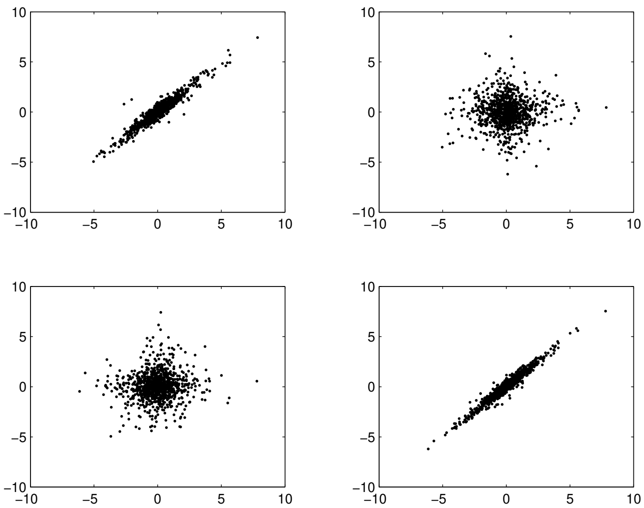


Figure 4.11: Convolutive Mean Field ICA - The Sources: This plot shows the generative and estimated sources plotted against each other in a 2x2 example with 40 time lags. In each plot, the generative source values are on the first axis, while the estimated source values are on the second. Clearly the estimation is reasonably successful.

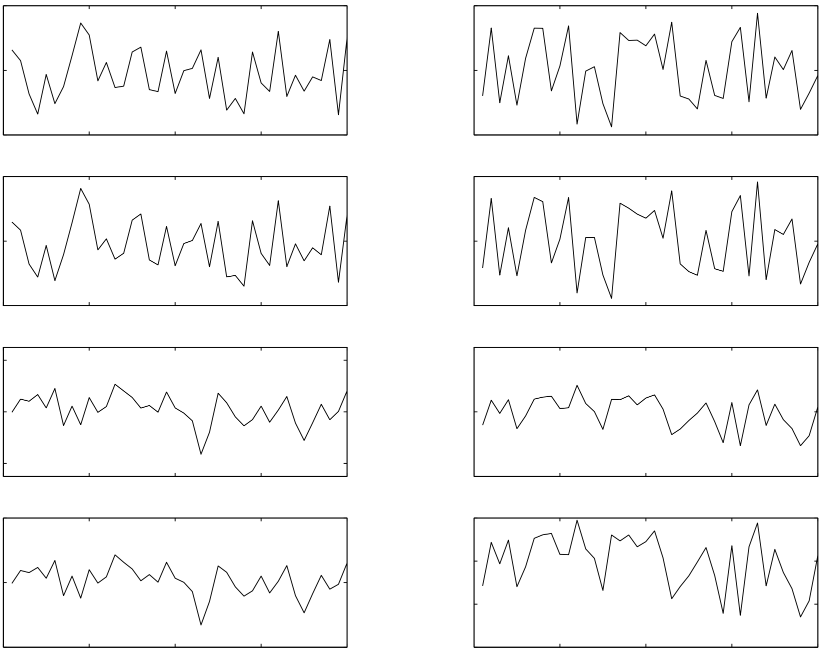


Figure 4.12: Convolutive Mean Field ICA - The filters: This plot shows the generative and estimated filters in an 2x2 example with 40 time lags. The two plots in the upper left corner is the generative (upper) and estimated (lower) values of the first entry, i.e. A_{11}^{τ} for $\tau = 0, 1, ..., 39$. The two upper right is the same for A_{12}^{τ}, the lower leftA_{21}^{τ} and lower right A_{22}^{τ}. Clearly the estimation is reasonably successful.

same slow convergence properties of the EM algorithm as the instantaneous, it also has multiple local maxima and thus poses a real and severe challenge to the optimization. To be efficient, one must therefore both use the Easy Gradient Recipe and techniques like simulated annealing - a setup which in practical terms may prove somewhat computationally heavy. In this thesis, the convolutive case has been developed and presented as a proof of concept and thus, much more can and should be done to make a practical and efficient approach to the real world data. That being said, the initial tests demonstrates that it is in fact possible to estimate the sources and the filters (mixing matrix entries) using the proposed setup, and with the flexibility with respect to constraints and prior on the parameters, the convolutive Mean Field ICA may indeed be useful in future applications.

Summary of Chapter 4

In this Chapter, the mean field theory has been introduced and the combined Mean Field ICA algorithm presented. A number of examples demonstrating aspects of accuracy and convergence have been presented to demonstrate that, as predicted, the Mean Field ICA algorithm converges intolerably slow – especially for low noise levels. Finally the Mean Field ICA algorithm has been expanded to a convolutive model and a simple example presented as a proof of concept.

Chapter 5

Energy Based Models

The Mean Field ICA is an example of a 'generative model' (GM), that is, a model which makes assumptions about how the data is generated and tries to estimate the parameters of this generative process. There is, however, also a fundamentally different way of doing ICA, namely the so-called filtering approach. In the filtering approach, one estimates the sources by use of the filters $\mathbf{W}$ (Note that $\mathbf{W}$ in this Chapter denotes a filtering matrix and not the noise covariance matrix), by setting $\hat{\mathbf{s}}_t = \mathbf{W}\mathbf{x}_t$ and then estimate $\mathbf{W}$ such that $\hat{\mathbf{s}}$ is fulfilling some reasonable criteria. In the square setting, where there for each time step is as many observations as there are sources, one can prove that most if not all approaches, generative and filtering, are essentially qualitatively equivalent. The optimization process, however, may be rather different for the two approaches, as described in [36]. But in the overcomplete (underdetermined) setting, for example, it is generally rather difficult to make precise statements about the relationship.

This Chapter presents an attempt to understand and compare generative and filtering approaches to the overcomplete case. It is joint work with Te-Won Lee and Jiucang Hao during my visit at University of California San Diego, and a paper [40] has recently been submitted to Neural Information Processing.

5.1 Introduction to EBM

The Energy Based Models are introduced in [52]. In this subsection, the derivation and main points of the EBM's are briefly sketched such that the analysis presented can be understood in the right context. For further details, the reader is referred to the well-written original paper.

5.1.1 Derivation

The Energy Based Models aim at demixing the measurements $\mathbf{X}$ by a filtering with $\mathbf{W}$ in the traditional way which for the entire data set can be written $\hat{\mathbf{S}} = \mathbf{W}\mathbf{X}$. The filtering coefficients $\mathbf{W}$ is determined through a construction of a model distribution $p_{\mathbf{W}}(\mathbf{x})$ which is approximated to the data distribution

$$p_0(\mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\delta(\mathbf{x}-\mathbf{x}_t) \tag{5.1}$$

through the Kullback-Leibler divergence. The model distribution is constructed in the following way using the source prior: An energy is defined in some sensible way – very often as $E(\mathbf{x};\mathbf{W}) = -\ln p_s(\mathbf{W}\mathbf{x})$. This ensures the property that choosing $\mathbf{W}$ such that the resulting estimated sources are not too unlikely, is rewarded through low energy levels and unlikely source values penalized with high energy levels. Other definitions of energy could be made, but this is especially appealing due to its calculational properties. The energy E is used in a Gibbs distribution, i.e.

$$p_{\mathbf{W}}(\mathbf{x}) = \frac{e^{-E(\mathbf{x};\mathbf{W})}}{Z(\mathbf{W})} = \frac{p_s(\mathbf{W}\mathbf{x})}{\int p_s(\mathbf{W}\mathbf{x})d\mathbf{x}} \tag{5.2}$$

which is chosen as the model distribution. The task now is to estimate the filtering matrix $\mathbf{W}$ from minimization of the Kullback-Leibler divergence, $KL(p^0||p_{\mathbf{W}})$.

5.1.2 Contrastive Divergence Learning

Using $p^{\infty}(x)$ to denote the model distribution $p_{\mathbf{W}}(x)$, for reasons to be explained shortly, the derivative Kullback-Leibler divergence with respect to the parameters W can be written

$$\frac{\partial KL}{\partial W_{ij}} = -\left\langle \frac{\partial E(x)}{\partial W_{ij}} \right\rangle_{p^{\infty}} + \left\langle \frac{\partial E(x)}{\partial W_{ij}} \right\rangle_{p^{0}}$$

The notation p^0 and p^{∞} are to be understood in the following way: The integral over the true model distribution p^{∞} is not possible to calculate analytically and therefore one has to resort to approximations. One approach is to construct a Markov Chain Monte Carlo (MCMC) approximation which starts in the data and then make steps in the data space $\mathbf{X}_{(0)}, \mathbf{X}_{(1)}, \mathbf{X}_{(2)}, ..., \mathbf{X}_{(n)}, ...$ and from these samples construct an estimated average denoted as averages over the distributions $p^0, p^1, p^2, ..., p^n, ...$ This integral will presumably converge, at least in the limit $n \to \infty$. In the trade off between accuracy and computational expenses the idea is to settle for $n < \infty$, i.e. use some p^n instead of p^∞.

$$\frac{\partial KL}{\partial W_{ij}} \cong -\Big\langle \frac{\partial E(x)}{\partial W_{ij}} \Big\rangle_{p^n} + \Big\langle \frac{\partial E(x)}{\partial W_{ij}} \Big\rangle_{p^0}$$

This approximation of taking n steps instead of ∞ steps, is referred to as 'n-step learning' and the surprising thing about it is, that it allegedly works for very few steps. In fact, examples can be constructed in which 1-step learning is producing good results. This shows that a refined mathematical approach sometimes with huge advantage can be substituted by a crude but efficient approximation. It also hints to the interpretation, that when integrating over the entire data space, of course not all regions are equally important and that the data is carrying important information about which regions to choose. Loosely speaking, the bias introduced by 1-step learning is a lesser problem than the importance of the regions indicated by the data.

To ensure that this approximation will converge it is desirable to prove that it is a derivative of some cost function. To that end, the inventors of the n-step learning, i.e. Hinton, defines in [10, 11] the Contrastive Divergence function

$$CD = KL(p^0||p^\infty) - KL(p^n||p^\infty)$$

and then take the derivative of CD with respect to the parameters $\mathbf{W}$, that is

$$\frac{\partial CD}{\partial W_{ij}} = -\Big\langle \frac{\partial E(x)}{\partial W_{ij}} \Big\rangle_{p^n} + \Big\langle \frac{\partial E(x)}{\partial W_{ij}} \Big\rangle_{p^0} - \frac{\partial KL(p^n||p^\infty)}{\partial p^n} \frac{\partial p^n}{\partial W_{ij}}$$

Hinton argues further that the last term in this sum typically is very small and can be ignored. Under that assumption, the n-step learning is equivalent to optimizing the CD and the process will therefore converge.

MacKay has some examples in the unpublished note [25] which show, some examples in which the 1-step learning only converges to the correct solution if the transition operator in the MCMC is chosen the right way. In the paper introducing the Energy Based Models [52], Teh et al. are using a hybrid Monte Carlo method, which is a so-called dynamical Monte Carlo method, see e.g. [31] for details. Numerical experiments with this setup for energy based models, show only modestly successful results: Taking too few steps potentially make the results too close to the data average, and taking too many steps does not seem to improve the average sufficiently. Thus, in order to test the filtering approach a different solution to the estimation is introduced using priors which are mixtures of gaussians.

5.2 Filtering and Generative Approaches

The EBM is obviously a filtering approach, but in contrast to some filtering approaches, it is using a specific and explicit source prior. This, one may argue,

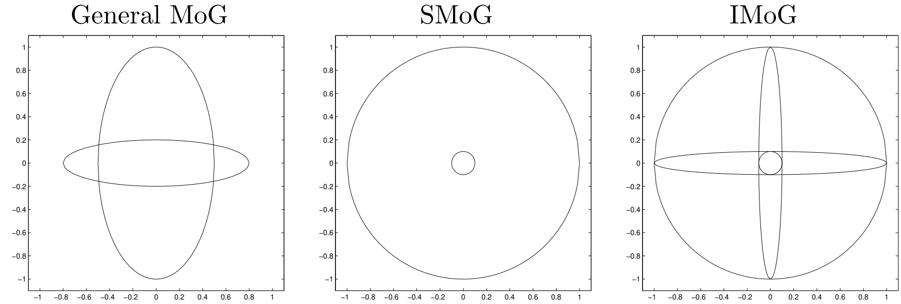


Figure 5.1: Contour plot of probability densities in 2D source space. The SMoG is the simple spherical structure and IMoG is the independent (factorized) distribution.

makes it borderline generative, but that quality offers an opportunity to compare a filtering approach with a generative approach in the overcomplete setting. To do that analytically, though, mixtures of gaussians (MoG) must be chosen as source priors to deal with the integrals involved.

5.2.1 Details of Mixture of Gaussians

Before going into details on the two approaches, a clarification on MoG is desirable to avoid common misconceptions. The general MoG and two distinct subsets of distributions that are important for the further analysis is described. The density of a d-dimensional centered MoG is

$$p(\mathbf{s}) = \sum_{\kappa} \frac{\rho_\kappa}{\sqrt{|2\pi \mathbf{D}_\kappa|}} \exp\Big[-\frac{1}{2}\mathbf{s}^T \mathbf{D}_\kappa^{-1}\mathbf{s}\Big] \tag{5.3}$$

where the weights ρ_κ sum to one. The matrix $\mathbf{D}_\kappa$ can be any positive definite matrix, but in this context often assumed diagonal, $\mathbf{D}_\kappa = \text{diag}(\sigma_{1\kappa}^2, \sigma_{2\kappa}^2, ..., \sigma_{d\kappa}^2)$. The marginal distribution for each s_i is itself a mixture of gaussians, $s_i \sim \sum_\kappa \rho_\kappa \mathcal{N}(0, \sigma_{i\kappa}^2)$, but note about the joint distribution that even if the coordinates s_i in each component are independent, i.e. if $\mathbf{D}_\kappa$ all are diagonal, the coordinates s_i are *not* in general independent in the bigger joint distribution $p(\mathbf{s})$.

Independent Mixture of Gaussian (IMoG) One subset of interest is the MoG constructed as a product of d 1-dimensional mixtures of gaussians, and therefore independent by construction. Denoting the parameters of $i'th$ marginal by $\sum_\kappa \rho_{i\kappa}\mathcal{N}(0, \sigma_{i\kappa}^2)$, the density of the joint distribution is given by

$$p(\mathbf{s}) = \sum_{\{\mathbf{k}\}} \frac{\tilde{\rho}_\mathbf{k}}{\sqrt{|2\pi \mathbf{D}_\mathbf{k}|}} \exp\Big[-\frac{1}{2}\mathbf{s}^T \mathbf{D}_\mathbf{k}^{-1}\mathbf{s}\Big] \tag{5.4}$$

The symbol $\{\mathbf{k}\}$ denotes all combinations of the vector $\mathbf{k}$ having length d, consisting of integers from 1 to K, $\mathbf{D_k} = \mathrm{diag}(\sigma^2_{1k_1}, \sigma^2_{2k_2}, ..., \sigma^2_{dk_d})$, and $\tilde{\rho}_{\mathbf{k}} = \prod_{i=1}^{d} \rho_{ik_i}$.

Spherical Mixture of Gaussian (SMoG) SMoG is a special subset of MoG with covariance matrices $\mathbf{D}_\kappa = \sigma^2_\kappa \mathbf{I}$, i.e. the density is spherical symmetric. IMoG and SMoG are subsets of MoG. Knowing the density of a MoG with diagonal covariances, its variables are independent if and only if all combinations of axis lengths are present. Therefore, since this by definition is not the case for SMoG, SMoG and IMoG are (almost) disjoint subsets of MoG, except for the case where there is only one component, i.e. when it is not a mixture, and the covariance of this single component is a positive constant times the identity.

5.2.2 MoG's for Generative Models and EBM

In [52], no assumptions on the prior is made, which makes the normalization part more difficult. To deal with this they use so-called n-step learning, a variant of a hybrid monte carlo approach, to approximately estimate the normalization part of the optimization. In this section, the prior is closer to be a MoG as Equation 1, which enables a calculation of the normalization integral in Equation 5.2 to obtain a closed form expression for the model distribution. The result is

$$p_{\mathbf{W}}(\mathbf{x}) = \sum_\kappa \gamma_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T \mathbf{W}^T \mathbf{D}_\kappa^{-1} \mathbf{W}\mathbf{x})}{\sqrt{|2\pi(\mathbf{W}^T \mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|}} \tag{5.5}$$

where γ_κ in general is dependent on $\mathbf{W}$ and $\mathbf{D}_{\kappa'}$ for all κ' through the following relation: Setting $\xi_\kappa = (|2\pi(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|/|2\pi\mathbf{D}_\kappa|)^{1/2}$, one can write $\gamma_\kappa = \rho_\kappa \xi_\kappa / \sum_{\kappa'} \rho_{\kappa'}\xi_{\kappa'}$. Note that in the square case $\xi_\kappa = 1$ and therefore $\gamma_\kappa = \rho_\kappa$ and in the case of SMoG priors $\xi_\kappa = (2\pi/\sigma^2_\kappa)^{(D-M)/2}$. The estimated optimal filtering matrix $\hat{\mathbf{W}}$ is determined by

$$\hat{\mathbf{W}} = \min_{\mathbf{W}} \Big[KL(p_0 || p_{\mathbf{W}}) \Big] = \max_{\mathbf{W}} \sum_t \ln p_{\mathbf{W}}(\mathbf{x}_t)$$

in which the gradient of the KL-divergence can be calculated analytically when the prior is chosen to mixtures of gaussians or some other analytically appealing distribution.

In the setup using a generative model, the model $\mathbf{X} = \mathbf{AS} + \mathbf{\Gamma}$ is assumed, where the additive white Gaussian noise $\mathbf{\Gamma}$ is included in order to deal with the overcomplete case. In the end, the limit of the noise variance go to zero is considered to obtain the noiseless result. Assuming white Gaussian noise and the prior on the sources $p_s(\mathbf{s})$ to be a MoG with covariance matrices $\mathbf{D}_\kappa$ and weights ρ_κ, the integration can be completed and the distribution over $\mathbf{x}$ can be written as

$$
\begin{aligned}
p(\mathbf{x}) &= \int p(\mathbf{x}|\mathbf{s})p(\mathbf{s})d\mathbf{s} \\
&= \sum_{\kappa} \rho_{\kappa} \frac{\sqrt{|2\pi\mathbf{\Phi}_{\kappa}^{-1}|}\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{\Psi}_{\kappa}\mathbf{x})}{\sqrt{|2\pi\mathbf{\Sigma}|}\sqrt{|2\pi\mathbf{D}_{\kappa}|}}
\end{aligned}
$$

where $\mathbf{\Sigma} = \sigma^2\mathbf{I}$ is the noise covariance matrix, $\mathbf{\Phi}_{\kappa} = \mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A} + \mathbf{D}_{\kappa}^{-1}$ and $\mathbf{\Psi}_{\kappa} = \mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\mathbf{A}\mathbf{\Phi}_{\kappa}^{-1}\mathbf{A}^T\mathbf{\Sigma}^{-1}$. Now, the limit of $\sigma^2 \to 0$ must be considered, but one need to do this with great care, since otherwise crucial details will vanish in the approximation. When calculating the limit of $\mathbf{\Psi}_{\kappa}$, the definition $\mathbf{\Sigma} = \sigma^2\mathbf{I}$ is inserted and the expression simplifies into

$$
\mathbf{\Psi}_{\kappa} = \frac{1}{\sigma^2}\left\{\mathbf{I} - \mathbf{A}(\mathbf{A}^T\mathbf{A} + \sigma^2\mathbf{D}_{\kappa}^{-1})\mathbf{A}^T\right\}
$$

Now defining $\mathbf{Q} = \mathbf{A}\mathbf{D}_{\kappa}\mathbf{A}^T/\sigma^2$ one can use Woodburys identity for inverse matrices and obtain

$$
\mathbf{A}(\mathbf{A}^T\mathbf{A} + \sigma^2\mathbf{D}_{\kappa}^{-1})\mathbf{A}^T = \mathbf{Q} - \mathbf{Q}(\mathbf{I} + \mathbf{Q})^{-1}\mathbf{Q}
$$

Since $\mathbf{Q}$ is symmetric and with very large values compared to $\mathbf{I}$, the right hand side can be approximated by $\mathbf{I} - \mathbf{Q}^{-1}$. To see this, write $\mathbf{Q}$ as $\mathbf{Q} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, for an orthogonal $\mathbf{V}$ and diagonal $\mathbf{\Lambda}$ and remember the identity $x - x^2/(1+x) = 1 - 1/(1+x)$ to obtain $\mathbf{I} - \mathbf{V}\mathrm{diag}(1/(1+\Lambda_{ii}/\sigma^2))\mathbf{V}^T \cong \mathbf{I} - \mathbf{V}\mathrm{diag}(1/(\Lambda_{ii}/\sigma^2))\mathbf{V}^T = \mathbf{I} - \mathbf{Q}^{-1}$. Inserting this into the equation containing $\mathbf{\Psi}_{\kappa}$, gives $\mathbf{\Phi}_k \to (\mathbf{A}^T\mathbf{D}_k\mathbf{A})^{-1}$.

When calculating the limit of the fraction containing the determinant $|2\pi\mathbf{\Phi}_{\kappa}^{-1}|$, one must use the fact that since $\mathbf{A}^T\mathbf{A}$ is symmetric there exists an orthogonal $\mathbf{V}$ such that $\mathbf{A}^T\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. Since $|\mathbf{V}| = 1$ it holds that

$$
|2\pi\mathbf{\Phi}_{\kappa}^{-1}|/|2\pi\mathbf{\Sigma}| = (2\pi\sigma^2)^{(d-m)}/|\mathbf{\Lambda} + \sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V}|
$$

Because σ^2 is assumed arbitrary small, only the diagonal of the sum of matrices contributes significantly to the determinant. And since furthermore $\mathbf{A}^T\mathbf{A}$ has rank m, the m first elements of the diagonal matrix will dominate together with the remaining $m - d$ factors

$$
|\mathbf{\Lambda} + \sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V}| \cong \prod_{i=1}^{m}\mathbf{\Lambda}_{ii}\prod_{j=m+1}^{d}(\sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V})_{jj}
$$

inserting this into the fraction above gives $\sqrt{|2\pi\mathbf{\Phi}_{\kappa}^{-1}|}/\sqrt{|2\pi\mathbf{\Sigma}|} \to w_{\kappa}/\sqrt{|\mathbf{A}\mathbf{A}^T|}$ where w_{κ} will be explained shortly.

The coefficients α_{κ} has the structure $\rho_{\kappa}w_{\kappa}\sqrt{|2\pi\mathbf{A}\mathbf{D}_{\kappa}\mathbf{A}^T|}/\sqrt{|\mathbf{A}\mathbf{A}^T|\cdot|2\pi\mathbf{D}_{\kappa}|}$. In the square case much of the difficulties of taking the noiseless limit disappears,

$w_\kappa = 1$ and one easily obtains $\alpha_\kappa = \rho_\kappa$. In the case of SMoG, setting $\mathbf{D}_\kappa = \sigma_\kappa^2 \mathbf{I}$, it is true that $w_\kappa = (2\pi\sigma_\kappa^2)^{(d-m)/2}$ and therefore $\alpha_\kappa = \rho_\kappa$. Supported by numerical results it is conjecture that this is also the case for the general overcomplete case. In short, the crucial limits are

$$\begin{aligned} \mathbf{\Psi}_\kappa &\rightarrow (\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T)^{-1} \\ \sqrt{|2\pi\mathbf{\Phi}_\kappa^{-1}|}/\sqrt{|2\pi\mathbf{\Sigma}|} &\rightarrow w_\kappa/\sqrt{|\mathbf{A}\mathbf{A}^T|} \end{aligned}$$

Here $w_\kappa = \prod_{i=m+1}^{d}(2\pi/(\mathbf{V}^T\mathbf{D}_\kappa^{-1}\mathbf{V})_{ii})^{1/2}$ is a constant with respect to σ^2, but depends on $\mathbf{D}_\kappa$ and on $\mathbf{A}$ through the unique orthogonal matrix $\mathbf{V}$. $\mathbf{V}$ fulfills the eigenvalue equation $\mathbf{A}^T\mathbf{A}\mathbf{V} = \mathbf{V}\mathbf{\Lambda}$, and eigenvalues in $\mathbf{\Lambda}$ are decreasing in size down the diagonal. The expression for w_κ is $w_\kappa = \prod_{i=m+1}^{d}(2\pi/(\mathbf{V}^T\mathbf{D}_\kappa^{-1}\mathbf{V})_{ii})^{1/2}$. With this limit taken care of, the maximum likelihood expression for $p(\mathbf{x})$ can be written as

$$p_{\mathbf{A}}(\mathbf{x}) = \sum_\kappa \rho_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T(\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T)^{-1}\mathbf{x})}{\sqrt{|2\pi\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T|}} \tag{5.6}$$

where the weights somewhat surprisingly turns out to be same as those of the prior.

All in all, the estimated generative mixing matrix $\hat{\mathbf{A}}$ is for the zero noise limit, the matrix maximizing the log likelihood

$$\hat{\mathbf{A}} = \max_{\mathbf{A}} \left[\ln \prod_{t=1}^{N} p_{\mathbf{A}}(\mathbf{x}_t) \right]$$

Note that the sources in this process has not been estimated, only the mixing matrix itself.

5.2.3 Bias Impact

Now the EBM and GM approaches derived in the previous section are compared and the significance of the differences and similarities is discussed. In the square case, total equivalence of all expressions is obtained by setting $\mathbf{W} = \mathbf{A}^{-1}$ and thus not surprisingly one can conclude, as done in [52], that the two approaches are equivalent when the number of observations equal the number of sources. Therefore the discussion and comparison in this subsection is almost entirely concerned with the overcomplete case.

In the overcomplete case it is not obvious how one should compare results on the generative $\mathbf{A}$ and the filtering matrix $\mathbf{W}$. The filtering approach does not retrieve the original sources, since for any matrix $\mathbf{W}$ it holds that $\mathbf{W}\mathbf{A}_g \neq \mathbf{I}$ because of the dimensionality: It is impossible to construct d m-dimensional orthogonal vectors when $d > m$. And one cannot in general compare the filter matrix $\mathbf{W}$ with the pseudo-inverse of $\mathbf{A}$, since this is not the optimal solution in all cases, see [23]. But using MoG as prior, both the model distribution $p_{\mathbf{W}}(\mathbf{x})$

of the EBM and the likelihood $p_{\mathbf{A}}(\mathbf{x})$ of the GM become MoG's with parameters which must be estimated to fit a common data set $\mathbf{X}$. In fact one ends up with two expressions which look rather similar

$$\max_{\mathbf{W}} \sum_{t=1}^{N} \ln p_{\mathbf{W}}(\mathbf{x}_t) \qquad \max_{\mathbf{A}} \sum_{t=1}^{N} \ln p_{\mathbf{A}}(\mathbf{x}_t) \tag{5.7}$$

The similarity is to some extend both genuine and deceptive: Both distributions are MoG, but the dependency of the weights and covariances on $\mathbf{W}$ and $\mathbf{A}$ are different. In this subsection EBM and GM are analyzed by comparing the covariances and weights of their MoG's.

5.2.3.1 Spherical MoG

First the special case when the priors are SMoG, i.e. the variances given as $\mathbf{D}_\kappa = \sigma_\kappa^2 \mathbf{I}$ is considered. In this case the sources are not assumed to be independent, which is interesting in itself, but it also serves as a clear example of properties which hold for the independent case as well. When assuming the priors to be SMoG, the model distribution simplifies significantly,

$$p_{\mathbf{W}}(\mathbf{x}) = \sum_\kappa \gamma_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W}\mathbf{x})}{\sqrt{|2\pi(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|}} \tag{5.8}$$

$$\gamma_\kappa = \frac{\rho_\kappa (1/\sigma_\kappa^2)^d}{\sum_{\kappa'} \rho_{\kappa'} (1/\sigma_{\kappa'}^2)^d}$$

where $\tilde{d} = (d-m)/2$.

The covariances of EBM and GM respectively can in fact become equal in the case of SMoG. The equation setting the covariances equal

$$(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1} = \mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T \qquad \forall\kappa \tag{5.9}$$

translates into $\mathbf{W}^T\mathbf{W}\mathbf{A}\mathbf{A}^T = \mathbf{I}$, which is fulfilled for $\mathbf{W} = \mathbf{A}^+$, where $\mathbf{A}^+$ denotes the pseudo inverse (Moore-Penrose) of the matrix $\mathbf{A}$. When $\mathbf{A}$ has full rank, the pseudo-inverse of the overcomplete $\mathbf{A}$ is given by $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$. But the equation is also fulfilled for any matrix $\mathbf{W} = \mathbf{U}\mathbf{A}^+$, where $\mathbf{U}$ is orthogonal and thus, there is an entire family of matrices which would make the covariances of the EBM equal to the covariances of the GM for a given $\mathbf{A}$. Conversely for any $\mathbf{W}$ one can choose $\mathbf{A} = \mathbf{W}^+$ to obtain the same covariances and and in this sense the two approaches have equal flexibility with respect to adjusting the covariances to the data. This is evident in the 4×2 example in Fig 5.2 a) and b).

The weights can be compared by examining the ratio $\gamma_\kappa/\rho_\kappa$ and this ratio differs strongly from 1 in most cases. From the expression of γ_κ in Eq. 5.8, it is

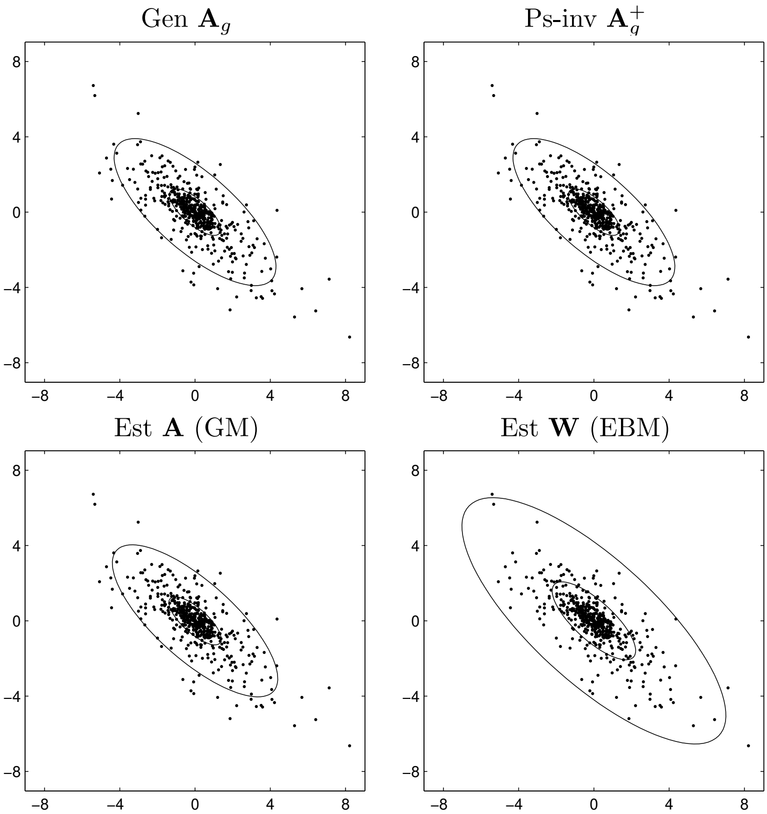


Figure 5.2: The covariances in case of SMoG priors for different settings. Plot a) Generative $\mathbf{A}$ in the GM. Plot b) Pseudo-inverse of the generative $\mathbf{A}$ in the EBM. Plot c) Estimated $\mathbf{A}$ in the GM. Plot d) Estimated $\mathbf{W}$ in the EBM.

clear that the constraint making the weights of GM and EBM equal is

$$\frac{(1/\sigma_\kappa^2)^d}{\sum_{\kappa'} \rho_{\kappa'} (1/\sigma_{\kappa'}^2)^d} \stackrel{?}{=} 1 \qquad \forall \kappa$$

which is impossible when the weights σ_κ^2 must be different for different κ. Thus, in the SMoG case, the weights of EBM and GM cannot be equal and furthermore the ratio $\gamma_\kappa/\rho_\kappa$ becomes relatively large for those κ where σ_κ^2 is very small and vice versa. Fig 5.3 is an illustration of this. Here a SMoG prior is assumed with two components is assumed which has $\sigma_1^2 = 1$ and $\rho_1 = \rho_2 = 0.5$ and σ_2^2 is varied between 0 and 2. The resulting ratio γ_2/ρ_2 is shown in Fig 5.3 a): The x-axis is σ_2^2, the y-axis is γ_2/ρ_2 and the 4 curves are plotted for different degrees of overcompleteness $\tilde{d}$=0 (Square) and $\tilde{d} = 1, 2, 10$ (Overcomplete). Clearly only

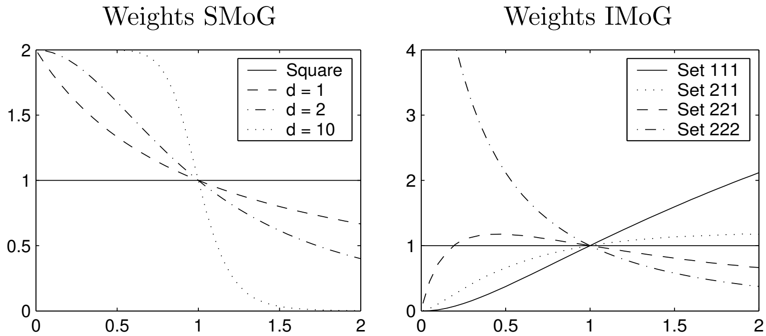


Figure 5.3: The Different Weights for EBM and GM. Plot a) demonstrates for SMoG priors that the differences in weights increase dramatically when the settings gets strongly overcomplete. Plot b) demonstrates that also for IMoG there is a significant difference in the weights. (See the text for details on the plots)

in the square case and for σ_2^2 values close to 1, is the ratio reasonably close to 1.

Another more specific example is shown in Fig 5.2 c) and d), which is the estimated covariances for GM and EBM in a 4×2 case. In this example the effect of the enhanced weight on smaller covariances is clear: When the weights of the smaller covariance is strong, the points far from origin is considered extreme, and the covariances are expanded accordingly. Thus, for this reason, EBM seem to favor larger covariances compared to GM.

5.2.3.2 Independent MoG

Now the case where the prior corresponds to a independent distribution in the sources, i.e. the prior is an IMoG is considered.

The covariance equation, Eq. 5.9, no longer simplifies such that the pseudo-inverse can be used to obtain equality. In fact, one can prove that there does not exists a $\mathbf{W}$ which fulfils Eq. 5.9 for any given $\mathbf{A}$ by the following argument: Assume Eq. 5.9 to be fulfilled, then the matrix $\mathbf{D}_\kappa^{-1}\mathbf{W}\mathbf{A}\mathbf{D}_\kappa$ must be independent of $\mathbf{D}_\kappa$. Since $\mathbf{D}_\kappa$ is diagonal, it holds that $(\mathbf{D}_\kappa^{-1}\mathbf{W}\mathbf{A}\mathbf{D}_\kappa)_{ij} = (\mathbf{W}\mathbf{A})_{ij}\sigma_{\kappa_i}^2/\sigma_{\kappa_j}^2$ and therefore $(\mathbf{W}\mathbf{A})_{ij}$ must be zero for $i \neq j$ to fulfil this constraint. But because of the dimensionality of $\mathbf{A}$ and $\mathbf{W}$ this is not possible. So the covariances of EBM and GM cannot become exactly equal and thus will fit slightly different to the observed data set $\mathbf{X}$.

Compared to the SMoG prior, the weights of the IMoG case are much more complicated because it is not possible to remove the matrix $\mathbf{W}$ from the expres-

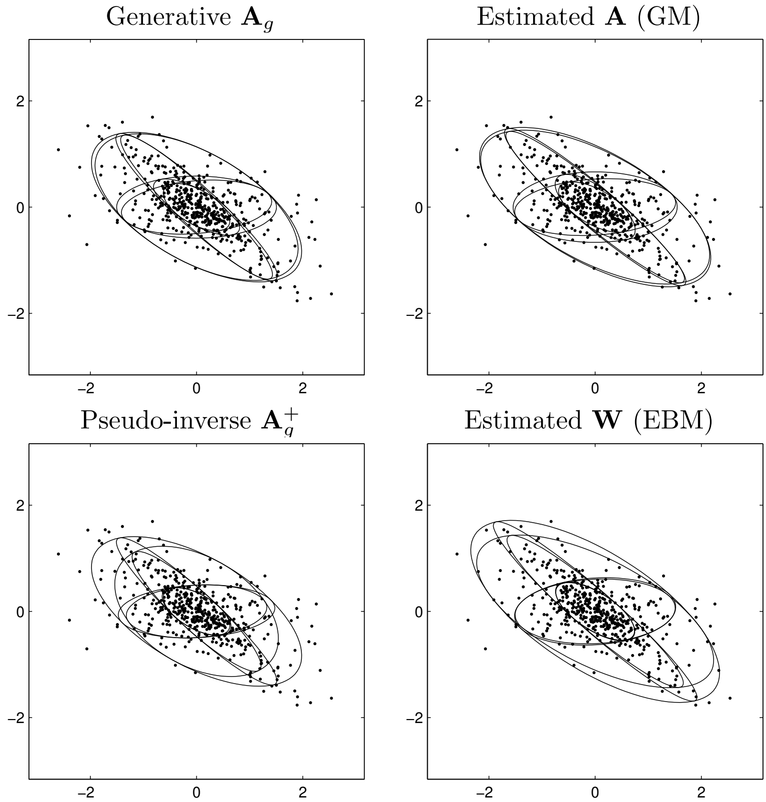


Figure 5.4: Example of MoG's adjusted to data in a 3×2 setting. The dots are the 2 dimensional data and the contours are the covariances (two standard deviations) of the MoG's.

sion. The weights are given by

$$\gamma_\kappa = \frac{\rho_\kappa \frac{\sqrt{|2\pi(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|}}{\sqrt{|2\pi\mathbf{D}_\kappa|}}}{\sum_{\kappa'} \rho_{\kappa'} \frac{\sqrt{|2\pi(\mathbf{W}^T\mathbf{D}_{\kappa'}^{-1}\mathbf{W})^{-1}|}}{\sqrt{|2\pi\mathbf{D}_{\kappa'}|}}}$$

As previously, the ratio $\gamma_\kappa/\rho_\kappa$ is investigated as σ_2^2 varies but since γ_κ depends on $\mathbf{W}$, focus is on the ratio $\langle\gamma_\kappa\rangle/\rho_\kappa$, where $\langle\gamma_\kappa\rangle$ is the average weight assuming $\mathbf{W}$ to be distributed as a normalized gaussian, i.e.

$$\langle\gamma_\kappa\rangle = \int \gamma_\kappa(\mathbf{W})\mathcal{N}_{\mathbf{W}}(\mathbf{0}, \mathbf{I})\mathrm{d}\mathbf{W}$$

As an example consider a 2×3 case, that is, 3 sources and 2 observations, where the sources each are distributed as a 1 dimensional MoG with variances σ_1^2 and σ_2^2. With respect to the average the 8 weights $\langle\gamma_\kappa\rangle$ can be grouped into 4 distinct groups (denoted 111, 112, 122, 222) depending on the combination of variances σ_1^2 and σ_2^2. The ratio $\langle\gamma_\kappa\rangle/\rho_\kappa$ is investigated in Fig 5.3 b): σ_1^2 is set to 1, σ_2^2 is varied on the x-axis and $\langle\gamma_\kappa\rangle/\rho_\kappa$ is on the y-axis.

As before, $\langle\gamma_\kappa\rangle/\rho_\kappa = 1$ would indicate that EBM and GM have the same weights, which is clearly not the case. In this example, as is true in general, an essential difference is that among the covariances in the IMoG there is always matrices of the form $\mathbf{D}_{\kappa'} = \sigma_{\kappa'}^2\mathbf{I}$ and therefore at least one of the $\gamma_{\kappa'}$ will becomes relatively large for small $\sigma_{\kappa'}^2$ and since all the weights γ_κ are connected through the normalization, this influences all the weights. Consequentially EBM tends to have larger covariances than GM also for the case of independent IMoG priors. Fig 5.4 demonstrates this for a 2×3 example, but the effect is not as extreme as in Fig 5.2, due to the smaller difference in dimensions.

