AN ENSEMBLE LEARNING APPROACH TO INDEPENDENT COMPONENT ANALYSIS

R. Choudrey, W.D. Penny and S.J. Roberts
Department of Engineering Science
University of Oxford
E-mail: riz,wpenny,sjrob@robots.ox.ac.uk

Abstract. Independent Component Analysis (ICA) is an important tool for extracting structure from data. ICA is traditionally performed under a maximum likelihood scheme in a latent variable model and in the absence of noise. Although extensively utilised, maximum likelihood estimation has well known drawbacks such as overfitting and sensitivity to local-maxima. In this paper, we propose a Bayesian learning scheme, *Variational Bayes* or *Ensemble Learning*, for both latent variables and parameters in the model. We extend current research in this area by utilising a wide variety of priors over model parameters, including noise, and learning the latent distribution as part of the ensemble learning procedure. We demonstrate the model by unmixing a linear mixture of musical signals.

INTRODUCTION

Independent Component Analysis (ICA) seeks to extract salient features and structure from a dataset where the dataset is assumed to be a linear mixture of independent underlying (hidden) features. The goal of ICA is to 'unmix' the dataset and recover these features. In the signal processing literature, this is referred to as Blind Source Separation (BSS). Signals are received at a number of sensors (the data) and are known to be a noisy mixture of a number of independent sources (the features). A BSS flavoured review of the literature can be found in the comprehensive [10], while [1], [7] and [20] approach the problem from a more statistical direction.

In this paper, we extend the work of [1], [9], [4] and [19]. ICA has traditionally been performed in the noise-less limit [3], [10], with noise often being dealt with as an extra source. Recently, however, Attias [1] extended ICA and incorporated full covariance noise into the ICA framework. The model, dubbed by Attias as Independent Factor Analysis (IFA), was subsequently learnt through a maximum likelihood EM algorithm.

Lappalainen introduces an ensemble learning formalism for ICA in [9], where the posterior over the 'ensemble' of hidden variables and parameters is approximated. The priors over the model parameters are all Gaussians and the approximating distribution to the posterior is itself taken to be a Gaussian. This simplifies the learning process but may be limited if the posterior is not well approximated by a Gaussian.

A similar formalism is used in [4] but where a richer variety of functional forms for the priors is used. In addition to this, an extra prior is placed over the variances in the mixing matrix prior in an attempt to automatically determine the numer of hidden sources, a practice known as Automatic Relevance Determination (ARD)[13]. No functional form for the approximating distribution to the posterior is assumed - the forms for the posteriors express themselves in the optimisation procedure. Crucially, however, the source model used in [4] is kept fixed and only the parameters of the sensor model are learnt. Therefore, if the source model is not known, or not chosen correctly, an incorrect and ill-fitting model will be learnt.

We take a similar approach to [4], but with a source model that is learnt from the data. In line with both [4] and [9], we choose a factorial Mixture of Gaussians as our source model and limit the noise covariance to diagonal form. An ensemble learning approach to a single 1-dimensional Mixture of Gaussians is explored in [19]. In this paper, we combine the work of [1], [9], [4] with [19] to produce a full ensemble learning formalism fo ICA.

THE MODEL

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

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{u} \tag{1}$$

where $\boldsymbol{H}$ is an $S \times L$ *mixing matrix* and $\boldsymbol{u}$ is S-dimensional additive noise. In the BSS nomenclature, S is the number of (observed) sensors and L is the number of (hidden) sources.

The noise is assumed to be Gaussian, with zero mean and diagonal precision matrix $\boldsymbol{\Lambda}$

$$p(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{u}; \mathbf{0}, \boldsymbol{\Lambda}) \tag{2}$$

Since the sources $\boldsymbol{x} = \{x_1, .., x_i, .., x_L\}$ are mutually independent, the distribution over $\boldsymbol{x}$ for data point n can be written as

$$p(\boldsymbol{x}^n) = \prod_{i=1}^{L} p(x_i^n) \tag{3}$$

where the product runs over the L sources.

The probability of observing data vector $\boldsymbol{y}^n$ given our generative model is then

$$p(\boldsymbol{y}^n|\boldsymbol{x}^n, \boldsymbol{H}, \boldsymbol{\Lambda}) = \left|\det(\frac{1}{2\pi}\boldsymbol{\Lambda})\right|^{\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{y}^n - \boldsymbol{H}\boldsymbol{x}^n)^T \boldsymbol{\Lambda}(\boldsymbol{y}^n - \boldsymbol{H}\boldsymbol{x}^n)\right] \quad (4)$$

In BSS, one attempts to uncover the hidden source signals that give rise to a set of observed sensor signals. In principle, this is achieved by calculating the posterior over the latent variables (sources) given the observed variables (sensor signals) and the model

$$p(\boldsymbol{x}^n|\boldsymbol{y}^n, \mathcal{M}) = \frac{p(\boldsymbol{y}^n|\boldsymbol{x}^n, \mathcal{M})p(\boldsymbol{x}^n|\mathcal{M})}{p(\boldsymbol{y}^n|\mathcal{M})} \quad (5)$$

where $p(\boldsymbol{x}^n|\mathcal{M})$ is the source model and $p(\boldsymbol{y}^n|\mathcal{M})$ is a normalising factor often called the marginal likelihood, or *evidence* [11] for model $\mathcal{M}$.

