Helsinki University of Technology
Dissertations in Computer and Information Science
Espoo 2005 Report D10

ADVANCES IN VARIATIONAL BAYESIAN NONLINEAR BLIND SOURCE SEPARATION

Antti Honkela

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Department of Computer Science and Engineering for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 13th of May, 2005, at 12 o'clock noon.

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory of Computer and Information Science

Distribution:
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O. Box 5400
FI-02015 TKK
FINLAND
Tel. +358-9-451 3272
Fax +358-9-451 3277
http://www.cis.hut.fi

Available in PDF format at http://lib.tkk.fi/Diss/2005/isbn9512276550/

ISBN 951-22-7654-2 (printed version)
ISBN 951-22-7655-0 (electronic version)
ISSN 1459-7020

Otamedia Oy
Espoo 2005

Honkela, A. (2005): **Advances in variational Bayesian nonlinear blind source separation**. Doctoral thesis, Helsinki University of Technology, Dissertations in Computer and Information Science, Report D10, Espoo, Finland.

Keywords: Bayesian learning, blind source separation, latent variable models, nonlinear blind source separation, nonlinear factor analysis, nonlinear models, post-nonlinear mixing, unsupervised learning, variational methods.

ABSTRACT

Linear data analysis methods such as factor analysis (FA), independent component analysis (ICA) and blind source separation (BSS) as well as state-space models such as the Kalman filter model are used in a wide range of applications. In many of these, linearity is just a convenient approximation while the underlying effect is nonlinear. It would therefore be more appropriate to use nonlinear methods.

In this work, nonlinear generalisations of FA and ICA/BSS are presented. The methods are based on a generative model, with a multilayer perceptron (MLP) network to model the nonlinearity from the latent variables to the observations. The model is estimated using variational Bayesian learning. The variational Bayesian method is well-suited for the nonlinear data analysis problems. The approach is also theoretically interesting, as essentially the same method is used in several different fields and can be derived from several different starting points, including statistical physics, information theory, Bayesian statistics, and information geometry. These complementary views can provide benefits for interpretation of the operation of the learning method and its results.

Much of the work presented in this thesis consists of improvements that make the nonlinear factor analysis and blind source separation methods faster and more stable, while being applicable to other learning problems as well. The improvements include methods to accelerate convergence of alternating optimisation algorithms such as the EM algorithm and an improved approximation of the moments of a nonlinear transform of a multivariate probability distribution. These improvements can be easily applied to other models besides FA and ICA/BSS, such as nonlinear state-space models. A specialised version of the nonlinear factor analysis method for post-nonlinear mixtures is presented as well.

Preface

This work has been carried out at the Neural Networks Research Centre of Helsinki University of Technology, hosted by the Laboratory of Computer and Information Science. The main source of funding for the work has been the Graduate School in Computational Methods of Information Technology (ComMIT). The work has also been supported by the IST Programme of the European Community, under the project BLISS, IST-1999-14190, and under the PASCAL Network of Excellence, IST-2002-506778. Personal grants from the Finnish Cultural Foundation are also gratefully acknowledged.

I wish to thank my instructor Dr. Harri Valpola for letting me take part in his ground-breaking work and guiding my attempts to develop it further. I also thank my two supervisors Prof. Juha Karhunen and Academy Prof. Erkki Oja for their support for my work. All of them also gave valuable comments on this manuscript.

I wish to express my gratitude to the co-authors of the publications of the thesis, Dr. Harri Valpola, Prof. Juha Karhunen, Alexander Ilin, Dr. Stefan Harmeling, and Leo Lundqvist. I also wish to thank all present and former members of the Bayes group, especially Dr. Harri Valpola, Tapani Raiko, Markus Harva and Tomas Östman for their help in many research related problems and interesting discussions.

The manuscript of the thesis was reviewed by Dr. Fabian Theis and Dr. Aki Vehtari. I am grateful for their comments on the text that helped me improve it.

I also wish to thank everyone working at the Laboratory of Computer and Information Science for making it such a nice and stimulating working environment. This includes especially present and former members of the 11 o'clock lunch group: Esa, Johan, Jukka, Markus, Miki and other less frequent visitors. Petteri and Jaakko P. are also acknowledged for interesting and challenging discussions.

Finally, I wish to thank Maija (and Miia) for their encouragement and support in writing the thesis.

Otaniemi, April 2005

Antti Honkela

Contents

Publications of the thesis

This thesis consists of an introductory part and the following seven publications.

I H. Lappalainen and A. Honkela. Bayesian Nonlinear Independent Component Analysis by Multi-Layer Perceptrons. In M. Girolami, ed., *Advances in Independent Component Analysis*, pp. 93–121, Springer-Verlag, 2000.

II A. Honkela, H. Valpola and J. Karhunen. Accelerating Cyclic Update Algorithms for Parameter Estimation by Pattern Searches. *Neural Processing Letters* 17(2), pp. 191–203, 2003.

III A. Honkela and H. Valpola. Variational Learning and Bits-Back Coding: an Information-Theoretic View to Bayesian Learning. *IEEE Transactions on Neural Networks* 15(4), pp. 800–810, 2004.

IV A. Ilin and A. Honkela. Postnonlinear Independent Component Analysis by Variational Bayesian Learning. In Proceedings of the Fifth International Conference on Independent Component Analysis and Blind Signal Separation (ICA 2004), Vol. 3195 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 766–773, 2004.

V A. Honkela, S. Harmeling, L. Lundqvist and H. Valpola. Using Kernel PCA for Initialisation of Variational Bayesian Nonlinear Blind Source Separation Method. In Proceedings of the Fifth International Conference on Independent Component Analysis and Blind Signal Separation (ICA 2004), Vol. 3195 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 790–797, 2004.

VI A. Honkela. Approximating Nonlinear Transformations of Probability Distributions for Nonlinear Independent Component Analysis. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IJCNN 2004), pp. 2169–2174, 2004.

VII A. Honkela and H. Valpola. Unsupervised Variational Bayesian Learning of Nonlinear Models. To appear in L. Saul, Y. Weiss, and L. Bottou, eds., *Advances in Neural Information Processing Systems 17*, The MIT Press, Cambridge, MA, USA, 2005.

List of abbreviations

BSS	Blind source separation
EM	Expectation maximisation
FA	Factor analysis
fMRI	Functional magnetic resonance imaging
HMM	Hidden Markov model
HNFA	Hierarchical nonlinear factor analysis
ICA	Independent component analysis
IFA	Independent factor analysis
iid	Independent identically distributed
KL	Kullback–Leibler (divergence)
KPCA	Kernel principal component analysis
MAP	Maximum a posteriori
MCMC	Markov chain Monte Carlo
MDL	Minimum description length
MEG	Magnetoencephalography
ML	Maximum likelihood
MLP	Multilayer perceptron (network)
MML	Minimum message length
MoG	Mixture of Gaussians
NFA	Nonlinear factor analysis
NIFA	Nonlinear independent factor analysis
NP	Nondeterministic polynomial (time)
PCA	Principal component analysis
pdf	Probability density function
PNL	Post-nonlinear
PNLFA	Post-nonlinear factor analysis
RBF	Radial basis function (network)
SSM	State-space model
TAP	Thouless–Anderson–Palmer
VB	Variational Bayesian
VMP	Variational message passing

List of symbols

$\overline{\theta}$	Mean of the parameter θ in the approximating posterior distribution q
$\widetilde{\theta}$	Variance of the parameter θ in the approximating posterior distribution q
$\langle\cdot\rangle$	Expectation, usually over the distribution q unless otherwise noted
$\mathbf{A} = (a_{ij})$	Mixing matrix in linear mixtures
$\mathbf{A}, \mathbf{B}, \mathbf{C}$	Matrices of the nonlinear generative mapping
A, B, C	Propositions
$\mathbf{a}, \mathbf{b}$	Bias vectors of the nonlinear generative mapping
$\mathcal{C}$	The variational Bayesian cost function
$\mathcal{C}_p, \mathcal{C}_q$	Parts of the cost function
$D_{\mathrm{KL}}(q\|\|p)$	The Kullback–Leibler divergence between the two distributions q and p
$\mathrm{diag}(\mathbf{x})$	A diagonal matrix with the elements of vector $\mathbf{x}$ on the main diagonal
$E(\mathbf{s})$	Energy function related to state $\mathbf{s}$
$\mathbf{e}_1, \ldots, \mathbf{e}_n$	The standard basis of $\mathbb{R}^n$
$\exp(\mathbf{x})$	Exponential function applied component-wise to the vector $\mathbf{x}$
$\epsilon_\theta, \epsilon_x$	Coding precisions of the parameters and the data
$\mathbf{f}, \mathbf{g}$	Nonlinear generative mappings
f_i	Post-nonlinear distortions in post-nonlinear ICA
$\mathcal{F}$	Differentiable manifold of all probability distributions
$\mathcal{F}_0$	Manifold of factorisable probability distributions
g_i	Separating nonlinearities in post-nonlinear ICA
H	Number of hidden neurons in an MLP network
$H(x)$	Entropy of the discrete random variable x
$h(x)$	Differential entropy of the continuous random variable x
$\mathcal{H}, \mathcal{H}_i$	The model
$L(x)$	The description length of parameter x
M	The dimensionality of $\mathbf{s}$
N	The dimensionality of $\mathbf{x}$

$\mathbf{n}, \mathbf{n}(t), \mathbf{m}, \mathbf{m}(t)$	Noise terms
$N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian or normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
$N(\mathbf{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma})$	As $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ but for variable $\mathbf{x}$
$p(x)$	The probability of event x, or the probability density function evaluated at point x
$\mathrm{pa}(\theta_i)$	Parents of node θ_i in a Bayesian network
$q(x)$	Approximating probability density function
$\mathbf{s}$	Random vector of the sources
$\mathbf{s}(t)$	Source vector corresponding to the observation vector $\mathbf{x}(t)$
$\boldsymbol{S}$	The set of all source values
T	The number of data samples
θ_i	A parameter of the model
$\boldsymbol{\theta}$	The vector of all model parameters
$\boldsymbol{\theta}_\mathbf{f}, \boldsymbol{\theta}_\mathbf{g}$	The parameters of the nonlinear generative mapping
$\mathbf{u}$	A vector of all the inputs of $\mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f})$
$\mathbf{W}$	Demixing matrix in linear mixtures
$\mathbf{w}(\mathbf{x})$	Demixing function in nonlinear mixtures
$\mathbf{x}$	Random vector of the observations
$\mathbf{x}(t)$	An observed data vector
$\boldsymbol{X}$	The set of all observations
$y_i(t)$	Intermediate values in a nonlinear mapping
ϕ	Scalar activation function $\mathbb{R} \to \mathbb{R}$
$\boldsymbol{\phi}$	A vector of activation functions

Chapter 1

Introduction

1.1 Motivation and overview

The modern society produces enormous amounts of data. This data can contain, for instance, important knowledge on functioning and properties of human genes and brains as well as of many human constructed systems and processes. The long exponential growth of computing power and increasing amounts of available data would enable us to use powerful analysis methods and complicated models to study the data. Unfortunately most widely used analysis methods are still based on limited models that are unable to capture the rich structure of the data presented to them.

One popular method for analysing multichannel measurement data is to transform the signals to a simpler and more meaningful representation, to find the underlying causes of the variation in the data. Latent variable models are a class of statistical models accomplishing this. The assumption behind them is that the observations depend on some unobserved latent variables that explain the essential variation in the data. Depending on the model, the latent variables may be, for instance, structurally simpler or fewer in number than the observed variables.

In signal processing terms, the above problem is often referred to as blind source separation (BSS) as the task is to recover hidden sources of the data blindly, that is with at most very limited additional information on the system generating the data. A popular example is the so-called cocktail party problem: how to focus on a single conversation in an environment with multitude of overlapping voices. BSS is typically an example of unsupervised learning, where the learning system is given the data and it must learn another useful representation for it without the help of an external teacher to provide the correct solution. Learning the model corresponds to statistical estimation of the structure and the parameters. The terms learn and estimate are in general used interchangeably throughout this work.

Linear latent variable models and methods such as state-space models (SSMs),

factor analysis (FA), principal component analysis (PCA) and independent component analysis (ICA) are widely used for data analysis in several applications ranging from engineering and medicine to social sciences, economics and psychology. The linearity means that the effects of different causes on the output are always additive, and equal absolute changes in the causes produce equal changes to the output regardless of the absolute values.

In many of the mentioned applications, there is no theoretical reason to assume the observations to depend linearly on the assumed latent variables. If both trait A and trait B make a person run fast, a person having both of them does not necessarily run twice as fast, even though additivity of the effects in linear models would suggest so. Similarly, a change of temperature from $+5°\mathrm{C}$ to $-5°\mathrm{C}$ can have significantly more drastic effects on many systems than a change from $+30°\mathrm{C}$ to $+20°\mathrm{C}$, although a linear model would predict equal magnitudes of effect in both cases.

Why, then, are linear models used? In most cases, the reasons are probably practical. Linear models are easy to apply and they can provide adequate results for most purposes. After all, a linear approximation of a smooth function is often very accurate locally. Tools for handling nonlinear models are not widely available, as much of the research in statistics before late 20th century has concentrated on methods that do not generalise beyond linear models.

Nonlinear models are significantly more difficult to estimate than linear models. The flexible nonlinear structure introduces new indeterminacies that can make the estimation problems ill-posed. Adding constraints to help resolve these indeterminacies while maintaining the essence of the problem is more difficult, if at all possible. The flexible models are also very susceptible to overfitting, that is modelling all the variation in the data, including noise, and not just the essentials. The solution to these problems used in this work is the variational approximation of Bayesian statistics. The Bayesian framework provides an optimal method of inference under uncertainty but leads to intractable algorithms for most practical problems. The variational approximation is a tractable alternative that seems to preserve most of the important benefits of the fully Bayesian approach. Not all of the problems can, however, be avoided and even with these advanced methods the results should always be verified carefully.

When attempting to gain knowledge of a certain phenomenon, data modelling discussed here only accounts for the latter half of the whole process. Before the modelling, important choices have to be made about data collection, representation and preprocessing. These choices are crucial for the success of the learning task. No amount of work in the latter stages can remedy some early mistakes. These methods are, however, much more application specific, and therefore out of the scope of this work.

The rest of this introductory part is organised as follows. A more mathematical introduction to linear latent variable models is presented in Chapter 2. This serves as a basis for illustrating the Bayesian methods in Chapter 3. Chapters 4 and 5 present reviews of linear and nonlinear blind source separation (BSS) methods, respectively. The variational Bayesian nonlinear BSS method and some of its variants are presented in detail in Chapter 6. Concluding remarks for the thesis

are presented in Chapter 7.

1.2 Contributions of the thesis

The aim of this work has been to develop practical methods of factor analysis and blind source separation for nonlinear mixtures. Much of the work consists of incremental improvements to existing methods. The improvements are formulated in general terms in order to be useful in other kinds of learning problems as well.

The most important scientific contributions of this thesis can be summarised as follows:

- Continuation of the development of a nonlinear blind source separation method based on variational Bayesian learning using a multilayer perceptron (MLP) network to model the nonlinearity.
- A pattern search method to accelerate convergence in cyclic update algorithms such as variational EM algorithms.
- A review of alternative interpretations of the variational Bayesian ensemble learning algorithm and of the benefits they can provide for applying the methods.
- A novel post-nonlinear ICA method capable of handling cases with non-invertible post-nonlinear distortions based on variational Bayesian learning.
- A study on using kernel PCA for initialisation of the nonlinear BSS method and an approach to kernel comparison using the variational method.
- A novel method of approximating the statistics of a nonlinear transform of a probability distribution when the nonlinearity is a layered mapping.

1.3 Contents of the publications and author's contributions

Publication I lays the foundations of the thesis by introducing the basic variational Bayesian nonlinear (independent) factor analysis method. The method uses a multilayer perceptron (MLP) network to model the nonlinearity and a Taylor approximation to evaluate its effects. Promising results are obtained in a difficult pulp process data analysis experiment with 30-dimensional observations. The work for the publication was mostly done by Dr. Harri Valpola (then Lappalainen) with the present author being mainly responsible for implementing the method and performing most of the experiments.

Publication II presents a method to accelerate convergence in optimisation algorithms typically used in variational Bayesian learning. While the improvement is not directly applicable to the method presented in Publication I, it has been

used successfully in closely related hierarchical nonlinear factor analysis method as well as others. The present author derived the algorithm, performed the experiments and wrote most of the paper. Dr. Harri Valpola and Prof. Juha Karhunen discussed the idea and assisted in writing the paper.

The variational Bayesian learning algorithm used in all the publications can be derived either from Bayesian statistics or from information-theoretic considerations. The benefits of this duality are studied in Publication III. Many of the ideas are originally due to Dr. Harri Valpola, who also assisted in writing the paper. The present author was responsible for processing the ideas to a publishable form, performing the experiments and writing most of the paper.

Post-nonlinear (PNL) mixtures provide a theoretically interesting and easier special case of the general nonlinear BSS problem. An adaptation of the general algorithm to this special case is presented in Publication IV. The resulting algorithm is more general than previous PNL algorithms as it is able to handle non-invertible post-nonlinear distortions and noisy mixtures. The work was done jointly with Mr. Alexander Ilin. The present author was responsible for defining and deriving the model and the approximations used. Mr. Ilin implemented the model and performed the experiments. The learning algorithm was developed and the paper written jointly by the authors.

The nonlinear BSS method with its gradient based learning algorithm and flexible multilayer perceptron (MLP) nonlinearity is sensitive to the initialisation used. In Publication I, the initialisation was performed using linear principal components as an initial guess of the sources. In Publication V, nonlinear kernel PCA (KPCA) is used instead of linear PCA in the initialisation. With a proper kernel selected with the help of the variational Bayesian criterion, the method is found to produce better results than the one using linear PCA initialisation. The present author coordinated the work and the experiments, and wrote most of the paper. The idea of using KPCA was proposed by Dr. Harri Valpola. Dr. Stefan Harmeling acted as the kernel expert performing the part of the experiments involving KPCA and writing the relevant part of the paper. Mr. Leo Lundqvist assisted in running the nonlinear BSS experiments.

Experiments showed the method of Publication I having stability problems when using more than 10 sources. This was found out to be caused by inaccuracy of the employed Taylor approximation with large source posterior variances typical with many sources. In Publication VI, a review of other possible approximations is presented and a novel more accurate alternative based on Gauss–Hermite quadratures is proposed.

Publication VII continues upon the work of Publication VI. The accuracy of the proposed new approximation in the context of the nonlinear BSS algorithm is studied in more detail, confirming the suspicions on inaccuracy of the Taylor approximation. The present author derived the approximation and the resulting learning algorithm, performed the experiments and wrote most of the paper. Dr. Harri Valpola discussed the idea and assisted in writing the paper.

Chapter 2

Prologue: Learning from data

A successful learning method consists of two components: a model and a learning algorithm for it. In the probabilistic framework, these can be mostly considered separately, but in practice the choice of one affects some aspects of the other as well. The learning algorithms used in this work are based on Bayesian methods described in Chapter 3. In order to present the theory in a more meaningful context, this chapter contains some general mathematical background on machine learning in Sec. 2.1 and an illustrative example model of factor analysis in Sec. 2.2.

2.1 Models and learning algorithms

This thesis deals with the problem of learning or estimating a model for a given set of observed data. From the point of view of this work, the starting point is always a given set of observed data $\boldsymbol{X}$. This comes usually as a set of real valued vectors $\mathbf{x}(t)$, $t = 1, \ldots, T$, which may or may not have a specific order. Much of the same methodology could, of course, be applied to discrete valued or nominal observations as well.

With the given data set, the goal is to find a model to describe its characteristics or to extract latent information from the data. The models studied in this thesis are parameterised probabilistic models defining a probability model for the data through a likelihood $p(\boldsymbol{X}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of the parameters of the model. For a selected class of models, the learning problem thus reduces to finding such values to these parameters that the model describes the observations well. In traditional statistics this would mean a single vector of values of the parameters. In the Bayesian approach used in this work, the result is instead a posterior distribution over all parameter values. The posterior can be used to evaluate weighted averages over all the values in a way that emphasises the most likely ones. The learning task may also involve selection between or averaging over different discrete model

structures, often a much more difficult problem.

The problems considered in this thesis are examples of *unsupervised learning*, where the learning system is expected to find a somehow useful representation of the data without explicit external guidance. Typical unsupervised learning problems include clustering, that is dividing the observations into disjoint sets of mutually similar elements, and different signal transform problems, where the observations are transformed to a more compact or otherwise more meaningful representation. In statistical terms, this corresponds to modelling the *joint distribution* of the latent variables and the parameters. These can be contrasted with *supervised learning*, where the goal is to simply model the relation between given input and output data, or in statistical terms the *conditional distribution* of output given the input (Haykin, 1999).

From modelling perspective, the models used in unsupervised learning usually have another set of variables $\boldsymbol{S}$, that has different elements $\mathbf{s}(t)$ associated to each observation $\mathbf{x}(t)$, $t = 1, \ldots, T$. The additional latent variables can denote the cluster to which the corresponding observation belongs to or provide otherwise a wholly new representation for the observation. Introduction of new parameters for each observation makes the problem of optimising the parameter values challenging as the number of parameters is typically very large, and further grows as the number T of observation samples is increased.

2.2 Probabilistic modelling and factor analysis

As a concrete example, let us consider the *factor analysis* (FA) model (Harman, 1960). Let $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$ be a vector of observed continuous random variables. They are assumed to be generated from unobserved latent factors $\mathbf{s} = [s_1, s_2, \ldots, s_M]^T$ through a linear mapping

$$x_i = \sum_{j=1}^{M} a_{ij} s_j + n_i, \quad i = 1, 2, \ldots, N \tag{2.1}$$

where $\mathbf{n} = [n_1, n_2, \ldots, n_N]^T$ are additional noise random variables corrupting the observed variables. By assuming that $M < N$, the model can be made to find a more compact representation of the data. The model can be expressed more compactly in vector notation

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n}, \tag{2.2}$$

where $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{N \times M}$ is the matrix of the linear mapping from $\mathbf{s}$ to $\mathbf{x}$, often referred to as the loading matrix. The factors $\mathbf{s}$ and noise $\mathbf{n}$ are assumed to be independent and have a Gaussian distribution with a diagonal covariance. The mean of $\mathbf{s}$ is assumed to be zero and the mean of $\mathbf{n}$ equal to $\boldsymbol{\mu}_n$. The model was first applied by Spearman (1904) to analysis of human intelligence.

For a given set of observations $\boldsymbol{X} = \{\mathbf{x}(t) \mid t = 1, \ldots, T\}$[1], the model translates to

[1] Strictly speaking, the set of observations should consist of tuples $(\mathbf{x}(t), t)$ to maintain the ordering and allow association to the corresponding source tuple $(\mathbf{s}(t), t)$. The shorthand notation is nevertheless used in the interest of brevity.

probability models for $\mathbf{x}(t)$ and the factors $\mathbf{s}(t)$

$$\mathbf{x}(t) \sim N(\mathbf{A}\mathbf{s}(t) + \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n) \tag{2.3}$$
$$\mathbf{s}(t) \sim N(\mathbf{0}, \mathbf{I}), \tag{2.4}$$

where the covariance matrix of the factors is assumed to be an identity matrix $\mathbf{I}$ while the covariance of the noise is an arbitrary positive definite diagonal matrix $\boldsymbol{\Sigma}_n$. The sources $\mathbf{s}(t)$ corresponding to each sample form the (ordered) set $\boldsymbol{S} = \{\mathbf{s}(t) \mid t = 1, \ldots, T\}$. The $(M+2)N$ parameters of the model are $\boldsymbol{\theta} = (\mathbf{A}, \boldsymbol{\Sigma}_n, \boldsymbol{\mu}_n)$.

Using these, the likelihood of the parameters, that is the probability of the data given the parameters may be written as

$$p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}) = \prod_{t=1}^{T} p(\mathbf{x}(t)|\mathbf{s}(t), \boldsymbol{\theta}) = \prod_{t=1}^{T} N(\mathbf{x}(t);\ \mathbf{A}\mathbf{s}(t) + \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n), \tag{2.5}$$

where

$$N(\mathbf{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-N/2} |\det \boldsymbol{\Sigma}|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right) \tag{2.6}$$

denotes the probability density function (pdf) of a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The noises $\mathbf{n}(t)$ and factors $\mathbf{s}(t)$ at different time instances are assumed to be independent identically distributed (iid) random variables, which implies that the vectors $\mathbf{x}(t)$ are independent for different t.