5.2.3.3 Discussion

The analysis is considering the two distributions $p_\mathbf{W}(\mathbf{x})$ and $p_\mathbf{A}(\mathbf{x})$ which for source priors chosen to be mixture of Gaussians, themselves are mixture of Gaussians - but differently parameterized. The two distributions are defined by the equations

$$\begin{aligned} p_\mathbf{W}(\mathbf{x}) &= \frac{p_s(\mathbf{Wx})}{\int p_s(\mathbf{Wx})d\mathbf{x}} \\ p_\mathbf{A}(\mathbf{x}) &= \lim_{\sigma^2\to 0}\int p(\mathbf{x}|\mathbf{s},\sigma^2)p_s(\mathbf{s})d\mathbf{s} \end{aligned}$$

The tempting conclusion, that when $\sigma^2 \to 0$ then the noise density becomes a delta function and the two approaches becomes equivalent, is demonstrated not to be quite true. For MoG source priors, both $p_\mathbf{A}$ and $p_\mathbf{W}$ are themselves MoG, but it is not true that the limit of the noise is interchangeable with the limit of the integral. That is, the detailed analysis demonstrates, that one cannot in general conclude that $p_\mathbf{A}$ is approaching $p_s(\mathbf{A}^+\mathbf{x})/|\mathbf{Jac}|$, where $\mathbf{Jac}$ is the Jacobian.

The question about which approach is better, the generative or the filtering, is not answered on a detailed level by this analysis. But the filtering approach has the property in the overcomplete case that the covariance are somewhat overestimated compared to the model generating the data. And also, preliminary numerical experiments suggests, that the generative model seem to reach slightly higher levels with respect to the cost function.

Summary of Chapter 5

In this Chapter, the Energy Based Models (EBM) have been introduced as a representative for the filtering approach. The relationship between the EBM and the generative models (GM) is analyzed by choosing the source prior distribution to be a mixture of Gaussians. Doing that it has been demonstrated, that EBM's have a slight bias in general and for so-called Spherical mixture of Gaussians in particular.

Chapter 6

Summary

> "What is important is to keep learning, to enjoy challenge, and to tolerate ambiguity. In the end there are no certain answers." *Martina Horner, President of Radcliffe College*

No matter the context in which it was said, the above quote should remind us that even in the so-called exact sciences, there are no certain answers. The best we can hope for in scientific areas such as physics, is improved trust to a theory which has passed an important and significant attempt to falsification. In mathematical modeling, on the other hand, one might believe that exact and exhaustive investigations are possible, but due to the fundamentally difficult problems of analytical integration and multidimensional global optimization, it is far from the truth. Scientists in the field of mathematical modeling, must still select experiments carefully to retrieve as much information possible with what always seems as too limited resources. In this thesis I have, in collaboration with colleagues, attempted to make informative numerical simulations and analytical investigations, in order to understand the properties and inner workings of the Mean Field ICA algorithm.

This final chapter summarizes the findings of the work presented in this thesis and suggests new directions for the further development and improvement of the Mean Field ICA algorithm.

6.1 Conclusions

The analysis and simulations of the EM algorithm and the combined Mean Field ICA algorithm, leads to the following comments and conclusions:

a) The Laplace approximation proves that in the low noise regime, the EM algorithm requires an intolerable number of iterations in order to converge. In fact, the number of iterations needed for convergence goes to infinity as the noise goes to zero. A further analysis of the variational Bayes EM algorithm proves that this approach is suffering from the exact same defect.

b) The above conclusion based on the Laplace approximation only utilizes the zeroth order term, and thus not any particular property of the source prior. Therefore, the results regarding the slow convergence holds not only for the traditional instantaneous ICA problem, but also for the convolutive ICA and models such as linear state space models, Kalman filter, Probabilistic PCA (pPCA), Factor Analysis (FA), etc.

c) The Laplace approximation also demonstrates that the first order correction to the update of the mixing matrix, is proportional to the update of the noiseless Bell & Sejnowski algorithm. This shows that the noise model is not providing a different estimate of the mixing matrix, unless the noise variance σ^2 is sufficiently large to make σ^4 important.

d) The so-called Hessian argument demonstrates that the slow convergence property, is not closely related to the linearity of the transformation function, but rather to the additive Gaussian noise.

e) Analysis and simulation shows that adjustments to the optimization improves the performance. Especially the so-called Easy Gradient Recipe has proven to be an efficient tool for doing advance quasi-Newton optimization, using nothing more than the EM steps. The improvements obtained by the Easy Gradient Recipe are indeed impressive.

f) Simulations suggests, that constraining the parameters of the Mean Field ICA improves the accuracy of that specific parameter. The accuracy on the other parameters, however, does not seem to change significantly.

g) Examples demonstrates that choosing a source prior correctly, e.g. to be positive, makes a critical difference in some cases. Also, in an overcomplete speech example, the linear response approach prevails where the ordinary factorized model repeatedly fails to estimate the mixing matrix. This indicates that linear response is vital for overcomplete problems.

h) The convolutive Mean Field ICA algorithm has been derived and proven successfully on a problem with filter length forty.

Finally, the analysis on the Energy Based Models – as an example of the filtering approach – shows that in the overcomplete setting, the filtering and generative approach are rather different, even when their cost functions are from the same family of distributions, namely mixture of Gaussians. In particular it is found that

i) In the special case of spherical mixtures of gaussians (SMoG) as source priors, the energy based models overestimates the covariances of the data. In that sense, the generative approach is outperforming the energy based models for SMoG priors. A similar, but less extreme effect is apparent for independent mixture of Gaussians (IMoG).

6.2 Outlook

The above conclusions provides some indications about the directions in which the research on the Mean Field ICA would be beneficial.

The most important extension to the Mean Field ICA is to combine the improved optimization with the advanced corrections techniques of the linear response and adaptive TAP. In order to use the Easy Gradient Recipe, one must have a cost function and a gradient, and for the linear response it has so far not been possible to construct a cost function which corresponds to the improved estimate of the source posterior covariance. But this is not the case for the adaptive TAP approach, which has a well defined cost function. A natural and indeed powerful extension of the work done so far, would be to use the Easy Gradient Recipe for optimization of the adaptive TAP estimates. Doing that, one would combine the efficiency of the new powerful optimization with the special abilities of the advanced estimates of the source posterior covariance to obtain fast and accurate results in overcomplete settings.

A different area in which the Mean Field ICA could still be developed is the computational complexity: The most computationally expensive part of the Mean Field ICA is the E-step in which the mean field equations are iterated until some reasonable convergence in the mean values is reached. This iterated update must be sequential to ensure convergence and is therefore rather expensive. Since more than 95% of the computational power is spend on this part of the algorithm, any improvements of this part of the process will drastically improve the overall performance of the algorithm. One way the process might get more efficient is to rewrite the mean field equation such that instead of iterating the mean function and the gamma variable successively, the problem is reformulated to be a zero of a slightly different function. In order to find this zero, one might use off-the-shelf optimization methods. This would make the E-step far more efficient in much the same was as the M-step was made more efficient by using the quasi-Newton optimization technique.

The Mean Field ICA has been very promising because of its rather modular nature which has opened up for the use of a wider selection of source priors and constraints on the parameters of the model. It has also, however, proved difficult to make all the many aspects and components of the Mean Field ICA algorithm work together to obtain a robust and consistent estimation. As for many algorithms as complex as the Mean Field ICA, robustness rather than new opportunities must be resolved for the Mean Field ICA to move from the proof of concept phase to be a trusted standard algorithm to be used in a broader scientific context. That being said, I believe it to be true that the Mean Field ICA algorithm not only has contributed significantly to the ICA community but also shown to a larger audience, that combining results from different research areas is indeed both promising and inspiring.

Appendix A

Dictionary

Annealing Annealing, or "Simulated Annealing" as it is also called, is a technique in which the temperature (which is often the noise) is kept under control during search of energy minima. In the beginning the temperature is kept high because this typically makes the number of false minima smaller and then graduatly the temperature is lowered to the level of interest.

Bethe Approximation The Bethe approximation is an approximation to the Gibbs free energy. In the typical mean field approach the approximating probability distribution is completely factorized, while in the Bethe approximation the approximating probability distribution is a product of factorized one and two-variable joint probabilities

Cavity Approach The Cavity approach is a way to calculate the approximate marginal probabilities in a spin system. The idea is to assume weak dependencies between the spins and then consider the spin system with the particular i.th spin of interest removed. With the cavity approach one can by some second order expansion rederive the TAP equations.

CDMA Code Division Multiple Access.

Energy Based Models The energy based model, is a separation approach different from Maximum likelihood and Information Maximization. It filters the observed signal and estimate the value of the filter by adjusting it such that a certain model distribution is closest possible to the data distribution. The energy based model has it name from the fact that the model distribution is constructed from an energy defined to the problem at hand.

FIR Finite Impulse Response.

Free Energy Free Energy Talking loosely about the "free energy", one often means the "Gibbs free energy". From some physical system one can at least in principle construct the Gibbs free energy from which many important other entities can be calculated. Among these entities are mean values and second moments. The Gibbs free energy is often minus the log likelihood and minimization of the Gibbs free energy therefore in those cases corresponds to the maximum likelihood approach.

Hamiltonian The Hamiltonian of some system is the total energy function. Often this function, when used as a cost function, must be minimized.

IIR Infinite Impulse Response.

Kikuchi Approximation Kikuchi Approximation The Kikuchi approximation is an approximation to the Gibbs free energy. In the naive mean field approach, the approximating distribution factorizes completely and in the Bethe approximation also two-variable joint probabilities are considered. In the Kikuchi approximation, joint probabilities of ever more variables are included in the approximating distribution and the accordingly more and more flexible free energy approximation is calculated.

Linear Response In the Naive mean field approximation, the approximating distribution is a product distribution and thus the susceptibility matrix is diagonal, i.e. the second moments factorizes. In the linear response approach one tries to improve this estimate of the susceptibility by differentiating the mean spin values with respect to the external field. For the true mean values, this yields a non-diagonal susceptibility and if the estimation of the mean spin values is good, this new estimation of the susceptibility is good as well.

MIMO Multiple Input Multiple Output.

Monte Carlo The term "Monte Carlo" covers a vast range of techniques to estimate mean values. The basic idea is that the mean values are based on more or less random samples from the variable space or random next-step in the variable space.

Naive Mean Field The Naive Naive Mean field is an approximation in which there is assumed no interaction between the variables. Often the interaction of the system is put into the external field of the naive mean field approximation as in Naive Mean Field ICA.

Newton Method The Newton Method is a way of updating the parameters of an energy landscape. Instead of the more primitive steepest descent, the Newton Method approximates the energy landscape with a quadratic landscape through a second order expansion in the present point. This is done using the gradient and the (inverse) hessian evaluated in the present point. In each step it moves directly to the minimum of the approximated quadratic landscape. Often the inverse hessian is rather computationally expensive and therefore approximations to this entity is used in so-called "Quasi-Newton Methods". Among these approximation techniques one finds Broyden-Fletcher-Goldfarb-Shanno (BFGS) and Davidson-Fletcher-Powell (DFP).

Non-Stationarity A signal which is non-stationary has statistical properties, such as moments, which change over time. The notion of non-stationarity, of course, applies for any stochastic process which has time-variant parameters.

Phase Transition Some physical systems like spin systems, have macroscopic properties, which change suddenly when parameters like temperature or pressure changes. These changes are loosely referred to as phase transitions. More formally a phase transition of a physical system is a point or manifold in parameters space in which the hamiltonian has a non continuity in the first, second or higher order derivative.

Replica Method Often one wants to analytically calculate some rather general averages, e.g. the generalization error. This can be done by use of a mathematical property which rewrite the free energy into an expression which is easier to handle mathematically. Well, easier, but still not really easy. This technique for analytical calculation is called the replica method.

Representation Often the term representation is used in the interpretation of the overcomplete situation where the columns of the mixing matrix is regarded as basis functions and the coordinates of the sources as weights applied to these basis function. In this picture, the sources "represents" the signal in the basis expressed by the columns of the mixing matrix.

SIMO Single Input Multiple Output.

Spin Glass A spin glass is a spin system which for low temperatures has a state in which the spins are not ordered. For standard spin systems, all the spins will flip in to some specific order like all-up, all-down, up-down or maybe even down-up-up. But for the spin glasses the Hamiltonian does not impose such a structure. If the interaction of the hamiltonian term is such that so called "frustration" can occur, the spin system is a spin glass. The low-temperature non-ordered state is called the spin glass phase.

Spin System In a spin system defined by some Hamiltonian, one can ask of the mean spin values and the second moments of the spin variables. The susceptibility is the fluctuations of the second moments around their mean values, i.e. the covariance matrix of the spin values.

TAP Equations The acronym TAP means "Thouless, Anderson, Palmer" since these three first presented a derivation which led to the TAP equations. Basically the TAP equations are fix point equations for the mean spin values which approximate the interaction between the spins by a certain reaction term. There are several ways to derive the TAP equations among these the cavity approach and heuristics.

Appendix B

Calculations

B.0.1 The Convolutive Model

The derivative of the log likelihood with respect to the entry A^{τ}_{ij} is given by

$$\begin{aligned}
& \frac{\partial}{\partial A^{\tau}_{ij}} \ln P(\mathbf{X}|\mathbf{A},\mathbf{W}) \\
= & \frac{1}{P(\mathbf{X}|\mathbf{A},\mathbf{W})} \frac{\partial}{\partial A^{\tau}_{ij}} \int P(\mathbf{X}|\mathbf{A},\mathbf{W},\mathbf{S}) P(\mathbf{S}) d\mathbf{S} \\
= & \int P(\mathbf{S}|\mathbf{X},\mathbf{A},\mathbf{W}) \frac{\partial}{\partial A^{\tau}_{ij}} \sum_{t=1}^{N} \Big[-\frac{1}{2}(\mathbf{x}_t - \mathbf{f}_t)^T \mathbf{W}^{-1} (\mathbf{x}_t - \mathbf{f}_t) \Big] d\mathbf{S} \\
= & \langle \sum_{t=1}^{N} \Big\{ (\mathbf{W}^{-1}\mathbf{X})_{it} s_{j,t-\tau} - (\mathbf{W}^{-1}\mathbf{f}) s_{j,t-\tau} \Big\} \rangle
\end{aligned}$$

Setting this expression equal to zero, the noise covariance matrix $\mathbf{W}$ can be removed

$$0 = \sum_{t=1}^{N} \Big\{ X_{it} \langle S_{j,t-\tau} \rangle - \sum_{\tau'=0}^{F-1} \sum_{k=1}^{d} A^{\tau'}_{ik} \langle S_{k,t-\tau'} S_{j,t-\tau} \rangle \Big\}$$

and isolating A^{τ}_{ij} in this expression, one obtains the result in Equation 2.21.

Similar with the noise covariance matrix if one differentiates without special attention to keeping $\mathbf{W}$ positive definite as it must be. It turns out that the result obtained is an outer product and therefore ensuring the solution to be positive definite and furthermore that this solution is in fact the maximum likelihood estimate. Differentiating the log likelihood with respect to W_{ij} and setting it to zero, one obtains

$$
\begin{aligned}
0 &= \frac{\partial}{\partial W_{ij}} \ln P(\mathbf{X}|\mathbf{A},\mathbf{W}) \\
&= \frac{1}{P(\mathbf{X}|\mathbf{A},\mathbf{W})} \frac{\partial}{\partial W_{ij}} \int d\mathbf{S} P(\mathbf{S}) P(\mathbf{X}|\mathbf{A},\mathbf{W},\mathbf{S}) \\
&= \int d\mathbf{S} \frac{P(\mathbf{S})}{P(\mathbf{X}|\mathbf{A},\mathbf{W})} \frac{\partial}{\partial W_{ij}} C \exp\Big[\sum_{t=1}^{N} -\frac{1}{2}(\mathbf{x}_t - \mathbf{f}_t)^T \mathbf{W}^{-1}(\mathbf{x}_t - \mathbf{f}_t)\Big] \\
&= \int d\mathbf{S} \frac{P(\mathbf{S})}{P(\mathbf{X}|\mathbf{A},\mathbf{W})} \Big\{ \frac{\partial C}{\partial W_{ij}} \exp(...) + C \frac{\partial}{\partial W_{ij}} \exp(...) \Big\} \\
&= \langle \frac{\partial \ln C}{\partial W_{ij}} + \frac{\partial}{\partial W_{ij}} \sum_{t=1}^{N} -\frac{1}{2}(\mathbf{x}_t - \mathbf{f}_t)^T \mathbf{W}^{-1}(\mathbf{x}_t - \mathbf{f}_t) \rangle \\
&= \langle -\frac{N}{2} \frac{\partial \ln \det \mathbf{W}}{\partial W_{ij}} - \frac{1}{2} \frac{\partial}{\partial W_{ij}} \mathrm{Tr}\Big[(\mathbf{X} - \mathbf{f})^T \mathbf{W}^{-1} (\mathbf{X} - \mathbf{f})\Big] \rangle \\
&= \langle -\frac{N}{2} W_{ij}^{-1} + \frac{1}{2} (\mathbf{W}^{-1}(\mathbf{X} - \mathbf{f})(\mathbf{X} - \mathbf{f})^T W^{-1})_{ij} \rangle
\end{aligned}
$$

That is the solution is

$$
\begin{aligned}
\hat{\mathbf{W}} &= \frac{1}{N} \langle (\mathbf{X} - \mathbf{f})(\mathbf{X} - \mathbf{f})^T \rangle \\
&= \frac{1}{N} \langle (\mathbf{X}\mathbf{X}^T - \mathbf{X}\mathbf{f}^T - \mathbf{f}\mathbf{X}^T + \mathbf{f}\mathbf{f}^T) \rangle \\
&= \frac{1}{N} \langle \mathbf{X}\mathbf{X}^T - \sum_{t=1}^{N} \sum_{\tau=0}^{F-1} \big(\mathbf{x}_t \mathbf{s}_{t-\tau}^T (\mathbf{A}^\tau)^T + \mathbf{A}^\tau s_{t-\tau} \mathbf{x}_t^T \big) \\
&\qquad + \sum_{t=1}^{N} \sum_{\tau,\tau'=1}^{F-1} \mathbf{A}^\tau \mathbf{s}_{t-\tau} \mathbf{s}_{t-\tau'}^T (\mathbf{A}^{\tau'})^T \rangle \\
&\cong \frac{1}{N} \Big[\mathbf{X}\mathbf{X}^T - \sum_{t=1}^{N} \sum_{\tau=0}^{F-1} \big(\mathbf{x}_t \mathbf{m}_{t-\tau}^T (\mathbf{A}^\tau)^T + \mathbf{A}^\tau \mathbf{m}_{t-\tau} \mathbf{x}_t^T \big) \\
&\qquad \sum_{t=1}^{N} \sum_{\tau,\tau'=1}^{F-1} \mathbf{A}^\tau \mathbf{M}_{t-\tau'}^{t-\tau} (\mathbf{A}^{\tau'})^T \Big]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{N}\Big[\mathbf{X}\mathbf{X}^T - \sum_{\tau=0}^{F-1}\Big(\mathbf{F}^\tau(\mathbf{A}^\tau)^T + \mathbf{A}^\tau(\mathbf{F}^\tau)^T\Big) \\
&\qquad + \sum_{\tau,\tau'=1}^{F-1}\mathbf{A}^\tau\mathbf{G}^{\tau,\tau'}(\mathbf{A}^{\tau'})^T\Big] \\
&= \frac{1}{N}\Big[\mathbf{X}\mathbf{X}^T - \sum_{\tau=0}^{F-1}\Big(\mathbf{F}^\tau(\mathbf{A}^\tau)^T + \mathbf{A}^\tau(\mathbf{F}^\tau)^T\Big) \\
&\qquad + \sum_{\tau,\tau'=1}^{F-1}\mathbf{A}^\tau\mathbf{G}^{\tau,\tau'}(\mathbf{A}^{\tau'})^T\Big] \\
&= \frac{1}{N}\Big[\mathbf{X}\mathbf{X}^T - \Big(\underline{\mathbf{F}\mathbf{A}}^T + \underline{\mathbf{A}\mathbf{F}}^T\Big) + \underline{\mathbf{A}\mathbf{G}\mathbf{A}}^T\Big]
\end{aligned}
$$

Which is the solution in Equation 2.22.

In Chapter 2, the matrices $\mathbf{F}$ and $\mathbf{G}$ are denoted with a star as in $\mathbf{F}_*$ and $\mathbf{G}_*$. This is to indicate that these entities are defined using the exact source posterior $\langle\cdot\rangle$ as opposed to the entities in Chapter 4 which are without stars indicating that these are computed with respect to the variational distribution $\langle\cdot\rangle_q$.

Appendix C

Publications

This appendix includes the published material of the PhD project.

C.1 ICA 2003: "Monaural ICA of White Noise Mixtures is Hard"

4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003), April 2003, Nara, Japan

MONAURAL ICA OF WHITE NOISE MIXTURES IS HARD

Lars Kai Hansen and Kaare Brandt Petersen

Informatics and Mathematical Modeling,
Technical University of Denmark,
DK-2800 Lyngby, DENMARK.
*e*mail: lkh,kbp@imm.dtu.dk

ABSTRACT

Separation of monaural linear mixtures of 'white' source signals is fundamentally ill-posed. In some situations it is not possible to find the mixing coefficients for the full 'blind' problem. If the mixing coefficients are known, the structure of the source prior distribution determines the source reconstruction error. If the prior is strongly multi-modal source reconstruction is possible with low error, while source signals from the typical 'long tailed' distributions used in many ICA settings can not be reconstructed. We provide a qualitative discussion of the limits of monaural blind separation of white noise signals and give a set of *no go* cases, finally, we use a so-called Mean Field approach to derive an algorithm for ICA of noisy monaural mixtures with a bi-modal source prior and demonstrate that low error source reconstructions are possible when the bi-modal source is close to binary. This is the first demonstration of blind source separation in noisy monaural mixtures without invoking temporal correlation information.

1. INTRODUCTION

Blind separation of linear signal mixtures is a fascinating problem with numerous applications. The over-complete separation problem consists in separating mixtures of more sources than measurements [8]. The extreme situation arises when the aim is to separate two or more signals mixed linearly into only one channel, and is referred to as *monaural* or single channel separation.

This work was supported by Oticon Fonden and EU project MAPAWAMO

Blind signal separation problem consist two sub-problems: 1) Learning the mixing coefficients from a sample of the mixture and 2) Estimating the sources. These two aspects of the problem are often considered together. In fact, in many EM-like algorithms one iterates between solving 1) and 2). But it turns out that in the monaural problem it is sometimes possible to learn the mixing coefficients without being able to fully recover the source signals.

A number of monaural separation schemes have been proposed, all based on the use of temporal structure [4, 13]. In this contribution we will focus on the harder problem of over-complete mixtures of sources with no temporal correlation as in [8].

We show that for source signals with densities belonging to the so-called symmetric stable family it is not possible to learn the mixing coefficients. For heavy tailed signals, in general, it is not possible to recover the sources even if the mixing coefficients are known.

Finally, we can provide a bit of good news: For certain multi-modal source priors (e.g., binary sources) one can in fact easily learn the mixing coefficients and subsequently recover the source signals. In fact we show how to use a Mean Field approach to derive an algorithm for ICA of noisy monaural mixtures with a bi-modal source prior and demonstrate that low error source reconstructions are possible when the bi-modal source is close to binary. This is the first demonstration of blind source separation in noisy monaural mixtures without invoking temporal correlation information. Separation of bi-modal sources is of practical relevance in telecommunications and in functional neuroimaging. In fMRI activation studies the observation is a mixture of hemodynamic fluctuations, including the response of neural tissue to on-off stimuli. Such stimulus induced components can have bi-modal histograms see [12].

2. LEARNING THE MIXING COEFFICIENTS

Let us consider monaural separation of white noise signals, i.e., signals without temporal dependencies and without measurement noise. In particular we study the model,

$$x^n = \mathbf{a} \cdot \mathbf{s}^n, \tag{1}$$

where $\mathbf{s}$ is a vector of D source signals $s_k, k = 1, ..., D$, while $\mathbf{a}$ is a vector of mixing coefficients, and $x^n, n = 1, ..., N$ is the univariate mixture. The sources are independent in time and space,

$$P(\mathbf{s}^1, \mathbf{s}^2, ..., \mathbf{s}^N) = \prod_{n=1}^{N} \prod_{k=1}^{D} P(s_k^n) \tag{2}$$

The blind source separation problem consist of estimating the mixing coefficients and the sources. The posterior distribution of mixing coefficients is given by

$$\begin{aligned} & P(\mathbf{a}|\{x^n\}) \propto P(\mathbf{a})P(\{x^n\}|\mathbf{a}) \\ & P(\{x^n\}|\mathbf{a}) = \\ & \prod_{n=1}^{N} \int D\mathbf{s}^n \delta\left(x^n - \mathbf{a} \cdot \mathbf{s}^n\right) \prod_{k=1}^{D} P(s_k^n) \end{aligned} \tag{3}$$

In general the likelihood function will be a function of the components of $\mathbf{a}$. If the source distributions are symmetric and identical, the posterior will ‘inherit’ reflection and permutation symmetries as in conventional ICA problems [7]. Hence, in the case of a one-dimensional mixture of two such sources there will be four degenerate ‘peaks’ in the likelihood (and in the posterior if the prior $P(\mathbf{a})$ is symmetric).

However, under certain source distributions the posterior can be more degenerate.

2.1. Symmetric stable signals

The class of symmetric α-stable ($S\alpha S$) distributions are heavy-tailed distributions investigated as models of impulsive noise and speech [5, 6]. The densities are denoted $S\alpha S(\mu, \alpha, \gamma)$ where μ is a location parameter, γ is a scale parameter and α is a shape parameter and they are defined by their closedness under addition

$$\begin{aligned} & s_1 \sim S\alpha S(\mu_1, \gamma_1 \alpha,) \wedge \ s_2 \sim S\alpha S(\mu_2, \alpha, \gamma_2) \\ & \Rightarrow \\ & s_1 + s_2 \sim S\alpha S\left(\mu_1 + \mu_2, \alpha, (|\gamma_1|^\alpha + |\gamma_2|^\alpha)^{1/\alpha}\right) \end{aligned}$$

The density functions for $S\alpha S$-variables can only be written in closed form for $\alpha = 2$ and $\alpha = 1$, where we obtain the normal distribution and the Cauchy distribution:

$$P(s|\gamma, \mu) = \frac{1}{\pi} \frac{\gamma}{\gamma^2 + (s - \mu)^2}. \tag{4}$$

The addition rule implies that information about the mixing coefficients is lost in stable mixtures: The likelihood function for a mixture of two unit-scale stable variables (with mixing coefficients a_1, a_2) is a function of the combination $(|a_1|^\alpha + |a_2|^\alpha)^{1/\alpha}$ only.

2.2. Temporally correlate d gaussian sources

In [1, 4, 13, 14] monaural separation was analyzed in the case of temporally correlated signals, typically speech signals. For temporally correlated and *stationary* signals the matrix

$$\mathbf{C}(\tau) = \langle \mathbf{s}(t)\mathbf{s}^\top(t + \tau)\rangle \tag{5}$$

is a diagonal matrix with elements describing the autocorrelation of the sources at lag τ. Now compute the autocorrelation function for the mixture signal

$$\begin{aligned} \gamma(\tau) &= \langle x(t)x(t + \tau)\rangle \\ &= \mathbf{a}^\top \mathbf{C}(\tau)\mathbf{a} \\ &= \sum_{k=1}^{D} (a_k)^2 C_{k,k}(\tau). \end{aligned} \tag{6}$$

We note that from a measurement of $\gamma(\tau)$ is it *not* possible to identify both $\mathbf{a}$ and the D source functions $C_{k,k}(\tau)$. Hence, we conclude that an ICA approach like the one of Molgedey and Schuster [9] is infeasible for monaural separation. Other schemes that either makes use of higher order moments or non-stationarity may still succeed in separating monaural mixtures. Since our main objective here is to discuss white noise mixtures, we will not pursue this topic further, instead we refer to [1, 4, 13, 14] for examples.

3. GEOMETRY OF MONAURAL WHITE NOISE SOURCE RECONSTRUCTION

The above degeneracy is specific for stable distributions and in general we expect that the mixing coefficients are learnable from samples of a monaural mixture. However, then the second problem arises: How accurate can the source signals be recovered?.

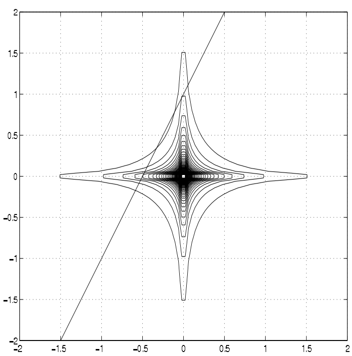


Fig. 1. Contours of the joint distribution of two sparse source signals. An observation $x = a_1 s_1 + a_2 s_2$ constrains the possible sources to a line. The 'true' source are drawn from the prior, hence, the line passes trough at least one arm of the cross formed by a contour line. In the two-dimensional case we see that a line determined by mixing coefficients in general position always intersects the cross in *two places*. The actual values of the mixing coefficients determine which of the two peaks in along the line is maximal, not the measurement x, hence, we conclude that any source reconstruction will have a large error for such sparse source mixing problems.

To illustrate the ill-posed nature of the source identification problem we first note that if the source are i.i.d. we are faced with a sequence of independent problems of the form: Find $\mathbf{s}$ from a single measurement of x, given $\mathbf{a}$. The constraint

$$x = \mathbf{a} \cdot \mathbf{s}, \tag{7}$$

defines a $D - 1$ dimensional hyperplane of solutions in the D-dimensional source space. The posterior distribution of the source signal is given by

$$P(\mathbf{s}|x, \mathbf{a}) = \frac{\delta\left(x - \mathbf{a} \cdot \mathbf{s}\right) P(\mathbf{s})}{P(x|\mathbf{a})} \tag{8}$$

Let us be specific and consider first two signals mixed in one receiving channel. In Figure 1 we show the geometry for a sparse (Cauchy) prior. The density contours form a 'cross' strongly peaked along the axes. For two independent sparse signals, there is low probability that they are both large at the same time. The constraint $x = a_1 s_1 + a_2 s_2$ forms a single linear relation between the component of $\mathbf{s}$, hence, the source prior determines which solution we choose, say in a maximum posteriory approach. Clearly, for a sparse prior we see that for all $\mathbf{a}$ there are two 'peaks', along the line. Since the source signals are drawn from the prior we face a situation where we need to choose between two competing solutions: The 'true' solution and a 'false' solution. Note, it is the coefficients $\mathbf{a}$ that determines which of the two peaks have the maximum density value, not x, hence we can expect a high reconstruction error rate for sparse priors.

Now why does the same argument not prevent the solution of the mixing of three signals in the binaural (two channels) case, as e.g., considered in [8, 7]? The scenario is depicted in Figure 2, the two measurements form two linear contraints leading to a 1D linear manifold of solutions within which the prior determines the posterior. Since the true sources are drawn from the prior, the line will typically intersect one of the six 'arms' of the density, but unlike the previous case, there is limited probability of a second "false" solution, corresponding to the line also intersecting another arm! It is easy to generalize this argument to higher dimensional mixtures in two channels. Source reconstruction with sparse sources is fundamentally ill-posed in one-dimensional mixtures and well-posed in typical two or higher dimensional channel case.

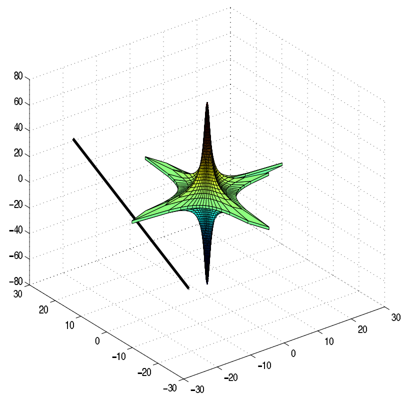


Fig. 2. Iso-surface for the joint prior distribution of three sparse sources. If we have access to two observation channels (microphones) each observation $x_j = a_{j1}s_1 + a_{j2}s_2 + a_{j3}s_3,\ j = 1, 2$ will constrain the possible sources to a plane, hence, the likelihood function contrains the posterior to the intersection of the two planes forming a linear manifold of possible sources. Since the 'true' sources are drawn from the prior the line will pass trough at least one arm of the star shaped iso-surface. However, for mixing coefficients in general position it is not likely that the line passes through two arms. Hence, the posterior is strongly peaked in the vicinity of the 'true' sources.

4. RECONSTRUCTION OF BI-MODAL SOURCE SIGNALS

The geometric origin of the sparse source reconstruction problem suggests that sources with a multi-modal prior will reconstruct better. Figure 3 illustrates the situation for a quasi-binary prior formed by a mixture of two gaussians. The prior has high density values in the neighborhood of the four source signal combinations $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$. For a typical set of mixing coefficients $\mathbf{a}$ the linear constraint imposed by the likelihood leads to a line of solutions intersecting one and only one of the four 'prior peaks', hence strongly selecting the correct solution.

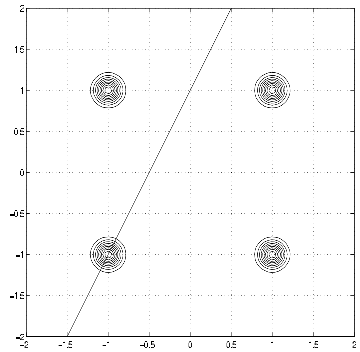


Fig. 3. Contours of the joint density of two source signals with a 'mixture of two gaussians' priors. The narrow widths create a 'quasi-binary' scenario. The observation of $x = a_1s_1 + a_2s_2$ again constrains the posterior to the intersection of the prior and the 'likelihood' line. The line only intersects one of the gaussians, hence the posterior is stongly peaked a the 'true' source configuration $(s_1, s_2) = (1, -1)$. If we observe a noisy mixture $x = a_1s_1 + a_2s_2 + \nu$, the line is broadened to to standard deviation of the noise process ν.

4.1. Mean Field approach for noisy ICA

Using the mean-field approach of [3] we have investigated the reconstruction of source signals from a *noisy* mixture $x = a_1s_1 + a_2s_2 + \nu$, where the sources both have a bi-modal prior density composed by a mixture of two gaussians as in Figure 3. The two gaussians are centered at $+/-1$ and have width σ^2

We assume that the additive noise is gaussian, $\nu \sim \mathcal{N}(0, \eta^2)$. The probability density of the measured data conditioned on the parameters, i.e., the likelihood function, is obtained by noting that $\nu^n = x^n - \mathbf{a}\cdot\mathbf{s}^n$, hence, $P(x^n|\mathbf{a},\mathbf{s}) = \mathcal{N}(\mathbf{a}\cdot\mathbf{s}^n, \eta^2)$. Bayes theorem then relates the source priors and the source posterior as

$$P(\mathbf{S}|\mathbf{a},\eta^2,\mathbf{x}) = \frac{P(\mathbf{x}|\mathbf{a},\eta^2,\mathbf{S})P(\mathbf{S})}{P(\mathbf{x}|\mathbf{a},\eta^2)}$$

where capital $\mathbf{S}$ is defined by $\mathbf{S}_i^n \equiv (\mathbf{s}^n)_i$. Maximizing the source posterior with respect to $\mathbf{a}$ and η^2 leads to the following estimators

$$\begin{aligned}\hat{\mathbf{a}} &= X\langle \mathbf{S}\rangle^T \langle \mathbf{S}\mathbf{S}^T\rangle^{-1} \\ \widehat{\eta^2} &= \frac{1}{N}\langle (x-\mathbf{a}\mathbf{S})(x-\mathbf{a}\mathbf{S})^T\rangle\end{aligned}$$

The bracket $< ... >$ denotes expectation with respect to the source posterior. These expressions depend on $\langle \mathbf{S}\rangle$ and $\langle \mathbf{S}\mathbf{S}^T\rangle$ approximately determined through Mean Field Theory (MFT) using $\mathbf{a}$ and η^2 [3]. In MFT the true source posterior is approximated by a factorized density function. The interaction not represented directly in the factorized model is compensated by a linear term - the so-called 'external field', in particular we introduce two auxiliary variables: The interaction $\mathbf{J} = \eta^{-2}\mathbf{a}^T\mathbf{a}$ and the external field $\mathbf{h} = \eta^{-2}\mathbf{a}^T\mathbf{x}$. From these we define:

$$\lambda_i^n = \lambda_i = J_{ii} \tag{9}$$

$$\gamma_i^n = (\mathbf{h} - (\mathbf{J} - \mathrm{diag}\mathbf{J})\langle \mathbf{s}\rangle)_i^n\,, \tag{10}$$

and the implicit non-linear equations for the sufficient statistics then read,

$$\begin{aligned}&\langle \mathbf{S}\rangle_{in} = \frac{1}{\lambda_i\sigma^2+1}\Big(\gamma_i^n\sigma^2 + \tanh\Big[\frac{\gamma_i^n}{\lambda_i^n\sigma^2+1}\Big]\Big) \\ &\langle \mathbf{S}\mathbf{S}^T\rangle_{ij} = \\ &\sum_n \Big[\frac{\delta_{ij}}{(\lambda_i\sigma^2+1)^2}\Big\{1+\sigma^2+\sigma^4(\lambda_i + (\gamma_i^n)^2) \\ &+2\gamma\sigma^2\tanh\Big(\frac{\gamma_i^n}{\lambda_i\sigma^2+1}\Big)\Big\} + (1-\delta_{ij})\langle \mathbf{S}\rangle_i^n\langle \mathbf{S}\rangle_j^n\Big]\end{aligned}$$

This way we arrive at a two-step algorithm. The first step in which $\mathbf{a}$ and η^2 are estimated using $\langle \mathbf{S}\rangle$ and $\langle \mathbf{S}\mathbf{S}^T\rangle$ and a second step in which $\langle \mathbf{S}\rangle$ and $\langle \mathbf{S}\mathbf{S}^T\rangle$ are estimated using $\mathbf{a}$ and η^2. We use the posterior means as our posterior source estimate, i.e. $\widehat{\mathbf{S}} = \langle \mathbf{S}\rangle$. In the limit $\sigma^2 \to 0$, i.e. the binary case, the posterior mean values minimize the mean bit error rate.

We set up a simulation to demonstrate that bi-modal monaural mixtures can be separated. The sources are bi-gaussian with a variable width, and furthermore we vary the additive noise level. The reconstruction error of the recovered source signals is measured as

$$E = \frac{1}{4\langle s_*^2\rangle}\langle (s_* - \widehat{s})^2\rangle \tag{11}$$

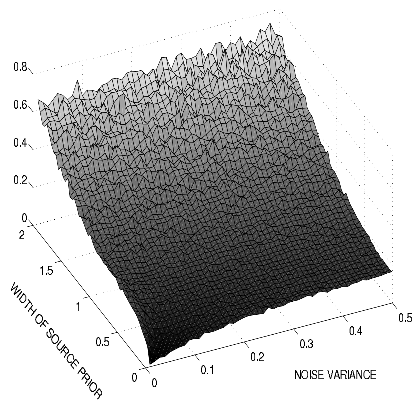


Fig. 4. The scaled mean square source reconstruction error vs. the width of the bi-gaussian source distributions and the noise variance. The gaussians were centered at $+/-1$ and the widths were identical and varied from zero (binary sources) to $\sigma = 2$. The mixing coefficients were $(a_1, a_2) = (\cos(\pi/8), sin(\pi/8))$. The zero error rate in the limit of low noise and narrow source priors is achieved for mixing coefficients in 'general position'. The sources were estimated using the mean field approach presented [3]

where $s_*, \widehat{s}$ are the 'true' and reconstructed sources respectively. The scaling assures that the error becomes equivalent to the *bit error rate* for binary sources. The reconstruction error is a function of the width of the source distribution and the noise variance as shown in Figure 4. For mixing coefficients in 'general position' we achieve perfect reconstruction in the binary, noise free limit.

5. CONCLUSION

In this paper we have discussed the problem of blind source separation based on a single white noise mixture signal. We have shown that in some situations it is impossible to identify the mixing coefficients, namely for α-stable mixtures. In general it is possible to identify the mixing coefficients, however, it may still be infeasible to reconstruct the source signals. A qualitative analysis lead us to conclude that heavy tail signals can only be reconstructed with a high error rate, hence, monaural mixtures are harder than two or more dimensional mixtures for simple topological reasons.

For signals with a multi-modal source distribution, however, source reconstruction is feasible and in the noise free limit, and in the specific case of binary sources the reconstruction can take place without errors.

Acknowledgment

We thank Preben Kidmose for discussions on stable signals, and the ISP-group at DTU for discussion of ICA problems.

6. REFERENCES

[1] R. Balan & and J.A. Rosca: *AR processes and sources can be reconstructed from degenerate mixtures*. Proceedings of the First International Conference on Independent Component Analysis and Blind Source Separation ICA'99, Aussois, France. pages 467-47 (1999).

[2] A. Bell & T.J. Sejnowski: *An Information-Maximization Approach to Blind Separation and Blind Deconvolution* Neural Computation **7**, 1129-59 (1995).