Source Model

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

One such distribution used by [1] and [4] is a factorised mixture of Gaussians (MoG) with L factors (i.e. sources) and m_i components per source:

$$\begin{aligned} p(\boldsymbol{x}^n|\boldsymbol{\theta}) &= \prod_{i=1}^{L} \sum_{q_i=1}^{m_i} p(q_i^n = q_i|\boldsymbol{\pi}_i)p(x_i^n|q_i, \mu_{i,q_i}, \beta_{i,q_i}) \quad (6) \\ &= \prod_{i=1}^{L} \sum_{q_i=1}^{m_i} \pi_{i,q_i}\mathcal{N}(x_i^n; \mu_{i,q_i}, \beta_{i,q_i}) \end{aligned}$$

where π_{i,q_i} are the mixing proportions and are equivalent to $p(q_i^n = q_i|\boldsymbol{\pi}_i)$, the prior probability of choosing component q_i of the i^{th} source. The mean and precision of component q_i in source i is μ_{i,q_i} and β_{i,q_i} respectively. The vectors of component parameters can be written as $\boldsymbol{\pi}_i = [\pi_1, \pi_2, \ldots, \pi_{m_i}]$, $\boldsymbol{\mu}_i = [\mu_1, \mu_2, \ldots, \mu_{m_i}]$ and $\boldsymbol{\beta}_i = [\beta_1, \beta_2, \ldots, \beta_{m_i}]$. The parameters of source i are $\boldsymbol{\theta}_i = \{\boldsymbol{\pi}_i, \boldsymbol{\mu}_i, \boldsymbol{\beta}_i\}$ and the complete parameter set of the source model is $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_L\}$. The variable q_i is an indicator variable indicating which component of the i^{th} source is chosen for generating x_i and takes on values of $\{q_i = 1, q_i = 2, \ldots, q_i = m_i\}$. The complete collection of possible source states is denoted $\boldsymbol{q} = \{\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_{\boldsymbol{m}}\}$ and runs over all $\boldsymbol{m} = \prod_i m_i$ possible combinations of source states. The probability of state $\boldsymbol{q}^n$ being chosen and generating source vector $\boldsymbol{x}^n$ is

$$\begin{aligned} p(\boldsymbol{x}^n, \boldsymbol{q}^n|\boldsymbol{\theta}) &= \prod_{i=1}^{L} p(q_i^n = q_i|\boldsymbol{\pi}_i)p(x_i^n|q_i, \mu_{i,q_i}, \beta_{i,q_i}) \quad (7) \\ &= p(\boldsymbol{q}^n|\boldsymbol{\pi})p(\boldsymbol{x}^n|\boldsymbol{q}^n, \boldsymbol{\theta}) \end{aligned}$$

where $\boldsymbol{\pi} = \{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_L\}$. Note that the product of L 1-dimensional MoGs in (6) is equivalent to a single MoG in L-dimensional space with $\boldsymbol{m}$ states.

The likelihood of the IID data $\boldsymbol{D} = \{\boldsymbol{y}^1, \boldsymbol{y}^2, \ldots, \boldsymbol{y}^N\}$ given the model parameters can now be written as

$$p(\boldsymbol{D}|\boldsymbol{H}, \boldsymbol{\Lambda}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \sum_{\boldsymbol{q} = \boldsymbol{1}}^{\boldsymbol{m}} \int p(\boldsymbol{y}^n|\boldsymbol{x}^n, \boldsymbol{H}, \boldsymbol{\Lambda}) p(\boldsymbol{x}^n|\boldsymbol{q}, \boldsymbol{\theta}) p(\boldsymbol{q}^n|\boldsymbol{\pi}) d\boldsymbol{x} \tag{8}$$

where $d\boldsymbol{x} = \prod_i dx_i$.

The parameters $\{\boldsymbol{H}, \boldsymbol{\Lambda}, \boldsymbol{\theta}\}$ of the model can be learnt through a maximum likelihood approach such as the Expectation-Maximisation (EM) algorithm ([6], [15])(see [1] for a comprehensive derivation of the EM algorithm with regard to ICA/IFA). The resultant values can then be used to reconstruct the sources via (5).

BAYESIAN INFERENCE AND ENSEMBLE LEARNING

The maximum likelihood approach to learning the parameters of the model is well documented (see [18], [5], [1] for an introduction), as are the pitfalls. Therefore, we will follow the approach taken in [4] and infer the posterior distributions over parameters $\{\boldsymbol{H}, \boldsymbol{\Lambda}\}$ of the model. In addition to [4], we will also infer the parameters, $\boldsymbol{\theta}$, of the factorised source distribution, $p(\boldsymbol{x})$. First, we will state the distribution over the hidden variables and the priors over the parameters.

