Chapter 4

Variational Bayesian learning of generative models

Juha Karhunen, Antti Honkela, Alexander Ilin, Tapani Raiko, Markus Harva, Harri Valpola, Erkki Oja

4.1 Bayesian modeling and variational learning: introduction

Unsupervised learning methods are often based on a generative approach where the goal is to find a model which explains how the observations were generated. It is assumed that there exist certain source signals (also called factors, latent or hidden variables, or hidden causes) which have generated the observed data through an unknown mapping. The goal of generative learning is to identify both the source signals and the unknown generative mapping.

The success of a specific model depends on how well it captures the structure of the phenomena underlying the observations. Various linear models have been popular, because their mathematical treatment is fairly easy. However, in many realistic cases the observations have been generated by a nonlinear process. Unsupervised learning of a nonlinear model is a challenging task, because it is typically computationally much more demanding than for linear models, and flexible models require strong regularization.

In Bayesian data analysis and estimation methods, all the uncertain quantities are modeled in terms of their joint probability distribution. The key principle is to construct the joint posterior distribution for all the unknown quantities in a model, given the data sample. This posterior distribution contains all the relevant information on the parameters to be estimated in parametric models, or the predictions in non-parametric prediction or classification tasks [1].

Denote by $\mathcal{H}$ the particular model under consideration, and by $\boldsymbol{\theta}$ the set of model parameters that we wish to infer from a given data set X. The posterior probability density $p(\boldsymbol{\theta}|X,\mathcal{H})$ of the parameters given the data X and the model $\mathcal{H}$ can be computed from the Bayes' rule

$$p(\boldsymbol{\theta}|X,\mathcal{H}) = \frac{p(X|\boldsymbol{\theta},\mathcal{H})p(\boldsymbol{\theta}|\mathcal{H})}{p(X|\mathcal{H})} \tag{4.1}$$

Here $p(X|\boldsymbol{\theta},\mathcal{H})$ is the likelihood of the parameters $\boldsymbol{\theta}$, $p(\boldsymbol{\theta}|\mathcal{H})$ is the prior pdf of the parameters, and $p(X|\mathcal{H})$ is a normalizing constant. The term $\mathcal{H}$ denotes all the assumptions made in defining the model, such as choice of a multilayer perceptron (MLP) network, specific noise model, etc.

The parameters $\boldsymbol{\theta}$ of a particular model $\mathcal{H}_i$ are often estimated by seeking the peak value of a probability distribution. The non-Bayesian maximum likelihood (ML) method uses to this end the distribution $p(X|\boldsymbol{\theta},\mathcal{H})$ of the data, and the Bayesian maximum a posteriori (MAP) method finds the parameter values that maximize the posterior probability density $p(\boldsymbol{\theta}|X,\mathcal{H})$. However, using point estimates provided by the ML or MAP methods is often problematic, because the model order estimation and overfitting (choosing too complicated a model for the given data) are severe problems [1].

Instead of searching for some point estimates, the correct Bayesian procedure is to use all possible models to evaluate predictions and weight them by the respective posterior probabilities of the models. This means that the predictions will be sensitive to regions where the probability mass is large instead of being sensitive to high values of the probability density [2]. This procedure optimally solves the issues related to the model complexity and choice of a specific model $\mathcal{H}_i$ among several candidates. In practice, however, the differences between the probabilities of candidate model structures are often very large, and hence it is sufficient to select the most probable model and use the estimates or predictions given by it.

A problem with fully Bayesian estimation is that the posterior distribution (4.1) has a highly complicated form except for in the simplest problems. Therefore it is too difficult to handle exactly, and some approximative method must be used. Variational methods form a class of approximations where the exact posterior is approximated with a simpler distribution [3]. In a method commonly known as *Variational Bayes (VB)* [1, 2] the misfit of the approximation is measured by the Kullback-Leibler (KL) divergence between two probability distributions $q(v)$ and $p(v)$. The KL divergence is defined by

$$D(q \parallel p) = \int q(v) \ln \frac{q(v)}{p(v)} dv \tag{4.2}$$

which measures the difference in the probability mass between the densities $q(v)$ and $p(v)$.

A key idea in the VB method is to minimize the misfit between the actual posterior pdf and its parametric approximation using the KL divergence. The approximating density is often taken a diagonal multivariate Gaussian density, because the computations become then tractable. Even this crude approximation is adequate for finding the region where the mass of the actual posterior density is concentrated. The mean values of the Gaussian approximation provide reasonably good point estimates of the unknown parameters, and the respective variances measure the reliability of these estimates.

A main motivation of using VB is that it avoids overfitting which would be a difficult problem if ML or MAP estimates were used. VB method allows one to select a model having appropriate complexity, making often possible to infer the correct number of sources or latent variables. It has provided good estimation results in the very difficult unsupervised (blind) learning problems that we have considered.

Variational Bayes is closely related to information theoretic approaches which minimize the description length of the data, because the description length is defined to be the negative logarithm of the probability. Minimal description length thus means maximal probability. In the probabilistic framework, we try to find the sources or factors and the nonlinear mapping which most probably correspond to the observed data. In the information theoretic framework, this corresponds to finding the sources and the mapping that can generate the observed data and have the minimum total complexity. The information theoretic view also provides insights to many aspects of learning and helps explain several common problems [4].

In the following subsections, we first present some recent theoretical improvements to VB methods and a practical building block framework that can be used to easily construct new models. After this we discuss practical models for nonlinear and non-negative blind source separation as well as multivariate time series analysis using nonlinear state-space models. A more structured extension of probabilistic relational models is also presented. Finally we present applications of the developed Bayesian methods to astronomical data analysis problems.

4.2 Theoretical improvements

Effect of posterior approximation

Most applications of variational Bayesian learning to ICA models reported in the literature assume a fully factorized posterior approximation $q(v)$, because this usually results in a computationally efficient learning algorithm. However, the simplicity of the posterior approximation does not allow for representing all possible solutions, which may greatly affect the found solution.