[3] P. Højen-Sørensen, O. Winther, and Lars Kai Hansen: Mean Field Approaches to Independent Component Analysis. Neural Computation **14** 889-918 (2002).

[4] G-J. Jang, T-W. Lee, Y-H. Oh: *Blind separation of single channel mixture using ICA basis functions*. 3rd International Conference on Independent Component Analysis and Blind Signal Separation. Dec. 9-12, 2001, San Diego, California, USA Eds: T-W. Lee et al. pp. 595-600, (2001).

[5] G.B. Georgiou, P. Tsakalides, and C. Kyriakis: *Alpha-stable modeling of noise and robust time-delay estimation in the presence of impulsive noise*. IEEE Transactions on Multimedia **1** 291-301 (1999).

[6] P. Kidmose: *Alpha-stable distributions in signal processing of audio signals*. In proceedings of the 41'st Conference on Simulation and Modelling, Scandinavian Simulation Society 87-94 (2000).

[7] T.-W. Lee *Independent Component Analysis: Theory and Applications*. Kluwer Academic Publ. Bostin (1998).

[8] M.S. Lewicki and T.J. Sejnowski: *Learning overcomplete representations* Neural Computation **12** 337-65 (2000).

[9] L. Molgedey & H. Schuster: *Separation of Independent Signals using Time-Delayed Correlations* Physical Review Letters **72** 3634-37 (1994).

[10] E. Moulines, J.-F. Cardoso, E. Gassiat. *Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models* Proc. ICASSP'97 Munich, vol. 5, pp. 3617-20 (1997).

[11] B.A. Olshausen: *Learning linear, sparse, factorial codes* A.I. Memo 1580, Massachusetts Institute of Technology (1996).

[12] K. Petersen, L.K. Hansen, T. Kolenda, E. Rostrup, S. Strother. *On the Independent Components of Functional Neuroimages*. In Proceedings of ICA-2000. Eds. P. Pajunen and J Karhunen, 615-620 (2000).

[13] S.T. Roweis: *One microphone source separation* in Advances in Neural Information Processing Systems **13** 793-799 (2001).

[14] E. Wan and A. Nelson: *Neural dual extended Kalman filtering: applications in speech enhancement and monaural blind signal separation* In Proceedings Neural Networks for Signal Processing Workshop. IEEE, 1997.

C.2 DSAGM 2004: "Using Mixture of Gaussians to Compare Approaches to Signal Separation"

Using Mixtures of Gaussians to Compare Approaches to Signal Separation

Kaare Brandt Petersen
Technical University of Denmark

1 Introduction

In the signal separation technique called Independent Component Analysis (ICA), one refers to a problem as being "square" if the there is as many measurements as sources to separate and "overcomplete" or "under-determined" if there is more sources than measurements. While square ICA is thoroughly investigated through many different approaches with well understood differences and similarities, the overcomplete case is still posing a difficult problem. Below is a short overview of some of the more interesting or illustrative of the approaches.

In 1999, Hagai Attias presented in [1] a Maximum Likelihood approach assuming the generative model $\mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon$. In this approach a model distribution is constructed and using Mixture of Gaussians as priors makes it possible to complete the relevant integrals and obtain a closed form expression for the distribution over $\mathbf{x}$. The model distribution is approximated to the data distribution through the Kullback Leibler divergence. This approach is extremely flexible while still having appealing analytical properties and the only real drawback is the bad scaling behavior: A sum over K^D must be computed, which can be rather larger for e.g. image data.

In 2000, Lewicki and Sejnowksi presented in [5] a Maximum Likelihood approach assuming the generative model $\mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon$. In this approach, the log likelihood of $\mathbf{A}$ is Taylor expanded to second order around the maximum a posteriori estimate of the sources, i.e. one approximate the likelihood with a gaussian, and the sources are in turn are estimated using the updated estimate of $\mathbf{A}$. In this approach we see the difficulty that most overcomplete techniques is trying to work around: The likelihood, or some other suitable cost-function, contains some integral involving the source prior and is therefore in general hard to solve. The approach of Lewicki and Sejnowski from 2000 substitutes the integral with a second order expansion and we shall see other possibilities in the following.

In 2001, Girolami presented in [2] a Maximum Likelihood approach assuming the generative model $\mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon$. In this approach, Girloami assumes Laplacian priors and the integral associated with the log likelihood is approximated through a variational scheme: The laplacian priors are reformulated in dual space which provides a lower bound of the likelihood to be optimized. The drawback of this approach is that the trick only works for laplacian priors and that the algorithm optimizes a lower boudn in stead of the log likelihood it self.

In 2002, Hojen-Soerensen, Winther and Hansen presented in [4], the so-called "Mean Field Approach" assuming the generative model $\mathbf{x} = \mathbf{As} + \epsilon$. The parameters $\mathbf{A}$ and possibly the noise covariance $\mathbf{W}$ are estimated through maximum likelihood assuming knowledge of the mean values of the sources and vice versa. That is, the integral of the log likelihood, translates into mean values of the sources, which are approximated with estimates of the mean values obtained from Mean Field theory. The nice feature of this approach is that we substitute a very complicated integral with an easier non-linear equation and that one can do this for any prior. The problem, of course is that al though Mean Field estimates are fairly accurate they are still approximations.

Also in 2002, Shriki, Sampolinski and Lee presented in [7] an interesting variant of Infomax on the filtering model $\hat{\mathbf{s}} = \mathbf{Wx}$. In the setup $\mathbf{y} = g(\mathbf{Wx})$, Shriki et. al. obtains a relation between $p(\mathbf{x})$ and $p(\mathbf{y})$ by assuming a noisy relation $\mathbf{y} = g(\mathbf{Wx}) + \epsilon$ and letting the noise go to zero. The main problem is that the limit is not taken properly care of and that the certainty of the result therefore is doubtful.

And finally in 2003, Teh, Welling, Osindero and Hinton presented in [8] the Energy Based Model on the filtering model $\hat{\mathbf{s}} = \mathbf{Wx}$. Through a setup using inspiration from physics, a model distribution for $\mathbf{x}$ is constructed and adjusted to be as close to the data distribution as possible through a Kullback Leibler divergence. Again the authors are faced with an intractable integral and this time it is approximated with the so-called "n-step Learning" which is a Hybrid Monte Carlo technique using n steps in the estimation of the integral. The remarkable claim of the paper is that very few steps such as $n = 1$, or $n = 3$ often will be sufficient to obtain overall convergence of the algorithm.

A common feature for most of the approaches is that they assume a noisy model and obtain a likelihood which involves a difficult integral which is then approximated in some way or another. The exception from this is the approach of Hagai Attias, but one can then argue that assuming priors to be mixtures of gaussians either is a restriction or an approximation.

1.1 This Paper

Thus, Independent Component Analysis (ICA) can be performed by a vast range of different methods. These can differ from each other by assumed properties such as noise or time-correlation, but also by the fundamental issue on whether they attempt to estimate the generative mixing matrix, denoted $\mathbf{A}$, or a filtering matrix denoted $\mathbf{W}$. In case of the same number of observations and sources, the square case, most if not all of these methods can be proven to be equivalent. But in the overcomplete case, where the number of observations are smaller than the number of sources, their differences becomes apparent and it is not easy to compare the results of generative and filtering approaches.

This paper makes an attempt to compare the result of two different methods in the overcomplete case: The Maximum Likelihood (ML), which estimates the generative $\mathbf{A}$, and the Energy Based Models (EBM) which seek to estimate a filtering matrix $\mathbf{W}$. This is done by assuming the priors to be centered mixtures of gaussians which makes it possible to compare the optimization schemes. This approach, with respect to ML, is closely related to the work of Hagai Attias in

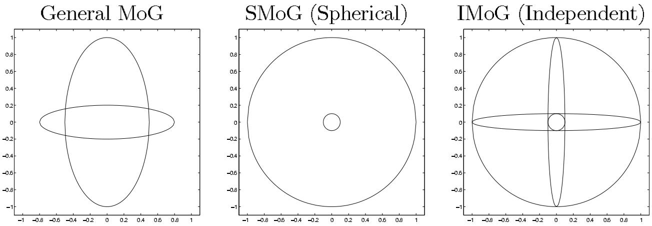


Figure 1: Contour plot of probability densities in 2D source space. and this is the only of the the three which in fact is independent in its variables, i.e. a Independent Mixture of Gaussians, IMoG.

[1], but where Attias is assuming completely general mixtures of gaussians and a noisy mixture model, the mixtures of gaussians in this paper are assumed centered for simplicity and, more importantly, the limit of zero noise is derived to be able to compare with the noiseless EBM.

The structure of the paper is as follows: In Sec 2, we introduce the reader to the mixture of gaussians to be used, and in Sec 3 we apply the mixtures of gaussians to the EBM and the ML. In Sec 4 we compare the results found and finally in Sec 5, we make a short summary.

2 Mixture of Gaussians

In this paper we make extensive use of the family of distributions known as mixtures of gaussians (MoG). To clear up some common misconceptions about MoG we introduce the general MoG and then discuss two important subsets of distributions. The density of a D-dimensional centered MoG is

$$p(\mathbf{s}) = \sum_{\kappa} \frac{\rho_\kappa}{\sqrt{|2\pi \mathbf{D}_\kappa|}} \exp\Big[-\frac{1}{2}\mathbf{s}^T \mathbf{D}_\kappa^{-1} \mathbf{s}\Big] \tag{1}$$

where the weights ρ_κ sum to one. The matrix $\mathbf{D}_\kappa$ can be any positive definite matrix, but in this context often assumed diagonal, $\mathbf{D}_\kappa = \mathrm{diag}(\sigma_{1\kappa}^2, \sigma_{2\kappa}^2, ..., \sigma_{D\kappa}^2)$. The marginal distribution for each s_i is itself a mixture of gaussians, $s_i \sim \sum_\kappa \rho_\kappa \mathcal{N}(0, \sigma_{i\kappa}^2)$, but note about the joint distribution that even if the coordinates s_i in each component are independent, i.e. if $\mathbf{D}_\kappa$ all are diagonal, the coordinates s_i are *not* in general independent in the bigger joint distribution $p(\mathbf{s})$.

One subset of interest is the MoG constructed as a product of D 1-dimensional mixtures of gaussians, and therefore independent by construction. Since the variables are independent we call this kind of MoG for Independent Mixtures of Gaussians (IMoG). Denoting the parameters of $i'th$ marginal by $\sum_\kappa \rho_{i\kappa}\mathcal{N}(0, \sigma_{i\kappa}^2)$, the density of the joint distribution is given by

$$p(\mathbf{s}) = \sum_{\{\mathbf{k}\}} \frac{\tilde{\rho}_{\mathbf{k}}}{\sqrt{|2\pi \mathbf{D}_{\mathbf{k}}|}} \exp\Big[-\frac{1}{2}\mathbf{s}^T \mathbf{D}_{\mathbf{k}}^{-1} \mathbf{s}\Big] \tag{2}$$

energy E is used in a Gibbs distribution, i.e.

$$p_{\mathbf{W}}(\mathbf{x}) = \frac{e^{-E(\mathbf{x};\mathbf{W})}}{Z(\mathbf{W})} = \frac{p_s(\mathbf{Wx})}{\int p_s(\mathbf{Wx})d\mathbf{x}} \tag{4}$$

which is chosen as our model distribution. In [8], Teh et. al. make no assumption on the prior, which makes the normalization part more difficult. To deal with this they use so-called n-step learning, a variant of a hybrid monte carlo approach, do approximately estimate the normalization part of the optimization. In this paper we in stead choose the prior to be a MoG, which enables us to calculate the integral of Eq. 4 and obtain an closed form expression for the model distribution. The result is

$$p_{\mathbf{W}}(\mathbf{x}) = \sum_{\kappa} \gamma_{\kappa} \frac{\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{W}^T\mathbf{D}_{\kappa}^{-1}\mathbf{W}\mathbf{x})}{\sqrt{|2\pi(\mathbf{W}^T\mathbf{D}_{\kappa}^{-1}\mathbf{W})^{-1}|}} \tag{5}$$

where γ_{κ} in general is dependent on $\mathbf{W}$ and $\mathbf{D}_{\kappa'}$ for all κ' through the following relation: Setting $\xi_{\kappa} = (|2\pi(\mathbf{W}^T\mathbf{D}_{\kappa}^{-1}\mathbf{W})^{-1}|/|2\pi\mathbf{D}_{\kappa}|)^{1/2}$, we can write $\gamma_{\kappa} = \rho_{\kappa}\xi_{\kappa}/\sum_{\kappa'}\rho_{\kappa'}\xi_{\kappa'}$. Note that in the square case $\xi_{\kappa} = 1$ and therefore $\gamma_{\kappa} = \rho_{\kappa}$ and in the case of SMoG priors we obtain $\xi_{\kappa} = (2\pi/\sigma_{\kappa}^2)^{(D-M)/2}$. The estimated optimal filtering matrix $\hat{\mathbf{W}}$ is determined by

$$\hat{\mathbf{W}} = \min_{\mathbf{W}}\Big[KL(p_{\mathbf{W}}||p_0)\Big]$$

in which the gradient of the KL-divergence can be calculated analytically when the prior is chosen to be mixtures of gaussians or some other analytically appealing distribution.

3.2 Maximum Likelihood

In the ML setup presented here we assume a generative model $\mathbf{X} = \mathbf{AS} + \mathbf{\Gamma}$, where we, in order to be able to deal with the overcomplete case, have added white gaussian noise, $\mathbf{\Gamma}$. In the end we let the noise variance go to zero to obtain the noiseless result.

Assuming white gaussian noise and the prior on the sources $p_s(\mathbf{s})$ to be a MoG with covariance matrices $\mathbf{D}_{\kappa}$ and weights ρ_{κ}, we can complete the integration and write the distribution over $\mathbf{x}$ as

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{s})p(\mathbf{s})d\mathbf{s} = \sum_{\kappa} \rho_{\kappa} \frac{\sqrt{|2\pi\mathbf{\Phi}_{\kappa}^{-1}|}\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{\Psi}_{\kappa}\mathbf{x})}{\sqrt{|2\pi\mathbf{\Sigma}|}\sqrt{|2\pi\mathbf{D}_{\kappa}|}}$$

where $\mathbf{\Sigma} = \sigma^2\mathbf{I}$ is the noise covariance matrix, $\mathbf{\Phi}_{\kappa} = \mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A} + \mathbf{D}_{\kappa}^{-1}$ and $\mathbf{\Psi}_{\kappa} = \mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\mathbf{A}\mathbf{\Phi}_{\kappa}^{-1}\mathbf{A}^T\mathbf{\Sigma}^{-1}$. We now want to consider the limit of $\sigma^2 \to 0$, but we need to do this with great care, since otherwise crucial details will vanish in the approximation. Using the Woodbury identity, singular value decomposition and some very good approximations (see the appendix for details), we obtain the following limits

$$\begin{aligned} \mathbf{\Psi}_{\kappa} &\to (\mathbf{A}\mathbf{D}_{\kappa}\mathbf{A}^T)^{-1} \\ \sqrt{|2\pi\mathbf{\Phi}_{\kappa}^{-1}|}/\sqrt{|2\pi\mathbf{\Sigma}|} &\to w_{\kappa}/\sqrt{|\mathbf{A}\mathbf{A}^T|} \end{aligned}$$

energy E is used in a Gibbs distribution, i.e.

$$p_{\mathbf{W}}(\mathbf{x}) = \frac{e^{-E(\mathbf{x};\mathbf{W})}}{Z(\mathbf{W})} = \frac{p_s(\mathbf{Wx})}{\int p_s(\mathbf{Wx})d\mathbf{x}} \tag{4}$$

which is chosen as our model distribution. In [8], Teh et. al. make no assumption on the prior, which makes the normalization part more difficult. To deal with this they use so-called n-step learning, a variant of a hybrid monte carlo approach, do approximately estimate the normalization part of the optimization. In this paper we in stead choose the prior to be a MoG, which enables us to calculate the integral of Eq. 4 and obtain an closed form expression for the model distribution. The result is

$$p_{\mathbf{W}}(\mathbf{x}) = \sum_{\kappa} \gamma_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{Wx})}{\sqrt{|2\pi(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|}} \tag{5}$$

where γ_κ in general is dependent on $\mathbf{W}$ and $\mathbf{D}_{\kappa'}$ for all κ' through the following relation: Setting $\xi_\kappa = (|2\pi(\mathbf{W}^T\mathbf{D}_\kappa^{-1}\mathbf{W})^{-1}|/|2\pi\mathbf{D}_\kappa|)^{1/2}$, we can write $\gamma_\kappa = \rho_\kappa\xi_\kappa/\sum_{\kappa'}\rho_{\kappa'}\xi_{\kappa'}$. Note that in the square case $\xi_\kappa = 1$ and therefore $\gamma_\kappa = \rho_\kappa$ and in the case of SMoG priors we obtain $\xi_\kappa = (2\pi/\sigma_\kappa^2)^{(D-M)/2}$. The estimated optimal filtering matrix $\hat{\mathbf{W}}$ is determined by

$$\hat{\mathbf{W}} = \min_{\mathbf{W}}\Big[KL(p_{\mathbf{W}}||p_0)\Big]$$

in which the gradient of the KL-divergence can be calculated analytically when the prior is chosen to be mixtures of gaussians or some other analytically appealing distribution.

3.2 Maximum Likelihood

In the ML setup presented here we assume a generative model $\mathbf{X} = \mathbf{AS} + \mathbf{\Gamma}$, where we, in order to be able to deal with the overcomplete case, have added white gaussian noise, $\mathbf{\Gamma}$. In the end we let the noise variance go to zero to obtain the noiseless result.

Assuming white gaussian noise and the prior on the sources $p_s(\mathbf{s})$ to be a MoG with covariance matrices $\mathbf{D}_\kappa$ and weights ρ_κ, we can complete the integration and write the distribution over $\mathbf{x}$ as

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{s})p(\mathbf{s})d\mathbf{s} = \sum_\kappa \rho_\kappa \frac{\sqrt{|2\pi\mathbf{\Phi}_\kappa^{-1}|}\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{\Psi}_\kappa\mathbf{x})}{\sqrt{|2\pi\mathbf{\Sigma}|}\sqrt{|2\pi\mathbf{D}_\kappa|}}$$

where $\mathbf{\Sigma} = \sigma^2\mathbf{I}$ is the noise covariance matrix, $\mathbf{\Phi}_\kappa = \mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A} + \mathbf{D}_\kappa^{-1}$ and $\mathbf{\Psi}_\kappa = \mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\mathbf{A}\mathbf{\Phi}_\kappa^{-1}\mathbf{A}^T\mathbf{\Sigma}^{-1}$. We now want to consider the limit of $\sigma^2 \to 0$, but we need to do this with great care, since otherwise crucial details will vanish in the approximation. Using the Woodbury identity, singular value decomposition and some very good approximations (see the appendix for details), we obtain the following limits

$$\begin{aligned} \mathbf{\Psi}_\kappa &\rightarrow (\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T)^{-1} \\ \sqrt{|2\pi\mathbf{\Phi}_\kappa^{-1}|}/\sqrt{|2\pi\mathbf{\Sigma}|} &\rightarrow w_\kappa/\sqrt{|\mathbf{A}\mathbf{A}^T|} \end{aligned}$$

Here w_κ is a constant with respect to σ^2, but depends on $\mathbf{D}_\kappa$ and on $\mathbf{A}$ through the unique orthogonal matrix $\mathbf{V}$. $\mathbf{V}$ which fulfills the eigen value equation $\mathbf{A}^T\mathbf{A}\mathbf{V} = \mathbf{V}\mathbf{\Lambda}$, such that the eigen values in $\mathbf{\Lambda}$ are decreasing in size down the diagonal. The expression for w_κ is $w_\kappa = \prod_{i=M+1}^{D}(2\pi/(\mathbf{V}^T\mathbf{D}_\kappa^{-1}\mathbf{V})_{ii})^{1/2}$. With this limit taken care of, we can write the maximum likelihood expression for $p(\mathbf{x})$ as

$$p_{\mathbf{A}}(\mathbf{x}) = \sum_\kappa \rho_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T(\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T)^{-1}\mathbf{x})}{\sqrt{|2\pi\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T|}} \tag{6}$$

where the weights somewhat surprisingly turns out to be same as those of the prior (see the appendix for details). The estimated generative mixing matrix $\hat{\mathbf{A}}$ is the matrix maximizing the log likelihood

$$\hat{\mathbf{A}} = \max_{\mathbf{A}}\Big[\ln P(\mathbf{X}|\mathbf{A})\Big]$$

Note that we have not estimated the sources in this process, only the generative mixing matrix.

4 Comparing EBM and ML

Now we compare the EBM and ML approaches derived in the previous section and discuss the significance of the differences and similarities. In the square case, we obtain total equivalence of all expressions setting $\mathbf{W} = \mathbf{A}^{-1}$ and thus not surprisingly we can conclude, as Teh et .al. does in [8], that the two approaches are equivalent when the number of observations equal the number of sources. Therefore the discussion and comparison in this sections is almost entirely concerned with the overcomplete case.

In the overcomplete case it is not obvious how one should compare results on the generative $\mathbf{A}$ and the filtering matrix $\mathbf{W}$. The filtering approach does not retrieve the original sources, since for any matrix $\mathbf{W}$ we have $\mathbf{W}\mathbf{A}_g \neq \mathbf{I}$ because of the dimensionality: It is impossible to construct D M-dimensional orthogonal matrices when $D > M$. And we cannot in general compare the filter matrix $\mathbf{W}$ with the pseudo-inverse of $\mathbf{A}$, since this is not the optimal solution in all cases [5]. But using MoG as prior, both the model distribution $p_{\mathbf{W}}(\mathbf{x})$ of the EBM and the loglikelihood $p_{\mathbf{A}}(\mathbf{x})$ of the ML becomes MoG's with parameters which must be estimated to fit a common data set $\mathbf{X}$. In fact we end up with two optimizations which look rather similar

$$0 = \frac{\partial}{\partial\mathbf{W}}\sum_{t=1}^{N}\ln p_{\mathbf{W}}(\mathbf{x}_t) \qquad 0 = \frac{\partial}{\partial\mathbf{A}}\sum_{t=1}^{N}\ln p_{\mathbf{A}}(\mathbf{x}_t)$$

The similarity is to some extend both genuine and deceptive: Both distributions are MoG, but the dependency of the weights and covariances on $\mathbf{W}$ and $\mathbf{A}$ are different. In this section we compare EBM and ML by comparing the covariances and weights of their MoG's.

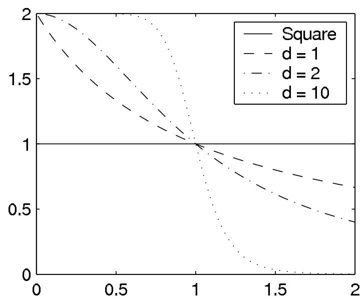


Figure 2: The Weights for EBM and ML. The plot demonstrates for SMoG priors that the differences in weights increase dramatically when the settings gets strongly overcomplete. (See the text for details on the plots)

4.1 Spherical MoG

We now consider the special case when the priors are SMoG, i.e. the variances are given as $\mathbf{D}_\kappa = \sigma_\kappa^2 \mathbf{I}$. In this case the sources are not assumed to be independent, which is interesting in its on right, but it also serves as a clear example of properties which holds for the more general cases as well. When we assume the priors to be SMoG, the model distribution simplifies significantly,

$$p_{\mathbf{W}}(\mathbf{x}) = \sum_\kappa \gamma_\kappa \frac{\exp(-\frac{1}{2}\mathbf{x}^T \mathbf{W}^T \mathbf{D}_\kappa^{-1} \mathbf{W} \mathbf{x})}{\sqrt{|2\pi (\mathbf{W}^T \mathbf{D}_\kappa^{-1} \mathbf{W})^{-1}|}}, \qquad \gamma_\kappa = \frac{\rho_\kappa (1/\sigma_\kappa^2)^d}{\sum_{\kappa'} \rho_{\kappa'} (1/\sigma_{\kappa'}^2)^d} \tag{7}$$

where $d = (D - M)/2$.

The Covariances

The covariances of EBM and ML respectively can in fact become equal in the case of SMoG. The equation setting the covariances equal

$$(\mathbf{W}^T \mathbf{D}_\kappa^{-1} \mathbf{W})^{-1} = \mathbf{A} \mathbf{D}_\kappa \mathbf{A}^T \qquad \forall \kappa \tag{8}$$

translates into $\mathbf{W}^T \mathbf{W} \mathbf{A} \mathbf{A}^T = \mathbf{I}$, which is fulfilled for $\mathbf{W} = \mathbf{A}^+$, where $\mathbf{A}^+$ denotes the pseudo inverse (Moore-Penrose) of the matrix $\mathbf{A}$. When $\mathbf{A}$ has full rank, the pseudo-inverse is given by $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$. But the equation is also fulfilled for any matrix $\mathbf{W} = \mathbf{U}\mathbf{A}^+$, where $\mathbf{U}$ is orthogonal and thus, there is an entire family of matrices which would make the covariances of the EBM equal to the covariances of the ML for a given $\mathbf{A}$. Conversely for any $\mathbf{W}$ we can choose $\mathbf{A} = \mathbf{W}^+$ to obtain the same covariances and and in this sense the two approaches have equal flexibility with respect to adjusting the covariances to the data. This is evident in the 4×2 example in Fig 3 a) and b).

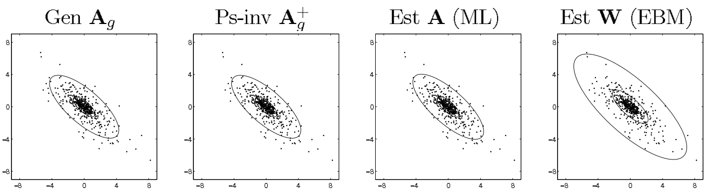


Figure 3: The covariances in case of SMoG priors for different settings. Plot a) Generative **A**. Plot b) Pseudo-inverse of the generative **A**. Plot c) Estimated **A**. Plot d) Estimated **W**.

The Weights

The weights can be compared by examining the ratio $\gamma_\kappa/\rho_\kappa$ and as we shall see, this ratio differs strongly from 1 in most cases. From the expression of γ_κ in Eq. 7, we see that the constrain making the weights of ML and EBM equal is

$$\frac{(1/\sigma_\kappa^2)^d}{\sum_{\kappa'} \rho_{\kappa'}(1/\sigma_{\kappa'}^2)^d} \stackrel{?}{=} 1 \qquad \forall \kappa$$

which is clearly impossible when the weights σ_κ^2 must be different for different κ. Thus, in the SMoG case, the weights of EBM and ML cannot be equal and furthermore the ratio $\gamma_\kappa/\rho_\kappa$ becomes relatively large for those κ where σ_κ^2 is very small and vice versa.

Fig 2 is a general illustration of this. Here we assume a SMoG prior with two components which has $\sigma_1^2 = 1$ and $\rho_1 = \rho_2 = 0.5$ and let σ_2^2 vary between 0 and 2. The resulting ratio γ_2/ρ_2 is shown in Fig 2: The x-axis is σ_2^2, the y-axis is γ_2/ρ_2 and the 4 curves are plottet for different degrees of overcompleteness d=0 (Square) and $d = 1, 2, 10$ (Overcomplete). Clearly only in the square case and for σ_2^2 values close to 1, is the ratio reasonably close to 1.

Another more specific example is shown in Fig 3 c) and d), which is the estimated covariances for ML and EBM in a 4×2 case. In this example the effect of the enhanced weight on smaller covariances is clear: When the weights of the smaller covariance is strong, the points far from origin is considered extreme, and the covariances are expanded accordingly. Thus, for this reason, EBM seem to favor larger covariances compared to ML.

5 Summary and Acknowledgements

Conclusively the use of Mixtures of Gaussians made it possible to compare Maximum Likelihood with Energy Based Models. The results show that in the overcomplete case with Spherical Mixtures of Gaussians as priors, the Energy Based Model is biased toward larger covariances compared to the Maximum Likelihood. One can show that this effect is also present for IMoG priors, though not at all as strong.

Finally I need to give due credit: This paper is closely related to the result of earlier work together with Jiucang Hao and Te-Won Lee from University of California San Diego (UCSD).

A Details of the Calculations

When calculating the limit of $\mathbf{\Psi}_\kappa$, we first set in the definition $\mathbf{\Sigma} = \sigma^2\mathbf{I}$ and simplify the expression into

$$\mathbf{\Psi}_\kappa = \frac{1}{\sigma^2}\left\{\mathbf{I} - \mathbf{A}(\mathbf{A}^T\mathbf{A} + \sigma^2\mathbf{D}_\kappa^{-1})\mathbf{A}^T\right\}$$

Now defining $\mathbf{Q} = \mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T/\sigma^2$ we can use Woodburys identity for inverse matrices and obtain

$$\mathbf{A}(\mathbf{A}^T\mathbf{A} + \sigma^2\mathbf{D}_\kappa^{-1})\mathbf{A}^T = \mathbf{Q} - \mathbf{Q}(\mathbf{I} + \mathbf{Q})^{-1}\mathbf{Q}$$

Since $\mathbf{Q}$ is symmetric and very large compared to $\mathbf{I}$, the right hand side can be approximated by $\mathbf{I} - \mathbf{Q}^{-1}$. To see this, write $\mathbf{Q}$ as $\mathbf{Q} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, for an orthogonal $\mathbf{V}$ and diagonal $\mathbf{\Lambda}$ and remember the identity $x - x^2/(1+x) = 1 - 1/(1+x)$ to obtain $\mathbf{I} - \mathbf{V}\mathrm{diag}(1/(1 + \Lambda_{ii}/\sigma^2))\mathbf{V}^T \cong \mathbf{I} - \mathbf{V}\mathrm{diag}(1/(\Lambda_{ii}/\sigma^2))\mathbf{V}^T = \mathbf{I} - \mathbf{Q}^{-1}$. Inserting this into the equation containing $\mathbf{\Psi}_\kappa$, gives the desired result.

When calculating the limit of the fraction containing the determinant $|2\pi\mathbf{\Phi}_\kappa^{-1}|$, we use the fact that since $\mathbf{A}^T\mathbf{A}$ is symmetric there exists an orthogonal $\mathbf{V}$ such that $\mathbf{A}^T\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. Since $|\mathbf{V}| = 1$ we get

$$|2\pi\mathbf{\Phi}_\kappa^{-1}|/|2\pi\mathbf{\Sigma}| = (2\pi\sigma^2)^{(D-M)}/|\mathbf{\Lambda} + \sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V}|$$

Since σ^2 are assumed arbitrary small, only the diagonal of the sum of matrices will contributed significantly to the determinant. And since further more $\mathbf{A}^T\mathbf{A}$ has rank M, the M first elements of the diagonal matrix will dominate together with the remaining $M - D$ factors

$$|\mathbf{\Lambda} + \sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V}| \cong \prod_{i=1}^{M}\mathbf{\Lambda}_{ii}\prod_{j=M+1}^{D}(\sigma^2\mathbf{V}^T\mathbf{D}^{-1}\mathbf{V})_{jj}$$

inserting this into the fraction above gives the desired result.

The coefficients α_κ has the structure $\rho_\kappa w_\kappa\sqrt{|2\pi\mathbf{A}\mathbf{D}_\kappa\mathbf{A}^T|}/\sqrt{|\mathbf{A}\mathbf{A}^T|\cdot|2\pi\mathbf{D}_\kappa|}$. In the square case much of the difficulties of taking the noiseless limit disappears, $w_\kappa = 1$ and we easily obtain $\alpha_\kappa = \rho_\kappa$. In the case of SMoG, setting $\mathbf{D}_\kappa = \sigma_\kappa^2\mathbf{I}$, we obtain $w_\kappa = (2\pi\sigma_\kappa^2)^{(D-M)/2}$ and therefore $\alpha_\kappa = \rho_\kappa$. Supported by numerical results we conjecture that this is also the case for the general overcomplete case.

References

[1] H. Attias, *Independent Factor Analysis*, Neural Computation 11, 803-851, 1999.

[2] M. Girolami, *A Variational Method for Learning Sparse and Overcomplete Representations*, Neural Computation, 13(11), pp 2517 - 2532, 2001.

[3] G. E. Hinton, *Training products of Experts by minimizing contrastive divergence*, Neural Computation, 14(8):1771-1800, 2002.

[4] P. Hojen-Sorensen, O. Winther, L. K. Hansen, *Mean-Field Approaches to Independent Component Analysis*, Neural Computation Volume 14, Issue 4, April 2002.

[5] M. S. Lewicki, T. J. Sejnowski, *Learning Overcomplete Representations*, Neural Computation, 12(2):337-65, 2000.

[6] R. M. Neal, *Probabilistic Inference Using Markov Chain Monte Carlo Methods*, Techinal Report CRG-TR-93-I, Department of Computer Science, ujniversity of Toronto, 1993.

[7] O. Shriki, H. Sampolinski, D.D. Lee, *An information maximization approach to overcomplete and recurrent representations* Neural Info. Proc. Sys. 13 (2001).

[8] Y. W. Teh, M. Welling, S. Osindero, G. Hinton *"Energy-Based Models for Sparse Overcomplete Representations"*, The Journal of Machine Learning Research - Special Issue on Independent Components Analysis, guest edited by Te-Won Lee, Jean-Franois Cardoso, Erkki Oja and Shun-ichi Amari. 2003.

C.3 ICASSP 2005: "The EM Algorithm in Independent Component Analysis"

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

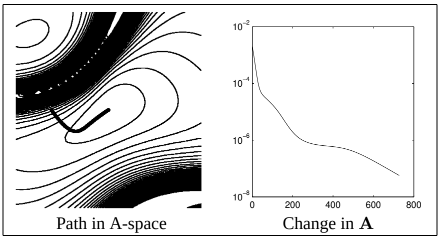


Fig. 1. MFICA: An example of EM slowdown. Left: The (projected) path of EM in A-space. Right: The change of $\mathbf{A}$ over iterations.

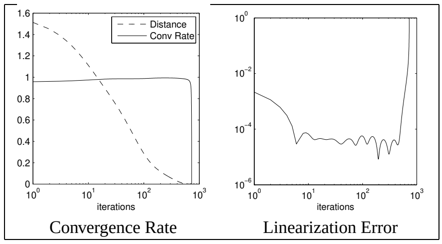


Fig. 2. MFICA: An example of EM slowdown. Left: The convergence rate. Right: The error induced when applying the approximation of Eq. 1

.

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

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

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.

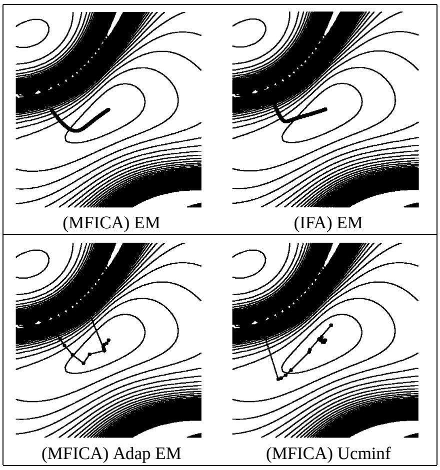


Fig. 3. The (projected) path in A-space for different optimization techniques on the example of Sec. 3.

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

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

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.

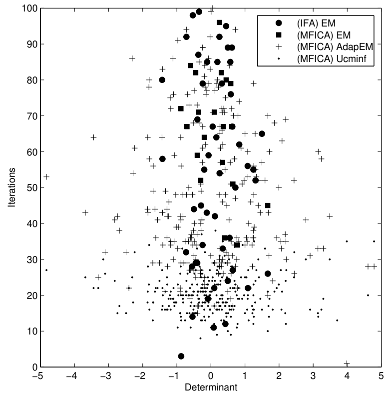


Fig. 4. The influence on the iterations of the determinant for 2x2 mixing matrices. On the first axis is $\det(\mathbf{A})$ and on the second axis is number of iterations for EM (MFICA).

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.

C.4 IMM-TechRep 2005: "Explaining slow convergence of EM in low noise linear mixtures"

TECHNICAL REPORT

Explaining slow convergence of EM in low noise linear mixtures

Kaare Brandt Petersen and Ole Winther
Informatics and Mathematical Modelling
Technical University of Denmark

January 2005

Contents

1 Introduction

This report conducts an investigation of the convergence properties of the EM algorithm used for linear mixture models. Since the linear mixture model is a rather general approach, the analysis is relevant for a wide range of models which to some degree are subsets of each other: Independent Component Analysis (ICA), probabilistic PCA, Factor Analysis (FA), Independent Factor Analysis (IFA) and Mean Field ICA.

1.1 Timeline on ICA Using EM

In order to present an overview of the results of using the EM algorithm for ICA and related models, is here a short list of selected set of papers each described in a few words.

- 1994 Belouchrani et al. [3]: Seemingly the first paper to suggest the EM algorithm for an ICA problem. They estimate the mixing matrix and the noise covariance. The sources here are assumed discrete QAM4 symbols and the integrals are possible to complete due to the discreteness of the signals.
- 1997 Moulines et al. [8]: Introduction of the EM algorithm for noisy convolutive mixtures with a focus on the instantaneous case. They use MoG as priors to avoid the intractable integrals in the posterior average.
- 1999 Bermond et al. [4]: Update of a square mixing matrix is Taylor expanded in the noise variance. Doing this, they are able to demonstrate several points: (i) In the low noise limit, EM is freezing in the sense that $\mathbf{A}_{n+1} = \mathbf{A}_n$. (ii) To first order in the noise variance, there is no correction for noise compared to Bell and Sejnowski (BS) [2]. That is, the noise model is not making a better estimate than BS when σ^4 is negligible.
- 1999 Attias [1]: A noisy instantaneous mixture model is solved with an EM algorithm using MoG priors. Compared to previous publications he is considering also source reconstruction and situation of having many sources.
- 2000 Lappalainen [7]: In the square case compensating for the slowdown proved earlier by a Taylor expansion in the noise variance, by removal of the "source parts" of the update. They also demonstrates that "Fast ICA" is in fact a special case of this speedup technique.
- 2001 Welling et al. [11]: A so-called constrained EM, where A is a scaling of a rotation, $\mathbf{A} = \alpha\mathbf{R}$. Priors are MoG.
- 2002 Højen-Sørensen et al. [6]: As Moulines an EM algorithm for ICA, but here the major contribution is the use of mean field theory for the posterior means which makes other priors than MoG possible.

- 2002 Deligne et al. [5]: A convolutive ICA using EM. Problems with convergence has been addressed through special initial conditions. MoG Priors.
- 2003 Salakhudinov et al. [10]: The adaptive over-relaxed approach to the problem of slow convergence. Not specific for the ICA using EM, but indeed applicable.
- 2005 Petersen et al. [9]: Small statistical Investigation for MF-ICA. In this they demonstrate the benefit of using the Adaptive EM and a BFGS method.

1.2 The contents of this report

The above list illustrates that many aspects have already been investigated. In the context of this report, especially the paper of Bermond et. al. [4] is central since it has presented many of the central results also to be presented in this report. The reason is that this report is documenting an independent rediscovery of the results. The contribution of this report is

- A detailed derivation.
- A different investigation of the overdetermined case.
- Insight into why Adaptive Overrelaxed EM works.

The structure of the report is as follows: In Sec 2 is described the observation model and the basics of the EM algorithm. Sec 3 is a derivation of the saddle point approximation, while Sec 4 is analyzing the EM algorithm using the saddle point approximation. Finally the conclusions are gathered in sec 5 and computational details are in the appendix.

2 Model and Estimation

2.1 Observation Model

The models under consideration are the linear mixture models, i.e. models of the type

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \boldsymbol{\eta}_t, \qquad t = 1, ..., N$$

where the sources are distributed according to some prior and the noise is zero-mean gaussian, $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$. The observation vectors $\mathbf{x}_t$ can be collected into a larger matrix $\mathbf{X}$ as columns, such that the dimensions of $\mathbf{X}$ is $M \times N$. Correspondingly we can collect all the source vectors into a larger source matrix $\mathbf{S}$ and write the mixing process including all time steps as $\mathbf{X} = \mathbf{AS} + \boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is the matrix of the noise. In this more compact notation, $\mathbf{X}$ conveniently denotes the entire data set and we only use the vector notation when ever it is needed to denote a single observation vector. From the assumption of gaussian noise we get a gaussian observation model

$$p(\mathbf{X}|\mathbf{S}) = \frac{1}{\sqrt{|2\pi\mathbf{\Sigma}|}^N} \exp\Big[-\frac{1}{2}\mathrm{Tr}[(\mathbf{X}-\mathbf{AS})^T\mathbf{\Sigma}^{-1}(\mathbf{X}-\mathbf{AS})]\Big]$$

in which the product over all time steps has been expressed as a trace in the exponential function.