The Priors

Because of source independence, it follows that the distribution over the MoG component indicator variables is

$$p(\boldsymbol{q}|\boldsymbol{\pi}) = \prod_{n=1}^{N} \prod_{i=1}^{L} \pi_{i,q_i} \tag{9}$$

For a given set of components, the distribution over the sources is

$$p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \prod_{i=1}^{L} \mathcal{N}(x_i^n; \mu_{i,q_i}, \beta_{i,q_i}) \tag{10}$$

Now we set priors over the parameters in question. The prior over the source model parameters is

$$p(\boldsymbol{\theta}) = p(\boldsymbol{\pi}) p(\boldsymbol{\mu}) p(\boldsymbol{\beta}) \tag{11}$$

where $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_L\}$ and $\boldsymbol{\beta} = \{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_L\}$. The prior over the mixing proportions is a symmetric Dirichlet

$$p(\boldsymbol{\pi}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \frac{\Gamma(m_i \lambda_{i0})}{\Gamma(\lambda_{i0})^{m_i}} \pi_{i,q_i}^{\lambda_{i0}-1} \tag{12}$$

the prior over the means is a Gaussian

$$p(\boldsymbol{\mu}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \mathcal{N}(\mu_{i,q_i}; m_{i0}, \tau_{i0}) \tag{13}$$

and the prior over the precisions is a Gamma

$$p(\boldsymbol{\beta}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \Gamma(\beta_{i,q_i}; b_{i0}, c_{i0}) \tag{14}$$

where we define the Gamma distribution as

$$\Gamma(w; b, c) = \frac{1}{\Gamma(c)} \frac{w^{c-1}}{b^c} \exp\left(-\frac{w}{b}\right) \tag{15}$$

The prior over the sensor noise precision, $\boldsymbol{\Lambda}$, is a Gamma

$$p(\boldsymbol{\Lambda}) = \prod_{j=1}^{S} \Gamma(\lambda_j; b_{\Lambda_j}, c_{\Lambda_j}) \tag{16}$$

and the prior over the mixing matrix is a Gaussian and in the same form as [4].

$$p(\boldsymbol{H}) = \prod_{i=1}^{L} \prod_{j=1}^{S} \mathcal{N}(h_{ji}|0, \alpha_i) \tag{17}$$

Each of the L columns of $\boldsymbol{H}$ has a precision α_i By monitoring the evolution of the precisions $\boldsymbol{\alpha}$, the number of relevent sources may be determined, a practice known as Automatic Relevance Determination (ARD) ([13], [16]). Finally, the prior over $\boldsymbol{\alpha}$ is a Gamma

$$p(\boldsymbol{\alpha}) = \prod_{i=1}^{L} \Gamma(\alpha_i; b_{\alpha_i}, c_{\alpha_i}) \tag{18}$$

Figure 1 shows the graphical model for our formalism.

Bayesian inference in such a model is computationally intensive and often intractable. An important tool in approximating posterior distributions is the *variational method* (see [8] for an excellent tutorial). In particular, we take the *varitional Bayes* or *ensemble learning* approach detailed in [12] and used in [4].

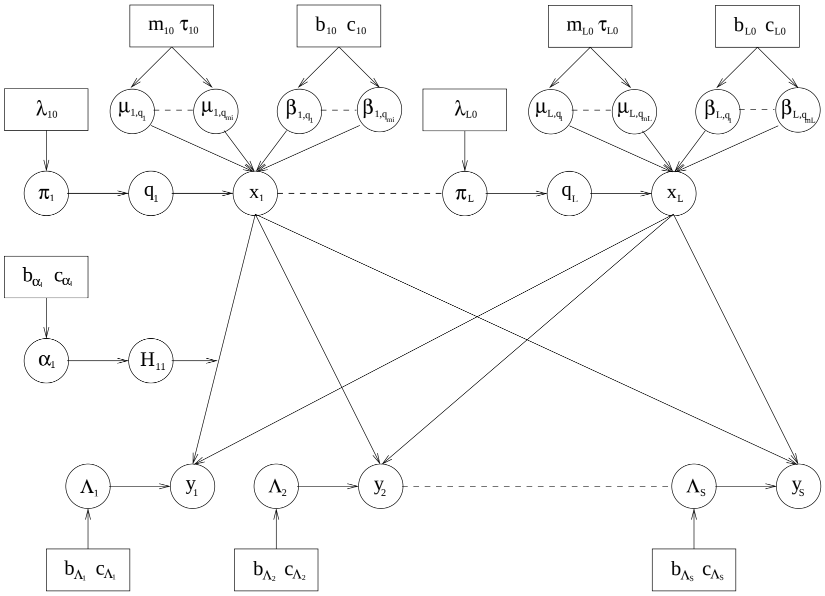


Figure 1: Bayesian Independent Component Analysis as a Graphical Model. Circles represent random variables and rectangles represent hyper-parameters