Our paper [5] shows that neglecting the posterior correlations of the sources $\mathbf{S}$ in the approximating density $q(\mathbf{S})$ introduces a bias in favor of the principal component analysis (PCA) solution. By the PCA solution we mean the solution which has an orthogonal mixing matrix. Nevertheless, if the true mixing matrix is close to orthogonal and the source model is strongly in favor of the desirable ICA solution, the optimal solution can be expected to be close to the ICA solution. In [5], we studied this problem both theoretically and experimentally by considering linear ICA models with either independent dynamics or non-Gaussian source models. The analysis also extends to the case of nonlinear mixtures.

Figure 4.1 presents experimental results illustrating the general trade-off of variational Bayesian learning between the misfit of the posterior approximation and the accuracy of the model. According to our assumption, the sources can be accurately modeled in the ICA solution and therefore the cost of inaccurate assumption would increase towards the ICA solution. As a result, the ICA solution is found for strongly non-Gaussian sources ($\nu = 1$). On the other hand, if the true mixing matrix is not orthogonal, the optimal posterior covariance of the sources could have posterior correlations between the sources. Then, the misfit of the posterior approximation of the sources is minimized in the PCA solution where the true posterior covariance would be diagonal. This is the reason why the PCA solution is found for the sources whose distribution is close to Gaussian ($\nu = 0.6$). In the intermediate cases ($\nu = 0.7, \nu = 0.9$), some compromise solutions, which lie in between the PCA and ICA solutions, can be found.

Accurate linearisation for learning nonlinear models

Learning of nonlinear models in the variational Bayesian framework fundamentally reduces to evaluating statistics of the data predicted by the model as a function of the parameters of the variational approximation of the posterior distribution. This is equivalent to evaluating statistics of a nonlinear transformation of the approximating probability distribution. A common approach that was also used in our earlier work on nonlinear models [6, 7] is to use a Taylor series approximation to linearise the nonlinearity. Unfortunately this approximation breaks down when the variance of the approximating distribution increases, and this leads to algorithmic instability.

For handling this problem, a new linearisation method based on replacing the local approach of the Taylor scheme with a more global approximation was proposed in [8, 9]. In case of multilayer perceptron (MLP) networks this can be done efficiently by replacing the nonlinear activation function of the hidden neurons by a linear function that would provide the same output mean and variance, as evaluated by Gauss–Hermite quadrature. The resulting approximation yields significantly more accurate estimates of the cost of the model while being computationally almost as efficient. This is illustrated in Figure 4.2.

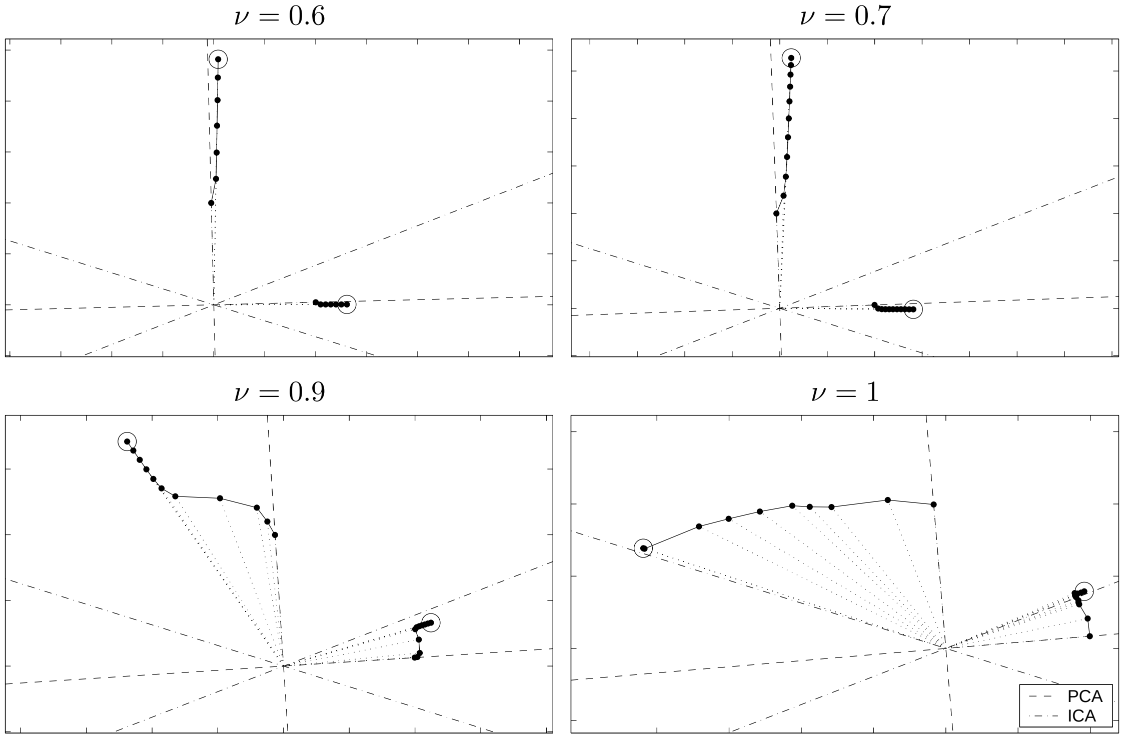

Figure 4.1: Separation results obtained with a model with super-Gaussian sources and fully factorial approximation for four test ICA problems. The parameter ν is the measure of the non-Gaussianity of the sources used in the test data. The dotted lines represent the columns of the mixing matrix during learning, the final solution is circled. The PCA and ICA directions are shown on the plots with the dashed and dashed-dotted lines respectively.