Even though the factor analysis model looks simple, estimating its parameters and the values of the factors requires some care. Individual parts of the model such as the loading matrix $\mathbf{A}$ can be estimated easily if the other parts are assumed to be known. This can be accomplished for instance using the *maximum likelihood* (ML) method of finding such a value for $\mathbf{A}$ that the likelihood (2.5) is maximised. This is equivalent to maximising the logarithm of the likelihood:

$$\begin{aligned} \mathbf{A}_{\mathrm{ML}} &= \arg\max_{\mathbf{A}} \log p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}) = \arg\max_{\mathbf{A}} \sum_{t=1}^{T} \log N(\mathbf{x}(t);\ \mathbf{A}\mathbf{s}(t) + \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n) \\ &= \arg\max_{\mathbf{A}} \sum_{t=1}^{T} -\frac{1}{2}(\mathbf{x}(t) - \boldsymbol{\mu}_n - \mathbf{A}\mathbf{s}(t))^T \boldsymbol{\Sigma}_n^{-1} (\mathbf{x}(t) - \boldsymbol{\mu}_n - \mathbf{A}\mathbf{s}(t)). \end{aligned} \tag{2.7}$$

Joint estimation of several variables is, however, more difficult and requires more advanced algorithms such as the ones discussed in Chapter 3.

The factor analysis problem was first presented a century ago and many efficient techniques for its solution have been presented. The FA model has, however, serious limitations such as non-uniqueness, that restrict its usefulness. By relaxing some of the assumptions of the model, interesting generalisations of FA can be derived that avoid these limitations. Generalisations based on relaxing the assumption of Gaussianity of $\mathbf{s}$ are presented in Chapter 4, while methods with a nonlinear mapping $\mathbf{f}$ with parameters $\boldsymbol{\theta}_{\mathbf{f}}$ from $\mathbf{s}$ to $\mathbf{x}$, leading to data model

$$\mathbf{x} = \mathbf{f}(\mathbf{s}, \boldsymbol{\theta}_{\mathbf{f}}) + \mathbf{n} \tag{2.8}$$

along with a potentially more general model for $\mathbf{s}$, are studied in Chapters 5 and 6, as well as in most of the publications.

Chapter 3

Bayesian inference

In this work, problems such as learning the factor analysis model presented in Section 2.2 are solved by methods of Bayesian statistics. The Bayesian methods provide a mathematically sound and robust machinery for dealing with such problems.

This chapter starts with a brief introduction to Bayesian inference in Sec. 3.1. Together with assorted additional tools presented in Sec. 3.2, this gives the theoretical background for the rest of the thesis. Exact Bayesian inference is typically computationally intractable, leading to the need for different approximation schemes presented in Sec. 3.3. For the purposes of this thesis, the variational approximation is the most important approximation. Several alternative interpretations for it are presented in Sec. 3.4. The information-theoretic interpretation is also discussed in more detail in Publication III. Actual learning algorithms based on the variational approximation are presented in Sec. 3.5. Many of the variational and other learning algorithms can be interpreted as optimisation of a given cost function. This aspect is studied in Sec. 3.6, together with presentation of a method to accelerate convergence in alternating optimisation algorithms as presented in Publication II.

3.1 Introduction to Bayesian inference

This section presents a brief introduction to some basic concepts of Bayesian inference. More thorough discussions can be found in many of the cited books.

3.1.1 Bayesian philosophy

The Bayesian approach is a theory of subjective probability. In the Bayesian approach, probability is a measure of the credibility of an uncertain event. Probabilities are not universal, they depend on the subject and his prior knowledge. Probabilities can be used to model all uncertainty, even though the phenomenon behind the event would be completely deterministic.

The subjectivity of Bayesian probabilities is noted by marking all probabilities as conditional to something. There is no such thing as absolute probability. The probability of heads in coin tossing, for example, is conditional to assumptions on weight distribution of the coin, mechanical parameters of the toss and other things that should be noted when specifying the probability. In Bayesian modelling, this is shown by the explicit dependence of the specified probabilities on the model $\mathcal{H}$.

Probabilities do not arise out of nothing. In order to evaluate the probability of a proposition given some evidence, a prior probability of the proposition must be specified. This is actually very natural, as it is impossible to interpret new evidence without *any* prior assumptions. In Bayesian analysis, these assumptions are simply written in the form of a prior probability.

3.1.2 Mathematical foundations

The rules of Bayesian probability theory can be derived from Cox axioms that represent basic requirements for sensible reasoning under uncertainty (Cox, 1946). The axioms and their most important consequences will be introduced here briefly. Similar results can also be derived from a slightly different starting point of optimal decision making (Bernardo and Smith, 2000).

Let A, B, C be propositions whose probabilities are to be evaluated. The propositions may assert, for instance, the occurrence of certain event or something else such as a statement of the value of a physical constant. The propositions are assumed to follow the laws of Boolean logic. Let the symbol $B \mid A$ denote some measure of reasonable credibility of proposition B, when proposition A is known to be true (Cox, 1946).

Cox axioms can now be stated as follows (MacKay, 2003, Ch. 2):

Axiom 1 The credibilities of propositions can be ordered.

Axiom 2 The credibility of conjunction of propositions C and B given A can be evaluated as a function of the credibility of B given A and the credibility of C given A and B, that is

$$(C \wedge B) \mid A = F(C \mid (B \wedge A), B \mid A). \tag{3.1}$$

Axiom 3 The credibility of the negation of proposition B given A only depends on the credibility of B given A, that is

$$\neg B \mid A = S(B \mid A). \tag{3.2}$$

Axiom 1 implies that real numbers can be used to denote the credibilities. Axiom 2 states that the credibility of the joint truth of two propositions can be evaluated sequentially as a function of what is the credibility of the first and what is the credibility of the second, given that the first is true. Axiom 3 requires a similar law for the negation of a proposition.

It can be shown (Cox, 1946) that with sufficient smoothness assumptions the only solutions for equations (3.1) and (3.2) are homeomorphic to $F(x, y) = xy$ and

$S(x) = 1 - x$. The credibilities of this standard representation can now be called probabilities, denoted by $p(B|A)$, and they obey the well known sum and product rules:

$$p(B|A) + p(\neg B|A) = 1 \tag{3.3}$$

$$p(C, B|A) = p(C|B, A)\, p(B|A). \tag{3.4}$$

3.1.3 Bayes' theorem and marginalisation principle

The Bayes' theorem

$$p(C|B, A) = \frac{p(B|C, A)p(C|A)}{p(B|A)} \tag{3.5}$$

can be derived as a corollary of the product rule (3.4). It shows how the probabilities should be updated in light of new information B and thus forms a basis for inference.

In Eq. (3.5), the proposition A denotes some background assumptions that underlie the whole inference. The object of interest is proposition C whose probability is re-evaluated in light of B. The term $p(B|C, A)$ is called the *likelihood* and $p(C|A)$ is the *prior probability* of C. When multiplied together and scaled properly they yield the *posterior probability* $p(C|B, A)$.

The other major principle implied by Eqs. (3.3) and (3.4) needed for inference is the marginalisation principle that tells how to handle undesired extra variables. Assuming propositions $C_1, \ldots, C_n$ are mutually exclusive and

$$\sum_{i=1}^{n} p(C_i|A) = 1, \tag{3.6}$$

the marginalisation principle can be written as

$$p(B|A) = \sum_{i=1}^{n} p(B, C_i|A) = \sum_{i=1}^{n} p(B|C_i, A)p(C_i|A). \tag{3.7}$$

This can be used to evaluate the denominator of Eq. (3.5) to yield

$$p(C|B, A) = \frac{p(B|C, A)p(C|A)}{\sum_i p(B|C_i, A)p(C_i|A)}, \tag{3.8}$$

where C_i represent all possible values of C.

3.1.4 The continuous case

The analysis presented so far deals only with the case where the number of possible different events is finite. In order to extend it to the case with an infinite number of events, additional mathematics from measure theory is required. The measure-theoretic details are not explicitly required in this work, but they are included here for completeness. A continuous generalisation of Cox axioms implies Kolmogorov

axioms as theorems and all the results of standard probability theory thus follow easily (Jaynes, 2003). Williams (1991) presents a nice general mathematical introduction to probability. The decision-theoretic approach to probability (Bernardo and Smith, 2000) also generalises to the continuous case.

The basic concept of mathematical probabilities is the *probability space* $(\Omega, \mathcal{F}, p)$, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra on Ω called the family of events and p is a countably additive probability measure on $(\Omega, \mathcal{F})$ satisfying $p(\emptyset) = 0$ and $p(\Omega) = 1$, where $\emptyset$ is the empty set. A *random variable* on $(\Omega, \mathcal{F}, p)$ is a measurable function $x : \Omega \to \mathbb{R}$ with respect to the Borel measure on $\mathbb{R}$.

A random variable x induces a probability measure $p_x : \mathcal{B} \to [0, 1]$ on Borel sets $\mathcal{B}$ of $\mathbb{R}$. This allows defining the *distribution function*

$$F_x(c) = p_x((\infty, c]) = p(x \leq c). \tag{3.9}$$

If the random variable is absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}$ as is the case with all the random variables encountered in this thesis, there exists a non-negative *probability density function* (pdf) $p_x : \mathbb{R} \to \mathbb{R}$ such that

$$p_x(B) = \int_B p_x(t)\, dt, \quad B \in \mathcal{B}. \tag{3.10}$$

The function p_x here is the Radon-Nikodyn derivative (Rudin, 1987) of the measure p_x with respect to the Lebesgue measure. To simplify notation, it shall henceforth be denoted by p. The definitions generalise naturally to *random vectors* $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ that are simply measurable functions $\mathbf{x} : \Omega \to \mathbb{R}^n$ to $\mathbb{R}^n$ instead of $\mathbb{R}$.

It can be shown that the sum and product rules, Bayes' theorem and marginalisation principle apply in the continuous case as well, with the summations replaced by corresponding integrals. The exact formulas are presented in more detail in the next section.

3.1.5 Basic continuous Bayesian modelling

Taking the factor analysis model from Sec. 2.2 as an example, the exact Bayesian learning procedure is now illustrated. Following the earlier discussion, the likelihood $p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H})$ of the parameters of the model for a given data set $\boldsymbol{X}$ can be written as in Eq. (2.5). As discussed in Sec. 3.1.1, an explicit dependence on the assumed factor analysis model is included here through $\mathcal{H}$.

Bayesian analysis additionally requires the specification of the prior probability $p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})$ of $\boldsymbol{S}$ and $\boldsymbol{\theta}$. The different sets of parameters are typically assumed independent so that $p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H}) = p(\boldsymbol{S}|\mathcal{H})p(\boldsymbol{\theta}|\mathcal{H})$. The prior for $\boldsymbol{S}$ is defined through Eq. (2.4). In case of the parameters the prior is usually chosen to be of simple *conjugate* form as discussed in Sec. 3.2.2. Multiplying the likelihood and the priors together yields the joint distribution of all the variables, $p(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})$. The form of the joint distribution completely specifies a model in Bayesian statistics.

All information on the parameters $\boldsymbol{\theta}$ and latent variables $\boldsymbol{S}$ given by the data $\boldsymbol{X}$ is given by the posterior distribution $p(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$. This distribution can be

evaluated using the Bayes' theorem

$$p(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H}) = \frac{p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})}{p(\boldsymbol{X}|\mathcal{H})}. \tag{3.11}$$

The denominator $p(\boldsymbol{X}|\mathcal{H})$ can be evaluated by marginalisation

$$p(\boldsymbol{X}|\mathcal{H}) = \iint_{\boldsymbol{S},\boldsymbol{\theta}} p(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})\, d\boldsymbol{S}\, d\boldsymbol{\theta}, \tag{3.12}$$

yielding an explicit formula for the posterior

$$p(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H}) = \frac{p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})}{\iint_{\boldsymbol{S},\boldsymbol{\theta}} p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H})\, d\boldsymbol{S}\, d\boldsymbol{\theta}}. \tag{3.13}$$

By ignoring the normalising denominator, Eq. (3.11) can also be written in a simplified form

$$p(\boldsymbol{S}, \boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H}) \propto p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H}). \tag{3.14}$$

For the remainder of this chapter, the symbol $\boldsymbol{\theta}$ shall be used to denote all the unknown variables of the model, including both parameters and latent variables.

3.1.6 Predictive inference

The development of Bayesian inference was justified through optimal inference under uncertainty and optimal decision making. Let us assume there is a quantity $\mathbf{x}$ whose value is to be predicted. There is a model for $\mathbf{x}$ specified through some parameters $\boldsymbol{\theta}$ with a prior $p(\boldsymbol{\theta}|\mathcal{H})$ and a likelihood $p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{H})$. The model defines a prior predictive distribution for $\mathbf{x}$ through averaging using marginalisation

$$p(\mathbf{x}|\mathcal{H}) = \int_{\boldsymbol{\theta}} p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{\theta}|\mathcal{H})\, d\boldsymbol{\theta}. \tag{3.15}$$

Additional observations $\boldsymbol{X}$ of the process generating $\mathbf{x}$ allow updating the prior of the parameters $p(\boldsymbol{\theta}|\mathcal{H})$ to the posterior $p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$ using the Bayes' theorem as shown above in Eq. (3.11). The posterior of the parameters can then be used to define the posterior predictive distribution of $\mathbf{x}$ through

$$p(\mathbf{x}|\boldsymbol{X}, \mathcal{H}) = \int_{\boldsymbol{\theta}} p(\mathbf{x}|\boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})\, d\boldsymbol{\theta}. \tag{3.16}$$

Here it is assumed that the only dependence of $\mathbf{x}$ on $\boldsymbol{X}$ is through the model and its parameters $\boldsymbol{\theta}$. This distribution yields the optimal predictions of the value of $\mathbf{x}$ given the model and the data $\boldsymbol{X}$. Optimal decisions on selecting actions can be made in this framework by evaluating the expected utilities of all actions by averaging over the posterior predictive distributions of different actions. This is probably the most important use of the posterior as making good decisions is, after all, usually the final goal of modelling and analysis.

3.1.7 Model comparison

The model describing the observation samples best is not necessarily the best model of the underlying phenomenon generating the data. In light of limited information on the phenomenon, it is better to use a simpler model that is probably approximately correct than a very complex model that is almost certainly seriously wrong. A simpler model is more likely to lead to reasonable predictions of future data and more reasonable analysis to causes of the observations. This is the reasoning behind the principle of Occam's Razor: the simplest adequate explanation of natural things should be preferred.

In the Bayesian approach, the Occamian effect is achieved by averaging. If there is not enough evidence to support a single complex model, different but equally likely other models will smoothen out its predictions. The averaging works naturally over many levels of model parameters and possibly different models. As the averaging procedure is computationally demanding, some averages are often replaced by selection of point estimates. The problem of finding the best single model over many candidates leads to the study of model comparison and selection.

In traditional maximum likelihood approaches, the parameters $\boldsymbol{\theta}$ of a model can be estimated by maximising the likelihood $p(\boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H})$, as in Eq. (2.7). The likelihood cannot, however, be used to compare different models, because in most situations a more complicated model always yields a higher likelihood. This leads to overfitting as the model is eventually able to model everything in the training data set perfectly. This has prompted the introduction of several different criteria for model comparison, most of them arising as some kind of approximations from exact Bayesian reasoning (Bishop, 1995).

In contrast, model comparison in the Bayesian framework is very easy. Different models $\mathcal{H}_1, \mathcal{H}_2, \ldots$ can be considered different propositions and their posterior probabilities $p(\mathcal{H}_i|\boldsymbol{X})$ can be evaluated with the standard procedure as $p(\mathcal{H}_i|\boldsymbol{X}) \propto p(\boldsymbol{X}|\mathcal{H}_i)p(\mathcal{H}_i)$. The key term $p(\boldsymbol{X}|\mathcal{H}_i)$ is called the *evidence* or *marginal likelihood*. It is evaluated by marginalising the joint distribution of the data and parameters over the parameters

$$p(\boldsymbol{X}|\mathcal{H}_i) = \int_{\boldsymbol{\theta}} p(\boldsymbol{X}, \boldsymbol{\theta}|\mathcal{H}_i)\, d\boldsymbol{\theta} = \int_{\boldsymbol{\theta}} p(\boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H}_i) p(\boldsymbol{\theta}|\mathcal{H}_i)\, d\boldsymbol{\theta}. \tag{3.17}$$

Averaging over all parameter values in Eq. (3.17) is the key to the success of Bayesian model comparison. It is not necessary to use the priors $p(\mathcal{H}_i)$ to penalise complex models, the evidence does it automatically. This is illustrated in Fig. 3.1. Its left subfigure shows the behaviour of an intuitive goodness measure of models as a function of the complexity of the model. The behaviour of evidence in model comparison is illustrated on the right subfigure in a complementary view of evidence of three selected models as function of the data set. This figure, due to MacKay (1992), shows how simple models distribute high evidence to a small class of data sets while complex models distribute lower evidence to a larger class of data sets. The optimum is typically found between these extremes.

Using a point estimate of a single model instead of averaging over all possibilities usually provides reasonable results for instance for the number of factors in a

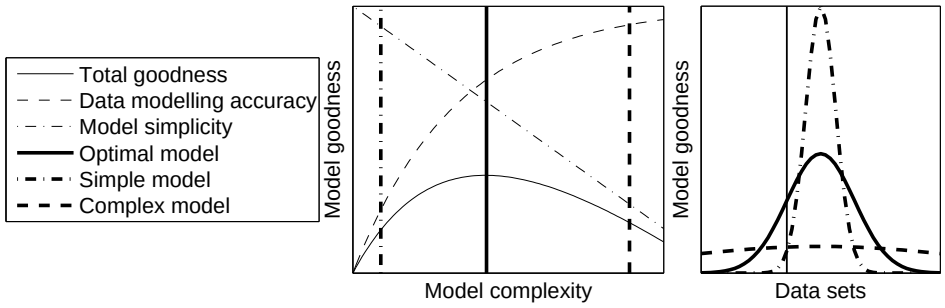

Figure 3.1: Bayesian model comparison through evidence is illustrated. The left subfigure shows an intuitive goodness of a model as a function of model complexity. Simple models provide an inadequate description of the data while with complex models the data is overfitted. An optimal model is found between the two extremes. The evidences $p(\boldsymbol{X}|\mathcal{H}_i)$ of three selected models (denoted vertical lines in the left subfigure) over all possible data sets are illustrated in the right subfigure, with the present data set denoted by a solid vertical line.

factor analysis or related model, where it is not completely unreasonable to assume an underlying true factor model. In this case the model space under study is essentially closed as it can be seen to contain the true model. This is an easier situation than an open model space, where it is not reasonable to think of any of the models as a correct one and it is therefore more difficult to interpret probabilities assigned to them (Bernardo and Smith, 2000).

3.2 Additional tools and concepts

This section presents additional general theoretical and practical tools that are used later.

3.2.1 Independence

Much of this work deals with random vectors and especially random vectors with independent components. Statistical independence of two random variables or components of a random vector means that information on one of them gives no additional information on the other. If a random vector $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ has a probability density, the independence of its components can be characterised by the form of the density as a product of marginals of the independent components

$$p(\mathbf{x}) = p(x_1)p(x_2)\cdots p(x_n). \tag{3.18}$$

This can be compared to a corresponding factorisation for a general random vector

$$p(\mathbf{x}) = p(x_1|x_2, x_3, \ldots, x_n)p(x_2|x_3, \ldots, x_n)\cdots p(x_n). \tag{3.19}$$

3.2.2 Conjugate models

Conjugate priors are often used to simplify inference. For a given class of likelihood functions $p(\boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H})$, the class $\mathcal{P}$ of priors $p(\boldsymbol{\theta}|\mathcal{H})$ is called a *conjugate* if the posterior $p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$ is of the same class $\mathcal{P}$.

This is a very useful property if the class $\mathcal{P}$ consists of a set of probability densities with the same functional form. In such a case the posterior distribution will also have the same functional form. Taking for instance a model for the mean of scalar Gaussian observations $\boldsymbol{X} = \{x(1), x(2), \ldots, x(T)\}$,

$$p(x(t)|\mu, \sigma_x^2, \mathcal{H}) = N(x(t);\ \mu, \sigma_x^2), \tag{3.20}$$

where the mean parameter μ is unknown and variance σ_x^2 is known. If the prior of μ is chosen to be Gaussian

$$p(\mu|\mathcal{H}) = N(\mu;\ \mu_\mu, \sigma_\mu^2), \tag{3.21}$$

the posterior will also be Gaussian

$$p(\mu|\boldsymbol{X}, \mathcal{H}) = N(\mu;\ \mu_1, \sigma_1^2) \tag{3.22}$$

with $\sigma_1^2 = \left[(\sigma_\mu^2)^{-1} + (n\sigma_x^2)^{-1}\right]^{-1}$ and $\mu_1 = (\sigma_1^2)^{-1}\left(\frac{\mu_\mu}{\sigma_\mu^2} + \frac{\sum_i x_i}{\sigma_x^2}\right)$ (Gelman et al., 1995).

3.2.3 Entropy and Kullback–Leibler divergence

It is often convenient to measure the uncertainty related to a random variable x with distribution $p(x)$. This is accomplished by the *entropy* of the distribution, which in the discrete case is

$$H(x) = -\sum_i p(x_i) \log p(x_i). \tag{3.23}$$

This measure of information was first introduced by Shannon (1948) and shown by him to provide a lower bound to the number of bits needed on average to communicate information on events following $p(x)$.[1] The discrete entropy is always non-negative and for finite sample spaces it is maximised by a uniform distribution $p(x_i) = 1/n$ that has the entropy $H(x) = \log n$.

The discrete entropy can be generalised to continuous *differential entropy* by replacing the summation with an integral

$$h(x) = -\int_{\mathbb{R}} p(x) \log p(x)\, dx. \tag{3.24}$$

The differential entropy differs from discrete entropy in that it is not an absolute quantity and reparameterisation of the variable usually changes the entropy (Shannon, 1948). Differential entropy has no lower bound and it may well be negative.

[1] Entropy can be measured in *bits* if base-2 logarithm is used in Eq. (3.23). In this thesis, natural base-e logarithm is used instead to yield measures in *nats*. Nats can always be converted to bits by dividing by $\ln 2$.

Nevertheless, it is often used to measure the information content of a continuous distribution. For random variables with a given variance that have a differential entropy, the entropy is maximal for those with a Gaussian distribution. Entropy can therefore be used as a measure of non-Gaussianity of a distribution.

As entropy measures the information content of a distribution, the information for discriminating two distributions p and q can be measured with the *relative entropy* or *Kullback–Leibler divergence* $D_{\mathrm{KL}}(p||q)$ (Kullback and Leibler, 1951). The divergence $D_{\mathrm{KL}}(p||q)$ measures how many more bits are needed to communicate information on events following p when using a code for distribution q (Cover and Thomas, 1991). The Kullback–Leibler divergence is defined in the continuous case by

$$D_{\mathrm{KL}}(p||q) = \int_{\mathbb{R}} p(x) \log \frac{p(x)}{q(x)} \, dx \tag{3.25}$$

or by corresponding summation in the discrete case. As Eq. (3.25) shows, the divergence is not symmetric. It also does not satisfy triangle inequality, so it is clearly not a metric. The divergence is, however, non-negative and only attains the value zero when $p = q$ almost everywhere. It is also invariant with respect to invertible reparameterisations of the variable (Kullback and Leibler, 1951).

3.2.4 Graphical models and Bayesian networks

The dependence relations of a probabilistic model can be easily represented by a directed acyclic graph called Bayesian network (Cowell, 1999). A very simple example of a Bayesian network visualising the factor analysis model is presented in Fig. 3.2. The connections of the network illustrate that $\mathbf{x}$ depends on $\mathbf{s}$ and $\mathbf{A}$, which are in turn independent. The shading of node $\mathbf{x}$ denotes an observed variable. This corresponds to a representation of the joint distribution of the variables by

$$p(\mathbf{x}, \mathbf{s}, \mathbf{A}|\mathcal{H}) = p(\mathbf{x}|\mathbf{s}, \mathbf{A}, \mathcal{H}) p(\mathbf{s}|\mathcal{H}) p(\mathbf{A}|\mathcal{H}). \tag{3.26}$$

The same procedure can be carried out in general, and the joint distribution of the variables in a Bayesian network is a product of local conditional distributions $p(\theta_i|\mathrm{pa}(\theta_i))$. Here $\mathrm{pa}(\theta_i)$ denotes the parents of θ_i, that is nodes from which edges to θ_i originate.

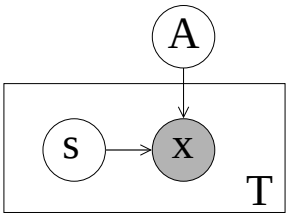

Figure 3.2: A Bayesian network representation of the factor analysis model. The shaded node represents observed variable and the box denotes corresponding variables occurring T times in iid pairs.

