Hierarchical Models of Variance Sources ⋆

Harri Valpola, Markus Harva and Juha Karhunen [1]

Helsinki University of Technology, Neural Networks Research Centre
P.O. Box 5400, FIN-02015 HUT, Espoo, Finland
firstname.lastname@hut.fi http://www.cis.hut.fi/projects/bayes/

Abstract

In many models, variances are assumed to be constant although this assumption is often unrealistic in practice. Joint modelling of means and variances is difficult in many learning approaches, because it can lead into infinite probability densities. We show that a Bayesian variational technique which is sensitive to probability mass instead of density is able to jointly model both variances and means. We consider a model structure where a Gaussian variable, called variance node, controls the variance of another Gaussian variable. Variance nodes make it possible to build hierarchical models for both variances and means. We report experiments with artificial data which demonstrate the ability of the learning algorithm to find variance sources explaining and characterizing well the variances in the multidimensional data. Experiments with biomedical MEG data show that variance sources are present in real-world signals.

1 Introduction

Most unsupervised learning [2] techniques model only changes in the means of different quantities while variances are assumed constant. This assumption is often known to be invalid but suitable techniques for jointly estimating both means and variances have been lacking. The basic problem is that if the mean is modelled by a latent variable model such as independent component analysis (ICA) (Hyvärinen et al., 2001), the modelling error of any single observation

⋆ This work is an extended version of the paper by Valpola et al. (2003a).

[1] This research has been funded by the European Commission project BLISS, and the Finnish Center of Excellence Programme (2000–2005) under the project New Information Processing Principles.

[2] Throughout this paper, we use the terms learning and estimation interchangeably. The former is used in neural networks literature.

can be made zero. If the learning method is based on maximising likelihood or posterior density, it runs into problems when trying to simultaneously estimate the variance as the density will become infinite when the variance approaches zero.

A simple example of the problem is given by factor analysis. Consider the model

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t)\,, \tag{1}$$

where $\mathbf{n}(t)$ is Gaussian noise and the matrix $\mathbf{A}$ maps the factors $\mathbf{s}(t)$ to the observation vectors $\mathbf{x}(t)$. The right-hand side of (1) consists of unknown quantities which are to be estimated. It is reasonable to assume that $\mathbf{n}(t)$ has a diagonal noise covariance since any correlation between the observed signals can be assumed to have been generated by the underlying factors.

A problem arises if one tries to estimate the diagonal elements of the noise covariance, that is the noise level of each individual observed signal, by a method which is sensitive to probability density, such as maximum likelihood or maximum a posteriori estimation. The likelihood of the data can be made infinite by copying one of the signals to one of the factors. The model can then explain that signal perfectly and the noise level of the signal can be set to zero.

In the above case, only as many unknown variance parameters are estimated as there are observed signals. This number is usually far smaller than the number of observation vectors in the temporal dimension. This shows that estimation of a very small number of unknown variances can compromise the estimation of means [3]. In addition, the total number of estimated parameters can be very small compared to the available data. In the factor analysis example, there can be a very large number of observed signals but estimation of a single factor signal combined with the estimation of the variances will still cause the problem.

This paper is motivated by the need to estimate features which would describe well both the means and variances of the observations. Good estimates of variances improve the estimates of the features which describe the means of the observations. The variance of a signal or a set of signals can also carry useful information.

In this paper we show how the problem can be solved by variational Bayesian learning. We are able to jointly estimate both the means and the variances by a hierarchical model because the learning criterion is based on posterior

[3] The converse is also true: the estimation of a small number of unknown means can interfere with the estimation of variances. The problem arises when the model can effectively use one mean and one variance parameter for a single observation.

probability mass [4] rather than on the problematic probability density. The cases mentioned above no longer pose problems because when the variance approaches zero, the posterior probability density will have an increasingly higher but at the same time narrower peak. The narrower peak compensates the higher density, resulting in a well behaving posterior probability mass.

The basic method used here was introduced in the preliminary conference paper by Valpola et al. (2001). The method relies on a set of building blocks that can be used to construct various latent variable models. In this paper we deal with building variance models using Gaussian variables and linear mappings. The new method is computationally of the same order as the well-known maximum likelihood and maximum a posteriori methods, which are based on simpler density estimation approaches.

In Section 2, we introduce the variance node, a Gaussian variable which converts predictions of mean into predictions of variance, and discuss various models which utilise it. Section 3 shows how these models are learned. Experiments where such models are applied to artificial and natural data are reported in Section 4.

2 Variance node

A variance node (Valpola et al., 2001) is a time-dependent Gaussian variable $u(t)$ which specifies the variance of another time-dependent Gaussian variable $\xi(t)$:

$$\xi(t) \sim N(\mu_\xi(t), \exp[-u(t)]) \,, \tag{2}$$

where $N(\mu, \sigma^2)$ is the Gaussian distribution and $\mu_\xi(t)$ is the prediction for the mean of $\xi(t)$ given by other parts of the model. As can be seen from (2), $u(t) = -\log \sigma^2$. This parametrisation is justified in Section 3.3.

Variance nodes are useful as such for modelling super-Gaussian distributions because a Gaussian variable ξ whose variance has fluctuations over time generates a super-Gaussian distribution (see e.g. Parra et al., 2001). Variance nodes alone cannot generate sub-Gaussian distributions [5], but in many cases sub-Gaussian models are not needed. This is particularly true in connection with dynamics. Real signals such as oscillations have sub-Gaussian distributions but their innovation processes are almost invariably super-Gaussian. Fig. 1(a) shows a schematic diagram of a linear ICA model which has super-Gaussian

[4] In this case, the probability mass is the integrated probability density over a volume of the parameter space.

[5] Mixture-of-Gaussian distributions can be used for sub-Gaussian distributions. See e.g. Attias (1999).

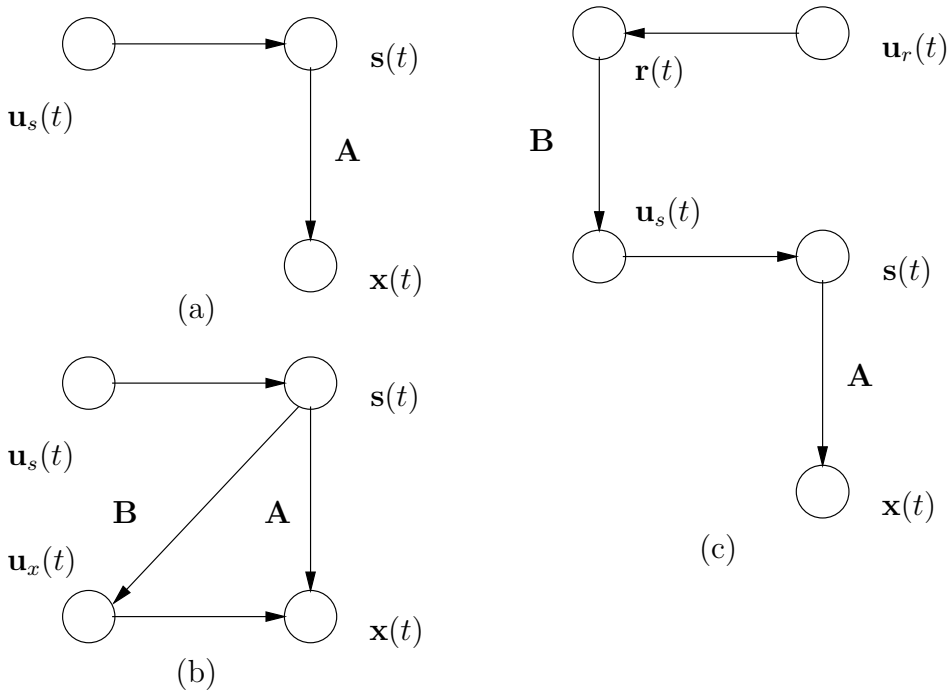

Fig. 1. Various model structures utilising variance nodes. Observations are denoted by $\mathbf{x}$, linear mappings by $\mathbf{A}$ and $\mathbf{B}$, sources by $\mathbf{s}$ and $\mathbf{r}$ and variance nodes by $\mathbf{u}$.

source distributions. These distributions are generated by Gaussian sources $\mathbf{s}$ which have variance nodes $\mathbf{u}_s$ attached to each source.

From the point of view of other parts of the model which predict the value of the variance node, the variance node is as any other Gaussian variable. This means that it enables to translate a conventional model of mean into a model of variance. A simple extension of ICA which utilises variance nodes in this way is shown in Fig. 1(b). The sources can model concurrent changes in both the observations $\mathbf{x}$ and the modelling errors of the observations through the variance nodes $\mathbf{u}_x$. Such a structure would be useful for instance in a case where a source characterises the rotation speed of a machine. It is plausible that the rotation speed affects the mean of a set of variables and the modelling error of another, possibly overlapping set of variables. Note that a model depicted in Fig. 1(b) always has more coefficients to be estimated than there are data. The estimation is nevertheless feasible with variational Bayesian learning.