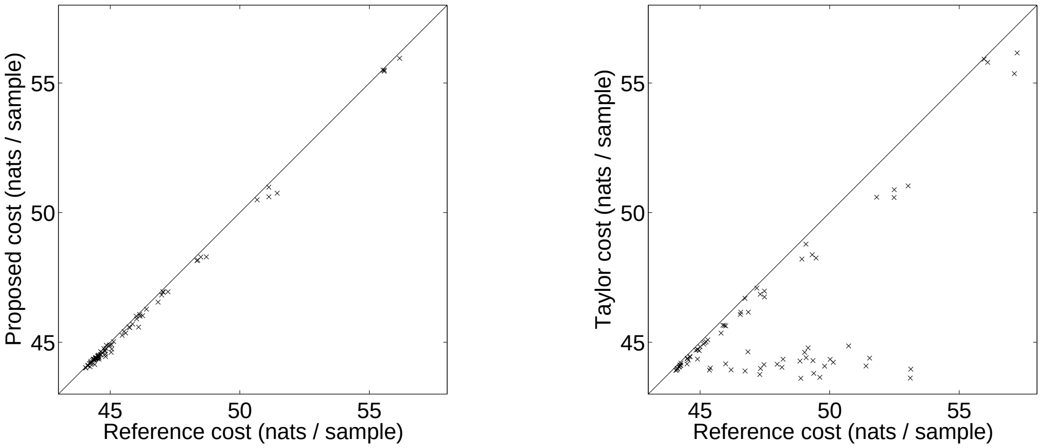

Figure 4.2: The attained values of the cost in different simulations as evaluated by the different approximations plotted against reference values evaluated by sampling. The left subfigure shows the values from experiments using the proposed approximation and the right subfigure from experiments using the Taylor approximation.

Partially observed values

It is well known that Bayesian methods provide well-founded and straightforward means for handling missing values in data. The same applies to values that are somewhere between observed and missing. So-called coarse data means that we only know that a data point belongs to a certain subset of all possibilities. So-called soft or fuzzy data generalises this further by giving weights to the possibilities. In [10], different ways of handling soft data are studied in context of variational Bayesian learning. A simple example is given in Figure 4.3. The approach called virtual evidence is recommended based on both theory and experimentation with real image data. Also, a justification is given for the standard preprocessing step of adding a tiny amount of noise to the data, when a continuous-valued model is used for discrete-valued data.

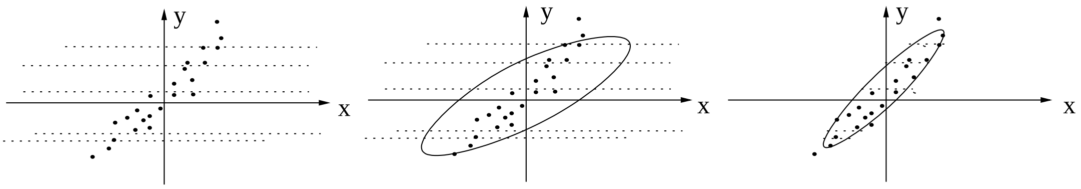

Figure 4.3: Some x-values of the data are observed only partially. They are marked with dotted lines representing their confidence intervals. Left: A simple data set for a factor analysis problem. Middle: In the compared approach, the model needs to adjust to cover the distributions. Right: In the proposed approach, the partially observed values are reconstructed based on the model.

4.3 Building blocks for variational Bayesian learning

In graphical models, there are lots of possibilities to build the model structure that defines the dependencies between the parameters and the data. To be able to manage the variety, we have designed a modular software package using C++/Python called the Bayes Blocks [11]. The theoretical background on which it is based on, was published in [12].

The design principles for Bayes Blocks have been the following. Firstly, we use standardized building blocks that can be connected rather freely and can be learned with local learning rules, i.e. each block only needs to communicate with its neighbors. Secondly, the system should work with very large scale models. We have made the computational complexity linear with respect to the number of data samples and connections in the model.

The building blocks include Gaussian variables, summation, multiplication, and nonlinearity. Recently, several new blocks were implemented including mixture-of-Gaussians and rectified Gaussians [13]. Each of the blocks can be a scalar or a vector. Variational Bayesian learning provides a cost function which can be used for updating the variables as well as optimizing the model structure. The derivation of the cost function and learning rules is automatic which means that the user only needs to define the connections between the blocks.

Examples of structures which can be build using the Bayes Blocks library can be found in Figure 4.4 in the following subsection as well as [12, 14].

Hierarchical modeling of variances

In many models, variances are assumed to have constant values although this assumption is often unrealistic in practice. Joint modeling of means and variances is difficult in many learning approaches, because it can give rise to infinite probability densities. In Bayesian methods where sampling is employed, the difficulties with infinite probability densities are avoided, but these methods are not efficient enough for very large models. Our variational Bayesian method [14], which is based on the building blocks framework, is able to jointly model both variances and means efficiently.

The basic building block in our models is the variance node, which is a time-dependent Gaussian variable $u(t)$ controlling the variance of another time-dependent Gaussian variable $\xi(t)$

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

Here $\mathcal{N}(\mu, \sigma^2)$ is the Gaussian distribution with mean μ and variance σ^2, and $\mu_\xi(t)$ is the mean of $\xi(t)$ given by other parts of the model.

Figure 4.4 shows three examples of usage of variance nodes. The first model does not have any upper layer model for the variances. There the variance nodes are useful as such for generating super-Gaussian distributions for $\mathbf{s}$, enabling us to find independent components. In the second model the sources can model concurrent changes in both the observations $\mathbf{x}$ and the modeling error of the observations through variance nodes $\mathbf{u}_x$. The third model is a hierarchical extension of the linear ICA model. The correlations and concurrent changes in the variances $\mathbf{u}_s(t)$ of conventional sources $\mathbf{s}(t)$ are modeled by higher-order variance sources $\mathbf{r}(t)$.

We have used the model of Fig. 4.4(c) for finding variance sources from biomedical data containing MEG measurements from a human brain [14]. The signals are contaminated by external artefacts such as digital watch, heart beat, as well as eye movements and blinks. The most prominent feature in the area we used from the dataset is the biting artefact. There the muscle activity contaminates many of the channels starting after 1600 samples.