The form of Eq. (3.26) can be compared to a general expression following Eq. (3.19),

$$p(\mathbf{x}, \mathbf{s}, \mathbf{A}|\mathcal{H}) = p(\mathbf{x}|\mathbf{s}, \mathbf{A}, \mathcal{H})p(\mathbf{s}|\mathbf{A}, \mathcal{H})p(\mathbf{A}|\mathcal{H}). \tag{3.27}$$

The main additional information provided by the Bayesian network here is the independence of $\mathbf{s}$ and $\mathbf{A}$ given the model.

3.3 Approximate inference

Let us consider the problem of inferring the parameters $\boldsymbol{\theta}$ of a model for data set $\boldsymbol{X}$. All additional assumptions of the model are denoted by $\mathcal{H}$. Exact Bayesian inference of the posterior $p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$ using Eq. (3.13) requires evaluating an integral over all possible values of the parameters $\boldsymbol{\theta}$. This possibly high dimensional integral is almost always analytically intractable and hence some approximation methods must be used.

The simplest and traditionally most popular approximations are point estimates that use only a single representative value of the parameters $\boldsymbol{\theta}$. This value can be obtained by either maximising the value of the likelihood $p(\boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H})$ as in the maximum likelihood (ML) method or the posterior pdf $p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$ as in the maximum *a posteriori* (MAP) method. The point estimates are relatively easy to evaluate at least in theory and can provide sufficient results for simple models and problems, but they are often too crude for more difficult problems. Even Laplace approximation, where a Gaussian is fitted to the posterior around the MAP point cannot help, if the MAP estimate itself is inaccurate. It can nevertheless be used to approximate integrals needed for instance for evidence that otherwise could not be evaluated at all using the point estimates (MacKay, 2003, Ch. 22 and 27).

Finding the ML or MAP estimate directly can be difficult in practice for models like factor analysis with different types of parameters $\boldsymbol{S}$ and $\boldsymbol{\theta}$. The problem can be simplified by the EM algorithm (Dempster et al., 1977), which alternates between the following two steps (Neal and Hinton, 1999):

E-step: Find the posterior of $\boldsymbol{S}$ given a current estimate of the parameters $\boldsymbol{\theta}^{(t-1)}$,

$$q^{(t)}(\boldsymbol{S}) = p(\boldsymbol{S}|\boldsymbol{X}, \boldsymbol{\theta}^{(t-1)}, \mathcal{H}).$$

M-step: Find $\boldsymbol{\theta}^{(t)}$ maximising the expected log-likelihood,

$$\boldsymbol{\theta}^{(t)} = \arg\max_{\boldsymbol{\theta}} \left\langle \log p(\boldsymbol{S}, \boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H}) \right\rangle_{q^{(t)}(\boldsymbol{S})},$$

where $\langle\rangle_q$ denotes the expectation over q.

The initial value $\boldsymbol{\theta}^{(0)}$ of the parameters can be selected randomly or using a more refined method if the model is prone to local optima. The EM algorithm is guaranteed not to decrease the log-likelihood $p(\boldsymbol{X}|\boldsymbol{\theta}, \mathcal{H})$. It will converge to a local maximum of the log-likelihood except in some very special cases (Dempster et al., 1977; Neal and Hinton, 1999). The convergence can be very slow in some cases, as discussed in Sec. 3.6.1.

3.3.1 Stochastic approaches

If a single point is insufficient in describing a complicated posterior distribution, what about a sample of points from the distribution? By increasing the sample size, the distribution can be described and required predictions evaluated to arbitrary accuracy.

As the posterior distribution cannot be easily specified without intractable integrals, it is usually not possible to sample from it directly. There are, however, several methods that allow sampling from a distribution with unknown scaling coefficient, as is most often the case. The most popular of these are the Markov chain Monte Carlo (MCMC) methods that are based on defining a Markov chain whose limiting distribution is the desired posterior (Gelman et al., 1995; MacKay, 2003). The most well-known of such methods are the *Metropolis-Hastings algorithm* and the *Gibbs sampler*.

The MCMC methods are very general, but they are often computationally very demanding. The only guarantee for the methods is that the distribution of the Markov chain will eventually converge to the posterior, but it is often very difficult to assess this convergence in practice (MacKay, 2003, Ch. 29). Getting independent samples from the posterior is also very difficult because of the dependencies between consecutive samples of the Markov chain.

3.3.2 Variational and naïve mean field methods

In addition to statistics, the problem of approximating a probability distribution has been studied in the context of statistical mechanics since early 20th century. In fact, the above-mentioned Metropolis algorithm was first proposed for simulating physical systems (Metropolis et al., 1953).

In this context, simple discrete particle systems can be abstracted to a vector $\mathbf{s}$ of M variables (particles) s_i that have binary values (± 1, spins) and a related energy function (Hamiltonian). In case of popular Ising and spin glass models[2], the Hamiltonian may be written as

$$E(\mathbf{s}) = -\sum_{i,j} J_{ij} s_i s_j - \sum_i h_i s_i, \tag{3.28}$$

where J_{ij} are interaction parameters between particles i and j, and h_i is an external field strength affecting s_i (Parisi, 1988; Mezard et al., 1987; MacKay, 2003). As in all physical systems, the system attempts to attain a state of minimal energy. In a thermal equilibrium, the probability of a configuration $\mathbf{s}$ follows the Boltzmann distribution

$$p(\mathbf{s}) = \frac{1}{Z} \exp\left(-\beta E(\mathbf{s})\right), \tag{3.29}$$

where β is a constant related to the temperature of the system and

$$Z = \sum_{\mathbf{s}} \exp(-\beta E(\mathbf{s})) \tag{3.30}$$

[2]In neural network literature, the same models are called *Boltzmann machines*.

is a normalising constant called partition function. The distribution $p(\mathbf{s})$ may be derived for instance from the variational principle of minimising the free energy (Parisi, 1988)

$$\begin{aligned}\mathcal{F}(q(\mathbf{s})) &= \langle E(\mathbf{s})\rangle - \frac{1}{\beta}H_q(\mathbf{s}) = \left\langle E(\mathbf{s}) + \frac{1}{\beta}\log q(\mathbf{s})\right\rangle \\ &= \frac{1}{\beta}\left\langle \log\frac{q(\mathbf{s})}{\exp(-\beta E(\mathbf{s}))}\right\rangle,\end{aligned} \tag{3.31}$$

where $\langle\cdot\rangle$ denotes expectation over the distribution $q(\mathbf{s})$ and $H_q(\mathbf{s})$ is the entropy of $q(\mathbf{s})$ as defined in Eq. (3.23). This can be shown to have its sole minimum for $q(\mathbf{s}) = p(\mathbf{s})$ (MacKay, 2003, Ch. 33).

Even though the model associated with the energy function (3.28) seems simple, it is very difficult to solve. The Boltzmann distribution (3.29) is very similar in form to the posterior distribution in Bayesian analysis and its evaluation suffers from the same difficulties, especially the sum over an exponentially growing space of $\mathbf{s}$ needed to evaluate the normalisation term Z. Even the problem of determining the "point estimate" of ground state $\mathbf{s}_0$ minimising $E(\mathbf{s})$ is NP-complete (Istrail, 2000).

The most popular approximation methods for solving the Ising model are the *mean field methods*. The simplest naïve mean field method uses a factorial approximation $q(\mathbf{s}) = \prod_i q(s_i)$ which is fitted to minimise the free energy (3.31) (Parisi, 1988). The name of the methods stems from replacing the specific interactions between the particles with an average effect, a mean field. In case of the Ising model with the approximation $q(s_i) = (1 + s_i m_i)/2$ with variational parameters m_i, the mean field solution can be found from a set of relatively simple coupled nonlinear equations (MacKay, 2003, Ch. 33)

$$m_i = \tanh\left(\beta\left[\sum_j J_{ij}m_j + h_i\right]\right), \quad i = 1,\ldots,M. \tag{3.32}$$

More advanced methods model some of the correlations between the variables. Some of these are discussed in Sec. 3.3.3.

From mean field to variational approximation

In statistical physics, the nature of the model and its parametrisation are often determined by physical considerations. This differs greatly from more general models of Bayesian statistics, where there are often several ways to parameterise essentially the same model. The same basic principles can nevertheless be applied there as well.

The approximation implied by the mean field method can be derived from probabilistic viewpoint as a variational approximation to the model evidence $p(\boldsymbol{X}|\mathcal{H})$ by introducing the approximating distribution $q(\boldsymbol{\theta})$ into the integral and applying

Jensen's inequality:

$$\begin{aligned} -\log p(\boldsymbol{X}|\mathcal{H}) &= -\log \int p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})\, d\boldsymbol{\theta} = -\log \int \frac{p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})}{q(\boldsymbol{\theta})} q(\boldsymbol{\theta})\, d\boldsymbol{\theta} \\ &\leq \int -\log \left(\frac{p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})}{q(\boldsymbol{\theta})} \right) q(\boldsymbol{\theta})\, d\boldsymbol{\theta}. \end{aligned} \tag{3.33}$$

This yields the cost function or variational free energy

$$\mathcal{C} = \int \log \left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})} \right) q(\boldsymbol{\theta})\, d\boldsymbol{\theta} = \left\langle \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})} \right\rangle . \tag{3.34}$$

The distribution $q(\boldsymbol{\theta})$ may be fully factorial as in the naïve mean field approach, but simple dependencies are often modelled as part of it. The specific variational approximation with the cost function (3.34) is often called Bayesian *ensemble learning* (MacKay, 1995), but it should not be mixed with ensemble averaging methods used with committee machines (Haykin, 1999).

In order to find out when there is equality in Eq. (3.33), it is easier to write the right hand side as

$$\begin{aligned} \mathcal{C} &= \int \log \left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H}) p(\boldsymbol{X}|\mathcal{H})} \right) q(\boldsymbol{\theta})\, d\boldsymbol{\theta} \\ &= -\log p(\boldsymbol{X}|\mathcal{H}) + \int \log \left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H})} \right) q(\boldsymbol{\theta})\, d\boldsymbol{\theta} \\ &= -\log p(\boldsymbol{X}|\mathcal{H}) + D_{\mathrm{KL}}\left(q(\boldsymbol{\theta}) || p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H}) \right), \end{aligned} \tag{3.35}$$

where $D_{\mathrm{KL}}(q||p)$ is the Kullback–Leibler divergence between the distributions q and p. Hence, the variational free energy will be equal to the negative log evidence when the approximating distribution is equal to the true posterior: $q(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H})$. Minimising the free energy or cost function $\mathcal{C}$ is equivalent to minimising the Kullback–Leibler divergence $D_{\mathrm{KL}}(q||p)$.

As the cost $\mathcal{C}$ provides a lower bound on the evidence, it can be used directly as a criterion for model comparison. It can also be shown that the result of using $\exp(-\mathcal{C}_i)$ evaluated for the model $\mathcal{H}_i$ in place of the evidence $p(\boldsymbol{X}|\mathcal{H}_i)$ corresponds to applying the same ensemble learning principle to a higher level learning problem between different models $\mathcal{H}_i$ (Lappalainen and Miskin, 2000).

There are several extensive tutorials on variational Bayesian (VB) methods and ensemble learning (Jordan et al., 1999; Lappalainen and Miskin, 2000; Ghahramani and Beal, 2001b; Jaakkola, 2001). Ensemble learning is also not the only variational method, there are others based on performing different variational transformations that introduce additional conditional independencies to make the inference easier (Jordan et al., 1999; Rustagi, 1976), but leading to a different optimisation criterion than Eq. (3.34). An actual example of such a technique is presented for instance by Girolami (2001).

3.3.3 Other deterministic approximations

While the simple variational approximation has been successfully applied to many learning problems, it has certain shortcomings. The method is based on approxi-

mating the model evidence, and can usually provide a reasonable approximation of it. This does not, however, imply that other quantities such as the marginals $q(\theta_i)$ of the approximate posterior $q(\boldsymbol{\theta})$ or their moments would be even close to those of the true posterior. An example of how the variational approximation cannot model the marginals is presented by Jaakkola (2001). MacKay (2003, Ch. 33) presents an example of qualitatively incorrect estimates of certain physical quantities of an Ising model. More realistic statistical examples include inability to separate the components in a linear ICA model (Højen-Sørensen et al., 2002; Ilin and Valpola, 2003) and biased parameter estimation in linear state-space models (Wang and Titterington, 2004).

The shortcomings of the naïve approximation stem from ignoring the dependencies between the variables. The simplest way to get a better approximation is to explicitly estimate the covariances $\langle s_i s_j \rangle - \langle s_i \rangle \langle s_j \rangle$. One of the simplest methods for this is to apply the linear response theory (Parisi, 1988; Opper and Winther, 2001). In case of the Ising model, this means differentiating the identity arising from (3.29)

$$\langle s_i \rangle = \frac{1}{Z} \sum_{\mathbf{s}} s_i \exp\left(-\beta E(\mathbf{s})\right) \tag{3.36}$$

with respect to h_j, $i \neq j$ to get (Parisi, 1988)

$$\langle s_i s_j \rangle - \langle s_i \rangle \langle s_j \rangle = \frac{1}{\beta} \frac{\partial \langle s_i \rangle}{\partial h_j}. \tag{3.37}$$

The evaluated covariance is correct only if the expectations are taken over p, but even expectations over q lead to reasonable approximations, even though q does not include the correlations. The approximation q in the linear response approach therefore consists of a factorised part that is updated in the usual way and the additional correlations that are evaluated separately. All expectations over q are evaluated using the version with additional correlations. More details on applying linear response theory to Ising models or Boltzmann machines are presented by Kappen and Rodríguez (1998).

The self-inconsistency of the linear response approach has led the mean field researchers to other advanced methods. The most famous of these are the TAP equations named after Thouless, Anderson and Palmer. The TAP equations can be derived for example as a second order approximation with respect to β of the exact field equations (Mezard et al., 1987)

$$\langle s_i \rangle = \left\langle \tanh\left(\beta \left[\sum_j J_{ij} s_j + h_i \right] \right) \right\rangle, \quad i = 1, \ldots, M. \tag{3.38}$$

The naïve mean field equations (3.32) arise from (3.38) as the first order approximation. While the naïve mean field method could be written in fairly general sense, the actual TAP equations are specific to the Ising model and not directly applicable to other models, although similar approximations can of course be developed.

Machine learning methods trying to avoid the biased marginal distributions of the variational approximation typically start from fitting the marginals and somehow

updating them for global coherence. This can be done using methods like *belief propagation* proposed by Pearl (1988). These methods update the approximation of the distribution of a node in a Bayesian network by receiving messages from neighbouring nodes, performing some computation combining the messages to form a new approximation for the node and sending new messages to the neighbours reflecting the new state of the node. The original belief propagation algorithm was meant only for tree-like networks with mainly discrete variables, but later developments include algorithms for a much wider class of graphs by loopy belief propagation (Pearl, 1988; Murphy et al., 1999) and more general continuous models by expectation propagation (Minka, 2001).

Although the message-passing algorithms presented above were derived in a very different way from the mean field algorithms of statistical physics, they are quite closely related. Yedidia et al. (2001) have shown that the belief propagation algorithm converges to a stationary point of Bethe free energy, a well known approximation of the true free energy. This has led to generalisations using better physical approximations. Similar relations have also been shown to exist between expectation propagation and adaptive TAP methods (Csató et al., 2002) while further connections between loopy belief propagation and Bethe free energy have been studied by Heskes (2004).

The methods presented in this section provide several alternatives to the variational approximation, each trying to avoid its weaknesses in a different way. The improvements very often come with a price of added complexity, which is probably the reason for limited practical use of most of the methods.

3.4 Alternative interpretations of the variational approximation

The variational or mean field approximation presented above in Sec. 3.3.2 is interesting because it can be derived from several seemingly unrelated starting points. These complementary views can help understand the method better and help in designing new ways to make learning even more efficient.

3.4.1 Bayesian analysis of approximations

The problem of approximating a distribution with a simpler one can be analysed in the general Bayesian framework by defining a loss function to evaluate the expected loss in utility for using only an approximation q instead of the distribution p corresponding to the actual beliefs. Natural requirements for such a loss function are smoothness and that $q = p$ should be its unique minimum. If the loss function is additionally assumed to be *local* in the sense that its value for a given event $\boldsymbol{\theta}_0 \in \Omega$ only depends on $\boldsymbol{\theta}_0$ and $q(\boldsymbol{\theta}_0)$, the only function satisfying the assumptions is (Bernardo and Smith, 2000)

$$D_{\mathrm{KL}}(p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H})||q(\boldsymbol{\theta})) = \int \log\left(\frac{p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H})}{q(\boldsymbol{\theta})}\right) p(\boldsymbol{\theta}|\boldsymbol{X},\mathcal{H})\, d\boldsymbol{\theta}. \tag{3.39}$$

Unfortunately this requires integration over the true posterior and is hence intractable.

This analysis yields a proper Bayesian justification for the originally *ad hoc* measure of Kullback–Leibler divergence (Kullback and Leibler, 1951). It does not, however, directly justify the switching of arguments of the divergence for the variational approximation. The switch is necessary for the sake of computational efficiency. It is however questionable whether the variational approximation has a loss function of its own, so it is probably not appropriate to call it a fully Bayesian method. The alternative interpretations can, however, provide partial justification for switching the arguments.

3.4.2 An information-theoretic view

Already in the 1960s, Wallace and Boulton (1968) proposed using Shannon's (1948) information theory and compact coding for approximate Bayesian inference. Their minimum-message-length (MML) inference provides many possible approximations, depending on the selected coding scheme. MML is related to the similar minimum-description-length (MDL) principle by Rissanen (1989). The information-theoretic minimum encoding approaches MML and MDL provide an appealing criterion for model selection by providing a very concrete implementation of Occam's Razor. The description lengths of the data encoding error and the model offer an attractive interpretation to the left side of Fig. 3.1. Model selection methods based on the criteria are widely used also in conjunction with other parameter learning methods.

From the viewpoint of this thesis, a particularly interesting coding was proposed by Wallace (1990). This scheme was later rediscovered and applied to neural networks by Hinton and van Camp (1993). The name *bits-back coding* is also due to them. This code relates the minimum encoding approaches directly with the variational approximation presented above.

The coding scheme employed in standard MML is to choose values of the parameters $\boldsymbol{\theta}$ and encode them up to the preselected precision ϵ_θ using

$$L(\boldsymbol{\theta}) = -\log\left(p(\boldsymbol{\theta}|\mathcal{H})\epsilon_\theta^{|\boldsymbol{\theta}|}\right) \tag{3.40}$$

bits, then model the data $\boldsymbol{X}$ using the chosen parameters and encode the modelling errors up to the precision ϵ_x using

$$L(\boldsymbol{X}|\boldsymbol{\theta}) = -\log\left(p(\boldsymbol{X}|\boldsymbol{\theta},\mathcal{H})\epsilon_x^{|\boldsymbol{X}|}\right) \tag{3.41}$$

bits. Here $|\boldsymbol{\theta}|$ and $|\boldsymbol{X}|$ denote the numbers of real-valued parameters and observations, respectively. If the effects of the precision of parameter specification ϵ_θ and data description ϵ_x are ignored, this is equivalent to MAP estimation. Taking the effects somehow into account results in more reliable methods capable of avoiding many of the pitfalls of point estimates.

Bits-back coding takes a radically different approach from the standard method of picking only one value of the parameters. Instead, a whole distribution $q(\boldsymbol{\theta})$ of

them is used. The actual coding is done by sampling a value from this distribution and using it. This leads to an expected code length of

$$\begin{aligned}\langle L(\boldsymbol{X})\rangle &= \langle L(\boldsymbol{\theta})\rangle + \langle L(\boldsymbol{X}|\boldsymbol{\theta})\rangle \\ &= \langle -\log p(\boldsymbol{\theta}|\mathcal{H})\rangle - |\boldsymbol{\theta}|\log\epsilon_\theta + \langle -\log p(\boldsymbol{X}|\boldsymbol{\theta},\mathcal{H})\rangle - |\boldsymbol{X}|\log\epsilon_x,\end{aligned} \tag{3.42}$$

where $\langle\cdot\rangle$ denotes expectation over the distribution $q(\boldsymbol{\theta})$. This is of course longer than the code length $L(\boldsymbol{X})$ attained by simply choosing a single optimal value for $\boldsymbol{\theta}$. This difference can, however, be compensated by using auxiliary information to select the value from $q(\boldsymbol{\theta})$. This information can later be recovered and should therefore not be included in the net code length. The amount of information that can be recovered to get *"bits back"* corresponds to the entropy of $q(\boldsymbol{\theta})$,

$$H_q(\boldsymbol{\theta}) = \langle -\log q(\boldsymbol{\theta})\rangle - |\boldsymbol{\theta}|\log\epsilon_\theta. \tag{3.43}$$

Subtracting this from the gross code length (3.42) yields

$$\begin{aligned}L_{\text{net}}(\boldsymbol{X}) &= \langle L(\boldsymbol{X})\rangle - H_q(\boldsymbol{\theta}) \\ &= \langle -\log p(\boldsymbol{\theta}|\mathcal{H})\rangle + \langle -\log p(\boldsymbol{X}|\boldsymbol{\theta},\mathcal{H})\rangle + \langle \log q(\boldsymbol{\theta})\rangle - |\boldsymbol{X}|\log\epsilon_x,\end{aligned} \tag{3.44}$$

where the coding precision of the parameters ϵ_θ cancels out. This allows using very high precision and thus using a very good discretisation of the continuous distribution of $\boldsymbol{\theta}$. As the coding precision of the data ϵ_x is independent of the model and its parameters, it can be ignored to get a final total code length of (Frey and Hinton, 1997)

$$\mathcal{C} = L_{\text{bits-back}}(\boldsymbol{X}) = \left\langle \log\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{X},\boldsymbol{\theta}|\mathcal{H})}\right\rangle, \tag{3.45}$$

which corresponds exactly to the variational free energy (3.34). The actual coding procedure needed to realise this is presented in Publication III. As the coding precision of the data ϵ_x was ignored, the resulting code length is not an absolute quantity but lacks this additive constant. As the cost function values for a given data set over different models are still comparable, this is usually not a problem in practice.

Although the information-theoretic approach is not based on as solid principles as the Bayesian approach, it can provide new views that are helpful in many applications. The code length interpretation gives an intuitive feeling to the meaning of the cost function and allows interpreting individual terms

$$\mathcal{C}(\theta_i) = L(\theta_i) = \left\langle \log\frac{q(\theta_i)}{p(\theta_i|\text{pa}(\theta_i),\mathcal{H})}\right\rangle \tag{3.46}$$

as the code lengths of individual parameters. As more important parameters are usually coded more precisely, this gives a relatively good indication of which parameters are likely to be redundant. These can then be pruned out if their contribution turns out to increase the total code length rather than to decrease it.

More examples of the benefits of the information-theoretic interpretation of learning algorithms in the analysis of their behaviour can be found in Sec. 4.3.1 and in Publication III.

3.4.3 An information-geometric view

Approximating the posterior with another distribution can also be viewed geometrically. This can be accomplished using information geometry, which is a theory of differential geometry of the manifold of probability distributions (Amari, 1985; Murray and Rice, 1993; Amari and Nagaoka, 2000). Probably the most important application of information geometry has been the development of the natural gradient, which is a correction to standard gradient based optimisation methods taking into the account the non-orthonormal coordinate system and thus accelerating learning (Amari et al., 1996; Amari, 1998). Information geometry has also recently been applied to the analysis of variational approximations (Tanaka, 1996, 2000, 2001; Amari et al., 2001; Ikeda et al., 2004).

Information geometry studies the space $\mathcal{F}$ of probability distributions $p(\boldsymbol{\theta}|\boldsymbol{X})$. If the distribution can be defined by a finite number of real valued parameters $\boldsymbol{\theta}$, the set of distributions $p(\boldsymbol{\theta}|\boldsymbol{X})$ has the structure of a manifold with coordinates $\boldsymbol{\theta}$. This manifold has a natural Riemannian metric

$$|d\boldsymbol{\theta}|^2 = \sum_{i,j} g_{ij}(\boldsymbol{\theta}) d\theta_i d\theta_j \tag{3.47}$$

with the metric tensor $g_{ij}(\boldsymbol{\theta})$ specified by Fisher information matrix

$$g_{ij}(\boldsymbol{\theta}) = \left\langle \frac{\partial \log p(\boldsymbol{\theta}|\boldsymbol{X})}{\partial \theta_i} \frac{\partial \log p(\boldsymbol{\theta}|\boldsymbol{X})}{\partial \theta_j} \right\rangle, \tag{3.48}$$

where the expectation is taken over $p(\boldsymbol{\theta}|\boldsymbol{X})$.