Linear ICA tries to find a representation where the sources are as independent as possible. In practice the estimated sources will be linearly uncorrelated but some other dependences remain. In particular, the variances of the sources have been found to often correlate in practice. This has motivated the development of models such as subspace ICA (De Lathauwer et al., 1995; Cardoso, 1998; Hyvärinen and Hoyer, 2000; Hyvärinen et al., 2001), where each subset of sources is assumed to have dependences while remaining independent of other subsets. Often the dependence within a subset is modelled in terms of

a common time-dependent latent variable governing the variance among the sources in the subset.

In this paper we present experiments with a hierarchical extension of the linear ICA model, shown in Fig. 1(c). The correlations and concurrent changes in the variances $\mathbf{u}_s(t)$ of conventional sources $\mathbf{s}(t)$ are modelled by higher-order variance sources $\mathbf{r}(t)$. As a special case, this model structure is able to perform subspace ICA. In that case, the variance of each conventional source would be modelled by only one of the variance sources, i.e. the mapping $\mathbf{B}$ would have only one non-zero entry on each row. Moreover, usually each subspace in subspace ICA has an equal dimension, i.e. each column of $\mathbf{B}$ has an equal number of non-zero entries. We are not going to impose such restrictions. The effects of variance sources can thus be overlapping.

Just as conventional sources of time-series data have temporal structure (Hyvärinen et al., 2001), variance sources of such data can be expected to change slowly, in fact, more slowly than the conventional sources. This is because the variance sources have similarity to the invariant features extracted by adaptive subspace SOM (Kohonen et al., 1997) and other related models, (e.g. Hyvärinen and Hoyer, 2000; Hyvärinen and Hurri, 2003). This is demonstrated in the experiment with magnetoencephalographic data in Section 4.

3 Variational Bayesian learning

Variational Bayesian learning techniques are based on approximating the true posterior probability density of the unknown variables of the model by a function with a restricted form. Currently the most common technique is ensemble learning which uses Kullback-Leibler divergence to measure the misfit between the approximation and the true posterior. It has been applied to ICA and a wide variety of other models (see e.g. Hinton and van Camp, 1993; Barber and Bishop, 1998; Attias, 1999; Miskin and MacKay, 2000; Ghahramani and Hinton, 2000; Choudrey et al., 2000; Chan et al., 2001; Valpola and Karhunen, 2002). An example of applying a variational technique other than ensemble learning to linear ICA has been given by Girolami (2001).

3.1 Cost function

In ensemble learning, the posterior approximation $q(\boldsymbol{\theta})$ of the unknown variables $\boldsymbol{\theta}$ is required to have a suitably factorial form

$$q(\boldsymbol{\theta}) = \prod_i q_i(\boldsymbol{\theta}_i)\,, \tag{3}$$

where $\boldsymbol{\theta}_i$ denotes a subset of the unknown variables. The misfit between the true posterior $p(\boldsymbol{\theta} \mid \mathbf{X})$ and its approximation $q(\boldsymbol{\theta})$ is measured by the Kullback-Leibler divergence. An additional term $-\log p(\mathbf{X})$, the negative logarithmic probability of the observations $\mathbf{X}$, is included to avoid calculation of the model evidence term $p(\mathbf{X}) = \int p(\mathbf{X}, \boldsymbol{\theta}) d\boldsymbol{\theta}$. The cost function then has the form (e.g. Barber and Bishop, 1998; Lappalainen and Miskin, 2000; Valpola and Karhunen, 2002)

$$\mathcal{C} = D(q(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta}|\mathbf{X})) - \log p(\mathbf{X}) = \left\langle \log \frac{q(\boldsymbol{\theta})}{p(\mathbf{X}, \boldsymbol{\theta})} \right\rangle , \tag{4}$$

where $\langle \cdot \rangle$ denotes expectation over the distribution $q(\boldsymbol{\theta})$. This shows that ensemble learning can be applied if the joint density $p(\mathbf{X}, \boldsymbol{\theta})$ of all the variables of the model and the posterior approximation $q(\boldsymbol{\theta})$ of the unknown variables can be defined.

Note that since $D(q \parallel p) \geq 0$, the cost function provides a lower bound $p(\mathbf{X}) \geq \exp(-\mathcal{C})$ for the model evidence $p(\mathbf{X})$. This has two important implications. First, the probability density $p(\mathbf{X})$ of the observations is usually finite resulting in a well-behaved cost function. This implies that the model can have any amount of unknown variances and means, but these quantities are marginalised over and the cost function is well behaved if the probability density of the data is finite. Infinite density can result for instance if there are several observations with exactly the same values. This has a probability of zero if the observations are assumed to have continuous values and nonzero variances. In practice numerical rounding or conventions such as replacement of missing values by zeros may have produced identical entries in the observations. This may result in ill-behaved cost function whose value can be infinite. Such cases can normally be avoided by adding small amounts of noise to the data and they are not caused by the underlying model.

The second implication is that cost function can be reliably used for optimising the model structure (see e.g. Valpola et al., 2003b). Since $p(\mathbf{X})$ is a short-hand notation for the probability of the data given a particular model structure, i.e. the likelihood of the model structure, the cost function relates to the likelihood of the model structure.

3.2 Learning

During learning, the factors $q_i(\boldsymbol{\theta}_i)$ are typically updated one at a time while keeping others fixed. The main reason is that it is often possible to find an analytical solution for $q_i(\boldsymbol{\theta}_i)$ which minimises (4) if $q_j(\boldsymbol{\theta}_j)$ with $j \neq i$ are constant. For each update of the posterior approximation $q_i(\boldsymbol{\theta}_i)$, the variables $\boldsymbol{\theta}_i$ require the prior distribution $p(\boldsymbol{\theta}_i \mid \text{parents})$ given by their parents and

the likelihood $p(\text{children} \mid \boldsymbol{\theta}_i, \text{co-parents})$ obtained from their children [6]. The relevant parts of the cost function (4) to be minimised are

$$C(q_i(\boldsymbol{\theta}_i)) = \left\langle \ln \frac{q_i(\boldsymbol{\theta}_i)}{p(\boldsymbol{\theta}_i \mid \text{parents}) p(\text{children} \mid \boldsymbol{\theta}_i, \text{co-parents})} \right\rangle_q + \text{const}\,, \quad (5)$$

where the expectation is taken over the posterior approximation $q(\boldsymbol{\theta})$ of all unknown variables.

In ensemble learning, conjugate priors are commonly used because they make it very easy to find the optimal $q_i(\boldsymbol{\theta}_i)$ which minimises (5). As an example, consider linear mappings with Gaussian variables. First, note that in (5), the negative logarithm of the prior and likelihood is needed. We shall call this quantity the potential. Gaussian prior has a quadratic potential. The likelihood arising from a linear mapping to Gaussian variables also has a quadratic potential. The sum of the potential is quadratic and the optimal posterior approximation can be shown to be the Gaussian distribution whose potential has the same second and first order terms. The minimisation thus boils down to adding the coefficients of second and first order terms of the prior and likelihood.

Minimising each of the factors $q_i(\boldsymbol{\theta}_i)$ separately is mathematically convenient, but it is often useful to consider alternatives which modify a large number of parameters simultaneously. We apply the optimisation method proposed by Honkela et al. (2003), which speeds up convergence significantly. The idea is to perform line searches in the directions obtained by combining changes resulting from separate minimisation. It is also often useful to design heuristics for proposing changes in the model structure.

3.3 Application to variance nodes

Here we use the method proposed by Valpola et al. (2001). For the sake of efficiency, the posterior approximation has a maximally factorial form, i.e. each unknown variable is approximated to be independent a posteriori of the rest of the variables:

$$q(\boldsymbol{\theta}) = \prod_i q_i(\theta_i)\,. \quad (6)$$

As we noted earlier, using conjugate priors has the benefit that the separate minimisations can be solved analytically. For inverse variance parameters the

[6] In a graphical model representation, each variable is conditionally dependent on its parents (see e.g. Jordan, 1999). In Fig. 1(a), for instance, the parent of source $s_i(t)$ is the variance node $u_{si}(t)$, the children are the observations $\mathbf{x}(t)$ and co-parents are other sources $s_j(t)$ as well as some other parameters of $\mathbf{x}(t)$.

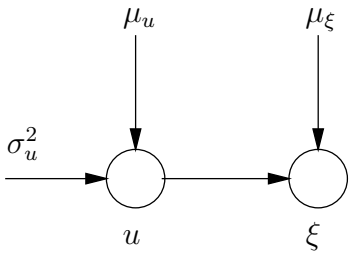

Fig. 2. Variance node u controls the variance of another Gaussian node ξ.

conjugate prior would be the Gamma distribution. However, it would be difficult to build a hierarchical model with Gamma-distributed variables and therefore we chose to have a Gaussian prior and parametrised the variance on logarithmic scale in Section 2.