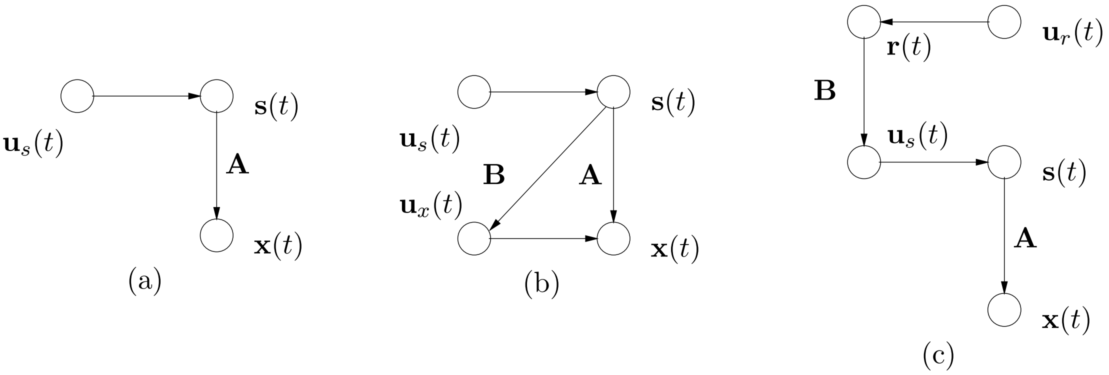

Figure 4.4: Various model structures utilizing variance nodes. Observations are denoted by $\mathbf{x}$, linear mappings by $\mathbf{A}$ and $\mathbf{B}$, sources by $\mathbf{s}$ and $\mathbf{r}$, and variance nodes by $\mathbf{u}$.

Some of the estimated ordinary sources $\mathbf{s}(t)$ are shown in Figure 4.5(a). The variance sources $\mathbf{r}(t)$ that were discovered are shown in Figure 4.5(b). The first variance source clearly models the biting artefact. This variance source integrates information from several conventional sources, and its activity varies very little over time. The second variance source appears to represent increased activity during the onset of the biting, and the third variance source seems to be related to the amount of rhythmic activity on the sources.

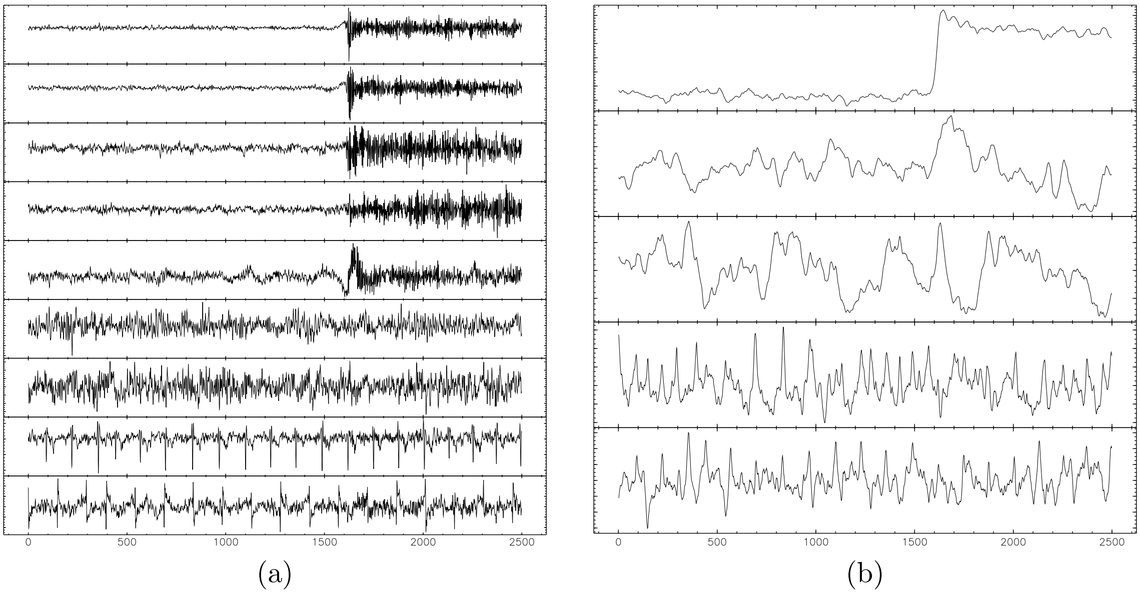

Figure 4.5: (a) Sources $\mathbf{s}(t)$ (nine out of 50) estimated from the data. (d) Variance sources $\mathbf{r}(t)$ which model the regularities found from the variances of the sources [14].

4.4 Nonlinear and non-negative blind source separation

Linear factor analysis (FA) [15] models the data so that it has been generated by sources through a linear mapping with additive noise. Under low noise the method reduces to principal component analysis (PCA). These methods are insensitive to orthogonal rotations of the sources as they only use second order statistics. This can be resolved in the low noise case by independent component analysis (ICA) by assuming independence of the sources and using higher order information [15]. Non-negativity constraints provide an alternative method of resolving the rotation indeterminacy. These methods can be used for blind source separation (BSS) of the sources.

We have applied variational Bayesian learning to nonlinear FA and BSS where the generative mapping from sources to data is not restricted to be linear. The general form of the model is

$$\mathbf{x}(t) = \mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_f) + \mathbf{n}(t)\,. \tag{4.3}$$

This can be viewed as a model about how the observations were generated from the sources. The vectors $\mathbf{x}(t)$ are observations at time t, $\mathbf{s}(t)$ are the sources, and $\mathbf{n}(t)$ the noise. The function $\mathbf{f}(\cdot)$ is a mapping from source space to observation space parametrized by $\boldsymbol{\theta}_f$.

