Variational Mixture of Bayesian Independent Component Analysers

R.A. Choudrey and S.J. Roberts
University of Oxford
Robotics Research
Oxford, U.K.

June 15, 2002

Abstract

There has been growing interest in subspace data modelling over the past few years. Methods such as Principal Component Analysis, Factor Analysis and Independent Component Analysis have gained in popularity and have found many applications in image modelling, signal processing and data compression to name just a few. As applications and computing power grow, more and more sophisticated analyses and meaningful representations are sought. Mixture modelling methods have been proposed for principal and factor analysers which exploit local Gaussian features in the subspace manifolds. Meaningful representations may be lost, however, if these local features are non-Gaussian and/or discontinuous. In this paper we propose extending the Gaussian analysers mixture model to an Independent Component Analysers mixture model. We employ recent developments in variational Bayesian inference and structure determination to construct a novel approach for modelling non-Gaussian, discontinuous manifolds. We automaticaly determine the local dimensionality of each manifold and use variational inference to calculate the optimum number of ICA components needed in our mixture model. We

demonstrate our framework on complex synthetic data and illustrate its application to real data by decomposing functional Magnetic Resonance Images into meaningful - and medically useful - features.

1 Introduction

The goal of pattern analysis and recognition is to extract information from some data. In order for this information to be useful, the distribution of data must be represented in some meaningful way. In many cases, insight may be gained by dividing the data into self-similar areas and analysing each of these clusters under some informative framework, for example using some understanding of the assumed data generating process. One such method is to model the data as being produced by a mixture of data generators (also called analysers), where each component generator is responsible for generating a particular cluster. The problems to overcome in this *mixture modelling* are to decide how many generators are needed, where to place them, and how to adjust them to best represent the data.

Mixtures of Gaussians (MoG) are widely used throughout the fields of machine learning and statistics for data modelling, where each generator is a Gaussian density. Despite their popularity, however, MoGs suffer from two serious drawbacks. The first is that, as the dimensionality S of the problem space increases, the size of each covariance matrix, S^2, becomes prohibitively large. This can be dealt with by assuming isotropic Gaussians (i.e. ignoring the covariance structure) but this greatly reduces the flexibility of the model class. This problem has been solved by Tipping and Bishop (1999) who replaced each Gaussian with a probabilistic Principal Component Analyser (PCA). This allowed the dimensionality of each covariance to be effectively reduced whilst maintaining the richness of the model class. This was formulated under a maximum-likelihood framework which - although efficient - doesn't allow one to infer the optimum number of generators needed. This mixture was modified

into a Mixture of Factor Analysers (FA) by Ghahramani and Beal (2000) where Bayesian inference was used to infer the optimum number of analysers.

The second problem with MoGs is that each component is a Gaussian, a strong assumption which is often violated in many natural clustering problems (Lee, Lewicki, & Sejnowski, 2000). Although MoGs are capable of modelling any distribution given enough components, the problem still remains of automatically grouping Gaussians which together describe some larger-scale feature. It is this second problem which we address in this paper. A solution is reached by extending the mixtures of probabilistic PCA/FA model to an Independent Component Analysis (ICA) mixture model. We improve on previous work on mixtures of ICA by Lee et al. (2000) and Penny and Roberts (2001) by incorporating a very flexible ICA model that can generate arbitrary densities using MoGs, and by bringing the formalism into the Bayesian arena. We use Bayesian inference to infer the optimum number of ICAs needed, and to automatically determine their ideal dimensionalities.

1.1 ICA

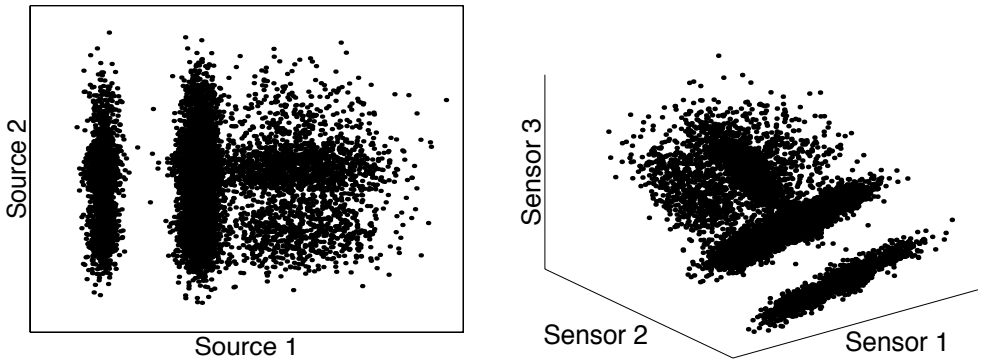
Independent Component Analysis seeks to extract salient features and structure from data which is assumed to be a linear mixture of independent underlying (hidden) features

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s} \tag{1}$$

where $\boldsymbol{x}$ is an observed S-dimensional data vector, $\boldsymbol{s}$ is the L-dimensional *source* vector and $\boldsymbol{A}$ is the SxL *bases* or *mixing*[1] matrix. An ICA model formulated in this way is only well posed if $L \leq S$ otherwise there are effectively more unknowns than equations in the model. The overcomplete problem ($L \geq S$) is an area of ongoing research and will not be dealt with here.

Which of $\boldsymbol{A}$ or $\boldsymbol{s}$ represent the features depends on one's view of the data. If the data are considered to be constructed from varying amounts of L static

[1]Please note that the term *mixing* refers to the linear mixing in the ICA model and *mixture* refers to stochastic generating method in a mixture model



(a) Original source density (b) Observed sensor density

Figure 1: The underlying and observed data densities

features then the columns of $\boldsymbol{A}$ represent the L static basis vectors and $\boldsymbol{s}$ represents the amount of each basis used for a given data-vector. For example, if the data represent an image, that image may be considered a mixture of underlying (maybe fewer) independent edges, textures etc.. The aim of ICA in this case is to 'unmix' the dataset and recover these representative features. If the data are time-varying, the data can be seen as a linear mixing of underlying (possibly lower-dimensional) signals represented by $\boldsymbol{s}$. For example, the data may be signals received at S microphones at a cocktail party. These signals are then simply mixings of L people talking. ICA is used to 'blindly' (i.e. without access to $\boldsymbol{s}$ and $\boldsymbol{A}$) separate the sensor signals into the underlying source signals $\boldsymbol{s}$. This is known as Blind Source Separation (BSS) in the signal processing literature. In either case, ICA can be recast as data density modelling. The data density $p(\boldsymbol{x})$ is a linear transform (i.e. scaling, rotation and possibly higher-dimensional projection) of an unknown, non-Gaussian manifold - the source density $p(\boldsymbol{s})$ (see Figure 1). The goal of ICA is to recover the source density and estimate the transform matrix.

ICA has traditionally been performed in the noise-less limit (Bell & Sejnowski, 1995; Lee, Girolami, Bell, & Sejnowski, 1998), with limited flexibility

in the source density model and with noise often being dealt with as an extra source. More recently, however, Attias (1999b) extended ICA by utilising a more flexible source model and incorporating full covariance noise explicitely into the ICA framework. The model - dubbed by Attias as Independent Factor Analysis (IFA) - was subsequently learnt through a maximum likelihood EM algorithm. Bayesian formalisms have also been introduced, most notably (under a variational framework) by Attias (1999a), Lappalainen (1999), Choudrey, Penny, and Roberts (2000) and Miskin and MacKay (2000). For a comprehensive review of ICA, see Hyvärinen (1999) or Roberts and Everson (2001).

1.2 ICA Mixture Model

ICA was originally developed as a solution to the BSS problem in auditory signal analysis by Herault and Jutten (1986). Since then, the BSS approach has been extended to human electroencephalograms (EEGs) (Makeig, Bell, Jung, & Sejnowski, 1996), financial data analysis (Back & Weigend, 1998), and telecommunications (Ristaniemi & Joutsensalo, 1999), amongst many other applications. Recently, however, ICA's remit has broadened to one of *representation* in general.

The way data is presented very much influences what patterns are found and how much information can be extracted from it. A primary goal in pattern recognition, then, is to find some intrinsic coordinate system from which structure is easier to extract. In this sense, ICA is a higher-order extension of 2nd-order methods such as PCA and FA. PCA/FA find representations of data that preserve the maximum amount of variance by effectively finding an intrinsic coordinate system for the covariance matrix. Projections of the data on to the bases found leads to decorrelation. ICA seeks a loftier goal: it seeks a representation in which the data are maximimally statistically independent. In this way, ICA aims to 'repackage' data, which may contain structure *across* its constituent dimensions, into independent signals which contain structure only *within* components, thereby by making patterns (hopefully) more cogent.

This application of ICA has led to interesting work in representation of text corpora (Isbell & Viola, 1999; Kolenda, Hansen, & Sigurdsson, 2000), of images, such as face representation (Bartlett, Lades, & Sejnowski, 1998) and of natural scenes (Bell & Sejnowski, 1997). This in turn has hinted at possible links between ICA and coding schemes in the primary visual cortex (Olshausen & Field, 1997; Lewicki & Olshausen, 1999), although we stress that this connection is a contentious one (Hateren & Schaaf, 1998).

It is this representational viewpoint that motivates our extension of ICA into a mixture of ICAs. Decomposing and representing data using ICA assumes the whole data distribution is adequetly described by one coordinate frame. This may not be appropriate for many problems, however. In the BSS problem, ICA assumes each data signal carries a constant mixture of source signals. Consider the scenario proposed by Lee, Lewicki, and Sejnowski (1999). There are two people talking to each other, but never at the same time, while there is music in the background. This cacophany is picked up by two microphones. At any one time, the microphones pick up a mixture of one voice and the music, or the other voice and the music, but never all three at the same time. Clearly in this case, standard ICA is an inadequate model. As shown by Lee et al. (1999), a mixture of ICAs is a more appropriate generative model. More generally, if the sensor density in Figure 1 consists of various self-similar, non-Gaussian manifolds, enforcing a single, global representation is not appropriate and will produce a sub-optimal representation. A more intuitive way of representing the data is the use of a number of local coordinate frames, each constructed under local conditions. This leads to a mixture of Independent Component Analysers.

ICA mixture models were first formulated by Lee et al. (1999) using the extended Infomax algorithm (Lee, Girolami, & Sejnowski, 1999) to switch the source model between sub-Gaussian and super-Gaussian regimes. The model was learnt through a combination of maximum likelihood and gradient ascent. Although well demonstrated, the source model could only switch

between Laplacian and bimodal densities and thus lacked flexibility. Penny and Roberts (2001) relaxed this constraint by utilising generalised exponential sources which can model a wide variety of kurtoses by the adjustment of a parameter. A basic method of model selection was also incorporated using the Bayesian Information Criterion (BIC) to infer the number of hidden sources (i.e. the intrinsic dimensionality of the local manifold). Due to the problem formulation used, only 2-dimensional or higher manifolds could be modelled. Although more flexible, the densities could only be unimodal and the learning scheme was also maximum likelihood.

In this paper, we present a ICA mixture model trained using Bayesian methods. In line with Attias (1999b), Lappalainen (1999), Choudrey et al. (2000) and Miskin and MacKay (2000), we choose a fully-adaptable factorial Mixture of Gaussians as the source model for our ICA components allowing us to recover arbitrary source densities. Essentialy, each ICA component will model self-similar areas (the features) as a mixture of Gaussian sub-features. To overcome the heavy computational load associated with Bayesian learning, we use the variational framework reviewed by Jaakkola and Jordan (2000) to make assumptions about the posterior, giving tractability to the Bayesian model. The variational framework is a whole class of methods for bounding intractable integrals. In particular, we use 'free-energy minimisation' (Hinton & Camp, 1993) or 'ensemble learning' (MacKay, 1995a), a variational method for Bayesian parameter estimation. We further extend the ensemble learning results of Choudrey et al. (2000) and Miskin and MacKay (2000) by utilising a MoG posterior density in a similar fashion to Attias (1999b) - rather than a single Gaussian - letting us capture much richer posteriors. This, in turn, leads to greater robustness under uncertainty (Choudrey & Roberts, 2001). The variational Bayesian method is carried through to the ICA mixture model, allowing model comparison, incorporation of prior knowledge of the mixture process and - by integrating out nuisance parameters - control of model complexity and, thus, over-fitting. By monitoring the *variational free energy* of

our models, we can compare model assumptions allowing us, for example, to infer the optimum number of components needed in our ICA mixture model (Attias, 1999a). We also employ Automatic Relevance Determination (ARD) (MacKay, 1995b) to suppress unsupported sources and thus effectively infer the number of latent dimensions of each ICA component as part of the learning process. This leads to the variational Bayesian Independent Component Analysers mixture model (vbICA-MM).

2 The Model

The probability of generating a data vector $\boldsymbol{x}^n$ from a C-component mixture model given assumptions $\mathcal{M}$ is:

$$p(\boldsymbol{x}^n|\mathcal{M}) = \sum_{c=1}^{C} p(c|\mathcal{M}_0)p(\boldsymbol{x}^n|\mathcal{M}_c, c) \tag{2}$$

A data vector is generated by choosing one of the C components stochastically under $p(c|\mathcal{M}_0)$ and then drawing from $p(\boldsymbol{x}^n|\mathcal{M}_c, c)$. $\mathcal{M} = \{\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_C\}$ is the vector of component model assumptions, $\mathcal{M}_c$, and assumptions about the mixture process, $\mathcal{M}_0$. The assumptions represent everything that essentially defines the model - values of fixed parameters, model structure, details of the component switching method, any prior information etc.. $p(\boldsymbol{x}^n|\mathcal{M})$ is known as the evidence for model $\mathcal{M}$ and quantifies the likelihood of the observed data under model $\mathcal{M}$.