2.2 Maximum Likelihood using the EM algorithm

In estimating the parameters $\mathbf{A}$ and $\mathbf{\Sigma}$, a maximum likelihood using the EM algorithm is used. By rewriting $p(\mathbf{X})$ using the hidden variables $\mathbf{S}$ it becomes apparent that we at least in principle have everything we need, knowing the observation model and the source priors. The parameters are estimated as the optimal points of the log likelihood

$$\mathbf{0} = \frac{\partial \ln p(\mathbf{X})}{\partial \mathbf{A}} \qquad \mathbf{0} = \frac{\partial \ln p(\mathbf{X})}{\partial \mathbf{\Sigma}}$$

These equations have the solution

$$\mathbf{A} = \mathbf{X}\langle\mathbf{S}^T\rangle\langle\mathbf{SS}^T\rangle^{-1} \qquad \mathbf{\Sigma} = \langle(\mathbf{X}-\mathbf{AS})(\mathbf{X}-\mathbf{AS})^T\rangle/m$$

where $\langle\cdot\rangle$ denotes average with respect to the source posterior $p(\mathbf{S}|\mathbf{X})$. In cases where $\mathbf{\Sigma}$ is further constrained in some way, it is straight forward to compute the corresponding expression. If we assume that we can compute the first and second moment of the sources with resepct to the source posterior, we now have the two basic steps which is put in to the EM algorithm:

$$\textbf{E-step}: \qquad \langle\mathbf{S}\rangle = \int \mathbf{S}p(\mathbf{S}|\mathbf{X})d\mathbf{S} \qquad \langle\mathbf{SS}^T\rangle = \int \mathbf{SS}^T p(\mathbf{S}|\mathbf{X})d\mathbf{S}$$

$$\textbf{M-step}: \qquad \mathbf{A} = \mathbf{X}\langle\mathbf{S}^T\rangle\langle\mathbf{SS}^T\rangle^{-1} \qquad \mathbf{\Sigma} = \langle(\mathbf{X}-\mathbf{AS})(\mathbf{X}-\mathbf{AS})^T\rangle/m$$

Starting from some reasonable values, we iteratively update our estimate of the source posterior moments and the model parameters, and we are guaranteed in each step an increase of the likelihood.

The source posterior moments are in general not easy to compute. If the source priors are gaussians (FA), mixture of gaussians (IFA) or some other well-chosen family of distribution, then we can compute the moments exactly, but otherwise we must resort to approximations as in MF-ICA, which in many cases are sufficiently accurate to produce good results.

3 Saddle Point Approximation

In this section, as a general approach to the difficulty of computing the source posterior moments, we make an approximation of the integral involved in the posterior and obtain an expansion of the moments in orders of the noise. In this way we are able to study some general properties of the algorithm in the low noise limit.

For simplicity of the equations, we assume for the rest of the paper that the noise is isotropic, $\mathbf{\Sigma} = \sigma^2\mathbf{I}$. In our experience this does not change any of the results obtained. Furthermore, since the time steps are assumed independent and the log likelihood therefore a sum over over time, we need in this section only to consider one single time step in the saddle point approximation.

3.1 Approximation of an Integral

The technique applied here is the so-called saddle-point approximation of the integral, which is well-known in statistical physics and in large parts of applied mathematics. To lowest order, it is basically an approximation of the integral $\mathcal{I} = \int \exp[-f(s)]ds$ which for very large values of $f(s)$ and if $f(s)$ is uni-modal, becomes dominated by its minimum $f(s_0)$. The point s_0 in which the function has its minimum is called the saddle-point, and the approximation is to lowest order $\mathcal{I} \cong \exp[-f(s_0)]$. The application of the technique in this paper is slightly more complicated since we have more dimensions, are expanding to higher order, and considering the low noise limit, but the basic idea is the same. Inspired by the equation $\ln p(\mathbf{x}) = \ln \int e^{\ln p(\mathbf{x},\mathbf{s})}d\mathbf{s}$, we define

$$g(\mathbf{s},\mathbf{h}) = -\sigma^2 \left(\ln p(\mathbf{x}|\mathbf{s}) + \ln p(\mathbf{s}) + \mathbf{h}^T\mathbf{s}\right)$$

Note that for $\mathbf{h} = \mathbf{0}$ the function g is a scaled version of the joint distribution $\ln p(\mathbf{x},\mathbf{s})$. The somewhat artificial variable $\mathbf{h}$ is a utility variable which will help us obtain the posterior moments as we shall se shortly. Defining the function $g(\mathbf{s},\mathbf{h})$ as above, we get that

$$\ln \mathcal{I} = \ln \int e^{-\frac{1}{\sigma^2}g(\mathbf{s},\mathbf{h})}d\mathbf{s}$$

is a moment generating function in the sense

$$\langle \mathbf{s} \rangle = \frac{\partial \ln \mathcal{I}}{\partial \mathbf{h}}\Big|_{\mathbf{h}=\mathbf{0}} \qquad \langle \mathbf{s}\mathbf{s}^T \rangle - \langle \mathbf{s} \rangle\langle \mathbf{s} \rangle^T = \frac{\partial^2 \ln \mathcal{I}}{\partial \mathbf{h}\partial \mathbf{h}^T}\Big|_{\mathbf{h}=\mathbf{0}}$$

where $\langle \cdot \rangle$ denotes average with respect to the source posterior $p(\mathbf{s}|\mathbf{x})$. For small σ^2 we can approximate $\ln \mathcal{I}$ by a Taylor expansion of $g(\mathbf{s},\mathbf{h})$ in the saddle point $\mathbf{s}_0$ defined by $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$. From this we obtain the saddle point approximation to the integral in $\ln \mathcal{I}$

$$\begin{aligned}
\ln \mathcal{I} &= \ln \int e^{-\frac{1}{\sigma^2}g(\mathbf{s},\mathbf{h})}d\mathbf{s} \\
&= \ln \int e^{-\frac{1}{\sigma^2}g(\mathbf{s}_0,\mathbf{h})-\frac{1}{2}\frac{1}{\sigma^2}(\mathbf{s}-\mathbf{s}_0)^T\mathbf{g}_0''(\mathbf{s}-\mathbf{s}_0)}d\mathbf{s} + \mathcal{O}(\sigma^2) \\
&= -\frac{1}{\sigma^2}g(\mathbf{s}_0) + \ln\sqrt{\det[2\pi\sigma^2(\mathbf{g}_0'')^{-1}]} + \mathcal{O}(\sigma^2) \qquad (1)
\end{aligned}$$

In the limit of small σ^2, the derivative of the second term is scaling as $\mathcal{O}(\sigma^4)$ and thus the approximation is efficiently dominated by the first term $-g(\mathbf{s}_0)/\sigma^2$.

3.2 Finding the Saddle Point

In the above we have used the saddle point without explicitly knowing its value. To find it, we make use of the definition $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$ but also an approximation of the saddle point in the zero-noise regime, i.e. we write $\mathbf{s}_0$ as a Taylor expansion in σ^2

$$\mathbf{s}_0 = \hat{\mathbf{s}}_0 + \sigma^2 \tilde{\mathbf{s}}_0 + \mathcal{O}(\sigma^4)$$

The zeroth order term $\hat{\mathbf{s}}_0$ is the least squares solution

$$\hat{\mathbf{s}}_0 = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x}$$

which simplifies further when $\mathbf{A}$ is square. We now want an expression for the first order term $\tilde{\mathbf{s}}_0$ and to find this we insert the combined expression for $\mathbf{s}_0$ into the equation $\mathbf{g}'(\mathbf{s}_0) = \mathbf{0}$ and rearrange the terms to first order in σ^2. Doing that we obtain

$$\tilde{\mathbf{s}}_0 = (\mathbf{A}^T\mathbf{A})^{-1}\left[\frac{\mathbf{p}'(\hat{\mathbf{s}}_0)}{p(\hat{\mathbf{s}}_0)} + \mathbf{h}\right]$$

Having an expression for the saddle point orders of σ^2, makes it possible for us to use the approximation of the integral to obtain the expressions for the posterior moments.

3.3 The Posterior Moments

Inserting into Eq. 1 and performing the calculations, we get

$$\langle \mathbf{s} \rangle = \left.\frac{\partial \ln \mathcal{I}}{\partial \mathbf{h}}\right|_{\mathbf{h}=\mathbf{0}} = \hat{\mathbf{s}}_0 + \sigma^2 \tilde{\mathbf{s}}_0 + \mathcal{O}(\sigma^4) \tag{2}$$

$$\langle \mathbf{s}\mathbf{s}^T \rangle - \langle \mathbf{s} \rangle \langle \mathbf{s} \rangle^T = \left.\frac{\partial^2 \ln \mathcal{I}}{\partial \mathbf{h} \partial \mathbf{h}^T}\right|_{\mathbf{h}=\mathbf{0}} = \sigma^2 (\mathbf{A}^T\mathbf{A})^{-1} \tag{3}$$

Note that the posterior mean is the saddle point. This is due to the fact that a minimization of $g(\mathbf{s}, \mathbf{h})$ corresponds to a maximization of $\ln[p(\mathbf{x}, \mathbf{s})]$ which through Bayes theorem has the same maximum as the posterior $p(\mathbf{s}|\mathbf{x})$. In the approximation of the integral we have implicitly assumed that the posterior is uni-modal and symmetric and therefore is the mean the same as the most probable value. The variance provide us with the second moment through

$$\langle \mathbf{s}\mathbf{s}^T \rangle = \hat{\mathbf{s}}\hat{\mathbf{s}}^T + \sigma^2 \mathbf{B}_t + \mathcal{O}(\sigma^4)$$

$$\mathbf{B}_t = (\mathbf{A}^T\mathbf{A})^{-1} + (\mathbf{A}^T\mathbf{A})^{-1}\frac{\mathbf{p}'(\hat{\mathbf{s}})}{p(\hat{\mathbf{s}})}\hat{\mathbf{s}}^T + \hat{\mathbf{s}}\frac{\mathbf{p}'(\hat{\mathbf{s}})}{p(\hat{\mathbf{s}})}^T (\mathbf{A}^T\mathbf{A})^{-1}$$

To summarize we now have approximations for the posterior moments in the low noise regime. In order to investigate the quality of this approximation, we have computed the exact and approximated moments for a mixture of gaussians. The result is plotted in Fig. 1. As expected is the approximation not good for large noise variance, but for noise variance in the area of 10^{-2} to 10^{-3} and smaller, the saddle point approximation is very accurate.

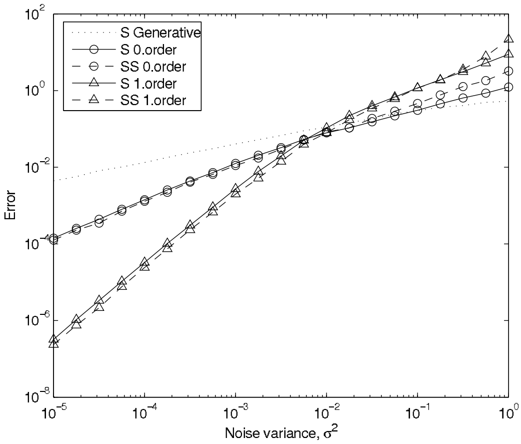


Figure 1: Approximation of the moments - how accurate is it? This plot shows how accurate the saddle point approximation is for different noise levels. The error is defined as the squared mean difference between the approximation and the true posterior mean. For perspective, we have included the generating sequence as well. The prior chosen here is a mixture of two zero-mean gaussians with variance 1 and 1/100.

3.4 The EM Updates

Combining the posterior moments for each time step into larger matrices for the entire data set, we get

$$\langle \mathbf{S} \rangle = \hat{\mathbf{S}} + \sigma^2 \tilde{\mathbf{S}} + \mathcal{O}(\sigma^4) \quad (4)$$
$$\langle \mathbf{S}\mathbf{S}^T \rangle = \hat{\mathbf{S}}\hat{\mathbf{S}}^T + \sigma^2 \mathbf{B} + \mathcal{O}(\sigma^4) \quad (5)$$

where $\mathbf{B} = \sum_t \mathbf{B}_t$. From this we can directly insert into Eq. XXX and obtain the EM updates of $\mathbf{A}$ and σ^2

$$\mathbf{A}_{n+1} = \mathbf{X}\hat{\mathbf{S}}(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1} + \sigma^2 \mathbf{X}\mathbf{F}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1} + \mathcal{O}(\sigma^4) \quad (6)$$
$$\sigma^2_{n+1} = \sigma^2_{bias} + \sigma^2\Big(\frac{\mathrm{rank}(\mathbf{A})}{m} - \frac{2}{m}\mathrm{Tr}(\mathbf{U})\Big) + \mathcal{O}(\sigma^4) \quad (7)$$

where $\mathbf{F} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{p}'(\hat{\mathbf{S}})/p(\hat{\mathbf{S}}) - \hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}\mathbf{B}$ and σ^2_{bias}, $\mathbf{U}$ will be explained shortly. These equations are general in the sense that they apply for both square and overdetermined mixing matrix and they simplify considerably in the square case which we analyze in the subsection below.

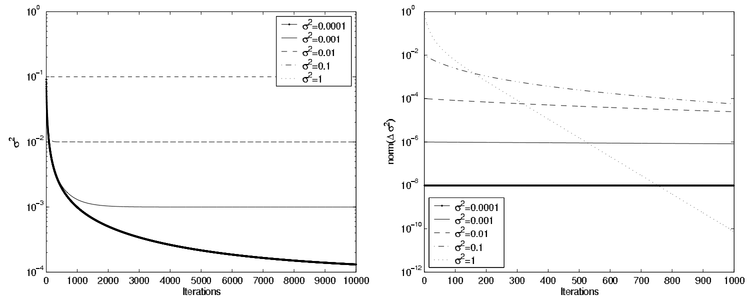


Figure 2: Convergence of the noise for a square FA model: The left plot shows how the noise converges to the levels of the generative model and that lower noise result in slower convergence. The right plot shows that the change of noise variance $\Delta\sigma^2$ is indeed proportional to σ^4. The drop in $\Delta\sigma^2$ seen for $\sigma^2 = 1$ is due to the fact that the model is converged from a very early stage.

4 Analysis

4.1 The Square Case

In the square case we have $\sigma^2_{bias} = 0$ and $\mathbf{U} = \mathbf{0}$, and the expressions simplify into

$$\begin{aligned}\mathbf{A}_{n+1} &= \mathbf{A}_n + \sigma^2 \tilde{\mathbf{A}}_n + \mathcal{O}(\sigma^4) \\ \sigma^2_{n+1} &= \sigma^2_n + \mathcal{O}(\sigma^4)\end{aligned}$$

$$\tilde{\mathbf{A}}_n = \mathbf{A}^{-T}\left(\mathbf{A}^T + \tfrac{1}{N}\hat{\mathbf{Q}}\mathbf{X}^T\right)\mathbf{A}$$

where $\mathbf{Q}_t = \mathbf{p}'(\mathbf{S}_t)/p(\mathbf{S}_t)$ and we have assumed the data are whited, $\mathbf{X}\mathbf{X}^T = N\mathbf{I}$. Many important conclusions can be reached from these equations: (i) the update of the mixing matrix is "freezing" in the sense that to zeroth order the new estimated mixing matrix is identical to the previous one. (ii) The first order correction of the mixing matrix is scaling with the change derived from the noiseless Bell & Sejnowski algorithm (the expression in the parenthesis). This shows that to fist order in the noise variance, there are no improvement of the noisy model compared to the noiseless counterpart. The consequence is that only for noise variances sufficiently large to make σ^4 important, is the noise model different (and hopefully better) than the noiseless model. (iii) The update of the noise is scaling with σ^4 which makes the change in the estimated noise extremely slow to converge when the noise levels are small.

4.2 The Overdetermined Case

To simplify the equations reasonably it is of even greater importance to assume that the data is whitened, i.e. $\mathbf{X}\mathbf{X}^T/N = \mathbf{I}$. When that is the case we have that

$$\mathbf{A}_{n+1} = \mathbf{A}_n + \sigma^2 \tilde{\mathbf{A}}_n + \mathcal{O}(\sigma^4)$$

$$\tilde{\mathbf{A}}_n = (\mathbf{I} - \mathbf{A}\mathbf{A}^+)\frac{\mathbf{X}\hat{\mathbf{Q}}^T}{N} - \mathbf{A} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\frac{\hat{\mathbf{Q}}\mathbf{X}^T}{N}\mathbf{A}$$

$$\sigma^2_{bias} = 1 - \frac{\text{rank}(\mathbf{A})}{m}, \qquad \mathbf{U} = \mathbf{\Delta}^T\mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+/N$$

where $\mathbf{\Delta} = \mathbf{I} - \mathbf{A}\mathbf{A}^+$. Following should be noted about these equations: (i) As in the square case, the update in the mixing matrix in freezing, resulting in extremely slow convergence in the low noise limit. (ii) In the correction term for mixing matrix is, we can recognize the structure which leads to the BS solution in the square case and see that it is the first term which is the difference. Not surprisingly, this term is increasing as the ratio of sensors and sources are increasing. (iii) The noise now also have a bias term. On an intuitive level this can be explained as the spill-over from the misjudgement of the number of sources.

4.3 Adaptive Overrelaxed EM

The EM-variant Adaptive Overrelaxed EM is found in [10] and is a very easy and general applicable speedup suggestion. Basically one is enlarging the step size proposed by EM with a factor η

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

When we combine the low noise analysis with this idea we get

$$\mathbf{A}_{n+1} = \mathbf{A}_n + \eta\sigma^2\tilde{\mathbf{A}} + \mathcal{O}(\sigma^4)$$

That is, the trick of the adaptive Overrelaxed EM is directly countering the problem of the small noise variance. On the downside we are no longer guaranteed an increase in the log likelihood for each step, but this can be controlled with a test-step modification and the combined algorithm works well on ICA [9].

5 Conclusions

The saddle point approximation and the resulting analysis has demonstrated that

- The linear mixture models have bad convergence properties in the low noise limit, both for square and overdetermined case, and both for the mixing matrix and the noise variance.

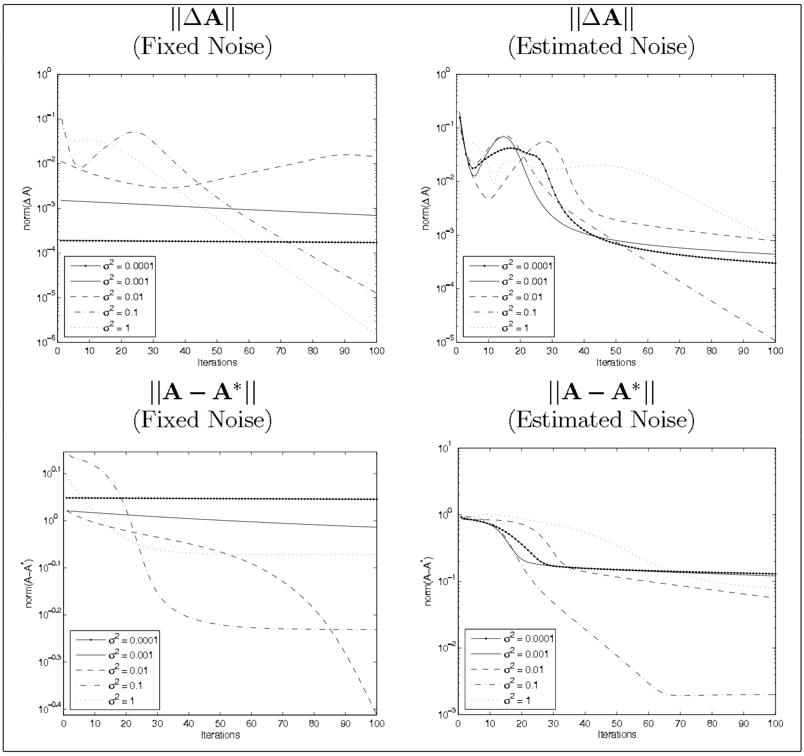


Figure 3: **Plot a)** upper left: The numerical difference in the mixing matrix in a problem with fixed noise at the generative level. Apart from the two largest noise levels, the delta values are exactly where they are predicted. The reason the two largest are not following the predict level is that they are already as converging to whatever solution are optimal for them. **Plot b)** upper right: Same as plot a) but without the fixed noise. Instead the noise is here estimated and results are therefore less clear. The predicted levels are clear obscured by the complications of the combined algorithm in which also initial conditions and choice of mixing matrix makes a difference. **Plot c)** lower left: The difference from the estimated mixing matrix to the optimal. As expected are the low-noise difference very slow to change. **Plot d)** lower right: As plot c) but with estimated noise. Clearly only the example with generative noise level of 1/100 is actually converging to an acceptable solution. This is because lower noise makes the algorithm freeze and higher noise levels make estimation impossible.

- The Overrelaxed Adaptive EM works because the step size directly counters the small noise variance.

Future work, which we are presently working on, is the generalize the presented results to non-linear observation models and to the Variational Bayes EM variant.

A Calculations and Details

In this section we make extensive use of the possible prewhitening of the data $\mathbf{X}\mathbf{X}^T = N\mathbf{I}$ and of the notation

$$\begin{aligned}
\boldsymbol{\alpha} &= \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T \\
&= \mathbf{A}\mathbf{A}^+ \\
\boldsymbol{\Delta} &= \mathbf{I} - \mathbf{A}\mathbf{A}^+ \\
\hat{\mathbf{Q}} &= \mathbf{p}'(\hat{\mathbf{S}})/p(\hat{\mathbf{S}})
\end{aligned}$$

A.1 The Moments

The moments, derived in the text, are repeated here for consistency

$$\begin{aligned}
\langle \mathbf{S} \rangle &= \hat{\mathbf{S}} + \sigma^2 \tilde{\mathbf{S}} + \mathcal{O}(\sigma^4) \\
\hat{\mathbf{S}} &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X} \\
\tilde{\mathbf{S}} &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{Q} \\
\langle \mathbf{S}\mathbf{S}^T \rangle &= \hat{\mathbf{S}}\hat{\mathbf{S}}^T + \sigma^2\mathbf{B} + \mathcal{O}(\sigma^4) \\
\hat{\mathbf{S}}\hat{\mathbf{S}}^T &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} \\
&= N(\mathbf{A}^T\mathbf{A})^{-1} \qquad \text{(whitening)} \\
\mathbf{B} &= N(\mathbf{A}^T\mathbf{A})^{-1} + (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{Q}\hat{\mathbf{S}}^T + \hat{\mathbf{S}}\mathbf{Q}^T(\mathbf{A}^T\mathbf{A})^{-1} \\
&= N(\mathbf{A}^T\mathbf{A})^{-1} + (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{Q}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} + (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X}\mathbf{Q}^T(\mathbf{A}^T\mathbf{A})^{-1} \\
&= (\mathbf{A}^T\mathbf{A})^{-1}[N\mathbf{A}^T\mathbf{A} + \mathbf{Q}\mathbf{X}^T\mathbf{A} + \mathbf{A}^T\mathbf{X}\mathbf{Q}^T](\mathbf{A}^T\mathbf{A})^{-1}
\end{aligned}$$

A.2 General (Non-square) identities

The following identities turn out to be useful in later calculations

$$\begin{aligned}
\mathbf{X}^T\mathbf{A}\hat{\mathbf{S}} &= \mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X} \\
\mathbf{X}^T\mathbf{A}\tilde{\mathbf{S}} &= \mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{Q} \\
\hat{\mathbf{S}}\hat{\mathbf{S}}^T &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} \\
\mathbf{A}^T\mathbf{A}\hat{\mathbf{S}}\hat{\mathbf{S}}^T &= \mathbf{A}^T\mathbf{X}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} \\
\mathbf{A}^T\mathbf{A}\mathbf{B} &= N\mathbf{I} + \mathbf{Q}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} + \mathbf{A}^T\mathbf{X}\mathbf{Q}^T(\mathbf{A}^T\mathbf{A})^{-1}
\end{aligned}$$

A.3 The Mixing Matrix

$$
\begin{aligned}
\mathbf{A} &= \mathbf{X}\langle\mathbf{S}\rangle^T\langle\mathbf{S}\mathbf{S}^T\rangle^{-1} \\
&= \mathbf{X}(\hat{\mathbf{S}}+\sigma^2\tilde{\mathbf{S}})^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T+\sigma^2\mathbf{B})^{-1} \\
&= \mathbf{X}(\hat{\mathbf{S}}+\sigma^2\tilde{\mathbf{S}})^T((\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}-\sigma^2(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}\mathbf{B}(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1})+\mathcal{O}(\sigma^4) \\
&= \mathbf{X}\hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}+\sigma^2(\mathbf{X}\tilde{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}-\mathbf{X}\hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}\mathbf{B}(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}) \\
&= \mathbf{X}\hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}+\sigma^2\mathbf{X}(\tilde{\mathbf{S}}^T-\hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}\mathbf{B})(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1} \\
&= \mathbf{X}\hat{\mathbf{S}}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}+\sigma^2\mathbf{X}\mathbf{F}^T(\hat{\mathbf{S}}\hat{\mathbf{S}}^T)^{-1}
\end{aligned}
$$

Overdetermined: Using $\boldsymbol{\alpha}_x = \mathbf{A}^T\mathbf{X}\mathbf{X}^T\mathbf{A}$

$$
\begin{aligned}
\hat{\mathbf{A}}_n &= \mathbf{X}\mathbf{X}^T\mathbf{A}(\mathbf{A}^T\mathbf{X}\mathbf{X}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{A} \\
\tilde{\mathbf{A}}_n &= \left\{\mathbf{X}\mathbf{Q}^T-\mathbf{X}\mathbf{X}^T\mathbf{A}\boldsymbol{\alpha}_x^{-1}[N\mathbf{A}^T\mathbf{A}+\mathbf{Q}\mathbf{X}^T\mathbf{A}+\mathbf{A}^T\mathbf{X}\mathbf{Q}^T]\right\}\boldsymbol{\alpha}_x^{-1}\mathbf{A}^T\mathbf{A}
\end{aligned}
$$

A.4 The Noise Variance

$$
\begin{aligned}
& \langle(\mathbf{X}-\mathbf{A}\mathbf{S})(\mathbf{X}-\mathbf{A}\mathbf{S})^T\rangle \\
=\ & \mathbf{X}\mathbf{X}^T+\mathbf{A}\langle\mathbf{S}\mathbf{S}^T\rangle\mathbf{A}^T-\mathbf{X}\langle\mathbf{S}\rangle^T\mathbf{A}^T-\mathbf{A}\langle\mathbf{S}\rangle\mathbf{X}^T \\
=\ & \mathbf{X}\mathbf{X}^T+\mathbf{A}(\hat{\mathbf{S}}\hat{\mathbf{S}}^T+\sigma^2\mathbf{B})\mathbf{A}^T-\mathbf{X}(\hat{\mathbf{S}}+\sigma^2\tilde{\mathbf{S}})^T\mathbf{A}^T-\mathbf{A}(\hat{\mathbf{S}}+\sigma^2\tilde{\mathbf{S}})\mathbf{X}^T \\
=\ & \mathbf{X}\mathbf{X}^T+\mathbf{A}\hat{\mathbf{S}}\hat{\mathbf{S}}^T\mathbf{A}^T-\mathbf{X}\hat{\mathbf{S}}^T\mathbf{A}^T-\mathbf{A}\hat{\mathbf{S}}\mathbf{X} \\
& \quad +\sigma^2\left(\mathbf{A}\mathbf{B}\mathbf{A}^T-\mathbf{X}\tilde{\mathbf{S}}^T\mathbf{A}^T-\mathbf{A}\tilde{\mathbf{S}}\mathbf{X}^T\right) \\
=\ & \mathbf{X}\mathbf{X}^T+\mathbf{A}\hat{\mathbf{S}}\hat{\mathbf{S}}^T\mathbf{A}^T-\mathbf{X}\hat{\mathbf{S}}^T\mathbf{A}^T-\mathbf{A}\hat{\mathbf{S}}\mathbf{X} \\
& \quad +\sigma^2\left(\hat{\mathbf{A}}\mathbf{B}\hat{\mathbf{A}^T}-\mathbf{X}\tilde{\mathbf{S}}^T\hat{\mathbf{A}}^T-\hat{\mathbf{A}}\tilde{\mathbf{S}}\mathbf{X}^T\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-\mathbf{A}\hat{\mathbf{S}})(\mathbf{X}-\mathbf{A}\hat{\mathbf{S}})^T+\sigma^2\left(\ldots_1\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-(\hat{\mathbf{A}}+\sigma^2\tilde{\mathbf{A}})\hat{\mathbf{S}})(\mathbf{X}-(\hat{\mathbf{A}}+\sigma^2\tilde{\mathbf{A}})\hat{\mathbf{S}})^T+\sigma^2\left(\ldots\right)+\mathcal{O}(\sigma^4) \\
=\ & ((\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})-\sigma^2\tilde{\mathbf{A}}\hat{\mathbf{S}})((\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})-\sigma^2\tilde{\mathbf{A}}\hat{\mathbf{S}})^T+\sigma^2\left(\ldots_1\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})(\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})^T-\sigma^2[(\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})\hat{\mathbf{S}}^T\tilde{\mathbf{A}}^T+\tilde{\mathbf{A}}\hat{\mathbf{S}}(\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})^T]+\sigma^2\left(\ldots_1\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})(\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})^T-\sigma^2[\ldots_2]+\sigma^2\left(\ldots_1\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})(\mathbf{X}-\hat{\mathbf{A}}\hat{\mathbf{S}})^T+\sigma^2\left(\ldots_1-\ldots_2\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{X}-\hat{\mathbf{A}}(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^T\mathbf{X})(\mathbf{X}-\hat{\mathbf{A}}(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^T\mathbf{X})^T+\sigma^2\left(\ldots_1-\ldots_2\right)+\mathcal{O}(\sigma^4) \\
=\ & (\mathbf{I}-\boldsymbol{\alpha})\mathbf{X}\mathbf{X}^T(\mathbf{I}-\boldsymbol{\alpha})^T+\sigma^2\left(\ldots_1-\ldots_2\right)+\mathcal{O}(\sigma^4) \\
=\ & N(\mathbf{I}-\boldsymbol{\alpha})(\mathbf{I}-\boldsymbol{\alpha})^T+\sigma^2\left(\ldots_1-\ldots_2\right)+\mathcal{O}(\sigma^4) \\
=\ & N\boldsymbol{\Delta}\boldsymbol{\Delta}^T+\sigma^2\left(\ldots_1-\ldots_2\right)+\mathcal{O}(\sigma^4)
\end{aligned}
$$

$$
\begin{aligned}
(\ldots_1) &= \hat{\mathbf{A}}\mathbf{B}\hat{\mathbf{A}}^T - \mathbf{X}\tilde{\mathbf{S}}^T\hat{\mathbf{A}}^T - \hat{\mathbf{A}}\tilde{\mathbf{S}}\mathbf{X}^T \\
&= \hat{\mathbf{A}}\Big[(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}[N\hat{\mathbf{A}}^T\hat{\mathbf{A}} + \mathbf{Q}\mathbf{X}^T\hat{\mathbf{A}} + \hat{\mathbf{A}}^T\mathbf{X}\mathbf{Q}^T](\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\Big]\hat{\mathbf{A}}^T \\
&\quad -\mathbf{X}((\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\mathbf{Q})^T\hat{\mathbf{A}}^T - \hat{\mathbf{A}}(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\mathbf{Q}\mathbf{X}^T \\
&= (\hat{\mathbf{A}}^+)^T[N\hat{\mathbf{A}}^T\hat{\mathbf{A}} + \mathbf{Q}\mathbf{X}^T\hat{\mathbf{A}} + \hat{\mathbf{A}}^T\mathbf{X}\mathbf{Q}^T]\hat{\mathbf{A}}^+ - \mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+ - (\hat{\mathbf{A}}^+)^T\mathbf{Q}\mathbf{X}^T \\
&= N\hat{\mathbf{A}}\hat{\mathbf{A}}^+ + (\hat{\mathbf{A}}^+)^T\mathbf{Q}\mathbf{X}^T\hat{\mathbf{A}}\hat{\mathbf{A}}^+ + (\hat{\mathbf{A}}\hat{\mathbf{A}}^+)^T\mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+ - \mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+ - (\hat{\mathbf{A}}^+)^T\mathbf{Q}\mathbf{X}^T \\
&= N\boldsymbol{\alpha} + (\hat{\mathbf{A}}^+)^T\mathbf{Q}\mathbf{X}^T(\hat{\mathbf{A}}\hat{\mathbf{A}}^+ - \mathbf{I}) + ((\hat{\mathbf{A}}\hat{\mathbf{A}}^+)^T - \mathbf{I})\mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+ \\
&= N\boldsymbol{\alpha} - \{(\hat{\mathbf{A}}^+)^T\mathbf{Q}\mathbf{X}^T\boldsymbol{\Delta} + \boldsymbol{\Delta}^T\mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+\}
\end{aligned}
$$

$$
\begin{aligned}
(\ldots_2) &= (\mathbf{X} - \hat{\mathbf{A}}\hat{\mathbf{S}})\hat{\mathbf{S}}^T\tilde{\mathbf{A}}^T + \tilde{\mathbf{A}}\hat{\mathbf{S}}(\mathbf{X} - \hat{\mathbf{A}}\hat{\mathbf{S}})^T \\
&= (\mathbf{X}\hat{\mathbf{S}}^T - \hat{\mathbf{A}}\hat{\mathbf{S}}\hat{\mathbf{S}}^T)\tilde{\mathbf{A}}^T + \tilde{\mathbf{A}}(\mathbf{X}\hat{\mathbf{S}}^T - \hat{\mathbf{A}}\hat{\mathbf{S}}\hat{\mathbf{S}}^T) \\
&= (N\hat{\mathbf{A}}(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1} - N\hat{\mathbf{A}}(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1})\tilde{\mathbf{A}}^T \\
&\qquad +\tilde{\mathbf{A}}(N(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^T - N(\hat{\mathbf{A}}^T\hat{\mathbf{A}})^{-1}\hat{\mathbf{A}}^T) \\
&= \mathbf{0}
\end{aligned}
$$

In the square case, $\boldsymbol{\Delta} = \mathbf{0}$ and $\boldsymbol{\alpha} = \mathbf{I}$, and therefore

$$
\begin{aligned}
\sigma^2_{n+1} &= \mathrm{Tr}(\langle(\mathbf{X} - \mathbf{A}\mathbf{S})(\mathbf{X} - \mathbf{A}\mathbf{S})^T\rangle)/(Nm) \\
&= \sigma^2 + \mathcal{O}(\sigma^4)
\end{aligned}
$$

In the overdetermined case we have

$$
\begin{aligned}
\sigma^2_{n+1} &= \mathrm{Tr}(\langle(\mathbf{X} - \mathbf{A}\mathbf{S})(\mathbf{X} - \mathbf{A}\mathbf{S})^T\rangle)/(Nm) \\
&= \mathrm{Tr}(\boldsymbol{\Delta}\boldsymbol{\Delta}^T)/m + \sigma^2\Big(\mathrm{Tr}(\boldsymbol{\alpha})/m - \mathrm{Tr}(\mathbf{U}^T + \mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&\qquad \text{where } \mathbf{U} = \boldsymbol{\Delta}^T\mathbf{X}\mathbf{Q}^T\hat{\mathbf{A}}^+/N \\
&= \mathrm{Tr}((\mathbf{I} - \boldsymbol{\alpha})(\mathbf{I} - \boldsymbol{\alpha})^T)/m + \sigma^2\Big(\mathrm{Tr}(\boldsymbol{\alpha})/m - \mathrm{Tr}(\mathbf{U}^T + \mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= \mathrm{Tr}(\mathbf{I} - 2\boldsymbol{\alpha} + \boldsymbol{\alpha}\boldsymbol{\alpha})/m + \sigma^2\Big(\mathrm{Tr}(\boldsymbol{\alpha})/m - 2\mathrm{Tr}(\mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= \mathrm{Tr}(\mathbf{I} - \boldsymbol{\alpha})/m + \sigma^2\Big(\mathrm{Tr}(\boldsymbol{\alpha})/m - 2\mathrm{Tr}(\mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= 1 - \frac{\mathrm{Tr}(\boldsymbol{\alpha})}{m} + \sigma^2\Big(\mathrm{Tr}(\boldsymbol{\alpha})/m - 2\mathrm{Tr}(\mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= 1 - \frac{\mathrm{rank}(\boldsymbol{\alpha})}{m} + \sigma^2\Big(\frac{\mathrm{rank}(\boldsymbol{\alpha})}{m} - 2\mathrm{Tr}(\mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= 1 - \frac{\mathrm{rank}(\mathbf{A})}{m} + \sigma^2\Big(\frac{\mathrm{rank}(\mathbf{A})}{m} - 2\mathrm{Tr}(\mathbf{U})/m\Big) + \mathcal{O}(\sigma^4) \\
&= \sigma^2_{bias} + \sigma^2\Big(\frac{\mathrm{rank}(\mathbf{A})}{m} - \frac{2}{m}\mathrm{Tr}(\mathbf{U})\Big) + \mathcal{O}(\sigma^4)
\end{aligned}
$$

References

[1] Hagai Attias. Independent factor analysis. *Neural Computation*, (11):803–851, 1999.

[2] Anthony J. Bell and Terrence J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[3] Adel Belouchrani and Jean-Francois Cardoso. A maximum likelihood source separation for discrete sources. In *Proceedings of EUSIPCO*, volume 2, pages 768–771, 1994.

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

[5] Sabine Deligne and Ramesh A. Gopinath. An em algorithm for convolutive independent component analysis. *Neurocomputing*, 49(1-4):187–211, 2002.

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

[7] Harry Lappalainen and Petteri Pajunen. Fast algorithms for bayesian independent component analysis. In *Proceedings of the ICA Conference*, 2000.

[8] Eric Moulines, Jean-Francois Cardoso, and Elisabeth Gassiat. Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In *Proceedings of the ICA Conference*, 1997.

[9] Kaare Brandt Petersen and Ole Winther. The em algorithm in independent component analysis. In *International Conference on Acoustics, Speech, and Signal Processing*, 2005.

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

[11] Max Welling and Markus Weber. A constrained em algorithm for independent component analysis. *Neural Computation*, 13, 2001.

C.5 Neural Computation 2005, I: "On the slow convergence of EM and VBEM in low noise linear mixtures"

(excluded due to copyright issues) (excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

C.6 Neural Computation 2005, II: "State-space models - from the EM algorithm to a gradient approach"

State-space models - from the EM algorithm to a gradient approach

Rasmus Kongsgaard Olsson

Kaare Brandt Petersen

Tue Lehn-Schiøler

Technical University of Denmark, 2800 Kongens Lyngby, Denmark

Abstract

Slow convergence is observed in the EM algorithm for linear state-space models. We propose to circumvent the problem by applying any off-the-shelf quasi-Newton-type optimizer, which operates on the gradient of the log-likelihood function. Such an algorithm is a practical alternative due to the fact that the exact gradient of the log-likelihood function can be computed by recycling components of the EM algorithm. We demonstrate the efficiency of the proposed method in three relevant instances of the linear state-space model. In high signal to noise ratios, where EM is particularly prone to converge slowly, it is

shown that gradient-based learning results in a sizable reduction of the computation time.

1 Introduction

State-space models are widely applied in cases where the data is generated by some underlying dynamics. Control engineering and speech enhancement are typical examples of applications of state-space models, where the state-space has a clear physical interpretation. Black-box modeling constitutes a different type of application: the state-space dynamics have no direct physical interpretation, only the generalization ability of the model matters, that is, the prediction error on unseen data.

A fairly general formulation of *linear* state space models (without deterministic input) is:

$$\mathbf{s}_t = \mathbf{F}\mathbf{s}_{t-1} + \mathbf{v}_t \tag{1}$$

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{n}_t \tag{2}$$

where equations (1) and (2) describe the state and observation spaces, respectively. State and observation vectors, $\mathbf{s}_t$ and $\mathbf{x}_t$, are random processes driven by i.i.d. zero-mean Gaussian inputs $\mathbf{v}_t$ and $\mathbf{n}_t$ with covariance $\mathbf{Q}$ and $\mathbf{R}$, respectively.

Optimization in state-space models by maximizing the log-likelihood, $\mathcal{L}(\boldsymbol{\theta})$, with respect to the parameters, $\boldsymbol{\theta} \equiv \{\mathbf{Q}, \mathbf{R}, \mathbf{F}, \mathbf{A}\}$, fall in two main categories based on either gradients (scoring) or Expectation Maximization (EM).