BSS and FA in problems with nonlinear mixing

In an earlier work [6] we have used multi-layer perceptron (MLP) network with tanh-nonlinearities to model the mapping $\mathbf{f}$:

$$\mathbf{f}(\mathbf{s}; \mathbf{A}, \mathbf{B}, \mathbf{a}, \mathbf{b}) = \mathbf{B} \tanh(\mathbf{A}\mathbf{s} + \mathbf{a}) + \mathbf{b}\,. \tag{4.4}$$

The mapping $\mathbf{f}$ is thus parameterized by the matrices $\mathbf{A}$ and $\mathbf{B}$ and bias vectors $\mathbf{a}$ and $\mathbf{b}$. MLP networks are well suited for nonlinear FA and BSS. First, they are universal function approximators which means that any type of nonlinearity can be modeled by them in principle. Second, it is easy to model smooth, nearly linear mappings with them. This makes it possible to learn high dimensional nonlinear representations in practice.

The more accurate linearisation presented in Section 4.2 increases stability of the algorithm in cases with a large number of sources when the posterior variances of the last weak sources are typically large.

Using the MLP network in nonlinear BSS leads to an optimisation problem with many local minima. This makes the method sensitive to initialisation. Originally we have used linear PCA to initialise the posterior means of the sources. This can lead to suboptimal results if the mixing is strongly nonlinear. In [16] nonlinear kernel PCA has been used for initialisation. With a proper choice of the kernel, this can lead to significant improvement in separation results.

An alternative hierarchical nonlinear factor analysis (HNFA) model for nonlinear BSS using the building blocks presented in Section 4.3 was introduced in [17]. HNFA is applicable to larger problems than the MLP based method, as the computational complexity is linear with respect to the number of sources. The efficient pruning facilities of Bayes Blocks also allow determining whether the nonlinearity is really needed and pruning it out when the mixing is linear, as demonstrated in [18].

Post-nonlinear FA and BSS

Our work [20] restricts the general nonlinear mapping in (4.3) to the important case of post-nonlinear (PNL) mixtures. The PNL model consists of a linear mixture followed by component-wise nonlinearities acting on each output independently from the others:

$$x_i(t) = f_i\left[\mathbf{a}_i^T \mathbf{s}(t)\right] + n_i(t) \qquad i = 1, \dots, n \tag{4.5}$$

The vector $\mathbf{a}_i$ in this equation denotes the i:th row of the mixing matrix $\mathbf{A}$. The sources $\mathbf{s}(t)$ are assumed to have Gaussian distributions in our model called *post-nonlinear factor analysis* (PNFA). The sources found with PNFA can be further rotated using any algorithm for linear ICA in order to obtain independent sources.

The development of PNFA was motivated by the comparison experiments [19] where we showed that the existing PNL methods cannot separate globally invertible post-nonlinear mixtures with non-invertible post-nonlinearities. The proposed technique learns the generative model of the observations and therefore it is applicable to such complex PNL mixtures. In [20], we show that PNFA can achive separation of signals in a very challenging BSS problem.

Non-negative BSS by rectified factor analysis

Linear factor models with non-negativity constraints have received a great deal of interest in a number of problem domains. In the variational Bayesian framework, positivity of the factors can be achieved by putting a non-negatively supported prior on the factors. The rectified Gaussian distribution is particularly convenient, as it is conjugate to the Gaussian likelihood arising in the FA model. Unfortunately, this solution has a serious technical limitation: it includes in practice the assumption that the factors have sparse distributions, meaning that the probability mass is concentrated near zero. This is because the location parameter of the prior has to be fixed to zero; otherwise the potentials arising both to the location and to the scale parameter become very awkward.

A way to circumvent the above mentioned problems is to reformulate the model using rectification nonlinearities. This can be expressed in the formalism of Eq. (4.3) using the following nonlinearity

$$\mathbf{f}(\mathbf{s};\, \mathbf{A}) = \mathbf{A}\,\mathbf{cut}(\mathbf{s}) \tag{4.6}$$

where **cut** is the componentwise rectification (or cut) function such that $[\mathbf{cut}(\mathbf{s})]_i = \max(s_i, 0)$. In [21], a variational learning procedure was derived for the proposed model and it was shown that it indeed overcomes the problems that exist with the related approaches. In Section 4.7 an application of the method to the analysis of galaxy spectra is presented.

4.5 Dynamic modelling using nonlinear state-space models

Nonlinear state-space models

In many cases, measurements originate from a dynamical system and form time series. In such cases, it is often useful to model the dynamics in addition to the instantaneous observations. We have extended the nonlinear factor analysis model by adding a nonlinear model for the dynamics of the sources $\mathbf{s}(t)$ [7]. This results in a state-space model where the sources can be interpreted as the internal state of the underlying generative process.

The nonlinear static model of Eq. (4.3) can easily be extended to a dynamic one by adding another nonlinear mapping to model the dynamics. This leads to source model

$$\mathbf{s}(t) = \mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_g) + \mathbf{m}(t)\,, \tag{4.7}$$

where $\mathbf{s}(t)$ are the sources (states), $\mathbf{m}$ is the Gaussian noise, and $\mathbf{g}(\cdot)$ is a vector containing as its elements the nonlinear functions modelling the dynamics.

As in nonlinear factor analysis, the nonlinear functions are modelled by MLP networks. The mapping $\mathbf{f}$ has the same functional form (4.4). Since the states in dynamical systems are often slowly changing, the MLP network for mapping $\mathbf{g}$ models the change in the value of the source:

$$\mathbf{g}(\mathbf{s}(\mathbf{t-1})) = \mathbf{s}(t-1) + \mathbf{D}\tanh[\mathbf{C}\mathbf{s}(t-1) + \mathbf{c}] + \mathbf{d}\,. \tag{4.8}$$

An important advantage of the proposed method is its ability to learn a high-dimensional latent source space. We have also reasonably solved computational and overfitting problems which have been major obstacles in developing this kind of unsupervised methods thus far. Potential applications for our method include prediction and process monitoring, control and identification.