Ensemble Learning

In the ensemble learning framework, the objective function to be maximised is the *negative free energy*, F

$$F = \langle \log p(\boldsymbol{D}, \boldsymbol{W}) \rangle_{p'(\boldsymbol{W})} + \mathcal{H}\left[p'(\boldsymbol{W})\right] \tag{19}$$

where $\boldsymbol{W} = \{\boldsymbol{\Lambda}, \boldsymbol{H}, \boldsymbol{\alpha}, \boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta}, \}$, the ensemble of hidden variables. The negative free energy forms a strict lower bound on the evidence of the model, with the difference being the Kullback-Leibler (KL) divergence between the true and approximating posteriors. Maximising this function is equivalent to minimising the KL divergence between the true and approximate posteriors. The first term in (19) is the expectation of the joint density of hidden and observed variables with respect to the approximating posterior $p'(\boldsymbol{W})$. The second term is the entropy of $p'(\boldsymbol{W})$

$$\mathcal{H}\left[p'(\boldsymbol{W})\right] = -\int p'(\boldsymbol{W}) \log p'(\boldsymbol{W}) \tag{20}$$

By choosing $p'(\boldsymbol{W})$ such that it factorises over the ensemble of hidden variables, $\boldsymbol{W}$, terms in each hidden variable can be maximised individually

$$p'(\boldsymbol{W}) = p'(\boldsymbol{\Lambda})p'(\boldsymbol{H})p'(\boldsymbol{\alpha})p'(\boldsymbol{x})p'(\boldsymbol{q})p'(\boldsymbol{\theta}) \tag{21}$$

where $p'(\boldsymbol{\theta}) = p'(\boldsymbol{\pi})p'(\boldsymbol{\mu})p'(\boldsymbol{\beta})$. We will also take the approach of [1], and stipulate that the posteriors over the sources factorise such that

$$p'(\boldsymbol{x}) = \prod_{i=1}^{L} p'(x_i) \tag{22}$$

and therefore

$$p'(\boldsymbol{H}) = \prod_{i=1}^{L} p'(\boldsymbol{H}_i) \tag{23}$$

where $\boldsymbol{H}_i$ is the i^{th} column of the mixing matrix, $\boldsymbol{H}$. This additional factorisation allows efficient scaling of computation with the number of hidden sources, with little loss of accuracy [1].

One may now proceed by specifying functional forms of each of the approximating posteriors and using these in (19) as shown by [9]. As shown in [12], however, there is no need to specify functional forms for the approximating posteriors as they 'fall-out' of the maximisation process.

By substituting $p(\boldsymbol{D}, \boldsymbol{W})$ and (21) into (19), we can obtain an expression for the negative free energy of our model. The term $p(\boldsymbol{D}, \boldsymbol{W})$ is straight forward to write down

$$p(\boldsymbol{D}, \boldsymbol{W}) = p(\boldsymbol{D}|\boldsymbol{H}, \boldsymbol{\Lambda}, \boldsymbol{x})p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\mu}, \boldsymbol{\beta})p(\boldsymbol{q}|\boldsymbol{\pi})p(\boldsymbol{\theta})p(\boldsymbol{H}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})p(\boldsymbol{\Lambda}) \tag{24}$$

The negative free energy for our model is now

$$\begin{aligned} F \quad = \quad & \langle \log p(\boldsymbol{D}|\boldsymbol{H}, \boldsymbol{\Lambda}, \boldsymbol{x}) \rangle_{p'(\boldsymbol{\Lambda})p'(\boldsymbol{H})p'(\boldsymbol{x})} \\ & + \langle \log p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\mu}, \boldsymbol{\beta}) \rangle_{p'(\boldsymbol{x})p'(\boldsymbol{q})p'(\boldsymbol{\mu})p'(\boldsymbol{\beta})} + \mathcal{H}[p'(\boldsymbol{x})] \\ & + \langle \log p(\boldsymbol{q}|\boldsymbol{\pi}) \rangle_{p'(\boldsymbol{q})p'(\boldsymbol{\pi})} + \mathcal{H}[p'(\boldsymbol{q})] \\ & + \langle \log p(\boldsymbol{\pi}) \rangle_{p'(\boldsymbol{\pi})} + \mathcal{H}[p'(\boldsymbol{\pi})] \\ & + \langle \log p(\boldsymbol{\mu}) \rangle_{p'(\boldsymbol{\mu})} + \mathcal{H}[p'(\boldsymbol{\mu})] \\ & + \langle \log p(\boldsymbol{\beta}) \rangle_{p'(\boldsymbol{\beta})} + \mathcal{H}[p'(\boldsymbol{\beta})] \\ & + \langle \log p(\boldsymbol{H}|\boldsymbol{\alpha}) \rangle_{p'(\boldsymbol{H})p'(\boldsymbol{\alpha})} + \mathcal{H}[p'(\boldsymbol{H})] \\ & + \langle \log p(\boldsymbol{\alpha}) \rangle_{p'(\boldsymbol{\alpha})} + \mathcal{H}[p'(\boldsymbol{\alpha})] \\ & + \langle \log p(\boldsymbol{\Lambda}) \rangle_{p'(\boldsymbol{\Lambda})} + \mathcal{H}[p'(\boldsymbol{\Lambda})] \end{aligned} \tag{25}$$