The principal approach to maximum likelihood in state-space models, and more generally in complex models, is to iteratively search the space of $\boldsymbol{\theta}$ for the local maximum of $\mathcal{L}(\boldsymbol{\theta})$ by taking steps in the direction of the gradient, $\nabla_{\theta}\mathcal{L}(\boldsymbol{\theta})$. A basic *ascend* algorithm can be improved by supplying curvature information, i.e. second-order derivatives, line-search, etc. Often, numerical methods are used to compute the gradient and the Hessian, due to the complexity associated with the computation of these quantities. In [Gupta and Mehra, 1974] and [Sandell and Yared, 1978] fairly complex recipes are given for the computation of the analytical gradient in the linear state-space model.

The EM algorithm, [Dempster et al., 1977], is an important alternative to gradient-based maximum-likelihood, partly due to its simplicity and convergence guarantees. It was first applied to the optimization of linear state-space models by [Shumway and Stoffer, 1982] and [Digalakis et al., 1993]. A general class of linear Gaussian (state-space) models was treated in [Roweis and Ghahramani, 1999], in which the EM algorithm was the main engine of estimation. In the context of Independent Component Analysis (ICA), the EM algorithm has been applied in among other [Moulines et al., 1997] and [Højen-Sørensen et al., 2002]. In [Olsson and Hansen, 2004, 2005], the EM algorithm was applied to the Convolutive ICA problem.

A number of authors have reported the slow convergence of the EM algorithm. In [Redner and Walker, 1984], impractically slow convergence in 2-component Gaussian mixture models is documented. This critique is, however, moderated by [Xu and Jordan, 1996]. Modifications have been suggested to speed up of the basic EM algorithm, see e.g. [Lachlan and Krishnan, 1997]. In [Jamshidian and Jennrich, 1997], the EM update, $\tilde{\mathbf{g}}_n \equiv \boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n$, is employed as a socalled generalized gradient in various variations of the quasi-Newton algorithm. The approach taken in [Meilijson, 1989] differs in that the gradient of the log-likelihood is derived from the E step, lending from a theorem originally shown by Fisher. Subsequently, a Newton-type step can be devised to replace the M step.

The main contribution of this paper is to demonstrate that specialized gradient-based optimization software can replace the EM algorithm, at little analytical cost, reusing the components of the EM algorithm itself. This procedure is termed the *Easy Gradient recipe*. Furthermore, empirical evidence, supporting the results in [Bermond and Cardoso, 1999], is presented to demonstrate that the signal-to-noise ratio (SNR) has a dramatic effect on the convergence speed of the EM algorithm. Under certain circumstances, i.e. in high SNR settings, the EM algorithm fails to converge in reasonable time.

Three applications of state-space models are investigated: 1) Sensor fusion for the black-box modelling of speech-to-face mapping problem. 2) Mean Field Independent Component Analysis (Mean Field ICA) for estimating a number of hidden independent sources that have been linearly and instantaneously mixed, 3) Convolutive Independent Component Analysis for convolutive mixtures.

In section 2, an introduction to EM and the Easy Gradient Recipe is given. More specifically, the relation between the various acceleration schemes is reviewed in section 2.3. In section 3, the implicated models are stated and in section 4 simulation results are presented.

2 Theory

Assume a model with observed variables $\mathbf{x}$, state space variables $\mathbf{s}$ and parameters $\boldsymbol{\theta}$: the calculation of the log-likelihood involves an integral over the state space variables:

$$\mathcal{L}(\boldsymbol{\theta}) = \ln p(\mathbf{x}|\boldsymbol{\theta}) = \ln \int p(\mathbf{x}|\mathbf{s},\boldsymbol{\theta})p(\mathbf{s}|\boldsymbol{\theta})d\mathbf{s} \tag{3}$$

The marginalization in equation (3) is intractable for most choices of distributions, hence direct optimization is rarely an option, even in the Gaussian case. Therefore a lower bound, B, is introduced on the log-likelihood, which is valid for any choice of probability density function, $q(\mathbf{s}|\boldsymbol{\phi})$:

$$B(\boldsymbol{\theta},\boldsymbol{\phi}) = \int q(\mathbf{s}|\boldsymbol{\phi}) \ln \frac{p(\mathbf{s},\mathbf{x}|\boldsymbol{\theta})}{q(\mathbf{s}|\boldsymbol{\phi})} d\mathbf{s} \leq \ln p(\mathbf{x}|\boldsymbol{\theta}) \tag{4}$$

At this point the problem seems to have been made *more* complicated, but the lower bound B has a number of appealing properties which makes the original task of finding the parameters easier. One important fact about B becomes clear when we rewrite it using Bayes' theorem

$$B(\boldsymbol{\theta},\boldsymbol{\phi}) = \ln p(\mathbf{x}|\boldsymbol{\theta}) - KL[q(\mathbf{s}|\boldsymbol{\phi})||p(\mathbf{s}|\mathbf{x},\boldsymbol{\theta})] \tag{5}$$

where KL denotes the Kullback-Leibler divergence between the two distributions. In the case that the variational distribution, q, is chosen to be exactly the posterior of the hidden variables, $p(\mathbf{s}|\mathbf{x},\boldsymbol{\theta})$, the bound, B, is equal to the log-likelihood. For this reason one often tries to choose the variational distribution flexible enough to include the true posterior and yet simple enough to make the necessary calculations as easy as possible. However, in cases where computational efficiency is the main priority, a simplistic q is chosen that does not necessarily include $p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$, and the optimum of B may differ from that of $\mathcal{L}$. Examples of applications involving both types of q are described in sections 3 and 4.

The approach is to maximize B with respect to $\boldsymbol{\phi}$, in order to make the lower bound as close as possible to the log-likelihood and then maximize the bound with respect to the parameters $\boldsymbol{\theta}$. This stepwise maximization can be achieved via the EM algorithm or by applying the *Easy Gradient Recipe*, see section 2.2.

2.1 The EM Update

The EM algorithm, as formulated in [Neal and Hinton, 1998], works in a straightforward scheme which is initiated with random values and iterated until suitable convergence is reached:

E: Maximize $B(\boldsymbol{\theta}, \boldsymbol{\phi})$ w.r.t. $\boldsymbol{\phi}$ keeping $\boldsymbol{\theta}$ fixed.

M: Maximize $B(\boldsymbol{\theta}, \boldsymbol{\phi})$ w.r.t. $\boldsymbol{\theta}$ keeping $\boldsymbol{\phi}$ fixed.

It is guaranteed that the lower bound function does not decrease on any combined E and M step. Figure 1 illustrates the EM algorithm. The convergence is often slow - e.g. the curvature of the bound function, B, might be be much higher than that of $\mathcal{L}$, resulting in very conservative parameter updates. As mentioned in the introduction, this is particularly a problem in latent variable models with low-power additive noise. In [Bermond and Cardoso, 1999] and [Petersen and Winther, 2005], it is demonstrated that the EM update of the parameter $\mathbf{A}$ in equation (2) scales with the observation noise level, $\mathbf{R}$. That is, as the signal-to-noise ratio increases, the M-step change in $\mathbf{A}$ decreases, and more iterations are required to converge.

2.2 The Easy Gradient Recipe

The key idea is to regard the bound, B, as a function of $\boldsymbol{\theta}$ only, as opposed to a function of both the parameters $\boldsymbol{\theta}$ and the variational parameters $\boldsymbol{\phi}$. As a result, the lower bound can be applied to reformulate the log-likelihood

$$\mathcal{L}(\boldsymbol{\theta}) = B(\boldsymbol{\theta}, \boldsymbol{\phi}_*) \tag{6}$$

where $\boldsymbol{\phi}_* = \boldsymbol{\phi}_*(\boldsymbol{\theta})$ satisfies the constraint $q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$. Comparing with equation (5), it is easy to see that $\boldsymbol{\phi}_*$ maximizes the bound. Since $\boldsymbol{\phi}_*$ is exactly minimizing the KL-divergence, the partial derivative of the bound with respect to $\boldsymbol{\phi}$ evaluated in the point $\boldsymbol{\phi}_*$, is equal to zero. Therefore the derivative of $B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$ is equal to the partial derivative

$$\frac{dB(\boldsymbol{\theta}, \boldsymbol{\phi}_*)}{d\boldsymbol{\theta}} = \frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)}{\partial \boldsymbol{\theta}} + \frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}\bigg|_{\boldsymbol{\phi}_*} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{\theta}}\bigg|_{\boldsymbol{\phi}_*} = \frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)}{\partial \boldsymbol{\theta}} \tag{7}$$

and due to the choice of $\boldsymbol{\phi}_*$, the derivative of the log-likelihood is the partial derivative of the bound

$$\frac{d\mathcal{L}(\boldsymbol{\theta})}{d\boldsymbol{\theta}} = \frac{dB(\boldsymbol{\theta}, \boldsymbol{\phi}_*)}{d\boldsymbol{\theta}} = \frac{\partial B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)}{\partial \boldsymbol{\theta}}$$

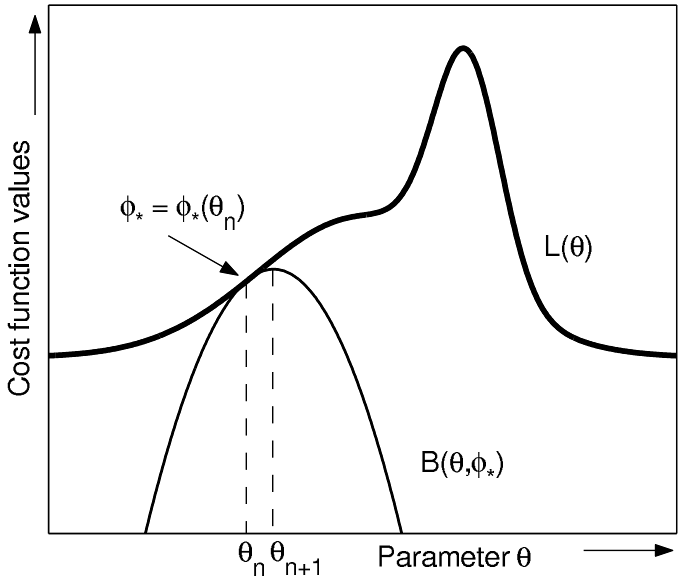


Figure 1: Schematic illustration of lower bound optimization for a one-dimensional estimation problem, where $\boldsymbol{\theta}_n$ and $\boldsymbol{\theta}_{n+1}$ are iterates of the standard EM algorithm. The log-likelihood function, $\mathcal{L}(\boldsymbol{\theta})$, is bounded from below by the function $B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$. The bound attains equality to $\mathcal{L}$ in $\boldsymbol{\theta}_n$ due to the choice of variational distribution: $q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta}_n)$. Furthermore in $\boldsymbol{\theta}_n$, the derivatives of the bound and the log-likelihood are identical. In many situations, the curvature of $B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$ is much higher than that of $\mathcal{L}(\boldsymbol{\theta})$, leading to small changes in the parameter, $\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n$.

which can be realized by combining equations (6) and (7).

In this way, exact values and gradients of the true log likelihood can be obtained using the lower bound. This observation is not new, since it is essentially the same which is used in [Salakhutdinov et al., 2003] to construct the so-called Expected Conjugated Gradient Algorithm (ECG). The novelty of the recipe is rather the practical recycling of low-complexity computations carried out in connection to the EM algorithm for a much more efficient optimization using *any* gradient-based optimizer. This can be expressed in MATLAB-style pseudo-code where a function `loglikelihood` receives as argument the parameter $\boldsymbol{\theta}$ and returns $\mathcal{L}$ and its gradient $\frac{d\mathcal{L}}{d\boldsymbol{\theta}}$:

	`function` $[\mathcal{L}, \frac{d\mathcal{L}}{d\boldsymbol{\theta}}]$ = `loglikelihood`$(\boldsymbol{\theta})$
1	Find $\boldsymbol{\phi}^*$ such that $\frac{\partial B}{\partial \boldsymbol{\phi}}\big|_{\boldsymbol{\phi}^*} = 0$
2	Calculate $\mathcal{L} = B(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$
3	Calculate $\frac{d\mathcal{L}}{d\boldsymbol{\theta}} = \frac{\partial B}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}, \boldsymbol{\phi}^*)$

Step **1**, and to some extend step **2**, are obtained by performing an E-step, while **3** requires only little programming that implements the gradients used to solve for the M-step. Compared to the EM algorithm, the main advantage is that the function value and gradient can be fed to any gradient-based optimizer, which in most cases provides a substantial improvement of the convergence properties. In that sense, it is possible to benefit from the speed-ups of advanced gradient-based optimization. In cases when q does not contain $p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$, the Easy Gradient Recipe will converge to generalized EM solutions.

The advantage of formulating the log-likelihood using the bound function, B, depends on the task at hand. In the linear state-space model, eq. (1) and (2), a naive computation of $\frac{d\mathcal{L}}{d\boldsymbol{\theta}}$ involves the somewhat complicated derivatives of the Kalman Filter equations with respect to each of the parameters in $\boldsymbol{\theta}$, this is explained in, e.g. [Gupta and Mehra, 1974]. Consequently, it leads to a cost of $\dim[\boldsymbol{\theta}]$ times the cost of one Kalman Filter filter sweep.[1] When using the Easy Gradient Recipe, $\frac{d\mathcal{L}}{d\boldsymbol{\theta}}$ is derived via the gradient of the bound, $\frac{\partial B}{\partial \boldsymbol{\theta}}$, see appendix A for a derivation example. These derivatives are often available in connection to the derivation of the M-step. In addition, the computational cost is dominated by steps **1** and **2**, which require only a Kalman smoothing, scaling as 2 Kalman Filter filter sweeps, hence constituting an important reduction of computation time as compared to the naive gradient computation. [Sandell and Yared, 1978] noted in in their investigation of linear state-space models that a reformulation of the problem resulted in a similar reduction of the computational costs.

2.3 Relation to other speed-up methods

In this section, a series of extensions to the EM-algorithm will be discussed. They have in common the utilization of conjugate gradient or quasi-Newton steps, that is, the inverse Hessian is approximated e.g. using the gradient. Step-lengthening methods, which are often simpler in terms of analytical cost, have recently been explored in [Salakhutdinov and Roweis, 2003] and

[1] The computational complexity of the Kalman filter is $\mathcal{O}[N(d_s)^3]$, where N is the data length.

[Honkela et al., 2003]. They are, however, out of the scope of this presentation.

In order to conveniently describe the methods, additional notation along the lines of [Jamshidian and Jennrich, 1997] is introduced: the combined EM operator which maps the current estimate of the parameters, $\boldsymbol{\theta}$, to the new one, is denoted $\hat{\boldsymbol{\theta}} = \mathbf{M}(\boldsymbol{\theta})$. The change in $\boldsymbol{\theta}$ due to an EM update then is $\tilde{\mathbf{g}}(\boldsymbol{\theta}) = \mathbf{M}(\boldsymbol{\theta}) - \boldsymbol{\theta}$. In a sense, $\tilde{\mathbf{g}}(\boldsymbol{\theta})$ can be regarded a generalized gradient of an underlying cost-function, which has zeros identical to those of $\mathbf{g}(\boldsymbol{\theta}) = \frac{d\mathcal{L}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}$. The Newton-Raphson update can then be devised as:

$$\Delta\boldsymbol{\theta} = -\mathbf{J}(\boldsymbol{\theta})^{-1}\tilde{\mathbf{g}}(\boldsymbol{\theta})$$

where the Jacobian is defined $\mathbf{J}(\boldsymbol{\theta}) = \mathbf{M}'(\boldsymbol{\theta}) - \mathbf{I}$. A number of methods approximate $\mathbf{J}(\boldsymbol{\theta})^{-1}$ or $\mathbf{M}'(\boldsymbol{\theta})$ in various ways. For instance, the quasi-Newton method QN1 of [Jamshidian and Jennrich, 1997] approximates $\mathbf{J}^{-1}(\boldsymbol{\theta})$ by employing the Broyden update, see [Broyden, 1965]. The QN2 algorithm of the same publication performs quasi-Newton optimization on $\mathcal{L}(\boldsymbol{\theta})$, but its update direction can be written as an additive correction to the EM step: $\mathbf{d} = \tilde{\mathbf{g}}(\boldsymbol{\theta}) - \Psi\mathbf{g}(\boldsymbol{\theta})$. The correction term is updated through Ψ, by means of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update, see e.g. [Bishop, 1995]. A search is performed in the direction of $\mathbf{d}$ so as to maximize $\mathcal{L}(\boldsymbol{\theta}+\lambda\mathbf{d})$, where λ is the step length. QN2 has the disadvantage compared to QN1 that $\mathcal{L}(\boldsymbol{\theta})$ and $\mathbf{g}(\boldsymbol{\theta})$ have to be computed in addition to $\tilde{\mathbf{g}}(\boldsymbol{\theta})$. This is also the case of the conjugate gradient (CG) accelerator of [Jamshidian and Jennrich, 1993].

Method	$\mathbf{M}$	B'	B''	$\mathcal{L}$
EM [Dempster et al., 1977]	x	-	-	-
QN1 [Jamshidian and Jennrich, 1997]	x	-	-	-
QN2 [Jamshidian and Jennrich, 1997]	x	x	-	x
CG [Jamshidian and Jennrich, 1993]	x	x	-	x
LANGE [Lange, 1995]	-	x	x	x
EGR	x	x	-	x

Table 1: Analytical costs associated with a selection of EM speed-ups that are based on Newton-like updates. The x's indicate whether the quantity is required by the algorithm. It should be noted, that B', required by the Easy Gradient Recipe (EGR) as well as QN2, CG and LANGE, often is a by-product of the derivation of $\mathbf{M}$.

Similarly in [Lange, 1995], the log-likelihood, $\mathcal{L}(\boldsymbol{\theta})$, is optimized directly through quasi Newton steps, i.e. in the direction of $\mathbf{d} = -\mathbf{J}_{\mathcal{L}}^{-1}(\boldsymbol{\theta})\mathbf{g}(\boldsymbol{\theta})$. However, the iterative adaptation to the Hessian, $\mathbf{J}_{\mathcal{L}}(\boldsymbol{\theta})$, involves the computation of $B''(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$, which may require considerable human effort in many applications. In table 1, the analytical cost associated with the discussed algorithms are summarized.

The Easy Gradient Recipe lends from QN2 in that quasi-Newton optimization on $\mathcal{L}(\boldsymbol{\theta})$ is carried out. The key point here is that, the specific Quasi-Newton update of QN2 can be replaced by any software package of choice, which performs gradient-based optimization. This has the advantage that various highly sophisticated algorithms can be tried for the particular problem at hand. Furthermore, we emphasize as [Meilijson, 1989], that the gradient can be computed from the derivatives involved in solving for the E step. This means that the advantages of gradient-based optimization are obtained conveniently at little cost. In this letter, a quasi-Newton gradient-based optimizer has been chosen. The implementation of the BFGS algorithm is due to [Nielsen, 2000], and has built-in line search and trust region monitoring.

3 Models

The EM algorithm and the Easy Gradient Recipe were applied to three different models that can all be fitted into the linear state-space framework:

3.1 Kalman Filter Based Sensor Fusion

The state-space model of equations (1)+(2) can be used to describe systems where two different types of signals are measured. The signals could be, e.g. sound and images as [Lehn-Schiøler et al., 2005], where speech and lip movements were the observables. In this case, the observation equation (2) can be split into two parts.

$$\mathbf{x}_t = \begin{pmatrix} \mathbf{x}_t^1 \\ \mathbf{x}_t^2 \end{pmatrix} = \begin{pmatrix} \mathbf{A}^1 \\ \mathbf{A}^2 \end{pmatrix} \mathbf{s}_t + \begin{pmatrix} \mathbf{n}_t^1 \\ \mathbf{n}_t^2 \end{pmatrix}$$

where $\mathbf{n}_t^1 \sim N(\mathbf{0}, \mathbf{R}^1)$ and $\mathbf{n}_t^2 \sim N(\mathbf{0}, \mathbf{R}^2)$. The innovation noise in the state-space equation (1) is defined as $\mathbf{v}_t \sim N(\mathbf{0}, \mathbf{Q})$. In the training phase, the parameters of the system, $\mathbf{F}$, $\mathbf{A}^1$, $\mathbf{A}^2$, $\mathbf{R}^1$,$\mathbf{R}^2$,$\mathbf{Q}$ are estimated by maximum likelihood using either EM or a gradient-based method. When the parameters have been learned, the state-space variable $\mathbf{s}$, which represents unknown hidden causes, can be deduced from one of the observations ($\mathbf{x}_1$ or $\mathbf{x}_2$) and the missing observation can be estimated by mapping from the state space.

3.2 Mean Field ICA

In Independent Component Analysis (ICA), one tries to separate linearly mixed sources using the assumed statistical independence of the sources. In many cases elaborate source priors are necessary, which calls for more advanced separation techniques such as Mean Field ICA. The method, which was first introduced in [Højen-Sørensen et al., 2002], can handle complicated source priors in an efficient approximative manner.

The model in equation (2) is identical to an instantaneous ICA model provided that $\mathbf{F} = \mathbf{0}$ and that $p(\mathbf{v}_t)$ is reinterpreted as the (non-Gaussian) source prior. The basic generative model of the instantaneous ICA is

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{n}_t \tag{8}$$

where $\mathbf{n}_t$ is assumed i.i.d. Gaussian and $\mathbf{s}_t = \mathbf{v}_t$ is assumed distributed by a factorized prior $\prod_i p(v_{it})$, which is independent in both time and dimension. The Mean Field ICA is only approximately compatible with the Easy Gradient Recipe, since the variational distribution $q(\mathbf{s}|\boldsymbol{\phi})$ is not guaranteed to contain the posterior $p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$. However, acceptable solutions are retrieved, when q is chosen sufficiently flexible.

3.3 Convolutive ICA

Acoustic mixture scenarios are characterized by sound waves emitted by a number of sound sources propagating through the air and arriving at the sensors in delayed and attenuated versions. The instantaneous mixture model of standard ICA, equation (8), is clearly insufficiently describing this situation. In *convolutive* ICA the signal path (delay and attenuation) is modelled by an FIR filter, i.e. a convolution of the source by the impulse responses of the signal path:

$$\mathbf{x}_t = \sum_k \mathbf{C}_k \mathbf{s}_{t-k} + \mathbf{n}_t \tag{9}$$

where $\mathbf{C}_k$ is the mixing filter matrix. Equation (9) and the source independence assumption can be fitted into the state-space formulation of equations (1) and (2), see [Olsson and Hansen, 2004, 2005], by making the following model choices: 1) Noise inputs $\mathbf{v}_t$ and $\mathbf{n}_t$ are i.i.d. Gaussian. 2) The state vector is *augmented* to contain time-lagged values, i.e.

$$\bar{\mathbf{s}}_t \equiv [s_{1,t} s_{1,t-1} \dots s_{2,t} s_{2,t-1} \dots s_{d_s,t} s_{1,t-1} \dots]^\top$$

3) State-space parameter matrices (e.g. $\mathbf{F}$) are constrained to a special format (certain elements are fixed to 0's and 1's) in order to ensure the independency of the sources mentioned above.

4 Results

The simulations that are documented in the following serve to illustrate the well-known advantage of advanced gradient-based learning over standard

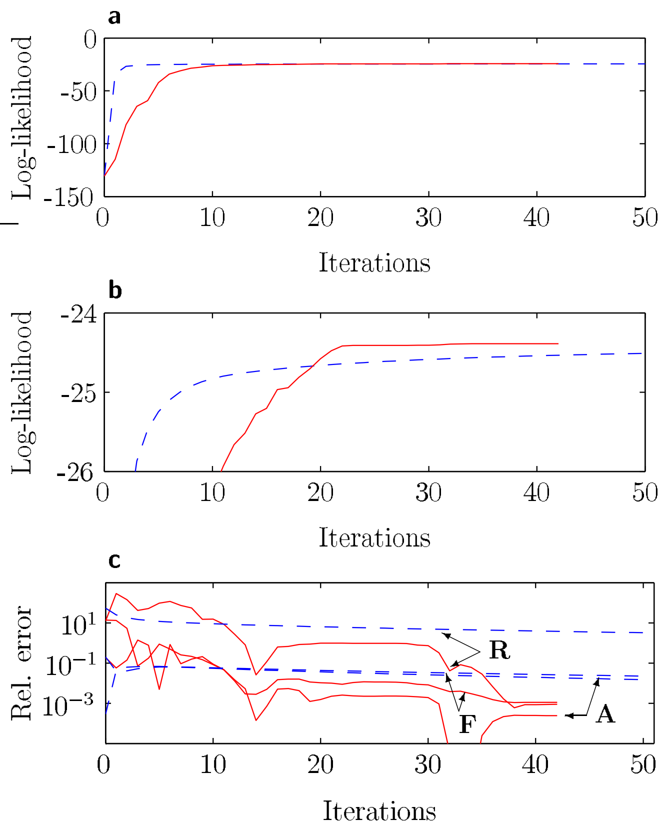


Figure 2: Convergence of EM (dashed) and a gradient-based method (dotted) in the ARMA(1,1) model. **a**: EM has faster initial convergence than the gradient-based method, but the final part is slow for EM. **b**: Zoom-in on the log-likelihood axis. Even after 50 iterations, EM has not reached the same level as the gradient based method. **c**: Parameter estimates convergence in terms of squared relative (to the generative parameters) error.

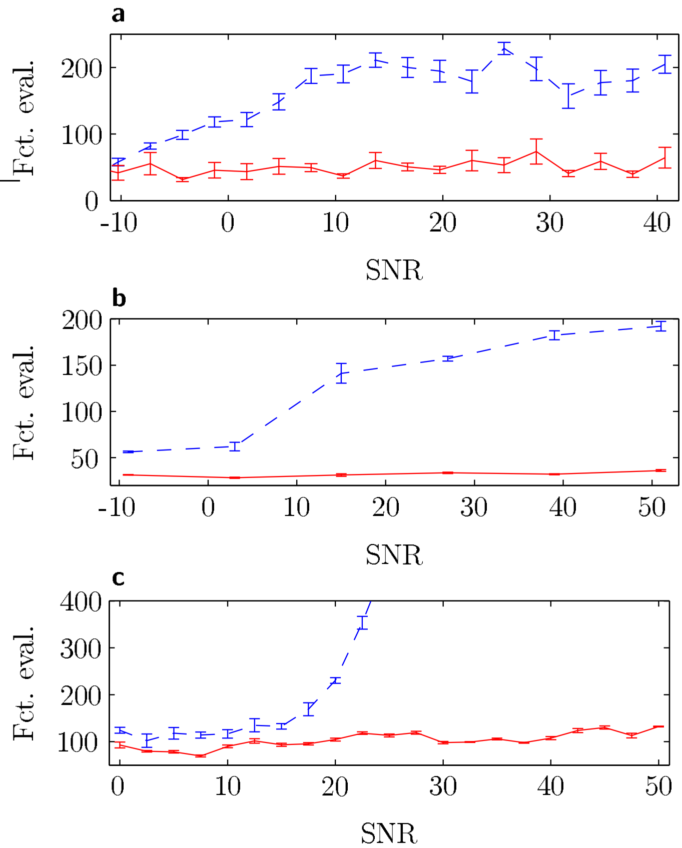


Figure 3: Number of function evaluations for EM (dashed) and gradient-based optimization (solid) to reach convergence as a function of signal to noise ratio for the three different problems. **a**: Kalman Filter Based Sensor Fusion. **b**: Mean Field ICA. **c**: Convolutive ICA. The level of convergence was defined as a relative change in log-likehood below 10^{-5} at which point the parameters did no longer change significantly. In the case of the EM algorithm, this sometimes occurred in plateau regions of the parameter space.

EM. Before advancing to the more involved applications described above, the advantage of gradient-based methods over EM will be explored for a one-dimensional linear state-space model, i.e. an ARMA(1,1) process. In this case, $\mathbf{F}$ and $\mathbf{A}$ are scalars as well as the observation variance $\mathbf{R}$ and the transition variance $\mathbf{Q}$. The latter is fixed to unity to resolve the inherent scale ambiguity of the model. As a consequence, the model has only 3 parameters. The BFGS optimizer mentioned in section 2 was used.

Figure 2 shows the convergence of both the EM algorithm and the gradient-based method. Initially, EM is fast, i.e. it rapidly approaches the maximum log-likelihood, but slows down as it gets closer to the optimum. The large dynamic range of the log-likelihood makes it difficult to ascertain the final increase in the log-likelihood, hence figure 2b provides a closeup on the log-likelihood scale. Table 2 gives an indication of the importance of the final increase. After 50 iterations, EM has reached a log-likelihood value of -24.5131, but the parameter values are still far off. After convergence, the log-likelihood has increased to -24.3883 which is still slightly worse than that obtained by the gradient-based method, but the parameters are now near the generative values. The BFGS algorithm used 43 iterations and 52 function evaluations. For large-scale state-space models, the computation time is all but dominated by the E-step computation. Hence, a function evaluation costs $\approx$1 E-step. Similar results are obtained when comparing the learning algorithms on the Kalman Filter Based Sensor Fusion, Mean Field ICA and Convolutive ICA problems.

As argued in section 2, it is demonstrated that the number of iterations re-

	Generative	Gradient	EM 50	EM ∞
Iterations	-	43	50	1800
Log-likelihood	-	-24.3882	-24.5131	-24.3883
F	0.5000	0.4834	0.5626	0.4859
A	0.3000	0.2953	0.2545	0.2940
R	0.0100	0.0097	0.0282	0.0103

Table 2: Estimation in the ARMA(1,1) model. The convergence of EM is slow compared to the gradient-based method. Note that after 50 EM iterations, the log-likelihood is relatively close to the value achieved at convergence, but the parameter values are far from the generative values.

quired by the EM algorithm to converge in state-space type models critically depends on the SNR. Figure 3 shows the performance of the two methods on the three different problems. The relevant comparison measure is computation time, which in the present examples are all but dominated by the E-step for EM and function evaluations (whose cost is also dominated by an E-step) for the gradient-based optimizer. The line search of the latter may require more function evaluations per iteration, but that was most often not the case for the BFGS algorithm of choice. The plots indicate that in the low-noise case, the EM algorithm requires relatively more time to converge whereas the gradient-based method performs equally well for all noise levels.

5 Conclusion

In applying the EM algorithm to maximum likelihood estimation in state-space models, we find, as many before us, that it has poor convergence properties in the low noise limit. Often a value 'close' to the maximum likelihood is reached in the first few iterations, while the final increase, which is crucial to the accurate estimation of the parameters, requires an excessive amount of iterations.

More importantly, we provide a simple scheme for efficient gradient-based optimization achieved by transformation from the EM formulation, i.e. the simple math and programming of the EM algorithms is preserved. Following this recipe, one can get the optimization benefits associated with *any* advanced gradient based-method. In this way, the tedious, problem-specific, analysis of the cost-function topology can be replaced with an off-the-shelf approach. Although the analysis provided in this article is limited to a set of linear mixture models, it is in fact applicable to *any* model subject to the EM algorithm, hence constituting a strong and general tool to be applied by the part of the neural community that uses the EM algorithm.

6 Acknowledgements

We thank Danish Oticon Fonden for supporting in part this research project. Inspiring discussions with Lars Kai Hansen and Ole Winther initiated this work.

Appendix A

In the following will be demonstrated how to obtain a partial derivative of the constrained bound function, $B(\boldsymbol{\theta}, \boldsymbol{\phi}_*)$. The derivation lends from [Shumway and Stoffer, 1982]. At $q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$, we can reformulate equation (4) as:

$$
\begin{aligned}
B(\boldsymbol{\theta}, \boldsymbol{\phi}_*) &= \left\langle \ln \frac{p(\mathbf{s}, \mathbf{x}|\boldsymbol{\theta})}{q(\mathbf{s}|\boldsymbol{\phi}_*)} \right\rangle \\
&= \langle \ln p(\mathbf{s}, \mathbf{x}|\boldsymbol{\theta}) \rangle - \langle \ln q(\mathbf{s}|\boldsymbol{\phi}_*) \rangle
\end{aligned}
$$

where the expectation $\langle \cdot \rangle$ is over the posterior $q(\mathbf{s}|\boldsymbol{\phi}_*) = p(\mathbf{s}|\mathbf{x}, \boldsymbol{\theta})$. Only the first term, $J(\boldsymbol{\theta}) = \langle \ln p(\mathbf{s}, \mathbf{x}|\boldsymbol{\theta}) \rangle$, depends on $\boldsymbol{\theta}$. The joint variable distribution factors due to the Markov property of the state-space model:

$$
p(\mathbf{s}, \mathbf{x}|\theta) = p(\mathbf{s}_1|\theta) \prod_{k=2}^{N} p(\mathbf{s}_k|\mathbf{s}_{k-1}, \theta) \prod_{k=1}^{N} p(\mathbf{x}_k|\mathbf{s}_k, \theta).
$$

where N is the length of the time series and the initial state, $\mathbf{s}_1$, is normally distributed with mean $\boldsymbol{\mu}_1$ and covariance $\boldsymbol{\Sigma}_1$. Using the fact that all variables are Gaussian, the expected log-distributions can be written out as:

$$
\begin{aligned}
J(\boldsymbol{\theta}) = & -\frac{1}{2}\Big\{ \dim(\mathbf{s}) \ln 2\pi + \ln |\Sigma_1| \\
& + \left\langle (\mathbf{s}_1 - \boldsymbol{\mu}_1)^\top \Sigma_1^{-1} (\mathbf{s}_1 - \boldsymbol{\mu}_0) \right\rangle \\
& + (N-1) \left[\dim(\mathbf{s}) \ln 2\pi + \ln |\mathbf{Q}| \right] \\
& + \sum_{k=2}^{N} \left\langle (\mathbf{s}_k - \mathbf{F}\mathbf{s}_{k-1})^\top \mathbf{Q}^{-1} (\mathbf{s}_k - \mathbf{F}\mathbf{s}_{k-1}) \right\rangle) \\
& + N \left[\dim(\mathbf{x}) \ln 2\pi + \ln |\mathbf{R}| \right] \\
& + \sum_{k=1}^{N} \left\langle (\mathbf{x}_k - \mathbf{A}\mathbf{s}_k)^\top \mathbf{R}^{-1} (\mathbf{x}_k - \mathbf{A}\mathbf{s}_k) \right\rangle \Big\}
\end{aligned}
$$

The gradient with respect to $\mathbf{A}$ is:

$$\begin{aligned}\frac{\partial J(\boldsymbol{\theta})}{\partial \mathbf{A}} &= -\frac{1}{2}\sum_{k=1}^{N}\left(2\mathbf{R}^{-1}\mathbf{A}\left\langle \mathbf{s}_k\mathbf{s}_k^\top\right\rangle - 2\mathbf{R}^{-1}\mathbf{x}_k\left\langle \mathbf{s}_k^\top\right\rangle\right)\\ &= -\mathbf{R}^{-1}\mathbf{A}\sum_{k=1}^{N}\left\langle \mathbf{s}_k\mathbf{s}_k^\top\right\rangle + \mathbf{R}^{-1}\sum_{k=1}^{N}\mathbf{x}_k\left\langle \mathbf{s}_k^\top\right\rangle)\end{aligned}$$

It is seen that the gradient depends on the marginal posterior source moments $\left\langle \mathbf{s}_k\mathbf{s}_k^\top\right\rangle$ and $\left\langle \mathbf{s}_k^\top\right\rangle$, which are provided by the Kalman smoother. The gradients with respect to $\mathbf{F}$ and $\mathbf{Q}$ furthermore requires the marginal moment $\left\langle \mathbf{s}_k\mathbf{s}_{k-1}^\top\right\rangle$.

The derivation of the gradients for the covariances $\mathbf{R}$ and $\mathbf{Q}$, must respect the symmetry of these matrices. Furthermore, steps must be taken to ensure the positive definiteness following a gradient step. This can be achieved e.g. by adapting $\mathbf{Q}_0$ instead of $\mathbf{Q}$ where $\mathbf{Q} = \mathbf{Q}_0\mathbf{Q}_0^\top$.

References

Bermond, O. and Cardoso, J.-F. (1999). Approximate likelihood for noisy mixtures. In *Proceedings of the ICA Conference*, pages 325–330.

Bishop, C. (1995). *Neural networks for pattern recognition.* Oxford University Press.

Broyden, C. G. (1965). A class of methods for solving nonlinear simultanoeus equations. *Math. Comput.*, 19:577–593.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistics Society, Series B*, 39:1–38.

Digalakis, V., Rohlicek, J., and Ostendorf, M. (1993). ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition. *IEEE Transactions on Speech and Audio Processing*, 1(4):431–442.

Gupta, N. K. and Mehra, R. K. (1974). Computational aspects of maximum likelihood estimation and reduction in sensitivity calculations. *IEEE Transactions on Automatic Control*, AC-19(1):774–783.

Højen-Sørensen, P. A., Winther, O., and Hansen, L. K. (2002). Mean field approaches to independent component analysis. *Neural Computation*, 14:889–918.

Honkela, A., Valpola, H., and Karhunen, J. (2003). Accelerating cyclic update algorithms for parameter estimation by pattern searches. *Neural processing letters*, 17(2):191–203.

Jamshidian, M. and Jennrich, R. I. (1993). Conjugate gradient acceleration of the EM algorithm. *Journal of the American Statistical Association*, 88:221–228.

Jamshidian, M. and Jennrich, R. I. (1997). Acceleration of the EM algorithm using quasi-Newton methods. *Journal of Royal Statistical Society, Series B*, 59(3):569–587.

Lachlan, G. J. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. John Wiley and Sons.

Lange, K. (1995). A quasi Newton acceleration of the em algorithm. *Statistica Sinica*, 5:1–18.

Lehn-Schiøler, T., Hansen, L. K., and Larsen, J. (2005). Mapping from speech to images using continuous state space models. In *Lecture Notes in Computer Science*, volume 3361, pages 136 – 145. Springer.

Meilijson, I. (1989). A fast improvement to the EM algorithm on its own terms. *Journal of Royal Statistical Society, Series B*, 51:127–138.

Moulines, E., Cardoso, J., and Cassiat, E. (1997). Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Proc. ICASSP*, volume 5, pages 3617–3620. The MIT Press.

Neal, R. M. and Hinton, G. E. (1998). *Learning in Graphical Models: A view of the EM algorithm that justifies incremental, sparse, and other variants*, pages 355–368. Dordrecht: Kluwer Academic Publishers.

Nielsen, H. B. (2000). UCMINF - an algorithm for unconstrained nonlinear optimization. Technical Report IMM-Rep-2000-19, Technical University of Denmark.

Olsson, R. K. and Hansen, L. K. (2004). Probabilistic blind deconvolution of non-stationary sources. In *12th European Signal Processing Conference*, pages 1697–1700.

Olsson, R. K. and Hansen, L. K. (2005). A harmonic excitation state-space

C.7 Neural Information Processing 2005: "Generative and Filtering Approaches for Overcomplete Representations"

(excluded due to copyright issues) (excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

(excluded due to copyright issues)

C.8 The Matrix Cookbook

The Matrix Cookbook

Kaare Brandt Petersen
Michael Syskind Pedersen

VERSION: JANUARY 5, 2005

What is this? These pages are a collection of facts (identities, approximations, inequalities, relations, ...) about matrices and matters relating to them. It is collected in this form for the convenience of anyone who wants a quick desktop reference .

Disclaimer: The identities, approximations and relations presented here were obviously not invented but collected, borrowed and copied from a large amount of sources. These sources include similar but shorter notes found on the internet and appendices in books - see the references for a full list.

Errors: Very likely there are errors, typos, and mistakes for which we apologize and would be grateful to receive corrections at kbp@imm.dtu.dk.

Its ongoing: The project of keeping a large repository of relations involving matrices is naturally ongoing and the version will be apparent from the date in the header.

Suggestions: Your suggestion for additional content or elaboration of some topics is most welcome at kbp@imm.dtu.dk.

Acknowledgements: We would like to thank the following for discussions, proofreading, extensive corrections and suggestions: Esben Hoegh-Rasmussen and Vasile Sima.

Keywords: Matrix algebra, matrix relations, matrix identities, derivative of determinant, derivative of inverse matrix, differentiate a matrix.