Detection of process state changes

One potential application for the nonlinear state-space model is process monitoring. In [22], variational Bayesian learning was shown to be able to learn a model which is capable of detecting an abrupt change in the underlying dynamics of a fairly complex nonlinear process. The process was artificially generated by nonlinearly mixing some of the states of three independent dynamical systems: two independent Lorenz processes and one harmonic oscillator. The nonlinear dynamic model was first estimated off-line using 1000 samples of the observed process. The model was then fixed and applied on-line to new observations with artificially generated changes of the dynamics.

Figures 4.6 and 4.7 show an experiment with a change generated at time instant T_{ch}, when the underlying dynamics of one of the Lorenz processes abruptly changes. The change detection method based on the estimated model readily detects the change raising an alarm after the time of change. The method is also able to find out in which states the change occurred (see Fig. 4.7) as the reason for the detected change can be localised by analysing the structure of the cost function.

Stochastic nonlinear model-predictive control

For being able to control the dynamical system, control inputs are added to the nonlinear state-space model. In [23], we study three different control schemes in this setting. Direct control is based on using the internal forward model directly. It is fast to use, but requires the learning of a policy mapping, which is hard to do well. Optimistic inference control is a novel method based on Bayesian inference answering the question: “Assuming success

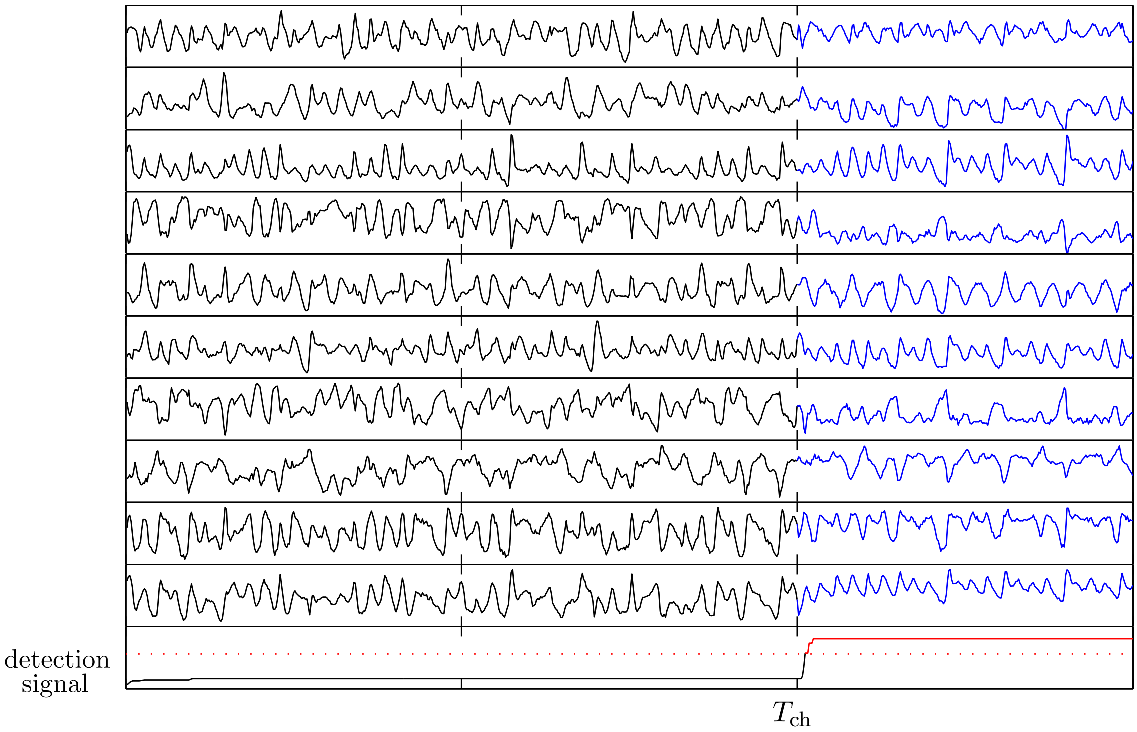

Figure 4.6: The monitored process (10 time series above) with the change simulated at T_{ch}. The change has been detected using the estimated model, the alarm signal is shown below.

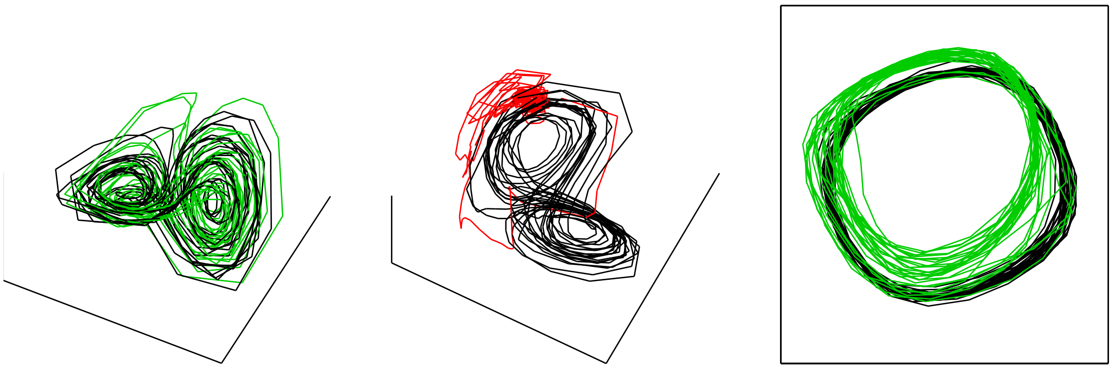

Figure 4.7: The estimated process states reconstructing the two original Lorenz processes and harmonic oscillator. The values after T_{ch} are shown as coloured curves. The cost contribution of the second process drastically changes after the time of change, which is used to localise the reason of the change.

in the end, what will happen in near future?" It is based on a single probabilistic inference but unfortunately neither of the two tested inference algorithms works well with it. The third control scheme is stochastic nonlinear model-predictive control, which is based on optimising control signals based on maximising a utility function.