The variable c indicates which component of the mixture model is chosen to generate a given data vector $\boldsymbol{x}$. If $p(c|\mathcal{M}_0)$ is a vector of probabilities and each component $p(\boldsymbol{x}^n|\mathcal{M}_c, c)$ is a Gaussian, then (2) simply describes a MoG. If the MoG is adapted through a maximum likelihood approach then $\mathcal{M}$ represents a list of point estimates for the corresponding parameters. In our mixture model, however, each component has a non-Gaussian density derived from the ICA model presented in the next section, and $\mathcal{M}$ represents assumptions

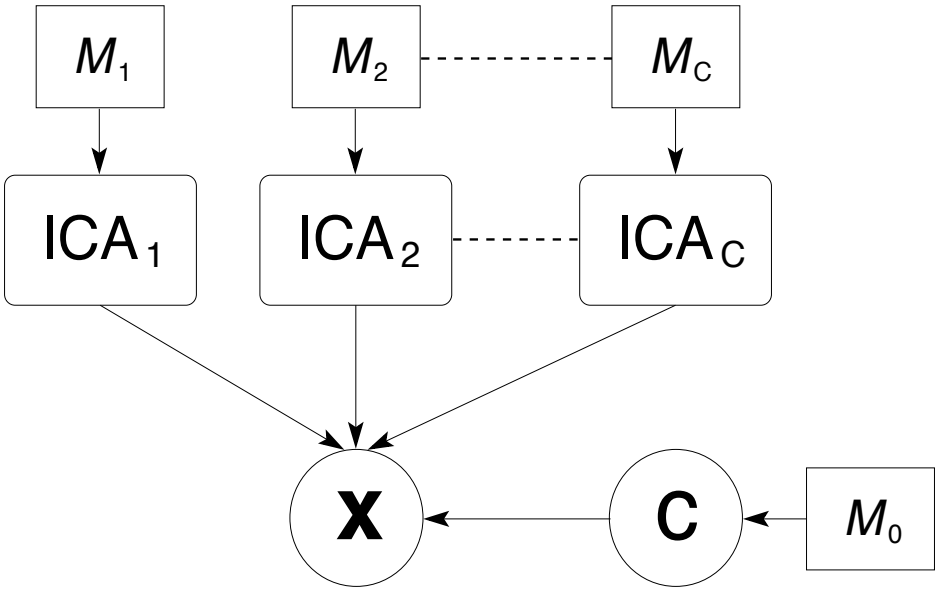


Figure 2: ICA mixture model

concerning the *distribution* of possible parameter values. Figure 2 shows a generative model for a data vector $\boldsymbol{x}$.

2.1 The ICA Model

In common with ICA in the literature, we choose a generative model to work with. The observed variables, $\boldsymbol{x}$, of dimension S are modelled as a linear combination of statistically independent latent variables, $\boldsymbol{s}_c$, of dimension L_c with added Gaussian noise

$$\boldsymbol{x} = \boldsymbol{A}_c \boldsymbol{s}_c + \boldsymbol{y}_c + \boldsymbol{e}_c \tag{3}$$

where $\boldsymbol{y}_c$ is an S-dimensional *bias* vector, $\boldsymbol{e}_c$ is S-dimensional additive noise and c represents the c^{th} ICA model. In signal processing nomenclature, S is the number of (observed) sensors and L_c is the number of latent (hidden) sources.

Equation (3) acts as a complete description for cluster c in the data density. The bias vector, $\boldsymbol{y}_c$, defines the position of the cluster in the S-dimensional data

space, $\boldsymbol{A}_c$ describes its orientation and $\boldsymbol{s}_c$ describes the underlying manifold. Using ICA, data produced by mixing Gaussian sources cannot be separated as they only have 2-moment distributions, so can never be more than decorrelated. The noise, $\boldsymbol{e}_c$, is assumed to be zero-mean Gaussian and isotropic

$$p(\boldsymbol{e}_c|0, \Lambda_c, c) = \mathcal{N}(\boldsymbol{e}_c; 0, \lambda_c I) \tag{4}$$

where $\lambda_c I$ is the precision [2]. The noise essentially absorbs any (spherical) Gaussianity present in the cluster.

The probability of observing data vector $\boldsymbol{x}^n$ under component c is then given by

$$p(\boldsymbol{x}^n|\theta_c, c) = \left(\frac{\lambda_c}{2\pi}\right)^{\frac{S}{2}} \exp[-E_c] \tag{5}$$

where $\theta_c = \{\boldsymbol{A}_c, \boldsymbol{s}_c^n, \boldsymbol{\lambda}_c\}$ and

$$E_c = \frac{\lambda_c}{2}(\boldsymbol{x}^n - \boldsymbol{A}_c\boldsymbol{s}_c^n - \boldsymbol{y}_c)^T(\boldsymbol{x}^n - \boldsymbol{A}_c\boldsymbol{s}_c^n - \boldsymbol{y}_c) \tag{6}$$

Since the sources $\boldsymbol{s}_c = \{s_{c,1}, .., s_{c,i}, .., s_{c,L_c}\}$ are - by definition - mutually independent, the distribution over $\boldsymbol{s}_c$ for data point n can be written as

$$p(\boldsymbol{s}_c^n|\mathcal{M}_{\boldsymbol{s}_c}, c) = \prod_{i=1}^{L_c} p(s_{c,i}^n|\mathcal{M}_{s_{c,i}}, c) \tag{7}$$

where the product runs over the L_c sources of component c and $\mathcal{M}_{\boldsymbol{s}_c}$ is the vector of source model assumptions.

$p(\boldsymbol{s}_c^n|\mathcal{M}_{\boldsymbol{s}_c}, c)$ is the source model for ICA component c. Traditionally a reciprocal-cosh (Bell & Sejnowski, 1995) or unimodal density (Miskin & MacKay, 2000) has been used. These, however, have limited flexibility either in kurtosis-representation or capturing multiple modes. Methods have been proposed for switching between super- and sub-Gaussian regimes (Girolami,

[2] Conventional notation specifies covariance rather than precision. We have found precisions easier to manipulate in the maths, so have abused the notation somewhat to make subsequent equations easier to read

1998; Lee et al., 1999, 2000), although these depend on heuristic stability analyses. More recently, Attias (1999b) introduced a Mixture of Gaussians (MoG) source model into his ICA formalism, potentially allowing any distribution to be modelled. This is the route we take.

2.2 ICA Source Model

The choice of a flexible and mathematically attractive (tractable) source model is crucial if a wide variety of source distributions are to be modelled; in particular, the source model should be capable of encompassing both super- and sub-Gaussian distributions (distributions with positive and negative kurtosis respectively) and complex multi-modal distributions.

One such distribution is a factorised mixture of 1-dimensional Gaussians with L_c factors (i.e. sources) and m_i components per source (see Figure 3)

$$\begin{aligned} p(\boldsymbol{s}_c^n|\boldsymbol{\varphi}_c, c) &= \prod_{i=1}^{L_c} \sum_{q_i=1}^{m_i} p(q_i^n = q_i|\boldsymbol{\pi}_i, c) p(s_{c,i}^n|\varphi_{c,i}, c) \\ &= \prod_{i=1}^{L_c} \sum_{q_i=1}^{m_i} \pi_{i,q_i} \mathcal{N}(s_{c,i}^n; \mu_{i,q_i}, \beta_{i,q_i}) \end{aligned} \tag{8}$$

where, for brevity, the ICA component subscript c has been dropped from parameters which can be seen to belong to ICA c from context. From now on, all subscripted parameters should be assumed to belong to the c^{th} ICA model, unless otherwise stated.

Equation (8) essentially describes the local features of cluster c - μ_{i,q_i} is the position of feature q_i w.r.t. the cluster centre, β_{i,q_i} is its size, and π_{i,q_i} its 'prominance' w.r.t. other features.

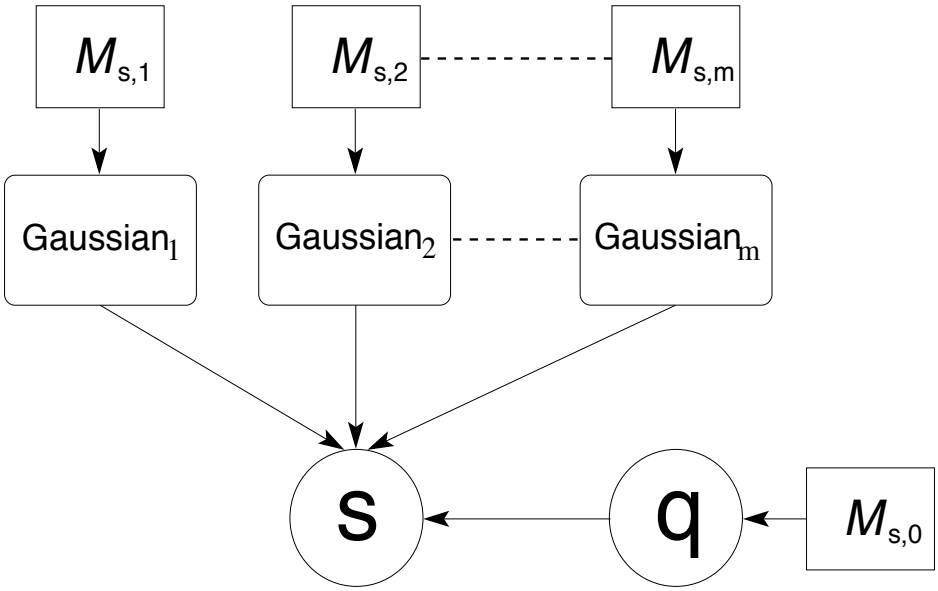


Figure 3: MoG source model

The mixture proportions $\pi_{i,q_i} = p(q_i^n = q_i|\boldsymbol{\pi}_i)$ are the prior probabilities of choosing component q_i of the i^{th} source (of the c^{th} ICA model etc.). q_i^n is a variable indicating which component of the i^{th} source is chosen for generating $s_{c,i}^n$ and takes on values of $\{q_i = 1, \ldots, q_i = m_i\}$ (where m_i, of course, depends on ICA model c). The mean and precision of Gaussian q_i in source i are μ_{i,q_i} and β_{i,q_i} respectively. The parameters of source i are $\varphi_{c,i} = \{\boldsymbol{\pi}_{c,i}, \boldsymbol{\mu}_{c,i}, \boldsymbol{\beta}_{c,i}\}$ where bold face indicates the vector of m_i parameters. The complete parameter set of the source model is $\boldsymbol{\varphi}_c = \{\varphi_{c,1}, \varphi_{c,2}, \ldots, \varphi_{c,L_c}\}$.

The complete collection of possible source states is denoted $\boldsymbol{q}_c = \{\boldsymbol{q}_{c,1}, \boldsymbol{q}_{c,2}, \ldots, \boldsymbol{q}_{c,\boldsymbol{m}}\}$ and runs over all $\boldsymbol{m} = \prod_i m_i$ possible combinations of source states. The probability of state $\boldsymbol{q}_c^n$ being chosen and generating source vector $\boldsymbol{s}_c^n$ is

$$\begin{aligned} p(\boldsymbol{s}_c^n, \boldsymbol{q}_c^n|\boldsymbol{\varphi}_c, c) &= \prod_{i=1}^{L_c} p(q_i^n = q_i|\boldsymbol{\pi}_i, c) p(s_{c,i}^n|q_i, \mu_{i,q_i}, \beta_{i,q_i}, c) \\ &= p(\boldsymbol{q}_c^n|\boldsymbol{\pi}_c, c) p(\boldsymbol{s}_c^n|\boldsymbol{q}_c^n, \boldsymbol{\varphi}_c, c) \end{aligned} \tag{9}$$

where $\boldsymbol{\pi}_c = \{\boldsymbol{\pi}_{c,1}, \boldsymbol{\pi}_{c,2}, \ldots, \boldsymbol{\pi}_{c,L_c}\}$ (similarly for $\boldsymbol{\mu}$ and $\boldsymbol{\beta}$). Note that the

product of L_c 1-dimensional MoGs in (8) is equivalent to a single MoG in L_c-dimensional space with $\boldsymbol{m}$ states.

By integrating and summing over the hidden variables, $\{\boldsymbol{s}_c, \boldsymbol{q}_c\}$, the likelihood of the IID data $\boldsymbol{X} = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^N\}$ given the model parameters $\Theta_c = \{\boldsymbol{A}_c, \boldsymbol{y}_c, \lambda_c, \boldsymbol{\varphi}_c\}$ can now be written as

$$p(\boldsymbol{X}|\Theta_c, c) = \prod_{n=1}^{N} \sum_{\boldsymbol{q}=1}^{\boldsymbol{m}} \int p(\boldsymbol{x}^n, \boldsymbol{s}_c^n, \boldsymbol{q}_c^n|\Theta_c, c) d\boldsymbol{s}_c \tag{10}$$

where $d\boldsymbol{s}_c = \prod_i ds_{c,i}$.