Variational approximation can be seen in the information-geometric framework as a method of finding an approximation for the true posterior $p \in \mathcal{F}$ in a submanifold $\mathcal{F}_0 \subset \mathcal{F}$ of tractable distributions. Geometrically, the optimal approximation is the projection of p on $\mathcal{F}_0$. Due to the complex nature of the geometry of $\mathcal{F}$, it is possible to define many such projections. The most important ones are the *e-projection* ($\alpha = 1$-projection of Amari) minimising the Kullback–Leibler divergence

$$\begin{aligned} q_e(\boldsymbol{\theta}) &= \arg\min_{q \in \mathcal{F}_0} D_{\mathrm{KL}}(q(\boldsymbol{\theta}) || p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})) \\ &= \arg\min_{q \in \mathcal{F}_0} \int \log\left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})}\right) q(\boldsymbol{\theta})\, d\boldsymbol{\theta} \end{aligned} \tag{3.49}$$

and the *m-projection* ($\alpha = -1$-projection of Amari) minimising the same divergence with the arguments reversed

$$\begin{aligned} q_m(\boldsymbol{\theta}) &= \arg\min_{q \in \mathcal{F}_0} D_{\mathrm{KL}}(p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H}) || q(\boldsymbol{\theta})) \\ &= \arg\min_{q \in \mathcal{F}_0} \int \log\left(\frac{p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})}{q(\boldsymbol{\theta})}\right) p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})\, d\boldsymbol{\theta}. \end{aligned} \tag{3.50}$$

In both cases, a corresponding e-connection (or m-connection) from p to q_e (or q_m) is orthogonal to the manifold $\mathcal{F}_0$ at q_e (or q_m) (Amari, 1985), in the sense of the Fisher Riemannian metric defined in Eq. (3.48). The geometry of the situation is illustrated in Fig. 3.3. While both e- and m-connections from p to the corresponding projections are geodesics and thus correspond to straight lines with respect to

the corresponding divergences, neither of them corresponds to the global minimum of the distance induced by the Riemannian metric (Amari, 1985). The connections between two probability distributions p_0 and p_1 can be written as

$$\log p_t = t \log p_1 + (1 - t) \log p_0 + c(t), \quad t \in [0, 1] \tag{3.51}$$

for the e-connection (exponential connection), where $c(t)$ is a normalising coefficient, and

$$p_t = tp_1 + (1 - t)p_0, \quad t \in [0, 1] \tag{3.52}$$

for the m-connection (mixture connection).

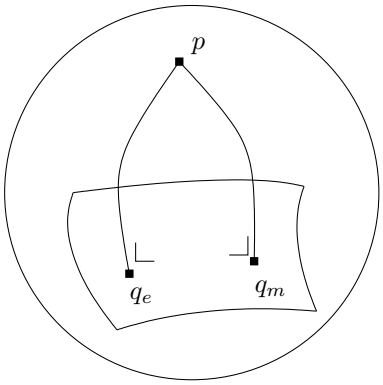

Figure 3.3: Geometry of the approximations q_e and q_m. The figure shows the true distribution p and the two projections to the submanifold $\mathcal{F}_0$.

Which projection should be used, then? Tanaka (2001) considers the m-projection more natural of the two from information-geometric perspective. Taking the Boltzmann machine considered above as an example, $\langle s_i \rangle_{q_m}$ is an unbiased estimate of s_i, whereas $\langle s_i \rangle_{q_e}$ is biased. Additionally, the m-projection is always unique whereas there may be several possible e-projections q_e. The major drawback with m-projection is that it involves integration over the intractable posterior $p(\boldsymbol{\theta}|\boldsymbol{X}, \mathcal{H})$ and is hence intractable.

3.4.4 Combining the views

As seen above, the same cost function can be viewed as a description length of the data as in Eq. (3.45), a physical variational free energy as in Eq. (3.31), or a lower bound on the model evidence as in Eq. (3.33). The minimiser of the cost function can also be viewed geometrically as a projection of the true posterior to a manifold of tractable distributions, although the properties of this projection differ slightly from the usual Euclidean projections.

All the different interpretations of the method have their own benefits. The variational Bayesian and the closely related mean field approach are theoretically best founded, although the approximation cannot be rigorously justified in a fully Bayesian manner. The mean field approach also shows the approximation as the first of a series of more accurate methods, based on the Taylor expansion of the

free energy. The information-theoretic view of the cost is easiest to interpret for most people and can therefore provide additional insights to the operation of the learning methods and the achieved results. The information-geometric approach provides a view to the space of the probability distributions. It could probably be used to analyse the behaviour of learning algorithms as they minimise the cost, similarly to the analysis of the EM algorithm by Amari (1995).

3.5 Algorithms for variational Bayesian learning

In the previous section, a wide range of justifications were presented for essentially the same method of minimising the cost (3.34). This defines a clear goal for the learning algorithms, but does not specify how to reach it. Neal and Hinton (1999) showed that the well-known EM algorithm can be interpreted as optimisation of a cost function of the form (3.34). This allows interpreting the variational methods as generalisations of the EM algorithm, leading to methods often referred to as variational EM. The actual algorithms can still vary.

Variational Bayesian (VB) methods can be roughly divided into two classes, those using a *free-form approximation* and those using a *fixed-form approximation* (Lappalainen and Miskin, 2000). The difference between the two is that in free-form approximation the model is restricted to allow free-form functional optimisation of the approximation while in fixed-form approximation the model is too complicated to allow general functional optimisation and the approximation is therefore restricted to be of a given fixed functional form such as Gaussian. The free-form approximation is more popular of the two as it is theoretically more attractive, but unfortunately it is not always applicable. Many of the applications of the methods combine both approaches for different parts of the model.

3.5.1 Free-form approximations and conjugate-exponential models

Most VB methods employing the free-form approximation are restricted to so-called *conjugate-exponential* models (Attias, 2000b; Ghahramani and Beal, 2001a,b; Bishop et al., 2003; Winn and Bishop, 2005). They are defined by Ghahramani and Beal (2001a) as models satisfying two conditions:

1. The complete data likelihood must be in the exponential family:

$$p(\mathbf{x}, \mathbf{s}|\boldsymbol{\theta}) = f(\mathbf{x}, \mathbf{s}) g(\boldsymbol{\theta}) \exp\{\boldsymbol{\phi}(\boldsymbol{\theta})^T \mathbf{u}(\mathbf{x}, \mathbf{s})\}. \tag{3.53}$$

2. The parameter prior must be conjugate to the complete data likelihood:

$$p(\boldsymbol{\theta}|\eta, \boldsymbol{\nu}) = h(\eta, \boldsymbol{\nu}) g(\boldsymbol{\theta})^\eta \exp\{\boldsymbol{\phi}(\boldsymbol{\theta})^T \boldsymbol{\nu}\}. \tag{3.54}$$

Here $\mathbf{u}, f, g, h$ are functions defining the exponential family, $\boldsymbol{\phi}(\boldsymbol{\theta})$ is the vector of natural parameters, and $\eta, \boldsymbol{\nu}$ are hyperparameters of the prior.

The class of conjugate-exponential models includes many interesting models such as all linear Gaussian models (Roweis and Ghahramani, 1999) as well as discrete models such as Boltzmann machines and discrete-variable belief networks. Mixture distributions such as mixtures-of-Gaussians do not belong to the conjugate-exponential class, but as their conjugates are still mixtures of similar distributions, extending the framework to cover these is relatively easy. The only potential problems arise from exponentially growing mixtures in models with mixtures connecting to other mixtures. Conjugate-exponential models with mixtures can be used to build a linear independent component analysis (ICA) model, but nonlinear continuous models are still impossible.

The conjugate-exponential framework guarantees that with suitable independence assumptions, the free-form approximate posterior will be of the same functional form as the conjugate prior. This allows very general learning algorithms as demonstrated by the variational message passing (VMP) framework by Winn and Bishop (2005). The VMP framework also forms the basis for the VIBES software package (Bishop et al., 2003).

3.5.2 Fixed-form approximations

Fixed-form approximations allow more flexible models than strict free-form approximations. Taking for example a Gaussian variable θ, the conjugate-exponential model is restricted to the form $\theta \sim N(\mu, \rho^{-1})$, where the mean parameter μ is again Gaussian and the precision parameter ρ has a Gamma distribution. This technique makes it very difficult to define a hierarchical model of variances, as one of the parameters of the Gamma distribution does not have a simple conjugate prior. Using a fixed-form approximation allows a simpler model $\theta \sim N(\mu, \exp(2v))$ with Gaussian v, leading to a straightforward hierarchy for the variances (Valpola et al., 2004).

Nonlinear models are another case where the fixed-form approximation is clearly the easier alternative. It would be possible to linearise the nonlinearity and thus approximate the likelihood with something that can be handled in the conjugate-exponential framework, but the result would not really be a free-form approximation.

Building blocks

The building blocks framework by Valpola et al. (2001) provides another complete solution for variational learning. It is not restricted to conjugate-exponential models but also allows more general models with fixed-form approximations. Only Gaussian and discrete variables are supported so far, but new types of variables can be added, including all the variables supported by the conjugate-exponential framework.

The building block framework has its restrictions, most importantly that there must be at most one path in the Bayesian network from the output of one node to the inputs of another. This restriction can always be avoided by adding more

hidden variables, but this has its own drawbacks. The building block framework is implemented in the Bayes blocks software package (Valpola et al., 2003a).

3.6 Optimisation algorithms

Many statistical learning approaches, including ML, MAP and variational Bayesian approaches are based on specifying some likelihood, probability density or other quantity as an objective function and optimising it. This transforms the learning problem into a nonlinear optimisation problem and allows using tools and techniques from that field. The specific structure of the problems does, however, often suggest easier methods.

3.6.1 Alternating optimisation and EM-like algorithms

Following Neal and Hinton (1999), the well-known EM algorithm can be interpreted as alternating optimisation of a given objective function over two sets of variables. The other set of variables is always kept fixed at their present values while the other is updated. The same principle can of course be generalised to more than two sets of variables, as is often done with more complex hierarchical models. An example of such algorithm is presented as Algorithm 3.1, which optimises a function $\mathcal{C} : \mathbb{R}^n \to \mathbb{R}$ iteratively by finding a minimum along each coordinate direction in turn while keeping the others fixed at their present values.

function optimise_alternating($\mathcal{C}, \mathbf{z}_1$):
    $\mathbf{z}_2 \leftarrow \mathbf{z}_1$
    for $i = 1, \ldots, n$:
        $\lambda \leftarrow \operatorname{argmin}_\lambda \mathcal{C}(\mathbf{z}_2 + \lambda \mathbf{e}_i)$
        $\mathbf{z}_2 \leftarrow \mathbf{z}_2 + \lambda \mathbf{e}_i$
    return $\mathbf{z}_2$

Algorithm 3.1: A single iteration of an alternating optimisation algorithm for function $\mathcal{C} : \mathbb{R}^n \to \mathbb{R}$. Vectors $\mathbf{e}_1, \ldots, \mathbf{e}_n$ denote the standard basis of $\mathbb{R}^n$.

The alternating optimisation procedure can be shown to converge to a local optimum under reasonable assumptions (Bezdek and Hathaway, 2003). As noted in Publication II, the convergence can, however, be very slow if the variable groups depend on each other. This is usually the case in probabilistic models when the level of the noise is low. This is illustrated in Fig. 3.4, which shows a contour plot of the posterior density of the parameters a and s in a scalar linear model $x = a \cdot s + n$. Viewing this as a generative model for x, it is clear that the value of a, for instance, can only be changed a little without affecting the reconstruction of x, if the value of s is not changed correspondingly. This problem is inherent to the alternating optimisation paradigm, and can only be resolved by updates affecting several variables simultaneously, thus proceeding in a diagonal direction instead of an axis-aligned one. Similar issues are also discussed by Raiko (2001).

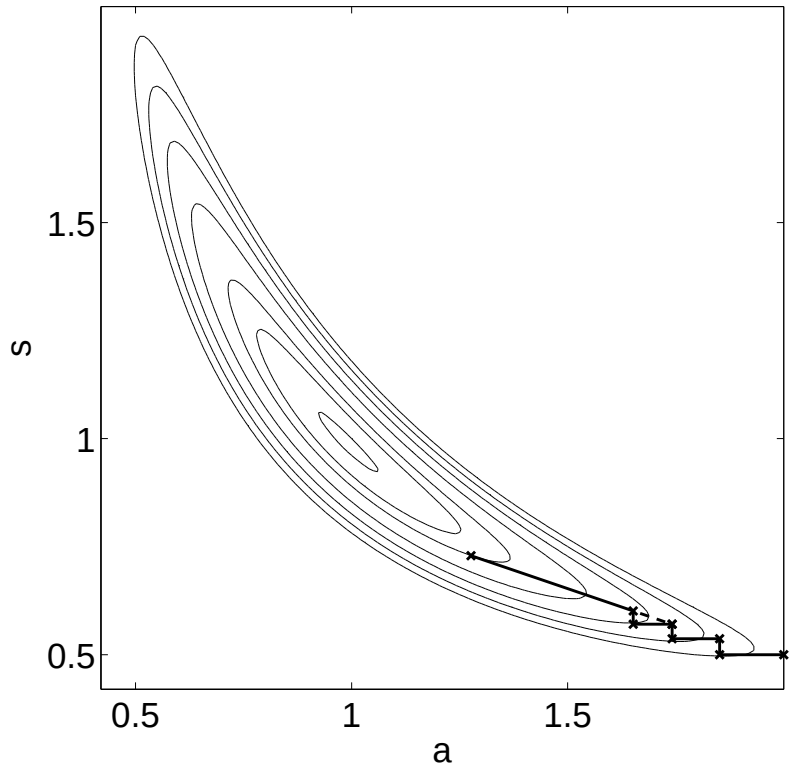

Figure 3.4: An illustration of the pattern search procedure with three rounds of cyclic updates followed by a line search. The contour plot shows the cost function (or in this case equivalently the posterior) in the problem of learning the parameters a and s of the model $x = a \cdot s + n$ for a single observation $x = 1$. The cost is thus of the form $\mathcal{C} \propto \log N(a \cdot s - 1, \sigma_n^2) N(a, \sigma_a^2) N(s, \sigma_s^2)$. The noise variance is 2 % of the prior variance of a and s. The plot includes only half of the posterior, as the distribution is symmetric about the origin.

3.6.2 Pattern search method

The pattern search methods (Hooke and Jeeves, 1961) are probably the simplest way to add diagonal updates to alternating optimisation algorithms. They do not require any derivatives and all the existing methodology can still be used, thus making them practically trivial to implement for different models. Modern optimisation literature tends to favour gradient-based methods more, but using them would require much more model-specific derivations. The diagonal updates are achieved by performing a line search (Bazaraa et al., 1993; Fletcher, 1987) in the direction defined by a complete set of alternating updates. A single iteration of such an algorithm consists of evaluating the combined direction $\Delta\mathbf{z}$ of a complete round of updates and performing a line search in this direction. This is illustrated graphically in Fig. 3.5 and also as Algorithm 3.2.

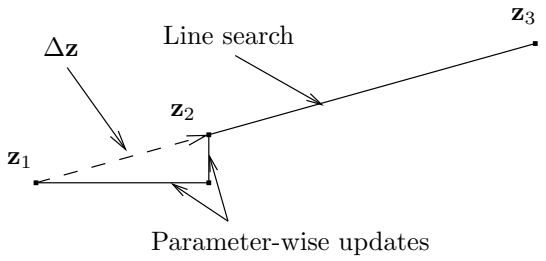

Figure 3.5: A schematic illustration of the pattern search algorithm. (From Publication II.)

function optimise_pattern($\mathcal{C}, \mathbf{z}_1$):
    $\mathbf{z}_2 \leftarrow$ optimise_alternating($\mathcal{C}, \mathbf{z}_1$)
    $\Delta\mathbf{z} \leftarrow \mathbf{z}_2 - \mathbf{z}_1$
    $\lambda \leftarrow \operatorname{argmin}_\lambda \mathcal{C}(\mathbf{z}_1 + \lambda \cdot \Delta\mathbf{z})$
    $\mathbf{z}_3 \leftarrow \mathbf{z}_1 + \lambda \cdot \Delta\mathbf{z}$
    return $\mathbf{z}_3$

Algorithm 3.2: A single iteration of the pattern search optimisation algorithm for function $\mathcal{C} : \mathbb{R}^n \to \mathbb{R}$.

A slightly modified version of the original algorithm by Hooke and Jeeves (1961) adapted to complement an alternating optimisation algorithm is proposed in Publication II. The main difference is to only use the pattern search step after several iterations of alternating updates, such as ten. This allows the iteration to stabilise to get the maximal benefit from the computationally expensive line search. The method is illustrated in Fig. 3.4. The method has been implemented in the Bayes blocks software package and has been successfully used with several models including hierarchical nonlinear factor analysis (Valpola et al., 2003c) and hierarchical variance models (Valpola et al., 2004).

Chapter 4

Linear independent component analysis

Many of the nonlinear methods presented in this thesis can be seen as generalisations of linear independent component analysis (ICA) algorithms, and a good understanding of the linear methods forms a basis for studying the nonlinear ones. This chapter gives a brief overview of linear ICA, starting with theoretical questions of separability in Sec. 4.1 and continuing with a brief review of the most prominent ICA algorithms in Sec. 4.2. Some of the potential difficulties of ICA are highlighted in Sec. 4.3, as these again have clear counterparts in the nonlinear problems. The discussion of overfitting phenomena in Sec. 4.3.1 is an excellent example of the benefits of the information-theoretic interpretation of variational Bayesian learning, as also discussed in Publication III.

4.1 Separability of linear mixtures

In the factor analysis model

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n}, \tag{4.1}$$

the factors $\mathbf{s}$ and noise $\mathbf{n}$ are assumed to be independent and both have a Gaussian distribution. These assumptions simplify computation significantly, but the Gaussianity causes problems with the identifiability of the model. As the Gaussian distribution of the factors is invariant with respect to orthogonal rotation, the factors can only be determined up to such a rotation. During the past century, a number of more or less arbitrary methods have been presented to resolve the indeterminacy by some *ad hoc* criterion (Harman, 1960; Hyvärinen et al., 2001).

The rotation indeterminacy is characteristic to the Gaussian distribution. According to the Darmois–Skitovich theorem (Darmois, 1953), two linear combinations

$$s'_1 = \sum_j b_j s_j \quad \text{and} \quad s'_2 = \sum_j c_j s_j$$

of independent random variables s_i can only be independent if all the variables s_j with $b_j c_j \neq 0$ are Gaussian.

The Darmois–Skitovich theorem implies that at least in the noiseless case when

$$\mathbf{x} = \mathbf{A}\mathbf{s}, \tag{4.2}$$

independent non-Gaussian sources $\mathbf{s}$ can be recovered from their non-singular mixtures $\mathbf{x}$ up to permutation and scaling (Rao, 1969; Comon, 1994). An alternative proof for the theorem based on the diagonality of the Hessian of the log-density of independent random vectors is given by Theis (2004). A more thorough review of the separability conditions especially for non-square mixings is presented by Eriksson and Koivunen (2004).

The separability result for noiseless case can be partially extended to the case with Gaussian noise $\mathbf{n}$ and non-Gaussian sources $\mathbf{s}$. Inverting the model to recover the exact values of the sources is of course not possible, but the mixing matrix $\mathbf{A}$ and the distributions of $\mathbf{s}$ and $\mathbf{n}$ can be determined up to permutation and scaling, provided that the matrix $\mathbf{A}$ has at least two nonzero elements in each column and satisfies an additional complicated rank condition (Kagan et al., 1973, Th. 10.4.3). The inclusion of noise also introduces a potential new translation indeterminacy between the means of $\mathbf{n}$ and $\mathbf{s}$.

4.2 Independent component analysis (ICA) and blind source separation (BSS)

The first source separation algorithm based on the non-Gaussian noiseless model (4.2) with independent sources was presented by Hérault and Jutten in the 1980s, as reviewed by Jutten and Taleb (2000). By 1990s, the method had been established as independent component analysis (ICA) (Jutten and Herault, 1991; Comon, 1994; Hyvärinen et al., 2001). Even though the assumption of a noise-free mixture is often unrealistic, the numerous proposed ICA algorithms work well in several real world situations, including for instance some quite noisy biomedical applications (Makeig et al., 1996; Vigário et al., 2000; Hyvärinen et al., 2001).

ICA is closely related to the more general problem of *blind source separation* (BSS), where the goal is to uncover the underlying hidden sources from their linear mixture of type (4.2). If the sources are non-Gaussian and independent, the BSS problem can be solved by an ICA algorithm, but in other situations other methods may be needed.

4.2.1 Classical algorithms

There are many very different algorithms for solving the basic linear ICA problem. As the methods used to solve the nonlinear problems are generalisations of linear methods, some of the most important ones will be briefly reviewed from this perspective. More thorough reviews are presented for instance by Haykin (2000) and Hyvärinen et al. (2001) as well as Cichocki and Amari (2002).

The first ICA algorithm proposed by Jutten and Herault (1991) was based on nonlinear decorrelation, that is minimising nonlinear correlations $E[f(\hat{s}_i)g(\hat{s}_j)]$ for suitable nonlinear functions f and g. Here $\hat{s}_i$ denotes an estimate of the source s_i. One more efficient variant of the same general principle is the minimisation of fourth order cross cumulants. Performing this by joint approximate diagonalisation of eigenmatrices leads to the popular JADE algorithm (Cardoso and Souloumiac, 1993; Cardoso, 1999).

Another popular principle for performing ICA is to make the sources as non-Gaussian as possible. This is often done by maximising the absolute value of the kurtosis of the sources or by maximising the negative differential entropy of the sources. In both of these cases Gaussian variables will attain the other extremum value. The most popular ICA algorithm based on these principles is the FastICA (Hyvärinen and Oja, 1997; Hyvärinen, 1999).

Other approaches to ICA include information-theoretic minimisation of mutual information between the estimated sources, and maximum likelihood estimation. These approaches are closely related and often lead to the same algorithms. The most popular algorithms are the Bell–Sejnowski algorithm that can be seen as either maximum likelihood estimation method or as implementation of the infomax principle of maximising the output entropy of a nonlinear separating neural network (Bell and Sejnowski, 1995). The algorithm's convergence can be improved significantly by using the information-geometric natural gradient (Amari et al., 1996; Douglas and Amari, 2000).

4.2.2 Bayesian ICA

The classical ICA models are based on the idealised noiseless mixing model (4.2) with a full-rank mixing matrix. As the data are under this assumption in an M-dimensional subspace, linear PCA can be used to decrease the dimensionality so that the residual mixing matrix is an orthogonal square matrix. If, however, the mixing is noisy, this procedure can yield biased results. A general noisy generative model based on (4.1) with non-Gaussian sources was first proposed by Attias (1999). His *independent factor analysis* (IFA) was based on mixture-of-Gaussians (MoG) source model and the EM algorithm. As the number of the sources increases, the exact E-step of the EM algorithm requires summations over an exponentially growing number of source state configurations. This is countered by using a variational approximation for the sources. Later developments of the IFA model include a simple dynamic source model with a hidden Markov model (HMM) modelling the probabilities of the MoG components (Attias, 2000a, 2001).

Attias's EM approach uses point estimates for the mixing matrix and cannot be used for model comparison. This shortcoming was corrected by fully variational Bayesian approaches by Lappalainen (1999) with a lognormal variance prior and fixed-form approximation, and later by several authors with an inverse-gamma source prior and free-form approximation (Miskin and MacKay, 2001; Choudrey and Roberts, 2001). Other variational approaches to Bayesian ICA using slightly different methods include a specific variational transformation to derive a lower bound for the marginal likelihood by approximating the sources by Gaussians by

Girolami (2001) and a mean field approach including linear response and TAP corrections to account for the posterior correlations of the sources by Højen-Sørensen et al. (2002). An approach to ICA using Bayesian Ying-Yang harmony learning is presented by Xu (2003). The variational method by Chan et al. (2003) includes handling of missing data. Several researchers have also proposed using variational Bayesian learning for mixtures of ICA models (Chan et al., 2002; Choudrey and Roberts, 2003).