Consider the likelihood potential which the variance node u receives from the Gaussian variable ξ whose variance it models. Due to the logarithmic scale, it is relatively well approximated by a quadratic function. This makes it feasible to approximate the posterior probability $q(u)$ of the variance node u by a Gaussian distribution: $q(u) \sim N(m_u, v_u)$. It also turns out that the cost function has an analytic form. The drawback is that the optimal $q(u)$ which minimises (5) can not be expressed analytically. However, numerical minimisation is fairly straightforward as shown in Appendix A.

Let us now have a closer look at the terms of the cost function (5) which correspond to $q(u)$. First, assume the following prior model (depicted in Figure 2) and posterior approximation:

$$p(\xi \mid \mu_\xi, u) = N(\xi; \mu_\xi, \exp(-u)) \tag{7}$$
$$p(u \mid \mu_u, \sigma_u^2) = N(u; \mu_u, \sigma_u^2) \tag{8}$$
$$q(\xi) = N(\xi; m_\xi, v_\xi) \tag{9}$$
$$q(u) = N(u; m_u, v_u)\,. \tag{10}$$

In other words, the prior means and variances (possibly provided by other parts of the model) are denoted by μ and σ^2 and the posterior means and variances are denoted by m and v, respectively.

The term $\langle \log q(u) \rangle$ in (5) equals

$$\langle \log q(u) \rangle = -\frac{1}{2} - \frac{1}{2} \log 2\pi v_u \tag{11}$$

which is the negative entropy of a Gaussian variable with mean m_u and vari-

ance v_u. The term $\langle -\log p(u \mid \text{parents}) \rangle$ equals

$$\left\langle -\log p(u \mid \mu_u, \sigma_u^2) \right\rangle = \frac{1}{2} \left\langle \log 2\pi\sigma_u + (u - \mu_u)^2/\sigma_u^2 \right\rangle = \frac{1}{2} \left[\log 2\pi + \langle \log \sigma_u \rangle + (m_u^2 + v_u - 2m_u \langle \mu_u \rangle + \left\langle \mu_u^2 \right\rangle) \left\langle 1/\sigma_u^2 \right\rangle \right] \quad (12)$$

since $\langle u \rangle = m$, $\langle u^2 \rangle = m_u^2 + v_u$ and according to (6), u is independent of μ_u and σ_u^2 and we further assume that μ_u and σ_u^2 are independent in $q(\boldsymbol{\theta})$. A similar derivation for the term $\langle -\log p(\text{children} \mid u) \rangle$ yields

$$\langle -\log p(\xi \mid \mu_\xi, u) \rangle = \frac{1}{2} \left\langle \log 2\pi \exp(-u) + (\xi - \mu_\xi)^2 / \exp(-u) \right\rangle = \frac{1}{2} \left[-m_u + \log 2\pi + (m_\xi^2 + v_\xi - 2m_\xi \langle \mu_\xi \rangle + \left\langle \mu_\xi^2 \right\rangle) \langle \exp(u) \rangle \right] . \quad (13)$$

It can be shown by simple integration that $\langle \exp(u) \rangle = \exp(m_u + v_u/2)$.

Collecting the terms related to m_u and v_u from (11)–(13), we obtain

$$C(m_u, v_u) = M m_u + V(m_u^2 + v_u) + E \exp(m_u + v_u/2) - \frac{1}{2} \ln v_u + \text{const} , \quad (14)$$

where the coefficients M, V and E are constants with respect to m_u and v_u. A numerical optimisation method for this function is derived in Appendix A.

Note that in the derivations of (12) and (13), the mean μ was assumed to be independent of the variance σ^2. Consider for instance a model with the structure depicted in Fig. 1(b). The sources $\mathbf{s}(t)$ model both the means and variances of the observations $\mathbf{x}(t)$. However, since the variance is modelled through the variance nodes $\mathbf{u}_x(t)$ which are approximated to be a posteriori independent of $\mathbf{s}(t)$ according to (6), the assumption is fulfilled. If $\mathbf{s}(t)$ would model both the means and variances of $\mathbf{x}(t)$ without the variance nodes, the terms resulting from $\langle -\log p(x_i(t) \mid \mathbf{s}(t)) \rangle$ would be far more complicated.

3.4 Computational complexity

Due to the approximation (6), the computational complexity of updating the posterior approximation $q_i(\theta_i)$ of variable θ_i is proportional to the number of connections it has with other variables. With model structures discussed in Section 2, most connections in the model arise in the linear mappings between Gaussian variables. The number of connections is typically far greater than the number of variance node variables. Hence the computational complexity of minimising the functions (14) is usually negligible compared to other computations. We shall therefore concentrate here on the computations related to linear mappings between Gaussian variables.

In (13), we have already computed the significant term which results from a Gaussian variable. For the moment, assume ξ to be any Gaussian variable in the model whose mean is modelled by a set of other Gaussian variables s_i and a_i (a subset of a_{ij}, the entries of $\mathbf{A}$) through a linear model

$$\mu_\xi = \sum_i a_i s_i \,. \tag{15}$$

We assumed a sum of products of two terms, but in general the analysis extends to an arbitrary combination of sums and products (Valpola et al., 2001). Note that for the moment, ξ can also be a variance node.

In (13), the following quantities are needed: $\langle \mu_\xi \rangle$ and $\left\langle \mu_\xi^2 \right\rangle$. Due to (6), the former quantity is simply

$$\langle \mu_\xi \rangle = \sum_i \langle a_i \rangle \langle s_i \rangle \tag{16}$$

where $\langle a_i \rangle$ and $\langle s_i \rangle$ are readily obtained from the mean parameters of $q(a_i)$ and $q(s_i)$. The latter quantity can be obtained by defining $\mathrm{Var}\{\cdot\}$ to be the variance over the probability $q(\boldsymbol{\theta})$ and noting that $\langle x^2 \rangle = \langle x \rangle^2 + \mathrm{Var}\{x\}$:

$$\left\langle \mu_\xi^2 \right\rangle = \langle \mu_\xi \rangle^2 + \mathrm{Var}\{\mu_\xi\} = \langle \mu_\xi \rangle^2 + \sum_i \mathrm{Var}\{a_i s_i\} \tag{17}$$

$$\begin{aligned} \mathrm{Var}\{a_i s_i\} &= \left\langle a_i^2 \right\rangle \left\langle s_i^2 \right\rangle - \langle a_i \rangle^2 \langle s_i \rangle^2 \\ &= \langle a_i \rangle^2 \mathrm{Var}\{s_i\} + \mathrm{Var}\{a_i\} \langle s_i \rangle^2 + \mathrm{Var}\{a_i\} \mathrm{Var}\{s_i\} \end{aligned} \tag{18}$$

where $\langle a_i \rangle$, $\mathrm{Var}\{a_i\}$, $\langle s_i \rangle$ and $\mathrm{Var}\{s_i\}$ are obtained from the mean and variance of $q(a_i)$ and $q(s_i)$.

We can now compare the required computations to the case where all quantities would have point estimates. For each multiplication in (15) we have one multiplication from (16) and four multiplications from (18) ($\langle s_i \rangle^2$and $\mathrm{Var}\{s_i\}$ can be added before multiplying with $\mathrm{Var}\{a_i\}$). For each addition in (15) we need four additions (two from (18) and two from the summations over i). In general the forward computations thus have the same order of computational complexity as methods using point estimates (based on posterior densities), and require four to five times more computation.

Typical methods for optimising probability density would have backward computations for gradients which in our case are related to the computation of the likelihoods. The computational complexities are comparable to forward computations. However, the likelihood provides richer information which allows to optimise a variable in one step assuming other variables fixed. In methods with point estimates this would correspond to using second order information (diagonal elements of the Hessian matrix) to optimise each parameter separately.

We can conclude that the computational complexity of the proposed method, which is sensitive to posterior probability mass, is of the same order as it would be with methods based on point estimates, which are sensitive to probability density. There are alternative methods such as sampling techniques which are sensitive to probability mass and could be applied to the present problem, joint estimation of means and variances, but they are computationally more complex.

3.5 Consequenses of the posterior approximation

In ensemble learning, the trade-off between efficiency and accuracy can be controlled by the restrictions imposed on the functional form of the approximation of the posterior probability. In general, using more factorised approximations decreases the computational load.

As we have seen, it is also possible to limit the functional form of the factors. In this work we have restricted the approximation of the posterior density of the variance nodes to be Gaussian. In fact, all posterior densities we use here are Gaussian, but for the rest of the variables, this is the minimum of the cost function and not an imposed restriction.

The Gaussian approximation of the posterior probability of variance nodes is mostly very accurate. The worst case occurs when the variance node u has only very vague prior information about the variance of its corresponding Gaussian node ξ. Then the posterior is very wide (with variance 2) and furthest away from Gaussian. Even then the Kullback-Leibler divergence between the Gaussian approximation and the unrestricted approximation can be shown to be no larger than $\ln \Gamma(1/2) + 1/2 - 1/2 \ln 2 \approx 0.15$. When the Gaussian prior of the variance node is informative, the posterior will be closer to Gaussian than in the worst case.