Contents

Notation and Nomenclature

$\mathbf{A}$	Matrix
$\mathbf{A}_{ij}$	Matrix indexed for some purpose
$\mathbf{A}_i$	Matrix indexed for some purpose
$\mathbf{A}^{ij}$	Matrix indexed for some purpose
$\mathbf{A}^n$	Matrix indexed for some purpose **or**
	The n.th power of a square matrix
$\mathbf{A}^{-1}$	The inverse matrix of the matrix $\mathbf{A}$
$\mathbf{A}^{+}$	The pseudo inverse matrix of the matrix $\mathbf{A}$
$\mathbf{A}^{1/2}$	The square root of a matrix (if unique), not elementwise
$(\mathbf{A})_{ij}$	The (i,j).th entry of the matrix $\mathbf{A}$
A_{ij}	The (i,j).th entry of the matrix $\mathbf{A}$
$\mathbf{a}$	Vector
$\mathbf{a}_i$	Vector indexed for some purpose
a_i	The i.th element of the vector $\mathbf{a}$
a	Scalar
$\Re z$	Real part of a scalar
$\Re \mathbf{z}$	Real part of a vector
$\Re \mathbf{Z}$	Real part of a matrix
$\Im z$	Imaginary part of a scalar
$\Im \mathbf{z}$	Imaginary part of a vector
$\Im \mathbf{Z}$	Imaginary part of a matrix
$\det(\mathbf{A})$	Determinant of $\mathbf{A}$
$\|\|\mathbf{A}\|\|$	Matrix norm (subscript if any denotes what norm)
$\mathbf{A}^T$	Transposed matrix
$\mathbf{A}^*$	Complex conjugated matrix
$\mathbf{A}^H$	Transposed and complex conjugated matrix
$\mathbf{A} \circ \mathbf{B}$	Hadamard (elementwise) product
$\mathbf{A} \otimes \mathbf{B}$	Kronecker product
$\mathbf{0}$	The null matrix. Zero in all entries.
$\mathbf{I}$	The identity matrix
$\mathbf{J}^{ij}$	The single-entry matrix, 1 at (i,j) and zero elsewhere
$\mathbf{\Sigma}$	A positive definite matrix
$\mathbf{\Lambda}$	A diagonal matrix

1 Basics

$$(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$$
$$(\mathbf{ABC}...)^{-1} = ...\mathbf{C}^{-1}\mathbf{B}^{-1}\mathbf{A}^{-1}$$
$$(\mathbf{A}^T)^{-1} = (\mathbf{A}^{-1})^T$$
$$(\mathbf{A}+\mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T$$
$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T$$
$$(\mathbf{ABC}...)^T = ...\mathbf{C}^T\mathbf{B}^T\mathbf{A}^T$$
$$(\mathbf{A}^H)^{-1} = (\mathbf{A}^{-1})^H$$
$$(\mathbf{A}+\mathbf{B})^H = \mathbf{A}^H + \mathbf{B}^H$$
$$(\mathbf{AB})^H = \mathbf{B}^H\mathbf{A}^H$$
$$(\mathbf{ABC}...)^H = ...\mathbf{C}^H\mathbf{B}^H\mathbf{A}^H$$

1.1 Trace and Determinants

$$\mathrm{Tr}(\mathbf{A}) = \sum_i \mathbf{A}_{ii} = \sum_i \lambda_i, \qquad \lambda_i = \mathrm{eig}(\mathbf{A})$$
$$\mathrm{Tr}(\mathbf{A}) = \mathrm{Tr}(\mathbf{A}^T)$$
$$\mathrm{Tr}(\mathbf{AB}) = \mathrm{Tr}(\mathbf{BA})$$
$$\mathrm{Tr}(\mathbf{A}+\mathbf{B}) = \mathrm{Tr}(\mathbf{A}) + \mathrm{Tr}(\mathbf{B})$$
$$\mathrm{Tr}(\mathbf{ABC}) = \mathrm{Tr}(\mathbf{BCA}) = \mathrm{Tr}(\mathbf{CAB})$$
$$\det(\mathbf{A}) = \prod_i \lambda_i \qquad \lambda_i = \mathrm{eig}(\mathbf{A})$$
$$\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B}), \qquad \text{if } \mathbf{A} \text{ and } \mathbf{B} \text{ are invertible}$$
$$\det(\mathbf{A}^{-1}) = \frac{1}{\det(\mathbf{A})}$$

1.2 The Special Case 2x2

Consider the matrix $\mathbf{A}$

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

Determinant and trace

$$\det(\mathbf{A}) = A_{11}A_{22} - A_{12}A_{21}$$

$$\mathrm{Tr}(\mathbf{A}) = A_{11} + A_{22}$$

Eigenvalues

$$\lambda^2 - \lambda \cdot \mathrm{Tr}(\mathbf{A}) + \det(\mathbf{A}) = 0$$

$$\lambda_1 = \frac{\mathrm{Tr}(\mathbf{A}) + \sqrt{\mathrm{Tr}(\mathbf{A})^2 - 4\det(\mathbf{A})}}{2} \qquad \lambda_2 = \frac{\mathrm{Tr}(\mathbf{A}) - \sqrt{\mathrm{Tr}(\mathbf{A})^2 - 4\det(\mathbf{A})}}{2}$$

$$\lambda_1 + \lambda_2 = \mathrm{Tr}(\mathbf{A}) \qquad \lambda_1\lambda_2 = \det(\mathbf{A})$$

Eigenvectors

$$\mathbf{v}_1 \propto \begin{bmatrix} A_{12} \\ \lambda_1 - A_{11} \end{bmatrix} \qquad \mathbf{v}_2 \propto \begin{bmatrix} A_{12} \\ \lambda_2 - A_{11} \end{bmatrix}$$

Inverse

$$\mathbf{A}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{bmatrix}$$

2 Derivatives

This section is covering differentiation of a number of expressions with respect to a matrix $\mathbf{X}$. Note that it is always assumed that $\mathbf{X}$ has *no special structure*, i.e. that the elements of $\mathbf{X}$ are independent (e.g. not symmetric, Toeplitz, positive definite). See section 2.5 for differentiation of structured matrices. The basic assumptions can be written in a formula as

$$\frac{\partial X_{kl}}{\partial X_{ij}} = \delta_{ik}\delta_{lj}$$

that is for e.g. vector forms,

$$\left[\frac{\partial \mathbf{x}}{\partial y}\right]_i = \frac{\partial x_i}{\partial y} \qquad \left[\frac{\partial x}{\partial \mathbf{y}}\right]_i = \frac{\partial x}{\partial y_i} \qquad \left[\frac{\partial \mathbf{x}}{\partial \mathbf{y}}\right]_{ij} = \frac{\partial x_i}{\partial y_j}$$

The following rules are general and very useful when deriving the differential of an expression ([10]):

$$\begin{aligned}
\partial \mathbf{A} &= 0 && (\mathbf{A} \text{ is a constant}) && (1)\\
\partial(\alpha \mathbf{X}) &= \alpha \partial \mathbf{X} && && (2)\\
\partial(\mathbf{X}+\mathbf{Y}) &= \partial \mathbf{X} + \partial \mathbf{Y} && && (3)\\
\partial(\mathrm{Tr}(\mathbf{X})) &= \mathrm{Tr}(\partial \mathbf{X}) && && (4)\\
\partial(\mathbf{X}\mathbf{Y}) &= (\partial \mathbf{X})\mathbf{Y} + \mathbf{X}(\partial \mathbf{Y}) && && (5)\\
\partial(\mathbf{X}\circ\mathbf{Y}) &= (\partial \mathbf{X})\circ\mathbf{Y} + \mathbf{X}\circ(\partial \mathbf{Y}) && && (6)\\
\partial(\mathbf{X}\otimes\mathbf{Y}) &= (\partial \mathbf{X})\otimes\mathbf{Y} + \mathbf{X}\otimes(\partial \mathbf{Y}) && && (7)\\
\partial(\mathbf{X}^{-1}) &= -\mathbf{X}^{-1}(\partial \mathbf{X})\mathbf{X}^{-1} && && (8)\\
\partial(\det(\mathbf{X})) &= \det(\mathbf{X})\mathrm{Tr}(\mathbf{X}^{-1}\partial \mathbf{X}) && && (9)\\
\partial(\ln(\det(\mathbf{X}))) &= \mathrm{Tr}(\mathbf{X}^{-1}\partial \mathbf{X}) && && (10)\\
\partial \mathbf{X}^T &= (\partial \mathbf{X})^T && && (11)\\
\partial \mathbf{X}^H &= (\partial \mathbf{X})^H && && (12)
\end{aligned}$$

2.1 Derivatives of a Determinant

2.1.1 General form

$$\frac{\partial \det(\mathbf{Y})}{\partial x} = \det(\mathbf{Y})\mathrm{Tr}\left[\mathbf{Y}^{-1}\frac{\partial \mathbf{Y}}{\partial x}\right]$$

2.1.2 Linear forms

$$\frac{\partial \det(\mathbf{X})}{\partial \mathbf{X}} = \det(\mathbf{X})(\mathbf{X}^{-1})^T$$

$$\frac{\partial \det(\mathbf{AXB})}{\partial \mathbf{X}} = \det(\mathbf{AXB})(\mathbf{X}^{-1})^T = \det(\mathbf{AXB})(\mathbf{X}^T)^{-1}$$

2.1.3 Square forms

If $\mathbf{X}$ is square and invertible, then

$$\frac{\partial \det(\mathbf{X}^T\mathbf{A}\mathbf{X})}{\partial \mathbf{X}} = 2\det(\mathbf{X}^T\mathbf{A}\mathbf{X})\mathbf{X}^{-T}$$

If $\mathbf{X}$ is not square but $\mathbf{A}$ is symmetric, then

$$\frac{\partial \det(\mathbf{X}^T\mathbf{A}\mathbf{X})}{\partial \mathbf{X}} = 2\det(\mathbf{X}^T\mathbf{A}\mathbf{X})\mathbf{A}\mathbf{X}(\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}$$

If $\mathbf{X}$ is not square and $\mathbf{A}$ is not symmetric, then

$$\frac{\partial \det(\mathbf{X}^T\mathbf{A}\mathbf{X})}{\partial \mathbf{X}} = \det(\mathbf{X}^T\mathbf{A}\mathbf{X})(\mathbf{A}\mathbf{X}(\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1} + \mathbf{A}^T\mathbf{X}(\mathbf{X}^T\mathbf{A}^T\mathbf{X})^{-1}) \tag{13}$$

2.1.4 Other nonlinear forms

Some special cases are (See [8])

$$\frac{\partial \ln \det(\mathbf{X}^T\mathbf{X})|}{\partial \mathbf{X}} = 2(\mathbf{X}^+)^T$$

$$\frac{\partial \ln \det(\mathbf{X}^T\mathbf{X})}{\partial \mathbf{X}^+} = -2\mathbf{X}^T$$

$$\frac{\partial \ln |\det(\mathbf{X})|}{\partial \mathbf{X}} = (\mathbf{X}^{-1})^T = (\mathbf{X}^T)^{-1}$$

$$\frac{\partial \det(\mathbf{X}^k)}{\partial \mathbf{X}} = k\det(\mathbf{X}^k)\mathbf{X}^{-T}$$

See [7].

2.2 Derivatives of an Inverse

From [15] we have the basic identity

$$\frac{\partial \mathbf{Y}^{-1}}{\partial x} = -\mathbf{Y}^{-1}\frac{\partial \mathbf{Y}}{\partial x}\mathbf{Y}^{-1}$$

from which it follows

$$\frac{\partial (\mathbf{X}^{-1})_{kl}}{\partial X_{ij}} = -(\mathbf{X}^{-1})_{ki}(\mathbf{X}^{-1})_{jl}$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}^{-1}\mathbf{b}}{\partial \mathbf{X}} = -\mathbf{X}^{-T}\mathbf{a}\mathbf{b}^T\mathbf{X}^{-T}$$

$$\frac{\partial \det(\mathbf{X}^{-1})}{\partial \mathbf{X}} = -\det(\mathbf{X}^{-1})(\mathbf{X}^{-1})^T$$

$$\frac{\partial \mathrm{Tr}(\mathbf{A}\mathbf{X}^{-1}\mathbf{B})}{\partial \mathbf{X}} = -(\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1})^T$$

2.3 Derivatives of Matrices, Vectors and Scalar Forms

2.3.1 First Order

$$\frac{\partial \mathbf{x}^T\mathbf{b}}{\partial \mathbf{x}} = \frac{\partial \mathbf{b}^T\mathbf{x}}{\partial \mathbf{x}} = \mathbf{b}$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}\mathbf{b}}{\partial \mathbf{X}} = \mathbf{a}\mathbf{b}^T$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}^T\mathbf{b}}{\partial \mathbf{X}} = \mathbf{b}\mathbf{a}^T$$

$$\frac{\partial \mathbf{a}^T\mathbf{X}\mathbf{a}}{\partial \mathbf{X}} = \frac{\partial \mathbf{a}^T\mathbf{X}^T\mathbf{a}}{\partial \mathbf{X}} = \mathbf{a}\mathbf{a}^T$$

$$\frac{\partial \mathbf{X}}{\partial X_{ij}} = \mathbf{J}^{ij}$$

$$\frac{\partial (\mathbf{X}\mathbf{A})_{ij}}{\partial X_{mn}} = \delta_{im}(\mathbf{A})_{nj} = (\mathbf{J}^{mn}\mathbf{A})_{ij}$$

$$\frac{\partial (\mathbf{X}^T\mathbf{A})_{ij}}{\partial X_{mn}} = \delta_{in}(\mathbf{A})_{mj} = (\mathbf{J}^{nm}\mathbf{A})_{ij}$$

2.3.2 Second Order

$$\frac{\partial}{\partial X_{ij}} \sum_{klmn} X_{kl}X_{mn} = 2\sum_{kl} X_{kl}$$

$$\frac{\partial \mathbf{b}^T\mathbf{X}^T\mathbf{X}\mathbf{c}}{\partial \mathbf{X}} = \mathbf{X}(\mathbf{b}\mathbf{c}^T + \mathbf{c}\mathbf{b}^T)$$

$$\frac{\partial (\mathbf{B}\mathbf{x}+\mathbf{b})^T\mathbf{C}(\mathbf{D}\mathbf{x}+\mathbf{d})}{\partial \mathbf{x}} = \mathbf{B}^T\mathbf{C}(\mathbf{D}\mathbf{x}+\mathbf{d}) + \mathbf{D}^T\mathbf{C}^T(\mathbf{B}\mathbf{x}+\mathbf{b})$$

$$\frac{\partial (\mathbf{X}^T\mathbf{B}\mathbf{X})_{kl}}{\partial X_{ij}} = \delta_{lj}(\mathbf{X}^T\mathbf{B})_{ki} + \delta_{kj}(\mathbf{B}\mathbf{X})_{il}$$

$$\frac{\partial (\mathbf{X}^T\mathbf{B}\mathbf{X})}{\partial X_{ij}} = \mathbf{X}^T\mathbf{B}\mathbf{J}^{ij} + \mathbf{J}^{ji}\mathbf{B}\mathbf{X} \qquad (\mathbf{J}^{ij})_{kl} = \delta_{ik}\delta_{jl}$$

See Sec 8.2 for useful properties of the Single-entry matrix $\mathbf{J}^{ij}$

$$\frac{\partial \mathbf{x}^T\mathbf{B}\mathbf{x}}{\partial \mathbf{x}} = (\mathbf{B} + \mathbf{B}^T)\mathbf{x}$$

$$\frac{\partial \mathbf{b}^T\mathbf{X}^T\mathbf{D}\mathbf{X}\mathbf{c}}{\partial \mathbf{X}} = \mathbf{D}^T\mathbf{X}\mathbf{b}\mathbf{c}^T + \mathbf{D}\mathbf{X}\mathbf{c}\mathbf{b}^T$$

$$\frac{\partial}{\partial \mathbf{X}}(\mathbf{X}\mathbf{b}+\mathbf{c})^T\mathbf{D}(\mathbf{X}\mathbf{b}+\mathbf{c}) = (\mathbf{D}+\mathbf{D}^T)(\mathbf{X}\mathbf{b}+\mathbf{c})\mathbf{b}^T$$

Assume $\mathbf{W}$ is symmetric, then

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{s}}(\mathbf{x}-\mathbf{A}\mathbf{s})^T\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s}) &= -2\mathbf{A}^T\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s})\\
\frac{\partial}{\partial \mathbf{x}}(\mathbf{x}-\mathbf{A}\mathbf{s})^T\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s}) &= -2\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s})\\
\frac{\partial}{\partial \mathbf{A}}(\mathbf{x}-\mathbf{A}\mathbf{s})^T\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s}) &= -2\mathbf{W}(\mathbf{x}-\mathbf{A}\mathbf{s})\mathbf{s}^T
\end{aligned}$$

2.3.3 Higher order and non-linear

$$\frac{\partial}{\partial \mathbf{X}}\mathbf{a}^T\mathbf{X}^n\mathbf{b} = \sum_{r=0}^{n-1}(\mathbf{X}^r)^T\mathbf{a}\mathbf{b}^T(\mathbf{X}^{n-1-r})^T \tag{14}$$

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{X}}\mathbf{a}^T(\mathbf{X}^n)^T\mathbf{X}^n\mathbf{b} &= \sum_{r=0}^{n-1}\Big[\mathbf{X}^{n-1-r}\mathbf{a}\mathbf{b}^T(\mathbf{X}^n)^T\mathbf{X}^r\\
&\qquad +(\mathbf{X}^r)^T\mathbf{X}^n\mathbf{a}\mathbf{b}^T(\mathbf{X}^{n-1-r})^T\Big]
\end{aligned} \tag{15}$$

See A.0.1 for a proof.

Assume $\mathbf{s}$ and $\mathbf{r}$ are functions of $\mathbf{x}$, i.e. $\mathbf{s} = \mathbf{s}(\mathbf{x}), \mathbf{r} = \mathbf{r}(\mathbf{x})$, and that $\mathbf{A}$ is a constant, then

$$\frac{\partial}{\partial \mathbf{x}}\mathbf{s}^T\mathbf{A}\mathbf{s} = \left[\frac{\partial \mathbf{s}}{\partial \mathbf{x}}\right]^T(\mathbf{A}+\mathbf{A}^T)\mathbf{s}$$

$$\frac{\partial}{\partial \mathbf{x}}\mathbf{s}^T\mathbf{A}\mathbf{r} = \left[\frac{\partial \mathbf{s}}{\partial \mathbf{x}}\right]^T\mathbf{A}\mathbf{s} + \left[\frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right]^T\mathbf{A}^T\mathbf{r}$$

2.3.4 Gradient and Hessian

Using the above we have for the gradient and the hessian

$$f = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}$$

$$\nabla_{\mathbf{x}} f = \frac{\partial f}{\partial \mathbf{x}} = (\mathbf{A}+\mathbf{A}^T)\mathbf{x} + \mathbf{b}$$

$$\frac{\partial^2 f}{\partial \mathbf{x}\partial \mathbf{x}^T} = \mathbf{A} + \mathbf{A}^T$$

2.4 Derivatives of Traces

2.4.1 First Order

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}) &= \mathbf{I} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}\mathbf{B}) &= \mathbf{B}^T \qquad (16) \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{B}\mathbf{X}\mathbf{C}) &= \mathbf{B}^T\mathbf{C}^T \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{B}\mathbf{X}^T\mathbf{C}) &= \mathbf{C}\mathbf{B} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}^T\mathbf{C}) &= \mathbf{C} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{B}\mathbf{X}^T) &= \mathbf{B}
\end{aligned}
$$

2.4.2 Second Order

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}^2) &= 2\mathbf{X} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}^2\mathbf{B}) &= (\mathbf{X}\mathbf{B}+\mathbf{B}\mathbf{X})^T \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}^T\mathbf{B}\mathbf{X}) &= \mathbf{B}\mathbf{X}+\mathbf{B}^T\mathbf{X} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}\mathbf{B}\mathbf{X}^T) &= \mathbf{X}\mathbf{B}^T+\mathbf{X}\mathbf{B} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{X}^T\mathbf{X}) &= 2\mathbf{X} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{B}\mathbf{X}\mathbf{X}^T) &= (\mathbf{B}+\mathbf{B}^T)\mathbf{X} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{B}^T\mathbf{X}^T\mathbf{C}\mathbf{X}\mathbf{B}) &= \mathbf{C}^T\mathbf{X}\mathbf{B}\mathbf{B}^T+\mathbf{C}\mathbf{X}\mathbf{B}\mathbf{B}^T \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}\left[\mathbf{X}^T\mathbf{B}\mathbf{X}\mathbf{C}\right] &= \mathbf{B}\mathbf{X}\mathbf{C}+\mathbf{B}^T\mathbf{X}\mathbf{C}^T \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}(\mathbf{A}\mathbf{X}\mathbf{B}\mathbf{X}^T\mathbf{C}) &= \mathbf{A}^T\mathbf{C}^T\mathbf{X}\mathbf{B}^T+\mathbf{C}\mathbf{A}\mathbf{X}\mathbf{B} \\
\frac{\partial}{\partial \mathbf{X}}\mathrm{Tr}\Big[(\mathbf{A}\mathbf{X}\mathbf{b}+\mathbf{c})(\mathbf{A}\mathbf{X}\mathbf{b}+\mathbf{c})^T\Big] &= 2\mathbf{A}^T(\mathbf{A}\mathbf{X}\mathbf{b}+\mathbf{c})\mathbf{b}^T
\end{aligned}
$$

See [7].

2.4.3 Higher Order

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}(\mathbf{X}^k) = k(\mathbf{X}^{k-1})^T$$

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}(\mathbf{A}\mathbf{X}^k) = \sum_{r=0}^{k-1} (\mathbf{X}^r \mathbf{A} \mathbf{X}^{k-r-1})^T$$

$$\begin{aligned} \frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}\left[\mathbf{B}^T\mathbf{X}^T\mathbf{C}\mathbf{X}\mathbf{X}^T\mathbf{C}\mathbf{X}\mathbf{B}\right] &= \mathbf{C}\mathbf{X}\mathbf{X}^T\mathbf{C}\mathbf{X}\mathbf{B}\mathbf{B}^T + \mathbf{C}^T\mathbf{X}\mathbf{B}\mathbf{B}^T\mathbf{X}^T\mathbf{C}^T\mathbf{X} \\ &\quad + \mathbf{C}\mathbf{X}\mathbf{B}\mathbf{B}^T\mathbf{X}^T\mathbf{C}\mathbf{X} + \mathbf{C}^T\mathbf{X}\mathbf{X}^T\mathbf{C}^T\mathbf{X}\mathbf{B}\mathbf{B}^T \end{aligned}$$

2.4.4 Other

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}(\mathbf{A}\mathbf{X}^{-1}\mathbf{B}) = -(\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1})^T = -\mathbf{X}^{-T}\mathbf{A}^T\mathbf{B}^T\mathbf{X}^{-T}$$

Assume $\mathbf{B}$ and $\mathbf{C}$ to be symmetric, then

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}\Big[(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1}\mathbf{A}\Big] = -(\mathbf{C}\mathbf{X}(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1})(\mathbf{A}+\mathbf{A}^T)(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1}$$

$$\begin{aligned} \frac{\partial}{\partial \mathbf{X}} \mathrm{Tr}\Big[(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{B}\mathbf{X})\Big] &= -2\mathbf{C}\mathbf{X}(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{B}\mathbf{X}(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1} \\ &\quad +2\mathbf{B}\mathbf{X}(\mathbf{X}^T\mathbf{C}\mathbf{X})^{-1} \end{aligned}$$

See [7].

2.5 Derivatives of Structured Matrices

Assume that the matrix $\mathbf{A}$ has some structure, i.e. is symmetric, toeplitz, etc. In that case the derivatives of the previous section does not apply in general. In stead, consider the following general rule for differentiating a scalar function $f(\mathbf{A})$

$$\frac{df}{dA_{ij}} = \sum_{kl} \frac{\partial f}{\partial A_{kl}} \frac{\partial A_{kl}}{\partial A_{ij}} = \mathrm{Tr}\left[\left[\frac{\partial f}{\partial \mathbf{A}}\right]^T \frac{\partial \mathbf{A}}{\partial A_{ij}}\right]$$

The matrix differentiated with respect to itself is in this document referred to as the *structure matrix* of $\mathbf{A}$ and is defined simply by

$$\frac{\partial \mathbf{A}}{\partial A_{ij}} = \mathbf{S}^{ij}$$

If $\mathbf{A}$ has no special structure we have simply $\mathbf{S}^{ij} = \mathbf{J}^{ij}$, that is, the structure matrix is simply the singleentry matrix. Many structures have a representation in singleentry matrices, see Sec. 8.2.7 for more examples of structure matrices.

2.5.1 Symmetric

If $\mathbf{A}$ is symmetric, then $\mathbf{S}^{ij} = \mathbf{J}^{ij} + \mathbf{J}^{ji} - \mathbf{J}^{ij}\mathbf{J}^{ij}$ and therefore

$$\frac{df}{d\mathbf{A}} = \left[\frac{\partial f}{\partial \mathbf{A}}\right] + \left[\frac{\partial f}{\partial \mathbf{A}}\right]^T - \operatorname{diag}\left[\frac{\partial f}{\partial \mathbf{A}}\right]$$

That is, e.g., ([5], [16]):

$$\frac{\partial \operatorname{Tr}(\mathbf{AX})}{\partial \mathbf{X}} = \mathbf{A} + \mathbf{A}^T - (\mathbf{A} \circ \mathbf{I}), \text{ see (20)} \tag{17}$$

$$\frac{\partial \det(\mathbf{X})}{\partial \mathbf{X}} = 2\mathbf{X} - (\mathbf{X} \circ \mathbf{I}) \tag{18}$$

$$\frac{\partial \ln \det(\mathbf{X})}{\partial \mathbf{X}} = 2\mathbf{X}^{-1} - (\mathbf{X}^{-1} \circ \mathbf{I}) \tag{19}$$

2.5.2 Diagonal

If $\mathbf{X}$ is diagonal, then ([10]):

$$\frac{\partial \operatorname{Tr}(\mathbf{AX})}{\partial \mathbf{X}} = \mathbf{A} \circ \mathbf{I} \tag{20}$$

3 Inverses

3.1 Exact Relations

3.1.1 The Woodbury identity

$$(\mathbf{A}+\mathbf{C}\mathbf{B}\mathbf{C}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}(\mathbf{B}^{-1}+\mathbf{C}^T\mathbf{A}^{-1}\mathbf{C})^{-1}\mathbf{C}^T\mathbf{A}^{-1}$$

If $\mathbf{P}, \mathbf{R}$ are positive definite, then (see [17])

$$(\mathbf{P}^{-1}+\mathbf{B}^T\mathbf{R}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{R}^{-1} = \mathbf{P}\mathbf{B}^T(\mathbf{B}\mathbf{P}\mathbf{B}^T+\mathbf{R})^{-1}$$

3.1.2 The Kailath Variant

$$(\mathbf{A}+\mathbf{B}\mathbf{C})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{I}+\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1}$$

See [4] page 153.

3.1.3 The Searle Set of Identities

The following set of identities, can be found in [13], page 151,

$$\begin{aligned}
(\mathbf{I}+\mathbf{A}^{-1})^{-1} &= \mathbf{A}(\mathbf{A}+\mathbf{I})^{-1} \\
(\mathbf{A}+\mathbf{B}\mathbf{B}^T)^{-1}\mathbf{B} &= \mathbf{A}^{-1}\mathbf{B}(\mathbf{I}+\mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})^{-1} \\
(\mathbf{A}^{-1}+\mathbf{B}^{-1})^{-1} &= \mathbf{A}(\mathbf{A}+\mathbf{B})^{-1}\mathbf{B} = \mathbf{B}(\mathbf{A}+\mathbf{B})^{-1}\mathbf{A} \\
\mathbf{A}-\mathbf{A}(\mathbf{A}+\mathbf{B})^{-1}\mathbf{A} &= \mathbf{B}-\mathbf{B}(\mathbf{A}+\mathbf{B})^{-1}\mathbf{B} \\
\mathbf{A}^{-1}+\mathbf{B}^{-1} &= \mathbf{A}^{-1}(\mathbf{A}+\mathbf{B})\mathbf{B}^{-1} \\
(\mathbf{I}+\mathbf{A}\mathbf{B})^{-1} &= \mathbf{I}-\mathbf{A}(\mathbf{I}+\mathbf{B}\mathbf{A})^{-1}\mathbf{B} \\
(\mathbf{I}+\mathbf{A}\mathbf{B})^{-1}\mathbf{A} &= \mathbf{A}(\mathbf{I}+\mathbf{A}\mathbf{B})^{-1}
\end{aligned}$$

3.2 Implication on Inverses

$$(\mathbf{A}+\mathbf{B})^{-1} = \mathbf{A}^{-1}+\mathbf{B}^{-1} \qquad \Rightarrow \qquad \mathbf{A}\mathbf{B}^{-1}\mathbf{A} = \mathbf{B}\mathbf{A}^{-1}\mathbf{B}$$

See [13].

3.2.1 A PosDef identity

Assume $\mathbf{P}, \mathbf{R}$ to be positive definite and invertible, then

$$(\mathbf{P}^{-1}+\mathbf{B}^T\mathbf{R}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{R}^{-1} = \mathbf{P}\mathbf{B}^T(\mathbf{B}\mathbf{P}\mathbf{B}^T+\mathbf{R})^{-1}$$

See [?].

3.3 Approximations

$$(\mathbf{I}+\mathbf{A})^{-1} = \mathbf{I}-\mathbf{A}+\mathbf{A}^2-\mathbf{A}^3+\ldots$$
$$\mathbf{A}-\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1}\mathbf{A} \cong \mathbf{I}-\mathbf{A}^{-1} \qquad \text{if } \mathbf{A} \text{ large and symmetric}$$

If σ^2 is small then

$$(\mathbf{Q}+\sigma^2\mathbf{M})^{-1} \cong \mathbf{Q}^{-1}-\sigma^2\mathbf{Q}^{-1}\mathbf{M}\mathbf{Q}^{-1}$$

3.4 Generalized Inverse

3.4.1 Definition

A generalized inverse matrix of the matrix $\mathbf{A}$ is any matrix $\mathbf{A}^-$ such that

$$\mathbf{A}\mathbf{A}^-\mathbf{A} = \mathbf{A}$$

The matrix $\mathbf{A}^-$ is not unique.

3.5 Pseudo Inverse

3.5.1 Definition

The pseudo inverse (or Moore-Penrose inverse) of a matrix $\mathbf{A}$ is the matrix $\mathbf{A}^+$ that fulfils

$$\begin{array}{ll} \text{I} & \mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A} \\ \text{II} & \mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+ \\ \text{III} & \mathbf{A}\mathbf{A}^+ \text{ symmetric} \\ \text{IV} & \mathbf{A}^+\mathbf{A} \text{ symmetric} \end{array}$$

The matrix $\mathbf{A}^+$ is unique and does always exist.

3.5.2 Properties

Assume $\mathbf{A}^+$ to be the pseudo-inverse of $\mathbf{A}$, then (See [3])

$$\begin{aligned} (\mathbf{A}^+)^+ &= \mathbf{A} \\ (\mathbf{A}^T)^+ &= (\mathbf{A}^+)^T \\ (c\mathbf{A})^+ &= (1/c)\mathbf{A}^+ \\ (\mathbf{A}^T\mathbf{A})^+ &= \mathbf{A}^+(\mathbf{A}^T)^+ \\ (\mathbf{A}\mathbf{A}^T)^+ &= (\mathbf{A}^T)^+\mathbf{A}^+ \end{aligned}$$

Assume $\mathbf{A}$ to have full rank, then

$$\begin{aligned} (\mathbf{A}\mathbf{A}^+)(\mathbf{A}\mathbf{A}^+) &= \mathbf{A}\mathbf{A}^+ \\ (\mathbf{A}^+\mathbf{A})(\mathbf{A}^+\mathbf{A}) &= \mathbf{A}^+\mathbf{A} \\ \text{Tr}(\mathbf{A}\mathbf{A}^+) &= \text{rank}(\mathbf{A}\mathbf{A}^+) \qquad \text{(See [14])} \\ \text{Tr}(\mathbf{A}^+\mathbf{A}) &= \text{rank}(\mathbf{A}^+\mathbf{A}) \qquad \text{(See [14])} \end{aligned}$$

3.5.3 Construction

Assume that $\mathbf{A}$ has full rank, then

$\mathbf{A}$ $n \times n$	Square	$\text{rank}(\mathbf{A}) = n$	$\Rightarrow$	$\mathbf{A}^+ = \mathbf{A}^{-1}$
$\mathbf{A}$ $n \times m$	Broad	$\text{rank}(\mathbf{A}) = n$	$\Rightarrow$	$\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$
$\mathbf{A}$ $n \times m$	Tall	$\text{rank}(\mathbf{A}) = m$	$\Rightarrow$	$\mathbf{A}^+ = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$

Assume $\mathbf{A}$ does not have full rank, i.e. $\mathbf{A}$ is $n \times m$ and $\text{rank}(\mathbf{A}) = r < \min(n, m)$. The pseudo inverse $\mathbf{A}^+$ can be constructed from the singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$, by

$$\mathbf{A}^+ = \mathbf{V}\mathbf{D}^+\mathbf{U}^T$$

A different way is this: There does always exists two matrices $\mathbf{C}$ $n \times r$ and $\mathbf{D}$ $r \times m$ of rank r, such that $\mathbf{A} = \mathbf{C}\mathbf{D}$. Using these matrices it holds that

$$\mathbf{A}^+ = \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T$$

See [3].

4 Complex Matrices

4.1 Complex Derivatives

In order to differentiate an expression $f(z)$ with respect to a complex z, the Cauchy-Riemann equations have to be satisfied ([7]):

$$\frac{df(z)}{dz} = \frac{\partial \Re(f(z))}{\partial \Re z} + i\frac{\partial \Im(f(z))}{\partial \Re z} \tag{21}$$

and

$$\frac{df(z)}{dz} = -i\frac{\partial \Re(f(z))}{\partial \Im z} + \frac{\partial \Im(f(z))}{\partial \Im z} \tag{22}$$

or in a more compact form:

$$\frac{\partial f(z)}{\partial \Im z} = i\frac{\partial f(z)}{\partial \Re z}. \tag{23}$$

A complex function that satisfies the Cauchy-Riemann equations for points in a region R is said yo be *analytic* in this region R. In general, expressions involving complex conjugate or conjugate transpose do not satisfy the Cauchy-Riemann equations. In order to avoid this problem, a more generalized definition of complex derivative is used ([12], [6]):

- Generalized Complex Derivative:

$$\frac{df(z)}{dz} = \frac{1}{2}\Big(\frac{\partial f(z)}{\partial \Re z} - i\frac{\partial f(z)}{\partial \Im z}\Big) \tag{24}$$

- Conjugate Complex Derivative

$$\frac{df(z)}{dz^*} = \frac{1}{2}\Big(\frac{\partial f(z)}{\partial \Re z} + i\frac{\partial f(z)}{\partial \Im z}\Big) \tag{25}$$

The Generalized Complex Derivative equals the normal derivative, when f is an analytic function. For a non-analytic function such as $f(z) = z^*$, the derivative equals zero. The Conjugate Complex Derivative equals zero, when f is an analytic function. The Conjugate Complex Derivative has e.g been used by [11] when deriving a complex gradient.
Notice:

$$\frac{df(z)}{dz} \neq \frac{\partial f(z)}{\partial \Re z} + i\frac{\partial f(z)}{\partial \Im z} \tag{26}$$

- Complex Gradient Vector: If f is a real function of a complex vector $\mathbf{z}$, then the complex gradient vector is given by ([9, p. 798])

$$\begin{aligned} \nabla f(\mathbf{z}) &= 2\frac{df(\mathbf{z})}{d\mathbf{z}^*} \\ &= \frac{\partial f(\mathbf{z})}{\partial \Re \mathbf{z}} + i\frac{\partial f(\mathbf{z})}{\partial \Im \mathbf{z}} \end{aligned} \tag{27}$$

- Complex Gradient Matrix: If f is a real function of a complex matrix $\mathbf{Z}$, then the complex gradient matrix is given by ([2])

$$\begin{aligned}\nabla f(\mathbf{Z}) &= 2\frac{df(\mathbf{Z})}{d\mathbf{Z}^*} \qquad (28)\\ &= \frac{\partial f(\mathbf{Z})}{\partial \Re\mathbf{Z}} + i\frac{\partial f(\mathbf{Z})}{\partial \Im\mathbf{Z}}\end{aligned}$$

  These expressions can be used for gradient descent algorithms.

4.1.1 The Chain Rule for complex numbers

The chain rule is a little more complicated when the function of a complex $u = f(x)$ is non-analytic. For a non-analytic function, the following chain rule can be applied ([?])

$$\begin{aligned}\frac{\partial g(u)}{\partial x} &= \frac{\partial g}{\partial u}\frac{\partial u}{\partial x} + \frac{\partial g}{\partial u^*}\frac{\partial u^*}{\partial x} \qquad (29)\\ &= \frac{\partial g}{\partial u}\frac{\partial u}{\partial x} + \Big(\frac{\partial g^*}{\partial u}\Big)^*\frac{\partial u^*}{\partial x}\end{aligned}$$

Notice, if the function is analytic, the second term reduces to zero, and the function is reduced to the normal well-known chain rule. For the matrix derivative of a scalar function $g(\mathbf{U})$, the chain rule can be written the following way:

$$\frac{\partial g(\mathbf{U})}{\partial \mathbf{X}} = \frac{\mathrm{Tr}((\frac{\partial g(\mathbf{U})}{\partial \mathbf{U}})^T \partial \mathbf{U})}{\partial \mathbf{X}} + \frac{\mathrm{Tr}((\frac{\partial g(\mathbf{U})}{\partial \mathbf{U}^*})^T \partial \mathbf{U}^*)}{\partial \mathbf{X}}. \qquad (30)$$

4.1.2 Complex Derivatives of Traces

If the derivatives involve complex numbers, the conjugate transpose is often involved. The most useful way to show complex derivative is to show the derivative with respect to the real and the imaginary part separately. An easy example is:

$$\begin{aligned}\frac{\partial \mathrm{Tr}(\mathbf{X}^*)}{\partial \Re\mathbf{X}} = \frac{\partial \mathrm{Tr}(\mathbf{X}^H)}{\partial \Re\mathbf{X}} &= \mathbf{I} \qquad (31)\\ i\frac{\partial \mathrm{Tr}(\mathbf{X}^*)}{\partial \Im\mathbf{X}} = i\frac{\partial \mathrm{Tr}(\mathbf{X}^H)}{\partial \Im\mathbf{X}} &= \mathbf{I} \qquad (32)\end{aligned}$$

Since the two results have the same sign, the conjugate complex derivative (25) should be used.

$$\begin{aligned}\frac{\partial \mathrm{Tr}(\mathbf{X})}{\partial \Re\mathbf{X}} = \frac{\partial \mathrm{Tr}(\mathbf{X}^T)}{\partial \Re\mathbf{X}} &= \mathbf{I} \qquad (33)\\ i\frac{\partial \mathrm{Tr}(\mathbf{X})}{\partial \Im\mathbf{X}} = i\frac{\partial \mathrm{Tr}(\mathbf{X}^T)}{\partial \Im\mathbf{X}} &= -\mathbf{I} \qquad (34)\end{aligned}$$

Here, the two results have different signs, the generalized complex derivative (24) should be used. Hereby, it can be seen that (**??**) holds even if $\mathbf{X}$ is a complex number.

$$\frac{\partial \mathrm{Tr}(\mathbf{A}\mathbf{X}^H)}{\partial \Re \mathbf{X}} = \mathbf{A} \tag{35}$$

$$i\frac{\partial \mathrm{Tr}(\mathbf{A}\mathbf{X}^H)}{\partial \Im \mathbf{X}} = \mathbf{A} \tag{36}$$

$$\frac{\partial \mathrm{Tr}(\mathbf{A}\mathbf{X}^*)}{\partial \Re \mathbf{X}} = \mathbf{A}^T \tag{37}$$

$$i\frac{\partial \mathrm{Tr}(\mathbf{A}\mathbf{X}^*)}{\partial \Im \mathbf{X}} = \mathbf{A}^T \tag{38}$$

$$\frac{\partial \mathrm{Tr}(\mathbf{X}\mathbf{X}^H)}{\partial \Re \mathbf{X}} = \frac{\partial \mathrm{Tr}(\mathbf{X}^H\mathbf{X})}{\partial \Re \mathbf{X}} = 2\Re \mathbf{X} \tag{39}$$

$$i\frac{\partial \mathrm{Tr}(\mathbf{X}\mathbf{X}^H)}{\partial \Im \mathbf{X}} = i\frac{\partial \mathrm{Tr}(\mathbf{X}^H\mathbf{X})}{\partial \Im \mathbf{X}} = i2\Im \mathbf{X} \tag{40}$$

By inserting (39) and (40) in (24) and (25), it can be seen that

$$\frac{\partial \mathrm{Tr}(\mathbf{X}\mathbf{X}^H)}{\partial \mathbf{X}} = \mathbf{X}^* \tag{41}$$

$$\frac{\partial \mathrm{Tr}(\mathbf{X}\mathbf{X}^H)}{\partial \mathbf{X}^*} = \mathbf{X} \tag{42}$$

Since the function $\mathrm{Tr}(\mathbf{X}\mathbf{X}^H)$ is a real function of the complex matrix $\mathbf{X}$, the complex gradient matrix (28) is given by

$$\nabla \mathrm{Tr}(\mathbf{X}\mathbf{X}^H) = 2\frac{\partial \mathrm{Tr}(\mathbf{X}\mathbf{X}^H)}{\partial \mathbf{X}^*} = 2\mathbf{X} \tag{43}$$

4.1.3 Complex Derivative Involving Determinants

Here, a calculation example is provided. The objective is to find the derivative of $\det(\mathbf{X}^H\mathbf{A}\mathbf{X})$ with respect to $\mathbf{X} \in \mathbb{C}^{m\times n}$. The derivative is found with respect to the real part and the imaginary part of $\mathbf{X}$, by use of (9) and (5), $\det(\mathbf{X}^H\mathbf{A}\mathbf{X})$ can be calculated as (see Sec. A.0.2 for details)

$$\begin{aligned}\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \mathbf{X}} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Re \mathbf{X}} - i\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Im \mathbf{X}}\Big) \\ &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big((\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A}\big)^T \end{aligned}\tag{44}$$

and the complex conjugate derivative yields

$$\begin{aligned}\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \mathbf{X}^*} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Re \mathbf{X}} + i\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Im \mathbf{X}}\Big) \\ &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1} \end{aligned}\tag{45}$$

5 Decompositions

5.1 Eigenvalues and Eigenvectors

5.1.1 Definition

The eigenvectors $\mathbf{v}$ and eigenvalues λ are the ones satisfying

$$\mathbf{A}\mathbf{v}_i = \lambda_i \mathbf{v}_i$$

$$\mathbf{AV} = \mathbf{VD}, \qquad (\mathbf{D})_{ij} = \delta_{ij}\lambda_i$$

where the columns of $\mathbf{V}$ are the vectors $\mathbf{v}_i$

5.1.2 General Properties

$$\begin{aligned} \text{eig}(\mathbf{AB}) &= \text{eig}(\mathbf{BA}) \\ \mathbf{A} \text{ is } n \times m &\Rightarrow \text{At most } \min(n,m) \text{ distinct } \lambda_i \\ \text{rank}(\mathbf{A}) = r &\Rightarrow \text{At most } r \text{ non-zero } \lambda_i \end{aligned}$$

5.1.3 Symmetric

Assume $\mathbf{A}$ is symmetric, then

$$\begin{aligned} \mathbf{V}\mathbf{V}^T &= \mathbf{I} \quad \text{(i.e. } \mathbf{V} \text{ is orthogonal)} \\ \lambda_i &\in \mathbb{R} \quad \text{(i.e. } \lambda_i \text{ is real)} \\ \text{Tr}(\mathbf{A}^p) &= \textstyle\sum_i \lambda_i^p \\ \text{eig}(\mathbf{I} + c\mathbf{A}) &= 1 + c\lambda_i \\ \text{eig}(\mathbf{A}^{-1}) &= \lambda_i^{-1} \end{aligned}$$

5.2 Singular Value Decomposition

Any $n \times m$ matrix $\mathbf{A}$ can be written as

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

where

$$\begin{array}{rcll} \mathbf{U} &=& \text{eigenvectors of } \mathbf{A}\mathbf{A}^T & n \times n \\ \mathbf{D} &=& \sqrt{\text{diag}(\text{eig}(\mathbf{A}\mathbf{A}^T))} & n \times m \\ \mathbf{V} &=& \text{eigenvectors of } \mathbf{A}^T\mathbf{A} & m \times m \end{array}$$

5.2.1 Symmetric Square decomposed into squares

Assume $\mathbf{A}$ to be $n \times n$ and symmetric. Then

$$\left[\ \mathbf{A}\ \right] = \left[\ \mathbf{V}\ \right]\left[\ \mathbf{D}\ \right]\left[\ \mathbf{V}^T\ \right]$$

where $\mathbf{D}$ is diagonal with the eigenvalues of $\mathbf{A}$ and $\mathbf{V}$ is orthogonal and the eigenvectors of $\mathbf{A}$.