Although variational Bayesian ICA has attracted many researchers and produced convincing results, it has its own drawbacks. Classical noiseless ICA algorithms usually proceed by first decorrelating (whitening) the data by principal component analysis (PCA) and only then using ICA to find the residual orthogonal rotation. In the case of noisy mixtures especially with non-isotropic noise, whitening is not acceptable. The mixing matrix $\mathbf{A}$ to be estimated is therefore non-orthogonal and this induces posterior dependencies between the sources $\mathbf{s}$. Using a factorial posterior approximation ignores these dependencies and may lead to inferior results (Højen-Sørensen et al., 2002; Ilin and Valpola, 2003). This issue is discussed in more detail in Sec. 4.3.2.

4.2.3 Algorithms using temporal information

Classical ICA algorithms do not assume any special relation between consecutive data samples. They will produce the same results even if the order of the observations is permuted arbitrarily. This allows the methods to be applied to many types of data, but the possible additional temporal information is lost.

If the true sources are independent time series with temporal dependencies, separating them is easier using temporal information than without it. In this case, the separation can be performed using second-order statistics alone by simultaneously diagonalising several time lagged covariance matrices. If the autocorrelations of different sources are different, introduction of several matrices resolves the rotation indeterminacy and independent components can be recovered. This approach is used in SOBI (Belouchrani et al., 1997) and TDSEP (Ziehe and Müller, 1998) algorithms. Another blind source separation method using temporal information is slow feature analysis by Wiskott and Sejnowski (2002). It tries to recover sources with minimal temporal change, that is slow features. It has been shown that under certain circumstances, slow feature analysis, TDSEP and SOBI algorithms are equivalent (Blaschke and Wiskott, 2004).

The TDSEP approach can also be combined with traditional ICA algorithms by jointly diagonalising the time lagged covariances and higher-order cumulants of JADE to get the general JADE_{TD} algorithm (Müller et al., 1999). Another approach achieving the same is to look for sources that have minimum algorithmic complexity as is done in the complexity pursuit algorithm (Hyvärinen, 2001b).

A third possible criterion for performing ICA is to utilise the non-stationarity of the source signals. This is typically achieved by inspecting the variances of the sources. If these vary slowly, it is possible to separate sources with Gaussian marginals and equal autocorrelations, which is not possible with any other method.

Algorithms utilising the non-stationarity have been presented by several authors (Matsuoka et al., 1995; Pham and Cardoso, 2001; Hyvärinen, 2001a). Bayesian methods capable of utilising the non-stationarity include a method using a hidden Markov model to model the Gaussian mixture coefficients in IFA by Attias (2000a) and variance models using the building blocks framework by Valpola et al. (2004).

4.3 Difficulties

With a number of efficient algorithms available, linear ICA is essentially a solved problem. Nevertheless, there are a few difficulties that can be encountered in careless application of the standard methods. The reasons of the difficulties are deeply rooted in the design principles of the methods and the same problems usually carry over to nonlinear methods based on the same principles.

4.3.1 Overfitting: Spikes and bumps

Application of standard linear ICA algorithms to a high-dimensional data set with a relatively small number of samples often leads to the recovery of *spikes*, sources that differ significantly from zero at very few time instances. Spikes are a natural overfitting result for traditional algorithms as they are in a sense maximally non-Gaussian and maximise a number of contrast functions. Finding spiky directions in a high-dimensional space is also almost always very easy (Hyvärinen et al., 1999; Särelä and Vigário, 2003).

Variational Bayesian ICA algorithms are not susceptible to spikes. Following the information-theoretic interpretation, this is very natural as a spike corresponds to using a full source to model only a single observation sample, a clearly inefficient coding practice. An example illustrating this can be found in Publication III.

Perfect spikes can only be found in data with no dependencies between consecutive samples. If the consecutive samples depend on each other, the spikes tend to spread out to wider *bumps*. From the point of view of the traditional algorithms, this does not change things very much, as bumpy signals are still extremely non-Gaussian. For Bayesian algorithms, however, the situation is entirely different. As a single bumpy source can be used to model several consecutive data samples, it can yield a compact code for the data. In a sense, this can be viewed as "misuse" of the model: the sources are used to segment the data while the corresponding mixing vectors model all the observations within the segments. The proper cure for bumps is to somehow include the dependencies between consecutive samples into the model, as discussed in Publication III.

4.3.2 Posterior correlations in variational Bayesian methods

As the goal of ICA is to extract sources that are independent of each other, one might assume the factorial posterior approximation neglecting the dependencies as those typically used in naïve mean field and variational methods would work

well. Unfortunately this is not the case, and neglecting the posterior dependencies may actually prevent separation of the sources (Højen-Sørensen et al., 2002; Ilin and Valpola, 2003).

In the ICA model, the sources are assumed to be independent *a priori*. This does not imply that they would also be independent *a posteriori*, that is when conditioned on the data. This can easily be seen for instance from the *explaining away* phenomenon, where one source explaining for example a spike in the observations removes the need for the others to explain it. The exact nature of the phenomenon varies between different models, but it should nevertheless demonstrate the existence of posterior dependencies between variables that are independent a priori.

In linear ICA-like models, the posterior dependencies of the sources arise mainly from non-orthogonal column vectors of the mixing matrix. They induce correlations of the form $\mathbf{\Sigma}_s \propto \left(\mathbf{A}^T \mathbf{\Sigma}_n^{-1} \mathbf{A}\right)^{-1}$. These correlations are minimised if the mixing matrix is orthogonal, which may lead to the method favouring the PCA solution instead of the correct ICA solution (Ilin and Valpola, 2003).

The experiments in Publication I suggest that the same problem also affects the nonlinear model. While modelling posterior covariances would be possible, it would be computationally very demanding and therefore an alternative solution of using a Gaussian source model and linear ICA post-processing is often used instead, as will be discussed in Sec. 6.5.

Chapter 5

Nonlinear blind source separation (BSS) and factor analysis

This chapter is the nonlinear counterpart of the previous one. It starts with theoretical separability considerations in Sec. 5.1. An important special case of post-nonlinear mixtures is presented in Sec. 5.2. A variational Bayesian approach to post-nonlinear ICA can be found in Publication IV. Algorithms for general nonlinear mixtures are briefly reviewed in Sec. 5.3 while the variational Bayesian approach is presented in detail in Chapter 6.

5.1 On the difficulty of nonlinear BSS

While sources can be separated rather easily from a linear mixture (2.2), the corresponding problem with a nonlinear mixture

$$\mathbf{x} = \mathbf{f}(\mathbf{s}) + \mathbf{n}, \tag{5.1}$$

where $\mathbf{f} : \mathbb{R}^M \to \mathbb{R}^N$ is a nonlinear function, is significantly more difficult. Any potential solution is clearly non-unique due to possible undetermined scalar nonlinearities in the sources. This follows from the fact that if random variables s_i and s_j are independent, so are $g_i(s_i)$ and $g_j(s_j)$ for any invertible $g_i, g_j : \mathbb{R} \to \mathbb{R}$ (Jutten et al., 2004). Unfortunately this is only the first of a list of indeterminacies.

5.1.1 Separability

In a sense, separating independent components with a nonlinear mapping is very simple, even too simple. In fact, any N-dimensional random vector $\mathbf{x}$ can be quite easily transformed nonlinearly to another N-dimensional random vector $\mathbf{y} =$

$\mathbf{g}(\mathbf{x})$ whose components are independent (Hyvärinen and Pajunen, 1999; Jutten et al., 2004). This can be accomplished by a simple construction similar to the Gram–Schmidt orthogonalisation procedure. The construction was first proposed by Darmois in the early 1950s.

In the construction, $\mathbf{y}$ can be assumed to have uniform density in the unit hypercube $[0,1]^N$. This yields the condition

$$p(\mathbf{x}) = p_{\mathbf{y}}(\mathbf{g}(\mathbf{x}))\,|\det D_{\mathbf{g}}(\mathbf{x})| = |\det D_{\mathbf{g}}(\mathbf{x})|, \tag{5.2}$$

where $D_{\mathbf{g}}$ is the Jacobian matrix of the function $\mathbf{g}$. Looking for a solution of the form

$$g_i(\mathbf{x}) = g_i(x_1, x_2, \ldots, x_i), \quad i = 1, 2, \ldots, N, \tag{5.3}$$

the determinant of the Jacobian reduces to a product of terms $\partial g_i(\mathbf{x})/\partial x_i$. On the other hand, $p(\mathbf{x})$ can be decomposed as

$$\begin{aligned} p(\mathbf{x}) &= p(x_1)p(x_2|x_1)p(x_3|x_1,x_2)\cdots p(x_N|x_1,x_2,\ldots,x_{N-1}) \\ &= |\det D_{\mathbf{g}}(\mathbf{x})| = \prod_{i=1}^{N} \frac{\partial g_i(x_1,x_2,\ldots,x_i)}{\partial x_i}. \end{aligned} \tag{5.4}$$

This is clearly satisfied if

$$\frac{\partial g_i(x_1,x_2,\ldots,x_i)}{\partial x_i} = p(x_i|x_1,x_2,\ldots,x_{i-1}), \quad i = 1,2,\ldots,N. \tag{5.5}$$

Integrating this yields a solution for g_i as the conditional cumulative density function of x_i given $x_1, x_2, \ldots, x_{i-1}$, for all $i = 1, 2, \ldots, N$.

As can be seen, the above construction contains many arbitrary choices, such as the use of uniform density and the assumed form of $\mathbf{g}$. It is therefore not very surprising that the separation result is not at all unique, as shall be shown next.

5.1.2 Uniqueness

Recalling the definition of independence of the components of random vector $\mathbf{x}$ from Eq. (3.18), it is clearly preserved by mappings performing a permutation of the components and possibly some scalar transformations as in

$$\mathbf{g}(\mathbf{x}) = [g_1(x_{\sigma(1)}), \ldots, g_n(x_{\sigma(n)})], \tag{5.6}$$

where $\sigma \in S_n$ is a permutation and $g_1, \ldots, g_n : \mathbb{R} \to \mathbb{R}$ are invertible scalar functions.

It can be shown (Taleb, 2002) that mappings of the form (5.6) with invertible $g_1, \ldots, g_n$ are in fact the only invertible mappings that map all random vectors with independent components to random vectors with independent components. This does not mean that there would be no other such mappings for specific random vectors. This can be seen from the following construction for two uniformly distributed random variables (Hyvärinen and Pajunen, 1999).

Let x_1 and x_2 be independent random variables that are uniformly distributed on the interval $[0, 1]$, thus jointly uniformly distributed in the unit square $[0, 1] \times [0, 1]$. Any transformation $\mathbf{g}$ of the variables that preserves the volume does not alter the distribution of the variables and hence their independence. This happens if

$$|\det D_{\mathbf{g}}(\mathbf{x})| = 1 \tag{5.7}$$

for all $\mathbf{x}$.

Volume preserving transformations of two variables are easy to represent by replacing the Cartesian coordinates x_1 and x_2 with polar coordinates r and θ specified by

$$x_1 = r \cos\theta, \quad x_2 = r \sin\theta. \tag{5.8}$$

A set of volume preserving transformations can now be defined by

$$r' = r, \quad \theta' \equiv \theta + f(r) \cdot \theta_0 \pmod{2\pi}, \tag{5.9}$$

where $f(r)$ is a suitable scalar function and $\theta_0 \neq 0$ is a constant. Choosing, for instance, a smooth $f(r)$ with $f(r) = 0$ for $r > \frac{2}{3}$ and $f(r) = 1$ for $r < \frac{1}{3}$ provides a smooth transformation from x_1 and x_2 to another pair of independent random variables x_1' and x_2' that is not of the form (5.6). Condition (5.7) can be easily verified to apply for this transformation.

This construction can be combined with the diagonalisation procedure presented in Sec. 5.1.1 to generate a class of nontrivial nonlinear mappings that are unrelated to each other, and each map the given random vector to one with independent components. This shows the non-uniqueness of nonlinear ICA: any random vector can be nonlinearly decomposed into independent components in several non-trivially related ways. In order to achieve blind nonlinear separation of sources, additional constraints are thus needed. The above constructions show that even constraints such as smoothness of the mixing or demixing mapping or knowing the actual source distributions are not enough to guarantee separation. In different approaches to nonlinear ICA and BSS, the actual constraints are typically implicitly defined by the model and methodology used.

5.1.3 Note on terminology

Despite the inherent impossibility of performing nonlinear ICA in a sense strictly generalising linear ICA, the term *nonlinear ICA* is nevertheless used in many places, including Publications I and VI. In those contexts, the term should in strict sense be interpreted as nonlinear BSS with assumption of non-Gaussianity and independence of the sources and additional regularising assumptions as provided by the variational Bayesian framework and the used model of the nonlinearity.

5.2 Post-nonlinear ICA

The intractability of general nonlinear ICA has opened research on more restricted nonlinear generalisations of linear ICA that would still be tractable. The most

popular of such models is post-nonlinear (PNL) ICA, where the mixing is restricted to be of the form

$$x_i = f_i\left(\sum_{j=1}^{M} a_{ij}s_j\right) + n_i, \quad i = 1, \ldots, N \tag{5.10}$$

where the scalar functions $f_i : \mathbb{R} \to \mathbb{R}$ are called the post-nonlinear distortions or post-nonlinearities and $\mathbf{A} = (a_{ij})$ is the mixing matrix. If the post-nonlinearities are assumed to be invertible and the noise n_i equal to zero, the model would seem to be separable with mostly the same restrictions as in the linear case, although no complete proof for the general result has been presented (Taleb and Jutten, 1999; Theis, 2004; Theis and Gruber, 2005).

Most PNL ICA algorithms work by learning an inverse of the mixing model such that

$$\hat{s}_j = \sum_{i=1}^{N} w_{ji}g_i(x_i), \tag{5.11}$$

where $g_i : \mathbb{R} \to \mathbb{R}$, $i = 1, \ldots, N$ are the inverses of f_i and $\mathbf{W} = (w_{ji})$ is the separating matrix. The model is learned by minimising the mutual information of the output vector $\hat{\mathbf{s}}$. The nonlinearities g_i can be modelled, for instance, by multilayer perceptron (MLP) networks (Taleb and Jutten, 1999). A review of different PNL ICA algorithms in presented by Jutten and Karhunen (2004).

The traditional PNL ICA is based on the assumption that both the linear mixing and all the individual post-nonlinearities are invertible. While this makes the model easier to handle and invert, there is no apparent reason why all separable PNL mixtures would have to be of this form. It is plausible that a linear mapping from a low dimensional space to a higher dimensional one followed by post-nonlinearities some of which may be non-invertible can produce separable mixtures, provided that the global mapping from sources to observations is injective. No formal proofs of necessary or sufficient conditions on separability of such mixtures exist and development of such proofs provides an important direction of future research.

Separability of certain such PNL mixtures with a general nonlinear BSS method was demonstrated empirically by Ilin et al. (2004a). Using the general nonlinear method to solve a simpler PNL problem of course ignores the additional information on the form of the problem. This has been addressed in Publication IV, which presents a variational Bayesian algorithm for PNL mixtures. The method uses a generative model of the type (5.10) with MLP networks to model the post-nonlinearities f_i. The approach also allows non-invertible post-nonlinearities and includes noise. A more detailed presentation of it can be found in Sec. 6.6.1 and Publication IV.

5.3 General nonlinear models and algorithms

Despite the ill-posed nature of the nonlinear ICA/FA problem, there are several nonlinear FA and BSS methods. This section is not intended as a thorough review

of these methods. Wall and Amemiya (2004) present a more complete review on traditional parametric statistical nonlinear FA models while the neural and BSS models are reviewed by Jutten and Karhunen (2004).

In general, many of the models proposed in literature are only suited for very low-dimensional data and the methods are demonstrated with mildly nonlinear two-dimensional mixtures. The two-dimensional case is significantly simpler than higher dimensional ones and such methods are mostly not considered here.

5.3.1 Nonlinear factor analysis

The need for nonlinear generalisations of the basic FA model was noted by several researchers in statistics already in the 1950s and 1960s. The first models were typically nonlinear in factors but linear in parameters, that is of the form

$$\mathbf{x} = \mathbf{f}(\mathbf{s}, \boldsymbol{\theta}_\mathbf{f}) = \boldsymbol{\theta}_\mathbf{f}\mathbf{g}(\mathbf{s}), \tag{5.12}$$

where $\mathbf{g}$ is a pre-specified nonlinear function, often a low-order polynomial. While the model is not directly linearisable in terms of $\mathbf{s}$ as the methods in Sec. 5.2, it is certainly not a general nonlinear model.

The first fully nonlinear FA models were presented by Yalcin and Amemiya in the 1990s. Their method is based on so-called errors-in-variables parameterisation (Yalcin and Amemiya, 2001)

$$\mathbf{x} = \begin{pmatrix} \mathbf{f}(\mathbf{s}, \boldsymbol{\theta}_\mathbf{f}) \\ \mathbf{s} \end{pmatrix} + \mathbf{n}, \tag{5.13}$$

where the factors are taken from selected channels of the observations, minus noise. With a suitably complicated noise model, this is of course equivalent to the standard model (5.1). The components of the nonlinearity $\mathbf{f}$ are typically polynomials with additional terms such as $\exp(\sum_i a_i s_i)$ or $(1 + \exp(\sum_i a_i s_i))^{-1}$.

5.3.2 Machine learning approaches

The first neural network model for nonlinear FA was proposed by Werbos (1992) at the same time as the first general statistical models appeared. His model included two MLP networks, one for mapping $\mathbf{s} \mapsto \mathbf{x}$ and one for $\mathbf{x} \mapsto \mathbf{s}$. An optional dynamic extension included another MLP for prediction of $\mathbf{s}$.

Another classical neural model for such a purpose are auto-associative MLP networks, that are MLP networks trained with input-output pairs $(\mathbf{x}, \mathbf{x})$. The number of neurons in a hidden layer is restricted to be smaller than the number of inputs and outputs, thus creating a bottleneck. The extracted nonlinear features can be retrieved from the values of the hidden neurons. With standard back-propagation this approach is very prone to overfitting and local minima, but more advanced learning methods such as flat minimum search can provide a method for nonlinear BSS (Hochreiter and Schmidhuber, 1999a,b) through sparseness of the extracted features.

MLP networks are also used as a basis of the variational Bayesian nonlinear BSS method presented in this thesis. In case of the variational Bayesian method, the MLP is used to model only the generative mapping $\mathbf{f}$ from $\mathbf{s}$ to $\mathbf{x}$. The method is presented in more detail in Chapter 6 as well as in Publications I, VI and VII.

The MISEP method by Almeida (2003, 2004) is a generalisation of the infomax method of linear ICA for nonlinear mixtures using an MLP network to model the nonlinearity. The source separation is supposedly based on the smoothness constraint provided by the MLP. While even mathematical $\mathcal{C}^\infty$-smoothness of the mapping is not sufficient for ensuring nonlinear separation in theory, the method does provide good separation results in several artificial examples as well as in a real nonlinear image mixture problem (Almeida and Faria, 2004). These results are probably due to the fact that even though an MLP network with enough hidden neurons is a universal approximator (Hornik et al., 1989; Funahashi, 1989), networks with a limited number of hidden neurons produce more restricted mappings. In fact, a network with invertible square weight matrices is a sufficiently specialised structure to allow limited theoretical analysis (Theis et al., 2002).

Kernel methods have recently become a popular method of producing nonlinear counterparts for many linear statistical methods (Schölkopf and Smola, 2002). They work for any method based on second-order statistics that can be evaluated through inner products of the observation vectors. The kernel methods are based on transforming the data nonlinearly with a mapping $\Theta : \mathbb{R}^N \to \mathcal{F}$ to a high-dimensional or even infinite-dimensional feature space $\mathcal{F}$ and performing the linear algorithm on the transformed data. This involves evaluating inner products of the transformed data vectors, but this can be done efficiently using the kernel trick of writing the inner product with the help of a kernel function k as

$$k(\mathbf{x}, \mathbf{y}) = \Theta(\mathbf{x}) \cdot \Theta(\mathbf{y}). \tag{5.14}$$

This makes it easy to define, for instance, a kernelised version of the linear PCA algorithm (Schölkopf et al., 1998). The kernel PCA algorithm is used in Publication V to aid the initialisation of the variational Bayesian nonlinear BSS method.

ICA is inherently based on higher-order statistics and is therefore not directly kernelisable. Separation of temporally correlated sources is, however, possible using only second-order statistics. Harmeling et al. (2003) propose a kernel method for nonlinear BSS of temporally correlated signals. The method is basically a kernelisation of the well-known TDSEP algorithm (Ziehe and Müller, 1998). The problem with the method is the selection of the essential components from the multitude generated by the algorithm. The kernel based nonlinear BSS method should not be mixed with KernelICA, which is a method for separation of linear mixtures using contrast functions based on kernel methods (Bach and Jordan, 2002).

The kernel BSS method is also closely related to the nonlinear version of independent slow feature analysis (Blaschke and Wiskott, 2004). The nonlinear slow feature analysis method works by mapping the data nonlinearly to a high-dimensional feature space and looking for the slow components there. In its basic form, the method requires explicit expansion in the feature space and will thus probably not scale to large problems. Being mostly equivalent to the kernel TDSEP, it also suffers from the same problem of identifying the meaningful components.

Chapter 6

Nonlinear BSS by variational Bayesian learning

The variational Bayesian nonlinear FA and BSS method, which is the central topic of this thesis, was first introduced in Publication I. Slightly different formulations of the method and new experimental results have since been presented in several publications (Valpola, 2000; Valpola et al., 2000, 2003b).

This chapter begins with discussion on why the Bayesian approach can solve the difficult nonlinear FA and BSS problems in Sec. 6.1. This is followed by a brief overview of the basic method including the model structure and learning algorithm in Sections 6.2 and 6.3, respectively. These include discussion of the initialisation of the method, following Publication V. A good approximation of the nonlinearity is central to the method. Potential approximations are presented in Sec. 6.4, following Publications VI and VII. Discussion on the choice of the source model and its implications are presented in Sec. 6.5. This is followed by discussion on variants of the method for post-nonlinear mixtures and dynamical models in Sec. 6.6, partly following Publication IV. Finally, some applications of the method are considered in Sec. 6.7. Matlab implementations of the basic nonlinear FA model and the nonlinear state-space model are available as free software (Valpola et al., 2002a).

6.1 On Bayesian nonlinear source separation

The non-uniqueness result of nonlinear FA and ICA presented in Sec. 5.1 means that there can be no correct nonlinear ICA solution and it is meaningless to talk of such a thing. There are always several solutions, some of which are better than others. This multitude of solutions fits quite naturally to the Bayesian formulation, where the goal is not to find a single solution, but an ensemble of possible solutions. The averaging process used in evaluating predictions will smooth down overly complex solutions in a similar manner as in the model selection example discussed in Sec. 3.1.7.

The variational approach used in this work has also other properties that further help regularise the problem. The flexible nonlinear model has many internal symmetries that are reflected in the true posterior. For purposes of the source separation problem, these symmetries are, however, not interesting. The variational approximation breaks the symmetry by ignoring the dependencies between different groups of parameters and the sources. The unimodal posterior approximation will find a broad region of potential solutions, thus returning essentially a single most plausible solution.

6.2 The model

Let us assume the observed data $\boldsymbol{X}$ follows the nonlinear mixing model of Eq. (5.1). The nonlinearity $\mathbf{f}$ is parameterised with a multilayer perceptron (MLP) network with single hidden layer with H hidden neurons (Haykin, 1999). This allows writing a generative model for a data vector $\mathbf{x}(t)$ as

$$\mathbf{x}(t) = \mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}) + \mathbf{n}(t) = \mathbf{B}\boldsymbol{\phi}(\mathbf{A}\mathbf{s}(t) + \mathbf{a}) + \mathbf{b} + \mathbf{n}(t), \tag{6.1}$$

where $\boldsymbol{\theta}_\mathbf{f} = (\mathbf{A}, \mathbf{B}, \mathbf{a}, \mathbf{b})$, $\mathbf{A} \in \mathbb{R}^{H\times M}$ and $\mathbf{B} \in \mathbb{R}^{N\times H}$ are the weight matrices and $\mathbf{a} \in \mathbb{R}^H$ and $\mathbf{b} \in \mathbb{R}^N$ are the bias vectors of the first and second layer of the MLP network, respectively. The weight matrices and bias vectors will hence be collectively called the weights of the MLP. The activation function ϕ is the standard hyperbolic tangent. It is applied component-wise to its argument vector.