The Posteriors

Maximising (25) yields the following analytic results

$$p'(\boldsymbol{q}) \quad = \quad \prod_{n=1}^{N} \prod_{i=1}^{L} \hat{\gamma}_{i,q_i}^{n} \tag{26}$$

$$p'(\boldsymbol{x}) \quad = \quad \prod_{n=1}^{N} \prod_{i=1}^{L} \mathcal{N}(x_i^n; \hat{\mu}_i^{(n)}, \hat{\beta}_i^{(n)}) \tag{27}$$

$$p'(\boldsymbol{\pi}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \frac{\Gamma(m_i \hat{\lambda}_{i,q_i})}{\Gamma(\hat{\lambda}_{i,q_i})^{m_i}} \pi_{i,q_i}^{\hat{\lambda}_{i,q_i}-1} \tag{28}$$

$$p'(\boldsymbol{\mu}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \mathcal{N}(\mu_{i,q_i}; \hat{m}_{i,q_i}, \hat{\tau}_{i,q_i}) \tag{29}$$

$$p'(\boldsymbol{\beta}) = \prod_{i=1}^{L} \prod_{q_i=1}^{m_i} \Gamma(\beta_{i,q_i}; \hat{b}_{i,q_i}, \hat{c}_{i,q_i}) \tag{30}$$

$$p'(\boldsymbol{H}) = \prod_{i=1}^{L} \prod_{j=1}^{S} \mathcal{N}(h_{ji}; \hat{m}_{h_{ji}}, \hat{\alpha}_{ji}) \tag{31}$$

$$p'(\boldsymbol{\alpha}) = \prod_{i=1}^{L} \Gamma(\alpha_i; \hat{b}_{\alpha_i}, \hat{c}_{\alpha_i}) \tag{32}$$

$$p'(\boldsymbol{\Lambda}) = \prod_{j=1}^{S} \Gamma(\lambda_j; \hat{b}_{\Lambda_j}, \hat{c}_{\Lambda_j}) \tag{33}$$

Although no functional forms for the approximating posteriors were assumed, the resultant posteriors are conjugate with the priors. Intuitively, the (approximate) posterior is equivalent to shifting the prior to a position which satifies both the demands of the data and the constraints of the prior. The parameters of the posteriors are updated 'versions' of the parameters of the priors. The update equations are detailed below[1]

- p'($\boldsymbol{q}$)

$$\gamma_{i,q_i}^{n} = \tilde{\pi}_{i,q_i} \tilde{\beta}_{i,qi} \exp\left[-\frac{\langle \beta_{i,q_i} \rangle}{2} \left(\langle x_i^{n\,2} \rangle - 2\langle x_i^n \rangle \langle \mu_{i,q_i} \rangle + \langle \mu_{i,q_i}^2 \rangle \right)\right] \tag{34}$$

$$\hat{\gamma}_{i,q_i}^{n} = \frac{\gamma_{i,q_i}^{n}}{\sum_{q_i'} \gamma_{i,q_i'}^{n}} \tag{35}$$

where

$$\tilde{\pi}_{i,q_i} = \exp\left[\Psi(\hat{\lambda}_{i,q_i}) - \Psi(\sum_{q_i'} \hat{\lambda}_{i,q_i'})\right] \tag{36}$$

$$\tilde{\beta}_{i,qi} = \hat{b}_{i,q_i} \exp\left[\Psi(\hat{c}_{i,q_i})\right] \tag{37}$$

and where (35) ensures that $\sum_{q_i} \gamma_{i,q_i}^{n} = 1$. Define the following variables

$$\bar{\pi}_{i,q_i} = \frac{1}{N} \sum_{n=1}^{N} \gamma_{i,q_i}^{n} \tag{38}$$

$$\bar{N}_{i,q_i} = N \bar{\pi}_{i,q_i} \tag{39}$$

[1]The term $\langle a \rangle$ is the expectation of a with respect to $p'(a)$

$$\bar{x}_{i,q_i} = \frac{1}{N}\sum_{n=1}^{N}\gamma^n_{i,q_i}\langle x_i^n\rangle \tag{40}$$

$$\tilde{x}^2_{i,q_i} = \frac{1}{N}\sum_{n=1}^{N}\gamma^n_{i,q_i}\langle x_i^{n\,2}\rangle \tag{41}$$

where $\bar{\pi}_{i,q_i}$ is the proportion of source i contributed by component q_i and $\bar{N}_{i,q_i}$ is the number of outputs of source i contributed by component q_i. The quantity $\bar{x}_{i,q_i}$ is the contribution of component q_i to the average of source i while $\tilde{x}^2_{i,q_i}$ is the contribution of component q_i to the second moment of source i.