If we stipulate a form for $p(c|\mathcal{M}_0)$ in (2)

$$p(\boldsymbol{X}|\mathcal{M}) = \sum_{c=1}^{C} p(c|\boldsymbol{\kappa}) p(\boldsymbol{X}|\Theta_c, c) \tag{11}$$

where $p(c|\boldsymbol{\kappa}) = \{p(c=1) = \kappa_1, p(c=2) = \kappa_2, \ldots, p(c=C) = \kappa_C\}$, then (5) - (7) and (8) can be substituted into (11) to yield a maximum likelihood model. This can then be learnt through an iterative process such as gradient descent (Lee et al., 2000) or the Expectation-Maximisation algorithm (Penny & Roberts, 2001). In this paper, however, we take the Bayesian route by also integrating out the parameters $\{\boldsymbol{\kappa}, \Theta_c\}$ in (11).

3 Bayesian Inference and Variational Learning

The maximum Likelihood approach to learning the parameters of a model is well documented (see work by Pearlmutter and Parra (1996), Cardoso (1997), and Attias (1999b) for an introduction), as are the pitfalls (e.g. over-fitting, no quantification of uncertainty in the model or model comparison). We choose to take the Bayesian approach and integrate out the parameters $\{\boldsymbol{\kappa}, \Theta_c\}$ and hidden variables $\{\boldsymbol{s}_c, \boldsymbol{q}_c\}$. This allows us to take our assumptions 'one-stage higher' i.e. rather than assume point-estimates for the parameters (very harsh and specific), we assume prior distributions over all possible parameter values. This is much more flexibile as it allows us to tailor the distributions to 1)

avoid over-fitting the data, 2) reduce the parameter search space to 'reasonable' values, and 3) code our inevitable uncertainty in the parameters into the model.

First, we will state the prior distributions over the hidden variables and model parameters.

3.1 The Priors

The priors are chosen to be appropriately conjugate to allow tractibility. All bold face parameters without sub- or super-scripts indicate the collection of C parameter vectors.

The prior over the ICA mixture indicator variables, $\boldsymbol{c} = \{c^1, c^2, \ldots, c^N\}$, simply factorises over the N data vectors

$$p(\boldsymbol{c}|\boldsymbol{\kappa}) = \prod_{n=1}^{N} \kappa_{c^n} \tag{12}$$

The prior over the ICA mixture coefficients κ is a Dirichlet

$$p(\boldsymbol{\kappa}) = \prod_{c=1}^{C} \frac{\Gamma(\sum_{c'} \iota_{c'})}{\Gamma(\iota_c)} \kappa_c^{\iota_c - 1} \tag{13}$$

Because of source independence, it follows that the distribution over the MoG component indicator variables, $p(\boldsymbol{q}_c|\boldsymbol{\pi}_c, c)$, is a product over all π_{i,q_i^n} where i indexes the sources and n the data

$$p(\boldsymbol{q}_c|\boldsymbol{\pi}_c, c) = \prod_{n=1}^{N} \prod_{i=1}^{L_c} \pi_{i,q_i^n} \tag{14}$$

For a given state $\boldsymbol{q}$, the distribution over the sources is

$$p(\boldsymbol{s}_c|\boldsymbol{q}_c, \boldsymbol{\varphi}_c, c) = \prod_{n=1}^{N} \prod_{i=1}^{L_c} \mathcal{N}(s_{c,i}^n; \mu_{i,q_i}, \beta_{i,q_i}) \tag{15}$$

The prior over the source model (MoG) parameters is a product of priors over $\boldsymbol{\pi}_c, \boldsymbol{\mu}_c, \boldsymbol{\beta}_c$.

$$p(\boldsymbol{\varphi}) = \prod_{c=1}^{C} p(\boldsymbol{\pi}_c) p(\boldsymbol{\mu}_c) p(\boldsymbol{\beta}_c) \tag{16}$$

The prior over the mixture proportions, $\boldsymbol{\pi}_c$, for the i^{th} source is a Dirichlet with parameter ρ_{i,q_i}.

$$p(\boldsymbol{\pi}) = \prod_{c=1}^{C} \prod_{i=1}^{L_c} \prod_{q_i=1}^{m_i} \frac{\Gamma(\sum_{q_i'} \rho_{i,q_i'})}{\Gamma(\rho_{i,q_i})} \pi_{i,q_i}^{\rho_{i,q_i}-1} \tag{17}$$

The prior over each MoG mean, μ_{i,q_i}, is a Gaussian with mean m_{i0} and precision τ_{i0} and the prior over the associated precision, β_{i,q_i}, is a Gamma with width and scale parameters b_{i0} and c_{i0} respectively.

$$p(\boldsymbol{\mu}) = \prod_{c=1}^{C} \prod_{i=1}^{L_c} \prod_{q_i=1}^{m_i} \mathcal{N}(\mu_{i,q_i}; m_{i0}, \tau_{i0}) \tag{18}$$

$$p(\boldsymbol{\beta}) = \prod_{c=1}^{C} \prod_{i=1}^{L_c} \prod_{q_i=1}^{m_i} \mathcal{G}(\beta_{i,q_i}; b_{i0}, c_{i0}) \tag{19}$$

where we define the Gamma distribution as

$$\mathcal{G}(w; b, c) = \frac{1}{\Gamma(c)} \frac{w^{c-1}}{b^c} \exp\left(-\frac{w}{b}\right) \tag{20}$$

The prior over the bias vector, $\boldsymbol{y}_c = \{y_1, y_2, \ldots, y_S\}$, is a product over S zero-mean Gaussians with precision τ_{y_j}

$$p(\boldsymbol{y}) = \prod_{c=1}^{C} \prod_{j=1}^{S} \mathcal{N}(y_{c,j}; 0, \tau_{y_j}) \tag{21}$$

The prior over the sensor noise precision, λ_c, is a Gamma distribution with width and scale parameters b_{λ_c} and c_{λ_c}.

$$p(\boldsymbol{\lambda}) = \prod_{c=1}^{C} \mathcal{G}(\lambda_c; b_{\lambda_c}, c_{\lambda_c}) \tag{22}$$

The prior over each element of the mixing matrix, $\boldsymbol{A}_{ji}$ is a zero-mean Gaussian with precision α_i for each column.

$$p(\boldsymbol{A}) = \prod_{c=1}^{C} \prod_{i=1}^{L_c} \prod_{j=1}^{S} \mathcal{N}(a_{ji}|0, \alpha_i) \tag{23}$$

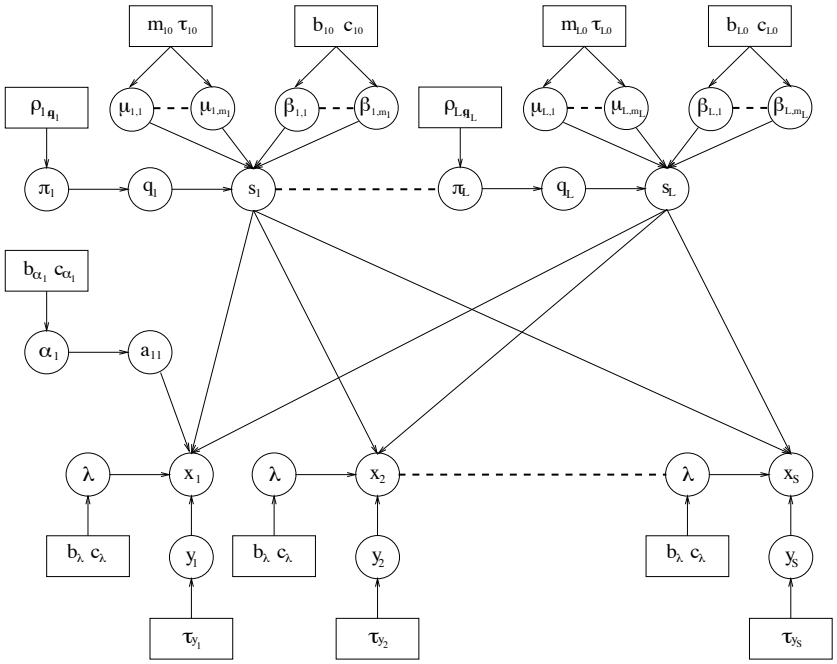


Figure 4: Graphical model of vbICA. Circles represent random variables and boxes denote parameters not governed by distributions (initially set by hand)

By monitoring the precisions $\boldsymbol{\alpha}_c = \{\alpha_1, \ldots, \alpha_{L_c}\}$, the relevance of each source may be automatically determined (ARD). If α_i is large, column i of $\boldsymbol{A}_c$ will be close to zero, indicating source i is irrelevant. Finally, the prior over each α_i is a Gamma$(b_{\alpha_i}, c_{\alpha_i})$.

$$p(\boldsymbol{\alpha}) = \prod_{c=1}^{C} \prod_{i=1}^{L_c} = \mathcal{G}(\alpha_{c,i}; b_{\alpha_i}, c_{\alpha_i}) \tag{24}$$

Figure 4 shows the graphical model for the variational Bayes ICA (vbICA) model. Bayesian inference in such a model is computationally intensive and often intractable. An important and efficient tool in approximating posterior distributions is the *variational method* (see Jordan, Ghahramani, Jaakkola, and Saul (1999) for an excellent tutorial). In particular, we take the *variational Bayes* approach detailed by Attias (1999a) and Jaakkola and Jordan (2000).

3.2 Variational Bayesian Learning

Consider the log evidence for data X

$$\log p(\boldsymbol{X}) = \log \frac{p(\boldsymbol{X}, \boldsymbol{W})}{p(\boldsymbol{W}|\boldsymbol{X})} \tag{25}$$

which simply follows from Bayes' rule and where the explicit dependence on model assumptions has been dropped for brevity. The term $\boldsymbol{W}$ is the vector of all hidden variables and unknown parameters. As the log evidence doesn't depend on $\boldsymbol{W}$, this can be re-written as

$$\begin{aligned} \log p(\boldsymbol{X}) &= \int p'(\boldsymbol{W}) \log \frac{p'(\boldsymbol{W})}{p'(\boldsymbol{W})} \frac{p(\boldsymbol{X}, \boldsymbol{W})}{p(\boldsymbol{W}|\boldsymbol{X})} d\boldsymbol{W} \\ &= \int p'(\boldsymbol{W}) \log \frac{p(\boldsymbol{X}, \boldsymbol{W})}{p'(\boldsymbol{W})} d\boldsymbol{W} + \int p'(\boldsymbol{W}) \log \frac{p'(\boldsymbol{W})}{p(\boldsymbol{W}|\boldsymbol{X})} d\boldsymbol{W} \\ &= F[\boldsymbol{W}] + KL[p'||p] \end{aligned} \tag{26}$$

where $p'(\boldsymbol{W})$ is some approximation to the posterior $p(\boldsymbol{W}|\boldsymbol{X})$ and where

$$F[\boldsymbol{W}] = \langle \log p(\boldsymbol{X}, \boldsymbol{W}) \rangle_{p'(\boldsymbol{W})} + \mathcal{H}[p'(\boldsymbol{W})] \tag{27}$$

$$KL[p'||p] = \int p'(\boldsymbol{W}) \log \frac{p'(\boldsymbol{W})}{p(\boldsymbol{W}|\boldsymbol{X})} d\boldsymbol{W} \tag{28}$$

$\mathcal{H}[p'(\boldsymbol{W})]$ is the entropy of $p'(\boldsymbol{W})$. The first term in (26), F, is known as the *mean-field* bound to the log evidence. Alternatively, $-F$ is called the *Helmholtz energy* or *variational free energy*. The second term is the *Kullback-Leibler divergence* (KL-divergence), a pseudo-distance that measures the difference between two densities. This term is strictly positive which means that F is a *strict lower bound* on the log evidence. By maximising F, not only do we minimise the KL-divergence between the approximating and true posterior, we also implicitely integrate out the unknowns $\boldsymbol{W}$. By choosing an appropriate form for the approximation $p'(\boldsymbol{W})$, we perform tractable Bayesian learning.

As F is a strict lower bound to the model (log) evidence, a wide variety of models and assumptions can be compared and contrasted by calculating F for each model. The higher F is, the higher the likelihood of the data under that model, and, therefore, the better that model is at 'explaining' the data.