Figure 4.8 shows simulations with a cart-pole swing-up task. The results confirm that selecting actions based on a state-space model instead of the observation directly has many benefits: First, it is more resistant to noise because it implicitly involves filtering. Second,

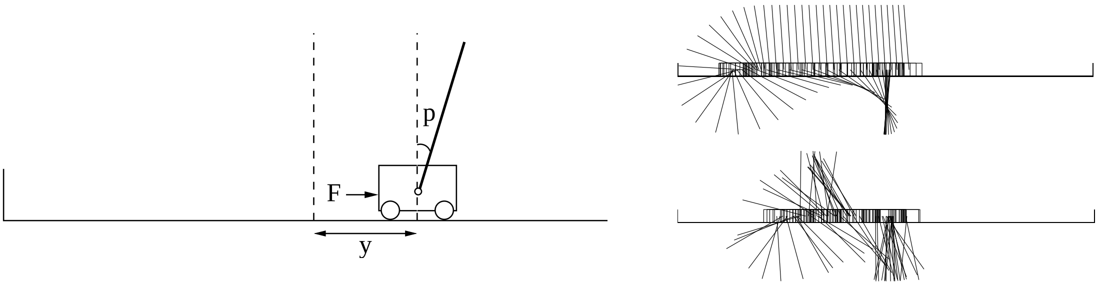

Figure 4.8: Left: the cart-pole system. The goal is to swing the pole to an upward position and stabilise it without hitting the walls. The cart can be controlled by applying a force to it. Top left: the pole is successfully swinged up by moving first to the left and then right. Bottom right: our controller works quite reliably even in the presence of serious observation noise [23].

the observations (without history) do not always carry enough information about the system state. Third, when nonlinear dynamics are modelled by a function approximator such as an multilayer perceptron network, a state-space model can find such a representation of the state that it is more suitable for the approximation and thus more predictable [23].

4.6 Relational models

Formerly, we have divided our models into two categories: static and dynamic. In static modelling, each observation or data sample is independent of the others. In dynamic models, the dependencies between consecutive observations are modelled. The generalisation of both is that the relations are described in the data itself, that is, each observation might have a different structure.

Many models have been developed for relational discrete data, and for data with nonlinear dependencies between continuous values. In [24], we combine two of these methods, relational Markov networks and hierarchical nonlinear factor analysis, resulting in using nonlinear models in a structure determined by the relations in the data. Experimental setup in the board game Go is depicted in Figure 4.9. The task is the collective regression of survival probabilities of blocks. The results suggest that regression accuracy can be improved by taking into account both relations and nonlinearities.

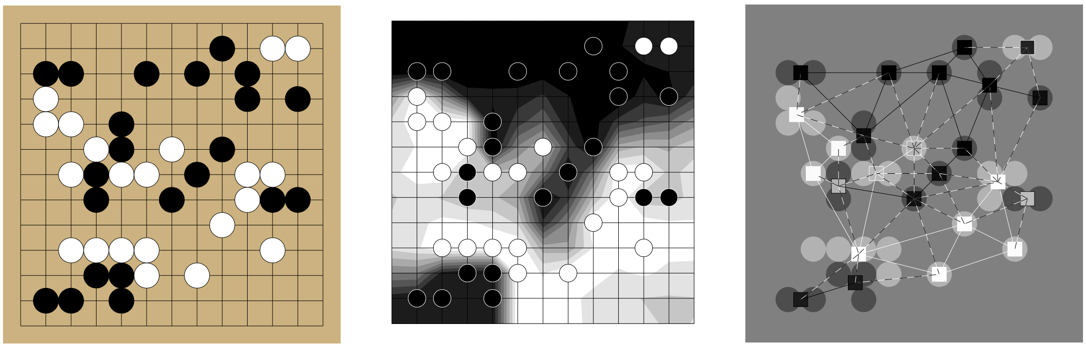

Figure 4.9: The leftmost subfigure shows the board of a Go game in progress. In the middle, the expected owner of each point is visualised with the shade of grey. For instance, the two white stones in the upper right corner are very likely to be captured. The rightmost subfigure shows the blocks with their expected owner as the colour of the square. Pairs of related blocks are connected with a line which is dashed when the blocks are of opposing colours.

Many real world sequences such as protein secondary structures or shell logs exhibit rich internal structures. Logical hidden Markov models have been proposed as one solution. They deal with logical sequences, i.e., sequences over an alphabet of logical atoms. This comes at the expense of a more complex model selection problem. Indeed, different abstraction levels have to be explored. In [25], we propose a novel method for selecting logical hidden Markov models from data called sagEM. sagEM combines generalized expectation maximization, which optimizes parameters, with structure search for model selection using inductive logic programming refinement operators. We provide convergence and experimental results that show sagEM's effectiveness.

4.7 Applications to astronomy

We have applied rectified factor analysis [21] described in Section 4.4 to the analysis of real stellar population spectra of elliptical galaxies. Ellipticals are the oldest galactic systems in the local universe and are well studied in physics. The hypothesis that some of these old galactic systems may actually contain young components is relatively new. Hence, we have investigated whether a set of stellar population spectra can be decomposed and explained in terms of a small set of unobserved spectral prototypes in a data driven but yet physically meaningful manner. The positivity constraint is important in this modelling application, as negative values of flux would not be physically interpretable.