- p'($\boldsymbol{x}$)

$$\hat{\mu}_i^{(n)} = \frac{1}{\hat{\beta}_i^{(n)}}\left[\bar{\mu}_i + \sum_{j=1}^{S}\langle\Lambda_j\rangle\langle h_{ji}\rangle\left(y_j^n - \hat{y}^n_{j,k\neq i}\right)\right] \tag{42}$$

$$\hat{\beta}_i^{(n)} = \bar{\beta}_i + \sum_{j=1}^{S}\langle\Lambda_j\rangle\langle h^2_{ji}\rangle \tag{43}$$

where

$$\bar{\mu}_i = \sum_{q_i=1}^{m_i}\hat{\gamma}^n_{i,q_i}\langle\beta_{i,q_i}\rangle\langle\mu_{i,q_i}\rangle \tag{44}$$

$$\bar{\beta}_i = \sum_{q_1=1}^{m_i}\hat{\gamma}^n_{i,q_i}\langle\beta_{i,q_i}\rangle \tag{45}$$

$$\hat{y}^n_{j,k\neq i} = \sum_{k\neq i}^{L}\langle h_{jk}\rangle\langle x_k^n\rangle \tag{46}$$

$$\hat{y}^n_j = \sum_{i=1}^{L}\langle h_{ji}\rangle\langle x_i^n\rangle \tag{47}$$

Note the intuitive form of (42) and (43). Source i 'sees' data y_j^n at sensor j and works out what it can 'explain away' given information in the rest of the model, i.e. $\hat{y}^n_{j,k\neq i}$. The residual information is then used to update its own parameters in a bid to explain what's 'left over'. This is analogous to Pearl's message passing algorithms [17], whereby a node (x_i) updates its belief (encapsulated in $\hat{\mu}_i$ and $\hat{\beta}_i$) using messages passed to it by it's parents (quantified by $\bar{\mu}_i$ and $\bar{\beta}_i$), its children (the sensors) and all other parents of it's children (i.e. all other sources and quantified by $\hat{y}_{j,k\neq i}$). This use of information from a variable's Markov Blanket runs through all the update equations.

- p'($\boldsymbol{\pi}$)

$$\hat{\lambda}_{i,q_i} = \lambda_{i0} + \bar{N}_{i,q_i} \tag{48}$$

- p'($\boldsymbol{\mu}$)

$$\hat{m}_{i,q_i} = \frac{\tau_{i0}}{\hat{\tau}_{i,q_i}} m_{i0} + \frac{\tau_{x_i}}{\hat{\tau}_{i,q_i}} m_{x_i} \tag{49}$$

$$\hat{\tau}_{i,q_i} = \tau_{i0} + \tau_{x_i} \tag{50}$$

where

$$m_{x_i} = \frac{\bar{x}_{i,q_i}}{\bar{\pi}_{i,q_i}} \tag{51}$$

$$\tau_{x_i} = \langle \beta_{i,q_i} \rangle \bar{N}_{i,q_i} \tag{52}$$

- p'($\boldsymbol{\beta}$)

$$\hat{b}_{i,q_i} = \left(\frac{1}{b_{i0}} + \frac{N}{2} \tilde{\sigma}_{i,q_i} \right)^{-1} \tag{53}$$

$$\hat{c}_{i,q_i} = c_{i0} + \frac{\bar{N}_{i,q_i}}{2} \tag{54}$$

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

$$\tilde{\sigma}_{i,q_i} = \tilde{x}^2_{i,q_i} - 2\langle \mu_{i,q_i} \rangle \bar{x}_{i,q_i} + \bar{\pi}_{i,q_i} \langle \mu^2_{i,q_i} \rangle \tag{55}$$

- p'($\boldsymbol{H}$)

$$\hat{m}_{h_{ji}} = \frac{\langle \Lambda_j \rangle}{\hat{\alpha}_{ji}} \sum_{n=1}^{N} \langle x_i^n \rangle \left[y_j^n - \hat{y}^n_{j,k \neq i} \right] \tag{56}$$

$$\hat{\alpha}_{ji} = \langle \alpha_i \rangle + \langle \Lambda_j \rangle \sum_{n=1}^{N} \langle x_i^{n\,2} \rangle \tag{57}$$

For non-isotropic noise, the precisions of the mixing matrix $\mathbf{H}$ are no longer column-independent. This suggests that ARD will only work with isotropic or near-isotropic noise.

- p'($\boldsymbol{\alpha}$)

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

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

- p'($\boldsymbol{\Lambda}$)

$$\hat{b}_{\Lambda_j} = \left(\frac{1}{b_{\Lambda_j}} + \frac{1}{2} \sum_{n=1}^{N} \left[y_j^n - \hat{y}_j^n \right]^2 \right)^{-1} \tag{60}$$

$$\hat{c}_{\Lambda_j} = c_{\Lambda_j} + \frac{N}{2} \tag{61}$$

The relevent moments are given by

$$\begin{aligned} \langle a \rangle &= mean(a) \\ \langle a^2 \rangle &= mean(a)^2 + variance(a) \end{aligned} \tag{62}$$

The update equations (34)-(61) are coupled and therefore must be solved iteratively. This is achieved by starting with initial guesses of the variables and cycling through the update equations using the moments calculated until convergence. Once trained, the model can be used to reconstruct hidden sources (to within a scaling and permutation) given a data-set by iterating (34), (42) and (43) until convergence.