3.3 Variational Learning for vbICA-MM

In our model, $\boldsymbol{W} = \{\boldsymbol{c}, \boldsymbol{s}, \boldsymbol{q}, \boldsymbol{\kappa}, \boldsymbol{\Theta}\}$. By choosing $p'(\boldsymbol{W})$ such that it factorises, terms in each hidden variable can be maximised individually. We choose the following factorisation

$$p'(\boldsymbol{W}) = p'(\boldsymbol{c})p'(\boldsymbol{s_c}|\boldsymbol{q}_c, c)p'(\boldsymbol{q}_c|c)p'(\boldsymbol{\kappa})p'(\boldsymbol{y})p'(\boldsymbol{\lambda})p'(\boldsymbol{A})p'(\boldsymbol{\alpha})p'(\boldsymbol{\varphi}) \tag{29}$$

where $p'(\boldsymbol{\varphi}) = p'(\boldsymbol{\pi})p'(\boldsymbol{\mu})p'(\boldsymbol{\beta})$ and $p'(a|b)$ is the approximating density of $p(a|b, \boldsymbol{X})$. We will also stipulate that the posteriors over the sources factorise such that

$$p'(\boldsymbol{s}_c, \boldsymbol{q}_c|c) = \prod_{i=1}^{L_c} p'(q_i|c)p'(s_{c,i}|q_i, c) \tag{30}$$

This additional factorisation allows efficient scaling of computation with the number of hidden sources, with little loss of accuracy (Miskin & MacKay, 2000). The term $p'(s_{c,i}|q_i, c)$ in (30) implies a mixture posterior source density for source i in ICA component c

$$\begin{aligned} p'(s^n_{c,i}|c) &= \sum_{q_i=1}^{m_i} p'(q^n_i = q_i|c)p'(s^n_{c,i}|q^n_i, c) && (31) \\ &= \sum_{q_i=1}^{m_i} \hat{\gamma}^n_{i,q_i} \mathcal{N}(s^n_{c,i}; \hat{\mu}^n_{i,q_i}, \hat{\beta}^n_{i,q_i}) && (32) \end{aligned}$$

where a posterior MoG density follows from from Bayes' rule.

By substituting $p(\boldsymbol{X}, \boldsymbol{W})$ and (29) into (27), we obtain expressions for the bound, F, to our model

$$F_{tot} = F_{mix} + \sum_{c=1}^{C} F_{ICA_c} \tag{33}$$

where

$$\begin{aligned} F_{mix} &= F\left[\boldsymbol{c}, \boldsymbol{\kappa}\right] \\ F_{ICA_c} &= F\left[s_c, \boldsymbol{q}_c, \boldsymbol{\lambda}_c, \boldsymbol{A}_c, \boldsymbol{\alpha}_c, \boldsymbol{\varphi}_c\right] && (34) \end{aligned}$$

The terms in equation (34) can be further factorised into contributions from each parameter. By monitoring subsets of F, we can infer the effect of local assumptions, for example F_{ICA_c} can be used instead of ARD to infer the most likely number of sources in ICA component c.

One may now proceed by specifying functional forms of each of the approximating posteriors and using these in (27) as shown by Lappalainen (1999). As shown by MacKay (1995a), however, there is no need to specify functional forms for (all) the approximating posteriors. The functional forms 'fall-out' of the maximisation process, helped by the factorised form of $p'(\boldsymbol{W})$ and the conjugacy of the priors. The optimal form for each posterior is simply given by

$$p'(W_k) \propto p(W_k) \exp\left[\langle \log p(\boldsymbol{X}, \boldsymbol{W}) \rangle_{\prod_{l \neq k} p'(W_l)}\right] \tag{35}$$

where the index k refers to the k^{th} parameter in $\boldsymbol{W}$.

If $p'(\boldsymbol{s_c}|\boldsymbol{q}_c, c) = p'(\boldsymbol{s}_c)$, free-form optimisation gives the ICA algorithms first presented by Choudrey et al. (2000). This factorisation gives a Gaussian posterior over $\boldsymbol{s}$. Using (29), however, is less restrictive and - as discussed below - more robust for an ICA mixture model. The posterior expectations of the data likelihood, (5), under the mixture density (32) are given by

$$\langle s_{c,i}^n | c \rangle = \sum_{q_i=1}^{m_i} p'(q_i^n = q_i | c) \langle {s_{c,i}}^n | q_i, c \rangle \tag{36}$$

$$\langle {s_{c,i}^n}^2 | c \rangle = \sum_{q_i=1}^{m_i} p'(q_i^n = q_i | c) \langle {s_{c,i}^n}^2 | q_i, c \rangle \tag{37}$$

where

$$p'(q_i^n = q_i | c) = \hat{\gamma}_{i,q_i}^n \tag{38}$$

$$\langle s_{c,i}^n | q_i^n, c \rangle = \hat{\mu}_{i,q_i}^n \tag{39}$$

$$\langle {s_{c,i}^n}^2 | q_i^n, c \rangle = (\hat{\mu}_{i,q_i}^n)^2 + \frac{1}{\hat{\beta}_{i,q_i}^n} \tag{40}$$

The advantages of a MoG posterior over the sources instead of a single Gaussian (Choudrey et al., 2000; Miskin & MacKay, 2000) are two-fold. Firstly, it

allows us to capture arbitrary densities, something not possible under a Gaussian posterior. Secondly, a MoG posterior is more robust under uncertainty, especially in the context of an ICA mixture model. To understand this more clearly, consider the update equations for $p'(s_{c,i}^n|c)$ and $p'(s_{c,i}^n|q_i^n, c)$ presented below (see Appendix A for notation)

$$\hat{\mu}_i^n \propto \sum_{q_i=1}^{m_i} \hat{\gamma}_{i,q_i}^n \langle \beta_{i,q_i} \rangle \langle \mu_{i,q_i} \rangle + \hat{\eta}_c^n \langle \lambda_c \rangle \sum_{j=1}^{S} \langle a_{ji} \rangle \left(x_j^n - \langle \hat{x}_{j,k\neq i}^n | c \rangle - \langle y_j \rangle \right) \quad (41)$$

$$\hat{\mu}_{i,q_i}^n \propto \langle \beta_{i,q_i} \rangle \langle \mu_{i,q_i} \rangle + \hat{\eta}_c^n \langle \lambda_c \rangle \sum_{j=1}^{S} \langle a_{ji} \rangle \left(x_j^n - \langle \hat{x}_{j,k\neq i}^n | \boldsymbol{q}_k^n, c \rangle - \langle y_j \rangle \right) \quad (42)$$

where the explicit dependence on c has been dropped for brevity. Equation (41) is the update under a Gaussian source posterior (posterior Gaussian mean) while equation (42) is the update under a MoG posterior (posterior mean for component q_i). The important feature to note is that the updates consist of prior (i.e. current) information from the MoG source model plus new data information. This information is combined and fed back up to the MoG source models. The MoGs then use this information to update their parameters.

In uncertain situations (i.e. high noise), $\langle \lambda_c \rangle$ tends towards 0, drastically down-weighting the data term. In the Gaussian posterior case, the MoG source model gets fed a *weighted average* across MoG components of the current component parameters; this leads the MoG components to update there parameters towards *common* values. If their is little data and/or data support, the MoGs will evolve towards the centroids of their respective densities rather than staying static. Over a number of iterations, they will effectively become the same single Gaussian.

If the source *posterior* is itself a MoG, then each component of the posterior MoG is responsible for its equivalent in the source *prior* MoG. In equation (42), as $\langle \lambda_c \rangle \to 0$, separate information is fed back to each component q_i - essentially there current parameter values. In this case, if there is little data and/or data support the source MoGs will remain static.

The difference is more accute in an ICA mixture model context. As well as the noise estimate, $\langle \lambda_c \rangle$, the data term is also weighted by the current ICAs responsibility for the current datum, $\hat{\eta}_c^n$. If ICA model c has little or no responsibility, its source parameters should remain static until it observes data deemed under its jurisdiction. Again this is only possible under a mixture posterior distribution for the sources.

4 Implementing vbICA-MM

The measure F is maximised using the free-form approach of (35). All the derived posteriors require solving a set of coupled parameter update equations (see Appendix A). In practice, this is best achieved by first initialising the posterior component responsibilities ($p'(\boldsymbol{c})$), use these to initialise each ICA component then commence learning on each ICA component. These components are then used to calculate the new posterior responsibilities and the learning process is repeated until convergence. The above steps may be conveniently implemented in the algorithm shown in pseudo-code form in Table 1. Once trained, the model can be used to reconstruct hidden source signals (to within a scaling and permutation) given a dataset and the (now fixed) model parameter distributions by calculating $\langle \boldsymbol{c} \rangle$, $\langle \boldsymbol{q}_c \rangle$ and $\langle \boldsymbol{s}_c \rangle$ under their respective posteriors.

4.1 Priors and Initialisation

Broad priors have little or no effect on the results as these allow the data to 'do the talking'. Such priors allow parameters and hidden variables to remain within sensible boundaries while encoding very little knowledge of what values they might take. Allowing the priors to become too broad, however, can reduce their regularising abilities leading to implausible magnitudes and dangers of over-fitting. Narrow priors, on the other hand, encode strong assumptions about possible parameter and variable values and have a discernable effect

initialise;
WHILE ($\Delta F_{(ica-mm)} <$ tolerance)
  FOR every ICA component
    WHILE ($\Delta F_{(ica)} <$ tolerance)
      update ICA observation model by cycling through
      equations (54)-(59) until convergence;
      update ICA source model by cycling through
      equations (60)-(66) until convergence;
      update ICA noise model using equations (67)-(70);
      calculate $F^{new}_{(ica)}$;
      calculate $\Delta F_{(ica)} \doteq |F^{new}_{(ica)} - F^{old}_{(ica)}|$;
    END WHILE;
  END FOR;

  update ICA-MM indicator probabilities and parameters
  using equations (71)-(73);
  calculate $F^{new}_{(ica-mm)}$;
  calculate $\Delta F_{(ica-mm)} \doteq |F^{new}_{(ica-mm)} - F^{old}_{(ica-mm)}|$;
END WHILE;

Table 1: Pseudo-code for vbICA-MM updates.

on the inference process. If the priors are too narrow, the model becomes inflexible and its ability to learn is comprimised.

For the ICA mixture model presented in this paper, we examined a wide variety of priors ($10^{-6} \leq b \leq 10^6$, $10^{-6} \leq c \leq 10^6$ for all Gamma distributions, scale parameter = 5-5000 for all Dirichlets and precision = $10^{-6} - 10^6$ for Gaussians). For Gamma distributions, the variance is given by b^2c and the mean by bc. Large values of c give distributions that are highly peaked around the mode, tending towards Gaussianity. Values of $c < 1$ have no mode and

are more like exponentials. Consequently, they encourage small values for the random variate while curtailing large values beyond the mean. We found $1 \leq b \leq 1000$ with $bc \approx 1$ a useful range for Gamma distributions. Gamma distributions govern scale parameters, in this case precisions. Values of $bc > 10$ have a narrowing effect on the posteriors, making them over-confident. Values of $bc < 10^{-1}$ with $c > 1$ have a smoothing effect on the posteriors, loosing detail in the process. Similarly, we have found precisions of the order of $10^0 - 10^{-3}$ a good range. The value of Dirichlet scale parameters act as 'pseudo-counts', so high values are very constraining. Appropriate values depend on the number of data-vectors. If N is the number of data-vectors, then values between $0.01N$ and $0.1N$ stop components dying while allowing the data to set the posterior values accurately. If the algorithm was recast as an online scheme, these values would have to be reexamined as the results would necessarily become more sensitive to the choice of priors.

The choice of initialisation is also important. As with any ICA formalism, the vbICA-MM algorithm finds a local maximum in the parameter space and, as such, results depend on initialisation. There are many ways to initialise; we recommend that the reader explore the different options available to work out which is most suitable for them.

We have investigated two initialisation schemes - random and SVD. We initialise the model by first running C-component k-means on the data. The class responsibilities generated are used to partition the data into C segments. These segments are used to initialise C ICA models either randomly or by using SVD on each segment. If SVD is used then the first L_c columns of the SVD data-matrix are used to initialise L_c MoGs using k-means, while residual singular values are used to estimate the noise. The first L_c columns of the SVD projection matrix are used to initialise the mixing matrix. We have found random initialisation the most useful overall, particularly for modelling clusters that are best described by non-orthogonal axes. Initialising using SVD (tantamount to initialising at the PCA solution) yields a quicker result, but

vbICA-MM can have problems with resolving directions in clusters exhibiting high correlation, often staying at - or close to - the PCA result. This is discussed further in Section 6.1.

One must also choose the number of components in each source MoG. Although this can be determined by monitoring the contibution to F by each source ($F[\varphi_{c,i}]$) as a function of components, this is time consuming. We have found that 3 components are enough for most datasets, although 5 should be used for image data.

5 Results

We first demonstrate the versatility of the vbICA-MM algorithm on 2- and 3- dimensional synthetic data, each grouped into 3 classes. The algorithm's ability to model intricate, highly non-Gaussian data is highlighted, as is its structure determination using ARD and the mean-field bound, F, to the log-evidence. We then use vbICA-MM on real functional Magnetic Resonance Imaging (fMRI) data to decompose the images into interpretable, independent features, and compare the results with previous methods. We set vague priors ($b = 1000$, $c = 1e-3$ for all Gamma distributions, scale parameter $= 5$ for all Dirichlets and precision = 1e-3 for Gaussians) to encode poor prior knowledge for both the synthetic and real datasets. Figures can be found at the end of Section 6.

5.1 Test Data

We tested the vbICA-MM algorithm on 2-dimensional and 3-dimensional synthetic test data drawn from three classes and with 10% added Gaussian noise. We set 3 components per source MoG and commenced learning using the algorithm presented in Table 1. Training continued until F changed by less than 0.01 percent or the number of iterations reached 200. Typically, iterations of each ICA component start high (approximately $100 - 200$ for the

non-orthogonal cluster presented below and < 100 for the orthogonal clusters), and end with only a handful of iterations by the end.