5.2.2 Square decomposed into squares

Assume $\mathbf{A}$ to be $n \times n$. Then

$$[\ \mathbf{A}\ ] = [\ \mathbf{V}\ ][\ \mathbf{D}\ ][\ \mathbf{U}^T\ ]$$

where $\mathbf{D}$ is diagonal with the square root of the eigenvalues of $\mathbf{A}\mathbf{A}^T$, $\mathbf{V}$ is the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{U}^T$ is the eigenvectors of $\mathbf{A}^T\mathbf{A}$.

5.2.3 Square decomposed into rectangular

Assume $\mathbf{V}_*\mathbf{D}_*\mathbf{U}_*^T = \mathbf{0}$ then we can expand the SVD of $\mathbf{A}$ into

$$[\ \mathbf{A}\ ] = [\ \mathbf{V} \mid \mathbf{V}_*\ ]\left[\begin{array}{c|c}\mathbf{D} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{D}_*\end{array}\right]\left[\begin{array}{c}\mathbf{U}^T \\ \hline \mathbf{U}_*^T\end{array}\right]$$

where the SVD of $\mathbf{A}$ is $\mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{U}^T$.

5.2.4 Rectangular decomposition I

Assume $\mathbf{A}$ is $n \times m$

$$[\quad \mathbf{A}\quad] = [\ \mathbf{V}\ ][\ \mathbf{D}\ ][\quad \mathbf{U}^T\quad]$$

where $\mathbf{D}$ is diagonal with the square root of the eigenvalues of $\mathbf{A}\mathbf{A}^T$, $\mathbf{V}$ is the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{U}^T$ is the eigenvectors of $\mathbf{A}^T\mathbf{A}$.

5.2.5 Rectangular decomposition II

Assume $\mathbf{A}$ is $n \times m$

$$[\quad \mathbf{A}\quad] = [\quad \mathbf{V}\quad]\left[\quad \mathbf{D}\quad \right]\left[\quad \mathbf{U}^T\quad \right]$$

5.2.6 Rectangular decomposition III

Assume $\mathbf{A}$ is $n \times m$

$$[\quad \mathbf{A}\quad] = [\ \mathbf{V}\ ][\quad \mathbf{D}\quad]\left[\quad \mathbf{U}^T\quad \right]$$

where $\mathbf{D}$ is diagonal with the square root of the eigenvalues of $\mathbf{A}\mathbf{A}^T$, $\mathbf{V}$ is the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{U}^T$ is the eigenvectors of $\mathbf{A}^T\mathbf{A}$.

5.3 Triangular Decomposition

5.3.1 Cholesky-decomposition

Assume $\mathbf{A}$ is positive definite, then

$$\mathbf{A} = \mathbf{B}^T\mathbf{B}$$

where $\mathbf{B}$ is a unique upper triangular matrix.

6 General Statistics and Probability

6.1 Moments of any distribution

6.1.1 Mean and covariance of linear forms

Assume $\mathbf{X}$ and $\mathbf{x}$ to be a matrix and a vector of random variables. Then

$$E[\mathbf{AXB}+\mathbf{C}] = \mathbf{A}E[\mathbf{X}]\mathbf{B}+\mathbf{C}$$

$$\mathrm{Var}[\mathbf{Ax}] = \mathbf{A}\mathrm{Var}[\mathbf{x}]\mathbf{A}^T$$

$$\mathrm{Cov}[\mathbf{Ax},\mathbf{By}] = \mathbf{A}\mathrm{Cov}[\mathbf{x},\mathbf{y}]\mathbf{B}^T$$

See [14].

6.1.2 Mean and Variance of Square Forms

Assume $\mathbf{A}$ is symmetric, $\mathbf{c} = E[\mathbf{x}]$ and $\mathbf{\Sigma} = \mathrm{Var}[\mathbf{x}]$. Assume also that all coordinates x_i are independent, have the same central moments $\mu_1, \mu_2, \mu_3, \mu_4$ and denote $\mathbf{a} = \mathrm{diag}(\mathbf{A})$. Then

$$E[\mathbf{x}^T\mathbf{Ax}] = \mathrm{Tr}(\mathbf{A\Sigma}) + \mathbf{c}^T\mathbf{Ac}$$

$$\mathrm{Var}[\mathbf{x}^T\mathbf{Ax}] = 2\mu_2^2\mathrm{Tr}(\mathbf{A}^2) + 4\mu_2\mathbf{c}^T\mathbf{A}^2\mathbf{c} + 4\mu_3\mathbf{c}^T\mathbf{Aa} + (\mu_4 - 3\mu_2^2)\mathbf{a}^T\mathbf{a}$$

See [14]

6.2 Expectations

Assume $\mathbf{x}$ to be a stochastic vector with mean $\mathbf{m}$, covariance $\mathbf{M}$ and central moments $\mathbf{v}_r = E[(\mathbf{x}-\mathbf{m})^r]$.

6.2.1 Linear Forms

$$\begin{aligned} E[\mathbf{Ax}+\mathbf{b}] &= \mathbf{Am}+\mathbf{b} \\ E[\mathbf{Ax}] &= \mathbf{Am} \\ E[\mathbf{x}+\mathbf{b}] &= \mathbf{m}+\mathbf{b} \end{aligned}$$

6.2.2 Quadratic Forms

$$\begin{aligned} E[(\mathbf{Ax}+\mathbf{a})(\mathbf{Bx}+\mathbf{b})^T] &= \mathbf{AMB}^T + (\mathbf{Am}+\mathbf{a})(\mathbf{Bm}+\mathbf{b})^T \\ E[\mathbf{xx}^T] &= \mathbf{M}+\mathbf{mm}^T \\ E[\mathbf{xa}^T\mathbf{x}] &= (\mathbf{M}+\mathbf{mm}^T)\mathbf{a} \\ E[\mathbf{x}^T\mathbf{ax}^T] &= \mathbf{a}^T(\mathbf{M}+\mathbf{mm}^T) \\ E[(\mathbf{Ax})(\mathbf{Ax})^T] &= \mathbf{A}(\mathbf{M}+\mathbf{mm}^T)\mathbf{A}^T \\ E[(\mathbf{x}+\mathbf{a})(\mathbf{x}+\mathbf{a})^T] &= \mathbf{M}+(\mathbf{m}+\mathbf{a})(\mathbf{m}+\mathbf{a})^T \end{aligned}$$

$$\begin{aligned}
E[(\mathbf{Ax}+\mathbf{a})^T(\mathbf{Bx}+\mathbf{b})] &= \mathrm{Tr}(\mathbf{AMB}^T)+(\mathbf{Am}+\mathbf{a})^T(\mathbf{Bm}+\mathbf{b}) \\
E[\mathbf{x}^T\mathbf{x}] &= \mathrm{Tr}(\mathbf{M})+\mathbf{m}^T\mathbf{m} \\
E[\mathbf{x}^T\mathbf{Ax}] &= \mathrm{Tr}(\mathbf{AM})+\mathbf{m}^T\mathbf{Am} \\
E[(\mathbf{Ax})^T(\mathbf{Ax})] &= \mathrm{Tr}(\mathbf{AMA}^T)+(\mathbf{Am})^T(\mathbf{Am}) \\
E[(\mathbf{x}+\mathbf{a})^T(\mathbf{x}+\mathbf{a})] &= \mathrm{Tr}(\mathbf{M})+(\mathbf{m}+\mathbf{a})^T(\mathbf{m}+\mathbf{a})
\end{aligned}$$

See [7].

6.2.3 Cubic Forms

Assume $\mathbf{x}$ to be independent, then

$$\begin{aligned}
E[(\mathbf{Ax}+\mathbf{a})(\mathbf{Bx}+\mathbf{b})^T(\mathbf{Cx}+\mathbf{c})] &= \mathbf{A}\mathrm{diag}(\mathbf{B}^T\mathbf{C})\mathbf{v}_3 \\
&\quad +\mathrm{Tr}(\mathbf{BMC}^T)(\mathbf{Am}+\mathbf{a}) \\
&\quad +\mathbf{AMC}^T(\mathbf{Bm}+\mathbf{b}) \\
&\quad +(\mathbf{AMB}^T+(\mathbf{Am}+\mathbf{a})(\mathbf{Bm}+\mathbf{b})^T)(\mathbf{Cm}+\mathbf{c}) \\
E[\mathbf{xx}^T\mathbf{x}] &= \mathbf{v}_3+2\mathbf{Mm}+(\mathrm{Tr}(\mathbf{M})+\mathbf{m}^T\mathbf{m})\mathbf{m} \\
E[(\mathbf{Ax}+\mathbf{a})(\mathbf{Ax}+\mathbf{a})^T(\mathbf{Ax}+\mathbf{a})] &= \mathbf{A}\mathrm{diag}(\mathbf{A}^T\mathbf{A})\mathbf{v}_3 \\
&\quad +[2\mathbf{AMA}^T+(\mathbf{Ax}+\mathbf{a})(\mathbf{Ax}+\mathbf{a})^T](\mathbf{Am}+\mathbf{a}) \\
&\quad +\mathrm{Tr}(\mathbf{AMA}^T)(\mathbf{Am}+\mathbf{a})
\end{aligned}$$

$$\begin{aligned}
E[(\mathbf{Ax}+\mathbf{a})\mathbf{b}^T(\mathbf{Cx}+\mathbf{c})(\mathbf{Dx}+\mathbf{d})^T] &= (\mathbf{Ax}+\mathbf{a})\mathbf{b}^T(\mathbf{CMD}^T+(\mathbf{Cm}+\mathbf{c})(\mathbf{Dm}+\mathbf{d})^T) \\
&\quad +(\mathbf{AMC}^T+(\mathbf{Am}+\mathbf{a})(\mathbf{Cm}+\mathbf{c})^T)\mathbf{b}(\mathbf{Dm}+\mathbf{d})^T \\
&\quad +\mathbf{b}^T(\mathbf{Cm}+\mathbf{c})(\mathbf{AMD}^T-(\mathbf{Am}+\mathbf{a})(\mathbf{Dm}+\mathbf{d})^T)
\end{aligned}$$

See [7].

7 Gaussians

7.1 One Dimensional

7.1.1 Density and Normalization

The density is

$$p(s) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(s-\mu)^2}{2\sigma^2}\right)$$

Normalization integrals

$$\int e^{-\frac{(s-\mu)^2}{2\sigma^2}} ds = \sqrt{2\pi\sigma^2}$$

$$\int e^{-(ax^2+bx+c)} dx = \sqrt{\frac{\pi}{a}} \exp\left[\frac{b^2-4ac}{4a}\right]$$

$$\int e^{c_2x^2+c_1x+c_0} dx = \sqrt{\frac{\pi}{-c_2}} \exp\left[\frac{c_1^2-4c_2c_0}{-4c_2}\right]$$

7.1.2 Completing the Squares

$$c_2x^2 + c_1x + c_0 = -a(x-b)^2 + w$$

$$-a = c_2 \qquad b = \frac{1}{2}\frac{c_1}{c_2} \qquad w = \frac{1}{4}\frac{c_1^2}{c_2} + c_0$$

or

$$c_2x^2 + c_1x + c_0 = -\frac{1}{2\sigma^2}(x-\mu)^2 + d$$

$$\mu = \frac{-c_1}{2c_2} \qquad \sigma^2 = \frac{-1}{2c_2} \qquad d = c_0 - \frac{c_1^2}{4c_2}$$

7.1.3 Moments

If the density is expressed by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(s-\mu)^2}{2\sigma^2}\right] \quad \text{or} \quad p(x) = C\exp(c_2x^2 + c_1x)$$

then the first few basic moments are

$$\begin{aligned}
\langle x \rangle &= \mu &&= \tfrac{-c_1}{2c_2} \\
\langle x^2 \rangle &= \sigma^2 + \mu^2 &&= \tfrac{-1}{2c_2} + \left(\tfrac{-c_1}{2c_2}\right)^2 \\
\langle x^3 \rangle &= 3\sigma^2\mu + \mu^3 &&= \tfrac{c_1}{(2c_2)^2}\left[3 - \tfrac{c_1^2}{2c_2}\right] \\
\langle x^4 \rangle &= \mu^4 + 6\mu^2\sigma^2 + 3\sigma^4 &&= \left(\tfrac{c_1}{2c_2}\right)^4 + 6\left(\tfrac{c_1}{2c_2}\right)^2\left(\tfrac{-1}{2c_2}\right) + 3\left(\tfrac{1}{2c_2}\right)^2
\end{aligned}$$

and the central moments are

$$\begin{array}{rclcl} \langle(x-\mu)\rangle &=& 0 &=& 0 \\ \langle(x-\mu)^2\rangle &=& \sigma^2 &=& \left[\frac{-1}{2c_2}\right] \\ \langle(x-\mu)^3\rangle &=& 0 &=& 0 \\ \langle(x-\mu)^4\rangle &=& 3\sigma^4 &=& 3\left[\frac{1}{2c_2}\right]^2 \end{array}$$

A kind of pseudo-moments (un-normalized integrals) can easily be derived as

$$\int \exp(c_2x^2 + c_1x)x^n dx = Z\langle x^n\rangle = \sqrt{\frac{\pi}{-c_2}} \exp\left[\frac{c_1^2}{-4c_2}\right] \langle x^n\rangle$$

From the un-centralized moments one can derive other entities like

$$\begin{array}{rclcl} \langle x^2\rangle - \langle x\rangle^2 &=& \sigma^2 &=& \frac{-1}{2c_2} \\ \langle x^3\rangle - \langle x^2\rangle\langle x\rangle &=& 2\sigma^2\mu &=& \frac{2c_1}{(2c_2)^2} \\ \langle x^4\rangle - \langle x^2\rangle^2 &=& 2\sigma^4 + 4\mu^2\sigma^2 &=& \frac{2}{(2c_2)^2}\left[1 - 4\frac{c_1^2}{2c_2}\right] \end{array}$$

7.2 Basics

7.2.1 Density and normalization

The density of $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ is

$$p(\mathbf{x}) = \frac{1}{\sqrt{\det(2\pi\mathbf{\Sigma})}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{m})\right]$$

Note that if $\mathbf{x}$ is d-dimensional, then $\det(2\pi\mathbf{\Sigma}) = (2\pi)^d \det(\mathbf{\Sigma})$.
Integration and normalization

$$\int \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{m})\right] d\mathbf{x} = \sqrt{\det(2\pi\mathbf{\Sigma})}$$

$$\int \exp\left[-\frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}\right] d\mathbf{x} = \sqrt{\det(2\pi\mathbf{A}^{-1})} \exp\left[\frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}\mathbf{b}\right]$$

$$\int \exp\left[-\frac{1}{2}\mathrm{Tr}(\mathbf{S}^T\mathbf{A}\mathbf{S}) + \mathrm{Tr}(\mathbf{B}^T\mathbf{S})\right] d\mathbf{S} = \sqrt{\det(2\pi\mathbf{A}^{-1})} \exp\left[\frac{1}{2}\mathrm{Tr}(\mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})\right]$$

The derivatives of the density are

$$\frac{\partial p(\mathbf{x})}{\partial \mathbf{x}} = -p(\mathbf{x})\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{m})$$

$$\frac{\partial^2 p}{\partial\mathbf{x}\partial\mathbf{x}^T} = p(\mathbf{x})\left(\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\right)$$

7.2.2 Linear combination

Assume $\mathbf{x} \sim \mathcal{N}(\mathbf{m}_x, \mathbf{\Sigma}_x)$ and $\mathbf{y} \sim \mathcal{N}(\mathbf{m}_y, \mathbf{\Sigma}_y)$ then

$$\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} + \mathbf{c} \sim \mathcal{N}(\mathbf{A}\mathbf{m}_x + \mathbf{B}\mathbf{m}_y + \mathbf{c}, \mathbf{A}\mathbf{\Sigma}_x\mathbf{A}^T + \mathbf{B}\mathbf{\Sigma}_y\mathbf{B}^T)$$

7.2.3 Rearranging Means

$$\mathcal{N}_{\mathbf{Ax}}[\mathbf{m},\mathbf{\Sigma}] = \frac{\sqrt{\det(2\pi(\mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A})^{-1})}}{\sqrt{\det(2\pi\mathbf{\Sigma})}}\mathcal{N}_{\mathbf{x}}[\mathbf{A}^{-1}\mathbf{m},(\mathbf{A}^T\mathbf{\Sigma}^{-1}\mathbf{A})^{-1}]$$

7.2.4 Rearranging into squared form

If $\mathbf{A}$ is symmetric, then

$$-\frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x}+\mathbf{b}^T\mathbf{x} = -\frac{1}{2}(\mathbf{x}-\mathbf{A}^{-1}\mathbf{b})^T\mathbf{A}(\mathbf{x}-\mathbf{A}^{-1}\mathbf{b})+\frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}\mathbf{b}$$

$$-\frac{1}{2}\mathrm{Tr}(\mathbf{X}^T\mathbf{A}\mathbf{X})+\mathrm{Tr}(\mathbf{B}^T\mathbf{X}) = -\frac{1}{2}\mathrm{Tr}[(\mathbf{X}-\mathbf{A}^{-1}\mathbf{B})^T\mathbf{A}(\mathbf{X}-\mathbf{A}^{-1}\mathbf{B})]+\frac{1}{2}\mathrm{Tr}(\mathbf{B}^T\mathbf{A}^{-1}\mathbf{B})$$

7.2.5 Sum of two squared forms

In vector formulation (assuming $\mathbf{\Sigma}_1,\mathbf{\Sigma}_2$ are symmetric)

$$\begin{aligned}
& -\frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\mathbf{\Sigma}_1^{-1}(\mathbf{x}-\mathbf{m}_1) \\
& -\frac{1}{2}(\mathbf{x}-\mathbf{m}_2)^T\mathbf{\Sigma}_2^{-1}(\mathbf{x}-\mathbf{m}_2) \\
= & -\frac{1}{2}(\mathbf{x}-\mathbf{m}_c)^T\mathbf{\Sigma}_c^{-1}(\mathbf{x}-\mathbf{m}_c)+C
\end{aligned}$$

$$\begin{aligned}
\mathbf{\Sigma}_c^{-1} &= \mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1} \\
\mathbf{m}_c &= (\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}(\mathbf{\Sigma}_1^{-1}\mathbf{m}_1+\mathbf{\Sigma}_2^{-1}\mathbf{m}_2) \\
C &= \frac{1}{2}(\mathbf{m}_1^T\mathbf{\Sigma}_1^{-1}+\mathbf{m}_2^T\mathbf{\Sigma}_2^{-1})(\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}(\mathbf{\Sigma}_1^{-1}\mathbf{m}_1+\mathbf{\Sigma}_2^{-1}\mathbf{m}_2) \\
& \quad -\frac{1}{2}\Big(\mathbf{m}_1^T\mathbf{\Sigma}_1^{-1}\mathbf{m}_1+\mathbf{m}_2^T\mathbf{\Sigma}_2^{-1}\mathbf{m}_2\Big)
\end{aligned}$$

In a trace formulation (assuming $\mathbf{\Sigma}_1,\mathbf{\Sigma}_2$ are symmetric)

$$\begin{aligned}
& -\frac{1}{2}\mathrm{Tr}((\mathbf{X}-\mathbf{M}_1)^T\mathbf{\Sigma}_1^{-1}(\mathbf{X}-\mathbf{M}_1)) \\
& -\frac{1}{2}\mathrm{Tr}((\mathbf{X}-\mathbf{M}_2)^T\mathbf{\Sigma}_2^{-1}(\mathbf{X}-\mathbf{M}_2)) \\
= & -\frac{1}{2}\mathrm{Tr}[(\mathbf{X}-\mathbf{M}_c)^T\mathbf{\Sigma}_c^{-1}(\mathbf{X}-\mathbf{M}_c)]+C
\end{aligned}$$

$$\begin{aligned}
\mathbf{\Sigma}_c^{-1} &= \mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1} \\
\mathbf{M}_c &= (\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}(\mathbf{\Sigma}_1^{-1}\mathbf{M}_1+\mathbf{\Sigma}_2^{-1}\mathbf{M}_2) \\
C &= \frac{1}{2}\mathrm{Tr}\Big[(\mathbf{\Sigma}_1^{-1}\mathbf{M}_1+\mathbf{\Sigma}_2^{-1}\mathbf{M}_2)^T(\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}(\mathbf{\Sigma}_1^{-1}\mathbf{M}_1+\mathbf{\Sigma}_2^{-1}\mathbf{M}_2)\Big] \\
& \quad -\frac{1}{2}\mathrm{Tr}(\mathbf{M}_1^T\mathbf{\Sigma}_1^{-1}\mathbf{M}_1+\mathbf{M}_2^T\mathbf{\Sigma}_2^{-1}\mathbf{M}_2)
\end{aligned}$$

7.2.6 Product of gaussian densities

Let $\mathcal{N}_{\mathbf{x}}(\mathbf{m},\mathbf{\Sigma})$ denote a density of $\mathbf{x}$, then

$$\mathcal{N}_{\mathbf{x}}(\mathbf{m}_1,\mathbf{\Sigma}_1)\cdot\mathcal{N}_{\mathbf{x}}(\mathbf{m}_2,\mathbf{\Sigma}_2)=c_c\mathcal{N}_{\mathbf{x}}(\mathbf{m}_c,\mathbf{\Sigma}_c)$$

$$\begin{aligned}
c_c &= \mathcal{N}_{\mathbf{m}_1}(\mathbf{m}_2,(\mathbf{\Sigma}_1+\mathbf{\Sigma}_2))\\
&= \frac{1}{\sqrt{\det(2\pi(\mathbf{\Sigma}_1+\mathbf{\Sigma}_2))}}\exp\left[-\frac{1}{2}(\mathbf{m}_1-\mathbf{m}_2)^T(\mathbf{\Sigma}_1+\mathbf{\Sigma}_2)^{-1}(\mathbf{m}_1-\mathbf{m}_2)\right]\\
\mathbf{m}_c &= (\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}(\mathbf{\Sigma}_1^{-1}\mathbf{m}_1+\mathbf{\Sigma}_2^{-1}\mathbf{m}_2)\\
\mathbf{\Sigma}_c &= (\mathbf{\Sigma}_1^{-1}+\mathbf{\Sigma}_2^{-1})^{-1}
\end{aligned}$$

but note that the product is not normalized as a density of $\mathbf{x}$.

7.3 Moments

7.3.1 Mean and covariance of linear forms

First and second moments. Assume $\mathbf{x}\sim\mathcal{N}(\mathbf{m},\mathbf{\Sigma})$

$$E(\mathbf{x})=\mathbf{m}$$

$$\mathrm{Cov}(\mathbf{x},\mathbf{x})=\mathrm{Var}(\mathbf{x})=\mathbf{\Sigma}=E(\mathbf{x}\mathbf{x}^T)-E(\mathbf{x})E(\mathbf{x}^T)=E(\mathbf{x}\mathbf{x}^T)-\mathbf{m}\mathbf{m}^T$$

As for any other distribution is holds for gaussians that

$$E[\mathbf{A}\mathbf{x}]=\mathbf{A}E[\mathbf{x}]$$

$$\mathrm{Var}[\mathbf{A}\mathbf{x}]=\mathbf{A}\mathrm{Var}[\mathbf{x}]\mathbf{A}^T$$

$$\mathrm{Cov}[\mathbf{A}\mathbf{x},\mathbf{B}\mathbf{y}]=\mathbf{A}\mathrm{Cov}[\mathbf{x},\mathbf{y}]\mathbf{B}^T$$

7.3.2 Mean and variance of square forms

Mean and variance of square forms: Assume $\mathbf{x}\sim\mathcal{N}(\mathbf{m},\mathbf{\Sigma})$

$$\begin{aligned}
E(\mathbf{x}\mathbf{x}^T) &= \mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T\\
E[\mathbf{x}^T\mathbf{A}\mathbf{x}] &= \mathrm{Tr}(\mathbf{A}\mathbf{\Sigma})+\mathbf{m}^T\mathbf{A}\mathbf{m}\\
\mathrm{Var}(\mathbf{x}^T\mathbf{A}\mathbf{x}) &= 2\sigma^4\mathrm{Tr}(\mathbf{A}^2)+4\sigma^2\mathbf{m}^T\mathbf{A}^2\mathbf{m}\\
E[(\mathbf{x}-\mathbf{m}')^T\mathbf{A}(\mathbf{x}-\mathbf{m}')] &= (\mathbf{m}-\mathbf{m}')^T\mathbf{A}(\mathbf{m}-\mathbf{m}')+\mathrm{Tr}(\mathbf{A}\mathbf{\Sigma})
\end{aligned}$$

Assume $\mathbf{x}\sim\mathcal{N}(\mathbf{0},\sigma^2\mathbf{I})$ and $\mathbf{A}$ and $\mathbf{B}$ to be symmetric, then

$$\mathrm{Cov}(\mathbf{x}^T\mathbf{A}\mathbf{x},\mathbf{x}^T\mathbf{B}\mathbf{x})=2\sigma^4\mathrm{Tr}(\mathbf{A}\mathbf{B})$$

7.3.3 Cubic forms

$$E[\mathbf{x}\mathbf{b}^T\mathbf{x}\mathbf{x}^T] = \mathbf{m}\mathbf{b}^T(\mathbf{M}+\mathbf{m}\mathbf{m}^T) + (\mathbf{M}+\mathbf{m}\mathbf{m}^T)\mathbf{b}\mathbf{m}^T + \mathbf{b}^T\mathbf{m}(\mathbf{M}-\mathbf{m}\mathbf{m}^T)$$

7.3.4 Mean of Quartic Forms

$$\begin{aligned}
E[\mathbf{x}\mathbf{x}^T\mathbf{x}\mathbf{x}^T] &= 2(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)^2 + \mathbf{m}^T\mathbf{m}(\mathbf{\Sigma}-\mathbf{m}\mathbf{m}^T) \\
&\quad +\mathrm{Tr}(\mathbf{\Sigma})(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T) \\
E[\mathbf{x}\mathbf{x}^T\mathbf{A}\mathbf{x}\mathbf{x}^T] &= (\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)(\mathbf{A}+\mathbf{A}^T)(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T) \\
&\quad +\mathbf{m}^T\mathbf{A}\mathbf{m}(\mathbf{\Sigma}-\mathbf{m}\mathbf{m}^T) + \mathrm{Tr}[\mathbf{A}\mathbf{\Sigma}(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)] \\
E[\mathbf{x}^T\mathbf{x}\mathbf{x}^T\mathbf{x}] &= 2\mathrm{Tr}(\mathbf{\Sigma}^2) + 4\mathbf{m}^T\mathbf{\Sigma}\mathbf{m} + (\mathrm{Tr}(\mathbf{\Sigma}) + \mathbf{m}^T\mathbf{m})^2 \\
E[\mathbf{x}^T\mathbf{A}\mathbf{x}\mathbf{x}^T\mathbf{B}\mathbf{x}] &= \mathrm{Tr}[\mathbf{A}\mathbf{\Sigma}(\mathbf{B}+\mathbf{B}^T)\mathbf{\Sigma}] + \mathbf{m}^T(\mathbf{A}+\mathbf{A}^T)\mathbf{\Sigma}(\mathbf{B}+\mathbf{B}^T)\mathbf{m} \\
&\quad +(\mathrm{Tr}(\mathbf{A}\mathbf{\Sigma}) + \mathbf{m}^T\mathbf{A}\mathbf{m})(\mathrm{Tr}(\mathbf{B}\mathbf{\Sigma}) + \mathbf{m}^T\mathbf{B}\mathbf{m})
\end{aligned}$$

$$\begin{aligned}
&E[\mathbf{a}^T\mathbf{x}\mathbf{b}^T\mathbf{x}\mathbf{c}^T\mathbf{x}\mathbf{d}^T\mathbf{x}] \\
= \;&(\mathbf{a}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{b})(\mathbf{c}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{d}) \\
&+(\mathbf{a}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{c})(\mathbf{b}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{d}) \\
&+(\mathbf{a}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{d})(\mathbf{b}^T(\mathbf{\Sigma}+\mathbf{m}\mathbf{m}^T)\mathbf{c}) - 2\mathbf{a}^T\mathbf{m}\mathbf{b}^T\mathbf{m}\mathbf{c}^T\mathbf{m}\mathbf{d}^T\mathbf{m}
\end{aligned}$$

$$\begin{aligned}
&E[(\mathbf{A}\mathbf{x}+\mathbf{a})(\mathbf{B}\mathbf{x}+\mathbf{b})^T(\mathbf{C}\mathbf{x}+\mathbf{c})(\mathbf{D}\mathbf{x}+\mathbf{d})^T] \\
= \;&[\mathbf{A}\mathbf{\Sigma}\mathbf{B}^T + (\mathbf{A}\mathbf{m}+\mathbf{a})(\mathbf{B}\mathbf{m}+\mathbf{b})^T][\mathbf{C}\mathbf{\Sigma}\mathbf{D}^T + (\mathbf{C}\mathbf{m}+\mathbf{c})(\mathbf{D}\mathbf{m}+\mathbf{d})^T] \\
&+[\mathbf{A}\mathbf{\Sigma}\mathbf{C}^T + (\mathbf{A}\mathbf{m}+\mathbf{a})(\mathbf{C}\mathbf{m}+\mathbf{c})^T][\mathbf{B}\mathbf{\Sigma}\mathbf{D}^T + (\mathbf{B}\mathbf{m}+\mathbf{b})(\mathbf{D}\mathbf{m}+\mathbf{d})^T] \\
&+(\mathbf{B}\mathbf{m}+\mathbf{b})^T(\mathbf{C}\mathbf{m}+\mathbf{c})[\mathbf{A}\mathbf{\Sigma}\mathbf{D}^T - (\mathbf{A}\mathbf{m}+\mathbf{a})(\mathbf{D}\mathbf{m}+\mathbf{d})^T] \\
&+\mathrm{Tr}(\mathbf{B}\mathbf{\Sigma}\mathbf{C}^T)[\mathbf{A}\mathbf{\Sigma}\mathbf{D}^T + (\mathbf{A}\mathbf{m}+\mathbf{a})(\mathbf{D}\mathbf{m}+\mathbf{d})^T]
\end{aligned}$$

$$\begin{aligned}
&E[(\mathbf{A}\mathbf{x}+\mathbf{a})^T(\mathbf{B}\mathbf{x}+\mathbf{b})(\mathbf{C}\mathbf{x}+\mathbf{c})^T(\mathbf{D}\mathbf{x}+\mathbf{d})] \\
= \;&\mathrm{Tr}[\mathbf{A}\mathbf{\Sigma}(\mathbf{C}^T\mathbf{D}+\mathbf{D}^T\mathbf{C})\mathbf{\Sigma}\mathbf{B}^T] \\
&+[(\mathbf{A}\mathbf{m}+\mathbf{a})^T\mathbf{B} + (\mathbf{B}\mathbf{m}+\mathbf{b})^T\mathbf{A}]\mathbf{\Sigma}[\mathbf{C}^T(\mathbf{D}\mathbf{m}+\mathbf{d}) + \mathbf{D}^T(\mathbf{C}\mathbf{m}+\mathbf{c})] \\
&+[\mathrm{Tr}(\mathbf{A}\mathbf{\Sigma}\mathbf{B}^T) + (\mathbf{A}\mathbf{m}+\mathbf{a})^T(\mathbf{B}\mathbf{m}+\mathbf{b})][\mathrm{Tr}(\mathbf{C}\mathbf{\Sigma}\mathbf{D}^T) + (\mathbf{C}\mathbf{m}+\mathbf{c})^T(\mathbf{D}\mathbf{m}+\mathbf{d})]
\end{aligned}$$

See [7].

7.3.5 Moments

$$E[\mathbf{x}] = \sum_k \rho_k \mathbf{m}_k$$

$$\mathrm{Cov}(\mathbf{x}) = \sum_k \sum_{k'} \rho_k\rho_{k'}(\mathbf{\Sigma}_k + \mathbf{m}_k\mathbf{m}_k^T - \mathbf{m}_k\mathbf{m}_{k'}^T)$$

7.4 Miscellaneous

7.4.1 Whitening

Assume $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ then

$$\mathbf{z} = \mathbf{\Sigma}^{-1/2}(\mathbf{x} - \mathbf{m}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

Conversely having $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ one can generate data $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ by setting

$$\mathbf{x} = \mathbf{\Sigma}^{1/2}\mathbf{z} + \mathbf{m} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$$

Note that $\mathbf{\Sigma}^{1/2}$ means the matrix which fulfils $\mathbf{\Sigma}^{1/2}\mathbf{\Sigma}^{1/2} = \mathbf{\Sigma}$, and that it exists and is unique since $\mathbf{\Sigma}$ is positive definite.

7.4.2 The Chi-Square connection

Assume $\mathbf{x} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ and $\mathbf{x}$ to be n dimensional, then

$$z = (\mathbf{x} - \mathbf{m})^T \mathbf{\Sigma}^{-1}(\mathbf{x} - \mathbf{m}) \sim \chi^2_n$$

7.4.3 Entropy

Entropy of a D-dimensional gaussian

$$H(\mathbf{x}) = \int \mathcal{N}(\mathbf{m}, \mathbf{\Sigma}) \ln \mathcal{N}(\mathbf{m}, \mathbf{\Sigma}) d\mathbf{x} = -\ln \sqrt{\det(2\pi\mathbf{\Sigma})} - \frac{D}{2}$$

7.5 One Dimensional Mixture of Gaussians

7.5.1 Density and Normalization

$$p(s) = \sum_k^K \frac{\rho_k}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{1}{2}\frac{(s-\mu_k)^2}{\sigma_k^2}\right]$$

7.5.2 Moments

An useful fact of MoG, is that

$$\langle x^n \rangle = \sum_k \rho_k \langle x^n \rangle_k$$

where $\langle \cdot \rangle_k$ denotes average with respect to the $k.th$ component. We can calculate the first four moments from the densities

$$p(x) = \sum_k \rho_k \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{1}{2}\frac{(x-\mu_k)^2}{\sigma_k^2}\right]$$

$$p(x) = \sum_k \rho_k C_k \exp\left[c_{k2}x^2 + c_{k1}x\right]$$

as

$$
\begin{array}{rclcl}
\langle x\rangle & = & \sum_k \rho_k \mu_k & = & \sum_k \rho_k \left[\frac{-c_{k1}}{2c_{k2}}\right] \\
\langle x^2\rangle & = & \sum_k \rho_k(\sigma_k^2+\mu_k^2) & = & \sum_k \rho_k \left[\frac{-1}{2c_{k2}} + \left(\frac{-c_{k1}}{2c_{k2}}\right)^2\right] \\
\langle x^3\rangle & = & \sum_k \rho_k(3\sigma_k^2\mu_k+\mu_k^3) & = & \sum_k \rho_k \left[\frac{c_{k1}}{(2c_{k2})^2}\left[3-\frac{c_{k1}^2}{2c_{k2}}\right]\right] \\
\langle x^4\rangle & = & \sum_k \rho_k(\mu_k^4+6\mu_k^2\sigma_k^2+3\sigma_k^4) & = & \sum_k \rho_k \left[\left(\frac{1}{2c_{k2}}\right)^2\left[\left(\frac{c_{k1}}{2c_{k2}}\right)^2 - 6\frac{c_{k1}^2}{2c_{k2}}+3\right]\right]
\end{array}
$$

If all the gaussians are centered, i.e. $\mu_k = 0$ for all k, then

$$
\begin{array}{rclcl}
\langle x\rangle & = & 0 & = & 0 \\
\langle x^2\rangle & = & \sum_k \rho_k \sigma_k^2 & = & \sum_k \rho_k \left[\frac{-1}{2c_{k2}}\right] \\
\langle x^3\rangle & = & 0 & = & 0 \\
\langle x^4\rangle & = & \sum_k \rho_k 3\sigma_k^4 & = & \sum_k \rho_k 3\left[\frac{-1}{2c_{k2}}\right]^2
\end{array}
$$

From the un-centralized moments one can derive other entities like

$$
\begin{array}{rcl}
\langle x^2\rangle - \langle x\rangle^2 & = & \sum_{k,k'} \rho_k\rho_{k'}\left[\mu_k^2+\sigma_k^2-\mu_k\mu_{k'}\right] \\
\langle x^3\rangle - \langle x^2\rangle\langle x\rangle & = & \sum_{k,k'} \rho_k\rho_{k'}\left[3\sigma_k^2\mu_k+\mu_k^3-(\sigma_k^2+\mu_k^2)\mu_{k'}\right] \\
\langle x^4\rangle - \langle x^2\rangle^2 & = & \sum_{k,k'} \rho_k\rho_{k'}\left[\mu_k^4+6\mu_k^2\sigma_k^2+3\sigma_k^4-(\sigma_k^2+\mu_k^2)(\sigma_{k'}^2+\mu_{k'}^2)\right]
\end{array}
$$

7.6 Mixture of Gaussians

7.6.1 Density

The variable $\mathbf{x}$ is distributed as a mixture of gaussians if it has the density

$$p(\mathbf{x}) = \sum_{k=1}^{K} \rho_k \frac{1}{\sqrt{\det(2\pi\mathbf{\Sigma}_k)}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{m}_k)^T\mathbf{\Sigma}_k^{-1}(\mathbf{x}-\mathbf{m}_k)\right]$$

where ρ_k sum to 1 and the $\mathbf{\Sigma}_k$ all are positive definite.