Complete factorisation is a strong approximation which can affect the quality of the estimated solution. For instance Ilin and Valpola (2003) have shown that it can compromise the quality of separation in ICA. This finding is relevant since we are using similar types of latent variable models with linear mappings. In practice this means that the proposed method may not be sufficient to find independent components. For this reason Valpola et al. (2003b) used a simple linear ICA algorithm for post-processing the results of an ensemble learning based algorithm for nonlinear ICA. It would also be possible to improve the posterior approximation but this would increase the computational complexity and have only little advantage over the simple post-processing approach.

4 Experiments

In this section, we report experiments with artificial data and real magnetoencephalographic (MEG) signals. The first experiment with artificial data uses the model structure depicted in Fig. 1(b) and its main goal is to demonstrate the feasibility and accuracy of joint estimation of means and variances.

In the second experiment with MEG signals, the model structure shown in Fig. 1(c) is used. The goal of this experiment is to give an example of variance sources found in real data and to motivate the claim that these type of sources can act as invariant features.

The experiments were realised using the library of building blocks proposed by Valpola et al. (2001). The code for running the simulations can be obtained at `http://www.cis.hut.fi/projects/bayes/`.

4.1 Learning scheme

The learning scheme is designed to minimise the cost function (4). The basic operation during learning is an iteration where all the terms $q_i(\theta_i)$ of $q(\boldsymbol{\theta})$ are updated one at a time by minimising (5). In addition, several other operations are performed:

- addition of weights;
- pruning of weights; and
- line search.

For line searches, we used the method proposed by Honkela et al. (2003). The idea is to monitor the individual updates during one iteration and then perform a line search simultaneously for all $q_i(\theta_i)$. We applied the line search after every tenth iteration.

The addition and pruning operations aim at optimising the model structure. The cost function (4) relates to the model evidence $p(\mathbf{X} \mid \text{model})$ which can be used to find the most likely model structure.

In general, addition takes place randomly and pruning is based on estimating whether the cost function can be decreased by removing a weight. The motivation for this is that ensemble learning can effectively prune out parts of the model which are not needed. The weights of the linear mappings can for instance approach zero. The cost function can usually be decreased by removing such weights. If all outgoing weights of a source have been removed, the source becomes useless and can be removed from the model. Ensemble

learning cannot, however, actively make room for a part of the model which may be added in the future. It usually takes some time for the rest of the model to accommodate to additions.

During learning, it is necessary to initialise some variables and keep them fixed for a while until other parts of the model have accommodated appropriately. We use evidence nodes, as we call them. They are attached to a variable θ_i, whose value we want to set, and provide an extra term for the likelihood $p(\text{children} \mid \theta_i)$. When $q_i(\theta_i)$ is updated, θ_i will be close to the value set by evidence node if the likelihood term has a narrow peak but θ_i can accommodate to other parts of the model if the likelihood term is wide. After each iteration, the extra term for the likelihood is decayed a little on the logarithmic scale, and the evidence node is removed when the extra term vanishes. The persistance of the initialisation can be controlled by the life-span of the evidence node.

4.2 Artificial data

In order to test the ability of the proposed method to jointly estimate the means and variances, we performed experiments with the model structure presented in Fig. 1(b). We tried to keep the experiment as simple as possible and generated a small data set which matches the assumed model. The model is summarised by the following set of equation:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}_x(t) \tag{19}$$
$$n_{xi}(t) \sim N(0, \exp -u_{xi}(t)) \tag{20}$$
$$\mathbf{u}_x(t) = \mathbf{B}\mathbf{s}(t) + \mathbf{n}_u(t) \tag{21}$$
$$n_{ui}(t) \sim N(\mu_{uxi}, \sigma^2_{uxi}) \tag{22}$$
$$s_i(t) \sim N(0, \exp -u_{si}(t)) \tag{23}$$
$$u_{si}(t) \sim N(\mu_{usi}, \sigma^2_{usi})\,. \tag{24}$$

The observations $\mathbf{x}(t)$ are generated by a linear mapping $\mathbf{A}$ from source vectors $\mathbf{s}(t)$. The observations are corrupted by additive Gaussian noise whose log-variance is obtained by a linear mapping $\mathbf{B}$ from the source vectors. According to the model, the log-variance of sources is modulated by the Gaussian variables $u_{si}(t)$.

In order to be able to better visualise the results, the log-variances of the sources were actually taken to be sinusoids with different frequencies. There were only three sources: only one of them affected both the means and variances of the observations while the two other were specialised to mean or variance. The data set consisting of 1000 observations vectors, whose dimension was ten, is shown in Fig. 3.

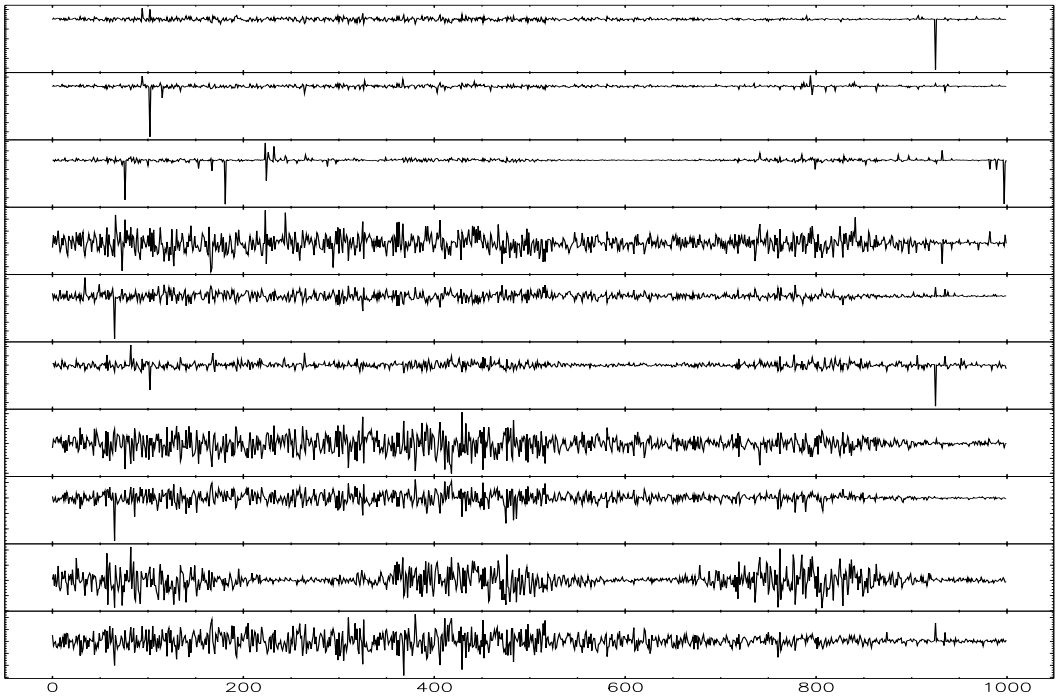

Fig. 3. Artificial data $\mathbf{x}(t)$ consisting of 10 time series with 1000 samples each.

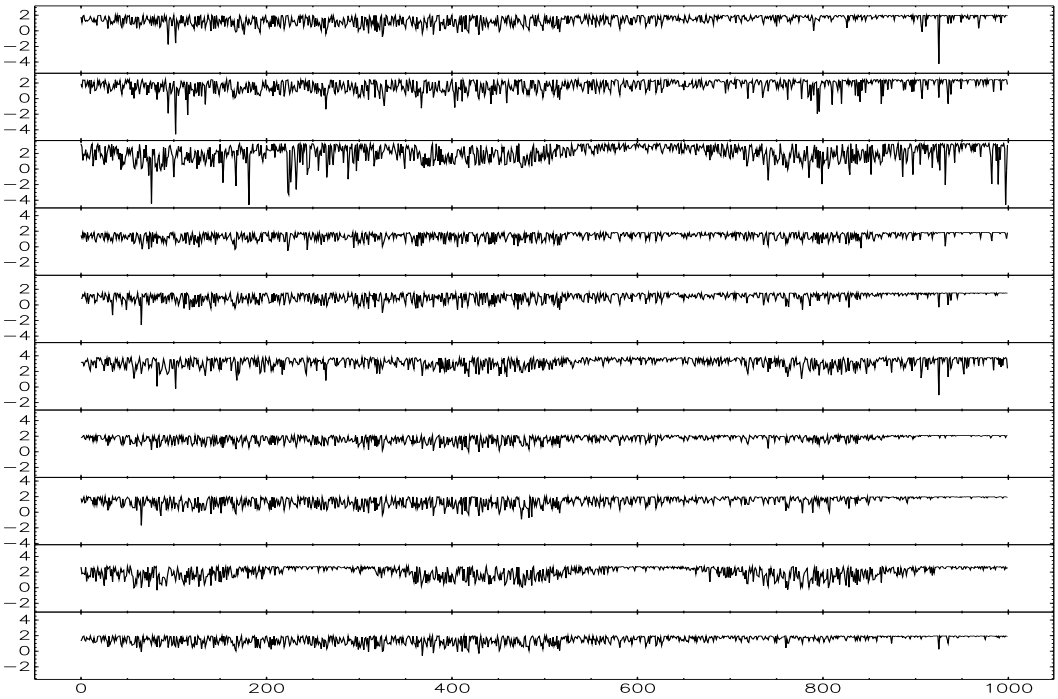

Fig. 4. The posterior means of variance nodes $\mathbf{u}_x(t)$ before adding the sources $\mathbf{s}(t)$.