In the 2d case, the underlying source manifolds were 2-dimensional. Figure 5(a) shows the 3 clusters, with 1 Gaussian cluster, 1 non-Gaussian cluster described by orthogonal directions, and 1 highly correlated cluster made up of 6 sub-clusters. For this dataset, vbICA-MM was intialised randomly. Training took 9 iterations of the vbICA-MM algorithm. Figure 5(b) shows the partition by vbICA-MM trained on 3000 points (1000 from each cluster) compared with the original data - only 12 (0.4%) data were assigned incorrectly. The 1st-deviation of the underlying MoG components are shown as ellipses. Note how the Gaussian source is deemed 1-dimensional while the 2-dimensional structure of the other classes has been clearly captured. A Gaussian source is unidentifiable from Gaussian noise in ICA. Its principle direction is captured by the single source while its 'spread' is absorbed by the (Gaussian) noise variance. The vbICA-MM data density model in Figure 5(e) shows the multi-modal structure captured within the clusters. In comparison, a MoG (5(c)) and the generalised exponential decorrelating ICA mixture model (gedecICA-MM) presented by Penny and Roberts (2001) (5(d)) show no structure within the classes. Subsequently, vbICA-MM gives a more efficient representation of the true data density, with little probability mass wasted in regions of low density within each cluster.

The data density of the 3d test data is shown in Figure 6(a). The 3 clusters are intrinsically 1-, 2- and 3-dimensional. We initialised a vbICA-MM using SVD and trained on 1500 points, 500 drawn from each cluster. Training took 5 iterations of the vbIVA-MM algorithm. Figure 6(b) depicts the model captured by vbICA-MM showing accurate representation of the clusters and cluster structures. The Hinton diagrams (Hinton, 1989) of the mixing matrices in Figures 6(c)-(e) show how vbICA-MM has used ARD to correctly infer the latent dimensionalities. Unsupported columns in the matrices have been suppressed by small ARD coefficients (inverse α_i's shown in Figure 7(a)), effec-

tively 'switching-off' unnecessary sources in the source reconstructions below. Another way of inferring the latent dimensionality of each ICA component is to monitor the free-energy of the model. As implied by equation (34), this is equivalent to monitoring the contribution each ICA component makes to the overall free-energy of the model. Figure 7(b) is a plot of the mean-field bound (i.e. the negative of the free-energy) for each ICA component, F_{ICA_c}. Each curve has had its maximum subtracted such that the trends in Figure 7(b) are more apparent. These curves confirm the latent dimensionality inferred by ARD. Plotting the overall mean-field bound, F, across components in Figure 7(c) correctly infers 3 clusters. There is a further peak at 5 components (and, indeed, progressively shallower ones at 7 and 9) as ICAs latch on to different sub-clusters in the 3d cube. It must be noted, however, that F is a bound to the *log* evidence, so these further peaks disappear when exponentiated to yield the evidence bound. In comparison, a Bayesian MoG infers 21 clusters, while the BIC in gedecICA-MM predicts 9. Figure 8 shows how well the supported sources' multi-modal densities have been modelled by the MoGs. The ability of vbICA-MM to capture undulating manifolds is fundamental in correct inferring the true number of clusters.

As a comparison, Figure 7(c) also plots the Bayesian Information Criterion (BIC) approximation found to the log-evidence using MAP estimates (BIC is the negative of Maximum Description Length modelling - see Rissanen (1994) for an introduction). Interestingly, this also picks out the correct model-order, with a higher score for the log-evidence. Attias (1999a) showed that the BIC is equivalent to variational Bayes in the infinite data limit. The BIC achieves a higher score presumably because it is a much simpler model, with no model averaging, although we must admit surprise at the difference. We must point out, however, that Penny, Roberts, and Everson (2001) have suggested that penalised likelihood is more applicable to ICA than other models due to the likely violation of the independence assumption in overcomplex models, leading to a penalising effect on the data likelihood itself. This suggests that a full

variational integration is not necessary in the high data/low noise limit and that penalised-likelihood MAP learning is adequate. With more noisy data, however, Penny et al. (2001) showed that model order selction using penalised likelihood quickly breaks down, so in uncertain situations a full Bayesian approach would be more robust.

5.2 Real data

We used the vbICA, variational Bayesian Mixture of Factor Analysers (vbMFA), gedecICA-MM and vbICA-MM algorithms on fMRI images of a slice through a tumour patient's brain. Data was collected using both T2 and proton density (PD) spin sequences, which are used directly to form a two-dimensional feature space. Two 100x100 pixel images were vectorised, and 2000 samples were randomly drawn from the subsequent 2d data. Models with a range of latent dimensions were trained on the 2000 data vectors, and the most likely models were then used to unmix the complete 10000 points dataset into the corresponding independent features.

Figure 9 shows the two original (colour) images used along with the feature extracted by vbICA and the gedecICA mixture model. Due to the inherent limitations of ICA, no more than two features could be extracted from the fMRI images. The vbICA algorithm favoured a 2-source ICA model, shown in Figure 9(b). The left-hand feature is simply a copy of one of the original images. The right-hand image has separated out a local feature, which is, in fact, cerebro-spinal fluid. The vbMFA algorithm infers an 8 component model with $F = -14813.75$, giving the features in 9(c). The Bayesion Information Criterion of the gedecICA-MM chose a 4-component model with 2 sources per ICA component, giving the eight overall features in Figure 9(d). Although the gedecICA-MM has managed to separate out the cerebro-spinal fluid, most features are simply scaled copies of each other and, therefore, the gedecICA-MM over-represents the fMRI images.

We trained 1-5 component models using vbICA-MM, with each ICA com-

ponent having the maximum 2 allowed sources. The negative free-energy (i.e. F) plot in Figure 10(a) shows that a 2-component model is preferred [3], while the Hinton plots in Figures 10(b)-(c) infer 1-source and 2-source ICA components. Compared to vbMFA, vbICA-MM had $F = -3618.8$, a substantial improvement. The 3 features extracted by the most likely model are presented in Figure 11 along with their learnt distributons. The single source of the first ICA component - shown in Figure 11(a) - is global 'background' brain-tissue detail. The second ICA component represents more local features where the central part of Figure 11(b) is, once again, the cerebro-spinal fluid. The two-tone background demarcates white and grey brain matter and has not been separated out, unlike some of the features shown in Figure 9. The reasons for this are as yet not clear. More interestingly, however, the second source has extracted 2 dark 'blobs'. These are the tumors, which neither vbICA, vbMFA or gedecICA-MM picked out. The features' respective MoGs can be interpreted as the distribution of colours in each feature. The tumors' distribution is heavily peaked around blue, with the left-hand tail capturing the yellow-green information. The other distributions similarly represent blue-red from left to right. The ability of vbICA-MM to capture multi-modal feature distributions is pivotal in allowing the complex background distribution to be separated from the rest. This representation is thus much more interpretable and efficient than either simple ICA or the gedecICA-MM.

Figure 12 illustrates how the original data density has been modelled and partitioned by both gedecICA-MM and vbICA-MM. The lighter cluster in Figure 12(d) is described by the 1d ICA component, while the darker area is the 2d ICA cluster. vbICA-MM has managed to partition the data density into 2 clusters which are not linearly separable.

[3]Incidently, the BIC prefered a 1-component model

6 Discussion

In this paper, we have presented an algorithm for modelling multi-dimensional, non-Gaussian data distributions. The vbICA-MM algorithm splits the distribution into self-similar areas, and uses ICA components to learn representations of these areas. The ICAs model these local cluster manifolds by forming independent directions in the underlying distribution. Automatic Relevance Determination selects the appropriate dimensionality of each manifold, and a Bayesian learning scheme allows the optimum number of ICA components to be inferred. We have demonstrated the algorithm on 2-dimensional and 3-dimensional data by faithfully modelling intricate, discontinuous clusters whilst simultaneously inferring their intrinsic dimensions, something not possible under previous ICA mixture model. We have shown how it is possible to compare assumptions underlying various models, in particular using the negative free energy, F, to ascertain the correct number of clusters. We have used vbICA-MM to decompose real fMRI images into interpretable, self-similar features, blindly and automatically.

6.1 Problems to Overcome

Although we have comprehensively demonstrated the versatility and robustness of the vbICA-MM algorithm, there are practical issues to address, in particular initialisation and speed.

As discussed in Section 4.1, the choice of initialisation can have an effect on the final solution. SVD initialisation always starts at the same place for a given dataset so always ends at the same solution. In practice, we have found different random initialisations generally lead to the same solution as well. However, if parts of the data density are highly correlated, the ICA component most responsible for that area will stay close to the PCA solution if initialised using SVD. Different random initialisations may lead to different soultions, or may converge onto the PCA/FA solution. ICA can be seen as a generalisation

of Principle Component Analysis and Factor Analysis. As shown by Attias (1999b), noiseless ICA with Gaussian sources is equivalent to PCA, while FA is the same with non-isotropic noise. This means the PCA/FA solution is a subset of all possible ICA solutions for a given dataset. As a particular cluster becomes more and more correlated (or, equivalently, its independent directions become less orthogonal), the second moment starts to dominate so the cluster becomes easier to describe by a Gaussian density. This means the PCA/FA maxima in the ICA solution space become more prominent so more and more initialisation points will lead to them. The nonorthogonal cluster in Figure 5 was chosen specifically as, empirically, it seems to be the 'limit' of correlation/nonorthogonality that vbICA-MM can resolve into independent directions. This particular cluster has a correlation of -0.67 and we needed - on average - 5 random initialisations before vbICA-MM 'latched-on, as indicated by the significantly higher value for F after 1 iteration for this initialisation. More investigation is needed into initialisation strategies to overcome this. One possible solution is to decorrelate or 'whiten' each cluster identified by the k-means step, then commence learning on these whitened data. We have had encouraging results using this, admittedly, rather *ad hoc* method. Also, the factorial approximation in (30) is no longer a good approximation for highly correlated data, so dropping this will also solve much of the problem, albeit with a corresponding reduction in speed.

Indeed, speed is the other problem. Bayesian learning is inherently slower than non-Bayesian methods due to the extra parameters that have to be estimated. In fact, Bayesian inference can be shown to be NP-hard (Cooper, 1990). The variational approximation brings tractability to the Bayesian formulation, and is speedier than the non-deterministic sampling regime, but it still remains understandably slower than its non-Bayesian counterparts. Interesting work by Valpola and Pajunen (2000) on speeding up Bayesian ICA may provide a basis for a faster vbICA-MM algorithm. Also, model-order selection for the number of ICA components can become prohibitive if a large

number are required. Some progress may be made in this area if, for example, some sort of birth-death (Roberts, Holmes, & Denison, 2001) and split-merge (Ueda, Nakano, Ghahramani, & Hinton, 1999) criteria could be enforced allowing active generating, culling, splitting and merging of components during the learning process. This is an area of on-going research.

6.2 Further Applications

Interesting data are - almost by definition - not Gaussian distributed. For data distributions that consist of self-similar non-Gaussian clusters, we believe this model will provide a better representation over current methods, such as Gaussian mixture modelling, Mixtures of Factor Analysers and plain-vanilla ICA. The kind of data that can benefit from such an analysis range from the prosaic to the provocative.

The most obvious area is in non-stationary blind source separation, as alluded to in Section 1.2. Standard ICA is now the de facto method for blindly separating mixtures of signals. It has shown to be very robust, effective and - in a probabilistic formulation - highly interpretable. However, ICA is essentially limited to stationary mixings of independent sources which number less than the sensors picking up the observations. ICA mixture models are an important step in overcoming these restrictions. Sudden changes in the mixing amounts will be encoded by vbICA-MM as a switching between ICA components. In the cocktail party problem, for example, this would include different voices cutting in and out over a background cacophony. Provided the number of voices (plus background) active at any one time does not exceed the total number of sensors, vbICA-MM can capture more sources than sensors. This was demonstrated in the fMRI example, where 3 features were extracted from 2-dimensional data. Furthermore, the independence assumption at the core of ICA holds *within* components but is not necessary *between* componenets. Sources which are not independent will be represented as sources within different ICA components by vbICA-MM. The application of vbICA-

MM to enhanced Blind Source Separation does not have to stop at auditory signals. The multi-modal nature of the source distributions lend themselves readily to image separation and other BSS problems.

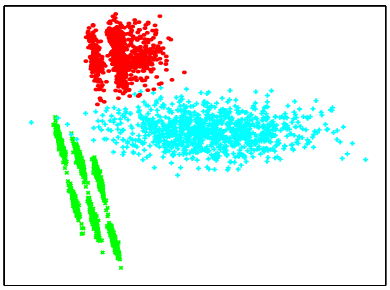
Clustering in general is another area that naturally comes under vbICA-MMs umbrella. Mixture modelling is the definitive formalism for finding clusters in data distributions. Interesting clusters are very often not Gaussian, so mixture models based on Gaussianity will generally not model such clusters well. The synthetic clusters in Section 5 were non-Gaussian and also contained structure within themselves. This confused both a Gaussian mixture model and the mixture of Factor Analysers model. The former did not model the data accurately, while the latter failed to infer the correct number of clusters. Lee et al. (2000) have shown that their ICA mixture model performed better at clustering the Iris data of Fisher (1936) than those based on Gaussian methods. The grouping of documents into semantic clusters is currently a very active area of research (see Landaur, Foltz, and Laham (1998) for an introduction). It is highly unlikely that such man-made corpora are intrinsically Gaussian in nature.