RESULTS

We mixed two sources of music sampled at 11kHz into two sets of six signals. We added 10 percent observation noise to one set and left the other noise-free. Each source MoG had 3 components. The hyper-parameters $\boldsymbol{b}_\Lambda$, $\boldsymbol{c}_\Lambda$, $\boldsymbol{b}_\alpha$, $\boldsymbol{c}_\alpha$, b_{i0} and c_{i0} were set to $\boldsymbol{b}_\Lambda = \boldsymbol{b}_\alpha = b_{i0} = 1000$ and $\boldsymbol{c}_\Lambda = \boldsymbol{c}_\alpha = c_{i0} = 10^{-3}$. The mixing matrix, $\boldsymbol{H}$, and starting values of $\langle \boldsymbol{x} \rangle$ were initialised using SVD. The initialised values of $\langle \boldsymbol{x} \rangle$ were fed to the source MoGs in order to initialise their parameters via k-means. The hyper-parameter m_{i0} was set to the mean of $\langle x_i \rangle$ while $\tau_{i0} = 10^{-3}$. The noise precision $\boldsymbol{\Lambda}$ was drawn from it's prior (14).

The optimisation of the distributions then proceeded in a 'see-saw' method [1]. The update equations for the individual source MoG parameters were iterated until convergence. This was followed by calculating $\langle \boldsymbol{x} \rangle$ by cyclically iterating (42) and (43). Finally, (56)-(61) were iterated until convergence. The whole procedure was repeated until the complete ensemble of hidden variables converged.

The quality of the reconstructions were assessed by evaluating the expected square error between the reconstructed and true sources

$$E_i = \frac{1}{N} \sum_{n=1}^{N} (\hat{x}_i^n - x_i^n)^2 \tag{63}$$

where $\hat{x}_i^n$ is the model reconstruction for source i given data vector $\boldsymbol{y}^n$.

Noise-Less ICA

We trained an ICA model with 6 latent dimensions on the zero-noise data-set. Figure 2a shows the 6 reconstructed source signals. All but two of the sources have been effectively 'switched-off' by ARD. Figure 3 is a Hinton diagram of the evaluated mixing matrix, $\boldsymbol{H}$. Of the original 6 specified latent dimensions, only 2 remain and, therefore, the true latent dimesionality has been determined. Figure 4 compares the reconstructed sources to the true sources. The lower-variance reconstruction on the left has an expected square error of 0.0414, while the high-variance source has $E = 0.0964$.

Noisy ICA

We trained a number of ICA models with up to 6 latent dimensions on the noisy data-set. Figure 2b shows the reconstructed source signals when the latent dimensionality is set to six. This time, the 'remaining' sources pick up noise. In this case, ARD is no longer viable as a method for determining the latent dimensionality. As far as the model is concerned, it can explain the data better using extra sources that generate the noise, rather than the noise being generated at the sensors. Indeed, when we inspect the evaluated noise covariance, we find variances of the order of 10^{-6} for all but one of the sensors.

We then trained ICA models with latent dimensionality 2, 3, 4 and 5. As the dimensionality increased, more and more of the 'explanation' for the noise was 'transfered' from the sensor noise covariance to the extra sources. The quality of the reconstruction of the true sources was also seen to vary, with expected square error for 6 sources being lower than that for 2 sources. The error for 3-5 sources, however, showed no discernable trend.

To determine the latent dimensionality in the presence of noise, a method similar to [2] and [19] can be used whereby the free-energy is monitored for a range of latent dimensions.

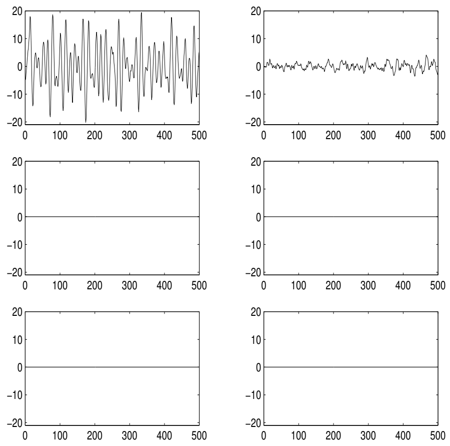
DISCUSSION

The proposed model performed extremely well when using noise-free data. The reconstructed sources were virtually identical to the true sources, to within a scaling and permutation. The true latent dimensionality was correctly determined using ARD. With noisy data, the model reconstructed the sources well, but with a higher expected square error. In this case, ARD did not determine the true number of sources - extra dimensions were used to 'explain' the noise. Model order estimation in this situation can be achieved by monitoring the free-energy for a range of latent dimensions.

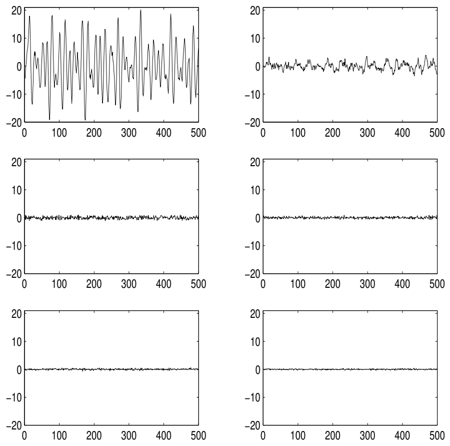
While in preparation, we received a pre-publish copy of [14] in which the above model is derived independently. For a more thorough derivation and evaluation of the model, the reader is strongly encouraged to refer to [14].