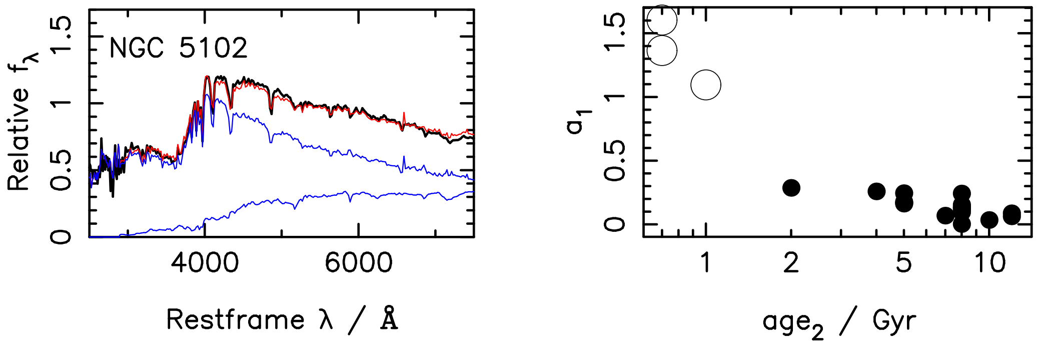

Figure 4.10: Left: the spectrum of a galaxy with its decomposition to a young and old component. Right: the age of the dominating stellar population against the mixing coefficient of the young component.

Using a set of 21 real stellar population spectra, we found that they can indeed be decomposed to prototypical spectra, especially to a young and old component [26]. Figure 4.10 shows one spectrum and its decomposition to these two components. The right subfigure shows the ages of the galaxies, known from a detailed astrophysical analysis, plotted against the first weight of the mixing matrix. The plot clearly shows that the first component corresponds to a galaxy containing a significant young stellar population.

References

[1] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press, 2003.

[2] H. Lappalainen and J. Miskin. Ensemble learning. In M. Girolami, editor, *Advances in Independent Component Analysis*, Springer, 2000, pages 75–92.

[3] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models* MIT Press, 1999, pages 105–161.

[4] A. Honkela and H. Valpola. Variational learning and bits-back coding: an information-theoretic view to Bayesian learning. *IEEE Transactions on Neural Networks*, 15(4):267–282, 2004.

[5] A. Ilin, and H. Valpola. On the effect of the form of the posterior approximation in variational learning of ICA models. *Neural Processing Letters*, 22(2):183–204, 2005.

[6] H. Lappalainen and A. Honkela. Bayesian nonlinear independent component analysis by multi-layer perceptrons. In Mark Girolami, editor, *Advances in Independent Component Analysis*, pages 93–121. Springer-Verlag, Berlin, 2000.

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

[8] A. Honkela. Approximating nonlinear transformations of probability distributions for nonlinear independent component analysis. In *Proc. 2004 IEEE Int. Joint Conf. on Neural Networks (IJCNN 2004)*, pages 2169–2174, Budapest, Hungary, 2004.

[9] A. Honkela and H. Valpola. Unsupervised variational Bayesian learning of nonlinear models. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 593–600. The MIT Press, Cambridge, MA, USA, 2005.

[10] T. Raiko. Partially observed values. In *Proc. 2004 IEEE Int. Joint Conf. on Neural Networks (IJCNN 2004)*, pages 2825–2830, Budapest, Hungary, 2004.

[11] H. Valpola, A. Honkela, M. Harva, A. Ilin, T. Raiko, and T. Östman. Bayes Blocks software library. `http://www.cis.hut.fi/projects/bayes/software/`, 2003.

[12] H. Valpola, T. Raiko, and J. Karhunen. Building blocks for hierarchical latent variable models. In *Proc. 3rd Int. Workshop on Independent Component Analysis and Signal Separation (ICA2001)*, pages 710–715, San Diego, California, December 2001.

[13] M. Harva, T. Raiko, A. Honkela, H. Valpola, and J. Karhunen. Bayes Blocks: An implementation of the variational Bayesian building blocks framework. In *Proc. 21st Conference on Uncertainty in Artificial Intelligence*, pages 259–266. Edinburgh, Scotland, 2005.

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

[15] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Wiley, 2001.

[16] A. Honkela, S. Harmeling, L. Lundqvist and H. Valpola. Using kernel PCA for initialisation of variational Bayesian nonlinear blind source separation method. In *Proc. 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, Vol. 3195 of *Lecture Notes in Computer Science*, Springer-Verlag, pages 790–797, 2004.

[17] H. Valpola, T. Östman, and J. Karhunen. Nonlinear independent factor analysis by hierarchical models. In *Proc. 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 257–262, Nara, Japan, 2003.

[18] A. Honkela, T. Östman, and R. Vigário. Empirical evidence of the linear nature of magnetoencephalograms. In *Proc. 13th European Symp. on Artificial Neural Networks (ESANN 2005)*, pages 285–290, Bruges, Belgium, 2005.

[19] A. Ilin, S. Achard, and C. Jutten. Bayesian versus constrained structure approaches for source separation in post-nonlinear mixtures. In *Proc. of the Int. Joint Conf. on Neural Networks (IJCNN 2004)*, pages 2181–2186, Budapest, Hungary, 2004.

[20] A. Ilin and A. Honkela. Post-nonlinear independent component analysis by variational Bayesian learning. In *Proc. 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation (ICA 2004)*, Vol. 3195 of *Lecture Notes in Computer Science*, Springer-Verlag, pages 766–773, 2004.

[21] M. Harva and A. Kabán. A variational Bayesian method for rectified factor analysis. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN'05)*, pages 185–190. Montreal, Canada, 2005.

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

[23] T. Raiko and M. Tornio. Learning nonlinear state-space models for control. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN'05)*, pages 815–820, Montreal, Canada, 2005.

[24] T. Raiko. Nonlinear relational Markov networks with an application to the game of Go. In Proc. Int. Conf. on Artificial Neural Networks (ICANN 2005), pages 989–996, Warsaw, Poland, September 2005.

[25] K. Kersting and T. Raiko. 'Say EM' for selecting probabilistic models for logical sequences. In *Proc. 21st Conference on Uncertainty in Artificial Intelligence*, pages 300–307, Edinburgh, Scotland, July 2005.

[26] L. Nolan, M. Harva, A. Kabán, and S. Raychaudhury. A data-driven Bayesian approach for finding young stellar populations in early-type galaxies from their UV-optical spectra. *Monthly Notices of the Royal Astronomical Society*, 366(1):321–338, 2006.