The reader may have noticed that in the use of ICA as a representative tool, we have verged on the evangelical. This, we strongly believe, is where ICA and its extensions will prove to be indispensable. Lee et al. (1999) have already shown that ICA mixture models based on Laplacian source densities are over 20 percent more efficient than PCA at coding natural scenes and images of newspaper text, and more efficient than standard ICA. As ICA is only beginning to leave its BSS beginnings behind, this area is, as yet, largely unexplored.

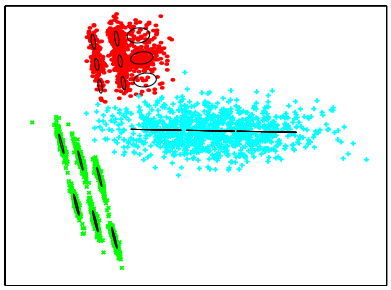
6.3 Future Directions

There are a number of modifications that could be made to vbICA-MM to take into account further information that one may have on the data. For example, the source models could be restricted to being the same dimensionality for

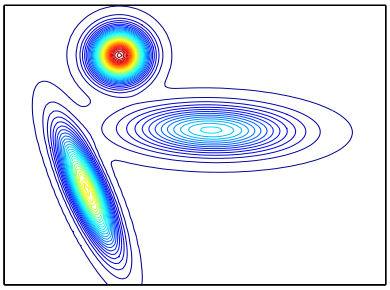
all ICA components if it is expected that the data lie on some dimensionally consistent manifold. Indeed, the source models could be shared amongst a number, if not all, of the ICA components if different regions of the data space are thought to be different projections of some handful of underlying manifolds. Temporal information be may introduced if the mixture process prior is conditioned across samples as in, for example ,a Markov process, leading to a Bayesian formulation of Hidden Markov ICA (Penny, Everson, & Roberts, 2000). vbICA-MM leads to a whole family of ICA mixture models depending on which assumptions are coded into the model. Because of the Bayesian formulation, the effectiveness of these models can be quantitatively compared. Again, this is an area of active research.



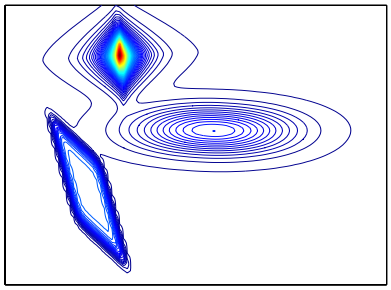
(a) Data density



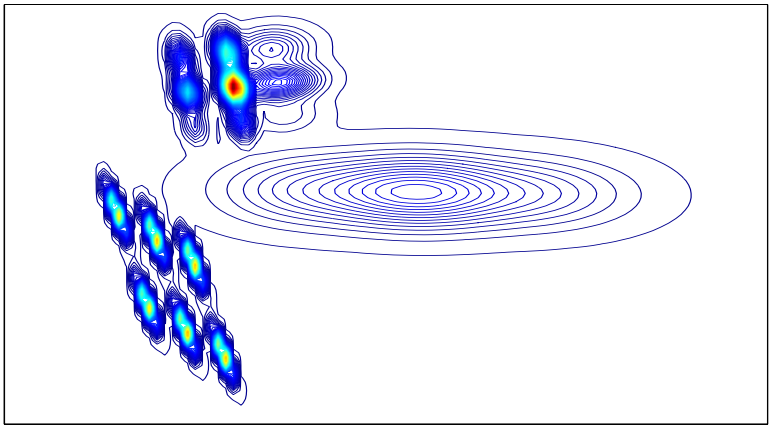
(b) vbICA-MM partition



(c) Gaussian mixture model

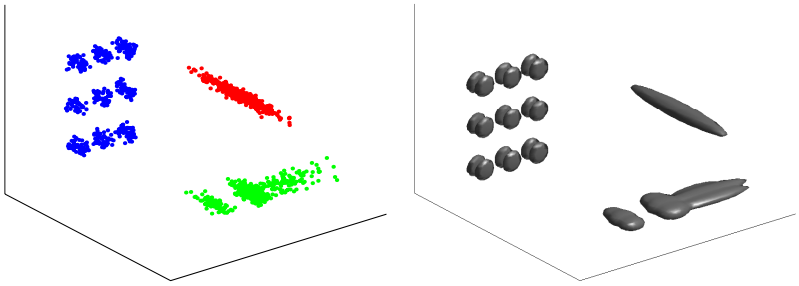


(d) gedec-ICA mixture model



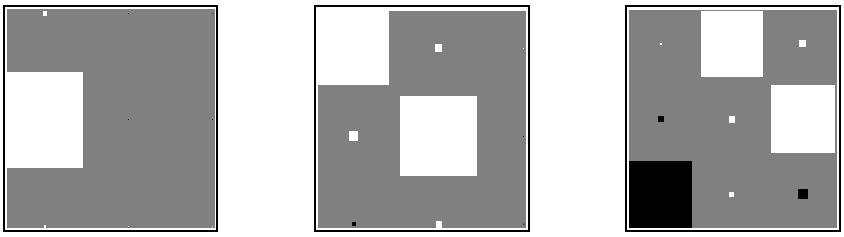
(e) vbICA mixture model

Figure 5: 2d test data results. Contours plot $0.001 \leq p(x) \leq 1$ in 250 intervals



(a) Data density

(b) vbICA-MM isosurface of $p(x) = 0.01$



(c) 1d cluster

(d) 2d cluster

(e) 3d cluster

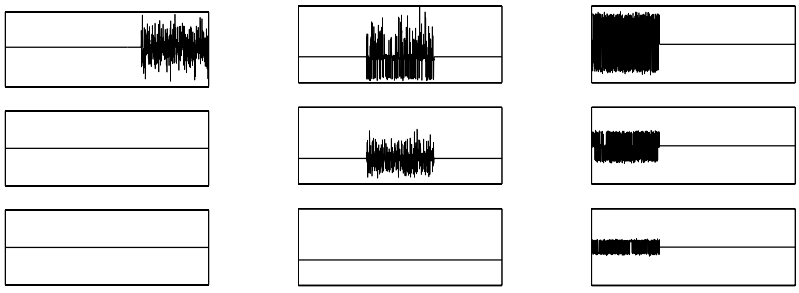
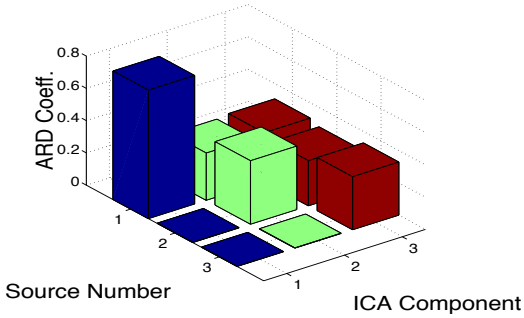
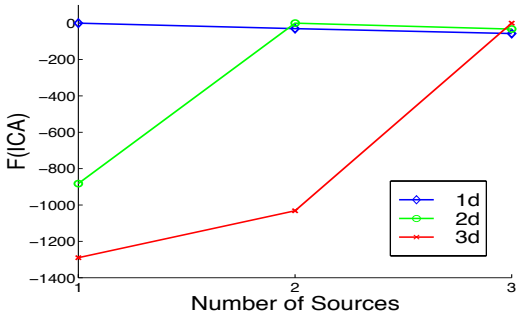


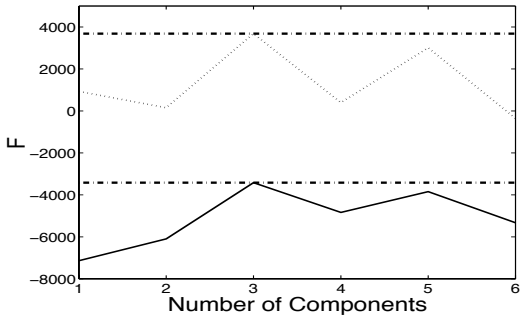
Figure 6: 3d test data results



(a) ARD Coefficients

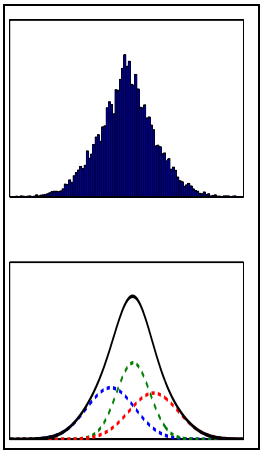


(b) F_{ICA} for each ICA component

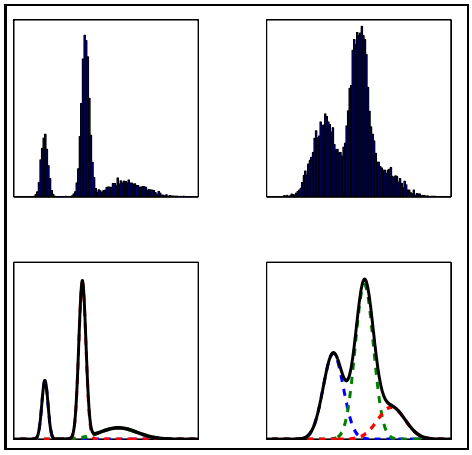


(c) Solid: vbICA-MM negative free-energy Dotted: BIC using MAP estimates

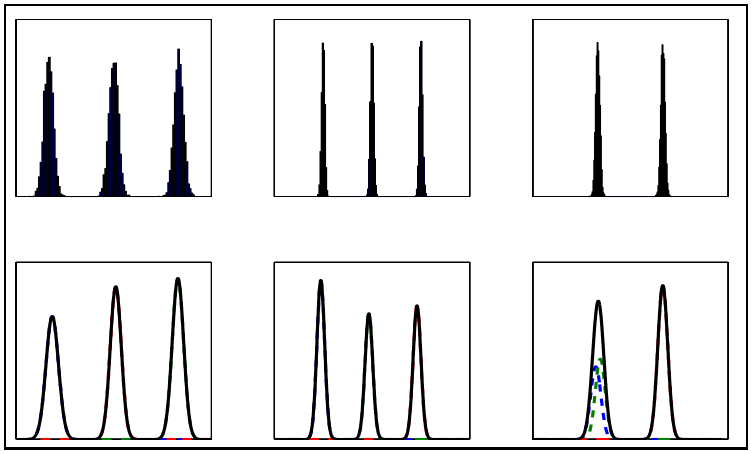
Figure 7: Structure determination using ARD and negative free energy



(a) ICA 1

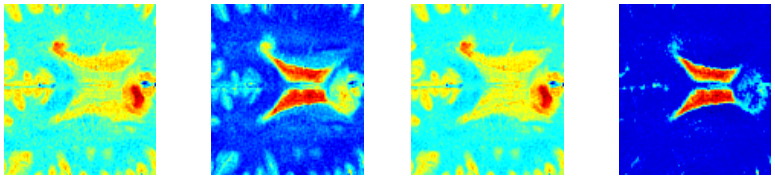


(b) ICA 2



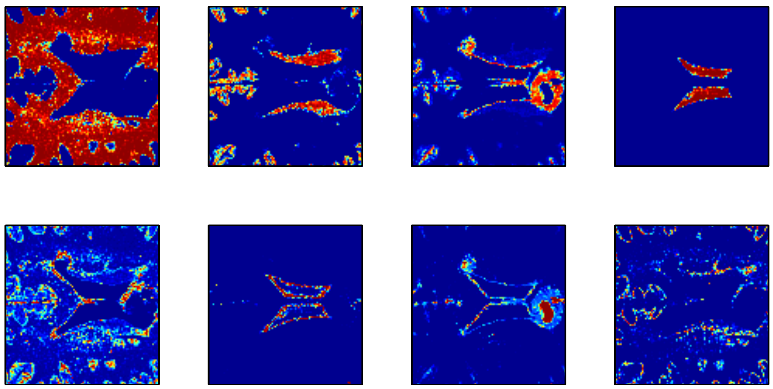
(c) ICA 3

Figure 8: Top: Original source distributions Bottom: Learnt distributions

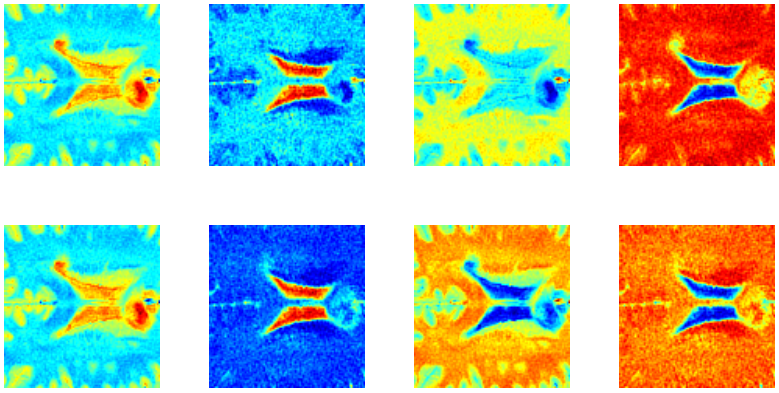


(a) Original data

(b) ICA features

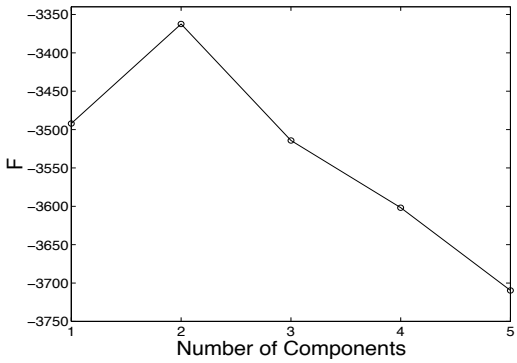


(c) vbMFA features

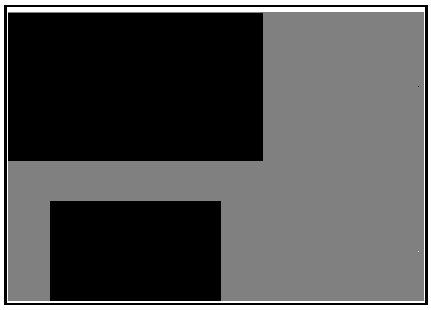


(d) gedecICA-MM features

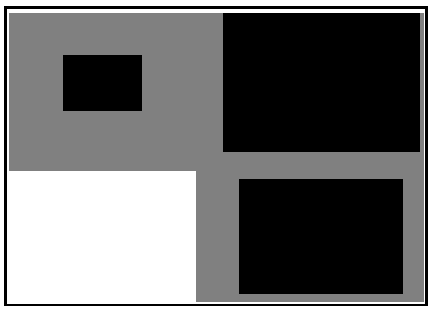
Figure 9: fMRI images and ICA/gdecICA-MM features extracted