The noise $\mathbf{n}(t)$ and all the weight matrices and bias vectors of the MLP are *a priori* assumed to be Gaussian and independent of each other. The noise is assumed to have a general diagonal covariance and a hierarchical lognormal variance prior of the form

$$p(\mathbf{n}(t)|\mathbf{v}_n) = N(\mathbf{n}(t);\ \mathbf{0}, \mathrm{diag}(\exp(2\mathbf{v}_n))) \tag{6.2}$$

$$p(v_{n_i}|m_{v_n}, v_{v_n}) = N(v_{n_i};\ m_{v_n}, \exp(2v_{v_n})), \tag{6.3}$$

where $\mathbf{v}_n = (v_{n_1}, \ldots, v_{n_N})$. As the variance model is not that important in this application, a conjugate inverse Gamma variance prior could be used here as well. Restriction to isotropic noise model with the noise covariance of the form $\lambda\mathbf{I}$ is also possible. The noise model and the generative model (6.1) imply the likelihood

$$p(\mathbf{x}(t)|\boldsymbol{\theta}_\mathbf{f}, \mathbf{s}(t), \mathbf{v}_n, \mathcal{H}) = N(\mathbf{x}(t);\ \mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}), \mathrm{diag}(\exp(2\mathbf{v}_n))). \tag{6.4}$$

The hierarchical priors of the MLP weights are similar to those of the noise except that the prior of $\mathbf{A}$ is fixed to unit variance to resolve the scaling ambiguity between $\mathbf{A}$ and $\mathbf{s}$, and the different columns of $\mathbf{B}$ have their own priors:

$$p(A_{ij}) = N(A_{ij};\ 0, 1) \tag{6.5}$$

$$p(B_{ij}|v_{B_j}) = N(B_{ij};\ 0, \exp(2v_{B_j})) \tag{6.6}$$

$$p(a_i|m_a, v_a) = N(a_i;\ m_a, \exp(2v_a)) \tag{6.7}$$

$$p(b_i|m_b, v_b) = N(b_i;\ m_b, \exp(2v_b)) \tag{6.8}$$

$$p(v_{B_j}|m_{v_B}, v_{v_B}) = N(v_{B_j};\ m_{v_B}, \exp(2v_{v_B})). \tag{6.9}$$

The highest level hyperparameters m_{v_n}, v_{v_n}, m_a, v_a, m_b, v_b, m_{v_B} and v_{v_B} have vague Gaussian priors $N(0, 100^2)$[1].

To complete the definition of the model, the prior of the sources $\mathbf{s}$ must still be determined. In this case there are several possibilities, leading to models with different benefits and drawbacks. The differences between the alternatives are discussed in more detail in Section 6.5 below.

The simplest source model is the Gaussian model used in nonlinear factor analysis (NFA). This is achieved by

$$p(\mathbf{s}(t)|\mathbf{v}_s) = N(\mathbf{s}(t);\ \mathbf{0}, \mathrm{diag}(\exp(2\mathbf{v}_s))) \tag{6.10}$$

$$p(v_{s_i}|m_{v_s}, v_{v_s}) = N(v_{s_i};\ m_{v_s}, \exp(2v_{v_s})), \tag{6.11}$$

where the hyperparameters m_{v_s} and v_{v_s} again have noninformative prior $N(0, 100^2)$. With these definitions, the sets of observations $\boldsymbol{X} = \{\mathbf{x}(t)|t\}$ and sources $\boldsymbol{S} = \{\mathbf{s}(t)|t\}$ are defined as usual. The parameter vector $\boldsymbol{\theta}$ contains everything described above, that is $\boldsymbol{\theta} = (\boldsymbol{\theta}_\mathbf{f}, \mathbf{v}_n, m_{v_n}, v_{v_n}, (v_{B_j}), m_a, v_a, m_b, v_b, m_{v_B}, v_{v_B}, \mathbf{v}_s, m_{v_s}, v_{v_s})$ including all j in v_{B_j}.

The Gaussian model is computationally simple, but it suffers from the same rotation indeterminacy as the linear FA model. This can be corrected in the same way linear IFA extends linear FA: by using a mixture of Gaussians as source prior. Introducing a new latent variable $M_i(t)$ to denote the active mixture component for source i at sample t leads to a nonlinear independent factor analysis (NIFA) model with a prior of the form

$$p(s_i(t)|M_i(t) = l, m_{sil}, v_{sil}) = N(s_i(t);\ m_{sil}, \exp(2v_{sil})) \tag{6.12}$$

$$p(m_{sil}|v_{m_s}) = N(m_{sil};\ 0, \exp(2v_{m_s})) \tag{6.13}$$

$$p(v_{sil}|m_{v_s}, v_{v_s}) = N(v_{sil};\ m_{v_s}, \exp(2v_{v_s})). \tag{6.14}$$

The mixing proportions have a logistic normal prior given by the softmax function

$$p(M_i(t) = l|c_{i\cdot}) = \exp(c_{il}) / \sum_{l'} \exp(c_{il'}) \tag{6.15}$$

$$p(c_{il}|v_c) = N(c_{il};\ 0, \exp(2v_c)). \tag{6.16}$$

All the highest level parameters v_{m_s}, m_{v_s}, v_{v_s} and v_c again have a noninformative prior $N(0, 100^2)$. The parameters $\boldsymbol{\theta}$ can be defined similarly as above.

In the following, the simpler NFA model is used instead of the more complex NIFA model. Most of the discussion generalises fairly easily to NIFA as well, as is shown in Publication I and in Valpola (2000).

6.3 Learning

In order to apply variational Bayesian learning to the NFA model defined above, there are several steps to consider. The form of the approximating distribution

[1] The data is usually preprocessed by scaling to approximately unit variance to ensure proper scaling of the weights so that the priors really are vague.

$q(\boldsymbol{S}, \boldsymbol{\theta})$ must be specified and the cost function (3.34) evaluated. An expression for the cost function allows defining update rules for all the variables. The flexible NFA model is susceptible to local minima and therefore requires a reasonable initialisation to achieve good results.

6.3.1 The variational approximation

In NFA, the approximating distribution $q(\boldsymbol{S}, \boldsymbol{\theta})$ is chosen to be fully factorial Gaussian distribution

$$q(\boldsymbol{S}, \boldsymbol{\theta}) = q(\boldsymbol{S})q(\boldsymbol{\theta}) = \prod_{i,t} q(s_i(t)) \prod_j q(\theta_j). \tag{6.17}$$

The individual factors are parameterised with variational parameters corresponding to the posterior mean and variance of the variable as

$$q(s_i(t)) = N(s_i(t);\ \overline{s}_i(t), \widetilde{s}_i(t)) \tag{6.18}$$

$$q(\theta_j) = N(\theta_j;\ \overline{\theta}_j, \widetilde{\theta}_j). \tag{6.19}$$

For parameters modelling the mean of a Gaussian, the Gaussian distribution is a conjugate prior and the optimal free-form approximation is of this form, assuming the factorisation. For parameters modelling the log-variance of a Gaussian, the Gaussian prior is not conjugate and the posterior approximation is only an approximation.

In case of NIFA, the approximation is not fully factorial. Instead, the dependences between $M_i(t)$ and $s_i(t)$ are modelled so that the approximation for them is of the form

$$q(M_i(t), s_i(t)) = q(s_i(t)|M_i(t))q(M_i(t)) \tag{6.20}$$

which yields a Gaussian mixture approximation for $s_i(t)$

$$q(s_i(t)) = \sum_l q(s_i(t)|M_i(t) = l)q(M_i(t) = l). \tag{6.21}$$

Otherwise the approximation is similar to NFA.

6.3.2 Evaluating the cost

The definition of the model and the approximating distribution allow evaluating the cost (3.34) as a function of the variational parameters. The cost can be split into two parts

$$\mathcal{C} = \mathcal{C}_q + \mathcal{C}_p = \langle \log q(\boldsymbol{S}, \boldsymbol{\theta}) \rangle + \langle -\log p(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{\theta}|\mathcal{H}) \rangle \,. \tag{6.22}$$

Using the factorisation of q, the term $\mathcal{C}_q$ splits into

$$\begin{aligned} \mathcal{C}_q = \langle \log q(\boldsymbol{S}, \boldsymbol{\theta}) \rangle &= \left\langle \log \prod_{i,t} q(s_i(t)) \prod_j q(\theta_j) \right\rangle \\ &= \sum_{i,t} \langle \log q(s_i(t)) \rangle + \sum_j \langle \log q(\theta_j) \rangle \,. \end{aligned} \tag{6.23}$$

The remaining terms are negative entropies of Gaussians having values depending only on the variance

$$\langle \log q(\theta_j) \rangle = -\frac{1}{2} - \frac{1}{2} \log(2\pi \widetilde{\theta}_j). \tag{6.24}$$

The $\mathcal{C}_p$ term is slightly more difficult. Using the definition of the model, it can be factored into terms

$$\begin{aligned} \mathcal{C}_p &= \langle -\log p(\boldsymbol{X}|\boldsymbol{S}, \boldsymbol{\theta}, \mathcal{H}) \rangle + \langle -\log p(\boldsymbol{S}|\boldsymbol{\theta}, \mathcal{H}) \rangle + \langle -\log p(\boldsymbol{\theta}|\mathcal{H}) \rangle \\ &= \sum_{i,t} \langle -\log p(x_i(t)|\mathbf{s}(t), \boldsymbol{\theta}, \mathcal{H}) \rangle + \sum_{i,t} \langle -\log p(s_i(t)|\boldsymbol{\theta}, \mathcal{H}) \rangle + \langle -\log p(\boldsymbol{\theta}|\mathcal{H}) \rangle \,. \end{aligned} \tag{6.25}$$

The terms in the second and third summand are expectations of negative logarithm of Gaussian pdf over Gaussian mean and log-variance parameters. These can be evaluated for example parameter $\theta \sim N(m, \exp(2v))$ through integrals of the form

$$\begin{aligned} \langle -\log p(\theta|m, v, \mathcal{H}) \rangle &= \iint -\log N(\theta;\ m, \exp(2v))\, q(m)\, q(v)\ dm\, dv \\ &= \frac{1}{2} \log(2\pi) + \overline{v} + \left[(\overline{\theta} - \overline{m})^2 + \widetilde{\theta} + \widetilde{m} \right] \exp(2\widetilde{v} - 2\overline{v}). \end{aligned} \tag{6.26}$$

More details on the evaluation of the integrals are presented by Lappalainen and Miskin (2000).

The terms of the first summand of Eq. (6.25),

$$\begin{aligned} \langle -\log p(x_i(t)|\mathbf{s}(t), \boldsymbol{\theta}, \mathcal{H}) \rangle &= \langle -\log N(x_i(t);\ f_i(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}), \exp(v_{n_i})) \rangle \\ &= \frac{1}{2} \log(2\pi) + \overline{v}_{n_i} + \left[(x_i(t) - \overline{f}_i(t))^2 + \widetilde{f}_i(t) \right] \exp(2\widetilde{v}_{n_i} - 2\overline{v}_{n_i}), \end{aligned} \tag{6.27}$$

are more difficult as they depend on the mean $\overline{f}_i(t)$ and variance $\widetilde{f}_i(t)$ of the outputs of the MLP network. Techniques for approximating these are presented below in Sec. 6.4.

6.3.3 Update algorithm

The posterior approximations of the parameters of the hierarchical model, that is those of the type m_θ and v_θ, can be updated using a standard variational EM algorithm (Lappalainen and Miskin, 2000) as also discussed in Publication I. The update rules for the sources $\boldsymbol{S}$ as well as the weights $\boldsymbol{\theta}_\mathbf{f}$ of the MLP network are more difficult and they are therefore presented in more detail.

Differentiating the split cost (6.22) with respect to $\widetilde{\theta}_j$ and using the evaluated result (6.24) yields

$$\frac{\partial \mathcal{C}}{\partial \widetilde{\theta}_j} = \frac{\partial \mathcal{C}_q}{\partial \widetilde{\theta}_j} + \frac{\partial \mathcal{C}_p}{\partial \widetilde{\theta}_j} = -\frac{1}{2\widetilde{\theta}_j} + \frac{\partial \mathcal{C}_p}{\partial \widetilde{\theta}_j}. \tag{6.28}$$

Setting this to zero leads to a fixed point update rule for the variances of the sources and MLP network weights

$$\widetilde{\theta}_j = \frac{1}{2} \left(\frac{\partial \mathcal{C}_p}{\partial \widetilde{\theta}_j} \right)^{-1}. \tag{6.29}$$

Blindly applying this rule may in some cases lead to instability. This can be corrected by some form of dampening, such as halving the step length in log scale until the update does not increase the value of the cost function. The variance must also not be set to a negative value, even if the derivative is negative. The required derivatives can be computed from the expression of the cost function. In case of inputs and weights of the MLP, this leads to similar computation as in backpropagation.

The means of $\boldsymbol{S}$ and $\boldsymbol{\theta}_{\mathbf{f}}$ are also updated with a gradient-based algorithm. The original algorithm in Publication I used a customised diagonal Newton approximation based on approximating the second derivatives with respect to the mean with derivatives with respect to the variance. This was later found out to be inefficient and was replaced with a standard conjugate gradient optimisation algorithm (Fletcher, 1987) in Publications VI and VII.

6.3.4 Initialisation

The MLP network and the gradient-based learning algorithms are notoriously prone to local minima (Fukumizu and Amari, 2000; Haykin, 1999). In order to achieve good results, the NFA method therefore requires a reasonable initialisation.

Starting from Publication I, the method has been initialised by setting the means of the sources to values given by suitable number of principal components of the data. The means of the MLP weights are initialised to random values while all the variances are initialised to small constant values. After this, only the MLP weights are updated during the first 20 iterations of the update algorithm[2] so that the model can learn a mapping from the PCA sources to the observations. The hyperparameters of the model are only updated after the first 100 iterations.

The PCA initialisation is easy to compute and sufficient for many purposes, but its linearity may sometimes lead to suboptimal results. To resolve this, kernel PCA (KPCA) was used in the initialisation in Publication V. The KPCA initialisations are sensitive to the choice of the kernel and its parameters, but with suitable choices they may yield significantly better results while using less time. The kernel can also be selected with the variational Bayesian criterion by running the learning algorithm for a few iterations with different initialisations and comparing the cost function values.

6.3.5 Model comparison and selection

As discussed in Sec. 3.1.7, the Bayesian approach allows direct comparison of different models through model evidence $p(\boldsymbol{X}|\mathcal{H})$. The cost function (3.34) used in the variational approach is essentially a lower bound on the evidence, and it can be used in model comparison.

In order to select the optimal number M of sources in the NFA model, the method

[2]The number of 20 iterations refers to the algorithm with conjugate gradient updates. With the original update algorithm, the number used to be 50.

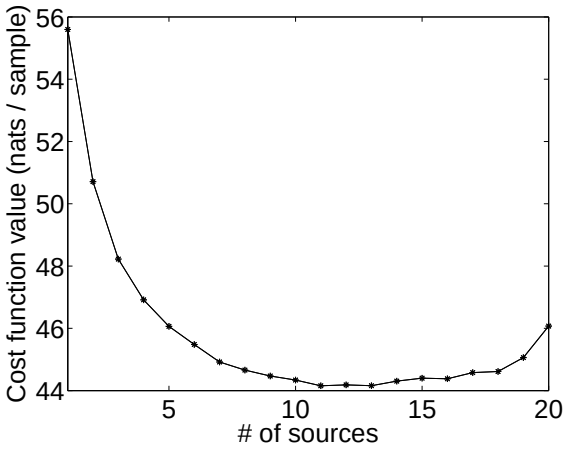

Figure 6.1: The value of the cost function is shown as a function of the number of sources. The MLP network had 40 hidden neurons. Four different initialisations were tested to find the mean value for each number of sources. The cost function saturates after around 11 sources and the deviations are due to different random initialisation of the network.

can be run several times using different values for M and comparing the costs. A plot of the cost function values typically shows a rather quick drop in the values as the number approaches the optimum. After the optimal value is passed, the cost function values may start to increase again, although more slowly as the method is able to prune out the unnecessary sources. This is illustrated in Fig. 6.1 for a data set of spectrograms of speech (Valpola et al., 2003b).

Selection of the optimal number of hidden neurons in the MLP network can be done similarly, although the procedure may not be as well-founded in this case. As the method can prune out unused hidden neurons it would be possible to always use a very large number, although that would be computationally inefficient. A practical compromise has been to experiment with a few values for the number of hidden neurons for a given data set but mostly use one reasonable value. The dependence of the cost on the number of hidden neurons in an experiment with the speech data set is illustrated in Fig. 6.2.

6.4 Approximating the nonlinearity

The mean $\overline{f}_i$ and variance $\widetilde{f}_i$ of the outputs of the MLP used in Eq. (6.27) can be evaluated by multidimensional Gaussian integrals

$$\overline{f}_i(t) = \iint f_i(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f})\, q(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f})\, d\mathbf{s}(t)\, d\boldsymbol{\theta}_\mathbf{f} \tag{6.30}$$

$$\widetilde{f}_i(t) = \iint \left(f_i(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f}) - \overline{f}_i(t)\right)^2 q(\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f})\, d\mathbf{s}(t)\, d\boldsymbol{\theta}_\mathbf{f}. \tag{6.31}$$

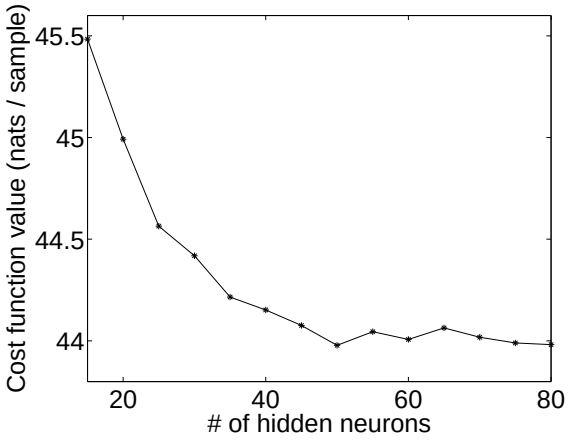

Figure 6.2: The value of the cost function is shown as a function of the number of hidden neurons in the MLP network modelling the nonlinear mapping from 11 sources to the speech data observations. The figure shows the mean result attained with four different initialisations. The value appears to saturate after approximately 50 neurons. The differences in values are significantly smaller than when comparing different numbers of sources.

These integrals depend on all the inputs and weights of the MLP network, thus leading to cases of the order of thousands of dimensions.

The textbook approach for evaluating these integrals numerically would use a Gaussian quadrature based on evaluating a weighted sum of the function values on an uneven grid of points. Unfortunately these methods do not scale to high dimensional problems. An n point approximation along one dimension leads to an n^d point grid in d dimensions, thus becoming intractable for large d even with very small n.

As much as one would want to have a faster algorithm, it is not possible in general. Curbera (2000) has shown that when the required error ϵ approaches zero, the worst-case complexity for evaluating that good approximation of the integral is of the order ϵ^{-d}, where d is the dimensionality of the input. Faster algorithms must therefore settle for larger potential error or utilise the specific structure of the problem at hand.

The very first approach for approximating the integrals in Eqs. (6.30) and (6.31) in context of nonlinear FA used by Lappalainen (1998) as well as Lappalainen and Giannakopoulos (1999) proceeded in the MLP network layer by layer and completely ignored the dependencies between different hidden neurons. This approach was very unreliable because of multiple paths of propagation of signals from the sources to the outputs. An example of the multiple paths is shown by the dashed lines in Fig. 6.3. Depending on the weights of the paths, the interference may be either reinforcing or suppressing. An improved approximation used in Publication I was to evaluate the moments of a full Taylor approximation of $\mathbf{f}$. This involved evaluating the Jacobian matrices of the MLP to track the interfering paths, thus

making the algorithm computationally more intensive. Even this method was later found to produce unreliable estimates in cases of large source posterior variance, which are common when trying to extract a large number of sources. This caused instability of the algorithm in such cases. This was corrected by an even better approximation studied in Publications VI and VII.

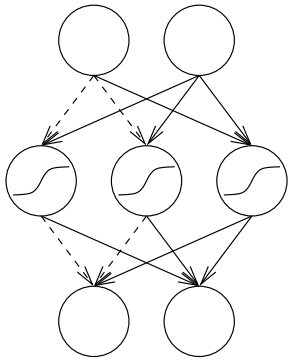

Figure 6.3: An illustration of the MLP network used to model the nonlinearity. The dashed lines show two interfering paths in the network from the same input to the same output.

6.4.1 Taylor approximation

The Taylor approximation is based on developing a Taylor series of the nonlinearity about the input mean. Denoting $\mathbf{u} = (\mathbf{s}(t), \boldsymbol{\theta}_\mathbf{f})$ with mean $\overline{\mathbf{u}}$ and covariance $\boldsymbol{\Sigma}_u$, the first order approximation of $f_i(\mathbf{u})$ can be written as

$$f_i(\mathbf{u}) \approx f_i(\overline{\mathbf{u}}) + \nabla_\mathbf{u} f_i(\overline{\mathbf{u}})(\mathbf{u} - \overline{\mathbf{u}}). \tag{6.32}$$

Using this in Eqs. (6.30) and (6.31) yields the approximations

$$\overline{f}_{i,\text{Taylor}} = f_i(\overline{\mathbf{u}}) \tag{6.33}$$

$$\widetilde{f}_{i,\text{Taylor}} = \nabla_\mathbf{u} f_i(\overline{\mathbf{u}}) \boldsymbol{\Sigma}_u \nabla_\mathbf{u} f_i(\overline{\mathbf{u}})^T. \tag{6.34}$$

The required derivatives $\nabla_\mathbf{u} f_i(\overline{\mathbf{u}})$ can be evaluated by propagating matrices of partial derivatives through the MLP. The resulting expression for instance for the derivatives with respect to the sources is

$$\frac{\partial}{\partial s_j} f_i(\overline{\mathbf{u}}) = \overline{\mathbf{B}}_{i,\cdot} \operatorname{diag}(\boldsymbol{\phi}'(\overline{\mathbf{y}}(t))) \overline{\mathbf{A}}_{\cdot,j}. \tag{6.35}$$

Here $\mathbf{y}(t) = \mathbf{A}\mathbf{s}(t)$ denotes the inputs of the hidden neurons. It is possible to use higher order approximations, but as discussed in Publication VI they are significantly less robust and therefore not useful in practice.

6.4.2 Other existing approximations

The most typical applications dependent on evaluating integrals like (6.30) and (6.31) are nonlinear extensions of Kalman filtering. Traditional algorithms such as extended Kalman filtering are mostly based on the Taylor approximation (Maybeck, 1979, 1982). The *unscented transform* and corresponding unscented Kalman filter were proposed by Julier and Uhlmann (1996) to help avoid some of the problems of the Taylor approximation. The filter has since been further refined for instance by Wan and van der Merwe (2001).

In a d-dimensional case, the unscented transform is based on selecting a set $\mathcal{Y}$ of $2d$ weighted points together with the mean point that describe well the input distribution. In case of diagonal input covariance, the points will reside on the coordinate axes at a distance governed by corresponding standard deviation. These points are then transformed individually to get a new set of points $\mathcal{Z}_i = \mathbf{f}(\mathcal{Y}_i)$. The output mean and covariance are then computed as weighted mean and covariance of the transformed points $\mathcal{Z}$. The procedure is illustrated in Fig. 6.4.

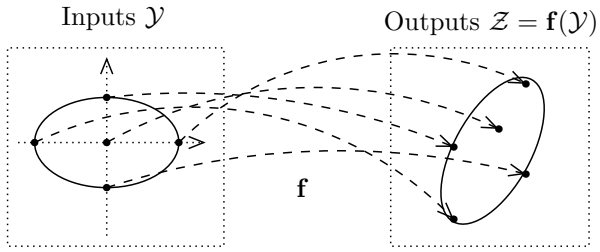

Figure 6.4: An illustration of the unscented transform. The selected points $\mathcal{Y}$ are mapped by $\mathbf{f}$ to $\mathcal{Z}$ and the weighted mean and covariance of the points $\mathcal{Z}$ are evaluated.

The unscented transform is intuitively appealing, but unfortunately it does not scale to high-dimensional problems and can even produce worse results than the Taylor approximation, as seen later in Sec. 6.4.4. The computational cost for the MLP case, which is linear in the total number of inputs and weights, can also get quite high when there are many sources.

6.4.3 Linearisation by Gauss–Hermite quadratures

As general Gaussian integration in high-dimensional spaces is extremely difficult, successful methods should take into account the specific form of the problem, if possible. The structure of the MLP network used in NFA is very specific: two layers of linear mappings with a layer of scalar nonlinearities ϕ at the hidden neurons in between. Without the nonlinear activation functions of the hidden neurons, the whole mapping would be linear and the mean and covariance of the output could be evaluated exactly. This suggests developing a better way to linearise the hidden neurons and therefore the whole mapping. This approach was studied in Publications VI and VII.