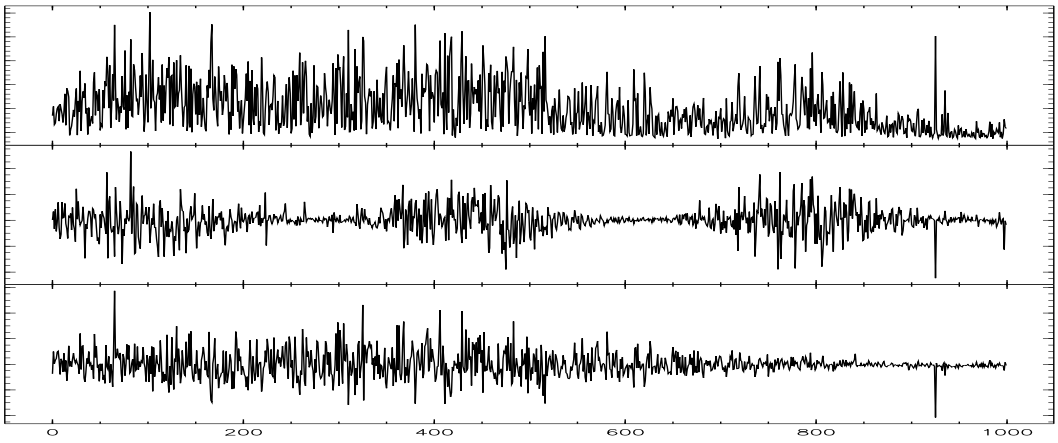

Fig. 5. The initialisation for the posterior means of the sources $\mathbf{s}(t)$.

The learning proceeded in phases. First, only the variance nodes $u_{xi}(t)$ were connected to the observations. The model was iterated 10 times to find reasonable initial values for the variance nodes, depicted in Fig. 4. In order to obtain reasonable initial values for the sources $\mathbf{s}(t)$, we normalised each time series $x_i(t)$ and $u_{xi}(t)$ to unit variance, and extracted a three dimensional subspace. It was then rotated using the FastICA algorithm (FastICA, 1998; Hyvärinen et al., 2001) to obtain the initial values presented in Fig. 5. The initialisation was done with evidence nodes connected to the sources. The evidence nodes decayed in 10 iterations.

Learning was then continued until a total of 1000 were completed. The weights were pruned every 100 iterations startig from 500 iterations and addition of weights was tried every 100 iterations starting from 550 iterations. The pruning was able to find a nearly correct structure of the model: one source was connected to all observations and variance nodes of the observations while the two other sources had lost all but four weights to either observations or variance nodes. None of the additions of the weights were accepted.

The final estimated sources together with the true underlying sources are shown in Fig. 6. The signal-to-noise ratio of the estimated sources were 22.4 dB for the source which was connected to both $\mathbf{x}(t)$ and $\mathbf{u}_x(t)$, 24.3 db for the source connected to $\mathbf{x}(t)$ and 9.5 dB for the source connected to $\mathbf{u}_x(t)$. This reflects the fact that more samples are needed to obtain an accurate estimate of variance than mean.

Initially the corresponding SNRs were 9.6 dB, 10.1 dB and -2.9 dB. The drastic improvement in the signal-to-noise ratios verifies that the model has been able to factor out the contributions of the sources to the means and variances of the observations. The proposed method is able to estimate the model even when the number of unknown variables is more than 1.6 times the number of observations.

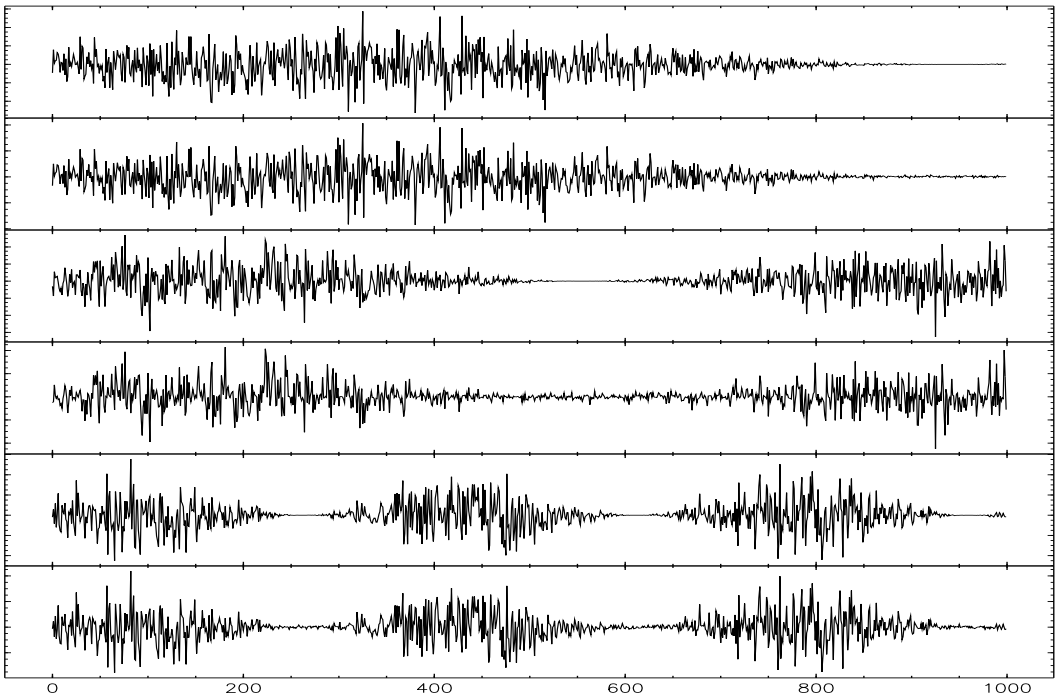

Fig. 6. The estimated posterior means for the three sources $\mathbf{s}(t)$ at the end of the learning and the true underlying sources comparison (the corresponding true source is above each estimated source).

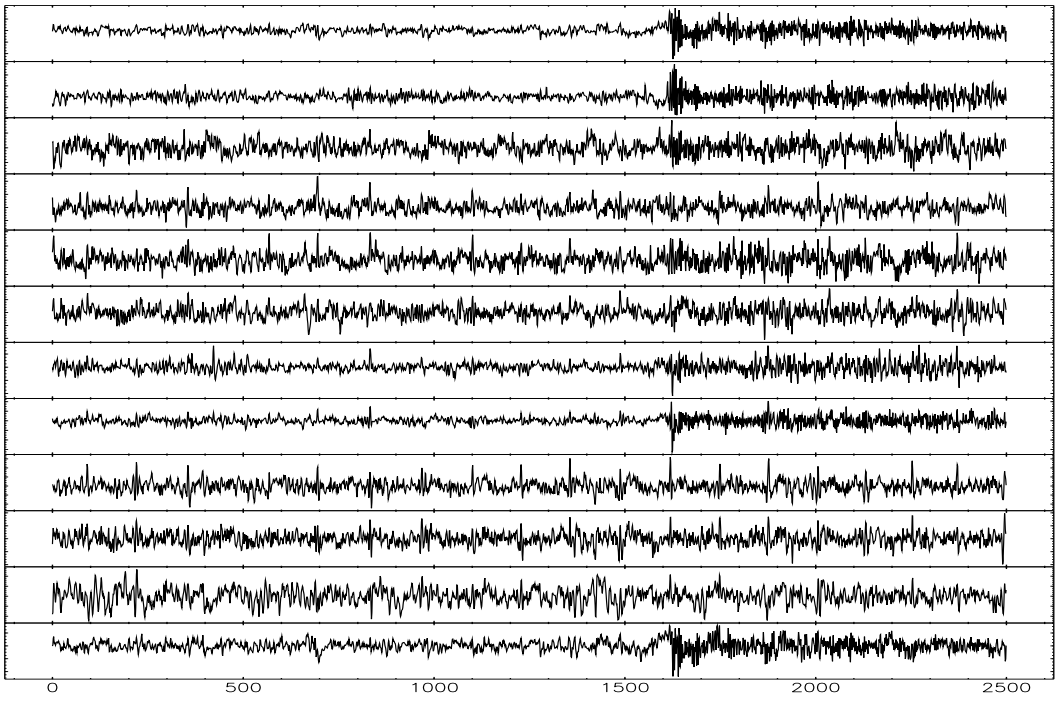

Fig. 7. MEG recordings (12 out of 122 time series).