8 Miscellaneous

8.1 Functions and Series

8.1.1 Finite Series

$$(\mathbf{X}^n - \mathbf{I})(\mathbf{X} - \mathbf{I})^{-1} = \mathbf{I} + \mathbf{X} + \mathbf{X}^2 + ... + \mathbf{X}^{n-1}$$

8.1.2 Taylor Expansion of Scalar Function

Consider some scalar function $f(\mathbf{x})$ which takes the vector $\mathbf{x}$ as an argument. This we can Taylor expand around $\mathbf{x}_0$

$$f(\mathbf{x}) \cong f(\mathbf{x}_0) + \mathbf{g}(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)$$

where

$$\mathbf{g}(\mathbf{x}_0) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\Big|_{\mathbf{x}_0} \qquad \mathbf{H}(\mathbf{x}_0) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T}\Big|_{\mathbf{x}_0}$$

8.1.3 Taylor Expansion of Vector Functions

8.1.4 Matrix Functions by Infinite Series

As for analytical functions in one dimension, one can define a matrix function for square matrices $\mathbf{X}$ by an infinite series

$$\mathbf{f}(\mathbf{X}) = \sum_{n=0}^{\infty} c_n \mathbf{X}^n$$

assuming the limit exists and is finite. If the coefficients c_n fulfils $\sum_n c_n x^n < \infty$, then one can prove that the above series exists and is finite, see [1]. Thus for any analytical function $f(x)$ there exists a corresponding matrix function $\mathbf{f}(\mathbf{x})$ constructed by the Taylor expansion. Using this one can prove the following results:

1) A matrix $\mathbf{A}$ is a zero of its own characteristic polynomium [1]:

$$p(\lambda) = \det(\mathbf{I}\lambda - \mathbf{A}) = \sum_n c_n \lambda^n \qquad \Rightarrow \qquad p(\mathbf{A}) = \mathbf{0}$$

2) If $\mathbf{A}$ is square it holds that [1]

$$\mathbf{A} = \mathbf{U}\mathbf{B}\mathbf{U}^{-1} \quad \Rightarrow \quad \mathbf{f}(\mathbf{A}) = \mathbf{U}\mathbf{f}(\mathbf{B})\mathbf{U}^{-1}$$

3) A useful fact when using power series is that

$$\mathbf{A}^n \to \mathbf{0} \text{ for } n \to \infty \qquad \text{if} \qquad |\mathbf{A}| < 1$$

8.1.5 Exponential Matrix Function

In analogy to the ordinary scalar exponential function, one can define exponential and logarithmic matrix functions:

$$e^{\mathbf{A}} = \sum_{n=0}^{\infty} \frac{1}{n!}\mathbf{A}^n = \mathbf{I} + \mathbf{A} + \frac{1}{2}\mathbf{A}^2 + ...$$

$$e^{-\mathbf{A}} = \sum_{n=0}^{\infty} \frac{1}{n!}(-1)^n\mathbf{A}^n = \mathbf{I} - \mathbf{A} + \frac{1}{2}\mathbf{A}^2 - ...$$

$$e^{t\mathbf{A}} = \sum_{n=0}^{\infty} \frac{1}{n!}(t\mathbf{A})^n = \mathbf{I} + t\mathbf{A} + \frac{1}{2}t^2\mathbf{A}^2 + ...$$

$$\ln(\mathbf{I} + \mathbf{A}) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}\mathbf{A}^n = \mathbf{A} - \frac{1}{2}\mathbf{A}^2 + \frac{1}{3}\mathbf{A}^3 - ...$$

Some of the properties of the exponential function are [1]

$$e^{\mathbf{A}}e^{\mathbf{B}} = e^{\mathbf{A}+\mathbf{B}} \qquad \text{if} \qquad \mathbf{AB} = \mathbf{BA}$$

$$(e^{\mathbf{A}})^{-1} = e^{-\mathbf{A}}$$

$$\frac{d}{dt}e^{t\mathbf{A}} = \mathbf{A}e^{t\mathbf{A}}, \qquad t \in \mathbb{R}$$

8.1.6 Trigonometric Functions

$$\sin(\mathbf{A}) = \sum_{n=0}^{\infty} \frac{(-1)^n\mathbf{A}^{2n+1}}{(2n+1)!} = \mathbf{A} - \frac{1}{3!}\mathbf{A}^3 + \frac{1}{5!}\mathbf{A}^5 - ...$$

$$\cos(\mathbf{A}) = \sum_{n=0}^{\infty} \frac{(-1)^n\mathbf{A}^{2n}}{(2n)!} = \mathbf{I} - \frac{1}{2!}\mathbf{A}^2 + \frac{1}{4!}\mathbf{A}^4 - ...$$

8.2 Indices, Entries and Vectors

Let $\mathbf{e}_i$ denote the column vector which is 1 on entry i and zero elsewhere, i.e. $(\mathbf{e}_i)_j = \delta_{ij}$, and let $\mathbf{J}^{ij}$ denote the matrix which is 1 on entry (i,j) and zero elsewhere.

8.2.1 Rows and Columns

$$i.th \text{ row of } \mathbf{A} = \mathbf{e}_i^T\mathbf{A}$$

$$j.th \text{ column of } \mathbf{A} = \mathbf{A}\mathbf{e}_j$$

8.2.2 Permutations

Let $\mathbf{P}$ be some permutation matrix, e.g.

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_2 & \mathbf{e}_1 & \mathbf{e}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_2^T \\ \mathbf{e}_1^T \\ \mathbf{e}_3^T \end{bmatrix}$$

then

$$\mathbf{AP} = \begin{bmatrix} \mathbf{Ae}_2 & \mathbf{Ae}_1 & \mathbf{Ae}_2 \end{bmatrix} \qquad \mathbf{PA} = \begin{bmatrix} \mathbf{e}_2^T\mathbf{A} \\ \mathbf{e}_1^T\mathbf{A} \\ \mathbf{e}_3^T\mathbf{A} \end{bmatrix}$$

That is, the first is a matrix which has columns of $\mathbf{A}$ but in permuted sequence and the second is a matrix which has the rows of $\mathbf{A}$ but in the permuted sequence.

8.2.3 Swap and Zeros

Assume $\mathbf{A}$ to be $n \times m$ and $\mathbf{J}^{ij}$ to be $m \times p$

$$\mathbf{AJ}^{ij} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_i & \dots & \mathbf{0} \end{bmatrix}$$

i.e. an $n \times p$ matrix of zeros with the i.th column of $\mathbf{A}$ in the placed of the j.th column. Assume $\mathbf{A}$ to be $n \times m$ and $\mathbf{J}^{ij}$ to be $p \times n$

$$\mathbf{J}^{ij}\mathbf{A} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{A}_j \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

i.e. an $p \times m$ matrix of zeros with the j.th row of $\mathbf{A}$ in the placed of the i.th row.

8.2.4 Rewriting product of elements

$$\begin{aligned} A_{ki}B_{jl} &= (\mathbf{Ae}_i\mathbf{e}_j^T\mathbf{B})_{kl} = (\mathbf{AJ}^{ij}\mathbf{B})_{kl} \\ A_{ik}B_{lj} &= (\mathbf{A}^T\mathbf{e}_i\mathbf{e}_j^T\mathbf{B}^T)_{kl} = (\mathbf{A}^T\mathbf{J}^{ij}\mathbf{B}^T)_{kl} \\ A_{ik}B_{jl} &= (\mathbf{A}^T\mathbf{e}_i\mathbf{e}_j^T\mathbf{B})_{kl} = (\mathbf{A}^T\mathbf{J}^{ij}\mathbf{B})_{kl} \\ A_{ki}B_{lj} &= (\mathbf{Ae}_i\mathbf{e}_j^T\mathbf{B}^T)_{kl} = (\mathbf{AJ}^{ij}\mathbf{B}^T)_{kl} \end{aligned}$$

8.2.5 Properties of the Singleentry Matrix

If $i = j$

$$\mathbf{J}^{ij}\mathbf{J}^{ij} = \mathbf{J}^{ij} \qquad (\mathbf{J}^{ij})^T(\mathbf{J}^{ij})^T = \mathbf{J}^{ij}$$
$$\mathbf{J}^{ij}(\mathbf{J}^{ij})^T = \mathbf{J}^{ij} \qquad (\mathbf{J}^{ij})^T\mathbf{J}^{ij} = \mathbf{J}^{ij}$$

If $i \neq j$

$$\mathbf{J}^{ij}\mathbf{J}^{ij} = \mathbf{0} \qquad (\mathbf{J}^{ij})^T(\mathbf{J}^{ij})^T = \mathbf{0}$$
$$\mathbf{J}^{ij}(\mathbf{J}^{ij})^T = \mathbf{J}^{ii} \qquad (\mathbf{J}^{ij})^T\mathbf{J}^{ij} = \mathbf{J}^{jj}$$

8.2.6 The Singleentry Matrix in Scalar Expressions

Assume $\mathbf{A}$ is $n \times m$ and $\mathbf{J}$ is $m \times n$, then

$$\mathrm{Tr}(\mathbf{A}\mathbf{J}^{ij}) = \mathrm{Tr}(\mathbf{J}^{ij}\mathbf{A}) = (\mathbf{A}^T)_{ij}$$

Assume $\mathbf{A}$ is $n \times n$, $\mathbf{J}$ is $n \times m$ and $\mathbf{B}$ is $m \times n$, then

$$\mathrm{Tr}(\mathbf{A}\mathbf{J}^{ij}\mathbf{B}) = (\mathbf{A}^T\mathbf{B}^T)_{ij}$$
$$\mathrm{Tr}(\mathbf{A}\mathbf{J}^{ji}\mathbf{B}) = (\mathbf{B}\mathbf{A})_{ij}$$
$$\mathrm{Tr}(\mathbf{A}\mathbf{J}^{ij}\mathbf{J}^{ij}\mathbf{B}) = \mathrm{diag}(\mathbf{A}^T\mathbf{B}^T)_{ij}$$

Assume $\mathbf{A}$ is $n \times n$, $\mathbf{J}^{ij}$ is $n \times m$ $\mathbf{B}$ is $m \times n$, then

$$\mathbf{x}^T\mathbf{A}\mathbf{J}^{ij}\mathbf{B}\mathbf{x} = (\mathbf{A}^T\mathbf{x}\mathbf{x}^T\mathbf{B}^T)_{ij}$$
$$\mathbf{x}^T\mathbf{A}\mathbf{J}^{ij}\mathbf{J}^{ij}\mathbf{B}\mathbf{x} = \mathrm{diag}(\mathbf{A}^T\mathbf{x}\mathbf{x}^T\mathbf{B}^T)_{ij}$$

8.2.7 Structure Matrices

The structure matrix is defined by

$$\frac{\partial \mathbf{A}}{\partial A_{ij}} = \mathbf{S}^{ij}$$

If $\mathbf{A}$ has no special structure then

$$\mathbf{S}^{ij} = \mathbf{J}^{ij}$$

If $\mathbf{A}$ is symmetric then

$$\mathbf{S}^{ij} = \mathbf{J}^{ij} + \mathbf{J}^{ji} - \mathbf{J}^{ij}\mathbf{J}^{ij}$$

8.3 Solutions to Systems of Equations

8.3.1 Existence in Linear Systems

Assume $\mathbf{A}$ is $n \times m$ and consider the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

Construct the augmented matrix $\mathbf{B} = [\mathbf{A}\ \mathbf{b}]$ then

Condition	*Solution*
$\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{B}) = m$	Unique solution $\mathbf{x}$
$\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{B}) < m$	Many solutions $\mathbf{x}$
$\text{rank}(\mathbf{A}) < \text{rank}(\mathbf{B})$	No solutions $\mathbf{x}$

8.3.2 Standard Square

Assume $\mathbf{A}$ is square and invertible, then

$$\mathbf{A}\mathbf{x} = \mathbf{b} \qquad \Rightarrow \qquad \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$

8.3.3 Degenerated Square

8.3.4 Over-determined Rectangular

Assume $\mathbf{A}$ to be $n \times m$, $n > m$ (tall) and $\text{rank}(\mathbf{A}) = m$, then

$$\mathbf{A}\mathbf{x} = \mathbf{b} \qquad \Rightarrow \qquad \mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b} = \mathbf{A}^{+}\mathbf{b}$$

that is *if* there exists a solution $\mathbf{x}$ at all! If there is no solution the following can be useful:

$$\mathbf{A}\mathbf{x} = \mathbf{b} \qquad \Rightarrow \qquad \mathbf{x}_{min} = \mathbf{A}^{+}\mathbf{b}$$

Now $\mathbf{x}_{min}$ is the vector $\mathbf{x}$ which minimizes $||\mathbf{A}\mathbf{x} - \mathbf{b}||^2$, i.e. the vector which is "least wrong". The matrix $\mathbf{A}^{+}$ is the pseudo-inverse of $\mathbf{A}$. See [3].

8.3.5 Under-determined Rectangular

Assume $\mathbf{A}$ is $n \times m$ and $n < m$ ("broad").

$$\mathbf{A}\mathbf{x} = \mathbf{b} \qquad \Rightarrow \qquad \mathbf{x}_{min} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b}$$

The equation have many solutions $\mathbf{x}$. But $\mathbf{x}_{min}$ is the solution which minimizes $||\mathbf{A}\mathbf{x} - \mathbf{b}||^2$ and also the solution with the smallest norm $||\mathbf{x}||^2$. The same holds for a matrix version: Assume $\mathbf{A}$ is $n \times m$, $\mathbf{X}$ is $m \times n$ and $\mathbf{B}$ is $n \times n$, then

$$\mathbf{A}\mathbf{X} = \mathbf{B} \qquad \Rightarrow \qquad \mathbf{X}_{min} = \mathbf{A}^{+}\mathbf{B}$$

The equation have many solutions $\mathbf{X}$. But $\mathbf{X}_{min}$ is the solution which minimizes $||\mathbf{A}\mathbf{X} - \mathbf{B}||^2$ and also the solution with the smallest norm $||\mathbf{X}||^2$. See [3].

Similar but different: Assume $\mathbf{A}$ is square $n \times n$ and the matrices $\mathbf{B}_0, \mathbf{B}_1$ are $n \times N$, where $N > n$, then if $\mathbf{B}_0$ has maximal rank

$$\mathbf{A}\mathbf{B}_0 = \mathbf{B}_1 \qquad \Rightarrow \qquad \mathbf{A}_{min} = \mathbf{B}_1\mathbf{B}_0^T(\mathbf{B}_0\mathbf{B}_0^T)^{-1}$$

where $\mathbf{A}_{min}$ denotes the matrix which is optimal in a least square sense. An interpretation is that $\mathbf{A}$ is the linear approximation which maps the columns vectors of $\mathbf{B}_0$ into the columns vectors of $\mathbf{B}_1$.

8.3.6 Linear form and zeros

$$\mathbf{A}\mathbf{x} = \mathbf{0}, \quad \forall\mathbf{x} \qquad \Rightarrow \qquad \mathbf{A} = \mathbf{0}$$

8.3.7 Square form and zeros

If $\mathbf{A}$ is symmetric, then

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = 0, \quad \forall\mathbf{x} \qquad \Rightarrow \qquad \mathbf{A} = \mathbf{0}$$

8.4 Block matrices

Let $\mathbf{A}_{ij}$ denote the ij.th block of $\mathbf{A}$.

8.4.1 Multiplication

Assuming the dimensions of the blocks matches we have

$$\left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \end{array}\right] \left[\begin{array}{c|c} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \hline \mathbf{B}_{21} & \mathbf{B}_{22} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{A}_{11}\mathbf{B}_{11} + \mathbf{A}_{12}\mathbf{B}_{21} & \mathbf{A}_{11}\mathbf{B}_{12} + \mathbf{A}_{12}\mathbf{B}_{22} \\ \hline \mathbf{A}_{21}\mathbf{B}_{11} + \mathbf{A}_{22}\mathbf{B}_{21} & \mathbf{A}_{21}\mathbf{B}_{12} + \mathbf{A}_{22}\mathbf{B}_{22} \end{array}\right]$$

8.4.2 The Determinant

The determinant can be expressed as by the use of

$$\begin{aligned} \mathbf{C}_1 &= \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \\ \mathbf{C}_2 &= \mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12} \end{aligned}$$

as

$$\det\left(\left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \end{array}\right]\right) = \det(\mathbf{A}_{22}) \cdot \det(\mathbf{C}_1) = \det(\mathbf{A}_{11}) \cdot \det(\mathbf{C}_2)$$

8.4.3 The Inverse

The inverse can be expressed as by the use of

$$\begin{aligned} \mathbf{C}_1 &= \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \\ \mathbf{C}_2 &= \mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12} \end{aligned}$$

as

$$\left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \end{array}\right]^{-1} = \left[\begin{array}{c|c} \mathbf{C}_1^{-1} & -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{C}_2^{-1} \\ \hline -\mathbf{C}_2^{-1}\mathbf{A}_{21}\mathbf{A}_{11}^{-1} & \mathbf{C}_2^{-1} \end{array}\right]$$

$$= \left[\begin{array}{c|c} \mathbf{A}_{11}^{-1} + \mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{C}_2^{-1}\mathbf{A}_{21}\mathbf{A}_{11}^{-1} & -\mathbf{C}_1^{-1}\mathbf{A}_{12}\mathbf{A}_{22}^{-1} \\ \hline -\mathbf{A}_{22}^{-1}\mathbf{A}_{21}\mathbf{C}_1^{-1} & \mathbf{A}_{22}^{-1} + \mathbf{A}_{22}^{-1}\mathbf{A}_{21}\mathbf{C}_1^{-1}\mathbf{A}_{12}\mathbf{A}_{22}^{-1} \end{array}\right]$$

8.4.4 Block diagonal

For block diagonal matrices we have

$$\left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{A}_{22} \end{array}\right]^{-1} = \left[\begin{array}{c|c} (\mathbf{A}_{11})^{-1} & \mathbf{0} \\ \hline \mathbf{0} & (\mathbf{A}_{22})^{-1} \end{array}\right]$$

$$\det\left(\left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{A}_{22} \end{array}\right]\right) = \det(\mathbf{A}_{11}) \cdot \det(\mathbf{A}_{22})$$

8.5 Matrix Norms

8.5.1 Definitions

A matrix norm is a mapping which fulfils

$$\begin{aligned} ||\mathbf{A}|| &\geq 0 \qquad ||\mathbf{A}|| = 0 \Leftrightarrow \mathbf{A} = \mathbf{0} \\ ||c\mathbf{A}|| &= |c|||\mathbf{A}||, \qquad c \in \mathbb{R} \\ ||\mathbf{A}+\mathbf{B}|| &\leq ||\mathbf{A}|| + ||\mathbf{B}|| \end{aligned}$$

8.5.2 Examples

$$\begin{aligned} ||\mathbf{A}||_1 &= \max_j \sum_i |A_{ij}| \\ ||\mathbf{A}||_2 &= \sqrt{\max \operatorname{eig}(\mathbf{A}^T\mathbf{A})} \\ ||\mathbf{A}||_p &= (\max_{||\mathbf{x}||_p=1} ||\mathbf{A}\mathbf{x}||_p)^{1/p} \\ ||\mathbf{A}||_\infty &= \max_i \sum_j |A_{ij}| \\ ||\mathbf{A}||_F &= \sqrt{\sum_{ij} |A_{ij}|^2} \qquad \text{(Frobenius)} \\ ||\mathbf{A}||_{max} &= \max_{ij} |A_{ij}| \\ ||\mathbf{A}||_{KF} &= ||\operatorname{sing}(\mathbf{A})||_1 \qquad \text{(Ky Fan)} \end{aligned}$$

where $\operatorname{sing}(\mathbf{A})$ is the vector of singular values of the matrix $\mathbf{A}$.

8.5.3 Inequalities

E. H. Rasmussen has in yet unpublished material derived and collected the following inequalities. They are collected in a table as below, assuming $\mathbf{A}$ is an $m \times n$, and $d = \min\{m, n\}$

	$\|\|\mathbf{A}\|\|_{max}$	$\|\|\mathbf{A}\|\|_1$	$\|\|\mathbf{A}\|\|_\infty$	$\|\|\mathbf{A}\|\|_2$	$\|\|\mathbf{A}\|\|_F$	$\|\|\mathbf{A}\|\|_{KF}$
$\|\|\mathbf{A}\|\|_{max}$		1	1	1	1	1
$\|\|\mathbf{A}\|\|_1$	m		m	$\sqrt{m}$	$\sqrt{m}$	$\sqrt{m}$
$\|\|\mathbf{A}\|\|_\infty$	n	n		$\sqrt{n}$	$\sqrt{n}$	$\sqrt{n}$
$\|\|\mathbf{A}\|\|_2$	$\sqrt{mn}$	$\sqrt{n}$	$\sqrt{m}$		1	1
$\|\|\mathbf{A}\|\|_F$	$\sqrt{mn}$	$\sqrt{n}$	$\sqrt{m}$	$\sqrt{d}$		1
$\|\|\mathbf{A}\|\|_{KF}$	$\sqrt{mnd}$	$\sqrt{nd}$	$\sqrt{md}$	d	$\sqrt{d}$	

which are to be read as, e.g.

$$||\mathbf{A}||_2 \leq \sqrt{m} \cdot ||\mathbf{A}||_\infty$$

8.6 Positive Definite and Semi-definite Matrices

8.6.1 Definitions

A matrix $\mathbf{A}$ is positive definite if and only if

$$\mathbf{x}^T\mathbf{A}\mathbf{x} > \mathbf{0}, \quad \forall \mathbf{x}$$

A matrix $\mathbf{A}$ is positive semi-definite if and only if

$$\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \mathbf{0}, \quad \forall \mathbf{x}$$

Note that if $\mathbf{A}$ is positive definite, then $\mathbf{A}$ is also positive semi-definite.

8.6.2 Eigenvalues

The following holds with respect to the eigenvalues:

$$\begin{array}{rcl} \mathbf{A} \text{ pos. def.} & \Rightarrow & \text{eig}(\mathbf{A}) > 0 \\ \mathbf{A} \text{ pos. semi-def.} & \Rightarrow & \text{eig}(\mathbf{A}) \geq 0 \end{array}$$

8.6.3 Trace

The following holds with respect to the trace:

$$\begin{array}{rcl} \mathbf{A} \text{ pos. def.} & \Rightarrow & \text{Tr}(\mathbf{A}) > 0 \\ \mathbf{A} \text{ pos. semi-def.} & \Rightarrow & \text{Tr}(\mathbf{A}) \geq 0 \end{array}$$

8.6.4 Inverse

If $\mathbf{A}$ is positive definite, then $\mathbf{A}$ is invertible and $\mathbf{A}^{-1}$ is also positive definite.

8.6.5 Diagonal

If $\mathbf{A}$ is positive definite, then $A_{ii} > 0, \forall i$

8.6.6 Decomposition I

The matrix $\mathbf{A}$ is positive semi-definite of rank r $\Leftrightarrow$ there exists a matrix $\mathbf{B}$ of rank r such that $\mathbf{A} = \mathbf{B}\mathbf{B}^T$

The matrix $\mathbf{A}$ is positive definite $\Leftrightarrow$ there exists an invertible matrix $\mathbf{B}$ such that $\mathbf{A} = \mathbf{B}\mathbf{B}^T$

8.6.7 Decomposition II

Assume $\mathbf{A}$ is an $n \times n$ positive semi-definite, then there exists an $n \times r$ matrix $\mathbf{B}$ of rank r such that $\mathbf{B}^T\mathbf{A}\mathbf{B} = \mathbf{I}$.

8.6.8 Equation with zeros

Assume $\mathbf{A}$ is positive semi-definite, then $\mathbf{X}^T\mathbf{A}\mathbf{X} = \mathbf{0} \quad \Rightarrow \quad \mathbf{A}\mathbf{X} = \mathbf{0}$

8.6.9 Rank of product

Assume $\mathbf{A}$ is positive definite, then $\text{rank}(\mathbf{B}\mathbf{A}\mathbf{B}^T) = \text{rank}(\mathbf{B})$

8.6.10 Positive definite property

If $\mathbf{A}$ is $n \times n$ positive definite and $\mathbf{B}$ is $r \times n$ of rank r, then $\mathbf{B}\mathbf{A}\mathbf{B}^T$ is positive definite.

8.6.11 Outer Product

If $\mathbf{X}$ is $n \times r$ of rank r, then $\mathbf{X}\mathbf{X}^T$ is positive definite.

8.6.12 Small pertubations

If $\mathbf{A}$ is positive definite and $\mathbf{B}$ is symmetric, then $\mathbf{A} - t\mathbf{B}$ is positive definite for sufficiently small t.

8.7 Integral Involving Dirac Delta Functions

Assuming $\mathbf{A}$ to be square, then

$$\int p(\mathbf{s})\delta(\mathbf{x} - \mathbf{A}\mathbf{s})d\mathbf{s} = \frac{1}{\det(\mathbf{A})}p(\mathbf{A}^{-1}\mathbf{x})$$

Assuming $\mathbf{A}$ to be "underdetermined", i.e. "tall", then

$$\int p(\mathbf{s})\delta(\mathbf{x} - \mathbf{A}\mathbf{s})d\mathbf{s} = \left\{ \begin{array}{ll} \frac{1}{\sqrt{\det(\mathbf{A}^T\mathbf{A})}}p(\mathbf{A}^+\mathbf{x}) & \text{if } \mathbf{x} = \mathbf{A}\mathbf{A}^+\mathbf{x} \\ 0 & \text{elsewhere} \end{array} \right\}$$

See [8].

8.8 Miscellaneous

For any $\mathbf{A}$ it holds that

$$\text{rank}(\mathbf{A}) = \text{rank}(\mathbf{A}^T) = \text{rank}(\mathbf{A}\mathbf{A}^T) = \text{rank}(\mathbf{A}^T\mathbf{A})$$

Assume $\mathbf{A}$ is positive definite. Then

$$\text{rank}(\mathbf{B}^T\mathbf{A}\mathbf{B}) = \text{rank}(\mathbf{B})$$

$$\mathbf{A} \text{ is positive definite} \quad \Leftrightarrow \quad \exists \mathbf{B} \text{ invertible, such that } \mathbf{A} = \mathbf{B}\mathbf{B}^T$$

A Proofs and Details

A.0.1 Proof of Equation 14

Essentially we need to calculate

$$\begin{aligned}
\frac{\partial(\mathbf{X}^n)_{kl}}{\partial X_{ij}} &= \frac{\partial}{\partial X_{ij}} \sum_{u_1,\ldots,u_{n-1}} X_{k,u_1} X_{u_1,u_2} \ldots X_{u_{n-1},l} \\
&= \delta_{k,i}\delta_{u_1,j} X_{u_1,u_2} \ldots X_{u_{n-1},l} \\
&\quad + X_{k,u_1}\delta_{u_1,i}\delta_{u_2,j} \ldots X_{u_{n-1},l} \\
&\quad \vdots \\
&\quad + X_{k,u_1} X_{u_1,u_2} \ldots \delta_{u_{n-1},i}\delta_{l,j} \\
&= \sum_{r=0}^{n-1} (\mathbf{X}^r)_{ki} (\mathbf{X}^{n-1-r})_{jl} \\
&= \sum_{r=0}^{n-1} (\mathbf{X}^r \mathbf{J}^{ij} \mathbf{X}^{n-1-r})_{kl}
\end{aligned}$$

Using the properties of the single entry matrix found in Sec. 8.2.5, the result follows easily.

A.0.2 Details on Eq. 47

$$\begin{aligned}
\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X}) &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\partial(\mathbf{X}^H\mathbf{A}\mathbf{X})] \\
&= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}(\partial(\mathbf{X}^H)\mathbf{A}\mathbf{X} + \mathbf{X}^H\partial(\mathbf{A}\mathbf{X}))] \\
&= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big(\mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\partial(\mathbf{X}^H)\mathbf{A}\mathbf{X}] + \mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\partial(\mathbf{A}\mathbf{X})]\big) \\
&= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big(\mathrm{Tr}[\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\partial(\mathbf{X}^H)] + \mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A}\partial(\mathbf{X})]\big)
\end{aligned}$$

First, the derivative is found with respect to the real part of $\mathbf{X}$

$$\begin{aligned}
\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Re\mathbf{X}} &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\Big(\frac{\mathrm{Tr}[\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\partial(\mathbf{X}^H)]}{\partial \Re\mathbf{X}} + \frac{\mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A}\partial(\mathbf{X})]}{\partial \Re\mathbf{X}}\Big) \\
&= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big(\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1} + ((\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A})^T\big)
\end{aligned}$$

Through the calculations, (16) and (35) were used. In addition, by use of (36), the derivative is found with respect to the imaginary part of $\mathbf{X}$

$$\begin{aligned}
i\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Im\mathbf{X}} &= i\det(\mathbf{X}^H\mathbf{A}\mathbf{X})\Big(\frac{\mathrm{Tr}[\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\partial(\mathbf{X}^H)]}{\partial \Im\mathbf{X}} + \frac{\mathrm{Tr}[(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A}\partial(\mathbf{X})]}{\partial \Im\mathbf{X}}\Big) \\
&= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big(\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1} - ((\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A})^T\big)
\end{aligned}$$

Hence, derivative yields

$$\begin{aligned}\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \mathbf{X}} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Re\mathbf{X}} - i\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Im\mathbf{X}}\Big) \\ &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\big((\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^H\mathbf{A}\big)^T\end{aligned}$$

and the complex conjugate derivative yields

$$\begin{aligned}\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \mathbf{X}^*} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Re\mathbf{X}} + i\frac{\partial \det(\mathbf{X}^H\mathbf{A}\mathbf{X})}{\partial \Im\mathbf{X}}\Big) \\ &= \det(\mathbf{X}^H\mathbf{A}\mathbf{X})\mathbf{A}\mathbf{X}(\mathbf{X}^H\mathbf{A}\mathbf{X})^{-1}\end{aligned}$$

Notice, for real $\mathbf{X}$, $\mathbf{A}$, the sum of (44) and (45) is reduced to (13).
Similar calculations yield

$$\begin{aligned}\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \mathbf{X}} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \Re\mathbf{X}} - i\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \Im\mathbf{X}}\Big) \\ &= \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)\big(\mathbf{A}\mathbf{X}^H(\mathbf{X}\mathbf{A}\mathbf{X}^H)^{-1}\big)^T \qquad (46)\end{aligned}$$

and

$$\begin{aligned}\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \mathbf{X}^*} &= \frac{1}{2}\Big(\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \Re\mathbf{X}} + i\frac{\partial \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)}{\partial \Im\mathbf{X}}\Big) \\ &= \det(\mathbf{X}\mathbf{A}\mathbf{X}^H)(\mathbf{X}\mathbf{A}\mathbf{X}^H)^{-1}\mathbf{X}\mathbf{A} \qquad (47)\end{aligned}$$

References

[1] Karl Gustav Andersson and Lars-Christer Boiers. *Ordinaera differentialekvationer.* Studenterlitteratur, 1992.

[2] Jörn Anemüller, Terrence J. Sejnowski, and Scott Makeig. Complex independent component analysis of frequency-domain electroencephalographic data. *Neural Networks*, 16(9):1311–1323, November 2003.

[3] S. Barnet. *Matrices. Methods and Applications.* Oxford Applied Mathematics and Computin Science Series. Clarendon Press, 1990.

[4] Christoffer Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[5] Robert J. Boik. Lecture notes: Statistics 550. Online, April 22 2002. Notes.

[6] D. H. Brandwood. A complex gradient operator and its application in adaptive array theory. *IEE Proceedings*, 130(1):11–16, February 1983. PTS. F and H.

[7] M. Brookes. Matrix reference manual, 2004. Website May 20, 2004.

[8] Mads Dyrholm. Some matrix results, 2004. Website August 23, 2004.

[9] Simon Haykin. *Adaptive Filter Theory.* Prentice Hall, Upper Saddle River, NJ, 4th edition, 2002.

[10] Thomas P. Minka. Old and new matrix algebra useful for statistics, December 2000. Notes.

[11] L. Parra and C. Spence. Convolutive blind separation of non-stationary sources. In *IEEE Transactions Speech and Audio Processing*, pages 320–327, May 2000.

[12] Laurent Schwartz. *Cours d'Analyse*, volume II. Hermann, Paris, 1967. As referenced in [9].

[13] Shayle R. Searle. *Matrix Algebra Useful for Statistics.* John Wiley and Sons, 1982.

[14] G. Seber and A. Lee. *Linear Regression Analysis.* John Wiley and Sons, 2002.

[15] S. M. Selby. *Standard Mathematical Tables.* CRC Press, 1974.

[16] Inna Stainvas. Matrix algebra in differential calculus. Neural Computing Research Group, Information Engeneering, Aston University, UK, August 2002. Notes.

[17] Max Welling. The kalman filter. Lecture Note.

Bibliography

[1] Hagai Attias. Independent Factor Analysis. *Neural Computation*, (11):803–851, 1999.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statistics*, 41:164–171, 1970.

[3] Adel Belouchrani and Jean-Francois Cardoso. A maximum likelihood source separation for discrete sources. In *Proceedings of EUSIPCO*, volume 2, pages 768–771, 1994.

[4] O. Bermond and Jean Francois Cardoso. Approximate Likelihood for Noisy Mixtures. In *Proceedings of the ICA Conference*, 1999.

[5] Christoffer Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[6] C. G. Broyden. A class of methods for solving nonlinear simultanoeus equations. *Math. Comput.*, 19:577–593, 1965.

[7] 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*, 39:1–38, 1977.

[8] L. K. Hansen and K. B. Petersen. Monaural Separation is Hard. In *Proceedings of ICA'2003 Fourth Int. Symp. on Independent Component Analysis and Blind Signal Separation, Nara Japan*, 2003.

[9] John Hertz, Anders Krogh, and Richard G. Palmer. *Introduction to the theory of neural computing.* Addison-Wesley, 1991.

[10] Geoffrey Hinton. Training Products of Experts by Minimizing Contrastive Divergence. Technical Report GCNU-TR-2000-04, Gatsby Computational Neuroscience Unit, University College London, 2000.

[11] Geoffrey Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, (17):1771–1800, 2002.

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

[13] S. C. Horng. Examples of sublinear convergence of the em algorithm. In *Proceedings of the Statistical Computing Section, American Statistical Association*, 1987.

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

[15] Aapo Hyvarinen and Errki Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13, 2000.

[16] Mortaza Jamshidian and Robert I. Jennrich. Conjugate gradient acceleration of the EM algorithm. *Journal of the American Statistical Association*, 88(421), 1993.

[17] Mortaza Jamshidian and Robert I. Jennrich. Conjugate gradient methods in confirmatory factor analysis. *Computational Statistics Data Anal.*, 17:247–263, 1994.

[18] Mortaza Jamshidian and Robert I. Jennrich. Acceleration of the EM algorithm using quasi-newton methods. *Journal of Royal Statistical Society, Series B*, 59(3):569–587, 1997.

[19] N. M. Laird, N. Lange, and D. O. Stram. Maximum likelihood computations with repeated measures: application of the EM algorithm. *Journal of American Statistics Association*, 82:97–105, 1987.

[20] Kenneth Lange. A quasi newton acceleration of the EM algorithm. *Statistica Sinica*, 5:1–18, 1995.

[21] Harry Lappalainen. Ensemble learning for independent component analysis. In *Proceedings of First International Workshop on Independent Component Analysis and Signal Separation, ICA*, 1999.

[22] Neil D. Lawrence and Christopher M. Bishop. Variational Bayes Independent Component Analysis. Technical report, Computer Laboratory, University of Cambridge, 2000.

[23] M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12:337–365, 2000.

[24] T. A. Louis. Finding the observed information matrix when using the EM algorithm. *Journal of Royal Statistical Society, Series B*, 44:226–233, 1982.

[25] David MacKay. Failures of the one-step learning algorithm. Unpublished Note, Cavendish Laboratory, Cambridge, 2001.

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

[27] X. L. Meng. On the rate of convergence of the EM algorithm. *The Annals of Statistics*, 22:326–339, 1994.

[28] X. L. Meng and Steven Pedlow. EM: A Bibliographic Review with Missing Papers. In *Proceedings of Statistical Computing Section, American Statistical Association*, pages 24–27, 1992.

[29] X. L. Meng and David van Dyk. The EM algorithm - an old folk-song sung to a fast new tune. *Journal of Royal Statistical Society, Series B*, 59(3):511–567, 1997.

[30] Eric Moulines, Jean-Francois Cardoso, and Elisabeth Gassiat. Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. In *Proceedings of the ICA Conference*, 1997.

[31] Radford M. Neal. Probabilistic inference using markov chain monte carlo. Technical Report CRG-TR-92-1, Department of Computer Science, University of Toronto, 1993.

[32] Radford M. Neal and Geoffrey Hinton. A new view of the EM algorithm that justifies incremental and other variants. Technical report, Department of Computer Science, University of Toronto, 1993.

[33] Hans Bruun Nielsen. UCMINF - An Algorithm for Unconstrained Nonlinear Optimization. Technical Report IMM-Rep-2000-19, Technical University of Denmark, 2000.

[34] Rasmus Kongsgaard Olsson, Tue Lehn-Schiøler, and Kaare Brandt Petersen. State-space models - from the EM algorithm to a gradient approach. *Submitted to Neural Computation*, 2005.

[35] G. Parisi. *Statistical Field Theory*. Addison-Wesley, 1988.

[36] M. S. Pedersen, J. Larsen, and U. Kjems. On the difference between updating the mixing matrix and updating the separation matrix. volume V, pages 297–300, Philadelphia, PA, USA, mar 2005.

[37] Michael Syskind Pedersen. The matrix cookbook, 2005.

[38] K. B. Petersen. Using mixture of gaussians to compare approaches to signal separation. In *Proceedings of 13th Danish Conference on Pattern Recognition and Image Analysis, DSAGM*, aug 2004.

[39] Kaare Brandt Petersen. Physics in signalprocessing - mean field ICA aproach to the problem of separating mixed binary signals. Master's thesis, University of Copenhagen, 2002.

[40] Kaare Brandt Petersen, Jiucang Hao, and Te-Won Lee. Generative and filtering approaches for overcomplete representations. *Neural Information Processing*, 8(1), 2005.

[41] Kaare Brandt Petersen and Ole Winther. Explaining slow convergence of EM in low noise linear mixtures. Technical Report 2005-2, Informatics and Mathematical Modelling, Technical University of Denmark, 2005.

[42] Kaare Brandt Petersen and Ole Winther. The EM Algorithm in Independent Component Analysis. In *International Conference on Acoustics, Speech, and Signal Processing*, 2005.

[43] Kaare Brandt Petersen, Ole Winther, and Lars Kai Hansen. On the convergence of EM and VBEM. *Neural Computation*, 17(9), 2005.

[44] Richard A. Redner and Homer F. Walker. Mixture densitites, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239, 1984.

[45] Donald B. Rubin and Dorothy T. Thayer. EM Algorithms for ML Factor Analysis. *Psychometrika*, 47(1), 1982.

[46] R. Salakhutdinov and S. Roweis. Adaptive overrelaxed bound optimization methods. In *Proceedings of International Conference on Machine Learning, ICML*. International Conference on Machine Learning, ICML, 2003.

[47] R. Salakhutdinov, S. Roweis, and Z. Ghahramani. Optimization with EM and expectation-conjugate-gradient. In *Proceedings of International Conference on Machine Learning, ICML*. International Conference on Machine Learning, ICML, 2003.

[48] N. R. Sandell and K. I. Yared. Maximum likelihood identification of state space models for linear dynamic systems. Technical Report ESL-R-814, Electronic Systems Lab, MIT, 1978.

[49] G. Seber and A. Lee. *Linear Regression Analysis*. John Wiley and Sons, 2002.

[50] S. Stigler. Citation patterns in the journals of statistics and probability. *Statist. Sci.*, 9:94–108, 1994.

[51] R. Sundberg. *Maximum Likelihood theory and applications for distributions generated when observing a function of an exponential variable.* PhD thesis, Stockholm University, 1972.

[52] Yee Whye Teh, Max Welling, Simon Osindero, and Geoffrey E. Hinton. Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research*, 4:1235–1260, 2003.

[53] C. F. Jeff Wu. On the covergence properties of the EM algortihm. *The Annals of Statistics*, 11(1):95–103, 1983.

[54] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized belief propagation. In *Advances in Neural Information Processing Systems (NIPS)*, volume Vol 13, pages 689–695, 2000.