(a) vbICA-MM negative free-energy

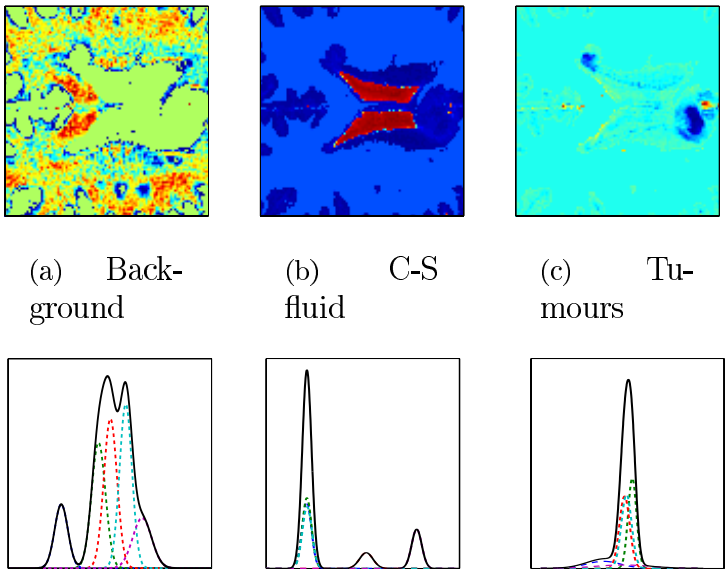


(b) 1d cluster



(c) 2d cluster

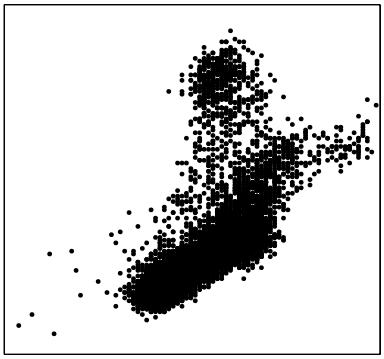
Figure 10: Inferred latent structure for fMRI images



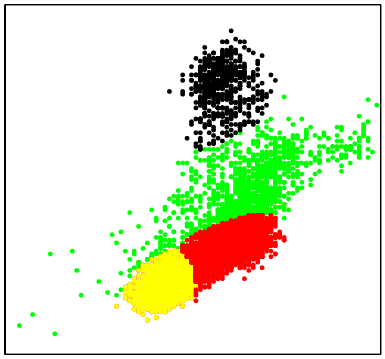
(a) Background (b) C-S fluid (c) Tumours

(d) ICA 1 (e) ICA 2

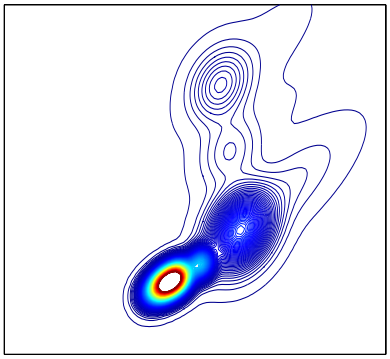
Figure 11: vbICA-MM features extracted from fMRI images and respective ICA component source distributions



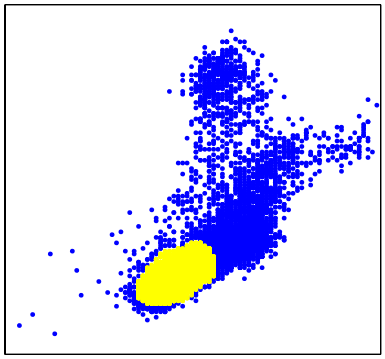
(a) fMRI data density



(b) gedecICA-MM partition



(c) vbICA-MM data model



(d) vbICA-MM partition

Figure 12: vbICA-MM model for fMRI images

A Update Equations

The parameter posteriors are given by

$$p'(\boldsymbol{s}_c|\boldsymbol{q}_c, c) = \prod_{n=1}^{N}\prod_{i=1}^{L_c}\mathcal{N}(s_{c,i}^n; \hat{\mu}_{i,q_i}^n, \hat{\beta}_{i,q_i}^n) \tag{43}$$

$$p'(\boldsymbol{A}) = \prod_{c=1}^{C}\prod_{i=1}^{L_c}\prod_{j=1}^{S}\mathcal{N}(a_{ji}; \hat{m}_{a_{ji}}, \hat{\alpha}_{ji}) \tag{44}$$

$$p'(\boldsymbol{\alpha}) = \prod_{c=1}^{C}\prod_{i=1}^{L_c}\mathcal{G}(\alpha_i; \hat{b}_{\alpha_i}, \hat{c}_{\alpha_i}) \tag{45}$$

$$p'(\boldsymbol{q}_c|c) = \prod_{n=1}^{N}\prod_{i=1}^{L_c}\hat{\gamma}_{i,q_i}^n \tag{46}$$

$$p'(\boldsymbol{\pi}) = \prod_{c=1}^{C}\prod_{i=1}^{L_c}\prod_{q_i=1}^{m_i}\frac{(\sum_{q_{i'}}\hat{\lambda}_{i,q_{i'}})}{\Gamma(\hat{\lambda}_{i,q_i})}\pi_{i,q_i}^{\hat{\lambda}_{i,q_i}-1} \tag{47}$$

$$p'(\boldsymbol{\mu}) = \prod_{c=1}^{C}\prod_{i=1}^{L_c}\prod_{q_i=1}^{m_i}\mathcal{N}(\mu_{i,q_i}; \hat{m}_{i,q_i}, \hat{\tau}_{i,q_i}) \tag{48}$$

$$p'(\boldsymbol{\beta}) = \prod_{c=1}^{C}\prod_{i=1}^{L_c}\prod_{q_i=1}^{m_i}\mathcal{G}(\beta_{i,q_i}; \hat{b}_{i,q_i}, \hat{c}_{i,q_i}) \tag{49}$$

$$p'(\boldsymbol{y}) = \prod_{c=1}^{C}\prod_{j=1}^{S}\mathcal{N}(y_j; \hat{m}_{y_j}, \hat{\tau}_{y_j}) \tag{50}$$

$$p'(\boldsymbol{\lambda}) = \prod_{c=1}^{C}\mathcal{G}(\lambda_c; \hat{b}_{\lambda_c}, \hat{c}_{\lambda_c}) \tag{51}$$

$$p'(\boldsymbol{c}) = \prod_{n=1}^{N}\prod_{c=1}^{C}\hat{\eta}_c^n \tag{52}$$

$$p'(\boldsymbol{\kappa}) = \prod_{c=1}^{C}\frac{(\sum_{c'}\hat{\iota}_{c'})}{\Gamma(\hat{\iota}_c)}\kappa_c^{\hat{\iota}_c-1} \tag{53}$$

Updated distribution parameters are hatted versions of the original parameters, $\langle a\rangle$ are expectations taken w.r.t. $p'(a)$ and $\langle a|b\rangle$ are expectations w.r.t.

$p'(a|b)$.

A.1 Observation Model

A.1.1 $p'(\boldsymbol{s}_c|\boldsymbol{q}_c, c)$

$$\hat{\mu}^n_{i,q_i} = \frac{1}{\hat{\beta}^n_{i,q_i}}\left[\langle\beta_{i,q_i}\rangle\langle\mu_{i,q_i}\rangle + \hat{\eta}^n_c\langle\lambda_c\rangle\sum_{j=1}^{S}\langle a_{ji}\rangle\left(x^n_j - \langle\hat{x}^n_{j,k\neq i}|\boldsymbol{q}^n_k, c\rangle - \langle y_j\rangle\right)\right] \quad (54)$$

$$\hat{\beta}^n_{i,q_i} = \langle\beta_{i,q_i}\rangle + \hat{\eta}^n_c\langle\lambda_c\rangle\sum_{j=1}^{S}\langle a^2_{ji}\rangle \quad (55)$$

where

$$\begin{aligned}
\hat{x}^n_{j,k\neq i} &= \sum_{k\neq i}^{L_c} a_{jk}s^n_{c,k} \\
\langle\hat{x}^n_{j,k\neq i}|c\rangle &= \sum_{k\neq i}^{L_c}\langle a_{jk}\rangle\langle s^n_{c,k}|c\rangle \\
\langle\hat{x}^n_{j,k\neq i}|\boldsymbol{q}^n_k, c\rangle &= \sum_{k\neq i}^{L_c}\langle a_{jk}\rangle\langle s^n_{c,k}|\boldsymbol{q}^n_k, c\rangle
\end{aligned}$$

and define

$$\begin{aligned}
\hat{x}^n_j &= \sum_{i=1}^{L_c} a_{ji}s^n_{c,i} \\
\langle\hat{x}^n_j|c\rangle &= \sum_{i=1}^{L_c}\langle a_{ji}\rangle\langle s^n_{c,i}|c\rangle \\
\langle\hat{x}^n_j|\boldsymbol{q}^n_i, c\rangle &= \sum_{i=1}^{L_c}\langle a_{ji}\rangle\langle s^n_{c,i}|\boldsymbol{q}^n_i, c\rangle
\end{aligned}$$

The expectations for the sources $\boldsymbol{s}$ are given by the expressions in Section 3.3. In practice, equation (54) has to be iterated for every i a number of times until $\hat{\mu}^n_{i,q_i}$ converges as it depends on every other $k \neq i$.

Note the intuitive form of (54) and (55). Source i 'sees' data x_j^n at sensor j and works out what it can 'explain away' given information in the rest of the model, i.e. $\hat{x}_{j,k\neq i}^n$. The residual information is then used to update its own parameters in a bid to explain what's 'left over'. This is analogous to the message passing algorithms of Pearl (1988), whereby a node ($s_{c,i}$) updates its belief (encapsulated in $\hat{\mu}_{i,q_i}^n$ and $\hat{\beta}_{i,q_i}^n$) using messages passed to it by it's parents (the MoG expectations), its children (the sensor data) and all other parents of it's children (i.e. all other sources and quantified by $\hat{x}_{j,k\neq i}$). This use of information from a variable's Markov Blanket runs through all the update equations.

A.1.2 $p'(\boldsymbol{A})$

$$\hat{m}_{a_{ji}} = \frac{\langle \lambda_c \rangle}{\hat{\alpha}_{ji}} \sum_{n=1}^{N} \hat{\eta}_c^n \langle s_{c,i}^n | c \rangle \left(x_j^n - \langle \hat{x}_{j,k\neq i}^n | c \rangle - \langle y_j \rangle \right) \tag{56}$$

$$\hat{\alpha}_{ji} = \langle \alpha_i \rangle + \langle \lambda_c \rangle \sum_{n=1}^{N} \hat{\eta}_c^n \langle {s_{c,i}^n}^2 | c \rangle \tag{57}$$

In practice, equation (56) has to be iterated for every i a number of times until $\hat{m}_{a_{ji}}$ converges.