4.3 Biomedical data

In these experiments, we used part of the MEG data set used by Vigário et al. (2000). The data consists of signals originating from brain activity. The signals

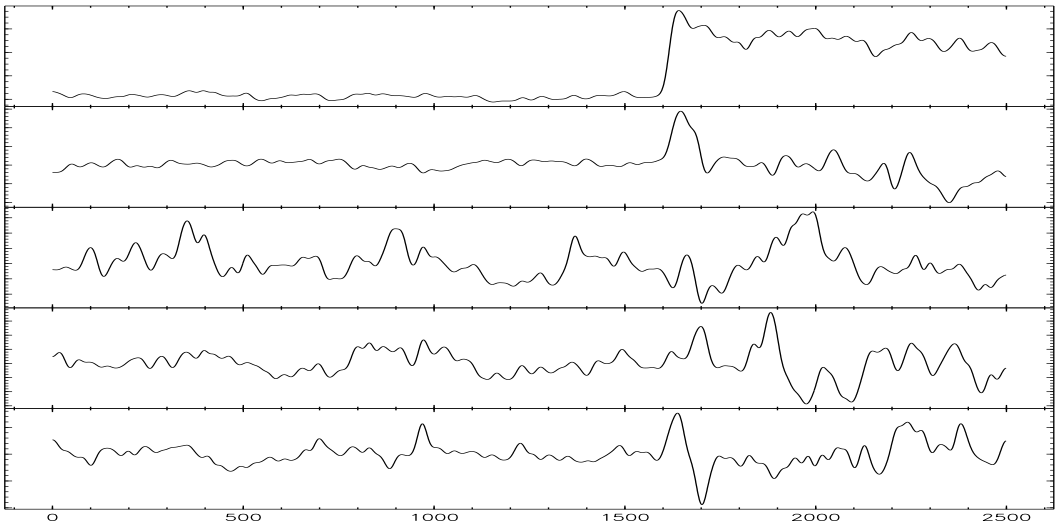

Fig. 8. The initialisation for the posterior means of the variance sources $\mathbf{r}(t)$.

are contaminated by external artefacts such as a digital watch, heart beat as well as eye movements and blinks. We used 2,500 samples of the original data set. The most prominent feature in this area is the biting artefact where muscle activity contaminates many of the channels starting after 1,600 samples as can be seen in Fig. 7.

According to the model used for this experiment, the observations are generated by conventional source vectors $\mathbf{s}(t)$ mapped linearly to the observation vectors $\mathbf{x}(t)$ which are corrupted by additive Gaussian noise $\mathbf{n}(t)$. For each source $s_i(t)$ there is a variance node $u_{si}(t)$ which represents the negative logarithm of the variance. The values of the variance nodes $\mathbf{u}_s(t)$ are further modelled by higher-level variance sources $\mathbf{r}(t)$ which map linearly to the variance nodes. Variance sources, too, have variance nodes $\mathbf{u}_r(t)$ attached to them.

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \tag{25}$$
$$s_i(t) \sim N(s_i(t-1), \exp -u_{si}(t)) \tag{26}$$
$$\mathbf{u}_s(t) = \mathbf{B}\mathbf{r}(t) + \mathbf{m}(t) \tag{27}$$
$$r_i(t) \sim N(r_i(t-1), \exp -u_{ri}(t)) \tag{28}$$

The additive Gaussian noise terms $\mathbf{n}(t)$ and $\mathbf{m}(t)$ are allowed to have non-zero bias. The model structure is shown in Fig. 1(c). Note that it makes sense to have two layers although the model is linear and all variables are Gaussian since the variance nodes $\mathbf{u}_s$ translate the higher-order source model into a prediction of variance. The variance sources are also responsible for generating super-Gaussian distributions for $\mathbf{s}(t)$ and $\mathbf{r}(t)$.

Both the sources and variance sources have a dynamic model. The predicted mean is taken to be the value at the previous time instant. This is reasonable since the MEG signals have strong temporal dependences.

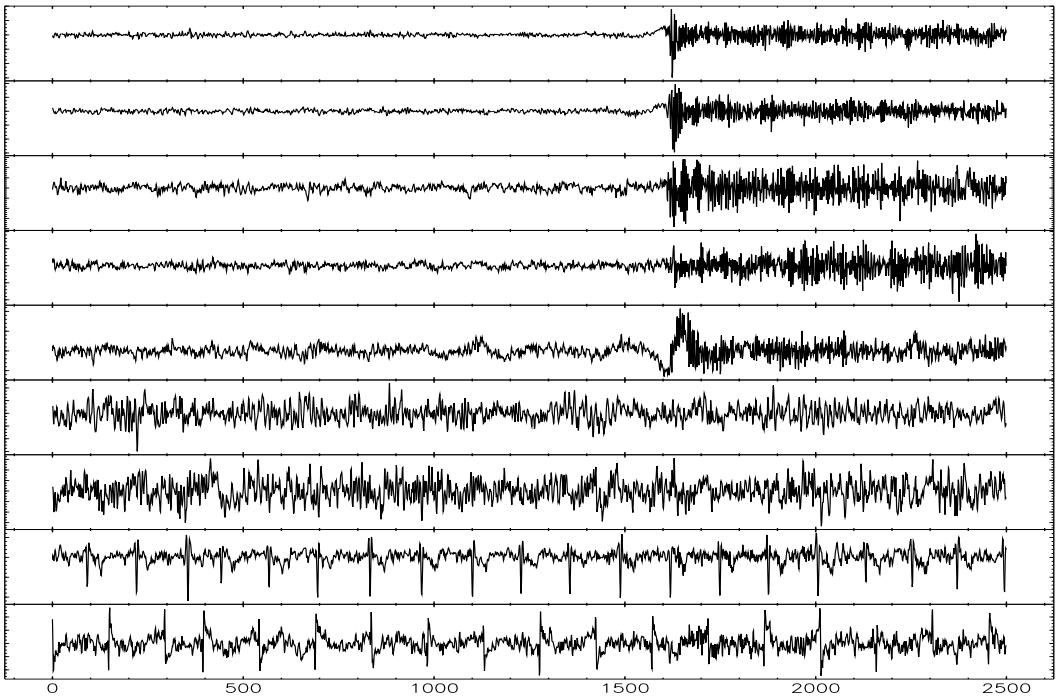

Fig. 9. Sources $\mathbf{s}(t)$ estimated from the MEG data (nine out of 50 sources).

Initially the model had 50 sources $\mathbf{s}(t)$ which were initialised using the independent components estimated from the observations by the FastICA algorithm (FastICA, 1998; Hyvärinen et al., 2001). The second layer with five variance sources $\mathbf{r}(t)$ was added after the first 20 iterations. It was initialised by taking the posterior means of the variance nodes $\mathbf{u}_s(t)$ of the sources, normalising the time series to unit variance, low-pass filtering and then computing the initialisations by principal component analysis. The initialisations of the variance sources $\mathbf{r}(t)$ are shown in Fig. 8. The evidence nodes for the initialisation of the sources decayed in 10 iterations while those for the variance sources decayed in 200 iterations.

Learning was continued until a total of 2,000 iterations had been accomplished. Weights were pruned every 200 iterations starting after the first 500 iterations and added every 200 iterations starting after the first 600 iterations. None of the sources lost all their weights during the structural optimisation.

Some of the sources and their variance nodes are depicted in Figs. 9 and 10, respectively. The conventional sources are comparable to those reported in the literature for this data set (Vigário et al., 2000).

The first variance source in Fig. 11 clearly models the biting artefact. This variance source integrates information from several conventional sources and its activity varies very little over time. This is partly due to the dynamics but experiments with a static model confirm that the variance source acts as an invariant feature which reliably detects the biting artefact.