The approximation uses one-dimensional Gauss–Hermite quadrature (Hildebrand,

1956). The quadrature is a general method for evaluating integrals of the form

$$I(\phi) = \int_{-\infty}^{\infty} \phi(y) N(y;\, 0, 1)\, dy, \tag{6.36}$$

where $\phi : \mathbb{R} \to \mathbb{R}$ is a scalar function. The integral is approximated with a finite sum

$$I_{\mathrm{GH}}(\phi) = \sum_{i=1}^{n} w_i \phi(t_i) \approx I(\phi). \tag{6.37}$$

For an approximation using n points, the weights w_i and abscissas t_i can be selected so that the result is exact for all polynomials up to order $2n$. A 3-point approximation has been used in this work as it provides a good compromise between accuracy and efficiency. The evaluation points can be easily transformed to handle general mean and variance of the input distribution to get general expectation of $\phi(y)$

$$\overline{\phi}(y)_{\mathrm{GH}} := \sum_{i=1}^{n} w_i \phi\left(\overline{y} + t_i \sqrt{\widetilde{y}}\right) \approx \langle \phi(y) \rangle = \int_{-\infty}^{\infty} \phi(y) N(y;\, \overline{y}, \widetilde{y})\, dy. \tag{6.38}$$

Variance of $\phi(y)$ can be evaluated through

$$\widetilde{\phi}(y)_{\mathrm{GH}} := \sum_{i=1}^{n} w_i \left[\phi\left(\overline{y} + t_i \sqrt{\widetilde{y}}\right) - \overline{\phi}(y)_{\mathrm{GH}}\right]^2 \approx \left\langle \left[\phi(y) - \langle \phi(y) \rangle\right]^2 \right\rangle. \tag{6.39}$$

Both the evaluated mean and variance of $\phi(y)$ depend on both mean and variance of y in a nonlinear manner capable of taking into account the specific form of function ϕ.

The evaluated mean and variance can be used to define an effective linearisation of the hidden neurons by finding a corresponding linear function that would yield the same mean and variance. This yields the effective linearisation

$$\langle \phi(y_i(t)) \rangle := \overline{\phi}(y_i(t))_{\mathrm{GH}} \tag{6.40}$$

$$\langle \phi'(y_i(t)) \rangle := \sqrt{\frac{\widetilde{\phi}(y_i(t))_{\mathrm{GH}}}{\widetilde{y}_i(t)}}. \tag{6.41}$$

The linearisation procedure is illustrated in Fig. 6.5. The effective linearisation is able to take into account the variance of the input, thus following the form of the function more globally when the variance is large. The effective linearisation of the hidden neurons can now be used in place of the derivatives in Eq. (6.35) and others to evaluate a global linearisation of $\mathbf{f}$. The mean and variance of the global linearisation can be evaluated exactly in a straightforward manner. More details on the procedure can be found in Publications VI and VII.

6.4.4 Comparisons

The results of a comparison of the accuracies of the moments evaluated with different approximations from Publication VI are presented in Fig. 6.6. In this

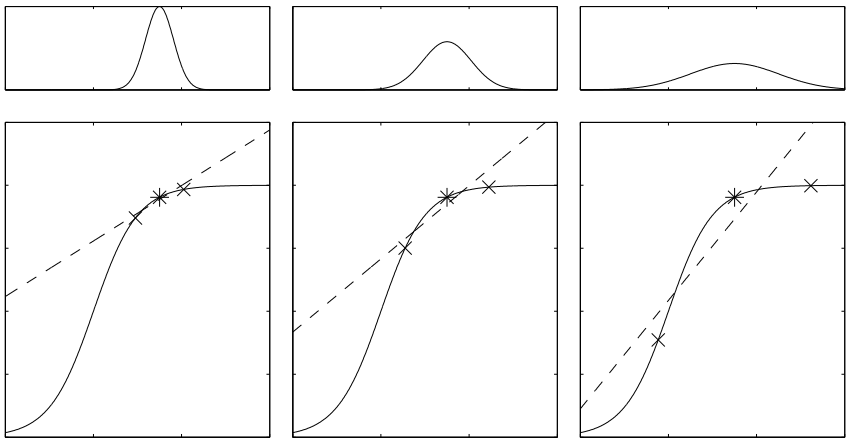

Figure 6.5: An illustration of the effective linearisations evaluated by the Gauss–Hermite quadrature. The upper panels show three Gaussian input distributions with same mean $\overline{y} = 1.5$ and different variances $\widetilde{y} = 0.1, 0.3, 1$. The nonlinear activation function is shown in solid line in the bottom panels together with the basis points (crosses) and linearisations (dashed lines) corresponding to the different input distributions.

comparison, random MLP networks with random inputs were used to test the evaluation of mean and diagonal elements of the covariance of the output. The MLP networks had 5 inputs, 30 hidden neurons and 10 outputs. The means of the distribution $q(\mathbf{s}, \boldsymbol{\theta}_\mathbf{f})$ over the inputs and the weights were selected randomly while the variances were all equal. The variance of the weights was fixed to a small value while the variance of the inputs was varied. All the results were compared to a reference value evaluated by sampling. The results show clear deterioration of the quality of the Taylor approximation, as the input variance increases. With large input variance, the second order approximation of the mean, which was used in Publication I, is significantly less accurate than even the first order approximation. The unscented transform is also surprisingly inaccurate due to the high dimensionality of the problem. The proposed approximation yields consistently the most accurate results.

The results of a more realistic comparison of the accuracies of the cost function approximations from Publication VII are presented in Fig. 6.7. This comparison is based on using the actual NFA learning algorithm with different approximations. Different approximations lead to different learning algorithms and thus different end results, and therefore there is no direct correspondence between the points in different figures. The results were compared against a reference value evaluated by using sampling to approximate the nonlinearity for the same approximating distribution. Each of the 80 marks in the figures represents the result of running the NFA algorithm using the same data set with 4 different random initialisations of the MLP and different number of sources, ranging from 1 to 20. The cost function results of the proposed algorithm show a very good correlation with the true cost, although the actual values are often slightly underestimated. In case of the Taylor approximation, there is a set of simulations whose reported costs are

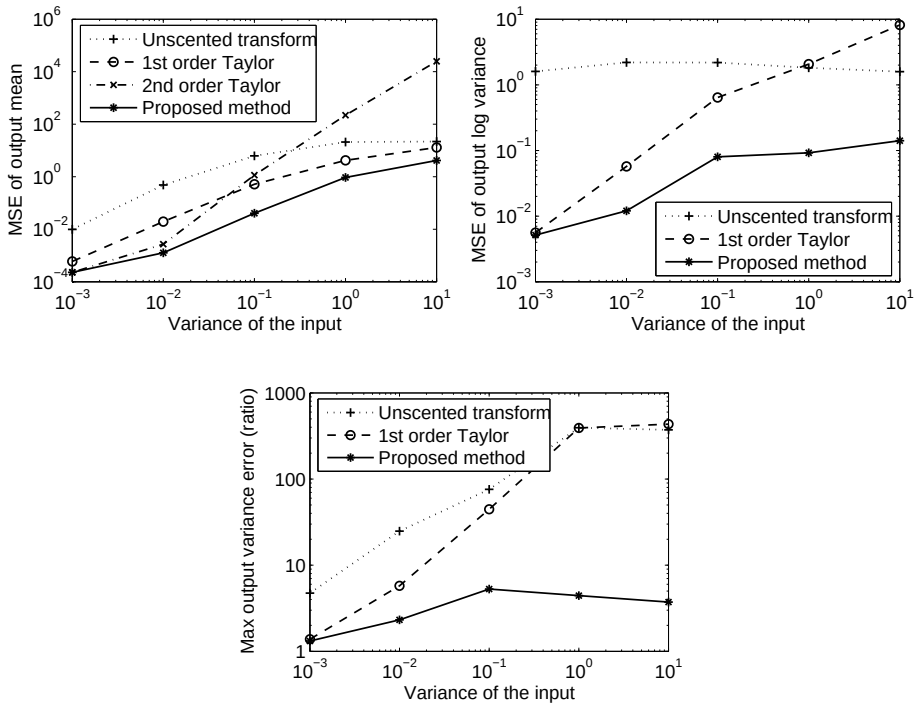

Figure 6.6: Accuracies of the moment approximations evaluated by different methods are compared to a reference value evaluated by sampling. The results are shown as a function of the variance of the input. The top left subfigure shows the mean square error of the mean and the top right subfigure the mean square error of the logarithm of the variance. The bottom subfigure shows the maximal underestimation of variance as a ratio of the true variance over the estimated variance. Underestimation of variance can cause underestimation of the cost function, making it the most serious error the methods can do. (From Publication VI.)

very low even though the true cost can be significantly higher. These correspond to the cases with large numbers of sources, as shown in Publication VII.

6.5 On different source models

Two different source models were presented above: a Gaussian nonlinear factor analysis (NFA) model in Eq. (6.10) and a mixture-of-Gaussians based nonlinear *independent* factor analysis (NIFA) model in Eq. (6.12). From purely theoretical perspective, the NIFA model is preferable as its non-Gaussian model is able to resolve the rotation indeterminacy inherent to the Gaussian model.

In practice, things are not quite that simple. The NIFA model with its fully factorial posterior approximation seems to suffer from similar problems of not being able to extract the true independent sources as linear algorithms based on similar principles, as discussed in Sec. 4.3.2.

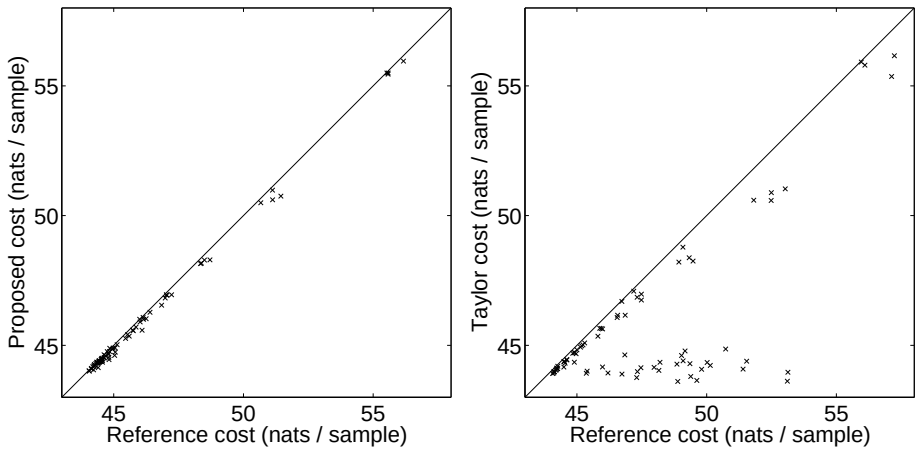

Figure 6.7: Accuracies of the NFA cost function approximations evaluated by different methods are compared to a reference value evaluated by sampling. (From Publication VII.)

Using the more complicated NIFA model slows down the computation and requires careful initialisation of the mixture components. As this seems to cause more trouble than offer benefits, we have usually chosen not to do it and only use plain NFA (Valpola et al., 2003b). In order to achieve BSS, standard linear ICA can be applied as postprocessing to the extracted sources. This approach resembles the one used by Ikeda (2000) as well as Ikeda and Toyama (2000) for linear separation, where linear factor analysis is used as preprocessing for ICA instead of PCA. Use of factor analysis preprocessing has been shown to lead to good results in case of noisy linear mixtures. The nonlinear case is, however, more difficult as the model may try to nonlinearly transform the sources to be more Gaussian than they should be. Some evidence of this behaviour can be seen in the experiments of Publication IV. Even with a Gaussian source model the sources are, however, not fully Gaussian and using linear ICA to determine the rotation is thus possible.

6.6 Variants and extensions

The basic NFA model can be easily specialised for post-nonlinear mixtures, or generalised to include dynamics of the sources or missing values in the observations.

6.6.1 Post-nonlinear mixtures

As discussed in Section 5.2, post-nonlinear (PNL) mixtures are restricted nonlinear mixtures of the form

$$x_i = f_i\left(\sum_{j=1}^{M} a_{ij} s_j\right) + n_i, \quad i = 1, \ldots, N \tag{6.42}$$

where the functions $f_i : \mathbb{R} \to \mathbb{R}$ are called the post-nonlinearities. PNL mixtures are a theoretically important special case of nonlinear ICA, because they can be proven to be separable if the post-nonlinearities are invertible and the mixing matrix satisfies certain regularity conditions (Taleb and Jutten, 1999; Theis, 2004).

A variational Bayesian approach to post-nonlinear mixtures is presented in Publication IV. The post-nonlinear factor analysis (PNLFA) model is based on the generative model

$$x_i = f_i\left(\sum_{j=1}^{M} a_{ij}s_j;\ \boldsymbol{\theta}_{f_i}\right) + n_i, \tag{6.43}$$

where the post-nonlinearities f_i are modelled with MLP networks with weights $\boldsymbol{\theta}_{f_i}$. This approach allows also non-invertible post-nonlinearities.

The evaluation of the cost function and the learning process of the PNLFA model are similar to those of the general NFA. Approximating the nonlinearity is a little easier, as there is only one "source" input with larger posterior variance. The posterior variances of the MLP weights are typically smaller as a single weight affects and thus gains evidence from several observations. This allows using a hybrid of a Gaussian quadrature with respect to the MLP inputs $y_i(t) = \sum_{j=1}^{M} a_{ij}s_j$ and a Taylor approximation with respect to the weights $\boldsymbol{\theta}_{f_i}$, as presented in Publication IV. The more general approach of Publications VI and VII could of course be used as well.

The source model used in PNLFA is Gaussian. It could in principle be rather easily replaced with mixtures-of-Gaussians, but simple post-processing of the extracted sources with linear ICA is often sufficient, as discussed above in Sec. 6.5.

6.6.2 Including dynamics: nonlinear state-space model

The NFA model can be extended by adding another MLP to model the dynamics of the sources $\mathbf{s}(t)$ through

$$\mathbf{s}(t) = \mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_g) + \mathbf{m}(t), \tag{6.44}$$

where $\mathbf{g}$ is a nonlinear mapping with parameters $\boldsymbol{\theta}_g$ modelling the dynamics and $\mathbf{m}(t)$ is an additional Gaussian noise or innovation process term. Eq. (6.44) generalises the NFA model to a *nonlinear state-space model* (nonlinear SSM) (Valpola et al., 2002b; Valpola and Karhunen, 2002). In this context, the variables $\mathbf{s}(t)$ are more commonly referred to as *states*. The mapping $\mathbf{g}$ is of the form $\mathbf{g}(\mathbf{s}) = \mathbf{s} + \mathbf{g}_{\mathrm{MLP}}(\mathbf{s})$ to use the MLP only to model the differences of consecutive states.

Evaluation of the cost and the learning process of the nonlinear SSM are again mostly similar to the NFA. The improved approximation presented in Sec. 6.4 could easily be applied to the nonlinear SSM as well. Initial experiments using the improved approximation for the nonlinearities $\mathbf{f}$ and $\mathbf{g}$ are promising.

6.6.3 Missing observations

The Bayesian framework facilitates easy handling of missing or partially missing elements in the data matrix (Raiko, 2004). The simplest method for accomplishing this is to simply ignore the gradients arising from the missing elements during learning. This approach has been applied to NFA by Raiko and Valpola (2001).

6.6.4 Hierarchical nonlinear factor analysis

The computational complexity of learning the NFA model is quadratic with respect to the number of the sources. This arises from the need to keep track of possible multiple paths of signal propagation from the MLP inputs through different hidden neurons to the outputs. An approximation ignoring the dependencies of the hidden neurons is too crude and leads to poor results. This leaves changing the model the only way to decrease the computational complexity.

One way of defining a more efficiently learnable new model is to introduce additional latent variables to the hidden neurons of the MLP-like network as is done in the hierarchical nonlinear factor analysis (HNFA) model by Valpola et al. (2003c). The HNFA model is defined by the equations

$$\mathbf{h}(t) \sim N(\mathbf{A}\mathbf{s}(t) + \mathbf{a}, \mathbf{\Sigma}_h) \tag{6.45}$$

$$\mathbf{x}(t) \sim N(\mathbf{C}\mathbf{s}(t) + \mathbf{B}\boldsymbol{\phi}(\mathbf{h}(t)) + \mathbf{b}, \mathbf{\Sigma}_x), \tag{6.46}$$

where $\mathbf{h}(t)$ are the latent variables modelling the values of the hidden neurons, $\mathbf{\Sigma}_h$ is the covariance matrix of the noise or innovation of the hidden neurons, $\mathbf{\Sigma}_x$ is the noise covariance matrix, $\boldsymbol{\phi}$ is a vector of activation functions and $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{a}$, $\mathbf{b}$ are the weights of the mapping. The activation function is chosen to be $\phi(y) = \exp(-y^2)$, for computational simplicity. The structure of the model is illustrated in Fig. 6.8.

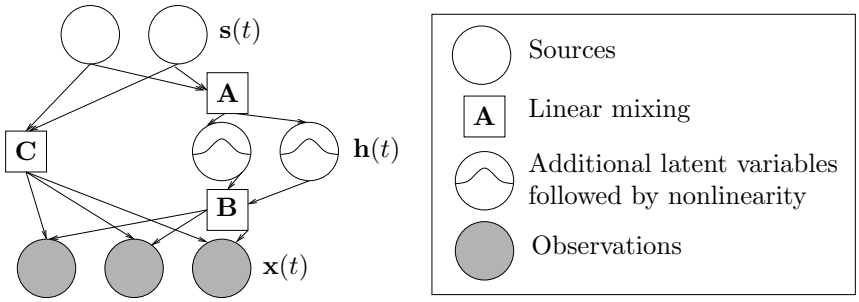

Figure 6.8: The HNFA model is illustrated. Square nodes correspond to weight matrices and round nodes to the variables with shaded nodes being observed and unshaded latent.

The HNFA model can be implemented efficiently using the building block framework (Valpola et al., 2001). As the additional latent variables $\mathbf{h}(t)$ are independent by definition of the factorial posterior approximation, the problem of multi-path

propagation is solved. The problem with this approach is that the new hidden neurons may try to act partially as new sources instead of simple computational units, thus making the interpretation of the results more difficult. Possible techniques for partially avoiding this problem are discussed in Publication III.

A comparison of the NFA and HNFA methods in predicting missing observations was presented by Raiko et al. (2003). The results indicate that in a nonlinear problem, HNFA typically lies between the inferior linear FA and slightly superior NFA in performance.

6.7 Applications

Nonlinear FA and BSS methods are relatively new and suffer from certain theoretical difficulties, and there are therefore not yet that many applications using them. The difficulty of interpreting or even analysing the nonlinear mapping compared to the single separating, mixing or loading matrix of the linear methods can also be problematic. One of the few applications of general nonlinear BSS is the problem of separation of two images printed on different sides of a semitransparent onion skin paper from scans of both sides of the paper as presented by Almeida and Faria (2004). The onion skin paper example is interesting, because it is a well-defined realistic and clearly nonlinear problem. Overall, experts of different application domains have expressed interest in the possibility of performing nonlinear FA and BSS, so the list of applications will hopefully get longer in the future.

The presented nonlinear factor analysis and BSS methods have been tested on a few real world data sets. Publication I presents results of the method using a set of measurements from an industrial pulp process. The data was preprocessed by a human expert to remove the time lags to make the instantaneous mixing model more suitable. The nonlinear method was able to find significantly more compact representation on the data than linear method. The estimated independent factors seem to have some interesting structure, although they have not been carefully analysed. As the data still has clear temporal structure, the results of the static model may well be mainly bumps and use of a dynamical model would be preferred, as discussed in Sec. 4.3.1.

Another benchmark data set that has been used consists of 30 dimensional spectrograms of Finnish speech. The preprocessing of the data follows the typical procedure used in speech recognition, where the energies of overlapping short term spectrograms are converted to the Mel scale mimicing the perception properties of the human ear. Results of experiments using this data set have been first presented by Valpola et al. (2003b) but later also in Publications VI and VII. The nonlinear model is again able to find more compact representation and attain higher marginal likelihood, but the estimated latent sources have not been analysed further. Again, a dynamical model would evidently be better suited for the data.

Analysis of biomedical imaging data is one of the most important applications of ICA and it is a natural application for the variational Bayesian methods as well. In magnetoencephalography (MEG), the mixing of magnetic fields from the brain to the sensors is governed by Maxwell equations and can be shown to be

linear and instantaneous (Vigário et al., 2000; Honkela et al., 2005). The interesting signals have a clear temporal structure, making it natural to use a slightly simplified version of the nonlinear SSM with linear observation mapping $\mathbf{f}$ but nonlinear dynamical mapping $\mathbf{g}$, as was done by Särelä et al. (2001). The method used by Särelä et al. (2001) was shown to be able to separate bursting rhythmic brain activity patterns that would be difficult to detect with traditional ICA methods. Future directions in this field include extensions to functional magnetic resonance imaging (fMRI) data, in which case even the mixing would appear to be nonlinear (Friston et al., 2000).

A temporal model of a time series facilitates both prediction of future values and detection of changes. The nonlinear SSM was applied by Ilin et al. (2004b) to change detection in an artificial time series consisting of two nonlinearly mixed Lorenz processes and a harmonic oscillator. The changes were detected by monitoring the log-probability $\log p(\mathbf{x}(t+1)|\mathbf{x}(t), \ldots, \mathbf{x}(1))$ of a new sample given the history. This can be approximated by the difference of cost functions (3.35) evaluated for the original and the augmented data set, when the Kullback–Leibler divergence term is assumed to be small

$$\log p(\mathbf{x}(t+1)|\mathbf{x}(t), \ldots, \mathbf{x}(1)) \approx \mathcal{C}(\{\mathbf{x}(t+1), \mathbf{x}(t), \ldots, \mathbf{x}(1)\}) - \mathcal{C}(\{\mathbf{x}(t), \ldots, \mathbf{x}(1)\}). \tag{6.47}$$

The method was shown to outperform conventional change detection methods by a clear margin. Initial experiments have shown that the nonlinear SSM method could be well suited for controlling the process in addition to monitoring (Raiko and Tornio, 2005).

Chapter 7

Conclusions

Most natural phenomena are inherently nonlinear, yet most statistical methods used to analyse them are linear. The methods presented in this thesis address this issue by providing nonlinear generalisations of several well-known statistical models including factor analysis (FA), independent component analysis (ICA) and blind source separation (BSS). They can also be easily generalised to nonlinear state-space models (SSMs).

After introducing the initial methods, much of the work presented concentrated on improving them by making them faster and more stable. One of the general purpose improvements presented was a method to accelerate convergence in alternating optimisation algorithms such as EM and variational EM algorithms using pattern searches. The method is very easy to implement and can yield significant speedups, especially in cases of low noise. Another important improvement was a novel approximation of the moments of a nonlinear transform of a probability distribution using effective linearisations evaluated by Gauss–Hermite quadratures. The method is interesting as the same problem is addressed by the Taylor approximation used in the extended Kalman filter as well as the more advanced unscented transform used in the unscented Kalman filter. These alternatives were both found to be significantly inferior to the proposed scheme at least in this application. The proposed initialisation of the variational Bayesian nonlinear BSS method using kernel PCA not only improved the performance of the nonlinear BSS, but suggested a potential method for kernel selection using the variational Bayesian criterion. A specialised version of the nonlinear BSS method for separation of post-nonlinear mixtures was presented as well. The variational Bayesian method incorporates easy handling of noise and allows separating mixtures with non-invertible post-nonlinearities, which is not possible with other existing techniques.

The nonlinear factor analysis and independent factor analysis models are not trouble-free. In case of linear mixtures of independent sources, the original sources can be recovered based on their independence alone. With nonlinear mixtures, this is not possible. All known theoretical separability results for nonlinear mixtures require strict limitations for the form of the nonlinearity, such as constraining the mixture to be only post-nonlinear. Temporal correlations of the sources can help

to resolve these indeterminacies and allow easier separation of independent dynamic processes. Additionally, the goal in applications with temporally correlated data and nonlinear SSMs is more often related to prediction of future behaviour of the time series or classifying different states, and therefore extracting a specific latent representation is not that important. Learning a good state representation and the dynamics is not easy, and a static model may offer a good starting point for experimentation.

An important problem with nonlinear models is the difficulty of interpretation of the results. In the linear case, the results can be summarised in a matrix clearly showing which latent variable affects which observed variable. In the nonlinear case this is not possible as the effects of different variables cannot be studied separately. New advanced visualisation and analysis techniques are therefore needed to fully utilise the presented methods.

The most obvious line of future work is to implement the improvements suggested in the thesis in the nonlinear SSM (Valpola and Karhunen, 2002) as well. This should help in improving the stability of the learning process and hopefully also significantly reducing the number of iterations needed for the method to converge.