A.1.3 $p'(\boldsymbol{\alpha}_c)$

$$\hat{b}_{\alpha_i} = \left(\frac{1}{b_{\alpha_i}} + \frac{1}{2} \sum_{j=1}^{S} \langle a_{ji}^2 \rangle \right)^{-1} \tag{58}$$

$$\hat{c}_{\alpha_i} = c_{\alpha_i} + \frac{S}{2} \tag{59}$$

A.2 Source Model

A.2.1 $p'(\boldsymbol{q}_c|c)$

$$\gamma^n_{i,q_i} = \tilde{\pi}_{i,q_i}\left(\frac{\tilde{\beta}_{i,qi}}{\hat{\beta}^n_{i,qi}}\right)^{\frac{1}{2}} \exp\left[\frac{1}{2}\left(\hat{\beta}^n_{i,qi}\hat{\mu}^{n^2}_{i,q_i} - \langle\beta_{i,qi}\rangle\langle\mu^2_{i,q_i}\rangle\right)\right] \tag{60}$$

$$\hat{\gamma}^n_{i,q_i} = \frac{\gamma^n_{i,q_i}}{\sum_{q'_i}\gamma^n_{i,q'_i}} \tag{61}$$

where

$$\tilde{\pi}_{i,q_i} = \exp\left[\Psi(\hat{\rho}_{i,q_i}) - \Psi(\sum_{q'_i}\hat{\rho}_{i,q'_i})\right]$$

$$\tilde{\beta}_{i,qi} = \hat{b}_{i,q_i}\exp\left[\Psi(\hat{c}_{i,q_i})\right]$$

and where (61) ensures that $\sum_{q_i}\hat{\gamma}^n_{i,q_i} = 1$. $\Psi(.)$ is the digamma function.

A.2.2 $p'(\boldsymbol{\pi})$

$$\hat{\rho}_{i,q_i} = \rho_{i,q_i} + \sum_{n=1}^{N}\hat{\gamma}^n_{i,q_i} \tag{62}$$

A.2.3 $p'(\boldsymbol{\mu})$

$$\hat{m}_{i,q_i} = \frac{1}{\hat{\tau}_{i,q_i}}\left(\tau_{i0}m_{i0} + \langle\beta_{i,q_i}\rangle\sum_{n=1}^{N}\hat{\gamma}^n_{i,q_i}\langle s^n_{c,i}|q^n_i, c\rangle\right) \tag{63}$$

$$\hat{\tau}_{i,q_i} = \tau_{i0} + \langle\beta_{i,q_i}\rangle\sum_{n=1}^{N}\hat{\gamma}^n_{i,q_i} \tag{64}$$

A.2.4 $p'(\boldsymbol{\beta})$

$$\hat{b}_{i,q_i} = \left(\frac{1}{b_{i0}} + \frac{1}{2}\tilde{\sigma}_{i,q_i}\right)^{-1} \tag{65}$$

$$\hat{c}_{i,q_i} = c_{i0} + \frac{1}{2}\sum_{n=1}^{N}\hat{\gamma}_{i,q_i}^{n} \tag{66}$$

where we define the average variance of component q_i in source i

$$\tilde{\sigma}_{i,q_i} = \sum_{n=1}^{N}\hat{\gamma}_{i,q_i}^{n}\left(\langle {s_{c,i}^{n}}^2 | q_i^n, c\rangle - 2\langle\mu_{i,q_i}\rangle\langle s_{c,i}^{n}|q_i^n,c\rangle + \langle\mu_{i,q_i}^2\rangle\right)$$

A.3 Noise Model

A.3.1 $p'(\boldsymbol{y})$

$$\hat{m}_{y_j} = \frac{\langle\lambda_c\rangle}{\hat{\tau}_{y_j}}\sum_{n=1}^{N}\hat{\eta}_c^n\left(x_j^n - \langle\hat{x}_j^n|c\rangle\right) \tag{67}$$

$$\hat{\tau}_{y_j} = \tau_{y_j} + \langle\lambda_c\rangle\sum_{n=1}^{N}\hat{\eta}_c^n \tag{68}$$

A.3.2 $p'(\lambda)$

$$\hat{b}_{\lambda_c} = \left[\frac{1}{b_{\lambda_c}} + \frac{1}{2}\sum_{j=1}^{S}\sum_{n=1}^{N}\hat{\eta}_c^n\langle\left(x_j^n - \hat{x}_j^n - y_j\right)^2\rangle\right]^{-1} \tag{69}$$

$$\hat{c}_{\lambda_c} = c_{\lambda_c} + \frac{S}{2}\sum_{n=1}^{N}\hat{\eta}_c^n \tag{70}$$

A.4 ICA Mixture Update

A.4.1 $p'(\boldsymbol{c})$

$$\eta_c^n = \tilde{\kappa}_c \tilde{\lambda}_c^{\frac{S}{2}} \prod_{j=1}^{S} \exp\left[\frac{\lambda_c}{2}\langle\left(x_j^n - \hat{x}_j^n - y_j\right)^2\rangle\right] \tag{71}$$

$$\hat{\eta}_c^n = \frac{\eta_c^n}{\sum_{c=1}^{C} \eta_c^n} \tag{72}$$

where

$$\begin{aligned} \tilde{\kappa}_c &= \exp\left[\Psi(\hat{\iota}_c) - \Psi(\sum_{c'} \hat{\iota}_{c'})\right] \\ \tilde{\lambda}_c &= \hat{b}_{\lambda_c} \exp\left[\Psi(\hat{c}_{\lambda_c})\right] \end{aligned}$$

where (72) ensures that $\sum_c \hat{\eta}_c^n = 1$.

A.4.2 $p'(\boldsymbol{\kappa})$

$$\hat{\iota}_c = \iota_c + \sum_{n=1}^{N} \hat{\eta}_c^n \tag{73}$$

The relevent moments are given by

$$\begin{aligned} \langle a \rangle &= \text{mean}(a) \\ \langle a^2 \rangle &= \text{mean}(a)^2 + \text{variance}(a) \end{aligned} \tag{74}$$

Note that for a mixture with only one ICA component, $\eta_c^n = 1$ for all n and equations (54)-(66) reduce to the update equations for a single ICA model.

Acknowledgements

The authors would like to thank Iead Rezek, Peter Sykacek and Evangelos Roussos for perceptive comments and discussions. Thanks to Mike Gibbs for solving computer related problems. Thanks also to the anonymous referees for very helpful and constructive criticism.

References

Attias, H. (1999a). Learning parameters and structure of latent variable models by variational Bayes. In *Electronic proceedings of the fifteenth annual conference on uncertainty in artificial intelligence (uai-1999).* http://www2.sis.pitt.edu/~dsl/UAI/uai.html.

Attias, H. (1999b). Independent Factor Analysis. *Neural Computation*, *11*, 803-851.

Back, A., & Weigend, A. (1998). A first application of Independent Component Analysis to extracting structure from stock returns. *International Journal of Neural Systems*, *8*(4), 473-484.

Bartlett, M., Lades, H., & Sejnowski, T. (1998). Independent components representations for face recognition. In *Proceedings of the SPIE symposium on electronic imaging: Science and technology: Conference on human vision and electronic imaging III* (p. 528-539).

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

Bell, A., & Sejnowski, T. (1997). The "independent components" of natural scenes are edge filters. *Vision Research*, *37*(23), 3327-3338.

Cardoso, J.-F. (1997). Infomax and maximum liklihood for blind source separation. *IEEE Letters on Signal Processing*, *4*, 112-114.

Choudrey, R., Penny, W., & Roberts, S. (2000). An ensemble learning approach to Independent Component Analysis. In *Proceedings of neural networks for signal processing x, sydney, december 2000.*

Choudrey, R., & Roberts, S. (2001). *Variational Bayesian Independent Component Analysis with flexible sources* (Tech. Rep. No. PARG-01-05). http://www.robots.ox.ac.uk/ sjrob/pubs.html: University of Oxford.

Cooper, G. (1990). Computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, *42*(3), 393-405.

Fisher, R. (1936). The use of multiple measurements in taxonomic problem. *Annals of Eugenics*, *7*(2), 179-188.

Ghahramani, Z., & Beal, M. (2000). Variational inference for Bayesian mixtures of factor analysers. In *Advances in neural information processing systems* (Vol. 12, p. 449-455).

Girolami, M. (1998). An alternative perspective on adaptive Independent Component Analysis algorithms. *Neural Computation*, *10*(8), 2103-2114.

Hateren, J. van, & Schaaf, A. van der. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings of the Royal Society of London, series B*, *265*, 359-366.

Herault, J., & Jutten, C. (1986). Space or time adaptive signal processing by neural models. In J. Denker (Ed.), *Proceedings aip conference: Neural networks for computing* (Vol. 151, p. 206-211).

Hinton, G. (1989). Connectionist learning procedures. *Artificial Intelligence*, *40*, 185-234.

Hinton, G., & Camp, D. van. (1993). Keeping neural networks simple by minimising the description length of the weights. In *Proceedings of colt-93.*

Hyvärinen, A. (1999). Survey on Independent Component Analysis. *Neural Computing Surveys*, *2*, 94-128. (Available from http://www.icsi.berkeley.edu/ jagota/NCS)

Isbell, B., & Viola, P. (1999). Restructuring sparse high dimensional data for effective retrieval. In M. Kearns, S. Solla, & D. Cohn (Eds.), *Advances in neural information processing systems* (Vol. 11, p. 480-486).

Jaakkola, T., & Jordan, M. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, *10*, 25-37.

Jordan, M., Ghahramani, Z., Jaakkola, T., & Saul, L. (1999). An introduction to variational methods for graphical models. In M. Jordan (Ed.), *Learning in graphical models.* Cambridge, Massachusetts: The MIT Press.

Kolenda, T., Hansen, L., & Sigurdsson, S. (2000). Independent Component Analysis in text. In M. Girolami (Ed.), *Advances in Independent Component Analysis.* Springer.

Landaur, T., Foltz, P., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, *25*, 295-284.

Lappalainen, H. (1999). Ensemble learning for Independent Component Analysis. In *Proceedings of the first international workshop on independent component analyis* (p. 7-12).

Lee, T., Girolami, M., Bell, A., & Sejnowski, T. (1998). A unifying information-theoretic framework for Independent Component Analysis. *International Journal of Mathematical and Computer Modelling.*

Lee, T., Girolami, M., & Sejnowski, T. (1999). Independent Component Analysis using an extendend infomax algorithm for mixed sub-Gaussian and super-Gaussian sources. *Neural Computation*, *11*(2), 409-433.

Lee, T., Lewicki, M., & Sejnowski, T. (1999). ICA mixture models for unsupervised classification and automatic context switching. In *International workshop on Independent Component Analysis* (p. 209-214).

Lee, T., Lewicki, M., & Sejnowski, T. (2000). ICA mixture models for unsupervised classification of non-Gaussian classes and automatic context switching in blind signal seperation. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, *22*(10).

Lewicki, M., & Olshausen, B. (1999). A probabilistic framework for the adaption and comparison of image codes. *Journal of the Otical Society of America*, *16*(7), 1587-1601.

MacKay, D. (1995a). Developments in probabilistic modelling with neural networks - ensemble learning. In *Proceedings of the third annual symposium on neural networks* (p. 191-198). Nijmegen, The Netherlands: Springer.

MacKay, D. (1995b). Probable networks and plausible predictions - a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, *6*, 469-505.

Makeig, S., Bell, A., Jung, T.-P., & Sejnowski, T. (1996). Independent Component Analysis for electroencephalographic data. In D. Touretzky, M. Mozer, & M. Hasselmo (Eds.), *Advances in neural information processing systems* (Vol. 8, p. 145-151).

Miskin, J., & MacKay, D. (2000). Application of ensemble learning to infra-red imaging. In *Proceedings of ica2000.*

Olshausen, B., & Field, D. (1997). Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, *37*, 3311-3325.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems* (2 ed.). San Mateo, CA: Morgan Kaufmann.

Pearlmutter, B., & Parra, L. (1996). A context-sensitive generalization of ICA. In *Iconip '96* (p. 151-157).

Penny, W., Everson, R., & Roberts, S. (2000). Advances in independent components analysis. Kluwer Academic Publishers. (Ed. Mark Girolami)

Penny, W., & Roberts, S. (2001). Mixtures of Independent Component Analysers. In *Artificial neural networks - icann2001* (p. 527-534).

Penny, W., Roberts, S., & Everson, R. (2001). ICA: model order selection and dynamic source models. In S. Roberts & R. Everson (Eds.), *Independent Component Analysis: Principles and practice* (p. 299-314). Cambridge University Press.

Rissanen, J. (1994). Mdl modeling - an introduction. In P. Grassberger & J.-P. Nadal (Eds.), *From statistical physics to statistical inference and back* (p. 95-104). Kluwer Academic.

Ristaniemi, T., & Joutsensalo, J. (1999). On the performance of blins source seperation in cdma downlink. In *Proceedings of ICA '99* (p. 437-442).

Roberts, S., & Everson, R. (Eds.). (2001). *Independent Component Analysis: Principles and practice.* Cambridge University Press.

Roberts, S., Holmes, C., & Denison, D. (2001). Minimum entropy data partitioning using reversible jump Markov chain Monte carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(8), 909-915.

Tipping, M., & Bishop, C. (1999). Mixtures of probabilistic principal component analyzers. *Neural Computation*, *11*(2), 443-482.

Ueda, N., Nakano, R., Ghahramani, Z., & Hinton, G. (1999). Smem algorithm for mixture models. *Advances in Neural Information Processing Systems*, *11*.

Valpola, H., & Pajunen, P. (2000). Fast algorithms for Bayesian Independent Component Analysis. In *Proceedings of ica2000* (p. 233-237). Helsinki, Finland.