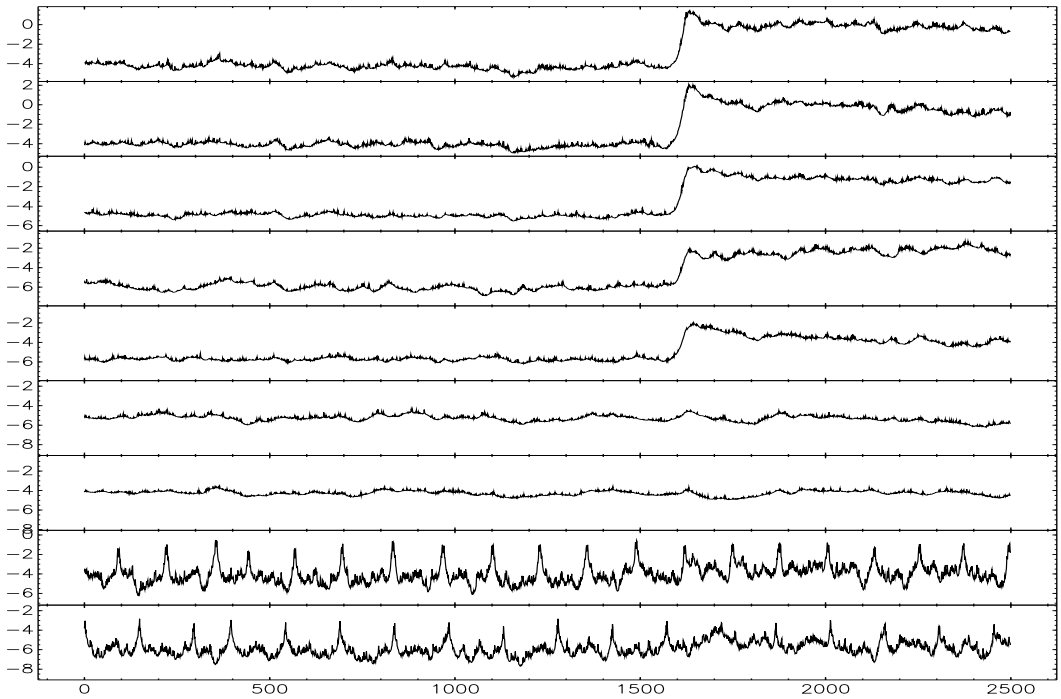

Fig. 10. Variance nodes $\mathbf{u}_s(t)$ corresponding to the sources shown in Fig. 9.

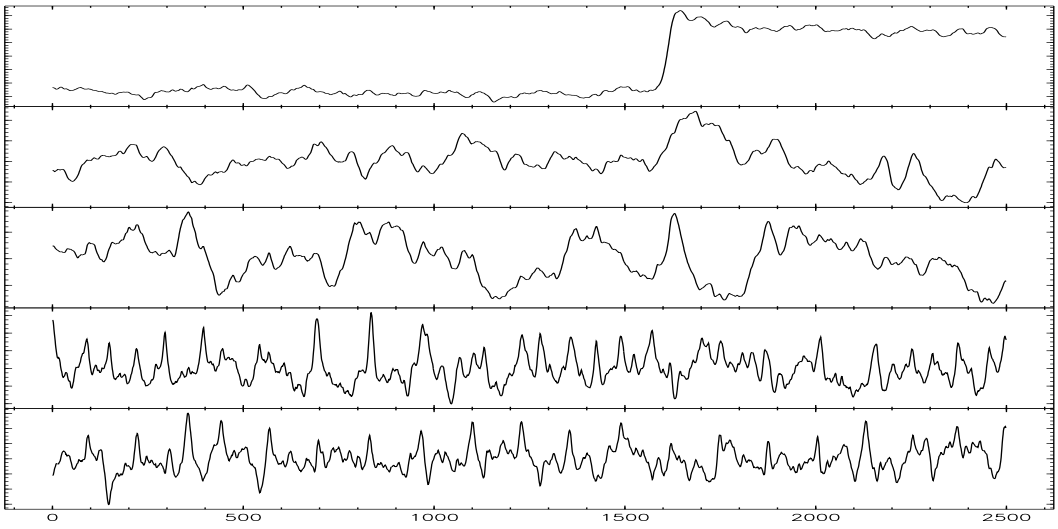

Fig. 11. Variance sources $\mathbf{r}(t)$ which model the regularities found in the variance nodes shown in Fig. 10.

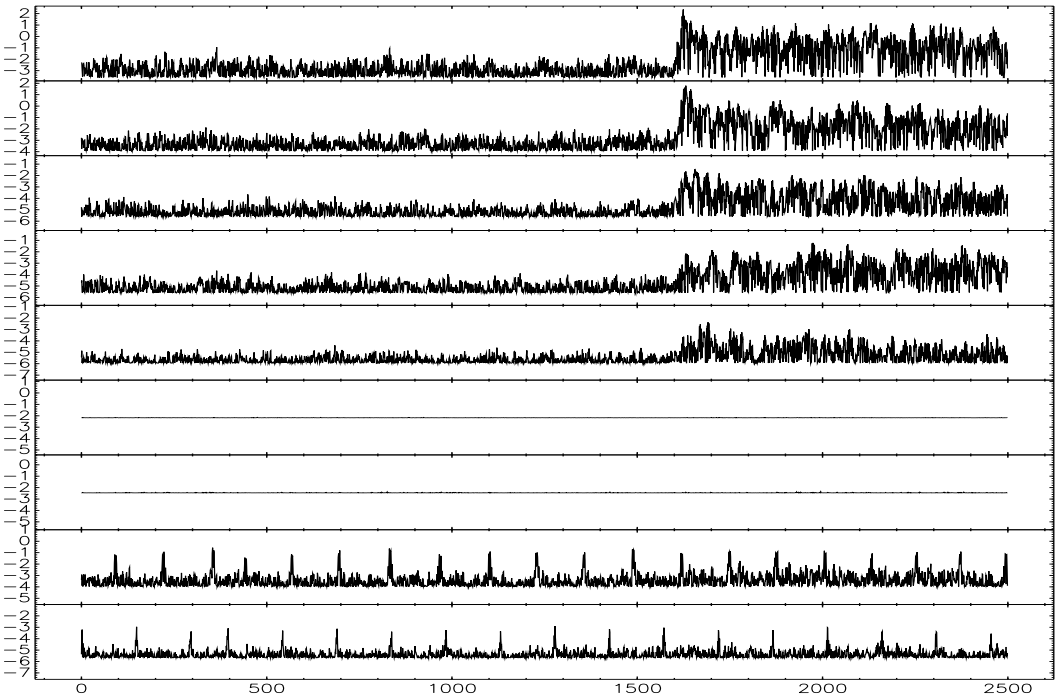

Fig. 12. Variance nodes $\mathbf{u}_s(t)$ corresponding to the sources shown in Fig. 9 in a control experiment which lacked the second layer with variance sources $\mathbf{r}(t)$.

The second variance source appears to represent increased activity during the onset of the biting. The third variance node seems to be related to the amount of rhythmic activity on the sources. Two such sources can be found in Fig. 9 (sixth and seventh source). Interestingly, we also found a source where the amount of rhythmic activity was negatively correlated with the ones shown in the figure. The two remaining variance sources appear to have features describing both the cardiac signal and the digital watch. These observations are supported by the estimated weights connecting the variance sources $\mathbf{r}(t)$ to the variance nodes $\mathbf{u}_s(t)$.

In order to demonstrate the merits of joint estimation of means and variances, we performed an experiment which was otherwise similar but lacked the second layer with variance sources. Figure 12 depicts the estimated variance nodes $\mathbf{u}_s(t)$ in this case. Compared to Fig. 10 the results are clearly noisier. In addition, one can see that the two signals corresponding to rhythmic activity have been estimated to have no significant variations in their variance. The differences are due to the variance sources $\mathbf{r}(t)$ which can integrate the variance information temporally and from several source signals and feed the information back to the variance nodes $\mathbf{u}_s(t)$. This information is then used in estimating the sources $\mathbf{s}(t)$. For instance the estimates of the sources related to the biting artefact have less activity prior to the onset of the biting.

5 Discussion

In statistics, a distribution characterised by changing variance is called heteroskedastic. Heteroskedasticity is known to be commonplace and there are various techniques for modelling the variance (see e.g. Bollerslev, 1986; Ghysels et al., 1996; Kim et al., 1998). However, previously mean has either been estimated separately from variance in order to avoid problems related to infinite probability densities or computationally expensive sampling techniques have been used. We have shown that it is possible to estimate both means and variances together efficiently. This has the benefit that the estimation of the mean can use the information about the variance and vice versa.

We reported experiments with two simple model structures which utilise variance nodes but we have only touched the tip of an iceberg. Since the variance nodes allow to translate models of mean into models of variance, we can go through a large number of models discussed in the literature and consider whether they are useful for modelling variance.

The goal of the experiments reported here was to demonstrate the basic principles of the method, but the learning scheme, for instance, can still be improved. It is likely to be useful to design model-specific heuristics for proposing structural changes and initialisations of sources.

The cost function used in ensemble learning has been crucial in solving the problem discussed in this paper. It correlates well with the quality of the model and does not suffer from overfitting or overlearning (see e.g. Valpola and Karhunen, 2002; Valpola et al., 2003b). It readily allows model comparison and measures how well various heuristics can improve learning.

References

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

Barber, D., Bishop, C., 1998. Ensemble learning in Bayesian neural networks. In: Bishop, C. (Ed.), Neural Networks and Machine Learning. Springer, Berlin, pp. 215–237.

Bollerslev, T., 1986. Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics 31, 307–327.

Cardoso, J.-F., 1998. Multidimensional independent component analysis. In: Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'98). Seattle, Washington, USA, May 12–15, pp. 1941–1944.

Chan, K., Lee, T.-W., Sejnowski, T., 2001. Variational learning of clusters of undercomplete nonsymmetric independent components. In: Proc. Int. Conf.

on Independent Component Analysis and Signal Separation (ICA2001). San Diego, USA, pp. 492–497.