Acknowledgements

Thanks to Iead Rezek for his coding and latex skills.



(a) Zero noise case: Recovered signals for a data set with true latent dimensionality 2. All but two of the signals have been ‘switched off’



(b) Observation noise of 10%: Recovered signals for a data set with true latent dimensionality 2. The extra sources have captured the noise

Figure 2: Reconstructed Signals

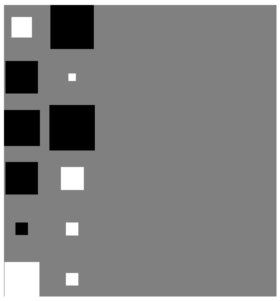


Figure 3: Hinton diagram of the mixing matrix, $\boldsymbol{H}$. All but two of the columns have been surpressed

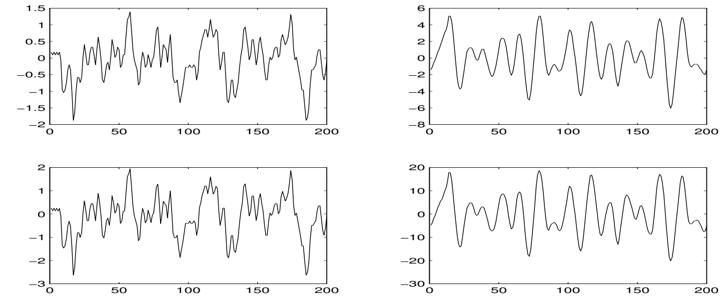


Figure 4: Top: Original source signals Bottom: The recovered signals in the zero noise case

REFERENCES

[1] H. Attias, "Independent Factor Analysis," **Neural Computation**, vol. 11, pp. 803–851, 1998.

[2] H. Attias, "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)**, Association for Uncertainty in AI (AUAI), http://www2.sis.pitt.edu/ dsl/UAI/uai.html, 1999.

[3] A. J. Bell and T. J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," **Neural Computation**, vol. 7, pp. 1129–1159, 1995.

[4] C. M. Bishop and N. D. Lawrence, "Variational Bayesian Independent Component Analysis," Techn. report, **Computer Laboratory, University of Cambridge**, 2000.

[5] Cardoso, "Infomax and maximum liklihood for blind source separation," **IEEE Letters on Signal Processing**, vol. 4, pp. 112–114, 1997.

[6] A. P. Dempster, N. M. Laird and R. D. B., "Maximum liklihood from incomplete data via the EM algorithm," **Journal of the Royal Statistical Society B**, vol. 39, pp. 1–38, 1977.

[7] R. Everson and S. J. Roberts, "Independent Component Analysis: A flexible nonlinearity and decorrelating manifold approach," **Neural Computation**, 1999.

[8] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola and L. K. Saul, "An Introduction To Variational Methods for Graphical Models," in M. I. Jordan (ed.), **Learning in Graphical Models**, Cambridge, Massachusetts: The MIT Press, 1999.

[9] H. Lappalainen, "Ensemble learning for independent component analysis," in **Proceedings of the First International Workshop on Independent Component Analyis**, 1999, pp. 7–12.

[10] T. W. Lee, G. M., A. J. Bell and T. J. Sejnowski, "A Unifying information-theoretic framework for Independent Component Analysis," **International Journal of Mathematical and Computer Modelling (In press)**, 1998.

[11] D. J. C. Mackay, **Bayesian Methods for Adaptive Models**, Ph.D. thesis, California Institute of Technology, 1991.

[12] D. J. C. Mackay, "Developments in Probabilistic Modelling with Neural Networks - Ensemble Learning," in **Proceedings of the third Annual Symposium on Neural Networks**, Nijmagen, The Netherlands: Springer, 1995, pp. 191–198.

[13] D. J. C. Mackay, "Probable Networks and Plausible Predictions - A Review of Practical Bayesian Methods for Supervised Neural Networks," **Network: Computation in Neural Systems**, vol. 6, pp. 469–505, 1995.

[14] J. W. Miskin and D. J. C. Mackay, "Ensemble Learning for Blind Source Seperation," To appear in ?

[15] R. D. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse and other variants," in M. I. Jordan (ed.), **Learning in Graphical Models**, Cambridge, Massachusetts: The MIT Press, 1999.

[16] R. M. Neal, **Bayesian Learning for Neural Networks**, Lecture Notes in Statistics, Springer, 1996.

[17] J. Pearl, **Probabilistic Reasoning in Intelligent Systems**, San Mateo,

CA: Morgan Kaufmann, 2nd edn., 1988 (revised 1992).
[18] B. Pearlmutter and L. Parra, "A context-sensitive generalization of ICA," in **ICONIPS '96**, 1996, pp. 151–157.
[19] W. D. Penny, "Variational Bayes for 1-dimensional Mixture Models," Techn. report, **Engineering Science, University of Oxford**, March 2000.
[20] S. J. Roberts, "Independent component analysis: Source assessment and separation, a Bayesian approach," **IEE Proceedings on Vision, Image, and Signal Processing**, vol. 145, no. 3, pp. 149–154, 1998.