The post-nonlinear model presented in Publication IV could be improved and compared to other similar methods. The nonlinearity approximation used there is an earlier variant of the Gauss–Hermite linearisation approach applied to the general nonlinear model in Publications VI and VII. The linearisation procedure could be applied to the PNL case as well. A mixture-of-Gaussians source model with linear response or TAP corrections to account for the posterior correlations of the sources would really show the full benefits of the approach. Necessary and sufficient conditions for separability of such mixtures should also be studied.

Classification of different dynamic regimes of an SSM can be done elegantly by including switching in the model. A variational Bayesian approach to linear switching SSMs was presented by Ghahramani and Hinton (2000), and a similar approach could probably be applied to the nonlinear case as well. An initial attempt of this was presented in the Master's thesis of the author (Honkela, 2001), but the specific model structure used there is far from optimal.

The algorithms presented here operate in batch mode, that is they require the full data set to be available all the time. This rules out certain potential applications and an on-line variant of the methods would be desirable. An on-line version of the variational learning methodology has been presented already by Sato (2001). Applying the methods to the nonlinear models should not be impossible, and similar methods have been applied to the building blocks framework and ICA by Honkela and Valpola (2003).

Bibliography

Almeida, L. B. (2003). MISEP – linear and nonlinear ICA based on mutual information. *Journal of Machine Learning Research*, 4:1297–1318.

Almeida, L. B. (2004). Linear and nonlinear ICA based on mutual information – the MISEP method. *Signal Processing*, 84(2):231–245.

Almeida, L. B. and Faria, M. (2004). Separating a real-life nonlinear mixture of images. In Puntonet, C. G. and Prieto, A., editors, *Proceedings of the Fifth International Conference on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, volume 3195 of *Lecture Notes in Computer Science*, pages 734–741, Granada, Spain. Springer-Verlag, Berlin.

Amari, S. (1985). *Differential-Geometrical Methods in Statistics*, volume 28 of *Lecture Notes in Statistics*. Springer-Verlag.

Amari, S. (1995). Information geometry of the EM and em algorithms for neural networks. *Neural Networks*, 8(9):1379–1408.

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276.

Amari, S., Cichocki, A., and Yang, H. (1996). A new learning algorithm for blind signal separation. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Advances in Neural Information Processing Systems 8*, pages 757–763. MIT Press, Cambridge, MA, USA.

Amari, S., Ikeda, S., and Shimokawa, H. (2001). Information geometry of α-projection in mean field approximation. In Opper, M. and Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, pages 241–257. The MIT Press, Cambridge, MA, USA.

Amari, S. and Nagaoka, H. (2000). *Methods of Information Geometry*. American Mathematical Society and Oxford University Press.

Attias, H. (1999). Independent factor analysis. *Neural Computation*, 11(4):803–851.

Attias, H. (2000a). Independent factor analysis with temporally structured sources. In Solla, S., Leen, T., and Müller, K.-R., editors, *Advances in Neural Information Processing Systems 12*, pages 386–392. MIT Press, Cambridge, MA, USA.

Attias, H. (2000b). A variational Bayesian framework for graphical models. In Solla, S., Leen, T., and Müller, K.-R., editors, *Advances in Neural Information Processing Systems 12*, pages 209–215. MIT Press, Cambridge, MA, USA.

Attias, H. (2001). ICA, graphical models and variational methods. In Roberts, S. and Everson, R., editors, *Independent Component Analysis: Principles and Practice*, pages 95–112. Cambridge University Press.

Bach, F. R. and Jordan, M. I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48.

Bazaraa, M. S., Sherali, H. D., and Shetty, C. M. (1993). *Nonlinear Programming: Theory and Algorithms*. J. Wiley, second edition.

Bell, A. and Sejnowski, T. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159.

Belouchrani, A., Abed-Meraim, K., Cardoso, J.-F., and Moulines, E. (1997). A blind source separation technique based on second-order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–44.

Bernardo, J. M. and Smith, A. F. M. (2000). *Bayesian Theory*. J. Wiley.

Bezdek, J. C. and Hathaway, R. J. (2003). Two new convergence results for alternating optimization. In Fogel, D. B. and Robinson, C. J., editors, *Computational Intelligence: The Experts Speak*, pages 149–164. J. Wiley and IEEE Press.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*. Clarendon Press.

Bishop, C. M., Spiegelhalter, D., and Winn, J. (2003). VIBES: A variational inference engine for Bayesian networks. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*, pages 793–800. MIT Press, Cambridge, MA, USA.

Blaschke, T. and Wiskott, L. (2004). Independent slow feature analysis and nonlinear blind source separation. In Puntonet, C. G. and Prieto, A., editors, *Proceedings of the Fifth International Conference on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, volume 3195 of *Lecture Notes in Computer Science*, pages 742–749, Granada, Spain. Springer-Verlag, Berlin.

Cardoso, J.-F. (1999). High-order contrasts for independent component analysis. *Neural Computation*, 11(1):157–192.

Cardoso, J.-F. and Souloumiac, A. (1993). Blind beamforming for non Gaussian signals. *IEE Proceedings-F*, 140:362–370.

Chan, K., Lee, T.-W., and Sejnowski, T. J. (2002). Variational learning of clusters of undercomplete nonsymmetric independent components. *Journal of Machine Learning Research*, 3:99–114.

Chan, K., Lee, T.-W., and Sejnowski, T. J. (2003). Variational Bayesian learning of ICA with missing data. *Neural Computation*, 15(8):1991–2011.

Choudrey, R. A. and Roberts, S. J. (2001). Flexible Bayesian independent component analysis for blind source separation. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 90–95, San Diego, USA.

Choudrey, R. A. and Roberts, S. J. (2003). Variational mixture of Bayesian independent component analyzers. *Neural Computation*, 15(1):213–252.

Cichocki, A. and Amari, S. (2002). *Adaptive Blind Signal and Image Processing*. J. Wiley.

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314.

Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. J. Wiley, New York.

Cowell, R. (1999). Introduction to inference for Bayesian networks. In Jordan, M. I., editor, *Learning in Graphical Models*, pages 9–26. The MIT Press, Cambridge, MA, USA.

Cox, R. T. (1946). Probability, frequency and reasonable expectation. *American Journal of Physics*, 14(1):1–13.

Csató, L., Opper, M., and Winther, O. (2002). TAP Gibbs free energy, belief propagation and sparsity. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 657–663. MIT Press, Cambridge, MA, USA.

Curbera, F. (2000). Delayed curse of dimension for Gaussian integration. *Journal of Complexity*, 16(2):474–506.

Darmois, G. (1953). Analyse générale des liaisons stochastiques. *Rev. Inst. Internationale Statist.*, 21:2–8.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)*, 39(1):1–38.

Douglas, S. C. and Amari, S. (2000). Natural-gradient adaptation. In Haykin, S., editor, *Unsupervised Adaptive Filtering, Vol. I*, pages 13–61. J. Wiley.

Eriksson, J. and Koivunen, V. (2004). Identifiability, separability, and uniqueness of linear ICA models. *IEEE Signal Processing Letters*, 11(7):601–604.

Fletcher, R. (1987). *Practical Methods of Optimization*. J. Wiley, second edition.

Frey, B. J. and Hinton, G. E. (1997). Efficient stochastic source coding and an application to a Bayesian network source model. *The Computer Journal*, 40(2/3):157–165.

Friston, K. J., Mechelli, A., Turner, R., and Price, C. J. (2000). Nonlinear responses in fMRI: The balloon model, Volterra kernels, and other hemodynamics. *NeuroImage*, 12(4):466–477.

Fukumizu, K. and Amari, S. (2000). Local minima and plateaus in hierarchical structures of multilayer perceptrons. *Neural Networks*, 13(3):317–327.

Funahashi, K. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3):183–192.

Gelman, A., Carlin, J., Stern, H., and Rubin, D. (1995). *Bayesian Data Analysis*. Chapman & Hall/CRC Press, Boca Raton, Florida.

Ghahramani, Z. and Beal, M. (2001a). Propagation algorithms for variational Bayesian learning. In Leen, T., Dietterich, T., and Tresp, V., editors, *Advances in Neural Information Processing Systems 13*, pages 507–513. The MIT Press, Cambridge, MA, USA.

Ghahramani, Z. and Beal, M. J. (2001b). Graphical models and variational methods. In Opper, M. and Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, pages 161–177. The MIT Press, Cambridge, MA, USA.

Ghahramani, Z. and Hinton, G. E. (2000). Variational learning for switching state-space models. *Neural Computation*, 12(4):831–864.

Girolami, M. (2001). A variational method for learning sparse and overcomplete representations. *Neural Computation*, 13(11):2517–2532.

Harman, H. H. (1960). *Modern Factor Analysis*. The University of Chicago Press.

Harmeling, S., Ziehe, A., Kawanabe, M., and Müller, K.-R. (2003). Kernel-based nonlinear blind source separation. *Neural Computation*, 15(5):1089–1124.

Haykin, S. (1999). *Neural Networks – A Comprehensive Foundation, 2nd ed.* Prentice-Hall.

Haykin, S., editor (2000). *Unsupervised Adaptive Filtering, Vol. 1: Blind Source Separation*. J. Wiley.

Heskes, T. (2004). On the uniqueness of loopy belief propagation fixed points. *Neural Computation*, 16(11):2379–2413.

Hildebrand, F. B. (1956). *Introduction to Numerical Analysis*. McGraw-Hill.

Hinton, G. E. and van Camp, D. (1993). Keeping neural networks simple by minimizing the description length of the weights. In *Proceedings of the 6th Annual ACM Conference on Computational Learning Theory*, pages 5–13, Santa Cruz, CA, USA.

Hochreiter, S. and Schmidhuber, J. (1999a). Feature extraction through LOCOCODE. *Neural Computation*, 11(3):679–714.

Hochreiter, S. and Schmidhuber, J. (1999b). LOCOCODE performs nonlinear ICA without knowing the number of sources. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA'99)*, pages 149–154, Aussois, France.

Højen-Sørensen, P., Winther, O., and Hansen, L. K. (2002). Mean-field approaches to independent component analysis. *Neural Computation*, 14(4):889–918.

Honkela, A. (2001). Nonlinear switching state-space models. Master's thesis, Helsinki University of Technology, Espoo.

Honkela, A., Östman, T., and Vigário, R. (2005). Empirical evidence of the linear nature of magnetoencephalograms. In *Proceedings of the 13th European Symposium on Artificial Neural Networks (ESANN 2005)*, Bruges, Belgium. To appear.

Honkela, A. and Valpola, H. (2003). On-line variational Bayesian learning. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 803–808, Nara, Japan.

Hooke, R. and Jeeves, T. A. (1961). 'Direct search' solution of numerical and statistical problems. *Journal of the ACM*, 8(2):212–229.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634.

Hyvärinen, A. (2001a). Blind source separation by nonstationarity of variance: a cumulant-based approach. *IEEE Transactions on Neural Networks*, 12(6):1471–1474.

Hyvärinen, A. (2001b). Complexity pursuit: Separating interesting components from time-series. *Neural Computation*, 13(4):883–898.

Hyvärinen, A., Karhunen, J., and Oja, E. (2001). *Independent Component Analysis*. J. Wiley.

Hyvärinen, A. and Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492.

Hyvärinen, A. and Pajunen, P. (1999). Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439.

Hyvärinen, A., Särelä, J., and Vigário, R. (1999). Spikes and bumps: Artefacts generated by independent component analysis with insufficient sample size. In *Proceedings of the International Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 425–429, Aussois, France.

Ikeda, S. (2000). ICA on noisy data: A factor analysis approach. In Girolami, M., editor, *Advances in Independent Component Analysis*, pages 201–215. Springer-Verlag, Berlin.

Ikeda, S., Tanaka, T., and Amari, S. (2004). Stochastic reasoning, free energy, and information geometry. *Neural Computation*, 16(9):1779–1810.

Ikeda, S. and Toyama, K. (2000). Independent component analysis for noisy data — MEG data analysis. *Neural Networks*, 13(10):1063–1074.

Ilin, A., Achard, S., and Jutten, C. (2004a). Bayesian versus constrained structure approaches for source separation in post-nonlinear mixtures. In *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IJCNN 2004)*, pages 2181–2186, Budapest, Hungary.

Ilin, A. and Valpola, H. (2003). On the effect of the form of the posterior approximation in variational learning of ICA models. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 915–920, Nara, Japan.

Ilin, A., Valpola, H., and Oja, E. (2004b). Nonlinear dynamical factor analysis for state change detection. *IEEE Transactions on Neural Networks*, 15(3):559–575.

Istrail, S. (2000). Statistical mechanics, three-dimensionality and NP-completeness: I. Universality of intractability for the partition function of the Ising model across non-planar surfaces. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 87–96, Portland, Oregon, USA.

Jaakkola, T. S. (2001). Tutorial on variational approximation methods. In Opper, M. and Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, pages 129–159. The MIT Press, Cambridge, MA, USA.

Jaynes, E. T. (2003). *Probability Theory: The Logic of Science*. Cambridge University Press, Cambridge, UK.

Jordan, M., Ghahramani, Z., Jaakkola, T., and Saul, L. (1999). An introduction to variational methods for graphical models. In Jordan, M., editor, *Learning in Graphical Models*, pages 105–161. The MIT Press, Cambridge, MA, USA.

Julier, S. and Uhlmann, J. K. (1996). A general method for approximating nonlinear transformations of probability distributions. Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford.

Jutten, C., Babaie-Zadeh, M., and Hosseini, S. (2004). Three easy ways for separating nonlinear mixtures? *Signal Processing*, 84(2):217–229.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24(1):1–10.

Jutten, C. and Karhunen, J. (2004). Advances in blind source separation (BSS) and independent component analysis (ICA) for nonlinear mixtures. *International Journal of Neural Systems*, 14(5):267–292.

Jutten, C. and Taleb, A. (2000). Source separation: From dusk till dawn. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 15–26, Helsinki, Finland.

Kagan, A. M., Linnik, Y. V., and Rao, C. R. (1973). *Characterization Problems in Mathematical Statistics*. J. Wiley.

Kappen, H. J. and Rodríguez, F. B. (1998). Efficient learning in boltzmann machines using linear response theory. *Neural Computation*, 10(5):1137–1156.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Lappalainen, H. (1998). Using an MDL-based cost function with neural networks. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'98)*, pages 2384–2389, Anchorage, Alaska, USA.

Lappalainen, H. (1999). Ensemble learning for independent component analysis. In *Proceedings of the International Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 7–12, Aussois, France.

Lappalainen, H. and Giannakopoulos, X. (1999). Multi-layer perceptrons as nonlinear generative models for unsupervised learning: A Bayesian treatment. In *Proceedings of the Ninth International Conference on Artificial Neural Networks (ICANN'99)*, pages 19–24, Edinburgh, UK.

Lappalainen, H. and Miskin, J. (2000). Ensemble learning. In Girolami, M., editor, *Advances in Independent Component Analysis*, pages 75–92. Springer-Verlag, Berlin.

MacKay, D. J. C. (1992). Bayesian interpolation. *Neural Computation*, 4(3):415–447.

MacKay, D. J. C. (1995). Developments in probabilistic modelling with neural networks – ensemble learning. In *Neural Networks: Artificial Intelligence and Industrial Applications. Proceedings of the 3rd Annual Symposium on Neural Networks*, pages 191–198.

MacKay, D. J. C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Makeig, S., Bell, A., Jung, T.-P., and Sejnowski, T. (1996). Independent component analysis of electroencephalographic data. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Advances in Neural Information Processing Systems 8*. MIT Press, Cambridge, MA, USA.

Matsuoka, K., Ohya, M., and Kawamoto, M. (1995). A neural net for blind separation of nonstationary signals. *Neural Networks*, 8(3):411–419.

Maybeck, P. S. (1979). *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, New York.

Maybeck, P. S. (1982). *Stochastic Models, Estimation, and Control*, volume 2. Academic Press, New York.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092.

Mezard, M., Parisi, G., and Virasoro, M. A. (1987). *Spin Glass Theory and Beyond*. World Scientific.

Minka, T. P. (2001). Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI 2001)*, pages 362–369, Seattle, Washington, USA.

Miskin, J. and MacKay, D. J. C. (2001). Ensemble learning for blind source separation. In Roberts, S. and Everson, R., editors, *Independent Component Analysis: Principles and Practice*, pages 209–233. Cambridge University Press.

Müller, K.-R., Philips, P., and Ziehe, A. (1999). $JADE_{TD}$: Combining higher-order statistics and temporal information for blind source separation (with noise). In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA'99)*, pages 87–92, Aussois, France.

Murphy, K., Weiss, Y., and Jordan, M. (1999). Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence (UAI 1999)*, pages 467–475, Stockholm, Sweden.

Murray, M. K. and Rice, J. W. (1993). *Differential Geometry and Statistics*. Chapman & Hall.

Neal, R. M. and Hinton, G. E. (1999). A view of the EM algorithm that justifies incremental, sparse, and other variants. In Jordan, M. I., editor, *Learning in Graphical Models*, pages 355–368. The MIT Press, Cambridge, MA, USA.

Opper, M. and Winther, O. (2001). From naive mean field theory to the TAP equations. In Opper, M. and Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, pages 7–20. The MIT Press, Cambridge, MA, USA.

Parisi, G. (1988). *Statistical Field Theory*. Addison-Wesley.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pham, D.-T. and Cardoso, J.-F. (2001). Blind separation of instantaneous mixtures of nonstationary sources. *IEEE Transactions on Signal Processing*, 49(9):1837–1848.

Raiko, T. (2001). Hierarchical nonlinear factor analysis. Master's thesis, Helsinki University of Technology, Espoo.

Raiko, T. (2004). Partially observed values. In *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IJCNN 2004)*, pages 2825–2830, Budapest, Hungary.

Raiko, T. and Tornio, M. (2005). Learning nonlinear state-space models for control. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'05)*. To appear.

Raiko, T. and Valpola, H. (2001). Missing values in nonlinear factor analysis. In *Proceedings of the 8th International Conference on Neural Information Processing (ICONIP'01)*, pages 822–827, Shanghai.

Raiko, T., Valpola, H., Östman, T., and Karhunen, J. (2003). Missing values in hierarchical nonlinear factor analysis. In *Proceedings of the International Conference on Artificial Neural Networks and Neural Information Processing - ICANN/ICONIP 2003*, pages 185–189, Istanbul, Turkey.

Rao, C. R. (1969). A decomposition theorem for vector variables with a linear structure. *The Annals of Mathematical Statistics*, 40(5):1845–1849.

Rissanen, J. (1989). *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore.

Roweis, S. and Ghahramani, Z. (1999). A unifying review of linear Gaussian models. *Neural Computation*, 11(2):305–345.

Rudin, W. (1987). *Real and Complex Analysis*. McGraw-Hill, third edition.

Rustagi, J. S. (1976). *Variational Methods in Statistics*. Academic Press, New York.

Särelä, J., Valpola, H., Vigário, R., and Oja, E. (2001). Dynamical factor analysis of rhythmic magnetoencephalographic activity. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 451–456, San Diego, USA.

Särelä, J. and Vigário, R. (2003). Overlearning problem in high-order ICA: analysis and solutions. *Journal of Machine Learning Research*, 4:1447–1469.

Sato, M. (2001). Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681.

Schölkopf, B. and Smola, A. (2002). *Learning with Kernels*. The MIT Press, Cambridge, MA, USA.

Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423 and 623–656.

Spearman, C. (1904). "General intelligence," objectively determined and measured. *American Journal of Psychology*, 15:201–293.

Taleb, A. (2002). A generic framework for blind source separation in structured nonlinear models. *IEEE Transactions on Signal Processing*, 50(8):1819–1830.

Taleb, A. and Jutten, C. (1999). Source separation in post-nonlinear mixtures. *IEEE Transactions on Signal Processing*, 47(10):2807–2820.

Tanaka, T. (1996). Information geometry of mean field theory. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E79-A(5):709–715.

Tanaka, T. (2000). Information geometry of mean-field approximation. *Neural Computation*, 12(8):1951–1968.

Tanaka, T. (2001). Information geometry of mean-field approximation. In Opper, M. and Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, pages 259–273. The MIT Press, Cambridge, MA, USA.

Theis, F. J. (2004). A new concept for separability problems in blind source separation. *Neural Computation*, 16(9):1827–1850.

Theis, F. J., Bauer, C., and Lang, E. W. (2002). Comparison of maximum entropy and minimal mutual information in a nonlinear setting. *Signal Processing*, 82(7):971–980.

Theis, F. J. and Gruber, P. (2005). On model identifiability in analytic postnonlinear ICA. *Neurocomputing*, 64:223–234.

Valpola, H. (2000). Nonlinear independent component analysis using ensemble learning: theory. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 251–256, Helsinki, Finland.

Valpola, H., Giannakopoulos, X., Honkela, A., and Karhunen, J. (2000). Nonlinear independent component analysis using ensemble learning: experiments and discussion. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 351–356, Helsinki, Finland.

Valpola, H., Harva, M., and Karhunen, J. (2004). Hierarchical models of variance sources. *Signal Processing*, 84(2):267–282.

Valpola, H., Honkela, A., and Giannakopoulos, X. (2002a). Matlab codes for the NFA and NSSM algorithms. *http://www.cis.hut.fi/projects/bayes/software/*.

Valpola, H., Honkela, A., Harva, M., Ilin, A., Raiko, T., and Östman, T. (2003a). Bayes blocks software library. *http://www.cis.hut.fi/projects/bayes/software/*.

Valpola, H., Honkela, A., and Karhunen, J. (2002b). An ensemble learning approach to nonlinear dynamic blind source separation using state-space models. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'02)*, pages 460–465, Honolulu, Hawaii, USA.

Valpola, H. and Karhunen, J. (2002). An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Computation*, 14(11):2647–2692.

Valpola, H., Oja, E., Ilin, A., Honkela, A., and Karhunen, J. (2003b). Nonlinear blind source separation by variational Bayesian learning. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E86-A(3):532–541.

Valpola, H., Östman, T., and Karhunen, J. (2003c). Nonlinear independent factor analysis by hierarchical models. In *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 257–262, Nara, Japan.

Valpola, H., Raiko, T., and Karhunen, J. (2001). Building blocks for hierarchical latent variable models. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 710–715, San Diego, USA.

Vigário, R., Särelä, J., Jousmäki, V., Hämäläinen, M., and Oja, E. (2000). Independent component approach to the analysis of EEG and MEG recordings. *IEEE Transactions on Biomedical Engineering*, 47(5):589–593.

Wall, M. M. and Amemiya, Y. (2004). A review of nonlinear factor analysis statistical methods. Research Report RC23392, IBM. To appear in book Factor Analysis at 100: Historical Developments and Future Directions.

Wallace, C. S. (1990). Classification by minimum-message-length inference. In Aki, S. G., Fiala, F., and Koczkodaj, W. W., editors, *Advances in Computing and Information – ICCI '90*, volume 468 of *Lecture Notes in Computer Science*, pages 72–81. Springer, Berlin.

Wallace, C. S. and Boulton, D. M. (1968). An information measure for classification. *The Computer Journal*, 11(2):185–194.

Wan, E. A. and van der Merwe, R. (2001). The unscented Kalman filter. In Haykin, S., editor, *Kalman Filtering and Neural Networks*, pages 221–280. J. Wiley, New York.

Wang, B. and Titterington, D. M. (2004). Lack of consistency of mean field and variational Bayes approximations for state space models. *Neural Processing Letters*, 20(3):151–170.

Werbos, P. (1992). Approximate dynamic programming for real-time control and neural modeling. In White, D. A. and Sofge, D. A., editors, *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*, chapter 13, pages 493–525. Van Nostrand Reinhold, New York.

Williams, D. (1991). *Probability with Martingales*. Cambridge University Press.

Winn, J. and Bishop, C. M. (2005). Variational message passing. *Journal of Machine Learning Research*, 6:661–694.

Wiskott, L. and Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770.

Xu, L. (2003). Independent component analysis and extensions with noise and time: A Bayesian Ying-Yang learning perspective. *Neural Information Processing – Letters and Reviews*, 1(1):1–52.

Yalcin, I. and Amemiya, Y. (2001). Nonlinear factor analysis as a statistical method. *Statistical Science*, 16(3):275–294.

Yedidia, J. S., Freeman, W. T., and Weiss, Y. (2001). Generalized belief propagation. In Leen, T., Dietterich, T., and Tresp, V., editors, *Advances in Neural Information Processing Systems 13*, pages 689–695. The MIT Press, Cambridge, MA, USA.

Ziehe, A. and Müller, K.-R. (1998). TDSEP — an effective algorithm for blind separation using time structure. In *Proceedings of the 8th International Conference on Artificial Neural Networks (ICANN'98)*, pages 675–680, Skövde, Sweden.