Choudrey, R., Penny, W., Roberts, S., 2000. An ensemble learning approach to independent component analysis. In: Proc. of the IEEE Workshop on Neural Networks for Signal Processing, Sydney, Australia, December 2000. IEEE Press, pp. 435–444.

De Lathauwer, L., De Moor, B., Vandewalle, J., 1995. Fetal electrocardiogram extraction by source subspace separation. In: Proc. IEEE Sig. Proc. / ATHOS Workshop on Higher-Order Statistics. pp. 134–138.

FastICA, 1998. The FastICA MATLAB package. Available at `http://www.cis.hut.fi/projects/ica/fastica/`.

Ghahramani, Z., Hinton, G. E., 2000. Variational learning for switching state-space models. Neural Computation 12 (4), 963–996.

Ghysels, E., Harvey, A. C., Renault, E., 1996. Stochastic volatility. In: Rao, C. R., Maddala, G. S. (Eds.), Statistical Methods in Finance. North-Holland, Amsterdam, pp. 119–191.

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

Hinton, G. E., van Camp, D., 1993. Keeping neural networks simple by minimizing the description length of the weights. In: Proc. of the 6th Ann. ACM Conf. on Computational Learning Theory. Santa Cruz, CA, USA, pp. 5–13.

Honkela, A., Valpola, H., Karhunen, J., 2003. Accelerating cyclic update algorithms for parameter estimation by pattern searches. Neural Processing Letters 17 (2), 191–203.

Hyvärinen, A., Hoyer, P., 2000. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. Neural Computation 12 (7), 1705–1720.

Hyvärinen, A., Karhunen, J., Oja, E., 2001. Independent Component Analysis. J. Wiley.

Hyvärinen, A., Hurri, J., 2003. Blind separation of sources that have spatiotemporal dependencies. Signal Processing Submitted.

Ilin, A., Valpola, H., 2003. On the effect of the form of the posterior approximation in variational learning of ICA models. In: Proc. of the 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003). Nara, Japan, pp. 915–920.

Jordan, M. (Ed.), 1999. Learning in Graphical Models. The MIT Press, Cambridge, MA, USA.

Kim, S., Shepard, N., Chib, S., July 1998. Stochastic volatility: likelihood inference and comparison with ARCH models. The Review of Economic Studies 65 (3), 361–393.

Kohonen, T., Kaski, S., Lappalainen, H., 1997. Self-organized formation of various invariant-feature filters in the Adaptive-Subspace SOM. Neural Computation 9 (6), 1321–1344.

Lappalainen, H., Miskin, J., 2000. Ensemble learning. In: Girolami, M. (Ed.), Advances in Independent Component Analysis. Springer-Verlag, Berlin, pp.

75–92.
Miskin, J., MacKay, D. J. C., 2000. Ensemble learning for blind image separation and deconvolution. In: Girolami, M. (Ed.), Advances in Independent Component Analysis. Springer-Verlag, pp. 123–141.
Parra, L., Spence, C., Sajda, P., 2001. Higher-order statistical properties arising from the non-stationarity of natural signals. In: Leen, T., Dietterich, T., Tresp, V. (Eds.), Advances in Neural Information Processing Systems 13. The MIT Press, Cambridge, MA, USA, pp. 786–792.
Valpola, H., Harva, M., Karhunen, J., 2003a. Hierarchical models of variance sources. In: Proc. 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003). Nara, Japan, pp. 83–88.
Valpola, H., Karhunen, J., 2002. An unsupervised ensemble learning method for nonlinear dynamic state-space models. Neural Computation 14 (11), 2647–2692.
Valpola, H., Östman, T., Karhunen, J., 2003b. Nonlinear independent factor analysis by hierarchical models. In: Proc. 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003). Nara, Japan, pp. 257–262.
Valpola, H., Raiko, T., Karhunen, J., 2001. Building blocks for hierarchical latent variable models. In: Proc. 3rd Int. Conf. on Independent Component Analysis and Signal Separation (ICA2001). San Diego, USA, pp. 710–715.
Vigário, R., Särelä, J., Jousmäki, V., Hämäläinen, M., Oja, E., 2000. Independent component approach to the analysis of EEG and MEG recordings. IEEE transactions on biomedical engineering 47 (5), 589–593.

A Minimisation for variance nodes

Here we show how to minimise a function

$$C(m, v) = Mm + V[m^2 + v] + E \exp(m + v/2) - \frac{1}{2} \ln v \,.$$

A unique solution exists when $V > 0$ and $E > 0$. This problem occurs when a Gaussian posterior with mean m and variance v is fitted to a probability distribution whose logarithm has both a quadratic and exponential part resulting from Gaussian prior and log-Gamma likelihoods, respectively, and Kullback-Leibler divergence is used as the measure of the misfit.

The minimisation is iterative. At each iteration, one Newton-iteration step for m and one fixed-point iteration step for v is performed. The steps are taken until they become smaller than a predefined threshold.

A.1 Newton iteration for m

Newton iteration for m is obtained by

$$m_{i+1} = m_i - \frac{\partial C(m_i, v_i)/\partial m_i}{\partial^2 C(m_i, v_i)/\partial m_i^2} = \\ m_i - \frac{M + 2Vm_i + E\exp(m_i + v_i/2)}{2V + E\exp(m_i + v_i/2)} \quad \text{(A.1)}$$

Newton iteration converges in one step if the second derivative remains constant. The step is too short if the second derivative decreases and too long if the second derivative increases. For stability, it is better to take too short than too long steps.

In this case, the second derivative always decreases if m decreases and vice versa. For stability it is therefore useful to restrict the increases in m because the increases are consistently over-estimated. We have found that restricting the increase to be at most four yields robust convergence.

A.2 Fixed-point iteration for v

A simple fixed-point iteration rule is obtained for v by solving the zero of the derivative:

$$0 = \frac{\partial C(m, v)}{\partial v} = V + \frac{E}{2}\exp(m + v/2) - \frac{1}{2v} \Leftrightarrow \\ v = \frac{1}{2V + E\exp(m + v/2)} \stackrel{def}{=} g(v) \quad \text{(A.2)}$$

$$v_{i+1} = g(v_i) \quad \text{(A.3)}$$

In general, fixed-point iterations are stable around the solution v_{opt} if $|g'(v_{\text{opt}})| < 1$ and converge the best when the derivative $g'(v_{\text{opt}})$ is near zero. In our case $g'(v_i)$ is always negative and can be less than -1, i.e. the solution can be an unstable fixed-point. This can be remedied by taking a weighted average of (A.3) and a trivial iteration $v_{i+1} = v_i$:

$$v_{i+1} = \frac{\xi(v_i)v_i + g(v_i)}{\xi(v_i) + 1} \stackrel{def}{=} f(v_i) \quad \text{(A.4)}$$

The weight ξ should be such that the derivative of f is close to zero at the optimal solution v_{opt} which is achieved exactly if $\xi(v_{\text{opt}}) = -g'(v_{\text{opt}})$.

It holds

$$g'(v) = -\frac{E/2\exp(m+v/2)}{\left[2V + E\exp(m+v/2)\right]^2} =$$
$$g^2(v)\left[V - \frac{1}{2g(v)}\right] = g(v)\left[Vg(v) - \frac{1}{2}\right] \Rightarrow$$
$$g'(v_{\text{opt}}) = v_{\text{opt}}\left[Vv_{\text{opt}} - \frac{1}{2}\right] \Rightarrow \xi(v_{\text{opt}}) = v_{\text{opt}}\left[\frac{1}{2} - Vv_{\text{opt}}\right]. \quad \text{(A.5)}$$

The last steps follow from the fact that $v_{\text{opt}} = g(v_{\text{opt}})$ and the requirement that $f'(v_{\text{opt}}) = 0$. We can assume that v is close to v_{opt} and use

$$\xi(v) = v\left[\frac{1}{2} - Vv\right]. \quad \text{(A.6)}$$

Note that the iteration (A.3) can only yield estimates with $0 < v_{i+1} < 1/2V$ which means that $\xi(v_{i+1}) > 0$. Therefore the step defined by (A.4) is always shorter than the step defined by (A.3).

Since we know that the solution lies between 0 and $1/2V$, we can set $v_0 = 1/2V$ if the current estimate is greater than $1/2V$.

In order to improve stability, step sizes need to be restricted. Increases in v are more problematic than decreases since the $\exp(m + v/2)$ term behaves more nonlinearly when v increases. Again, we have found experimentally that restricting the increase to be at most four yields robust convergence.

A.3 Summary of the iteration

(1) Set $v_0 \leftarrow min(v_0, 1/2V)$.
(2) Iterate
   (a) Solve new m by (A.1) under the restriction that the maximum step is 4
   (b) Solve new v by (A.6) and (A.4) under the restriction that the maximum step is 4

   until both steps are smaller than 10^{-4}.